

TUGAS AKHIR
MEMBANGUN SISTEM APLIKASI PENGISIAN KUISIONER
SKALA LIKERT BESERTA INTERPRETASI

(Diajukan Sebagai Salah Satu Syarat Untuk Memperoleh Gelar Sarjana Statistika)



Disusun oleh :

HARRY DJANUAR

No. Mhs : 00 611 013

JURUSAN STATISTIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS ISLAM INDONESIA
JOGJAKARTA
2005

LEMBAR PENGESAHAN PEMBIMBING
MEMBANGUN SISTEM APLIKASI PENGISIAN KUISIONER
SKALA LIKERT BESERTA INTERPRETASI

TUGAS AKHIR

Oleh :

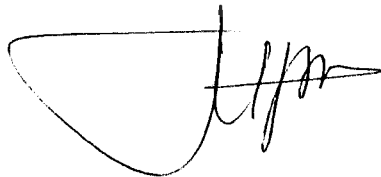
Nama : Harry DJanuar

No Mhs : 00 611 013

Skripsi ini telah disahkan dan disetujui untuk di uji
pada tanggal 8 April 2005

Mengetahui,

Dosen Pembimbing I



(Drs. Supriyono, M.Sc)

Dosen Pembimbing II



(Jaka Nugraha, M.Si)

LEMBAR PENGESAHAN PENGUJI
MEMBANGUN SISTEM APLIKASI PENGISIAN KUISIONER
SKALA LIKERT BESERTA INTERPRETASI
2005

TUGAS AKHIR

Oleh :

Nama : Harry DJanuar

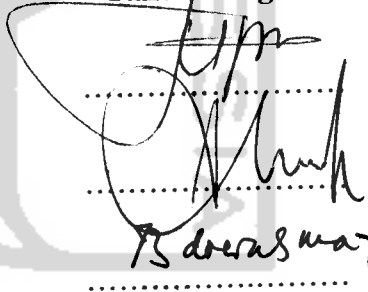
No. Mhs : 00 611 013

Telah Dipertahankan Dihadapan Tim Penguji Sebagai Salah Satu Syarat Untuk
Memperoleh Gelar Sarjana Jurusan Statistika
Fakultas Matematika Dan Ilmu Pengetahuan Alam
Universitas Islam Indonesia

Tim Penguji :


- **Drs. Supriyono, M.Sc**
- **Jaka Nugraha, M.Si**
- **Abdurrahman, M.Si**

Tanda Tangan


.....
.....
.....
.....

Mengetahui,

Dekan Fakultas Matematika Dan Ilmu Pengetahuan Alam

Universitas Islam Indonesia

(Jaka Nugraha, M.Si)

MOTTO

“Jika Allah menolongmu, tiadalah orang yang dapat mengalahkanmu”.

(Ali Imran 160)

“Sesungguhnya disamping kesukaran terdapat kemudahan”

(Al Insyirah 5)

Orang berilmu lebih utama daripada orang yang selalu berpuasa, bersholat, dan berjihad. Apabila mati orang yang berilmu, maka terdapat suatu kekosongan selain oleh penggantinya (yang berilmu juga).

(Khalifah Ali bin Abi Thalib)

Orang tidak banyak ditentukan oleh apa yang dimilikinya ketika lahir, melainkan apa yang ia perbuat atas dirinya sendiri.

(Alexander Graham Bell)

Saya percaya bukan pada apa yang saya pikir, tetapi apa yang saya tahu.

(Sir Arthur Conan Doyle)

PERSEMBAHAN

Dengan penuh kesabaran, ketekunan & keikhlasan hati
kupersembahkan hasil karyaku ini kepada:

- Allah SWT, yang telah memberikan taufiq, hidayah, keridhoan, serta lindungan-Nya sehingga tugas akhir ini dapat diselesaikan.
- Ayah dan bundaku yang tercinta yang telah memberikan semuanya, cinta, kasih sayang, do'a, spirit, nasehat, uang dan pelajaran yang tak ternilai harganya.
- Henny yang maniez yang selalu kakak banggakan dan selalu menjadi prioritas kakak.
- Adik - adikku yang kusayangi yang telah memberikan dorongan dan dukungan untukku, kakak selalu mendukung kalian.
- Semua keluarga baik keluarga Ayah maupun keluarga Ibu, yang telah memberikan do'a dukungan, semangat, perhatian selama ini.

KATA PENGANTAR



Assalamu'alaikum Wr. Wb.

Puji syukur Alhamdulillah kami panjatkan atas kehadiran Allah Swt, yang telah memberikan rahmat dan hidayahnya, serta kesehatan dan kemudahan, sehingga penelitian dan penyusunan Tugas Akhir ini dapat terselesaikan. Tak lupa shalawat beriring salam semoga tercurah kepada Nabi Muhammad SAW beserta keluarganya, sahabat-sahabatnya dan kepada kita semua. Amin.

Penulisan Tugas Akhir yang berjudul **MEMBANGUN SISTEM APLIKASI PENGISIAN KUISIONER SKALA LIKERT BESERTA INTERPRETASI** dilaksanakan sebagai salah satu syarat untuk memperoleh gelar sarjana pada jurusan Statistika.

Pada kesempatan ini penyusun menyampaikan terima kasih atas terselesaikannya skripsi ini kepada yang terhormat :

1. Bapak Supriyono, M.Sc, sebagai pembimbing I yang telah membimbing penyusun dari awal hingga akhir.
2. Bapak Jaka Nugraha, M.Si, selaku Dekan Fakultas Matematika Dan Ilmu Pengetahuan Alam dan juga sebagai pembimbing II.
3. Ibu.Rohmatul Fajriyah, M.Si, selaku Ketua Jurusan Statistika.
4. Ayah dan Ibu yang selalu memberi spirit dan kasih sayangnya.
5. Henny yang selalu buat masalah, tapi kakak akan selalu dibelakangmu.
6. Adik-adikku yang maniez2, makasih atas supportnya dan do'anya.

7. *And Always Remember* U2, Jenggot, Refi, Tobrih, Andi. Kalian sobatku, makasih atas kebersamaan selama ini. Jangan lupakan masa kuliah.
8. Teman-teman kost *BrownStone* n *eks-BrownStone*, mas dan mba' Prie, Yhokers, By-use, Magath, Wisnu, Andre, Ebiet, Guswan, Udin, duo_Agung n Junior2 (Thanks for be relation patners in the kost). Special thank's to Ican (thank's for private Delphi).
9. Teman-teman *Komunitas Statistika* Fajar, Radja, Waming, Hamazi, Irfan, Sigit, Alif, Anggun, Renita, Indo, Agus, Anita n Smua yang kenal ma DJ.
10. Teman-teman seperjuangan Statistik khususnya anak-anak angkatan 2000, angkat nama Statistika.
11. Dan semua pihak yang telah membantu penyusunan ini sehingga berjalan dengan baik.

Penyusun menyadari sepenuhnya bahwa Skripsi ini masih jauh dari kesempurnaan. Oleh karena itu saran dan kritik guna penyempurnaan skripsi ini sangat peyusun harapkan dari semua pihak. Akhir kata penyusun berharap semoga skripsi ini akan dapat bermanfaat bagi semua pihak yang membutuhkan dimasa yang akan datang. Amin.

Wassalamu'alaikum Wr. Wb.

Yogyakarta, 25 April 2005,

Penyusun

DAFTAR ISI

HALAMAN JUDUL	i
LEMBAR PENGESAHAN DOSEN PEMBIMBING	ii
LEMBAR PENGESAHAN DOSEN PENGUJI	iii
HALAMAN MOTTO	iv
HALAMAN PERSEMBAHAN	v
KATA PENGANTAR	vi
DAFTAR ISI	viii
DAFTAR TABEL	xii
DAFTAR GAMBAR	xiii
ABSTRAKSI	xv
BAB I PENDAHULUAN	1
1.1 Latar Belakang Masalah.....	1
1.2 Rumusan Masalah	3
1.3 Batasan Masalah.....	3
1.4 Tujuan Penelitian.....	4
1.5 Manfaat Penelitian.....	4
1.6 Metodologi penelitian	5
1.7 Sistematika Penulisan.....	6
BAB II LANDASAN TEORI	9
2.1 Sistem	9
2.1.1 Karakteristik Sistem	10
2.1.2 Klasifikasi Sistem.....	11
2.2 Konsep Dasar Informasi.....	13
2.3 Konsep Basis Data	14
2.3.1 Istilah Dalam Basis Data	16
2.3.2 Operasi Dasar Basis Data	18
2.3.3 Kegunaan Basis Data/Syarat Basis Data.....	19
2.4 Diagram Konteks.....	20
2.5 Data Flow Diagram	21

2.6 Relasi Antar Tabel.....	22
2.7 Borland Delphi 7	23
2.8 Adobe Photoshop CS	24
2.9 SWISHmax.....	25
2.10 MySQL Database	25
2.11 Jenis dan Analisis Data	26
2.12 Metode Analisis Data.....	28
BAB III ANALISIS KEBUTUHAN PERANGKAT LUNAK	31
3.1 Metode Analisis.....	31
3.2 Hasil Analisis	31
3.3 Perangkat Keras.....	33
3.4 Pembuatan Sistem Perangkat Lunak	33
3.4.1 Perancangan Perangkat Lunak dan Interface	33
3.4.2 Pembuatan Perangkat Lunak dan Grafis Antar Muka.....	34
3.5 Implementasi Perangkat Lunak Statistika.....	34
BAB IV PERANCANGAN PERANGKAT LUNAK.....	36
4.1 Metode Perancangan	36
4.2 Hasil Perancangan	36
4.2.1 Diagram Konteks.....	36
4.2.2 Diagram Arus Data (Data Flow Diagram)	37
4.2.2.1 DFD Level 1.....	37
4.2.3 Relasi Antar Tabel.....	39
4.2.4 Desain Basis Data.....	40
4.2.5 Perancangan Bagian Antarmuka	43
4.2.5.1 Perancangan Tampilan Awal (Operator).....	43
4.2.5.2 Perancangan Administrasi.....	44
4.2.5.3 Perancangan Ubah Password	44
4.2.5.4 Perancangan Konfigurasi Kuisisioner.....	45
4.2.5.5 Perancangan Informasi Responden	46
4.2.5.6 Perancangan Frekuensi Data Kuisisioner	46
4.2.5.7 Perancangan Interpretasi Kuisisioner.....	47
4.2.5.8 Perancangan Tampilan Awal (Responden).....	48

4.2.5.9 Perancangan Tampilan Kuisisioner (Responden).....	49
4.2.5.10 Perancangan Tampilan Pembuat	50
BAB V IMPLEMENTASI PERANGKAT LUNAK	51
5.1 Implementasi Secara Umum	51
5.2 Alasan Pemilihan Perangkat Lunak	51
5.3 Batasan Implementasi	52
5.4 Tahapan Pembuatan Proqram.....	52
5.5 Implementasi Perangkat Lunak	53
5.6 Pembuatan Program	53
5.6.1 Admin dan Operator	53
5.6.1.1 Menu Utama	53
5.6.1.2 Login	54
5.6.1.3 Ubah Password.....	55
5.6.1.4 Administrasi	56
5.6.1.5 Konfigurasi Kuisisioner	57
5.6.1.6 Informasi Responden.....	60
5.6.1.7 Frekuensi Data Kuisisioner	64
5.6.1.8 Interpretasi Kuisisioner	65
5.6.2 Responden	68
5.6.2.1 Tampilan Awal	68
5.6.2.2 Kuisisioner	70
5.6.3 Tampilan Pembuat	72
BAB VI ANALISIS KINERJA PERANGKAT LUNAK	73
6.1 Pengujian Program	73
6.2 Pengujian Dan Analisis	73
6.2.1 Uji Normal.....	73
6.2.2 Uji Tidak Normal	81
6.3 Hasil Keluaran Sistem	84
6.4 Hasil Pengujian	85
6.5 Resume Pengujian Sistem	86
BAB VII PENUTUP	87
7.1 Kesimpulan	87

7.2 Saran	87
DAFTAR PUSTAKA	89
LAMPIRAN	



DAFTAR TABEL

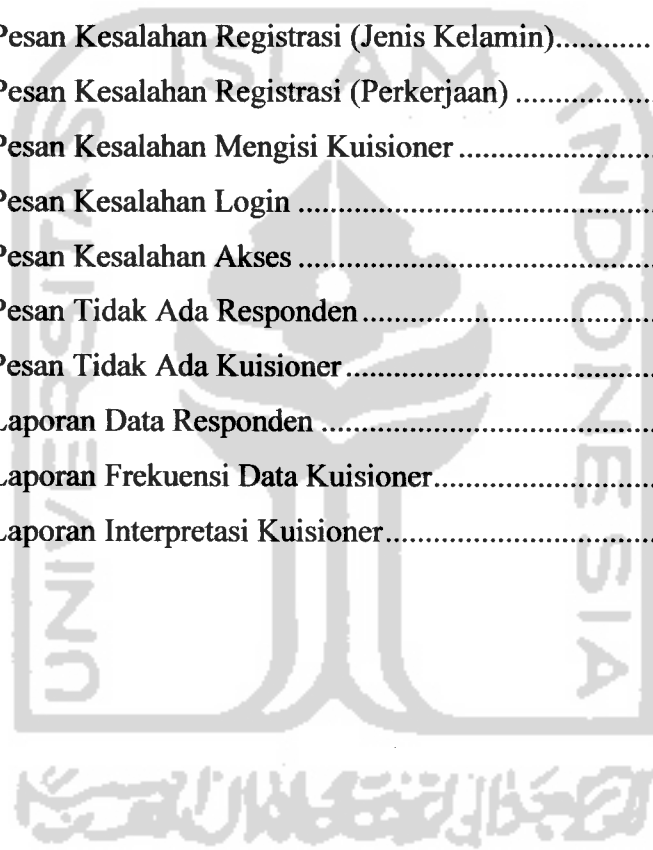
Tabel 4.1 Tabel Data User.....	40
Tabel 4.2 Tabel Informasi	40
Tabel 4.3 Tabel Topik	41
Tabel 4.4 Tabel Pertanyaan.....	41
Tabel 4.5 Tabel Perkerjaan.....	42
Tabel 4.6 Tabel Responden.....	42
Tabel 4.7 Tabel Nilai.....	43



DAFTAR GAMBAR

Gambar 4.1	Diagram Konteks.....	37
Gambar 4.2	DFD Level 1.....	38
Gambar 4.3	Relasi Antar Tabel.....	39
Gambar 4.4	Rancangan Tampilan Awal.....	43
Gambar 4.5	Rancangan Tampilan Administrasi.....	44
Gambar 4.6	Rancangan Tampilan Ubah Password.....	44
Gambar 4.7	Rancangan Tampilan Konfigurasi Kuisisioner.....	45
Gambar 4.8	Rancangan Tampilan Informasi Responden.....	46
Gambar 4.9	Rancangan Tampilan Frekuensi Data Kuisisioner.....	47
Gambar 4.10	Rancangan Tampilan Interpretasi Kuisisioner.....	48
Gambar 4.11	Rancangan Tampilan Awal (Responden).....	49
Gambar 4.12	Rancangan Tampilan Kuisisioner (Responden).....	49
Gambar 4.13	Rancangan Tampilan Pembuat.....	50
Gambar 5.1	Tampilan Menu Utama.....	54
Gambar 5.2	Tampilan Ubah Password.....	56
Gambar 5.3	Tampilan Administrasi.....	57
Gambar 5.4	Tampilan Edit Topik.....	58
Gambar 5.5	Tampilan Edit Pertanyaan.....	59
Gambar 5.6	Tampilan Edit Informasi.....	60
Gambar 5.7	Tampilan Konfigurasi Kuisisioner.....	60
Gambar 5.8	Tampilan Informasi Responden.....	63
Gambar 5.9	Tampilan Frekuensi Data Kuisisioner.....	65
Gambar 5.10	Tampilan Interpretasi Kuisisioner.....	68
Gambar 5.11	Tampilan Tampilan Awal (Responden).....	70
Gambar 5.12	Tampilan Kuisisioner.....	72
Gambar 5.13	Tampilan Pembuat.....	72
Gambar 6.1	Menu Login Admin.....	74
Gambar 6.2	Menu Input Topik.....	75
Gambar 6.3	Menu Input Pertanyaan.....	75

Gambar 6.4	Menu Input Informasi	76
Gambar 6.5	Menu Data Responden	76
Gambar 6.6	Menu Frekuensi Data Kuisisioner	77
Gambar 6.7	Menu Interpretasi Data Kuisisioner	78
Gambar 6.8	Menu Registrasi Responden	79
Gambar 6.9	Menu Input Kuisisioner	79
Gambar 6.10	Menu Login Operator	80
Gambar 6.11	Pesan Kesalahan Registrasi (Umur)	81
Gambar 6.12	Pesan Kesalahan Registrasi (Jenis Kelamin).....	81
Gambar 6.13	Pesan Kesalahan Registrasi (Perkerjaan)	82
Gambar 6.14	Pesan Kesalahan Mengisi Kuisisioner	82
Gambar 6.15	Pesan Kesalahan Login	82
Gambar 6.16	Pesan Kesalahan Akses	83
Gambar 6.17	Pesan Tidak Ada Responden	83
Gambar 6.18	Pesan Tidak Ada Kuisisioner	84
Gambar 6.19	Laporan Data Responden	84
Gambar 6.20	Laporan Frekuensi Data Kuisisioner.....	85
Gambar 6.21	Laporan Interpretasi Kuisisioner.....	85



TUGAS AKHIR
MEMBANGUN SISTEM APLIKASI PENGISIAN KUISIONER SKALA
LIKERT BESERTA INTERPRETASI

Oleh : Harry Djanuar
Dibawah Bimbingan : Drs. Supriyono, M.Sc dan Jaka Nugraha, M.Si

ABSTRAKSI

Telah dibangun sistem aplikasi pengisian kuisisioner skala likert beserta interpretasi. Sistem ini merupakan salah satu alternatif penyebaran kuisisioner. Sistem ini mencakup konfigurasi kuisisioner yang akan ditampilkan, proses data kuisisioner dan kuisisioner untuk responden. Skala kuisisioner yang digunakan dalam sistem ini adalah skala Likert. Pembuatan sistem ini menggunakan perangkat lunak Borland Delphi 7 dan untuk struktur basis data menggunakan perangkat lunak MySQL. Sedangkan perangkat lunak tambahan yang digunakan ialah Photoshop CS dan SWISHmax. Dalam konfigurasi kuisisioner meliputi edit topik (maksimal 5), edit pertanyaan (maksimal 10 tiap topik) dan edit informasi. Dalam proses data kuisisioner meliputi informasi data responden, frekuensi data kuisisioner dan interpretasi data kuisisioner. Sedangkan kuisisioner untuk responden meliputi registrasi dan pengisian kuisisioner. Hasil keluaran dari sistem aplikasi ini berupa laporan yaitu laporan informasi responden, laporan frekuensi data kuisisioner dan interpretasi data kuisisioner.

Kata-kata kunci : Borland Delphi 7, Basis data, Kuisisioner, Skala Likert, Interpretasi.

DEVELOPING APPLICATION SYSTEM THROUGH FILLING THE QUESTIONER WITH THE INTERPRETATION OF LIKERT SCALE

Author :
Harry Djanuar
00 611 013

ABSTRACT

It already develop the application system on filling the questioner with the likert scale interpretation. This system is one of the alternative on speading the questioner. Actually this system is involve the questioner configuration that will be presented, the data questioner process and the respondent questioner. The researcher use the likert scale and use the soft sets of equipment that we call Borland Delphi 7 and for database structure we use MySQL. Meanwhile, the additional soft sets of equipment that we use are Photoshop CS and SWISHmax. In configurating the questioner it involve topik editing (max 5 topik), question editing (max 10 for each topik) and information editing. In questioner data process it involve responden data information, frequency data of konsumen and interpretation questioner data. Meanwhile the respondent questioner involve registration and filling questioner. The output of this application system are use a report, which is the information respondent report, the data questioner report and interpretation data questioner report.

Key Word : Borland Delphi 7, Database, Questioner, Likert Scale, Interpretation.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Saat ini, teknologi komputer beserta aplikasinya mengalami kemajuan yang sangat pesat. Hampir seluruh aspek kehidupan manusia dapat didukung oleh komputer beserta aplikasinya. Dahulu komputer hanyalah sebagai alat penghitung dan pengolah data saja. Namun, kini komputer adalah alat yang serba bisa, cepat dan canggih. Tentunya perkembangan teknologi komputer ini membuat kemudahan, kemajuan bagi kehidupan manusia.

Sejalan dengan perkembangan teknologi informasi dan komputer yang sangat pesat, diantaranya perkembangan kecepatan, ketepatan dan kemampuan kerja yang berkembang secara terus menerus. Kemampuan tersebut tidak hanya terbatas pada aspek perangkat keras (*hardware*) saja, tetapi juga diikuti dengan perangkat lunak (*software*), bahkan sudah banyak perangkat lunak yang dibuat khusus untuk aplikasi-aplikasi tertentu yang dibutuhkan dan digunakan untuk keperluan kerja perusahaan, lembaga pendidikan, instansi, organisasi dan lain-lain. Hal ini menunjukkan suatu indikasi bahwa pemanfaatan komputer dan fasilitas komputer sangat diperlukan diberbagai aspek bidang kehidupan sebagai alat pengambilan keputusan dan untuk memproses suatu data guna menghasilkan informasi yang optimal.

Perkembangan teknologi komputer pun pada intinya berusaha untuk semakin mendayagunakan kemampuan perhitungan diatas, dengan memperbaiki

kinerja otak komputer atau CPU (*Central Processing Unit*) dari mulai teknologi XT yang sudah usang sampai teknologi Pentium IV dewasa ini. Dengan bantuan komputer dari pengolahan data statistika dihasilkan informasi yang relevan menjadi lebih cepat dan akurat, sesuatu yang dibutuhkan bagi para pengambil keputusan, karena informasi tepat tapi lambat tersajinya akan menjadi basi, sedang informasi yang cepat namun tidak akurat akan menghasilkan yang bisa salah.

Seiring dengan kemajuan perangkat keras (*hardware*) komputer, tidak kalah pesat perkembangan perangkat lunak (*software*) komputer, termasuk perangkat lunak dalam pengolahan data statistika seperti Microstat, SAS, MicroTSP, MINITAB, SPSS, MAPLE dan lainnya. Perangkat lunak tersebut merupakan perangkat khusus yang hanya menyajikan pengolahan data statistika dan grafik yang relevan untuk membantu pengambilan keputusan di bidang statistika.

Adapun perangkat lunak lainnya yang juga bisa digunakan dalam perhitungan statistika, karena pada intinya sama melakukan perhitungan-perhitungan matematis. Salah satu yang ingin diungkap disini adalah perangkat lunak Borland Delphi. BORLAND Delphi merupakan salah satu perangkat lunak (*software*) yang bersifat fleksibel dan umum. Program ini bisa digunakan oleh beberapa kalangan akademisi atau ilmuwan dalam membantu pekerjaan sehari-hari baik diinstansi pendidikan maupun industri. Pemanfaatan program ini membantu menyelesaikan berbagai permasalahan, baik dibidang pendidikan, keilmuan dan lain-lain.

Adapun pustaka-pustaka yang akan dipelajari dalam penelitian ini adalah pustaka tentang pengisian kuisioner. Persoalan yang diselidiki adalah bahwa tidak

adanya perangkat lunak yang menyediakan media komunikasi yang baik antara pengguna (*user*) program statistika dengan Borland Delphi secara langsung dan mudah dalam pengisian kuisioner. Hal inilah yang menjadi acuan peneliti dalam membuat media komunikasi khusus yang lebih mudah dalam pengisian kuisioner dengan memanfaatkan perangkat lunak.

1.2 Rumusan Masalah

Permasalahan yang dijadikan objek penelitian dan pengembangan tugas akhir ini adalah bagaimana cara membuat sistem aplikasi kuisioner yang dapat digunakan oleh operator dan responden dengan mudah serta dapat diolah menjadi suatu informasi yang dapat menjadi acuan untuk analisis data kuisioner selanjutnya.

1.3 Batasan Masalah

Untuk menghindari meluasnya materi pembahasan tugas akhir ini, maka penulis membatasi perangkat lunak yang dikembangkan hanya dengan mencakup hal-hal berikut:

1. Membangun perangkat lunak Statistika dimulai dari tampilan awal program sampai tampilan fungsi statistika. Penjelasan tampilan meliputi cara pemasukkan data (*input data*) dan analisis terhadap *output* yang dihasilkan.
2. Sistem yang dibangun hanya memberikan fasilitas :
 - a. Mengedit kuisioner yang akan diberikan ke responden.
 - b. Informasi mengenai kuisioner yang diisi oleh responden.

- c. Pengisian kuisisioner bisa dilakukan secara jaringan.
3. Aplikasi yang dibuat hanya terbatas pada Sistem Operasi Microsoft Windows.
4. Menggunakan Borland Delphi 7.0 dan MySQL versi 2.2 sebagai software *database* untuk membangun sistem.
5. Skala yang digunakan dalam kuisisioner ialah skala Likert.

1.4 Tujuan Penelitian

Yang menjadi tujuan penelitian ini berdasarkan dari latar belakang permasalahan yang ada adalah :

1. Pembuatan sistem aplikasi pengisian kuisisioner dengan memanfaatkan pustaka-pustaka Borland Delphi.
2. Mengembangkan sistem aplikasi pengisian kuisisioner yang telah ada dan diharapkan pengembangan yang telah dilakukan terhadap sistem sebelumnya dapat melengkapi jenis aplikasi pengisian kuisisioner elektronik.

1.5 Manfaat Penelitian

Manfaat yang ingin dicapai dalam pembuatan tugas akhir ini adalah:

1. Menghasilkan aplikasi yang mudah, cepat dan diharapkan dapat membantu untuk meningkatkan efektifitas dalam pengisian kuisisioner.
2. Penelitian yang dibuat dapat menjadi acuan bagi pengembangan sistem aplikasi yang lebih baik dan lebih lengkap.

1.6 Metodologi Penelitian

Adapun metodologi yang digunakan untuk membangun Sistem Aplikasi Pengisian Kuisioner Skala Likert Beserta Interpretasi seperti terurai di atas adalah sebagai berikut:

a. Pengumpulan Data

Pengumpulan data merupakan metode yang digunakan untuk mengumpulkan data-data yang diperlukan dalam penelitian dengan metode studi pustaka yaitu mencari data-data yang dibutuhkan dari buku-buku atau literatur yang berkaitan dengan permasalahan yang dibahas.

b. Metode Pengembangan Sistem

Dari data yang telah didapat kemudian dianalisis, dirancang kemudian diimplementasikan dengan tahap-tahap sebagai berikut:

1. Analisis Kebutuhan

Tahap analisis bertujuan untuk mengetahui dan menerjemahkan permasalahan serta kebutuhan perangkat lunak untuk membangun Sistem Aplikasi Pengisian Kuisioner Skala Likert Beserta Interpretasi.

2. Perancangan

Tahap perancangan sistem terdiri dari beberapa tahap yaitu: perancangan antarmuka (*interface*) yang berfungsi untuk memberikan gambaran mengenai perangkat lunak yang dibangun

agar *user friendly* serta permodelan sistem. Perancangan yang dibuat meliputi :

- Perancangan Tampilan Awal (Operator)
- Perancangan Administrasi
- Perancangan Ubah Password
- Perancangan Konfigurasi Kuisisioner
- Perancangan Informasi Responden
- Perancangan Frekuensi Data Kuisisioner
- Perancangan Interpretasi Kuisisioner
- Perancangan Tampilan Awal (Responden)
- Perancangan Tampilan Kuisisioner (Responden)
- Perancangan Tampilan Pembuat

3. Implementasi

Setelah tahap perancangan sistem selesai, maka hasil dari perancangan tersebut dapat diimplementasikan.

4. Analisis Kerja

Analisis kerja dilakukan untuk menguji aplikasi yang dikembangkan. Pengujian dilakukan dengan dua tahap yaitu pengujian normal dan pengujian tidak normal.

1.7 Sistematika Penulisan

Untuk mempermudah pembahasan tugas akhir ini maka dalam penyusunannya penulis membagi pokok-pokok permasalahan kedalam tujuh bab sebagai berikut :

BAB I PENDAHULUAN

Bab ini memuat latar belakang masalah, perumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, metodologi penelitian dan sistematika penulisan.

BAB II LANDASAN TEORI

Bab ini membahas dan menguraikan teori-teori tentang kuisisioner, *software* yang digunakan, mekanisme kerja aplikasi pengisian kuisisioner melalui PC dan komponen-komponen yang diperlukan untuk membuat aplikasi kuisisioner.

BAB III ANALISIS KEBUTUHAN PERANGKAT LUNAK

Bab ini berisi penjelasan tentang kebutuhan perangkat lunak dari Sistem Aplikasi Pengisian Kuisisioner Skala Likert Beserta Interpretasi melalui metode analisis. Dari metode analisis didapatkan hasil analisis kebutuhan perangkat lunak meliputi analisis kebutuhan masukan, analisis kebutuhan keluaran dan analisis kebutuhan antarmuka.

BAB IV PERANCANGAN SISTEM

Metode Perancangan

Metode perancangan memuai uraian tentang metode perancangan sistem yang dipakai dalam aplikasi sistem informasi yang akan dibuat.

Hasil Perancangan

Bagian ini memuat hasil perancangan perangkat lunak yang merupakan terjemahan perangkat lunak, yang meliputi pembuatan Diagram Konteks, Diagram Arus Data (*Data Flow Diagram*), Struktur Tabel, Relasi Antar Tabel dan Perancangan Bagian Antarmuka.

BAB V IMPLEMENTASI PERANGKAT LUNAK

Bab ini membahas batasan implementasi perangkat lunak dan perangkat keras yang digunakan untuk membangun dan mengimplementasikan Sistem Aplikasi Pengisian Kuisisioner Skala Likert Beserta Interpretasi serta hasil implementasi yang telah dirancang sebelumnya.

BAB VI ANALISIS KERJA PERANGKAT LUNAK

Bab ini berisi pengujian terhadap aplikasi yang telah dibangun.

BAB VII PENUTUP

Bab ini berisi tentang kesimpulan dan saran yang dapat diambil berdasarkan penelitian dan diakhiri dengan saran untuk perbaikan Sistem Aplikasi Pengisian Kuisisioner Skala Likert Beserta Interpretasi untuk masa yang akan datang.

BAB II

LANDASAN TEORI

2.1 Sistem

Dalam mendefinisikan sistem terdapat dua kelompok pendekatan, yaitu yang menekankan pada prosedurnya dan yang menekankan pada komponen atau elemennya. Pendekatan sistem yang menekankan pada prosedur, definisi sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran tertentu. Pendekatan ini menekankan pada urutan operasi dalam sistem. Prosedur (*prosedure*) didefinisikan oleh Ricard F. Neuschel sebagai suatu urutan operasi klerikal (tulis menulis), biasanya melibatkan beberapa orang didalam suatu atau lebih departemen, yang diterapkan untuk menjamin penanganan yang seragam dari transaksi-transaksi bisnis yang terjadi. Pendekatan sistem yang lebih menekankan pada elemen atau komponennya, definisi sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu [Jogiyanto HM.1999].

Suatu sistem mempunyai maksud tertentu. Ada yang menyebutkan maksud dari suatu sistem adalah untuk mencapai suatu tujuan (*goal*) dan ada yang menyebutkan untuk mencapai suatu sasaran (*objectives*). Tujuan biasanya dihubungkan dengan ruang lingkup yang lebih luas dan sasaran dalam ruang yang lebih sempit. Seringkali tujuan dan sasaran digunakan bergantian dan tidak dibedakan [Jogiyanto HM.1999].

2.1.1 Karakteristik Sistem

Suatu sistem mempunyai karakteristik atau sifat-sifat tertentu, yaitu mempunyai komponen-komponen (*components*), batas sistem (*boundary*), lingkungan luar sistem (*environments*), penghubung (*interface*), masukan (*input*), keluaran (*output*), pengolah (*process*) dan sasaran (*objectives*) atau tujuan (*goal*) [Jogiyanto HM.1999].

a. Komponen Sistem

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang artinya saling bekerja sama membentuk suatu kesatuan. Komponen-komponen sistem atau elemen- elemen sistem dapat berupa suatu subsistem atau bagian-bagian sistem.

b. Batas Sistem

Batas sistem (*Boundary*) merupakan suatu daerah yang membatasi antara suatu sistem dengan sistem yang lain atau dengan lingkungan luarnya.

c. Lingkungan Luar Sistem

Lingkungan Luar (*Environment*) dari suatu sistem adalah apapun diluar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dapat bersifat menguntungkan dan dapat bersifat merugikan sistem tersebut.

d. Penghubung Sistem

Penghubung (*interface*) merupakan media penghubung antara suatu subsistem dengan subsistem yang lainnya. Melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem yang lainnya.

e. **Masukan Sistem**

Masukan (*input*) adalah energi yang dimasukkan ke dalam sistem. Masukan dapat berupa masukan perawatan (*maintenance input*) dan masukan signal (*signal input*). *Maintenance input* adalah energi yang dimasukkan supaya sistem tersebut dapat beroperasi. *Signal input* adalah energi yang diproses untuk didapatkan keluaran.

f. **Keluaran Sistem**

Keluaran (*output*) adalah hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan. Keluaran dapat merupakan masukan untuk subsistem yang lain atau kepada supra sistem.

g. **Pengolah Sistem**

Suatu sistem dapat mempunyai suatu bagian pengolah yang akan merubah masukan menjadi keluaran.

h. **Sasaran Sistem**

Suatu sistem pasti mempunyai suatu tujuan atau sasaran. Kalau suatu sistem tidak mempunyai sasaran, maka operasi sistem tidak akan ada gunanya. Sasaran dari sistem sangat menentukan sekali masukan yang dibutuhkan sistem dan keluaran yang akan dihasilkan sistem. Suatu sistem dikatakan berhasil apabila mengenai sasaran atau tujuannya.

2.1.2 **Klasifikasi Sistem**

Sistem dapat diklasifikasikan dari beberapa sudut pandangan, diantaranya adalah sebagai berikut ini [Jogiyanto HM.1999].

1. Sistem diklasifikasikan sebagai sistem abstrak (*abstract system*) dan sistem fisik (*physical system*).

Sistem abstrak adalah sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik. Sistem fisik merupakan sistem yang ada secara fisik.

2. Sistem diklasifikasikan sebagai sistem alamiah (*natural system*) dan sistem buatan manusia (*human made system*).

Sistem alamiah adalah sistem yang terjadi melalui proses alam, tidak dibuat secara manusia. Sistem buatan manusia adalah sistem yang dirancang oleh manusia. Sistem buatan manusia yang melibatkan interaksi antara manusia dengan mesin disebut dengan *human-machine system* atau ada yang menyebut *man-machine system*.

3. Sistem diklasifikasikan sebagai sistem tertentu (*deterministic system*) dan sistem tak tentu (*probabilistic system*).

Sistem tertentu beroperasi dengan tingkah laku yang sudah dapat diprediksi. Interaksi diantara bagian-bagiannya dapat dideteksi dengan pasti, sehingga keluaran dari sistem dapat diramalkan. Sistem tak tentu adalah sistem yang kondisi masa depannya tidak dapat diprediksi karena mengandung unsur probabilitas.

4. Sistem diklasifikasikan sebagai sistem tertutup (*closed system*) dan sistem terbuka (*open system*).

Sistem tertutup merupakan sistem yang tidak berhubungan dan tidak terpengaruh dengan lingkungan luarnya. Sistem ini bekerja secara otomatis tanpa adanya turut campur tangan dari pihak luarnya. Secara teoritis sistem tertutup ini ada, tetapi kenyataannya tidak ada sistem yang benar-benar tertutup, yang ada

hanyalah *relatively closed system* (secara relatif tertutup, tidak benar-benar tertutup). Sistem terbuka adalah sistem yang berhubungan dan terpengaruh dengan lingkungan luarnya. Sistem ini menerima masukan dan menghasilkan keluaran untuk lingkungan luar atau subsistem yang lainnya.

2.2 Konsep Dasar Informasi

Informasi adalah data yang telah diproses atau data yang memiliki arti. Sumber informasi adalah data. Data adalah kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata. Kejadian adalah sesuatu yang terjadi pada saat tertentu. Data terdiri dari fakta – fakta dan angka –angka yang relatif tidak berarti bagi pemakai, sehingga perlu diolah menjadi informasi agar dapat lebih berguna bagi pemakai.

Nilai informasi yang ditentukan dari dua hal, yaitu manfaat dan biaya mendapatkan informasi tersebut. Suatu informasi dikatakan bernilai bila manfaatnya lebih efektif dibandingkan dengan biaya mendapatkannya.

Spesialis informasi (*information specialist*) merupakan istilah yang digunakan untuk menggambarkan pegawai perusahaan yang bertanggung jawab penuh dalam mengembangkan dan memelihara sistem informasi. Ada lima golongan utama dalam spesialis informasi, yaitu :

a. Analisis sistem (*system analyst*)

Analisis sistem bekerjasama dengan pemakai mengembangkan sistem baru dan memperbaiki sistem yang ada. Analisis sistem adalah pakar dalam mendefinisikan masalah dan menyiapkan dokumentasi tertulis mengenai cara komputer membantu memecahkan masalah.

b. Pengelola *database* (*database administrator*)

Pengelola *database* bekerjasama dengan pemakai dan analis sistem menciptakan *database* yang berisi data yang diperlukan untuk menghasilkan informasi bagi pemakai.

c. Spesialis jaringan (*network specialist*)

Spesialis jaringan bekerjasama dengan analis sistem dan pemakai membentuk jaringan komunikasi data yang menyatukan komputer-komputer yang digunakan. Spesialis jaringan menggabungkan keahlian bidang komputer dan telekomunikasi.

d. Programmer

Programmer menggunakan dokumentasi yang disiapkan oleh analis sistem untuk membuat kode-kode instruksi yang menyebabkan komputer mengubah data menjadi informasi yang diperlukan oleh pemakai.

e. Operator (*administrator*)

Operator bertugas mengoperasikan peralatan komputer berskala besar seperti komputer *mainframe* dan komputer mini. Operator memantau layar komputer, mengganti ukuran kertas printer, mengelola perpustakaan tape dan *disk storage* serta melakukan tugas-tugas serupa lainnya.

2.3 Konsep Basis Data

Basis data terdiri dari 2 kata, yaitu Basis dan Data. Basis kurang lebih dapat diartikan sebagai markas atau gudang, tempat bersarang atau berkumpul.

Sedangkan Data adalah representasi fakta dunia nyata yang mewakili suatu objek seperti manusia (pegawai, siswa, nasabah, pembeli, pelanggan), barang, hewan, peristiwa, konsep, keadaan, dan sebagainya, yang direkam dalam bentuk angka, huruf, simbol, teks, gambar, bunyi, atau kombinasinya [Fathansyah.1999].

Basis data sendiri dapat didefinisikan dalam berbagai sudut pandang seperti :

- a. Himpunan sekelompok data (arsip) yang saling berhubungan yang diorganisasi sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah.
- b. Kumpulan data yang saling berhubungan yang disimpan secara bersama sedemikian rupa dan tanpa pengulangan (redudansi) yang tidak perlu, untuk memenuhi berbagai kebutuhan.
- c. Kumpulan file/tabel/arsip yang saling berhubungan yang disimpan dalam media penyimpanan elektronik.

Satu hal yang juga harus diperhatikan, bahwa basis data bukan hanya sekedar penyimpanan data secara elektronik (dengan bantuan komputer). Artinya, tidak semua bentuk penyimpanan data secara elektronik bisa disebut basis data. Kita juga dapat menyimpan dokumen berisi data dalam file teks (dengan program pengolah kata), file *spread sheet*, dan lain-lain, tetapi tidak bisa disebut sebagai basis data. Karena di dalamnya tidak ada pemilahan dan pengelompokkan data sesuai jenis/fungsi data, sehingga akan menyulitkan dalam pencarian data kelak. Yang sangat ditekankan dalam basis data adalah pengaturan/pemilahan/pengelompokkan/pengorganisasian data yang akan kita simpan sesuai fungsi/jenisnya. Pemilahan/pengelompokkan/pengorganisasian ini dapat

berbentuk sejumlah file/tabel terpisah atau dalam bentuk pendefinisian kolom-kolom/field-field data dalam setiap file/tabel.

2.3.1 Istilah Dalam Basis Data

Basis data merupakan kumpulan file yang saling berelasi, relasi tersebut biasa ditunjukkan dengan kunci dari tiap file yang ada. Maka untuk mempermudah dalam penggunaannya, basis data menggunakan beberapa istilah seperti [Kristanto.1999]:

a. Entity

Entity adalah orang, tempat kejadian atau konsep yang informasinya direkam. Pada bidang Administrasi Siswa misalnya, entity adalah siswa, buku, pembayaran, nilai test.

b. Attribute

Setiap entity mempunyai attribute atau sebutan untuk mewakili suatu entity. Seorang siswa dapat dilihat dari atributnya, misalnya nama, nomor siswa, alamat, nama orang tua, hobby. Attribute juga disebut sebagai data elemen, data field, data item.

c. Data value (nilai atau isi data)

Data value adalah data aktual atau informasi yang disimpan pada tiap data elemen atau attribute. Attribute nama karyawan menunjukkan tempat dimana informasi data karyawan disimpan, sedang data value adalah Sutrisno, Budiman, merupakan isi data nama karyawan tersebut.

d. **Record/Tuple**

Kumpulan elemen-elemen yang saling berkaitan menginformasikan tentang suatu entity secara lengkap. Satu record mewakili satu data atau informasi tentang seseorang misalnya, nomor karyawan, nama karyawan, alamat, kota, tanggal masuk.

e. **File**

Kumpulan record-record sejenis yang mempunyai panjang elemen yang sama, attribute yang sama, namun berbeda-beda data valuenya.

f. **Database**

Kumpulan file-file yang mempunyai kaitan antara satu file dengan file yang lain sehingga membentuk satu bangunan data untuk menginformasikan satu perusahaan, instansi dalam batasan tertentu. Bila terdapat file yang tidak dapat dipadukan atau dihubungkan dengan file yang lainnya berarti file tersebut bukanlah kelompok dari satu database, ia akan dapat membentuk satu database sendiri.

g. **Database Management System (DBMS)**

Kumpulan file yang saling berkaitan bersama dengan program untuk pengelolaannya disebut sebagai DBMS. Database adalah kumpulan datanya, sedang program pengelolaannya berdiri sendiri dalam satu paket program yang komersial untuk membaca data, mengisi data, menghapus data, melaporkan data dalam database.

2.3.2 Operasi Dasar Basis Data

Operasi-operasi dasar yang berkenaan dengan basis data dapat meliputi [Fathansyah.1999] :

- a. Pembuatan basis data baru (*create database*), yang identik dengan pembuatan lemari arsip yang baru.
- b. Penghapusan basis data (*drop database*), yang identik dengan perusakan lemari arsip (sekaligus beserta isinya jika ada).
- c. Pembuatan file/tabel baru ke suatu basis data (*create table*), yang identik dengan penambahan map arsip baru ke sebuah lemari arsip yang telah ada.
- d. Penghapusan file/tabel dari suatu basis data (*drop table*), yang identik dengan perusakan map arsip lama yang ada di sebuah lemari arsip.
- e. Penambahan/pengisian data baru ke sebuah file/tabel di sebuah basis data (*insert*), yang identik dengan penambahan lembaran arsip ke sebuah map arsip.
- f. Pengambilan data dari sebuah file/tabel (*retrieve/search*), yang identik dengan pencarian lembaran arsip dari sebuah map arsip.
- g. Perubahan data dari sebuah file/tabel (*update*), yang identik dengan perbaikan isi lembaran arsip yang ada di sebuah lemari arsip.
- h. Penghapusan data dari sebuah file/tabel (*delete*), yang identik dengan penghapusan sebuah lembaran arsip yang ada di sebuah map arsip.

Operasi yang berkenaan dengan pembuatan objek (basis data dan tabel) merupakan operasi awal yang hanya dilakukan sekali dan berlaku seterusnya. Sedangkan operasi yang berkaitan dengan isi tabel (data) merupakan operasi rutin yang akan berlangsung berulang-ulang dan karena itu operasi inilah yang lebih

tepat mewakili aktivitas pengelolaan (*management*) dan pengolahan (*processing*) data dalam basis data.

2.3.3 Kegunaan Basis Data/Syarat Basis Data

Penyusunan satu basis data digunakan untuk mengatasi masalah-masalah pada penyusunan data yaitu :

a. Redundansi dan inkonsistensi data

Penyimpanan di beberapa tempat untuk data yang sama ini disebut sebagai redundansi dan mengakibatkan pemborosan ruang penyimpanan dan juga biaya untuk mengakses jadi lebih tinggi. Penyimpanan data yang berulang-ulang di beberapa file dapat mengakibatkan juga inkonsistensi (tidak konsisten).

b. Kesulitan pengaksesan data

Untuk menyelesaikan masalah ini, DBMS yang mampu mengambil data secara langsung dengan bahasa yang familiar dan mudah digunakan (*user friendly*).

c. Isolasi data untuk standarisasi

Jika data tersebar dalam beberapa file dalam bentuk format yang tidak sama, maka ini menyulitkan dalam menulis program aplikasi untuk mengambil dan menyimpan data. Maka haruslah data dalam satu basis data dibuat satu format sehingga mudah dibuat program aplikasinya.

d. *Multiple user* (banyak pemakai)

Dalam rangka mempercepat semua daya guna sistem dan mendapat responsi waktu yang cepat, beberapa sistem memungkinkan banyak pemakai untuk

meng"update" data secara simultan. Salah satu alasan mengapa basis data dibangun karena nantinya data tersebut digunakan oleh banyak orang dalam waktu yang berbeda, diakses oleh program yang sama tapi beda orang dan waktu.

e. Masalah keamanan (*security*)

Keamanan ini dapat diatur lewat program yang dibuat oleh pemrogram atau fasilitas keamanan dari *operating sistem* misalnya Novell Netware untuk Local Area Network.

f. Masalah integrasi (kesatuan)

Basis data berisi file-file yang saling berkaitan, masalah utama adalah bagaimana kaitan antara file tersebut terjadi.

g. Masalah data *independence* (kebebasan data)

Apapun perubahan yang terjadi dalam basis data, semua perintah akan mengalami kestabilan tanpa perlu ada yang diubah.

2.4 Diagram Konteks

Context Diagram (CD) adalah DFD (bagian DFD yang berfungsi memetakan model lingkungan), yang dipresentasikan dengan lingkaran tunggal yang mewakili keseluruhan sistem.

Contexts Diagram menyoroti sejumlah karakteristik penting sistem, yaitu :

- Kelompok pemakai, organisasi atau sistem lain dimana sistem melakukan komunikasi yang disebut sebagai *terminator*.
- Data masuk, data yang diterima sistem dari lingkungan dan harus diproses dengan cara tertentu.

- Data keluar, data yang dihasilkan sistem dan diberikan ke dunia luar.
- Penyimpanan data (*data store*) yang digunakan secara bersama antara sistem dengan *terminator*. Data ini dapat dibuat oleh sistem dan digunakan oleh lingkungan luar atau sebaliknya, dibuat oleh lingkungan dan digunakan oleh sistem.
- Batasan antara sistem dan lingkungan (*rest of the world*).

2.5 Data Flow Diagram

Model ini menggambarkan sistem sebagai jaringan antar fungsi yang berhubungan satu sama lain dengan aliran penyimpanan data. Sebagai perangkat analis, model ini hanya mampu memodelkan sistem dari satu sudut pandang yaitu sudut pandang fungsi. Ada empat komponen dalam model ini, yaitu :

- Proses

Proses ini menunjukkan transformasi dari masukan menjadi keluaran, dalam hal ini sejumlah masukan dapat menjadi hanya satu keluaran ataupun sebaliknya. Proses pada didefinisikan dengan kata tunggal, atau kalimat sederhana. Pada sejumlah kasus definisi ini dapat berupa nama departemen, bagian dalam suatu organisasi, komputer, peralatan elektronik.

- Aliran

Komponen ini dipresentasikan dengan menggunakan panah yang menuju ke atau dari proses. Digunakan untuk menggambarkan paket data atau informasi dari satu bagian ke bagian lain dari sistem dimana penyimpanan mewakili lokasi penyimpanan data.

- Penyimpanan

Komponen ini digunakan untuk memodelkan kumpulan data atau paket data. Notasi yang digunakan adalah sejajar, segiempat dengan sudut melengkung persegi panjang.

- Terminator

Komponen ini dipresentasikan menggunakan persegi panjang yang mewakili *entity* luar dimana sistem berkomunikasi. Notasi ini biasanya melambangkan orang atau sekelompok orang misalnya organisasi diluar sistem, grup, departemen, perusahaan pemerintah dan berada diluar kontrol sistem yang dimodelkan.

2.6 Relasi Antar Tabel

Basis data merupakan kumpulan dari data-data yang saling berhubungan satu dengan yang lainnya. Data-data tersebut disimpan dalam sebuah file atau tabel yang memiliki keterangan tertentu, dan apabila direlasikan akan menghasilkan informasi yang kompleks. Hubungan antar tabel ini direlasikan dengan kunci relasi (relation key) yang merupakan kunci utama dari masing-masing file/tabel. Keberadaan tabel relasi dapat memudahkan dalam pemeliharaan data dan menghindarkan kerangkapan data, sehingga informasi yang diperoleh akan lebih akurat dan meminimalkan pembuatan record. Perencanaan basis data yang tepat akan menyebabkan paket program relational akan bekerja secara optimal.

2.7 Borland Delphi 7

Delphi merupakan program aplikasi database dengan lingkungan programan visual. Memakai delphi adalah cara yang paling sederhana untuk membangun aplikasi berbasis Windows dan visual. Kelebihan Delphi dibandingkan dengan program aplikasi lainnya adalah produktifitasnya. Produktifitas dari pengembangan perangkat lunak dibagi menjadi lima atribut penting, yaitu :

- *Kualitas* dari lingkungan pengembangan visual.
- *Kecepatan* compiler dibandingkan dengan kompleksitasnya.
- *Kekuatan* dari bahasa pemrograman dibandingkan dengan kompleksitasnya.
- *Fleksibilitas* dari arsitektur basis data.
- *Pola desain dan pemakaian* yang diwujudkan oleh frameworknya.

Ada tiga versi dari Delphi :

- *Personal*, untuk versi pemula dan ditujukan untuk pelajar yang ingin mendalami Delphi.
- *Professional*, untuk pengembang aplikasi yang professional, berisi semua kemampuan dasar, pemrograman database, server web dan beberapa alat bantu lainnya.
- *Enterprise*, untuk pengembang aplikasi enterprise, berisi fasilitas XML, arsitektur *three-tier* dan fasilitas lainnya.

Form

Inti dari setiap aplikasi Delphi adalah form. Form dapat dianggap sebagai sebuah windows. Defaultnya, Delphi memberikan sebuah form pada setiap project

dan menampilkan sebuah form kosong pada saat memulai Delphi. Form dipakai untuk menempatkan komponen, kontrol dan lain-lain. Informasi mengenai form disimpan di dalam dua file, yaitu file **.dmf** dan **.pas**. File **.dmf** berisi informasi mengenai tampilan form, ukuran, lokasi dan seterusnya. Program untuk mengontrol form disimpan di file **.pas**, atau disebut juga dengan **unit**. Form mempunyai **property**. Dengan **property** tersebut dapat mengubah tampilan form dengan mengubah warna, ukuran, lokasi dan lain-lain. Sebuah form mempunyai sejumlah **event handler**.

Properties

Sebuah objek diakses melalui **property**-nya. **Property** dari objek Delphi mempunyai nilai yang dapat diubah pada saat desain tanpa perlu menuliskan program. Jika ingin mengubah **property** pada saat aplikasi dijalankan maka harus menuliskan sedikit program. Di Delphi, semua form dan semua komponen (**Visual Component**) mempunyai **properties**. Form mempunyai ukuran (tinggi dan lebar), warna **background**, batas dan juga nama sehingga dapat mengontrol penampilan objek-objek tersebut dengan memanipulasi atribut atau **property**-nya.

2.8 Adobe Photoshop CS

Pemilihan menggunakan software ini karena kemampuan Adobe Photoshop dalam memberikan kemudahan untuk merancang desain gambar yang sangat berguna dalam perancangan desain antarmuka (*interface*) pada aplikasi yang akan dibangun. Tipe data gambar yang dihasilkan oleh Adobe Photoshop antara lain: **.psd**, **.bmp**, **.jpg**, **.gif** (animasi), **.gif**, **.png**, **.tif**.

2.9 SWISHmax

SWISH untuk membuat animasi yang kompleks dalam waktu cepat. Dengan SWISH, animasi yang begitu kompleks dapat dibuat dalam hitungan detik. Terlebih lagi dalam versi terbarunya, yaitu SWISHmax, skrip telah ditingkatkan sehingga dapat memaksimalkan kreasi dalam membuat animasi dan interaksi. Movie SWISH diorientasikan untuk diekspor ke dalam format file .swf (file Flash Player) yang kemudian dimasukkan dalam aplikasi yang akan dibangun sehingga tampilan menjadi lebih menarik [Arry. 2005].

2.10 MySQL Database

MySQL adalah *engine* basis data yang mendukung bahasa *query* SQL basis data. SQL adalah standarisasi cara komunikasi ke basis data dengan segala bentuk tanpa mempertimbangkan metode penyimpanan dan pemanggilan.

Berangkat dari software yang *shareware*, MySQL kini populer melalui versi 2.2 menjadi software *open source* yang berarti gratis. MySQL juga tersedia di lingkungan windows, software MySQL di lingkungan windows di pasang pada direktori c:\mysql.

Seperti halnya SQL *engine* yang lain, MySQL mempunyai tiga subbahasa, yaitu *Data Definition Language* (DDL), *Data Manipulation Language* (DML), dan *Data Control Language* (DCL). DDL berfungsi pada obyek database, seperti membuat tabel, mengubah tabel, dan menghapus tabel. DML untuk obyek tabel, seperti melihat, menambah, menghapus dan mengubah isi tabel. Sedangkan DCL untuk kepentingan *securitas database*, seperti memberikan hak akses ke *database* dan menghapus hak tersebut dari *database* [Arif.2003].

2.11 Jenis dan Analisis Data

Pada suatu proses penelitian sering hanya terdapat satu jenis data yaitu kuantitatif atau kualitatif saja, tetapi mungkin juga gabungan keduanya. Dalam analisis data juga terdapat dua macam, yaitu analisis data kuantitatif dan analisis data kualitatif.

Data kualitatif adalah data yang berbentuk kata, kalimat, skema dan gambar. Data kuantitatif adalah data yang berbentuk angka atau data kualitatif yang diangkakan. Data kualitatif yang diangkakan (skorsing) misalnya terdapat dalam skala pengukuran. Suatu pernyataan atau pertanyaan yang memerlukan alternative jawaban sangat puas, puas, kurang puas dan tidak puas dimana masing – masing sangat puas diberi angka 4, puas 3, kurang puas 2, tidak puas 1.

Skala pengukuran merupakan yang digunakan sebagai acuan untuk menentukan panjang pendeknya interval yang ada dalam alat ukur, sehingga alat ukur tersebut bila digunakan dalam pengukuran akan menghasilkan data kuantitatif. Dengan skala pengukuran ini, maka nilai variable yang diukur dengan instrument tertentu dapat dinyatakan dalam bentuk angka, sehingga akan lebih akurat, lebih efisien dan komunikatif. Berbagai skala yang digunakan untuk penelitian social antara lain adalah :

1. Skala Likert

Skala Likert digunakan untuk mengukur sikap, pendapat dan persepsi seseorang dan atau sekelompok orang tentang fenomena social. Dengan *Skala Likert*, maka variable yang akan diukur dijabarkan sebagai titik tolak untuk menyusun butir – butir yang menggunakan *Skala Likert* dapat berupa kata – kata antara lain :

1. Sangat Setuju / Puas /Penting
2. Setuju / Puas /Penting
3. Kurang Setuju / Puas /Penting
4. Tidak Setuju / Puas /Penting

Untuk keperluan analisis kuantitatif, maka jawaban itu dapat diberi skor, misalnya:

- | | |
|--|---|
| 1. Sangat Setuju / Puas /Penting diberi skor | 4 |
| 2. Setuju / Puas /Penting diberi skor | 3 |
| 3. Kurang Setuju / Puas /Penting diberi skor | 2 |
| 4. Tidak Setuju / Puas /Penting diberi skor | 1 |

2. Skala Guttman

Skala pengukuran dengan tipe ini, akan didapatkan jawaban yang tegas yaitu “Ya – Tidak”, “Benar – Salah” dan lain-lain. Data yang diperoleh dapat berupa data interval atau rasio. Jadi jika pada *Skala Likert* terdapat interval dari kata “Sangat setuju” sampai “Tidak setuju, maka *Skala Guttman* hanya ada dua interval yaitu “Setuju” atau “Tidak setuju”. Penelitian dengan menggunakan *Skala Guttman* dilakukan bila ingin mendapatkan jawaban yang tegas terhadap suatu permasalahan yang ditanyakan.

3. Skala Scale

Skala Scale merupakan data mentah yang diperoleh dari angka kemudian ditafsirkan dalam pengertian kualitatif. Reponden menjawab, senang atau tidak senang, setuju atau tidak setuju, pernah atau tidak pernah adalah merupakan data

kualitatif. Dalam *Skala Scale*, responden tidak akan menjawab salah satu dari jawaban kualitatif yang disediakan, tetapi menjawab salah satu jawaban kuantitatif yang telah disediakan.

4. Samantict deferensial

Skala pengukuran yang berbentuk *Samantict deferensial* dikembangkan oleh Osgood. Skala ini juga digunakan untuk mengukur sikap, hanya bentuknya checklist, tetapi tersusun dalam satu garis kontinyu yang jawabannya sangat positifnya terletak di bagian kanan garis, dan jawabannya yang sangat negative terletak dibagian kiri garis, atau sebaliknya. Data yang diperoleh adalah data interval, dan biasanya skala ini digunakan untuk mengukur sikap/karakteristik tertentu yang dipunyai oleh seseorang.

2.12 Metode Analisis Data

Dalam menganalisis data kuisioner digunakan metode deskriptif kualitatif. Dalam hal ini, terdapat 3 tipe jawaban yaitu *Setuju*, *Puas* dan *Penting* dan digunakan 4 skala (*Likert*) yang terdiri dari *Sangat Setuju/Puas/Penting*, *Setuju/Puas/Penting*, *Kurang Setuju/Puas/Penting* dan *Tidak Setuju/Puas/Penting*.

Ke-empat penilaian tersebut diberikan bobot sebagai berikut :

- a. Jawaban Sangat Setuju/Puas/Penting diberi bobot 4.
- b. Jawaban Setuju/Puas/Penting diberi bobot 3.
- c. Jawaban Kurang Setuju/Puas/Penting diberi bobot 2.
- d. Jawaban Tidak Setuju/Puas/Penting diberi bobot 1.

Kemudian dilakukan interpretasi terhadap data kuisioner berdasarkan rata-rata. Perhitungan rata-rata menggunakan rumus sebagai berikut :

$$\bar{x} = \frac{\sum_{i=1}^n x_i}{n} \dots\dots\dots(2.1)$$

Di mana,

\bar{x} = rata-rata skor

x_i = skor jawaban variabel ke-i.

n = jumlah responden.

Seandainya skor rata-rata bulat, maka dengan mudah menginterpretasikannya. Namun apabila skor rata-rata bilangan pecahan, maka dapat menggunakan skala numerik linier. Untuk mendapatkan skala tersebut, cari dulu rentang skala (RS) dengan rumus sebagai berikut :

$$RS = \frac{m - n}{b} \dots\dots\dots(2.2)$$

Di mana,

m = angka tertinggi didalam pengukuran.

n = angka terendah dalam pengukuran.

b = banyaknya kelas yang dibentuk.

Karena skala yang digunakan dalam Sistem Aplikasi Pengisian Kuisioner Skala Likert Beserta Interpretasi tetap tidak berubah, maka rentang skala adalah 0,75 dan skala numeriknya sebagai berikut :

Tidak Setuju/Puas/Penting : $1 \leq \bar{x} \leq 1,75$

Kurang Setuju/Puas/Penting : $1,75 < \bar{x} \leq 2,5$

Setuju/Puas/Penting : $2,5 < \bar{x} \leq 3,25$

Sangat Setuju/Puas/Penting : $\bar{x} > 3,25$



BAB III

ANALISIS KEBUTUHAN PERANGKAT LUNAK

3.1 Metode Analisis

Tahap analisis merupakan tahap yang sangat penting, karena jika dalam tahap ini terdapat kesalahan, maka akan mempengaruhi tahap-tahap selanjutnya dalam pembuatan perangkat lunak.

Dalam penelitian ini, metode analisis yang digunakan adalah metode dengan pendekatan terstruktur, yaitu masalah yang dirumuskan algoritmanya.

Tahap analisis ini, terdapat langkah-langkah dasar yang harus dilakukan, yaitu :

- a. *Identify*, yaitu mengidentifikasi masalah
- b. *Understand*, yaitu memahami cara kerja sistem yang ada
- c. *Analyze*, yaitu menganalisis sistem
- d. *Report*, yaitu membuat laporan hasil analisis

3.2 Hasil Analisis

a. Data Masukan

Data masukan merupakan sumber data yang dipergunakan oleh sistem yang akan dirancang dan dibuat untuk diolah menjadi informasi yang diinginkan.

Adapun data masukan yaitu :

1. Rancangan Kuisisioner

Rancangan yang dimaksud adalah rancangan kuisisioner yang akan dijawab oleh responden. Dalam hal ini operator yang akan merancang sesuai dengan kebutuhan. Yang dirancang oleh operator yaitu :

- Informasi, yang berisi tentang prakata dari peneliti
- Topik, yaitu topik yang menjadi bahan kuisisioner. Dalam hal ini, jumlah topik dibatasi yaitu maksimal 5 topik
- Pertanyaan, merupakan pertanyaan-pertanyaan yang akan dijawab oleh responden. Setiap topik dibatasi hanya maksimal 10 pertanyaan

2. Data Kuisisioner

Data kuisisioner adalah data responden dalam mengisi kuisisioner yang telah dirancang oleh operator.

b. Data Keluaran

1. Kuisisioner

Kuisisioner ini merupakan hasil rancangan dari operator yang akan dijawab oleh responden.

2. Proses Data

Hasil kuisisioner serta hasil uji statistik dari kuisisioner yang telah diisi oleh responden.

3.3 Perangkat Keras

Perangkat keras yang dibutuhkan untuk dapat menunjang sistem informasi ini dapat berjalan dengan baik harus memiliki minimal spesifikasi sebagai berikut :

1. Intel Pentium II atau yang lebih tinggi
2. Microsoft Windows 98 SE
3. RAM 64 MB
4. VGA Card 8 MB
5. Disk Free Space 8 MB untuk Program

3.4 Pembuatan Sistem Perangkat Lunak

3.4.1 Perancangan perangkat lunak dan interface

Pada pencangan perangkat lunak ini berisi rancangan tampilan-tampilan yang akan muncul pada komputer, yang memudahkan pengguna dalam menggunakan perangkat sistem tersebut. Tampilan yang dimaksud adalah grafis antarmuka (*interface*) dari perangkat lunak yang dibangun dari Borland Delphi didukung juga dengan Photoshop dan Swish, yang bertujuan menyediakan media komunikasi yang baik dan mudah antara pengguna dan perangkatnya. Rancangan ini dibuat dengan bahasa yang sederhana dan umum, sehingga dapat mudah dipahami oleh pengguna. Rancangan ini dimulai dari tampilan muka, tampilan informasi dan tampilan fungsi Statistika.

3.4.2 Pembuatan perangkat lunak dan grafis antar muka

Pada pembuatan *interface* perangkat lunak, dibangun dari Borland Delphi sendiri dengan bantuan sintaksis-sistaksis yang terdapat dalam Borland Delphi serta Photoshop dan Swish. Borland Delphi merupakan cara paling sederhana untuk membangun aplikasi berbasis Windows. Photoshop adalah sebuah program untuk merancang desain gambar yang sangat berguna dalam perancangan desain antarmuka (*interface*) pada aplikasi yang akan dibangun. Sedangkan Swish untuk membuat animasi yang kompleks dalam waktu cepat yang akan berguna untuk mempercantik perancangan antarmuka (*interface*) pada aplikasi yang akan dibangun.

3.5 Implementasi Perangkat Lunak Statistika

Dalam implementasi perangkat lunak ini dimulai dari tampilan awal sampai tampilan fungsi statistika. Pada tampilan fungsi Statistika dijelaskan cara memasukkan data, penjelasan tombol-tombol fungsi Statistika untuk proses data dan tombol fungsi lain yang ada pada tampilan tersebut.

a. Sistem Operasi Windows 98 SE

Windows 98 SE sebagai sistem operasi yang digunakan untuk menjalankan aplikasi-aplikasi yang dibutuhkan untuk membangun sistem.

b. Borland Delphi 7.0

Untuk pembuatan program sistem aplikasi pengisian kuisisioner skala likert digunakan bahasa pemrograman Borland Delphi 7.0 yang memberikan fasilitas-fasilitas visual yang dapat mempermudah dan

memperindah interface suatu program aplikasi. Kemampuan Borland Delphi dapat digunakan untuk merancang program aplikasi yang memiliki tampilan seperti program aplikasi yang berbasis windows.

c. MySQL 2.2

MySQL dipilih karena kecepatannya dalam pemrosesan database, saat ini MySQL merupakan yang paling cepat di dunia, dengan kecepatannya inilah MySQL banyak dipakai. MySQL dapat menampung database sebesar 100 GB tanpa ada penurunan yang signifikan dalam pemrosesannya, ini merupakan hal yang luar biasa bagi software yang gratis. MySQL merupakan software yang *multi platform* dapat berjalan hampir di semua sistem operasi.

d. SWISHmax

SWISH untuk membuat animasi yang kompleks dalam waktu cepat. Dengan SWISH, animasi yang begitu kompleks dapat dibuat dalam hitungan detik. Terlebih lagi dalam versi terbarunya, yaitu SWISHmax, skrip telah ditingkatkan sehingga dapat memaksimalkan kreasi dalam membuat animasi dan interaksi. Movie SWISH diorientasikan untuk diekspor ke dalam format file .swf (file Flash Player) yang kemudian dimasukkan dalam aplikasi yang akan dibangun sehingga tampilan menjadi lebih menarik.

BAB IV

PERANCANGAN SISTEM

4.1 Metode Perancangan

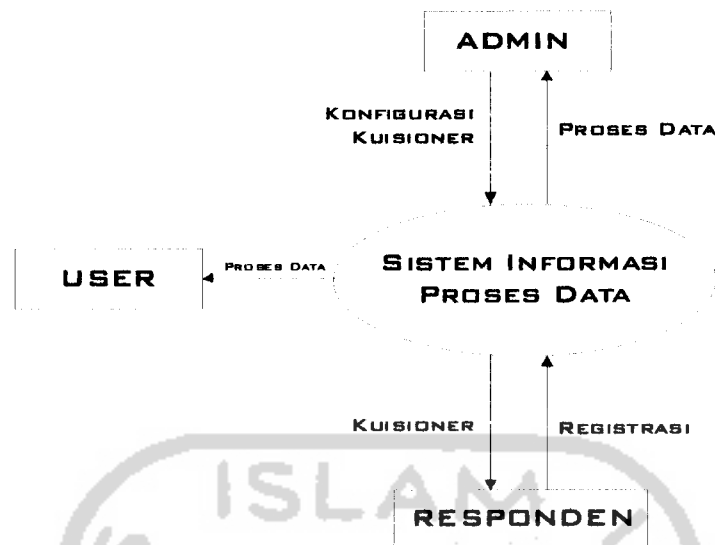
Metode yang digunakan dalam perancangan Sistem Aplikasi Pengisian Kuisisioner Skala Likert Beserta Interpretasi adalah menggunakan metode perancangan berarah pada aliran data, aliran data tersebut meliputi input data dari form input kuisisioner dan input data kuisisioner.

4.2 Hasil Perancangan

Hasil perancangan dibagi menjadi : Diagram Konteks, Diagram Arus Data (*Data Flow Diagram*), Struktur Tabel, Relasi Antar Tabel dan Perancangan Bagian Antarmuka.

4.2.1 Diagram Konteks

Diagram Konteks ini menggambarkan proses aliran data secara umum. Diagram konteks digunakan untuk menggambar hubungan *input / output* antara sistem dengan dunia luarnya (kesatuan luar), suatu diagram konteks sistem selalu mengandung satu proses saja yang mewakili proses dari seluruh sistem. Untuk lebih jelasnya lihat gambar 4.1.



Gambar 4.1 Diagram Konteks

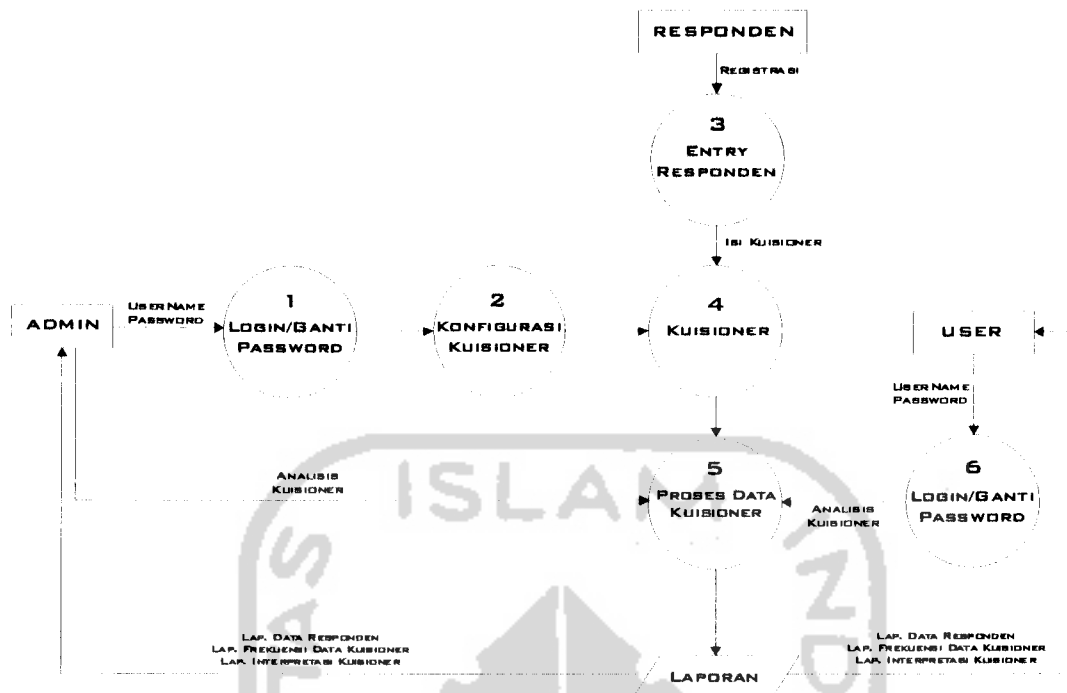
Keterangan Diagram Konteks :

Diagram Konteks merupakan gambaran dari seluruh sistem secara umum, dimana administrator sistem bisa berhubungan dengan sistem untuk proses login, ganti password, mengedit user, konfigurasi kuisisioner dan analisis data kuisisioner. Operator melakukan proses login, ganti password dan proses data kuisisioner. Responden berhubungan dengan sistem pada proses pengisian kuisisioner.

4.2.2 Diagram Arus Data (Data Flow Diagram)

4.2.2.1 DFD level 1

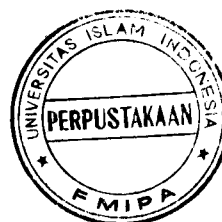
DFD Level 1 menggambarkan semua proses yang terjadi pada Sistem Aplikasi Pengisian Kuisisioner Skala Likert, disini ada 6 proses utama yaitu : Login untuk Admin, Konfigurasi Kuisisioner, Kuisisioner, Proses Data Kuisisioner, Login untuk Operator. Laporan DFD Level 1 dapat dilihat pada gambar 4.2.



Gambar 4.2 DFD Level 1

Keterangan DFD Level 1:

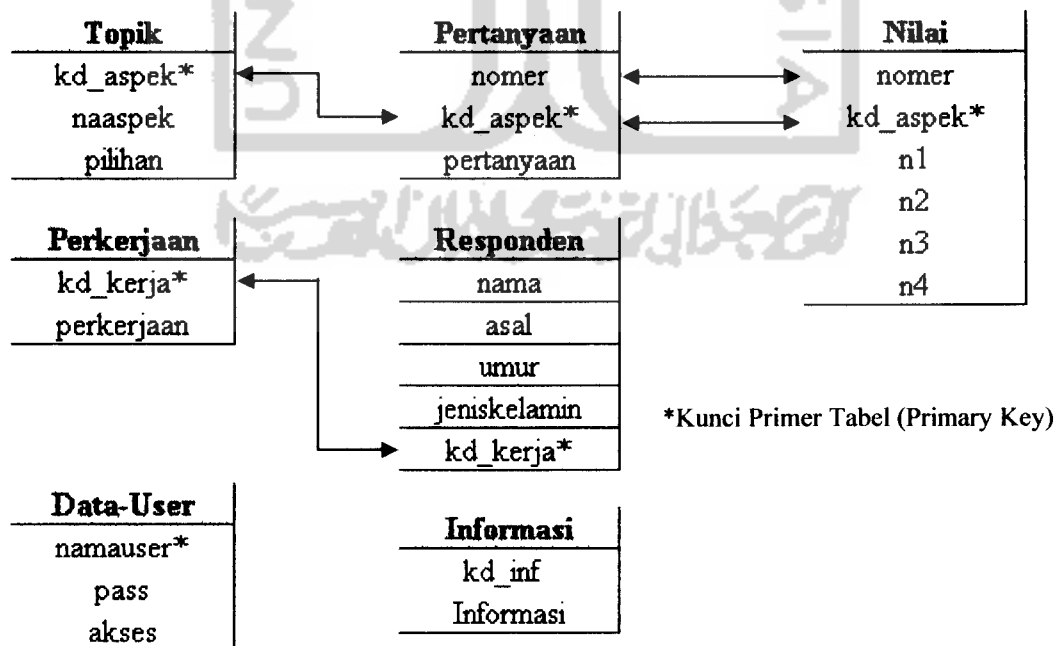
Setelah administrator melakukan proses login dengan memasukkan *username* dan *password*, admin dapat melakukan proses konfigurasi kuisioner. Kemudian responden melakukan registrasi dan mengisi kuisioner yang telah disetup oleh administrator. Selanjutnya administrator dan operator melakukan proses data kuisioner yang telah diisi oleh responden sesudah melakukan login terlebih dahulu. Kemudian administrator dan operator akan mendapatkan hasil laporan dari proses data kuisioner. Untuk responden berhubungan dengan sistem pada saat melakukan proses pengisian kuisioner.



4.2.3 Relasi Antar Tabel

Basis data (*database*) merupakan kumpulan dari data yang saling berhubungan yang satu dengan yang lainnya. Data-data tersebut disimpan dalam sebuah file atau tabel yang memiliki keterangan tertentu, dan apabila direlasikan akan menghasilkan informasi yang kompleks. Hubungan antar tabel ini dihasilkan dengan kunci relasi (*relational key*) yang merupakan kunci utama dari masing-masing file atau tabel. Keberadaan tabel relasi dapat memudahkan dalam pemeliharaan data dan menghindarkan kerangkapan data, sehingga informasi yang diperoleh akan lebih akurat dan meminimalkan pembuatan *record*.

Perencanaan *database* yang tepat akan menyebabkan paket program relasional akan bekerja secara optimal. Hubungan antar tabel yang satu dengan tabel yang lainnya untuk relasi tabel pada Sistem Aplikasi Pengisian Kuisisioner Skala Likert Beserta Interpretasi dapat dilihat pada gambar 4.3.



Gambar 4.3 Relasi Antar Tabel

4.2.4 Desain Basis Data

Basis data merupakan salah satu komponen yang penting pada sistem informasi, karena berfungsi sebagai penyedia informasi bagi para pemakainya. Desain basis data dari Sistem Aplikasi Pengisian Kuisisioner Skala Likert Beserta Interpretasi adalah sebagai berikut:

1. Tabel Data User

Tabel data user berfungsi untuk menyimpan data Admin dan Operator beserta password. Struktur tabel data user dapat dilihat pada tabel 4.1.

Tabel 4.1 Tabel Data User

Name	Type	Null	Default	Key
namauser	varchar(15)	No		PRI
Pass	varchar(32)	No		
Akses	tinyint(3)unsigned	No	1	

2. Tabel Informasi

Tabel informasi berfungsi untuk menyimpan data informasi mengenai kuisisioner yang akan diisi oleh responden. Struktur tabel informasi dilihat pada tabel 4.2.

Tabel 4.2 Tabel Informasi

Name	Type	Null	Default	Key
kd_inf	char(3)	no		
Informasi	text	yes		

Tabel 4.7 Tabel Nilai

Name	Type	Null	Default	Key
nomer	int(10)unsigned	No	1	
kd_aspek	int(10)unsigned	No	1	PRI
n4	int(10)unsigned	No	0	
n3	int(10)unsigned	No	0	
n2	int(10)unsigned	No	0	
n1	int(10)unsigned	No	0	

4.2.5 Perancangan Bagian Antarmuka

4.2.5.1 Perancangan Tampilan Awal (Operator)

Rancangan tampilan awal ditunjukkan pada gambar 4.4 berikut :



Gambar 4.4 Rancangan Tampilan Awal

Tampilan awal terdapat menu pilihan. Menu pilihan digunakan untuk membuka menu-menu pada sistem. Serta terdapat panel untuk memasukkan user

2. *Delphi* merupakan program aplikasi database berbasis objek pascal, yang telah *familier* digunakan oleh penulis.
3. MySQL dipilih karena kecepatannya dalam pemrosesan database.

5.3 Batasan Implementasi

Sebelum rancangan sistem baru diterapkan perlu dilakukan pengujian, yang merupakan langkah penting untuk mendapatkan hasil rancangan sistem yang baik. Pengelolaan data kuisisioner dalam hal interpretasi yang biasanya dilakukan secara manual namun diharapkan dapat dilakukan alternatif dengan menggunakan bantuan sistem ini. Disamping itu dapat juga dilakukan pengelompokkan responden dalam hal usia, jenis kelamin dan pekerjaan responden.

Dalam tahap pembuatan Sistem Aplikasi Pengisian Kuisisioner ini, implementasi dibatasi pada proses pemasukan data, pengeditan data, penghapusan data, perhitungan interpretasi dan proses pembuatan laporan. Program aplikasi ini sementara dibuat untuk dapat berjalan pada jaringan komputer lokal (*intranet*) yang selanjutnya dapat dikembangkan jaringan *internet*.

5.4 Tahapan Pembuatan Program

Pembuatan program Sistem Aplikasi Pengisian Kuisisioner Skala Likert Beserta Interpretasi dengan *Delphi* ini, melalui 3 tahap :

1. Tahap pembuatan *database*

Pembuatan program diawali dengan mengimplementasikan hasil rancangan sistem ke dalam tabel-tabel yang akan diintegrasikan menjadi sebuah *database*, perangkat lunak yang digunakan adalah MySQL.

3. Tabel Topik

Tabel kelompok berfungsi untuk menyimpan data topik. Struktur tabel topik dapat dilihat pada tabel 4.3.

Tabel 4.3 Tabel Topik

Name	Type	Null	Default	Key
kd_aspek	int(3)unsigned	No	0	PRI
Naspek	varchar(50)	Yes		
Pilihan	tinyint(3)unsigned	Yes		1

4. Tabel Pertanyaan

Tabel Pertanyaan berfungsi untuk menyimpan data pertanyaan yang kemudian ditampilkan ke kuisioner. Struktur tabel pertanyaan dapat dilihat pada tabel 4.4.

Tabel 4.4 Tabel Pertanyaan

Name	Type	Null	Default	Key
Nomer	int(10)unsigned	No	1	
kd_aspek	tinyint(10)unsigned	No	1	PRI
pertanyaan	varchar(50)	Yes		

5. Tabel Perkerjaan

Tabel Perkerjaan berfungsi untuk menyimpan data perkerjaan yang diberi kode. Struktur tabel perkerjaan dapat dilihat pada tabel 4.5.

Tabel 4.5 Tabel Perkerjaan

Name	Type	Null	Default	Key
Nama	varchar(10)	Yes		
Asal	varchar(20)	Yes		
Umur	int(10)unsigned	No	0	
jeniskelamin	char(1)	No	P	
kd_kerja	tyinyint(1)	No	6	PRI

6. Tabel Responden

Tabel Responden berfungsi untuk menyimpan data responden. Struktur tabel responden dapat dilihat pada tabel 4.6.

Tabel 4.6 Tabel Responden

Name	Type	Null	Default	Extra
Nama	varchar(10)	Yes		
Asal	varchar(20)	Yes		
Umur	int(10)unsigned	No	0	
jeniskelamin	char(1)	No	P	
kd_kerja	tyinyint(1)	No	6	

7. Tabel Nilai

Tabel nilai berfungsi untuk menyimpan data kuisisioner yang telah diisi oleh responden. Struktur tabel nilai dapat dilihat pada tabel 4.7.

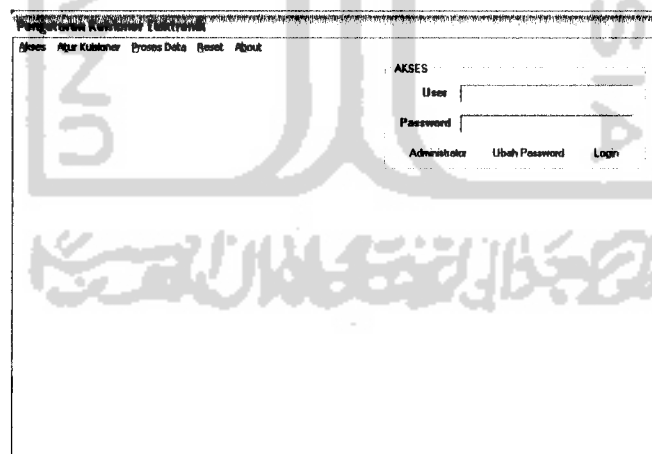
Tabel 4.7 Tabel Nilai

Name	Type	Null	Default	Key
nomer	int(10)unsigned	No	1	
kd_aspek	int(10)unsigned	No	1	PRI
n4	int(10)unsigned	No	0	
n3	int(10)unsigned	No	0	
n2	int(10)unsigned	No	0	
n1	int(10)unsigned	No	0	

4.2.5 Perancangan Bagian Antarmuka

4.2.5.1 Perancangan Tampilan Awal (Operator)

Rancangan tampilan awal ditunjukkan pada gambar 4.4 berikut :



Gambar 4.4 Rancangan Tampilan Awal

Tampilan awal terdapat menu pilihan. Menu pilihan digunakan untuk membuka menu-menu pada sistem. Serta terdapat panel untuk memasukkan user

2. *Delphi* merupakan program aplikasi database berbasis objek pascal, yang telah *familier* digunakan oleh penulis.
3. MySQL dipilih karena kecepatannya dalam pemrosesan database.

5.3 Batasan Implementasi

Sebelum rancangan sistem baru diterapkan perlu dilakukan pengujian, yang merupakan langkah penting untuk mendapatkan hasil rancangan sistem yang baik. Pengelolaan data kuisisioner dalam hal interpretasi yang biasanya dilakukan secara manual namun diharapkan dapat dilakukan alternatif dengan menggunakan bantuan sistem ini. Disamping itu dapat juga dilakukan pengelompokkan responden dalam hal usia, jenis kelamin dan pekerjaan responden.

Dalam tahap pembuatan Sistem Aplikasi Pengisian Kuisisioner ini, implementasi dibatasi pada proses pemasukan data, pengeditan data, penghapusan data, perhitungan interpretasi dan proses pembuatan laporan. Program aplikasi ini sementara dibuat untuk dapat berjalan pada jaringan komputer lokal (*intranet*) yang selanjutnya dapat dikembangkan jaringan *internet*.

5.4 Tahapan Pembuatan Program

Pembuatan program Sistem Aplikasi Pengisian Kuisisioner Skala Likert Beserta Interpretasi dengan *Delphi* ini, melalui 3 tahap :

1. Tahap pembuatan *database*

Pembuatan program diawali dengan mengimplementasikan hasil rancangan sistem ke dalam tabel-tabel yang akan diintegrasikan menjadi sebuah *database*, perangkat lunak yang digunakan adalah MySQL.

dan password, serta 3 *Button* untuk Administrasi (Admin), Ubah Password (Operator) dan Login.

4.2.5.2 Perancangan Administrasi

Rancangan Administrasi ditunjukkan pada gambar 4.5

Name User	USER	PASSWORD	KONFIRMASI PASSWORD	AKSES
				PIN

Tambah Edit Update

Gambar 4.5 Rancangan Administrasi

Rancangan ini digunakan oleh admin untuk mengganti password bagi admin serta mengedit/menambah Operator. Data masukan terdiri dari user, password, konfirmasi password dan akses. Akses disini untuk menentukan apakah user tersebut merupakan admin atau operator. Data dimasukkan ke tabel user.

4.2.5.3 Perancangan Ubah Password

Rancangan Ubah Password ditunjukkan pada gambar 4.6

USER

PASSWORD BARU

KONFIRMASI PASSWORD

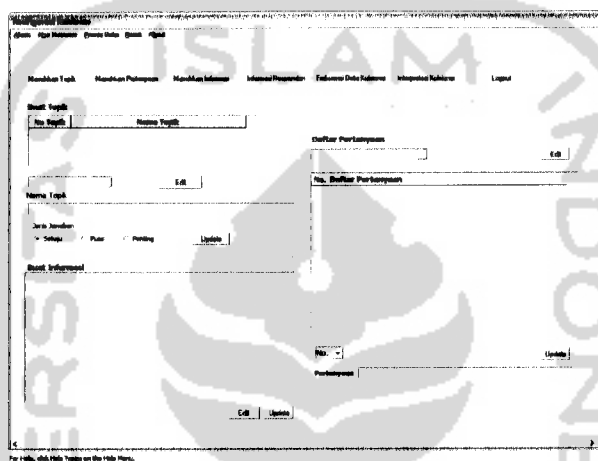
Cancel Update

Gambar 4.6 Rancangan Ubah Password

Rancangan ini digunakan Operator untuk mengganti password. Data masukan terdiri dari user, password baru dan konfirmasi password. Data dimasukkan ke tabel user.

4.2.5.4 Perancangan Konfigurasi Kuisisioner

Rancangan Konfigurasi Kuisisioner ditunjukkan pada gambar 4.7

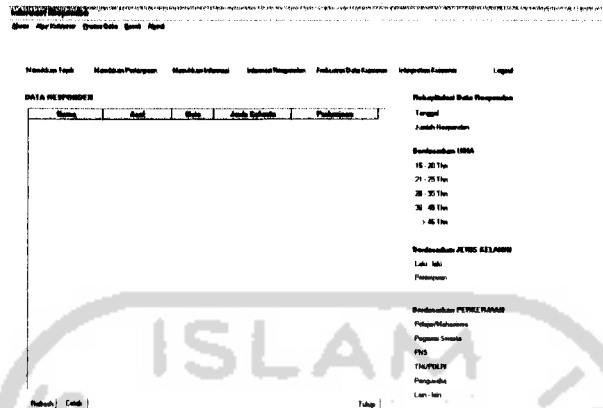


Gambar 4.7 Rancangan Konfigurasi Kuisisioner

Rancangan konfigurasi kuisisioner digunakan untuk mengatur kuisisioner yang akan ditampilkan. Data masukan terdiri dari Topik (max. 5 topik), Pertanyaan (max. 10 pertanyaan/topik) dan informasi. Data topik dimasukkan ke tabel topik, data pertanyaan dimasukkan ke tabel pertanyaan dan data informasi dimasukkan ke tabel informasi. Selain itu juga terdapat tombol yang akan memudahkan user, terdiri dari *button* Masukkan Topik, *button* Masukkan Pertanyaan, *button* Masukkan Informasi, *button* Informasi Responden, *button* Frekuensi Data Kuisisioner, *button* Interpretasi Kuisisioner dan Logout.

4.2.5.5 Perancangan Informasi Responden

Rancangan Informasi Responden ditunjukkan pada gambar 4.8

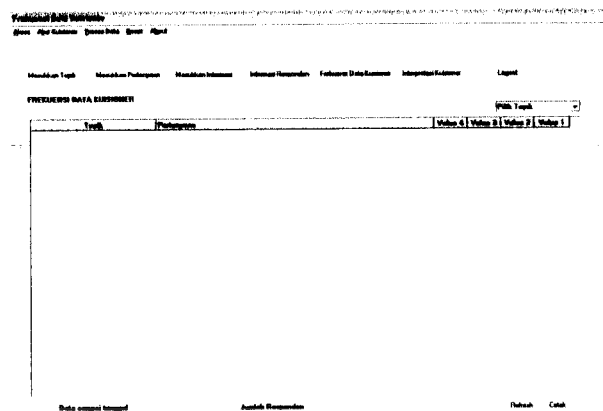


Gambar 4.8 Rancangan Informasi Responden

Rancangan Informasi Responden digunakan untuk mengetahui rekapitulasi responden yang telah mengisi kuisisioner serta dikelompokkan berdasarkan usia, jenis kelamin dan pekerjaan. Data diperoleh dari tabel responden. Selain itu juga terdapat tombol yang akan memudahkan user, terdiri dari *button* Masukkan Topik, *button* Masukkan Pertanyaan, *button* Masukkan Informasi, *button* Informasi Responden, *button* Frekuensi Data Kuisisioner, *button* Interpretasi Kuisisioner dan Logout. Pada *button* cetak digunakan untuk rancangan keluaran laporan untuk digunakan mencetak laporan.

4.2.5.6 Perancangan Frekuensi Data kuisisioner

Rancangan Frekuensi Data kuisisioner ditunjukkan pada gambar 4.9

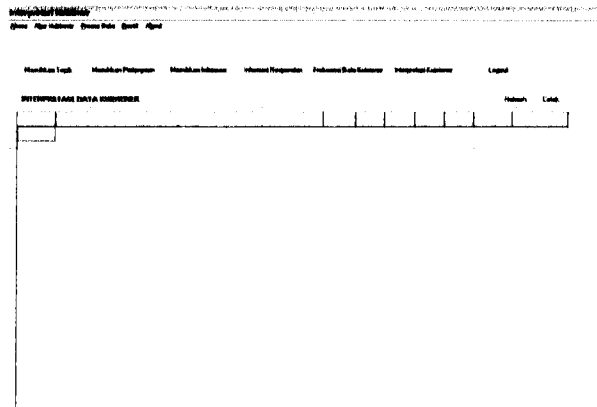


Gambar 4.9 Rancangan Frekuensi Data kuisioner

Rancangan Frekuensi Data Kuisioner digunakan untuk mengetahui rekapitulasi frekuensi jawaban dari responden dan terdapat pilihan apabila frekuensi berdasarkan item topik. Data diperoleh dari tabel nilai, topik dan pertanyaan. Selain itu juga terdapat tombol yang akan memudahkan user, terdiri dari *button* Masukkan Topik, *button* Masukkan Pertanyaan, *button* Masukkan Informasi, *button* Informasi Responden, *button* Frekuensi Data Kuisioner, *button* Interpretasi Kuisioner dan Logout. Pada *button* cetak digunakan untuk rancangan keluaran laporan untuk digunakan mencetak laporan.

4.2.5.7 Perancangan Interpretasi Kuisioner

Rancangan Interpretasi Kuisioner ditunjukkan pada gambar 4.10



Gambar 4.10 Rancangan Interpretasi Kuisisioner

Rancangan Interpretasi Kuisisioner digunakan untuk mengetahui interpretasi dari tiap-tiap pertanyaan. Data yang digunakan dari tabel nilai yang kemudian dihitung interpretasi tiap-tiap pertanyaan. Selain itu juga terdapat tombol yang akan memudahkan user, terdiri dari *button* Masukkan Topik, *button* Masukkan Pertanyaan, *button* Masukkan Informasi, *button* Informasi Responden, *button* Frekuensi Data Kuisisioner, *button* Interpretasi Kuisisioner dan Logout. Pada *button* cetak digunakan untuk rancangan keluaran laporan untuk digunakan mencetak laporan.

4.2.5.8 Perancangan Tampilan Awal (Responden)

Rancangan Tampilan Awal (Responden) ditunjukkan pada gambar 4.11

Gambar 4.11 Rancangan Tampilan Awal (Responden)

Rancangan Tampilan Awal digunakan untuk registrasi bagi responden serta informasi seputar kuisisioner yang akan diisi oleh responden. Data masukan pada registrasi terdiri dari nama, asal, usia, jenis kelamin, dan pekerjaan. Data yang diperoleh dari registrasi dimasukkan ke tabel responden. Sedangkan informasi diambil dari tabel informasi.

4.2.5.9 Perancangan Tampilan Kuisisioner (Responden)

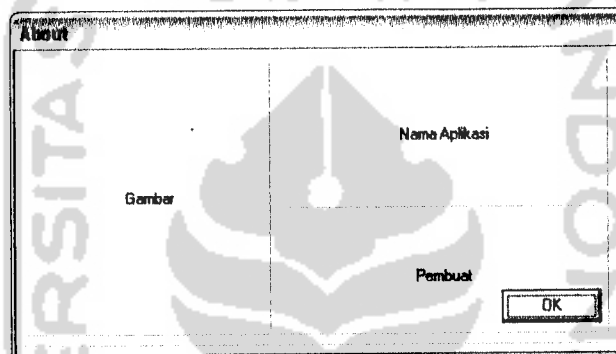
Rancangan Tampilan Kuisisioner (Responden) ditunjukkan pada gambar 4.12

Gambar 4.12 Rancangan Tampilan Kuisisioner (Responden)

Rancangan Tampilan Kuisisioner digunakan untuk menampilkan pertanyaan-pertanyaan yang akan dijawab oleh responden. Data diambil dari tabel pertanyaan dan untuk jawaban dari responden dimasukkan ke tabel nilai. Jumlah topik dan pertanyaan diatur oleh admin.

4.2.5.10 Perancangan Tampilan Pembuat

Rancangan Tampilan Pembuat ditunjukkan pada gambar 4.13



Gambar 4.13 Rancangan Tampilan Pembuat

Rancangan Tampilan Pembuat digunakan sebagai informasi mengenai pembuat sistem aplikasi tersebut.

BAB V

IMPLEMENTASI PERANGKAT LUNAK

5.1 Implementasi Secara Umum

Sistem Aplikasi Pengisian Kuisisioner Skala Likert Beserta Interpretasi diimplementasikan dengan menggunakan bahasa pemrograman *Borland Delphi 7.0*. Implementasi sistem merupakan tahap dimana sistem siap untuk dioperasikan pada keadaan yang sebenarnya, sehingga akan diketahui apakah sistem yang dibuat benar-benar dapat menghasilkan tujuan yang diinginkan.

Sebelum program diterapkan dan diimplementasikan, maka program harus *error free* (bebas kesalahan). Kesalahan program yang mungkin terjadi antara lain kesalahan penulisan bahasa, kesalahan sewaktu proses atau kesalahan logikal. Setelah program bebas dari kesalahan, program dites dengan memasukkan data untuk diolah.

5.2 Alasan Pemilihan Perangkat Lunak

Perangkat lunak yang digunakan adalah *Borland Delphi 7.0* dan *MySQL*.

Alasan pemilihan perangkat lunak ini karena :

1. *Borland Delphi 7.0* merupakan perangkat lunak untuk pengembangan sistem berbasis sistem operasi *windows* dan mudah digunakan karena bersifat *Visual Programming Language* yang berarti dapat dilakukan pemrograman dengan desain program secara *visual*.

2. Tahap pemrograman *visual*

Pada tahap ini mula-mula merancang *form* yang akan dipakai pada program. Perancangan *form* beserta komponen-komponen yang dilibatkan ditangani dengan melibatkan perangkat yang tersedia dalam paket *Delphi 7.0*.

3. Tahap penulisan kode

Pada tahap ini menuliskan kode-kode yang dilekatkan pada komponen-komponen yang dipakai.

5.5 Implementasi Perangkat Lunak

Pembuatan program dengan Borland Delphi 7.0 melalui dua tahap, yaitu :

a. Tahap pembuatan visual

Pada tahap pemrograman visual, dimulai dengan merancang *form* yang dipakai untuk media input atau proses sebagai *interface*-nya. Perancangan form beserta kontrol-kontrolnya dibuat dengan menggunakan perangkat-perangkat yang tersedia dalam *Visual Delphi*.

b. Tahap penulisan kode

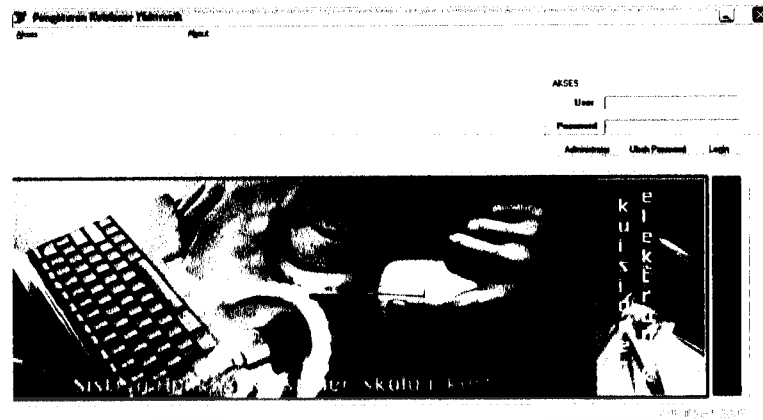
Pada tahap penulisan kode, akan ditulis kode yang diletakkan pada kontrol-kontrol yang dipakai.

5.6 Pembuatan Program

5.6.1 Admin dan Operator

5.6.1.1 Menu Utama

Pada menu utama terdapat beberapa prosedur yang digunakan, diantaranya untuk menjalankan menu lainnya:



Gambar 5.1 Tampilan Menu Utama

5.6.1.2 Login

Pada menu login terdapat beberapa prosedur yang digunakan. Prosedur menu login :

```

procedure TForm1.SpeedButton1Click(Sender: TObject); // tombol
login
var form3:Tform3;
begin
    Q1.Close;
    Q1.SQL.Clear;
    Q1.SQL.Add('select pass,akses from datauser where pass=md5(:UP)
and namauser=:NM ');
    Q1.params[0].AsString:=Edit2.Text;
    Q1.params[1].AsString:=Edit1.Text;
    Q1.Open;
    if (Q1.FieldByName('pass').AsString<>'') then
    begin
        if (Q1.FieldByName('akses').value=1) then
        begin
            Edit1.Clear;
            Edit2.Clear;
            Edit1.SetFocus;
            form3 := Tform3.Create(application);
            Form3.Aturl.Enabled := false;
            form3.ShowModal;
        end
    else
        begin
            Edit1.Clear;

```

```

        Edit2.Clear;
        Edit1.SetFocus;
        form3 := TForm3.Create(application);
        form3.ShowModal;
    end;
end
else
begin
    Messagedlg('Nama user atau password Anda
salah!', mtWarning, [mbOk], 0);
    Edit2.Clear;
    Edit1.SetFocus;
end;
end;
end;

```

5.6.1.3 Ubah Password

Pada menu ubah password terdapat beberapa prosedur yang digunakan.

Prosedur menu Ubah Password :

```

procedure TForm1.SpeedButton2Click(Sender: TObject); // tombol
ubah password
var form5:Tform5;
begin
    Q1.Close;
    Q1.SQL.Clear;
    Q1.SQL.Add('select pass from datauser where pass=md5(:UP) and
namauser=:NM');
    Q1.params[0].AsString:=Edit2.Text;
    Q1.params[1].AsString:=Edit1.Text;
    Q1.Open;
    if (Q1.FieldByName('pass').AsString<>'') then
    begin
        form5 := TForm5.Create(application);
        Form5.LUser.Caption:= Form1.Edit1.Text;
        edit1.Clear;
        Edit2.Clear;
        Edit1.SetFocus;
        form5.ShowModal;
    end
    else
    begin
        Messagedlg('Nama user atau password Anda
salah!', mtWarning, [mbOk], 0);
        Edit2.Clear;
        Edit1.SetFocus;
    end;
end;
end;

```

Gambar 5.2 Tampilan ubah pasword

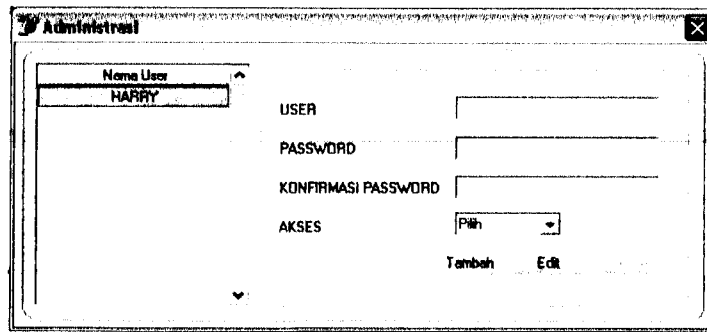
5.6.1.4 Administrasi

Pada menu ini terdapat prosedur untuk menambah user yaitu admin maupun operator dan mengedit user. Prosedur menu administrasi :

```

procedure TForm1.SpeedButton3Click(Sender: TObject); // tombol
administrator
var form4:Tform4;
begin
    Q1.Close;
    Q1.SQL.Clear;
    Q1.SQL.Add('select pass, akses from datauser where
pass=md5(:UP) and namauser=:NM');
    Q1.params[0].AsString:=Edit2.Text;
    Q1.params[1].AsString:=Edit1.Text;
    Q1.Open;
    if (Q1.FieldByName('pass').AsString<>'') then
    begin
        if Q1.FieldByName('akses').Value = 2 then
        begin
            Edit1.Clear;
            Edit2.Clear;
            Edit1.SetFocus;
            form4 := Tform4.Create(application);
            form4.ShowModal;
        end
        else MessageDlg('Anda tidak memiliki akses ini!', mterror,
[mbOK], 0);
        end
    else
    begin
        MessageDlg('Nama user atau password Anda
salah!', mtWarning, [mbOk], 0);
        Edit2.Clear;
        Edit1.SetFocus;
    end;
end;
end;

```



Gambar 5.3 Tampilan Administrasi

5.6.1.5 Konfigurasi Kuisisioner

Pada menu ini digunakan untuk mengatur kuisisioner yang akan ditampilkan pada kuisisioner meliputi topik, pertanyaan dan informasi.

Pada menu topik prosedurnya :

```

procedure TForm3.SpeedButton7Click(Sender: TObject); // Tombol
update
begin
  if (MessageDlg('Anda ingin mengupdate topik no: ' +
    IntToStr(ComboBox1.ItemIndex+1) + ' ?', MtConfirmation,
    [mbYes,mbNo], 0) = mrYes) then
    Begin
      Q4.Close;
      Q4.SQL.Clear;
      Case ComboBox1.ItemIndex of
        0: Q4.SQL.Add('UPDATE topik SET NASpek=:NA, PILIHAN = :OP
WHERE kd_aspek=1');
        1: Q4.SQL.Add('UPDATE topik SET NASpek=:NA, PILIHAN = :OP
WHERE kd_aspek=2');
        2: Q4.SQL.Add('UPDATE topik SET NASpek=:NA, PILIHAN = :OP
WHERE kd_aspek=3');
        3: Q4.SQL.Add('UPDATE topik SET NASpek=:NA, PILIHAN = :OP
WHERE kd_aspek=4');
        4: Q4.SQL.Add('UPDATE topik SET NASpek=:NA, PILIHAN = :OP
WHERE kd_aspek=5');
      end;
      Q4.Params[0].AsString := Edit1.Text ;
      if (R1.Checked) then Q4.Params[1].Value := 1 ;
      if (R2.Checked) then Q4.Params[1].Value := 2 ;
      if (R3.Checked) then Q4.Params[1].Value := 3 ;
      Q4.ExecSQL;
      Q1.Close;
      Q1.Open;
      MessageDlg('Topik telah sukses di Update', MtInformation,
[mbOk], 0);
      Edit1.Enabled := false;
    end;
end;

```

```

GroupBox6.Enabled :=false ;
ComboBox1.Enabled :=true ;
ComboBox1.ItemIndex := -1;
ComboBox1.Text := 'Pilih Topik';
DBGrid1.Enabled := true;
ComboBox1.SetFocus;
Edit1.Text := ('');
SpeedButton7.Enabled := false;
end;
end;

```

Gambar 5.4 Tampilan Edit Topik

Pada menu pertanyaan prosedurnya :

```

procedure TForm3.SpeedButton2Click(Sender: TObject); // tombol
update
begin
  if (MessageDlg('Anda ingin mengupdate pertanyaan: ' +
    IntToStr(ComboBox4.ItemIndex+1) + ' ?', MtConfirmation,
    [mbYes,mbNo], 0) = mrYes) then
    begin
      Q4.Close;
      Q4.SQL.Clear;
      Q4.SQL.Add('UPDATE pertanyaan SET pertanyaan=:pr WHERE
nomer=:no AND kd aspek=:kd');
      Q4.Params[0].AsString := edit2.Text;
      Q4.Params[1].Value := ComboBox4.ItemIndex+1;
      Q4.Params[2].Value := Q2.fieldbyname('kd_aspek').Value;
      Q4.ExecSQL;
      Q2.Close;
      Q2.Open;
      ComboBox4.ItemIndex := -1;
      Edit2.Text := ('');
      ComboBox4.SetFocus;
    end;
end;

```


Daftar Pertanyaan

No. Daftar Pertanyaan

No. ▾

Pertanyaan

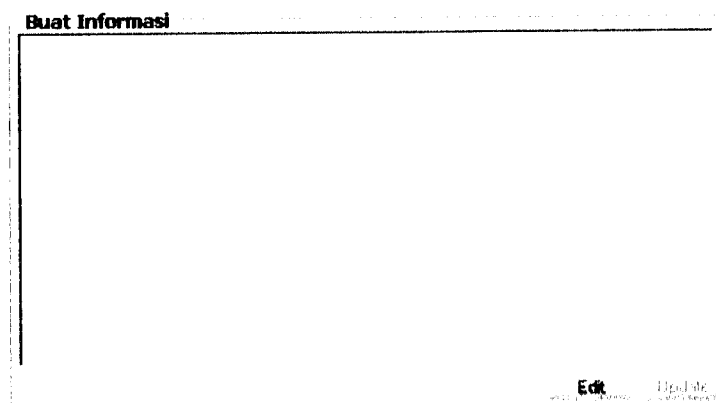
Gambar 5.5 Tampilan Edit Pertanyaan

Pada menu informasi prosedurnya :

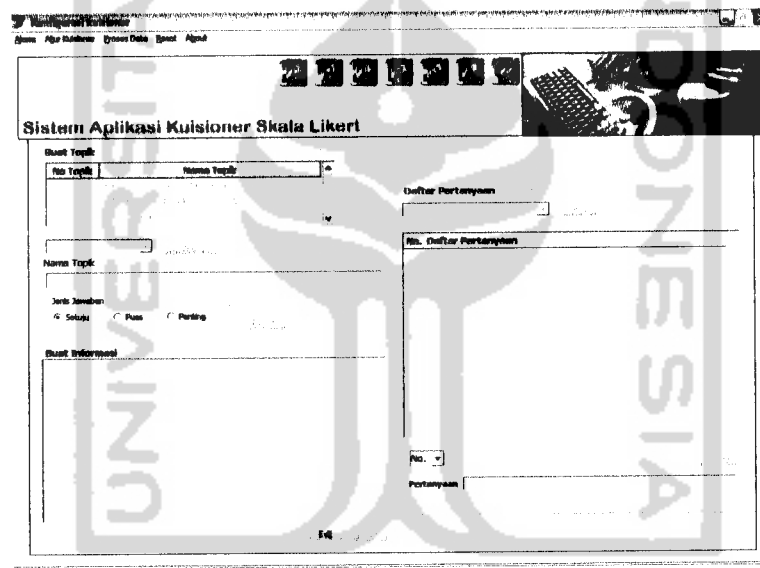
```

procedure TForm3.SpeedButton5Click(Sender: TObject); // tombol
update
begin
    Q4.Close;
    Q4.SQL.Clear;
    Q4.SQL.Add('UPDATE informasi SET informasi= :inf WHERE
kd_inf=:id');
    Q4.params[0].Value := mem01.Text;
    Q4.Params[1].AsString := 'top';
    Q4.ExecSQL;
    Mem01.Enabled := false;
    MessageDlg('Informasi telah sukses di Update', MtInformation,
[mbOk], 0);
    SpeedButton5.Enabled := false;
    SpeedButton4.Enabled := true;
end;

```



Gambar 5.6 Tampilan Edit Informasi



Gambar 5.7 Tampilan Konfigurasi Kuisioner

5.6.1.6 Informasi Responden

Pada menu informasi responden terdapat prosedur untuk menampilkan secara keseluruhan responden yang telah melakukan registrasi serta pengelompokan berdasarkan usia, jenis kelamin dan pekerjaan. Pada menu ini prosedurnya :

```

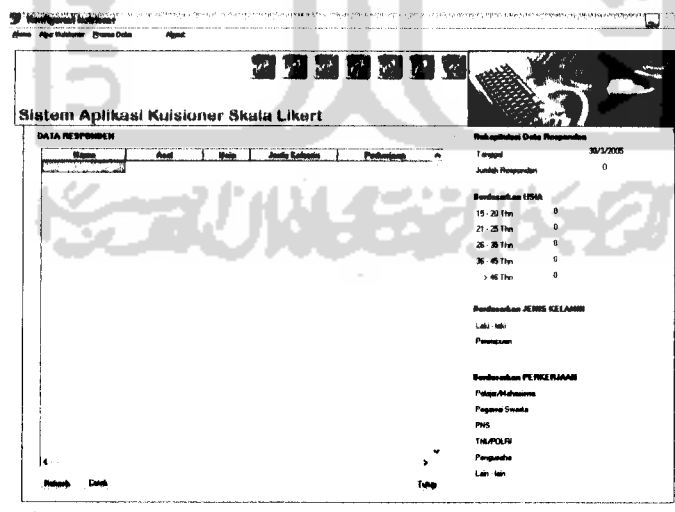
procedure TForm3.InformasiResponden1Click(Sender: TObject);
var tahun, bulan, tanggal: word;
    perintah,tmp: string;
begin
    Reset1.Enabled := false;
    NoteBook1.ActivePage := ('2') ;
    Q6.Close;
    Q6.SQL.Clear;
    Q6.SQL.Add('SELECT nama, asal, umur, jeniskelamin, pekerjaan
FROM          responden,          pekerjaan          WHERE
responden.kd_kerja=pekerjaan.kd_kerja');
    Q6.Open;
    decodedate(date,tahun,bulan,tanggal);
    edit4.Text:= ' ' + intToStr(tanggal) + '/' +
IntToStr(bulan) + '/' + IntToStr(tahun);
    if Q6.RecordCount<>0 then
    begin
        perintah:= 'SELECT COUNT(jeniskelamin) as jumlah FROM
responden';
        edit5.Text:= jumlah(perintah);
        perintah:= 'SELECT COUNT(umur) as jumlah FROM responden where
umur>=15 and umur<=20';
        edit8.Text:= jumlah(perintah);
        perintah:= 'SELECT COUNT(umur) as jumlah FROM responden where
umur>=21 and umur<=25';
        edit9.Text:= jumlah(perintah);
        perintah:= 'SELECT COUNT(umur) as jumlah FROM responden where
umur>=26 and umur<=35';
        edit10.Text:= jumlah(perintah);
        perintah:= 'SELECT COUNT(umur) as jumlah FROM responden where
umur>=36 and umur<=45';
        edit11.Text:= jumlah(perintah);
        perintah:= 'SELECT COUNT(umur) as jumlah FROM responden where
umur>=46';
        edit12.Text:= jumlah(perintah);
        if edit8.Text<>'0' then
Str((StrToInt(edit8.Text)/StrToInt(edit5.Text))*100:2:2,tmp)
        else tmp:= '0';
        edit13.Text:= tmp + ' %';
        if edit9.Text<>'0' then
Str((StrToInt(edit9.Text)/StrToInt(edit5.Text))*100:2:2,tmp)
        else tmp:= '0';
        edit14.Text:= tmp + ' %';
        if edit10.Text<>'0' then
Str((StrToInt(edit10.Text)/StrToInt(edit5.Text))*100:2:2,tmp)
        else tmp:= '0';
        edit15.Text:= tmp + ' %';
        if edit11.Text<>'0' then
Str((StrToInt(edit11.Text)/StrToInt(edit5.Text))*100:2:2,tmp)
        else tmp:= '0';
        edit16.Text:= tmp + ' %';
        if edit12.Text<>'0' then
Str((StrToInt(edit12.Text)/StrToInt(edit5.Text))*100:2:2,tmp)
        else tmp:= '0';
        edit17.Text:= tmp + ' %';
        perintah:= 'SELECT COUNT(jeniskelamin) as jumlah FROM
responden where jeniskelamin='L''';
        edit18.Text:= jumlah(perintah);
    end;
end;

```

```

SpeedButton8.Enabled:= false;
SpeedButton14.Enabled:= false;
Edit5.Text := '';
Edit8.Text := '';
Edit9.Text := '';
Edit10.Text := '';
Edit11.Text := '';
Edit12.Text := '';
Edit13.Text := '';
Edit14.Text := '';
Edit15.Text := '';
Edit16.Text := '';
Edit17.Text := '';
Edit18.Text := '';
Edit19.Text := '';
Edit20.Text := '';
Edit21.Text := '';
Edit22.Text := '';
Edit23.Text := '';
Edit24.Text := '';
Edit25.Text := '';
Edit26.Text := '';
Edit27.Text := '';
Edit28.Text := '';
Edit29.Text := '';
Edit30.Text := '';
Edit31.Text := '';
Edit32.Text := '';
Edit33.Text := '';
end;
end;

```



Gambar 5.8 Tampilan Informasi Responden

5.6.1.7 Frekuensi Data Kuisisioner

Pada menu Frekuensi Data Kuisisioner terdapat prosedur untuk menampilkan secara keseluruhan pilihan responden yang telah melakukan pengisian kuisisioner. Pada menu ini prosedurnya :

```

procedure TForm3.AnalisisKuisisioner1Click(Sender: TObject);
var i: byte;
    tahun, bulan, tanggal: word;
    perintah,tmp: string;
begin
    Reset1.Enabled :=false;
    NoteBook1.ActivePage := ('3');
    decodedate(date,tahun,bulan,tanggal);
    edit34.Text:= intToStr(tanggal) + '/' + IntToStr(bulan) + '/' +
IntToStr(tahun);
    Q3.Close;
    Q3.SQL.Clear;
    Q3.SQL.Add('select NASpek from topik');
    Q3.Open;
    for i:= 1 to Q3.RecordCount do
    begin
        Q3.Next;
    end;
    edit6.Text:=jumlah('SELECT COUNT(jeniskelamin) as jumlah FROM
responden');
    Q7.Close;
    Q7.SQL.Clear;
    Q7.SQL.Add('select distinct NASpek, pertanyaan, n1, n2, n3, n4
from topik, pertanyaan, nilai where nilai.nomer=pertanyaan.nomer
and nilai.kd_aspek=pertanyaan.kd_aspek and
pertanyaan.kd_aspek=topik.kd_aspek and
pertanyaan.pertanyaan<>'');
    Q7.Open;
    if Q7.RecordCount<>0 then
    begin
        perintah:= 'SELECT SUM(n4) as jumlah FROM nilai';
        edit35.Text:= jumlah(perintah);
        perintah:= 'SELECT SUM(n3) as jumlah FROM nilai';
        edit36.Text:= jumlah(perintah);
        perintah:= 'SELECT SUM(n2) as jumlah FROM nilai';
        edit37.Text:= jumlah(perintah);
        perintah:= 'SELECT SUM(n1) as jumlah FROM nilai';
        edit38.Text:= jumlah(perintah);

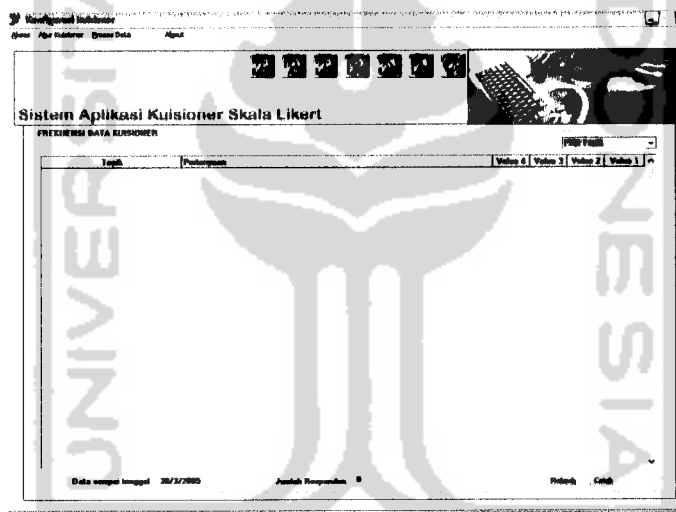
edit43.Text:=IntToStr(StrToInt(edit35.Text)+StrToInt(edit36.Text)+
StrToInt(edit37.Text)+StrToInt(edit38.Text));
        if edit35.Text<>'0' then
Str((StrToInt(edit35.Text)/StrToInt(edit43.Text))*100:2:2,tmp)
        else tmp:= '0';
        edit39.Text:= tmp + ' %';
        if edit36.Text<>'0' then
Str((StrToInt(edit36.Text)/StrToInt(edit43.Text))*100:2:2,tmp)
        else tmp:= '0';

```

```

edit40.Text:= tmp + ' %';
if edit37.Text<>'0' then
Str((StrToInt(edit37.Text)/StrToInt(edit43.Text))*100:2:2,tmp)
else tmp:= '0';
edit41.Text:= tmp + ' %';
if edit38.Text<>'0' then
Str((StrToInt(edit38.Text)/StrToInt(edit43.Text))*100:2:2,tmp)
else tmp:= '0';
edit42.Text:= tmp + ' %';
end
else
begin
MessageDlg('Data kuisioner tidak ada.', MtInformation, [mbOk],
0);
SpeedButton11.Enabled:= false;
SpeedButton15.Enabled:= false;
end;
end;

```



Gambar 5.9 Tampilan Frekuensi Data Kuisioner

5.6.1.8 Interpretasi Kuisioner

Pada menu Interpretasi Kuisioner terdapat prosedur untuk menampilkan interpretasi dari semua pertanyaan yang telah dijawab oleh responden. Pada menu ini prosedurnya :

```

procedure TForm3.LaporanKuisioner1Click(Sender: TObject);
var p,i: integer;
    tmpratan1: real;
    tmpratastr, tmpilihan: string;

```

```

begin
  NoteBook1.ActivePage := ('4');
  Reset1.Enabled := false;
  StrG;
  QN.Close;
  QN.SQL.Clear;
  QN.SQL.Add('SELECT nilai.kd_aspek, pertanyaan, n4, n3, n2, n1,
pilihan FROM nilai, pertanyaan, topik WHERE nilai.kd_aspek =
pertanyaan.kd_aspek AND pertanyaan.kd_aspek = topik.kd_aspek AND
nilai.nomer = pertanyaan.nomer AND pertanyaan <> ''');
  QN.Open;
  if (QN.RecordCount<>0) then
    begin
      Strgr.RowCount:= QN.RecordCount+1;
      for p:=1 to QN.RecordCount do
        begin
          if (QN.fieldByName('pilihan').Value = 1) then
            tmppilihan:= 'Setuju';
          if (QN.fieldByName('pilihan').Value = 2) then
            tmppilihan:= 'Puas';
          if (QN.fieldByName('pilihan').Value = 3) then
            tmppilihan:= 'Penting';
          Strgr.Cells[0,p]:= ' ' +
QN.fieldByName('kd_aspek').AsString;
          Strgr.Cells[1,p]:= ' ' +
QN.fieldByName('pertanyaan').AsString;
          Strgr.Cells[2,p]:= ' ' +
QN.fieldByName('n4').AsString;
          Strgr.Cells[3,p]:= ' ' +
QN.fieldByName('n3').AsString;
          Strgr.Cells[4,p]:= ' ' +
QN.fieldByName('n2').AsString;
          Strgr.Cells[5,p]:= ' ' +
QN.fieldByName('n1').AsString;
          Strgr.Cells[6,p]:= ' ' +
IntToStr((QN.fieldByName('n4').Value*4)+(QN.fieldByName('n3').Valu
e*3)+(QN.fieldByName('n2').Value*2)+QN.fieldByName('n1').Value);
          tmpratanl:=
StrToInt(strgr.Cells[6,p])/(QN.fieldByName('n4').Value +
QN.fieldByName('n3').Value + QN.fieldByName('n2').Value +
QN.fieldByName('n1').Value);
          Str(tmpratanl:2:2,tmpratastr);
          Strgr.Cells[7,p]:= ' ' + tmpratastr;
          if ((tmpratanl >= 1) and (tmpratanl <= 1.75)) then
            Strgr.Cells[8,p]:= ' Tidak ' + tmppilihan;
          if ((tmpratanl > 1.75) and (tmpratanl <= 2.50)) then
            Strgr.Cells[8,p]:= ' Kurang ' + tmppilihan;
          if ((tmpratanl > 2.50) and (tmpratanl <= 3.25)) then
            Strgr.Cells[8,p]:= ' ' + tmppilihan;
          if (tmpratanl > 3.25) then Strgr.Cells[8,p]:= '
Sangat ' + tmppilihan;
          QN.Next;
        end;
      Q8.Close;
      Q8.SQL.Clear;
      Q8.SQL.Add('DELETE FROM tmp');
      Q8.ExecSQL;
    end;

```

```

for i:=1 to strgr.RowCount-1 do
  begin
    Q8.Close;
    Q8.SQL.Clear;
    Q8.SQL.Add('INSERT INTO tmp (A, B, C, D, E, F, G, H,
I) VALUES (:a, :b, :c, :d, :e, :f, :g, :h, :i)');
    Q8.Params[0].AsString:= strgr.Cells[0,i];
    Q8.Params[1].AsString:= strgr.Cells[1,i];
    Q8.Params[2].AsString:= strgr.Cells[2,i];
    Q8.Params[3].AsString:= strgr.Cells[3,i];
    Q8.Params[4].AsString:= strgr.Cells[4,i];
    Q8.Params[5].AsString:= strgr.Cells[5,i];
    Q8.Params[6].AsString:= strgr.Cells[6,i];
    Q8.Params[7].AsString:= strgr.Cells[7,i];
    Q8.Params[8].AsString:= strgr.Cells[8,i];
    Q8.ExecSQL;
  end;
end
else
  begin
    Strgr.RowCount:= 2;
    MessageDlg('Data kuisisioner tidak ada.', MtInformation,
[mbOk], 0);
    Q8.Close;
    Q8.SQL.Clear;
    Q8.SQL.Add('DELETE FROM tmp');
    Q8.ExecSQL;
    for i:=1 to strgr.RowCount-1 do
      begin
        Q8.Close;
        Q8.SQL.Clear;
        Q8.SQL.Add('INSERT INTO tmp (A, B, C, D, E, F, G, H, I)
VALUES (:a, :b, :c, :d, :e, :f, :g, :h, :i)');
        Q8.Params[0].AsString:= strgr.Cells[0,i];
        Q8.Params[1].AsString:= strgr.Cells[1,i];
        Q8.Params[2].AsString:= strgr.Cells[2,i];
        Q8.Params[3].AsString:= strgr.Cells[3,i];
        Q8.Params[4].AsString:= strgr.Cells[4,i];
        Q8.Params[5].AsString:= strgr.Cells[5,i];
        Q8.Params[6].AsString:= strgr.Cells[6,i];
        Q8.Params[7].AsString:= strgr.Cells[7,i];
        Q8.Params[8].AsString:= strgr.Cells[8,i];
        Q8.ExecSQL;
      end;
    SpeedButton3.Enabled:= false;
    SpeedButton16.Enabled:= false;
  end;
end;
end;

```



```

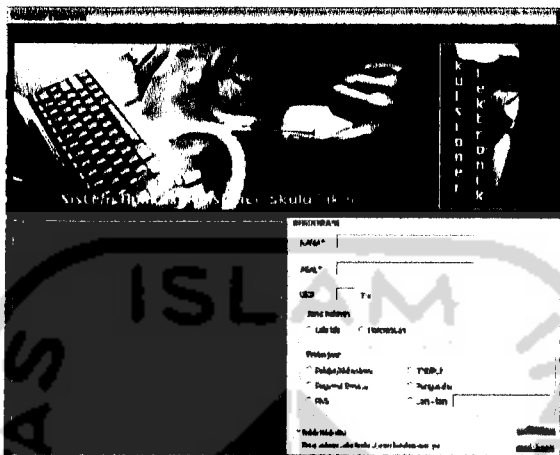
Q2.Params[0].AsString:=edit3.Text;
Q2.Params[1].AsString:=edit2.Text;
Q2.Open;
if Q2.RecordCount=0 then
begin
Q1.Close;
Q1.SQL.Clear;
Q1.SQL.Add('INSERT INTO responden (nama, asal, umur,
jeniskelamin, kd_kerja) VALUES (:na, :asl, :um, :jk, :pr)');
Q1.Params[0].AsString := Edit3.Text;
Q1.Params[1].AsString := Edit2.Text;
Q1.Params[2].Value := StrToInt(Edit1.Text);
  if (R1.Checked = true) then Q1.Params[3].Value :=
'L';
  if (R2.Checked = true) then Q1.Params[3].Value :=
'P';
  if (RP1.Checked = true) then Q1.Params[4].AsString
:= '1';
  if (RP2.Checked = true) then Q1.Params[4].AsString
:= '2';
  if (RP3.Checked = true) then Q1.Params[4].AsString
:= '3';
  if (RP4.Checked = true) then Q1.Params[4].AsString
:= '4';
  if (RP5.Checked = true) then Q1.Params[4].AsString
:= '5';
  if (RP6.Checked = true) then Q1.Params[4].AsString
:= '6';
Q1.ExecSQL;
Edit1.Clear;
Edit2.Clear;
Edit3.Clear;
R1.Checked := false;
R2.Checked := false;
RP1.Checked := false;
RP2.Checked := false;
RP3.Checked := false;
RP4.Checked := false;
RP5.Checked := false;
RP6.Checked := false;
Edit3.SetFocus;
Form2.ShowModal;
Edit4.Text := '';
end
else
  messagedlg('Nama dan asal anda sudah terdaftar!',
MtWarning, [mbOk], 0);
end;
end;
end
else
begin
  messagedlg('Anda belum memasukkan umur', MtWarning,
[mbOk], 0);
  Edit3.SetFocus;
end;
end
else

```

```

begin
    messagedlg('Tidak ada pengisian kuisioner', MtWarning,
[mbOk], 0);
end;
end;

```



Gambar 5.11 Tampilan Awal (Responden)

5.6.2.2 Kuisisioner

Pada menu Kuisisioner digunakan untuk menampilkan pertanyaan yang akan dijawab oleh responden. Pada menu ini prosedurnya :

```

procedure TForm2.SpeedButton1Click(Sender: TObject);
var validasi: boolean;
    perintah: string;
    i: byte;
begin
    validasi:= true;
    If ((R1.Visible = true) and (R1.ItemIndex = -1)) then validasi:=
false;
    If ((R2.Visible = true) and (R2.ItemIndex = -1)) then validasi:=
false;
    If ((R3.Visible = true) and (R3.ItemIndex = -1)) then validasi:=
false;
    If ((R4.Visible = true) and (R4.ItemIndex = -1)) then validasi:=
false;
    If ((R5.Visible = true) and (R5.ItemIndex = -1)) then validasi:=
false;
    If ((R6.Visible = true) and (R6.ItemIndex = -1)) then validasi:=
false;
    If ((R7.Visible = true) and (R7.ItemIndex = -1)) then validasi:=
false;
    If ((R8.Visible = true) and (R8.ItemIndex = -1)) then validasi:=
false;

```

```

    If ((R9.Visible = true) and (R9.ItemIndex = -1)) then validasi:=
false;
    If ((R10.Visible = true) and (R10.ItemIndex = -1)) then
validasi:= false;
    If (validasi= false) then MessageDlg('Anda belum menjawab semua
pertanyaan', mtWarning, [mbOK], 0)
    Else
        begin
            if (messagedlg('Anda yakin dengan opsi pilihan untuk
topik ini?', mtconfirmation, [mbyes,mbno], 0) = mryes) then
                begin
                    masukkannilai(R1,1,kaspek);
                    masukkannilai(R2,2,kaspek);
                    masukkannilai(R3,3,kaspek);
                    masukkannilai(R4,4,kaspek);
                    masukkannilai(R5,5,kaspek);
                    masukkannilai(R6,6,kaspek);
                    masukkannilai(R7,7,kaspek);
                    masukkannilai(R8,8,kaspek);
                    masukkannilai(R9,9,kaspek);
                    masukkannilai(R10,10,kaspek);
                    inc(kaspek);
                    perintah:='SELECT * FROM pertanyaan WHERE kd_aspek='
+IntToStr(kaspek) + ' and pertanyaan<>''''';
                    select2(perintah);
                    if (Q2.RecordCount<>0) then
                        begin
                            perintah:='select naspek,pilihan from topik where
kd_aspek=' + IntToStr(kaspek);
                            select1(perintah);
                            Panel2.Caption:=Q1.fieldbyname('naspek').AsString;
                            hilang;
                            for i:=1 to Q2.RecordCount do
                                begin
                                    tampilan(Q1.FieldName('pilihan').Value);
                                    seting(i, IntToStr(i) + ' ' +
Q2.fieldbyname('pertanyaan').AsString);
                                    Q2.Next;
                                end;
                            end;
                        end
                    else
                        begin
                            Messagedlg('Terima kasih atas partisipasi Anda
dalam mengisi kuisisioner ini', mtInformation, [mbOk], 0);
                            close();
                        end;
                    end;
                end;
            end;
        end;
    end;
end;

```

FORMA PENGISIAN KUISIONER SKALA LIKERT

Alternatif Jawaban

SP P KP TP

SP P KP TP

SP P KP TP

SP P KP TP

SP P KP TP

SP P KP TP

SP P KP TP

SP P KP TP

SP P KP TP

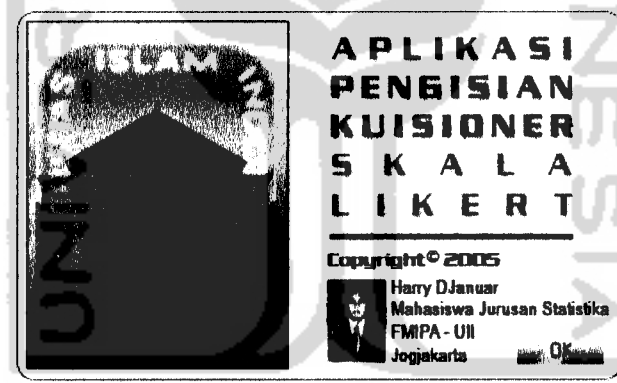
SP P KP TP

Bat = Baik SB = Sangat Baik S = Baik KB = Kurang Baik TB = Tidak Baik
 SP = Sangat Paus P = Paus KP = Kurang Paus TP = Tidak Paus
 SP = Sangat Pering P = Pering KP = Kurang Pering TP = Tidak Pering

Gambar 5.12 Tampilan Kuisisioner

5.6.3 Tampilan Pembuat

Pada tampilan pembuat digunakan untuk mengetahui pembuat sistem aplikasi.



Gambar 5.13 Tampilan Pembuat

BAB VI

ANALISIS KINERJA PERANGKAT LUNAK

6.1 Pengujian Program

Pemrograman merupakan kegiatan menulis kode program yang akan dieksekusi oleh komputer. Kode program yang ditulis oleh program harus berdasarkan dokumentasi yang disediakan oleh analisis sistem, dari hasil rancangan sistem secara rinci. Sebelum program diterapkan maka program harus bebas dari kesalahan-kesalahan. Oleh karena itu program harus diuji. Pengujian program perlu dilakukan sebelum program tersebut diterapkan kedalam lingkungan yang sebenarnya. Pengujian tersebut dilakukan untuk menemukan kesalahan-kesalahan yang mungkin terjadi.

6.2 Pengujian Dan Analisis

Dalam tahap ini akan dijelaskan tentang pengujian program aplikasi yang digunakan pada Sistem Aplikasi Pengisian Kuisisioner Skala Likert Beserta Interpretasi, dengan pengujian ini diharapkan tingkat kesalahan baik dalam pengolahan data maupun sistem itu sendiri menjadi sangat minim bahkan tidak ada. Pada pengujian ini, kami mengambil contoh kuisisioner yang menjadi kuisisioner dalam tugas akhir Akhmadi tentang faktor-faktor yang mempengaruhi pelanggan FMIPA UII. Pengujian kinerja perangkat lunak di bagi menjadi 2 bagian yaitu uji normal dan uji tak normal.

6.2.1 Uji Normal

Pengujian normal (*normal testing*) dilakukan dengan memberikan masukan yang menurut spesifikasi awal.

a. Pengujian Admin

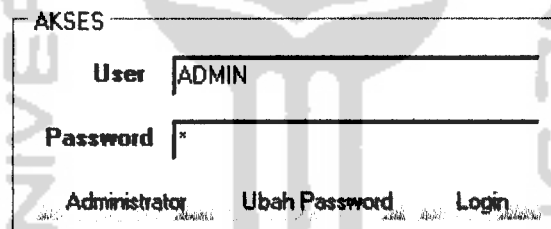
1. Input Menu Login

Pada menu login admin diberikan contoh masukan atau *input* untuk menguji keluaran atau *output* yang dihasilkan. Contoh data login admin :

Username :ADMIN

Password :1

Hasil dari masukan data login admin tersebut dapat dilihat pada gambar 6.1.



Gambar 6.1 Menu Login Admin

2. Input Konfigurasi Data Kuisisioner

Pada menu konfigurasi kuisisioner pemasukan data dibagi menjadi 3 input yaitu input topik, input pertanyaan dan input informasi. Untuk input topik diambil contoh :

No Topik : 1

Nama Topik : ASPEK PELAYANAN

Jenis Jawaban : Puas

Hasil input topik dapat dilihat pada gambar 6.2

No Topik	Nama Topik
1	ASPEK PELAYANAN
2	ASPEK FASILITAS
3	ASPEK DOSEN

Pilih Topik: [v] Edit

Nama Topik: _____

Jenis Jawaban: Sebaiknya Puas Penting Update

Gambar 6.2 Menu Input Topik

Untuk input pertanyaan diambil contoh :

Pilih Topik : ASPEK PELAYANAN

No Pertanyaan : 1

Pertanyaan : Kerapihan penampilan karyawan

Hasil input pertanyaan dapat dilihat pada gambar 6.3

No.	Daftar Pertanyaan
1	Kerapihan penampilan karyawan
2	Kesesuaian jam kerja karyawan
3	Hasil kerja Karyawan dapat dipercaya
4	Ketepatan pemenuhan janji oleh Karyawan
5	Keaktifan memberikan informasi Akademik yang baru
6	Keramahan karyawan
7	Cepat tanggapnya Karyawan memenuhi kebutuhan
8	
9	
10	

No. [v] Update

Pertanyaan: _____

Gambar 6.3 Menu Input Pertanyaan

Untuk input Informasi diambil contoh :

Informasi : Assalamu'alaikum, Wr. Wb.

Hasil input informasi dapat dilihat pada gambar 6.4

Buat Informasi

ASSALAMU'ALAIKUM WR. WB.

Rekan-rekan Mahasiswa yang Terhormat,

Kepuasan, kenyamanan dan kemudahan layanan bagi mahasiswa adalah tujuan bagi setiap Fakultas dan akan terus menerus harus diupayakan oleh Fakultas hingga benar-benar mempunyai kualitas yang baik. Agar supaya peningkatan kualitas Layanan Fakultas lebih bermanfaat bagi mahasiswa, saya mohon kesediaan rekan-rekan Mahasiswa untuk memberikan pendapat mengenai layanan yang ada di Fakultas MIPA dengan cara mengisi angket, yang akan kami jaga kerahasiannya.

Atas perhatian dan kerjasamanya, kami ucapkan terimakasih.

WASSALAMU'ALAIKUM WR. WB.

Hormat Kami,

Edi Update

Gambar 6.4 Menu Input Informasi

3. Menu Data Responden

Pada menu ini, Admin dapat melakukan rekapitulasi data responden yang mengisi kuisisioner meliputi nama, asal, umur, jenis kelamin dan pekerjaan. Hasil data responden dapat dilihat pada gambar 6.5 :

No	Nama	Jk	Umur	Jenis Kelamin	Pekerjaan	Tampil
21		P			Polisi/Polisman	
22		L			Polisi/Polisman	
26		P			Polisi/Polisman	
21		P			Polisi/Polisman	
27		L			Polisi/Polisman	
19		L			Polisi/Polisman	
19		L			Polisi/Polisman	
21		P			Polisi/Polisman	
26		P			Polisi/Polisman	
20		P			Polisi/Polisman	
23		P			Polisi/Polisman	
23		P			Polisi/Polisman	
22		L			Polisi/Polisman	
22		L			Polisi/Polisman	
19		L			Polisi/Polisman	
19		L			Polisi/Polisman	
19		L			Polisi/Polisman	
20		P			Polisi/Polisman	
30		P			Polisi/Polisman	
21		P			Polisi/Polisman	
21		P			Polisi/Polisman	
30		P			Polisi/Polisman	
30		P			Polisi/Polisman	
21		P			Polisi/Polisman	

Rekapitulasi Data Responden	
Tanggal	24/4/2015
Jumlah Responden	100
Berdasarkan USIA	
15 - 20 Th	54 54.00 %
21 - 25 Th	46 46.00 %
26 - 30 Th	0 0 %
31 - 40 Th	0 0 %
> 40 Th	0 0 %
Berdasarkan JENIS KELAMIN	
Laki-laki	36 36.00 %
Perempuan	64 64.00 %
Berdasarkan PEKERJAAN	
Polisi/Polisman	100 100.00 %
Pegawai Swasta	0 0 %
PHS	0 0 %
UMK/UMK	0 0 %
Pengusaha	0 0 %
Lain-lain	0 0 %

Gambar 6.5 Menu Data Responden

4. Menu Frekuensi Data Kuisiomer

Pada menu ini, Admin dapat merekapitulasi frekuensi data kuisiomer yang telah diisi oleh responden. Hasil menu frekuensi data kuisiomer dapat dilihat pada gambar 6.6 :

Aspek	Frekuensi	Persentase
ASPEK PELAYANAN	3	63
ASPEK PELAYANAN	0	95
ASPEK PELAYANAN	1	95
ASPEK PELAYANAN	3	41
ASPEK PELAYANAN	26	31
ASPEK PELAYANAN	4	57
ASPEK PELAYANAN	1	50
ASPEK FASILITAS	7	61
ASPEK FASILITAS	7	48
ASPEK FASILITAS	11	68
ASPEK FASILITAS	7	64
ASPEK FASILITAS	4	70
ASPEK FASILITAS	4	46
ASPEK FASILITAS	7	95
ASPEK DOSEN	2	53
ASPEK DOSEN	2	37
ASPEK DOSEN	2	52
ASPEK DOSEN	3	95
ASPEK DOSEN	3	44
ASPEK DOSEN	3	52
ASPEK DOSEN	4	94

Jumlah Responden : 100
 Total Frekuensi : 2100
 Nilai 4 : 184 4.95 %
 Nilai 3 : 1152 54.86 %
 Nilai 2 : 659 31.38 %
 Nilai 1 : 185 8.81 %

Gambar 6.6 Menu Frekuensi Data Kuisiomer

5. Menu Interpretasi Data Kuisiomer

Pada menu ini, admin dapat menginterpretasi kuisiomer yang dalam proses telah dilakukan perhitungan interpretasi. Hasil menu interpretasi data kuisiomer dapat dilihat pada gambar 6.7 :

Sistem Aplikasi Kuisisioner Skala Likert

Pilih, Print, Refresh, Delete

INTERPRETASI DATA KUISIONER									
No	Kategori	VM1	VM2	VM3	VM4	VM5	Total	Rata-rata	Kategori
1	Kemampuan pemrosesan informasi	3	53	29	5	254	244		Puas
1	Kemampuan pemrosesan informasi	0	56	41	4	251	251		Puas
1	Hard disk komputer cepat dimuncikan	1	56	33	11	246	246		Kurang Puas
1	Kemampuan pemrosesan yang lebih cepat	3	41	47	9	238	238		Kurang Puas
1	Kemampuan pemrosesan informasi Aljabar yang baru	26	31	28	15	268	268		Puas
1	Kemampuan pemrosesan informasi	4	57	31	8	257	257		Puas
1	Cepat tanggapnya respon terhadap kebutuhan	1	55	36	8	249	249		Kurang Puas
2	Kemampuan yang lebih	7	51	23	9	266	266		Puas
2	Kemampuan yang lebih	7	49	31	13	250	250		Kurang Puas
2	Kemampuan komputer untuk SIMAK	11	68	13	8	260	262		Puas
2	Kemampuan Lahan dan	7	64	26	3	275	275		Puas
2	Kemampuan pemrosesan informasi	4	70	15	11	267	267		Puas
2	Kemampuan dalam pemrosesan SP3	4	49	38	10	246	246		Kurang Puas
2	Pelajaran dan komputer dari area kerja	7	56	19	18	252	252		Puas
3	Kemampuan dalam pemrosesan	2	53	34	11	246	246		Kurang Puas
3	Pemrosesan dalam pemrosesan	2	57	38	3	258	258		Puas
3	Kemampuan pemrosesan dengan lebih	2	62	29	7	259	259		Puas
3	Kemampuan dalam pemrosesan	3	55	39	3	258	258		Puas
3	Kemampuan pemrosesan data IAS	3	44	40	13	237	237		Kurang Puas
3	Kemampuan pemrosesan informasi	3	45	36	6	207	183		Kurang

Gambar 6.7 Menu Interpretasi Data Kuisisioner

b. Pengujian Responden

1. Input Registrasi Responden

Pada menu ini, responden melakukan registrasi yaitu input nama, asal, umur, jenis kelamin dan jenis pekerjaan. Untuk input registrasi responden diambil contoh :

Umur : 21 thn

Jenis Kelamin : Perempuan

Perkerjaan : Pelajar/Mahasiswa

Hasil menu registrasi responden dapat dilihat pada gambar 6.8 :

REGISTRASI

NAMA*

ASAL*

USIA Thn

Jenis Kelamin
 Laki-laki Perempuan

Perkerjaan
 Pelajar/Mahasiswa TNI/Polri
 Pegawai Swasta Pengusaha
 PNS Lain - lain

*Boleh tidak diisi
 Kerahasiaan data Anda dijamin kerahasiaannya

Gambar 6.8 Menu Registrasi Responden

2. Input Kuisiонер

Pada menu ini, responden melakukan pengisian kuisiонер yang telah disetup oleh admin. Hasil menu input kuisiонер dapat dilihat pada gambar 6.9 :

ASPEK PELAYANAN

Alternatif Jawaban

1. Ketepatan penempatan karyawan	<input type="radio"/> C	<input type="radio"/> SP	<input type="radio"/> P	<input type="radio"/> KP	<input type="radio"/> TP
2. Kesesuaian jam kerja karyawan	<input type="radio"/> C	<input type="radio"/> SP	<input type="radio"/> P	<input type="radio"/> KP	<input type="radio"/> TP
3. Hasil kerja Karyawan dapat dipercaya	<input type="radio"/> C	<input type="radio"/> SP	<input type="radio"/> P	<input type="radio"/> KP	<input type="radio"/> TP
4. Ketepatan pemenuhan janji oleh Karyawan	<input type="radio"/> C	<input type="radio"/> SP	<input type="radio"/> P	<input type="radio"/> KP	<input type="radio"/> TP
5. Keaktifan memberikan informasi Akademi yang baru	<input type="radio"/> C	<input type="radio"/> SP	<input type="radio"/> P	<input type="radio"/> KP	<input type="radio"/> TP
6. Keramahan karyawan	<input type="radio"/> C	<input type="radio"/> SP	<input type="radio"/> P	<input type="radio"/> KP	<input type="radio"/> TP
7. Dapat tanggapnya Karyawan memenuhi kebutuhan	<input type="radio"/> C	<input type="radio"/> SP	<input type="radio"/> P	<input type="radio"/> KP	<input type="radio"/> TP

Ket: SS - Sangat Sesuai S - Sesuai KS - Kurang Sesuai TS - Tidak Sesuai
 SP - Sangat Puan P - Puan KP - Kurang Puan TP - Tidak Puan
 SPy - Sangat Pening Py - Pening KPy - Kurang Pening TPy - Tidak Pening

Ditambah dengan honor..._J

Gambar 6.9 Menu Input Kuisiонер

c. Pengujian Operator

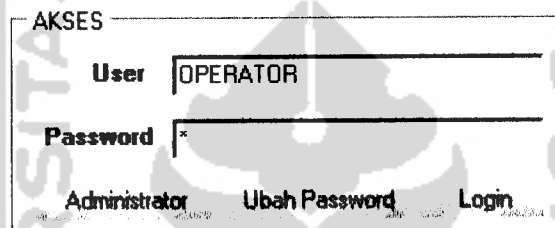
1. Input Menu Login

Pada menu login operator diberikan contoh masukan atau *input* untuk menguji keluaran atau *output* yang dihasilkan. Contoh data login admin :

Username : OPERATOR

Password : 2

Hasil dari masukan data login operator tersebut dapat dilihat pada gambar 6.10 :



AKSES

User OPERATOR

Password *

Administrator Ubah Password Login

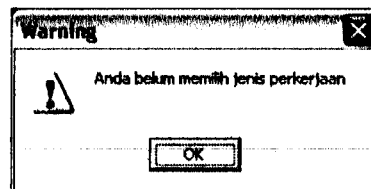
Gambar 6.10 Menu Login Operator

2. Menu Data Responden

Pada menu ini, operator dapat melakukan rekapitulasi data responden yang mengisi kuisisioner meliputi nama, asal, umur, jenis kelamin dan pekerjaan. Hasil data responden dapat dilihat pada gambar 6.5.

3. Menu Frekuensi Data Kuisisioner

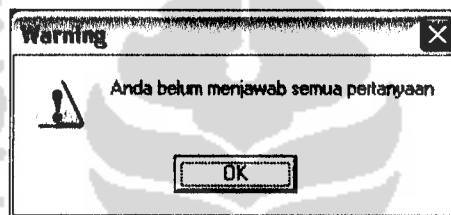
Pada menu ini, operator dapat merekapitulasi frekuensi data kuisisioner yang telah diisi oleh responden. Hasil menu frekuensi data kuisisioner dapat dilihat pada gambar 6.6 .



Gambar 6.13 Pesan Kesalahan Registrasi (Perkerjaan)

2. Input Kuisisioner

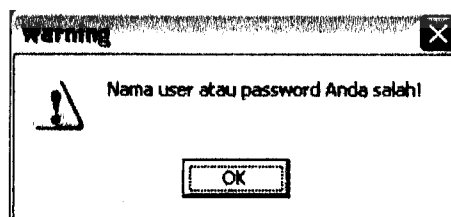
Pada menu ini, apabila responden tidak menjawab semua pertanyaan maka akan ditampilkan pesan peringatan seperti pada gambar 6.14.



Gambar 6.14 Pesan Kesalahan Mengisi Kuisisioner

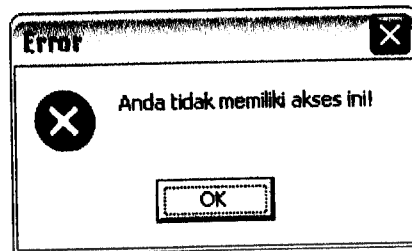
3. Input Menu Login

Pada menu login, apabila *username* dan *password* tidak cocok maka akan ditampilkan pesan peringatan seperti pada gambar 6.15



Gambar 6.15 Pesan Kesalahan Login

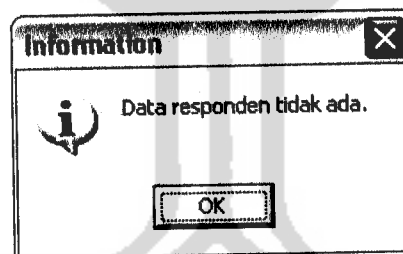
Pada menu login, apabila operator memilih menu administrator maka akan ditampilkan pesan kesalahan seperti pada gambar 6.16.



Gambar 6.16 Pesan Kesalahan Akses

4. Menu Data Responden

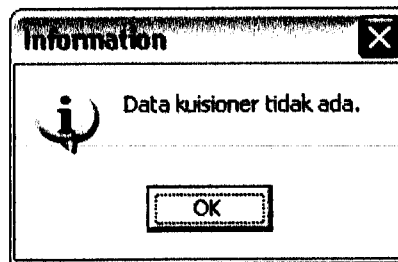
Dalam proses rekapitulasi, apabila data responden tidak ada maka akan muncul pesan informasi seperti pada gambar 6.17.



Gambar 6.17 Pesan Tidak Ada Responden

5. Menu Frekuensi Data Kuisiner

Dalam proses rekapitulasi frekuensi, apabila data kuisiner tidak ada maka akan muncul pesan informasi seperti pada gambar 6.18.



Gambar 6.18 Pesan Tidak Ada Kuisisioner

6.3 Hasil Keluaran Sistem

Setelah sistem dicoba untuk diuji, maka didapat hasil keluaran sistem yang berupa laporan. Laporan yang dikeluarkan sistem terdiri dari Laporan Data Responden, Laporan Frekuensi Data Kuisisioner dan Laporan Interpretasi Kuisisioner.

Tanggal : 4/24/2005
Jumlah Responden : 100

Jam : 7:01:47 PM

DATA RESPONDEN

Berdasarkan	Jenis	Jumlah	Persentase
Usia	15 - 20 Thn	54	54.00 %
Usia	21 - 25 Thn	46	46.00 %
Usia	26 - 35 Thn	0	0 %
Usia	36 - 45 Thn	0	0 %
Usia	46 Thn Keatas	0	0 %
Berdasarkan	Jenis	Jumlah	Persentase
Jenis Kelamin	Laki-laki	36	36.00 %
Jenis Kelamin	Perempuan	64	64.00 %
Berdasarkan	Jenis	Jumlah	Persentase
Perkerjaan	Pelajar/Mahasiswa	100	100.00 %
Perkerjaan	Pegawai Swasta	0	0 %
Perkerjaan	PNS	0	0 %
Perkerjaan	TNI/POLRI	0	0 %
Perkerjaan	Pengusaha	0	0 %
Perkerjaan	Lain-lain	0	0 %

Gambar 6.19 Laporan Data Responden

Tanggal : 4/25/2005

Jam : 2:24:30 AM

Jumlah Responden : 100

FREKUENSI DATA KUISIONER

Aspek	Pernyataan	1	2	3	4	5
ASPEK PELAYANAN	Kerapuhan penampilan karyawan	3	63	29	5	
ASPEK PELAYANAN	Kesesuaian jam kerja karyawan	0	55	41	4	
ASPEK PELAYANAN	Hasil kerja Karyawan dapat dipercaya	1	55	33	11	
ASPEK PELAYANAN	Ketepatan pemenuhan janji oleh Karyawan	3	41	47	9	
ASPEK PELAYANAN	Keaktifan memberikan informasi Akademik yang baru	26	31	28	15	
ASPEK PELAYANAN	Keramahan karyawan	4	57	31	8	
ASPEK PELAYANAN	Cepat tanggapnya Karyawan memenuhi kebutuhan	1	55	36	8	
ASPEK FASILITAS	Kebersihan ruang kuliah	7	61	23	9	
ASPEK FASILITAS	Kebersihan kamar mandi/toilet	7	49	31	13	
ASPEK FASILITAS	Kelengkapan komputer untuk SIMAK	11	68	13	8	
ASPEK FASILITAS	Kelengkapan Laboratorium	7	64	26	3	
ASPEK FASILITAS	Kelengkapan perpustakaan Fakultas	4	70	15	11	
ASPEK FASILITAS	Kemudahan dalam pembayaran SPP	4	48	38	10	
ASPEK FASILITAS	Pelayanan bus kampus dan area parkir	7	56	19	18	
ASPEK DOSEN	Ketepatan waktu mengajar	2	53	34	11	
ASPEK DOSEN	Penjelasan materi waktu mengajar	2	57	38	3	
ASPEK DOSEN	Kesesuaian mengajar dengan silabus	2	62	29	7	
ASPEK DOSEN	Ketepatan waktu praktikum	3	55	39	3	
ASPEK DOSEN	Kecepatan memberikan nilai UAS	3	44	40	13	
ASPEK DOSEN	Kemudahan dalam bimbingan (termasuk KP/TA)	3	52	39	6	
ASPEK DOSEN	Kemudahan pertemuan dengan pihak Dekanat	4	56	30	10	

Gambar 6.20 Laporan Frekuensi Data Kuisisioner

Tanggal : 4/25/2005

Jam : 2:33:55 AM

Jumlah Responden : 100

INTERPRETASI KUISIONER

Aspek	Pernyataan	1	2	3	4	5	Mean	Standar Deviasi	Kategori
1	Kerapuhan penampilan karyawan	3	63	29	5	294	2.94		Ruas
1	Kesesuaian jam kerja karyawan	0	55	41	4	251	2.51		Ruas
1	Hasil kerja Karyawan dapat dipercaya	1	55	33	11	240	2.40		Kurang Ruas
1	Ketepatan pemenuhan janji oleh Karyawan	3	41	47	9	238	2.38		Kurang Ruas
1	Keaktifan memberikan informasi Akademik yang baru	26	31	28	15	208	2.08		Ruas
1	Keramahan karyawan	4	57	31	8	267	2.67		Ruas
1	Cepat tanggapnya Karyawan memenuhi kebutuhan	1	55	36	8	240	2.40		Kurang Ruas
2	Kebersihan ruang kuliah	7	61	23	9	288	2.88		Ruas
2	Kebersihan kamar mandi/toilet	7	49	31	13	280	2.80		Kurang Ruas
2	Kelengkapan komputer untuk SIMAK	11	68	13	8	282	2.82		Ruas
2	Kelengkapan Laboratorium	7	64	26	3	275	2.75		Ruas
2	Kelengkapan perpustakaan Fakultas	4	70	15	11	287	2.87		Ruas
2	Kemudahan dalam pembayaran SPP	4	48	38	10	248	2.48		Kurang Ruas
2	Pelayanan bus kampus dan area parkir	7	56	19	18	252	2.52		Ruas
3	Ketepatan waktu mengajar	2	53	34	11	240	2.40		Kurang Ruas
3	Penjelasan materi waktu mengajar	2	57	38	3	258	2.58		Ruas
3	Kesesuaian mengajar dengan silabus	2	62	29	7	269	2.69		Ruas
3	Ketepatan waktu praktikum	3	55	39	3	258	2.58		Ruas
3	Kecepatan memberikan nilai UAS	3	44	40	13	237	2.37		Kurang Ruas
3	Kemudahan dalam bimbingan (termasuk KP/TA)	3	52	39	6	252	2.52		Ruas
3	Kemudahan pertemuan dengan pihak Dekanat	4	56	30	10	254	2.54		Ruas

Gambar 6.21 Laporan Interpretasi Kuisisioner

BAB VII

PENUTUP

7.1 Kesimpulan

Setelah menyelesaikan pembuatan Sistem Aplikasi Pengisian Kuisisioner Skala Likert Beserta Interpretasi, maka dapat diambil kesimpulan sebagai berikut :

1. Sistem Aplikasi Pengisian Kuisisioner Skala Likert Beserta Interpretasi yang mempunyai fasilitas konfigurasi kuisisioner, pengisian kuisisioner oleh responden, informasi data responden, frekuensi data kuisisioner dan perhitungan interpretasi, dapat menjadi salah satu alternatif dalam penyebaran kuisisioner.
2. Laporan yang dihasilkan memberikan informasi yang lengkap mengenai informasi data responden, frekuensi data kuisisioner dan data perhitungan interpretasi.

7.2 Saran-saran

Dengan ini disadari bahwa Sistem Aplikasi Pengisian Kuisisioner Skala Likert Beserta Interpretasi ini masih sederhana dan terbatas, banyak kekurangan dan kelemahannya, maka disarankan:

1. Sistem ini dapat dikembangkan untuk kuisisioner yang jumlah pertanyaan (lebih dari 10 pertanyaan tiap topik) dan topik lebih banyak (lebih dari 5 topik).

2. Sistem ini hanya dapat digunakan untuk kuisioner yang skala pengukurannya ialah skala Likert sehingga perlu dikembangkan agar sistem ini dapat digunakan oleh kuisioner dengan semua skala pengukuran.
3. Sistem ini juga dapat dapat dikembangkan agar sistem ini dapat diaplikasikan dalam jaringan internet.



DAFTAR PUSTAKA

- Alam, M Agus J. *Belajar Sendiri Borland Delphi 5.0*. Jakarta: PT Elex Media Komputindo. 2000.
- Fathansyah. *Basis Data*. Bandung: CV. Informatika. 1999.
- Martina, I. *36 Jam Belajar Komputer Delphi 5.0*. Jakarta: PT Elex Media Komputindo. 2000.
- Martina, I. *36 Jam Belajar Komputer Pemrograman Visual Borland Delphi 7*. Jakarta: PT Elex Media Komputindo. 2004.
- Jogiyanto HM. *Analisis dan Desain Sistem Informasi : Pendekatan Terstruktur Teori dan Praktek Aplikasi Bisnis*. Yogyakarta: Andi Offset. 1999.
- Kristanto, Harianto. *Konsep dan Perancangan Database*. Yogyakarta: Andi Offset. 1999.
- Syarif, Arry Maulana. *Cara Cepat Membuat Animasi FLASH Menggunakan SWISHmax*. Yogyakarta: AndiOffset. 2005.
- Simamora, B. *Panduan Riset Perilaku Konsumen*. Jakarta: Gramedia Pustaka Utama. 2002.
- Akhmadi. *Membuat Perangkat Lunak Untuk Sistem Informasi Manajemen Mutu Di Universitas Islam Indonesia*. Tugas Akhir. Fakultas Matematika dan Ilmu Pengetahuan Alam. Yogyakarta. 2003.
- Nurchahyo, Arif. *Membangun Sistem Informasi Data Pelanggan PDAM Beserta Perhitungan Efisiensi Debit Air*. Tugas Akhir. Fakultas Teknologi Industri. Yogyakarta. 2003.



LAMPIRAN

1. Daftar Isi

DATA RESPONDEN

Berdasarkan	Jenis	Jumlah	Persentase
Usia	15 - 20 Thn	54	54.00 %
Usia	21 - 25 Thn	46	46.00 %
Usia	26 - 35 Thn	0	0 %
Usia	36 - 45 Thn	0	0 %
Usia	46 Thn Keatas	0	0 %
Berdasarkan	Jenis	Jumlah	Persentase
Jenis Kelamin	Laki-laki	36	36.00 %
Jenis Kelamin	Perempuan	64	64.00 %
Berdasarkan	Jenis	Jumlah	Persentase
Perkerjaan	Pelajar/Mahasiswa	100	100.00 %
Perkerjaan	Pegawai Swasta	0	0 %
Perkerjaan	PNS	0	0 %
Perkerjaan	TNI/POLRI	0	0 %
Perkerjaan	Pengusaha	0	0 %
Perkerjaan	Lain-lain	0	0 %

tanggal : 4/25/2005

Jam : 4:39:02 AM

jumlah Responden : 100

FREKUENSI DATA KUISIONER

Topik	Pertanyaan	Nilai 4	Nilai 3	Nilai 2	Nilai 1
ASPEK PELAYANAN	Kerapihan penampilan karyawan	3	63	29	5
ASPEK PELAYANAN	Kesesuaian jam kerja karyawan	0	55	41	4
ASPEK PELAYANAN	Hasil kerja Karyawan dapat dipercaya	1	55	33	11
ASPEK PELAYANAN	Ketepatan pemenuhan janji oleh Karyawan	3	41	47	9
ASPEK PELAYANAN	Keaktifan memberikan informasi Akademik yang baru	26	31	28	15
ASPEK PELAYANAN	Keramahan karyawan	4	57	31	8
ASPEK PELAYANAN	Cepat tanggapnya Karyawan memenuhi kebutuhan	1	55	36	8
ASPEK FASILITAS	Kebersihan ruang kuliah	7	61	23	9
ASPEK FASILITAS	Kebersihan kamar mandi/toilet	7	49	31	13
ASPEK FASILITAS	Kelengkapan komputer untuk SIMAK	11	68	13	8
ASPEK FASILITAS	Kelengkapan Laboratorium	7	64	26	3
ASPEK FASILITAS	Kelengkapan perpustakaan Fakultas	4	70	15	11
ASPEK FASILITAS	Kemudahan dalam pembayaran SPP	4	48	38	10
ASPEK FASILITAS	Pelayanan bus kampus dan area parkir	7	56	19	18
ASPEK DOSEN	Ketepatan waktu mengajar	2	53	34	11
ASPEK DOSEN	Penjelasan materi waktu mengajar	2	57	38	3
ASPEK DOSEN	Kesesuaian mengajar dengan silabus	2	62	29	7
ASPEK DOSEN	Ketepatan waktu praktikum	3	55	39	3
ASPEK DOSEN	Kecepatan memberikan nilai UAS	3	44	40	13
ASPEK DOSEN	Kemudahan dalam bimbingan (termasuk KP/TA)	3	52	39	6
ASPEK DOSEN	Kemudahan pertemuan dengan pihak Dekanat	4	56	30	10
REKAPITULASI		4.95 %	54.86 %	31.36 %	8.81 %



Tanggal : 4/25/2005

Jam : 4:39:08 AM

Jumlah Responden : 100

INTERPRETASI KUISIONER

Topik	Pertanyaan	Nilai 4	Nilai 3	Nilai 2	Nilai 1	Skor	Rata-rata	Interpretasi
1	Ketepatan penampihan karyawan	3	63	29	5	264	2.64	Puas
1	Kesesuaian jam kerja karyawan	0	55	41	4	251	2.51	Puas
1	Hasil kerja Karyawan dapat dipercaya	1	55	33	11	246	2.46	Kurang Puas
1	Ketepatan pemenuhan janji oleh Karyawan	3	41	47	9	238	2.38	Kurang Puas
1	Keaktifan memberikan informasi Akademik yang baru	26	31	28	15	268	2.68	Puas
1	Keramahan karyawan	4	57	31	8	257	2.57	Puas
1	Cepat tanggapnya Karyawan memenuhi kebutuhan	1	55	36	8	249	2.49	Kurang Puas
2	Kebersihan ruang kuliah	7	61	23	9	266	2.66	Puas
2	Kebersihan kamar mandi/kolilet	7	49	31	13	250	2.5	Kurang Puas
2	Kelengkapan komputer untuk SIMAK	11	68	13	8	282	2.82	Puas
2	Kelengkapan Laboratorium	7	64	26	3	275	2.75	Puas
2	Kelengkapan perpustakaan Fakultas	4	70	15	11	267	2.67	Puas
2	Kemudahan dalam pembayaran SPP	4	48	38	10	246	2.46	Kurang Puas
2	Pelayanan bus kampus dan area parkir	7	56	19	18	252	2.52	Puas
3	Ketepatan waktu mengajar	2	53	34	11	246	2.46	Kurang Puas
3	Penjelasan materi waktu mengajar	2	57	38	3	258	2.58	Puas
3	Kesesuaian mengajar dengan silabus	2	62	29	7	259	2.59	Puas
3	Ketepatan waktu praktikum	3	55	39	3	258	2.58	Puas
3	Kecepatan memberikan nilai UAS	3	44	40	13	237	2.37	Kurang Puas
3	Kemudahan dalam bimbingan (termasuk KP/TA)	3	52	39	6	252	2.52	Puas
3	Kemudahan pertemuan dengan pihak Dekanat	4	56	30	10	254	2.54	Puas

LISTING PROGRAM TAMPILAN AWAL ADMIN

```
unit Unit1;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics,
  Controls, Forms,
  Dialogs, Menus, OleCtrls, ShockwaveFlashObjects_TLB, ExtCtrls,
  StdCtrls,
  Buttons, DB, DBTables, jpeg;

type
  TForm1 = class(TForm)
    ShockwaveFlash1: TShockwaveFlash;
    GroupBox1: TGroupBox;
    Label1: TLabel;
    Label2: TLabel;
    Edit1: TEdit;
    Edit2: TEdit;
    SpeedButton1: TSpeedButton;
    SpeedButton2: TSpeedButton;
    SpeedButton3: TSpeedButton;
    Q1: TQuery;
    Database1: TDatabase;
    Image2: TImage;
    Image3: TImage;
    Image4: TImage;
    Panel1: TPanel;
    SpeedButton4: TSpeedButton;
    Image1: TImage;
    procedure SpeedButton1Click(Sender: TObject);
    procedure Edit1KeyPress(Sender: TObject; var Key: Char);
    procedure Edit2KeyPress(Sender: TObject; var Key: Char);
    procedure About1Click(Sender: TObject);
    procedure Logout2Click(Sender: TObject);
    procedure SpeedButton3Click(Sender: TObject);
    procedure SpeedButton2Click(Sender: TObject);
    procedure SpeedButton4Click(Sender: TObject);

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

var
  Form1: TForm1;

implementation

uses Unit2, Unit3, Unit4, Unit5;

{$R *.dfm}

procedure TForm1.SpeedButton1Click(Sender: TObject);
```



```

var form3:Tform3;
begin
  Q1.Close;
  Q1.SQL.Clear;
  Q1.SQL.Add('select pass,akses from datauser where pass=md5(:UP)
and namauser=:NM ');
  Q1.params[0].AsString:=Edit2.Text;
  Q1.params[1].AsString:=Edit1.Text;
  Q1.Open;
  if (Q1.FieldByName('pass').AsString<>'') then
    begin
      if (Q1.FieldByName('akses').value=1) then
        begin
          Edit1.Clear;
          Edit2.Clear;
          Edit1.SetFocus;
          form3 := Tform3.Create(application);
          form3.Aturl.Enabled := false;
          form3.Imgel1.Enabled:= false;
          form3.Imgel2.Enabled:= false;
          form3.Imgel3.Enabled:= false;
          form3.Notebook1.ActivePage:= ('2');
          form3.ShowModal;
        end
      else
        begin
          Edit1.Clear;
          Edit2.Clear;
          Edit1.SetFocus;
          form3 := Tform3.Create(application);
          form3.ShowModal;
        end;
      end
    else
      begin
        MessageDlg('Nama user atau password Anda
salah!',mtWarning,[mbOk],0);
        Edit2.Clear;
        Edit1.SetFocus;
      end;
    end;
end;

procedure TForm1.Edit1KeyPress(Sender: TObject; var Key: Char);
begin
  if key=#13 then edit2.SetFocus;
end;

procedure TForm1.Edit2KeyPress(Sender: TObject; var Key: Char);
begin
  if key=#13 then SpeedButton1.Click;
end;

procedure TForm1.About1Click(Sender: TObject);
begin
  form2.ShowModal;
end;

procedure TForm1.Logout2Click(Sender: TObject);

```

```
begin
  close;
end;

procedure TForm1.SpeedButton3Click(Sender: TObject);
var form4:Tform4;
begin
  Q1.Close;
  Q1.SQL.Clear;
  Q1.SQL.Add('select pass, akses from datauser where
pass=md5(:UP) and namauser=:NM');
  Q1.params[0].AsString:=Edit2.Text;
  Q1.params[1].AsString:=Edit1.Text;
  Q1.Open;
  if (Q1.FieldByName('pass').AsString<>'') then
  begin
    if Q1.FieldByName('akses').Value = 2 then
    begin
      Edit1.Clear;
      Edit2.Clear;
      Edit1.SetFocus;
      form4 := TForm4.Create(application);
      form4.ShowModal;
    end
    else MessageDlg('Anda tidak memiliki akses ini!', mterror,
[mbOK], 0);
    end
  else
  begin
    MessageDlg('Nama user atau password Anda
salah!', mtWarning, [mbOk], 0);
    Edit2.Clear;
    Edit1.SetFocus;
  end;
end;

procedure TForm1.SpeedButton2Click(Sender: TObject);
var form5:Tform5;
begin
  Q1.Close;
  Q1.SQL.Clear;
  Q1.SQL.Add('select pass from datauser where pass=md5(:UP) and
namauser=:NM');
  Q1.params[0].AsString:=Edit2.Text;
  Q1.params[1].AsString:=Edit1.Text;
  Q1.Open;
  if (Q1.FieldByName('pass').AsString<>'') then
  begin
    form5 := TForm5.Create(application);
    Form5.LUser.Caption:= Form1.Edit1.Text;
    edit1.Clear;
    Edit2.Clear;
    Edit1.SetFocus;
    form5.ShowModal;
  end
  else
  begin
```

```

        Messagedlg('Nama user atau password Anda
salah!', mtWarning, [mbOk], 0);
        Edit2.Clear;
        Edit1.SetFocus;
    end;
end;
procedure TForm1.SpeedButton4Click(Sender: TObject);
begin
    Close;
end;
end.

```

LISTING PROGRAM TAMPILAN ADMINISTRASI

```

unit Unit5;

interface

uses
    Windows, Messages, SysUtils, Variants, Classes, Graphics,
    Controls, Forms,
    Dialogs, Buttons, StdCtrls, jpeg, ExtCtrls, DB, DBTables;

type
    TForm5 = class(TForm)
        Image1: TImage;
        Label1: TLabel;
        Label2: TLabel;
        Label3: TLabel;
        I2: TEdit;
        I1: TEdit;
        TOk: TSpeedButton;
        SpeedButton2: TSpeedButton;
        Image2: TImage;
        Image3: TImage;
        LUser: TLabel;
        Databasel: TDatabase;
        Q1: TQuery;
        procedure SpeedButton2Click(Sender: TObject);
        procedure I1KeyPress(Sender: TObject; var Key: Char);
        procedure I2KeyPress(Sender: TObject; var Key: Char);
        procedure TOkClick(Sender: TObject);
    private
        { Private declarations }
    public
        { Public declarations }
    end;

var
    Form5: TForm5;

implementation

{$R *.dfm}

```

```
procedure TForm5.SpeedButton2Click(Sender: TObject);
begin
  Close;
end;

procedure TForm5.I1KeyPress(Sender: TObject; var Key: Char);
begin
  if key=#13 then I2.SetFocus;
end;

procedure TForm5.I2KeyPress(Sender: TObject; var Key: Char);
begin
  if key=#13 then Tok.Click;
end;

procedure TForm5.TOkClick(Sender: TObject);
begin
  if (I1.Text <> '') then
  begin
    if (I1.Text = I2.Text) then
    begin
      Q1.Close;
      Q1.SQL.Clear;
      Q1.SQL.Add('UPDATE datauser SET pass= md5(:ps) WHERE
namauser=:us');
      Q1.Params[0].AsString:= I1.Text;
      Q1.Params[1].AsString:= LUser.Caption;
      Q1.ExecSQL;
      MessageDlg('Password anda telah sukses diupdate, silahkan login
kembali', mtInformation, [mbOK], 0);
      Close();
    end
    else
    begin
      MessageDlg('Konfirmasi password anda tidak sama', mtWarning,
[mbOK], 0);
      I2.Clear;
      I2.SetFocus;
    end;
  end
  else
  begin
    MessageDlg('Masukkan password baru anda', mtWarning, [mbOK],
0);
    I1.SetFocus;
  end;
end;

end.
```

LISTING PROGRAM TAMPILAN UBAH PASSWORD

```
unit Unit5;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics,
  Controls, Forms,
  Dialogs, Buttons, StdCtrls, jpeg, ExtCtrls, DB, DBTables;

type
  TForm5 = class(TForm)
    Image1: TImage;
    Label1: TLabel;
    Label2: TLabel;
    Label3: TLabel;
    I2: TEdit;
    I1: TEdit;
    Tok: TSpeedButton;
    SpeedButton2: TSpeedButton;
    Image2: TImage;
    Image3: TImage;
    LUser: TLabel;
    Databasel: TDatabase;
    Q1: TQuery;
    procedure SpeedButton2Click(Sender: TObject);
    procedure I1KeyPress(Sender: TObject; var Key: Char);
    procedure I2KeyPress(Sender: TObject; var Key: Char);
    procedure TokClick(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  Form5: TForm5;

implementation

{$R *.dfm}

procedure TForm5.SpeedButton2Click(Sender: TObject);
begin
  Close;
end;

procedure TForm5.I1KeyPress(Sender: TObject; var Key: Char);
begin
  if key=#13 then I2.SetFocus;
end;

procedure TForm5.I2KeyPress(Sender: TObject; var Key: Char);
begin
  if key=#13 then Tok.Click;
end;
```

```

procedure TForm5.TOkClick(Sender: TObject);
begin
if (I1.Text <> '') then
begin
if (I1.Text = I2.Text) then
begin
Q1.Close;
Q1.SQL.Clear;
Q1.SQL.Add('UPDATE datauser SET pass= md5(:ps) WHERE
namauser=:us');
Q1.Params[0].AsString:= I1.Text;
Q1.Params[1].AsString:= LUser.Caption;
Q1.ExecSQL;
MessageDlg('Password anda telah sukses diupdate, silahkan login
kembali', mtInformation, [mbOK], 0);
Close();
end
else
begin
MessageDlg('Konfirmasi password anda tidak sama', mtWarning,
[mbOK], 0);
I2.Clear;
I2.SetFocus;
end;
end
else
begin
MessageDlg('Masukkan password baru anda', mtWarning, [mbOK],
0);
I1.SetFocus;
end;
end;
end.

```

**LISTING PROGRAM TAMPILAN KONFIGURASI DAN PROSES DATA
KUISIONER**

```

unit Unit3;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics,
  Controls, Forms,
  Dialogs, jpeg, ExtCtrls, Menus, StdCtrls, ComCtrls, Grids,
  DBGrids,
  Buttons, DB, DBTables, QRCtrls, QuickRpt;

type
  TEdit = class(stdctrls.TEdit)
  private
    FAlignment: TAlignment;
    procedure SetAlignment(Value:TAlignment);
  Protected

```

```
procedure CreateParams(Var Params:TCreateParams); OverRide;  
Public  
Property Alignment:TAlignment Read FAlignment Write  
SetAlignment;  
end;
```

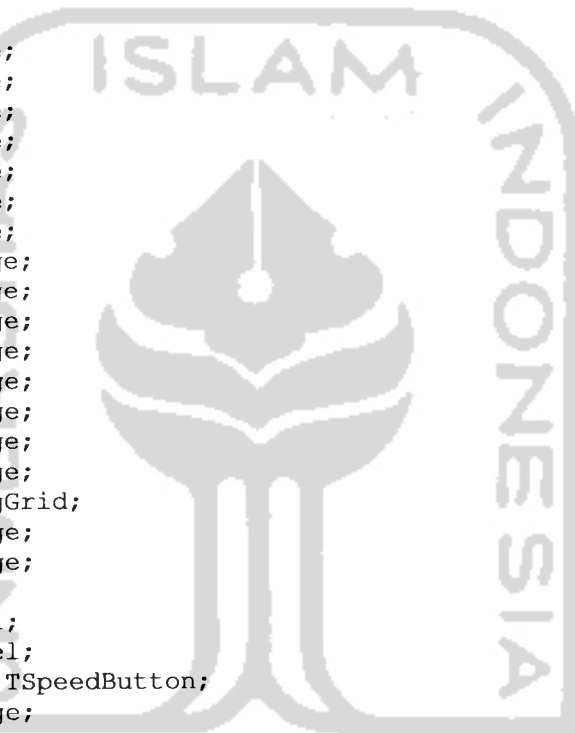
```
type
```

```
TForm3 = class(TForm)  
  Image1: TImage;  
  MainMenu1: TMainMenu;  
  Mulai1: TMenuItem;  
  Login1: TMenuItem;  
  Logout2: TMenuItem;  
  Aturl: TMenuItem;  
  MasukkanKuisisioner1: TMenuItem;  
  UbahKuisisioner1: TMenuItem;  
  TransferKuisisioner1: TMenuItem;  
  ProsesData1: TMenuItem;  
  InformasiResponden1: TMenuItem;  
  AnalisisKuisisioner1: TMenuItem;  
  LaporanKuisisioner1: TMenuItem;  
  Notebook1: TNotebook;  
  GroupBox1: TGroupBox;  
  Panel1: TPanel;  
  Label1: TLabel;  
  ComboBox1: TComboBox;  
  Edit1: TEdit;  
  GroupBox2: TGroupBox;  
  Mem1: TMemo;  
  GroupBox3: TGroupBox;  
  Panel3: TPanel;  
  Edit2: TEdit;  
  Label2: TLabel;  
  DBGrid1: TDBGrid;  
  DBGrid2: TDBGrid;  
  ComboBox2: TComboBox;  
  SpeedButton1: TSpeedButton;  
  SpeedButton4: TSpeedButton;  
  SpeedButton5: TSpeedButton;  
  SpeedButton6: TSpeedButton;  
  SpeedButton7: TSpeedButton;  
  GroupBox4: TGroupBox;  
  SpeedButton9: TSpeedButton;  
  SpeedButton10: TSpeedButton;  
  DBGrid3: TDBGrid;  
  GroupBox5: TGroupBox;  
  SpeedButton11: TSpeedButton;  
  SpeedButton12: TSpeedButton;  
  DBGrid4: TDBGrid;  
  Label7: TLabel;  
  Label6: TLabel;  
  Edit6: TEdit;  
  Databasel: TDatabase;  
  Q1: TQuery;  
  Q2: TQuery;  
  DataSource1: TDataSource;  
  DataSource2: TDataSource;  
  GroupBox6: TGroupBox;
```

```
R1: TRadioButton;  
R2: TRadioButton;  
R3: TRadioButton;  
Q3: TQuery;  
Q4: TQuery;  
PopupMenu1: TPopupMenu;  
Edit7: TMenuItem;  
Hapus1: TMenuItem;  
ComboBox4: TComboBox;  
SpeedButton2: TSpeedButton;  
PopupMenu2: TPopupMenu;  
Edit3: TMenuItem;  
ambah1: TMenuItem;  
Q5: TQuery;  
Reset1: TMenuItem;  
ResetTopik1: TMenuItem;  
ResetPertanyaan1: TMenuItem;  
ResetInformasi1: TMenuItem;  
ResetData1: TMenuItem;  
About1: TMenuItem;  
DataSource3: TDataSource;  
Q6: TQuery;  
Q7: TQuery;  
DataSource4: TDataSource;  
ResetData2: TMenuItem;  
GroupBox7: TGroupBox;  
GroupBox8: TGroupBox;  
GroupBox9: TGroupBox;  
Label3: TLabel;  
Label8: TLabel;  
Label9: TLabel;  
Label10: TLabel;  
Label11: TLabel;  
Label12: TLabel;  
Label13: TLabel;  
Label14: TLabel;  
Label15: TLabel;  
Label16: TLabel;  
Label17: TLabel;  
Label18: TLabel;  
Label19: TLabel;  
GroupBox10: TGroupBox;  
Label4: TLabel;  
Label5: TLabel;  
Edit4: TEdit;  
Edit5: TEdit;  
Edit8: TEdit;  
Edit9: TEdit;  
Edit10: TEdit;  
Edit11: TEdit;  
Edit12: TEdit;  
Edit13: TEdit;  
Edit14: TEdit;  
Edit15: TEdit;  
Edit16: TEdit;  
Edit17: TEdit;  
Edit18: TEdit;  
Edit19: TEdit;
```



```
Edit20: TEdit;
Edit21: TEdit;
Edit22: TEdit;
Edit23: TEdit;
Edit24: TEdit;
Edit25: TEdit;
Edit26: TEdit;
Edit27: TEdit;
Edit28: TEdit;
Edit29: TEdit;
Edit30: TEdit;
Edit31: TEdit;
Edit32: TEdit;
Edit33: TEdit;
QN: TQuery;
Image2: TImage;
Image3: TImage;
Image4: TImage;
Image5: TImage;
Image6: TImage;
Image7: TImage;
Image8: TImage;
Image10: TImage;
Image11: TImage;
Image12: TImage;
Image13: TImage;
Image14: TImage;
Image15: TImage;
Image16: TImage;
Image17: TImage;
Strgr: TStringGrid;
Image18: TImage;
Image19: TImage;
Edit34: TEdit;
Panel2: TPanel;
Label20: TLabel;
SpeedButton3: TSpeedButton;
Image20: TImage;
Image21: TImage;
SpeedButton13: TSpeedButton;
Notebook2: TNotebook;
QuickRep1: TQuickRep;
QRBand1: TQRBand;
Image22: TImage;
SpeedButton14: TSpeedButton;
Notebook3: TNotebook;
QuickRep2: TQuickRep;
Query2: TQuery;
QRBand4: TQRBand;
QRBand5: TQRBand;
QRBand6: TQRBand;
QRLabel17: TQRLabel;
QRLabel18: TQRLabel;
QRLabel19: TQRLabel;
QRLabel10: TQRLabel;
QRLabel11: TQRLabel;
QRLabel12: TQRLabel;
QRLabel13: TQRLabel;
```



```

ResetTopik1.Enabled := false;
ResetPertanyaan1.Enabled := true;
ResetInformasi1.Enabled := false;
GroupBox3.Enabled := true;
GroupBox1.Enabled := false;
GroupBox2.Enabled := false;
NoteBook1.ActivePage := ('1');
ComboBox2.Enabled := true;
ComboBox2.Items.Clear;
ComboBox2.SetFocus;
Q3.Close;
Q3.SQL.Clear;
Q3.SQL.Add('select NAspek from topik');
Q3.Open;
for i:= 1 to Q3.RecordCount do
begin
    ComboBox2.Items.Add(Q3.FieldByName('NAspek').AsString);
    Q3.Next;
end;
Q2.Close;
Q2.Open;
end;

procedure TForm3.SpeedButton6Click(Sender: TObject);
begin
    GroupBox6.Enabled := true ;
    Edit1.Enabled := true ;
    SpeedButton7.Enabled := true ;
    Edit1.SetFocus;
    ComboBox1.Enabled := false ;
    DBGrid1.Enabled := false;
    SpeedButton6.Enabled := false ;
end;

procedure TForm3.ComboBox1Select(Sender: TObject);
Var Perintah : string;
begin
    if (ComboBox1.ItemIndex <> -1) then
    begin
        SpeedButton6.Enabled := true ;
        Q3.Close;
        Q3.SQL.Clear;
        Case ComboBox1.ItemIndex of
            0: Perintah := 'Select * from topik where kd_aspek = 1';
            1: Perintah := 'Select * from topik where kd_aspek = 2';
            2: Perintah := 'Select * from topik where kd_aspek = 3';
            3: Perintah := 'Select * from topik where kd_aspek = 4';
            4: Perintah := 'Select * from topik where kd_aspek = 5';
        end;
        Q3.SQL.Add(perintah);
        Q3.Open;
        Edit1.Text := Q3.FieldByName('NAspek').AsString;
        Case Q3.FieldByName('pilihan').Value of
            1: Begin
                R1.Checked := true ;
                R2.Checked := false ;
                R3.Checked := false ;
            End;
        end;
    end;
end;

```

```

2: Begin
    R1.Checked := false ;
    R2.Checked := true ;
    R3.Checked := false ;
End;
3: Begin
    R1.Checked := false ;
    R2.Checked := false ;
    R3.Checked := true ;
End;
end;
end;
end;

procedure TForm3.InformasiResponden1Click(Sender: TObject);
var tahun, bulan, tanggal: word;
    perintah,tmp: string;
begin
    Reset1.Enabled := false;
    NoteBook1.ActivePage := ('2') ;
    Q6.Close;
    Q6.SQL.Clear;
    Q6.SQL.Add('SELECT nama, asal, umur, jeniskelamin, pekerjaan
FROM responden, pekerjaan WHERE
responden.kd_kerja=pekerjaan.kd_kerja');
    Q6.Open;
    decodedate(date,tahun,bulan,tanggal);
    edit4.Text:= ' ' + intToStr(tanggal) + '/' +
IntToStr(bulan) + '/' + IntToStr(tahun);
    if Q6.RecordCount<>0 then
        begin
            perintah:= 'SELECT COUNT(jeniskelamin) as jumlah FROM
responden';
            edit5.Text:= jumlah(perintah);
            perintah:= 'SELECT COUNT(umur) as jumlah FROM responden where
umur>=15 and umur<=20';
            edit8.Text:= jumlah(perintah);
            perintah:= 'SELECT COUNT(umur) as jumlah FROM responden where
umur>=21 and umur<=25';
            edit9.Text:= jumlah(perintah);
            perintah:= 'SELECT COUNT(umur) as jumlah FROM responden where
umur>=26 and umur<=35';
            edit10.Text:= jumlah(perintah);
            perintah:= 'SELECT COUNT(umur) as jumlah FROM responden where
umur>=36 and umur<=45';
            edit11.Text:= jumlah(perintah);
            perintah:= 'SELECT COUNT(umur) as jumlah FROM responden where
umur>=46';
            edit12.Text:= jumlah(perintah);
            if edit8.Text<>'0' then
                Str((StrToInt(edit8.Text)/StrToInt(edit5.Text))*100:2:2,tmp)
            else tmp:= '0';
            edit13.Text:= tmp + ' %';
            if edit9.Text<>'0' then
                Str((StrToInt(edit9.Text)/StrToInt(edit5.Text))*100:2:2,tmp)
            else tmp:= '0';
            edit14.Text:= tmp + ' %';
        end;
    end;
end;

```

```

        if edit10.Text<>'0' then
Str((StrToInt(edit10.Text)/StrToInt(edit5.Text))*100:2:2,tmp)
        else tmp:= '0';
        edit15.Text:= tmp + ' %';
        if edit11.Text<>'0' then
Str((StrToInt(edit11.Text)/StrToInt(edit5.Text))*100:2:2,tmp)
        else tmp:= '0';
        edit16.Text:= tmp + ' %';
        if edit12.Text<>'0' then
Str((StrToInt(edit12.Text)/StrToInt(edit5.Text))*100:2:2,tmp)
        else tmp:= '0';
        edit17.Text:= tmp + ' %';
        perintah:= 'SELECT COUNT(jeniskelamin) as jumlah FROM
responden where jeniskelamin='L''';
        edit18.Text:= jumlah(perintah);
        perintah:= 'SELECT COUNT(jeniskelamin) as jumlah FROM
responden where jeniskelamin='P''';
        edit19.Text:= jumlah(perintah);
        if edit18.Text<>'0' then
Str((StrToInt(edit18.Text)/StrToInt(edit5.Text))*100:2:2,tmp)
        else tmp:= '0';
        edit20.Text:= tmp + ' %';
        if edit19.Text<>'0' then
Str((StrToInt(edit19.Text)/StrToInt(edit5.Text))*100:2:2,tmp)
        else tmp:= '0';
        edit21.Text:= tmp + ' %';
        perintah:= 'SELECT COUNT(kd_kerja) as jumlah FROM responden
where kd_kerja=1';
        edit22.Text:= jumlah(perintah);
        perintah:= 'SELECT COUNT(kd_kerja) as jumlah FROM responden
where kd_kerja=2';
        edit23.Text:= jumlah(perintah);
        perintah:= 'SELECT COUNT(kd_kerja) as jumlah FROM responden
where kd_kerja=3';
        edit24.Text:= jumlah(perintah);
        perintah:= 'SELECT COUNT(kd_kerja) as jumlah FROM responden
where kd_kerja=4';
        edit25.Text:= jumlah(perintah);
        perintah:= 'SELECT COUNT(kd_kerja) as jumlah FROM responden
where kd_kerja=5';
        edit26.Text:= jumlah(perintah);
        perintah:= 'SELECT COUNT(kd_kerja) as jumlah FROM responden
where kd_kerja=6';
        edit27.Text:= jumlah(perintah);
        if edit22.Text<>'0' then
Str((StrToInt(edit22.Text)/StrToInt(edit5.Text))*100:2:2,tmp)
        else tmp:= '0';
        edit28.Text:= tmp + ' %';
        if edit23.Text<>'0' then
Str((StrToInt(edit23.Text)/StrToInt(edit5.Text))*100:2:2,tmp)
        else tmp:= '0';
        edit29.Text:= tmp + ' %';
        if edit24.Text<>'0' then
Str((StrToInt(edit24.Text)/StrToInt(edit5.Text))*100:2:2,tmp)
        else tmp:= '0';
        edit30.Text:= tmp + ' %';
        if edit25.Text<>'0' then
Str((StrToInt(edit25.Text)/StrToInt(edit5.Text))*100:2:2,tmp)

```

```

        else tmp:= '0';
        edit31.Text:= tmp + ' %';
        if edit26.Text<>'0' then
Str((StrToInt(edit26.Text)/StrToInt(edit5.Text))*100:2:2,tmp)
        else tmp:= '0';
        edit32.Text:= tmp + ' %';
        if edit27.Text<>'0' then
Str((StrToInt(edit27.Text)/StrToInt(edit5.Text))*100:2:2,tmp)
        else tmp:= '0';
        edit33.Text:= tmp + ' %';
        end
    else
    begin
        MessageDlg('Data responden tidak ada.', MtInformation,
[mbOk], 0);
        SpeedButton8.Enabled:= false;
        SpeedButton14.Enabled:= false;
        Edit5.Text := '';
        Edit8.Text := '';
        Edit9.Text := '';
        Edit10.Text := '';
        Edit11.Text := '';
        Edit12.Text := '';
        Edit13.Text := '';
        Edit14.Text := '';
        Edit15.Text := '';
        Edit16.Text := '';
        Edit17.Text := '';
        Edit18.Text := '';
        Edit19.Text := '';
        Edit20.Text := '';
        Edit21.Text := '';
        Edit22.Text := '';
        Edit23.Text := '';
        Edit24.Text := '';
        Edit25.Text := '';
        Edit26.Text := '';
        Edit27.Text := '';
        Edit28.Text := '';
        Edit29.Text := '';
        Edit30.Text := '';
        Edit31.Text := '';
        Edit32.Text := '';
        Edit33.Text := '';
    end;
end;

procedure TForm3.AnalisisKuisisioner1Click(Sender: TObject);
var i: byte;
    tahun, bulan, tanggal: word;
    perintah,tmp: string;
begin
    Reset1.Enabled :=false;
    NoteBook1.ActivePage := ('3');
    decodedate(date,tahun,bulan,tanggal);
    edit34.Text:= intToStr(tanggal) + '/' + IntToStr(bulan) + '/' +
IntToStr(tahun);
    Q3.Close;

```

```

Q3.SQL.Clear;
Q3.SQL.Add('select NASpek from topik');
Q3.Open;
  for i:= 1 to Q3.RecordCount do
  begin
    Q3.Next;
  end;
  edit6.Text:=jumlah('SELECT COUNT(jeniskelamin) as jumlah FROM
responden');
  Q7.Close;
  Q7.SQL.Clear;
  Q7.SQL.Add('select distinct NASpek, pertanyaan, n1, n2, n3, n4
from topik, pertanyaan, nilai where nilai.nomer=pertanyaan.nomer
and          nilai.kd_aspek=pertanyaan.kd_aspek          and
pertanyaan.kd_aspek=topik.kd_aspek          and
pertanyaan.pertanyaan<>'');
  Q7.Open;
  if Q7.RecordCount<>0 then
  begin
    perintah:= 'SELECT SUM(n4) as jumlah FROM nilai';
    edit35.Text:= jumlah(perintah);
    perintah:= 'SELECT SUM(n3) as jumlah FROM nilai';
    edit36.Text:= jumlah(perintah);
    perintah:= 'SELECT SUM(n2) as jumlah FROM nilai';
    edit37.Text:= jumlah(perintah);
    perintah:= 'SELECT SUM(n1) as jumlah FROM nilai';
    edit38.Text:= jumlah(perintah);

edit43.Text:=IntToStr(StrToInt(edit35.Text)+StrToInt(edit36.Text)+
StrToInt(edit37.Text)+StrToInt(edit38.Text));
    if          edit35.Text<>'0'          then
Str((StrToInt(edit35.Text)/StrToInt(edit43.Text))*100:2:2,tmp)
    else tmp:='0';
    edit39.Text:= tmp + ' %';
    if          edit36.Text<>'0'          then
Str((StrToInt(edit36.Text)/StrToInt(edit43.Text))*100:2:2,tmp)
    else tmp:='0';
    edit40.Text:= tmp + ' %';
    if          edit37.Text<>'0'          then
Str((StrToInt(edit37.Text)/StrToInt(edit43.Text))*100:2:2,tmp)
    else tmp:='0';
    edit41.Text:= tmp + ' %';
    if          edit38.Text<>'0'          then
Str((StrToInt(edit38.Text)/StrToInt(edit43.Text))*100:2:2,tmp)
    else tmp:='0';
    edit42.Text:= tmp + ' %';
  end
  else
  begin
    MessageDlg('Data kuisiner tidak ada.', MtInformation, [mbOk],
0);
    SpeedButton11.Enabled:= false;
    SpeedButton15.Enabled:= false;
  end;
end;

procedure TForm3.SpeedButton7Click(Sender: TObject);
begin

```

```

if (Messagedlg('Anda ingin mengupdate topik no: ' +
IntToStr(ComboBox1.ItemIndex+1) + ' ?', MtConfirmation,
[mbYes,mbNo], 0) = mrYes) then
  Begin
    Q4.Close;
    Q4.SQL.Clear;
    Case ComboBox1.ItemIndex of
      0: Q4.SQL.Add('UPDATE topik SET NASpek=:NA, PILIHAN = :OP
WHERE kd_aspek=1');
      1: Q4.SQL.Add('UPDATE topik SET NASpek=:NA, PILIHAN = :OP
WHERE kd_aspek=2');
      2: Q4.SQL.Add('UPDATE topik SET NASpek=:NA, PILIHAN = :OP
WHERE kd_aspek=3');
      3: Q4.SQL.Add('UPDATE topik SET NASpek=:NA, PILIHAN = :OP
WHERE kd_aspek=4');
      4: Q4.SQL.Add('UPDATE topik SET NASpek=:NA, PILIHAN = :OP
WHERE kd_aspek=5');
    end;
    Q4.Params[0].AsString := Edit1.Text ;
    if (R1.Checked) then Q4.Params[1].Value := 1 ;
    if (R2.Checked) then Q4.Params[1].Value := 2 ;
    if (R3.Checked) then Q4.Params[1].Value := 3 ;
    Q4.ExecSQL;
    Q1.Close;
    Q1.Open;
    Messagedlg('Topik telah sukses di Update', MtInformation,
[mbOk], 0);
    Edit1.Enabled := false;
    GroupBox6.Enabled :=false ;
    ComboBox1.Enabled :=true ;
    ComboBox1.ItemIndex := -1;
    ComboBox1.Text := 'Pilih Topik';
    DBGrid1.Enabled := true;
    ComboBox1.SetFocus;
    Edit1.Text := ('');
    SpeedButton7.Enabled := false;
  end;
end;

procedure TForm3.Edit7Click(Sender: TObject);
begin
  ComboBox1.ItemIndex := Q1.fieldbyname('kd_aspek').Value-1;
  Edit1.Text := Q1.fieldbyname('NASpek').AsString;
  Case Q1.fieldbyname('pilihan').Value of
    1: Begin
      R1.Checked := true ;
      R2.Checked := false ;
      R3.Checked := false ;
    End;
    2: Begin
      R1.Checked := false ;
      R2.Checked := true ;
      R3.Checked := false ;
    End;
    3: Begin
      R1.Checked := false ;
      R2.Checked := false ;
      R3.Checked := true ;
    End;
  end;
end;

```

```

        End;
    end;
    SpeedButton6.Enabled := true;
    SpeedButton6.Click;
end;

procedure TForm3.Hapus1Click(Sender: TObject);
begin
    if (MessageDlg('Anda ingin manghapus topik no: ' +
Q1.fieldbyname('kd_aspek').AsString + ' ?', MtConfirmation,
[mbYes,mbNo], 0) = mrYes) then
    Begin
        Q4.Close;
        Q4.SQL.Clear;
        Q4.SQL.Add('UPDATE topik SET NAspek=:NA, PILIHAN =1 WHERE
kd_aspek=:KD');
        Q4.Params[0].AsString := '';
        Q4.Params[1].AsString := Q1.fieldbyname('kd_aspek').AsString;
        Q4.ExecSQL;
        MessageDlg('Topik telah sukses dihapus!',MtInformation,
[mbOk],0);
        Q1.Close;
        Q1.Open;
    end;
end;

procedure TForm3.PopupMenu1Popup(Sender: TObject);
begin
    if (Q1.FieldByName('NAspek').AsString = '') then Hapus1.Visible
:= false
    else Hapus1.Visible := true;
end;

procedure TForm3.FormClose(Sender: TObject; var Action:
TCloseAction);
begin
    action := caFree;
end;

procedure TForm3.Edit1KeyPress(Sender: TObject; var Key: Char);
begin
    if key=#13 then SpeedButton7.Click;
end;

procedure TForm3.ComboBox1KeyPress(Sender: TObject; var Key:
Char);
begin
    if key=#13 then SpeedButton6.Click;
end;

procedure TForm3.ComboBox2Select(Sender: TObject);
Var Tanyaan : string;
begin
    if (ComboBox2.ItemIndex <> -1) then
    begin
        SpeedButton1.Enabled := true ;
        Q2.Close;
        Q2.SQL.Clear;

```



```

        case ComboBox2.ItemIndex of
            0: Tanyaan := 'Select * from pertanyaan where kd_aspek =
1';
            1: Tanyaan := 'Select * from pertanyaan where kd_aspek =
2';
            2: Tanyaan := 'Select * from pertanyaan where kd_aspek =
3';
            3: Tanyaan := 'Select * from pertanyaan where kd_aspek =
4';
            4: Tanyaan := 'Select * from pertanyaan where kd_aspek =
5';
        end;
        Q2.SQL.Add(Tanyaan);
        Q2.Open;
    end;
end;

procedure TForm3.Edit3Click(Sender: TObject);
begin
    Panel3.Enabled := true;
    ComboBox4.ItemIndex := Q2.fieldbyname('nomer').Value-1;
    Edit2.Text := Q2.fieldbyname('pertanyaan').AsString;
    SpeedButton2.Enabled := true;
    Edit2.SetFocus;
end;

procedure TForm3.SpeedButton1Click(Sender: TObject);
begin
    Panel3.Enabled := true;
    SpeedButton2.Enabled := true;
    ComboBox4.SetFocus;
end;

procedure TForm3.SpeedButton2Click(Sender: TObject);
begin
    if (MessageDlg('Anda ingin mengupdate pertanyaan: ' +
IntToStr(ComboBox4.ItemIndex+1) + ' ?', MtConfirmation,
[mbYes,mbNo], 0) = mrYes) then
    begin
        Q4.Close;
        Q4.SQL.Clear;
        Q4.SQL.Add('UPDATE pertanyaan SET pertanyaan= :pr WHERE
nomer=:no AND kd_aspek=:kd');
        Q4.Params[0].AsString := edit2.Text;
        Q4.Params[1].Value := ComboBox4.ItemIndex+1;
        Q4.Params[2].Value := Q2.fieldbyname('kd_aspek').Value;
        Q4.ExecSQL;
        Q2.Close;
        Q2.Open;
        ComboBox4.ItemIndex := -1;
        Edit2.Text := ('');
        ComboBox4.SetFocus;
    end;
end;

procedure TForm3.ambah1Click(Sender: TObject);
begin
    if (MessageDlg('Anda ingin manghapus pertanyaan no: ' +
Q2.fieldbyname('kd_aspek').AsString + ' ?', MtConfirmation,
[mbYes,mbNo], 0) = mrYes) then

```

```

Begin
    Q4.Close;
    Q4.SQL.Clear;
    Q4.SQL.Add('UPDATE pertanyaan SET pertanyaan= :pr WHERE
nomer=:no AND kd_aspek=:kd');
    Q4.Params[0].AsString := '';
    Q4.Params[1].Value := Q2.fieldbyname('nomer').Value;
    Q4.Params[2].Value := Q2.fieldbyname('kd_aspek').Value;
    Q4.ExecSQL;
    MessageDlg('Pertanyaan telah sukses dihapus!',MtInformation,
[mbOk], 0);
    Q2.Close;
    Q2.Open;
end;
end;

procedure TForm3.ComboBox2KeyPress(Sender: TObject; var Key:
Char);
begin
    if key=#13 then SpeedButton1.Click
end;

procedure TForm3.Edit2KeyPress(Sender: TObject; var Key: Char);
begin
    if key=#13 then SpeedButton2.Click;
end;

procedure TForm3.ComboBox4KeyPress(Sender: TObject; var Key:
Char);
begin
    if key=#13 then edit2.SetFocus;
end;

procedure TForm3.SpeedButton5Click(Sender: TObject);
begin
    Q4.Close;
    Q4.SQL.Clear;
    Q4.SQL.Add('UPDATE informasi SET informasi= :inf WHERE
kd_inf=:id');
    Q4.params[0].Value := memol.Text;
    Q4.Params[1].AsString := 'top';
    Q4.ExecSQL;
    Memol.Enabled := false;
    MessageDlg('Informasi telah sukses di Update', MtInformation,
[mbOk], 0);
    SpeedButton5.Enabled := false;
    SpeedButton4.Enabled := true;
end;

procedure TForm3.FormCreate(Sender: TObject);
begin
    Memol.Lines.Add(Q5.fieldbyname('informasi').Value);
    Edit5.Alignment:= taCENTER;
    Edit8.Alignment:= taCENTER;
    Edit9.Alignment:= taCENTER;
    Edit10.Alignment:= taCENTER;
    Edit11.Alignment:= taCENTER;
    Edit12.Alignment:= taCENTER;

```

```
Edit13.Alignment:= taCENTER;
Edit14.Alignment:= taCENTER;
Edit15.Alignment:= taCENTER;
Edit16.Alignment:= taCENTER;
Edit17.Alignment:= taCENTER;
Edit18.Alignment:= taCENTER;
Edit19.Alignment:= taCENTER;
Edit20.Alignment:= taCENTER;
Edit21.Alignment:= taCENTER;
Edit22.Alignment:= taCENTER;
Edit23.Alignment:= taCENTER;
Edit24.Alignment:= taCENTER;
Edit25.Alignment:= taCENTER;
Edit26.Alignment:= taCENTER;
Edit27.Alignment:= taCENTER;
Edit28.Alignment:= taCENTER;
Edit29.Alignment:= taCENTER;
Edit30.Alignment:= taCENTER;
Edit31.Alignment:= taCENTER;
Edit32.Alignment:= taCENTER;
Edit33.Alignment:= taCENTER;
Edit6.Alignment:= taCENTER;
Edit35.Alignment:= taCENTER;
Edit36.Alignment:= taCENTER;
Edit37.Alignment:= taCENTER;
Edit38.Alignment:= taCENTER;
Edit39.Alignment:= taCENTER;
Edit40.Alignment:= taCENTER;
Edit41.Alignment:= taCENTER;
Edit42.Alignment:= taCENTER;
Edit43.Alignment:= taCENTER;
end;

procedure TForm3.About1Click(Sender: TObject);
begin
    form2.ShowModal;
end;

procedure TForm3.ResetData(Item: Byte);
begin
    Q4.Close;
    Q4.SQL.Clear;
    case item of
        1: Q4.SQL.Add('UPDATE topik SET NASpek= '''' WHERE
NASpek<>'''');
        2: Q4.SQL.Add('UPDATE pertanyaan SET pertanyaan= '''' WHERE
pertanyaan<>'''');
        3: Q4.SQL.Add('UPDATE informasi SET informasi= '''' WHERE
kd_inf='''top''');
        4: Q4.SQL.Add('DELETE FROM responden');
        5: Q4.SQL.Add('UPDATE nilai SET n1= 0, n2=0, n3=0, n4=0 WHERE
n1<>0 or n2<>0 or n3<>0 or n4<>0');
    end;
    Q4.ExecSQL;
end;

procedure TForm3.ResetTopik1Click(Sender: TObject);
begin
```

```

        if (QN.fieldByName('pilihan').Value = 3) then
tmppilihan:= 'Penting';
        Strgr.Cells[0,p]:= ' ' +
QN.fieldByName('kd_aspek').AsString;
        Strgr.Cells[1,p]:= ' ' +
QN.fieldByName('pertanyaan').AsString;
        Strgr.Cells[2,p]:= ' ' +
QN.fieldByName('n4').AsString;
        Strgr.Cells[3,p]:= ' ' +
QN.fieldByName('n3').AsString;
        Strgr.Cells[4,p]:= ' ' +
QN.fieldByName('n2').AsString;
        Strgr.Cells[5,p]:= ' ' +
QN.fieldByName('n1').AsString;
        Strgr.Cells[6,p]:= ' ' +
IntToStr((QN.fieldByName('n4').Value*4)+(QN.fieldByName('n3').Value*3)+(QN.fieldByName('n2').Value*2)+QN.fieldByName('n1').Value);
        tmpratanl:=
StrToInt(strgr.Cells[6,p])/(QN.fieldByName('n4').Value +
QN.fieldByName('n3').Value + QN.fieldByName('n2').Value +
QN.fieldByName('n1').Value);
        Str(tmpratanl:2:2,tmpratastr);
        Strgr.Cells[7,p]:= ' ' + tmpratastr;
        if ((tmpratanl >= 1) and (tmpratanl <= 1.75)) then
Strgr.Cells[8,p]:= ' Tidak ' + tmppilihan;
        if ((tmpratanl > 1.75) and (tmpratanl <= 2.50)) then
Strgr.Cells[8,p]:= ' Kurang ' + tmppilihan;
        if ((tmpratanl > 2.50) and (tmpratanl <= 3.25)) then
Strgr.Cells[8,p]:= ' ' + tmppilihan;
        if (tmpratanl > 3.25) then Strgr.Cells[8,p]:= '
Sangat ' + tmppilihan;
        QN.Next;
    end;

    Q8.Close;
    Q8.SQL.Clear;
    Q8.SQL.Add('DELETE FROM tmp');
    Q8.ExecSQL;
    for i:=1 to strgr.RowCount-1 do
    begin
        Q8.Close;
        Q8.SQL.Clear;
        Q8.SQL.Add('INSERT INTO tmp (A, B, C, D, E, F, G, H,
I) VALUES (:a, :b, :c, :d, :e, :f, :g, :h, :i)');
        Q8.Params[0].AsString:= strgr.Cells[0,i];
        Q8.Params[1].AsString:= strgr.Cells[1,i];
        Q8.Params[2].AsString:= strgr.Cells[2,i];
        Q8.Params[3].AsString:= strgr.Cells[3,i];
        Q8.Params[4].AsString:= strgr.Cells[4,i];
        Q8.Params[5].AsString:= strgr.Cells[5,i];
        Q8.Params[6].AsString:= strgr.Cells[6,i];
        Q8.Params[7].AsString:= strgr.Cells[7,i];
        Q8.Params[8].AsString:= strgr.Cells[8,i];
        Q8.ExecSQL;
    end;
    end
else
    begin

```

```
Strgr.RowCount:= 2;
MessageDlg('Data kuisiner tidak ada.', MtInformation,
[mbOk], 0);
Q8.Close;
Q8.SQL.Clear;
Q8.SQL.Add('DELETE FROM tmp');
Q8.ExecSQL;
for i:=1 to strgr.RowCount-1 do
begin
    Q8.Close;
    Q8.SQL.Clear;
    Q8.SQL.Add('INSERT INTO tmp (A, B, C, D, E, F, G, H, I)
VALUES (:a, :b, :c, :d, :e, :f, :g, :h, :i)');
    Q8.Params[0].AsString:= strgr.Cells[0,i];
    Q8.Params[1].AsString:= strgr.Cells[1,i];
    Q8.Params[2].AsString:= strgr.Cells[2,i];
    Q8.Params[3].AsString:= strgr.Cells[3,i];
    Q8.Params[4].AsString:= strgr.Cells[4,i];
    Q8.Params[5].AsString:= strgr.Cells[5,i];
    Q8.Params[6].AsString:= strgr.Cells[6,i];
    Q8.Params[7].AsString:= strgr.Cells[7,i];
    Q8.Params[8].AsString:= strgr.Cells[8,i];
    Q8.ExecSQL;
end;
SpeedButton3.Enabled:= false;
SpeedButton16.Enabled:= false;
end;
end;

procedure TForm3.ResetData2Click(Sender: TObject);
begin
    if (MessageDlg('Anda yakin ingin menghapus semua DATA HASIL
KUISIONER ?', MtConfirmation, [mbYes, mbNo], 0) = mrYes) then
        ResetData(5);
end;

function TForm3.jumlah(perintah: string): string;
begin
    QN.Close;
    QN.SQL.Clear;
    QN.SQL.Add(perintah);
    QN.Open;
    result:= QN.fieldbyname('jumlah').asString;
end;

procedure TForm3.Imgel1Click(Sender: TObject);
begin
    MasukkanKuisiner1.Click;
end;

procedure TForm3.Imgel2Click(Sender: TObject);
begin
    UbahKuisiner1.Click;
end;

procedure TForm3.Imgel3Click(Sender: TObject);
begin
    TransferKuisiner1.Click;
end;
```

```
end;

procedure TForm3.Imagel4Click(Sender: TObject);
begin
    InformasiResponden1.Click;
end;

procedure TForm3.Imagel5Click(Sender: TObject);
begin
    AnalisisKuisisioner1.Click;
end;

procedure TForm3.Imagel7Click(Sender: TObject);
begin
    Logout2.Click;
end;

procedure TForm3.StrG;
begin
    Strgr.Cells[0,0]:= ' No Topik';
    Strgr.Cells[1,0]:= ' Pertanyaan';
    Strgr.Cells[2,0]:= ' Value 4';
    Strgr.Cells[3,0]:= ' Value 3';
    Strgr.Cells[4,0]:= ' Value 2';
    Strgr.Cells[5,0]:= ' Value 1';
    Strgr.Cells[6,0]:= ' Skors';
    Strgr.Cells[7,0]:= ' Rata-rata';
    Strgr.Cells[8,0]:= ' Interpretasi';
end;

procedure TForm3.Imagel6Click(Sender: TObject);
begin
    LaporanKuisisioner1.Click;
end;

procedure TForm3.SpeedButton13Click(Sender: TObject);
begin
    LaporanKuisisioner1.Click;
end;

procedure TForm3.SpeedButton12Click(Sender: TObject);
begin
    AnalisisKuisisioner1.Click;
end;

Procedure TForm3.Res(i:Byte);
begin
    Query5.Close;
    Query5.SQL.Clear;
    Query5.SQL.Add('UPDATE tmp1 SET B=:jm, C=:pr WHERE A=:us');
    case i of
        1 : Query5.ParamByName('us').AsString:= '15 - 20 Thn';
        2 : Query5.ParamByName('us').AsString:= '21 - 25 Thn';
        3 : Query5.ParamByName('us').AsString:= '26 - 35 Thn';
        4 : Query5.ParamByName('us').AsString:= '36 - 45 Thn';
        5 : Query5.ParamByName('us').AsString:= '46 Thn Keatas';
        6 : Query5.ParamByName('us').AsString:= 'Laki-laki';
        7 : Query5.ParamByName('us').AsString:= 'Perempuan';
    end;
```

```
8 : Query5.ParamByName('us').AsString:= 'Pelajar/Mahasiswa';
9 : Query5.ParamByName('us').AsString:= 'Pegawai Swasta';
10 : Query5.ParamByName('us').AsString:= 'PNS';
11 : Query5.ParamByName('us').AsString:= 'TNI/POLRI';
12 : Query5.ParamByName('us').AsString:= 'Pengusaha';
13 : Query5.ParamByName('us').AsString:= 'Lain-lain';
end;
Query5.ParamByName('jm').AsString:= Satu(i);
Query5.ParamByName('pr').AsString:= Dua(i);
Query5.ExecSQL;
end;
```

```
function TForm3.Satu(i:Byte):String;
var temp:string;
```

```
begin
  Case i of
    1: temp:=Edit8.Text;
    2: temp:=Edit9.Text;
    3: temp:=Edit10.Text;
    4: temp:=Edit11.Text;
    5: temp:=Edit12.Text;
    6: temp:=Edit18.Text;
    7: temp:=Edit19.Text;
    8: temp:=Edit22.Text;
    9: temp:=Edit23.Text;
    10: temp:=Edit24.Text;
    11: temp:=Edit25.Text;
    12: temp:=Edit26.Text;
    13: temp:=Edit27.Text;
  end;
  result:=temp;
end;
```

```
function TForm3.Dua(i:Byte):String;
var temp:string;
```

```
begin
  Case i of
    1: temp:=Edit13.Text;
    2: temp:=Edit14.Text;
    3: temp:=Edit15.Text;
    4: temp:=Edit16.Text;
    5: temp:=Edit17.Text;
    6: temp:=Edit20.Text;
    7: temp:=Edit21.Text;
    8: temp:=Edit28.Text;
    9: temp:=Edit29.Text;
    10: temp:=Edit30.Text;
    11: temp:=Edit31.Text;
    12: temp:=Edit32.Text;
    13: temp:=Edit33.Text;
  end;
  result:=temp;
end;
```

```
Procedure TForm3.Res1(i:Byte);
```

```
begin
  Query6.Close;
  Query6.SQL.Clear;
```

```
Query6.SQL.Add('UPDATE tmp2 SET B=:b1, C=:c1, D=:d1, E=:e1, F=:f1,
G=:g1, H=:h1, I=:i1 WHERE A=''persen''');
Query6.ParamByName('b1').AsString:= Edit39.Text;
Query6.ParamByName('c1').AsString:= Edit40.Text;
Query6.ParamByName('d1').AsString:= Edit41.Text;
Query6.ParamByName('e1').AsString:= Edit42.Text;
Query6.ParamByName('f1').AsString:= Edit35.Text;
Query6.ParamByName('g1').AsString:= Edit36.Text;
Query6.ParamByName('h1').AsString:= Edit37.Text;
Query6.ParamByName('i1').AsString:= Edit38.Text;
Query6.ExecSQL;
end;
```

```
procedure TForm3.SpeedButton14Click(Sender: TObject);
var i :Byte;
begin
Res(1);
for i:=2 to 13 do Res(i);
Query4.Close;
Query4.Open;
Query1.Close;
Query1.Open;
QuickRepl.Print;
end;
```

```
procedure TForm3.SpeedButton11Click(Sender: TObject);
begin
Res1(1);
Query7.Close;
Query7.Open;
Query4.Close;
Query4.Open;
Query2.Close;
Query2.Open;
QuickRep2.Print;
end;
```

```
procedure TForm3.SpeedButton3Click(Sender: TObject);
begin
Query4.Close;
Query4.Open;
Query3.Close;
Query3.Open;
QuickRep3.Print;
end;
```

```
procedure TForm3.SpeedButton8Click(Sender: TObject);
var i :Byte;
begin
Res(1);
for i:=2 to 13 do Res(i);
Query4.Close;
Query4.Open;
Query1.Close;
Query1.Open;
QuickRepl.Preview;
end;
```



```

procedure TForm3.SpeedButton15Click(Sender: TObject);
begin
Res1(1);
Query7.Close;
Query7.Open;
Query4.Close;
Query4.Open;
Query2.Close;
Query2.Open;
QuickRep2.Preview;
end;

procedure TForm3.SpeedButton16Click(Sender: TObject);
begin
Query4.Close;
Query4.Open;
Query3.Close;
Query3.Open;
QuickRep3.Preview;
end;

procedure TForm3.SpeedButton9Click(Sender: TObject);
begin
  InformasiResponden1.Click;
end;

end.

```

LISTING PROGRAM TAMPILAN AWAL RESPONDEN

```

unit Unit1;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics,
  Controls, Forms,
  Dialogs, jpeg, ExtCtrls, StdCtrls, OleCtrls,
  ShockwaveFlashObjects_TLB,
  Buttons, DB, DBTables;

type
  TForm1 = class(TForm)
    ShockwaveFlash1: TShockwaveFlash;
    GroupBox1: TGroupBox;
    GroupBox2: TGroupBox;
    Label1: TLabel;
    Label2: TLabel;
    SpeedButton1: TSpeedButton;
    Databasel: TDatabase;
    Edit1: TEdit;
    Edit2: TEdit;
    Q1: TQuery;
    Q2: TQuery;
    Memol: TMemo;
    Label3: TLabel;

```

```

Edit3: TEdit;
GroupBox3: TGroupBox;
R2: TRadioButton;
R1: TRadioButton;
Label4: TLabel;
Label5: TLabel;
SpeedButton2: TSpeedButton;
GroupBox4: TGroupBox;
RP1: TRadioButton;
RP2: TRadioButton;
RP3: TRadioButton;
RP4: TRadioButton;
RP5: TRadioButton;
RP6: TRadioButton;
Edit4: TEdit;
Label6: TLabel;
Timer1: TTimer;
Image1: TImage;
Image2: TImage;
procedure SpeedButton1Click(Sender: TObject);
procedure Edit1KeyPress(Sender: TObject; var Key: Char);
procedure FormCreate(Sender: TObject);
procedure SpeedButton2Click(Sender: TObject);
procedure Edit3KeyPress(Sender: TObject; var Key: Char);
procedure Edit2KeyPress(Sender: TObject; var Key: Char);
procedure RP6Click(Sender: TObject);
procedure Timer1Timer(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  Form1: TForm1;

implementation

uses Unit2, Unit3;

{$R *.dfm}

procedure TForm1.SpeedButton1Click(Sender: TObject);
var form2 : TForm2;
begin
  Form2 := TForm2.Create(application);
  if (form2.Caption<>'') then
    begin
      if (edit1.Text<>'') then
        begin
          if (StrToInt(edit1.Text) < 15) then
            messagedlg('Usia anda belum mencukupi untuk mengisi
kuisisioner ini (minimal 15thn)',MtWarning, [mbOk], 0)
          else
            begin
              if ((R1.Checked = false) and (R2.Checked = false)) then
                messagedlg('Anda belum memilih jenis
kelamin',MtWarning, [mbOk], 0)
            end
          end
        end
      end
    end
  end
end

```

```

else if ((RP1.Checked = false) and (RP2.Checked = false)
and (RP3.Checked = false) and (RP4.Checked = false) and
(RP5.Checked = false) and (RP6.Checked = false)) then
    messagedlg('Anda belum memilih jenis pekerjaan',
MtWarning, [mbOk], 0)
else
    begin
        Q2.Close;
        Q2.SQL.Clear;
        Q2.SQL.Add('select nama from responden where nama=:nm
and asal=:as');
        Q2.Params[0].AsString:=edit3.Text;
        Q2.Params[1].AsString:=edit2.Text;
        Q2.Open;
        if Q2.RecordCount=0 then
            begin
                Q1.Close;
                Q1.SQL.Clear;
                Q1.SQL.Add('INSERT INTO responden (nama, asal, umur,
jeniskelamin, kd_kerja) VALUES (:na, :asl, :um, :jk, :pr)');
                Q1.Params[0].AsString := Edit3.Text;
                Q1.Params[1].AsString := Edit2.Text;
                Q1.Params[2].Value := StrToInt(Edit1.Text);
                if (R1.Checked = true) then Q1.Params[3].Value :=
'L';
                if (R2.Checked = true) then Q1.Params[3].Value :=
'P';
                if (RP1.Checked = true) then Q1.Params[4].AsString
:= '1';
                if (RP2.Checked = true) then Q1.Params[4].AsString
:= '2';
                if (RP3.Checked = true) then Q1.Params[4].AsString
:= '3';
                if (RP4.Checked = true) then Q1.Params[4].AsString
:= '4';
                if (RP5.Checked = true) then Q1.Params[4].AsString
:= '5';
                if (RP6.Checked = true) then Q1.Params[4].AsString
:= '6';

                Q1.ExecSQL;
                Edit1.Clear;
                Edit2.Clear;
                Edit3.Clear;
                R1.Checked := false;
                R2.Checked := false;
                RP1.Checked := false;
                RP2.Checked := false;
                RP3.Checked := false;
                RP4.Checked := false;
                RP5.Checked := false;
                RP6.Checked := false;
                Edit3.SetFocus;
                Form2.ShowModal;
                Edit4.Text := '';
            end
        else
            messagedlg('Nama dan asal anda sudah terdaftar!',
MtWarning, [mbOk], 0);

```

```
        end;
        end;
    end
    else
        begin
            messagedlg('Anda belum memasukkan umur', MtWarning,
[mbOk], 0);
            Edit3.SetFocus;
        end;
    end
    else
        begin
            messagedlg('Tidak ada pengisian kuisioner', MtWarning,
[mbOk], 0);
        end;
    end;
end;

procedure TForm1.Edit1KeyPress(Sender: TObject; var Key: Char);
begin
    if not (((key >= #48) and (key <= #57)) or (key =
chr(VK_BACK))) then key := #0;
end;

procedure TForm1.FormCreate(Sender: TObject);
begin
    //ShockwaveFlash1.Movie:= GetCurrentDir + '\Awal.swf';
    mem01.Lines.Add(Q2.fieldByName('informasi').Value);
    Q2.Close;
    Q2.SQL.Clear;
    Q2.SQL.Add('select * from pertanyaan where kd_aspek = 1 and
pertanyaan = ''''');
    Q2.Open;
    if (Q2.RecordCount=10) then
        SpeedButton1.Enabled := false
    else speedbutton1.Enabled := true;
end;

procedure TForm1.SpeedButton2Click(Sender: TObject);
begin
    Form3.showModal;
end;

procedure TForm1.Edit3KeyPress(Sender: TObject; var Key: Char);
begin
    if key=#13 then Edit2.SetFocus;
end;

procedure TForm1.Edit2KeyPress(Sender: TObject; var Key: Char);
begin
    if key=#13 then Edit1.SetFocus;
end;

procedure TForm1.RP6Click(Sender: TObject);
begin
    Edit4.SetFocus;
end;

procedure TForm1.Timer1Timer(Sender: TObject);
```

```
begin
  Q2.Close;
  Q2.SQL.Clear;
  Q2.SQL.Add('select * from pertanyaan where kd_aspek = 1 and
pertanyaan = ''''');
  Q2.Open;
  if (Q2.RecordCount=10) then
    SpeedButton1.Enabled := false
  else speedbutton1.Enabled := true;

end;

end.
```

LISTING PROGRAM TAMPILAN KUISIONER

```
unit Unit2;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics,
  Controls, Forms,
  Dialogs, StdCtrls, ExtCtrls, jpeg, Buttons, DB, DBTables;

type
  TForm2 = class(TForm)
    Notebook1: TNotebook;
    Image1: TImage;
    Panel1: TPanel;
    R1: TRadioGroup;
    L1: TLabel;
    R2: TRadioGroup;
    R3: TRadioGroup;
    R4: TRadioGroup;
    R5: TRadioGroup;
    R6: TRadioGroup;
    R7: TRadioGroup;
    R8: TRadioGroup;
    R9: TRadioGroup;
    R10: TRadioGroup;
    Panel2: TPanel;
    L2: TLabel;
    L3: TLabel;
    L4: TLabel;
    L5: TLabel;
    L6: TLabel;
    L7: TLabel;
    L8: TLabel;
    L9: TLabel;
    L10: TLabel;
    Q1: TQuery;
    Panel8: TPanel;
    Q2: TQuery;
    Q3: TQuery;
    Q4: TQuery;
```

```
Label1: TLabel;
Label3: TLabel;
Label2: TLabel;
Image2: TImage;
SpeedButton1: TSpeedButton;
Label4: TLabel;
procedure seting(item:byte; isi:string);
procedure hilang;
procedure tampilan(item:byte);
procedure Select1(perintah:string);
procedure Select2(perintah:string);
procedure RubahPilihan(a:TRadioGroup;pilihan:byte);
procedure SpeedButton2Click(Sender: TObject);
procedure SpeedButton1Click(Sender: TObject);
procedure FormClose(Sender: TObject; var Action:
TCloseAction);
procedure FormCreate(Sender: TObject);
procedure masukkanilai(item:TRadioGroup;urut, kaspek:byte);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  Form2 : TForm2;
  kaspek: byte;
implementation

{$R *.dfm}

procedure TForm2.Select1(perintah:string);
begin
  Q1.close;
  Q1.SQL.Clear;
  Q1.SQL.Add(perintah);
  Q1.Open;
end;

procedure TForm2.Select2(perintah:string);
begin
  Q2.close;
  Q2.SQL.Clear;
  Q2.SQL.Add(perintah);
  Q2.Open;
end;

procedure TForm2.RubahPilihan(a:TRadioGroup;pilihan:byte);
begin
  a.Items.Clear;
  case pilihan of
    1:begin
      a.Items.Add('SS');
      a.Items.Add('S');
      a.Items.Add('KS');
      a.Items.Add('TS');
```

```
end;
2:begin
    a.Items.Add('SP');
    a.Items.Add('P');
    a.Items.Add('KP');
    a.Items.Add('TP');
end;
3:begin
    a.Items.Add('SPg');
    a.Items.Add('Pg');
    a.Items.Add('KPg');
    a.Items.Add('TPg');
end;
end;
end;

procedure TForm2.hilang;
begin
    R1.Visible:=False;
    R2.Visible:=False;
    R3.Visible:=False;
    R4.Visible:=False;
    R5.Visible:=False;
    R6.Visible:=False;
    R7.Visible:=False;
    R8.Visible:=False;
    R9.Visible:=False;
    R10.Visible:=False;
    R1.ItemIndex:= -1;
    R2.ItemIndex:= -1;
    R3.ItemIndex:= -1;
    R4.ItemIndex:= -1;
    R5.ItemIndex:= -1;
    R6.ItemIndex:= -1;
    R7.ItemIndex:= -1;
    R8.ItemIndex:= -1;
    R9.ItemIndex:= -1;
    R10.ItemIndex:= -1;
    L1.Visible:=False;
    L2.Visible:=False;
    L3.Visible:=False;
    L4.Visible:=False;
    L5.Visible:=False;
    L6.Visible:=False;
    L7.Visible:=False;
    L8.Visible:=False;
    L9.Visible:=False;
    L10.Visible:=False;
end;

procedure TForm2.seting(item:byte; isi:string);
begin
    case item of
        1: Begin
            R1.Visible:= True;
            L1.Visible:= True;
            L1.Caption:= isi;
            End;
    end;
end;
```

```
2: Begin
  R2.Visible:= True;
  L2.Visible:= True;
  L2.Caption:= isi;
  End;
3: Begin
  R3.Visible:= True;
  L3.Visible:= True;
  L3.Caption:= isi;
  End;
4: Begin
  R4.Visible:= True;
  L4.Visible:= True;
  L4.Caption:= isi;
  End;
5: Begin
  R5.Visible:= True;
  L5.Visible:= True;
  L5.Caption:= isi;
  End;
6: Begin
  R6.Visible:= True;
  L6.Visible:= True;
  L6.Caption:= isi;
  End;
7: Begin
  R7.Visible:= True;
  L7.Visible:= True;
  L7.Caption:= isi;
  End;
8: Begin
  R8.Visible:= True;
  L8.Visible:= True;
  L8.Caption:= isi;
  End;
9: Begin
  R9.Visible:= True;
  L9.Visible:= True;
  L9.Caption:= isi;
  End;
10: Begin
  R10.Visible:= True;
  L10.Visible:= True;
  L10.Caption:= isi;
  End;
  end;
end;

procedure TForm2.tampilan(item:byte);
begin
  case item of
    1:begin
      RubahPilihan(R1,1);
      RubahPilihan(R2,1);
      RubahPilihan(R3,1);
      RubahPilihan(R4,1);
      RubahPilihan(R5,1);
      RubahPilihan(R6,1);
```



```
        RubahPilihan(R7,1);
        RubahPilihan(R8,1);
        RubahPilihan(R9,1);
        RubahPilihan(R10,1);
    end;
2:begin
    RubahPilihan(R1,2);
    RubahPilihan(R2,2);
    RubahPilihan(R3,2);
    RubahPilihan(R4,2);
    RubahPilihan(R5,2);
    RubahPilihan(R6,2);
    RubahPilihan(R7,2);
    RubahPilihan(R8,2);
    RubahPilihan(R9,2);
    RubahPilihan(R10,2);
end;
3:begin
    RubahPilihan(R1,3);
    RubahPilihan(R2,3);
    RubahPilihan(R3,3);
    RubahPilihan(R4,3);
    RubahPilihan(R5,3);
    RubahPilihan(R6,3);
    RubahPilihan(R7,3);
    RubahPilihan(R8,3);
    RubahPilihan(R9,3);
    RubahPilihan(R10,3);
end;
4:begin
    RubahPilihan(R1,4);
    RubahPilihan(R2,4);
    RubahPilihan(R3,4);
    RubahPilihan(R4,4);
    RubahPilihan(R5,4);
    RubahPilihan(R6,4);
    RubahPilihan(R7,4);
    RubahPilihan(R8,4);
    RubahPilihan(R9,4);
    RubahPilihan(R10,4);
end;
5:begin
    RubahPilihan(R1,5);
    RubahPilihan(R2,5);
    RubahPilihan(R3,5);
    RubahPilihan(R4,5);
    RubahPilihan(R5,5);
    RubahPilihan(R6,5);
    RubahPilihan(R7,5);
    RubahPilihan(R8,5);
    RubahPilihan(R9,5);
    RubahPilihan(R10,5);
end;
end;
end;

procedure TForm2.masukkannilai(item:TRadioGroup;urut,kaspek:byte);
var n1,n2,n3,n4: integer;
```

```

begin
  if (item.Visible=true) then
    begin
      Q3.Close;
      Q3.SQL.Clear;
      Q3.SQL.Add('select n4,n3,n2,n1 from nilai where nomer=:no and
kd_aspek=:ks');
      Q3.Params[0].Value:=urut;
      Q3.Params[1].Value:=kaspek;
      Q3.Open;
      n1:= Q3.fieldByName('n1').Value;
      n2:= Q3.fieldByName('n2').Value;
      n3:= Q3.fieldByName('n3').Value;
      n4:= Q3.fieldByName('n4').Value;
      Q4.Close;
      Q4.SQL.Clear;
      case item.ItemIndex of
        0: begin
          Q4.SQL.Add('UPDATE nilai SET n4= :nb WHERE nomer=:no
AND kd_aspek=:ks');
          Q4.Params[0].Value:= n4+1;
          end;
        1: begin
          Q4.SQL.Add('UPDATE nilai SET n3= :nb WHERE nomer=:no
AND kd_aspek=:ks');
          Q4.Params[0].Value:= n3+1;
          end;
        2: begin
          Q4.SQL.Add('UPDATE nilai SET n2= :nb WHERE nomer=:no
AND kd_aspek=:ks');
          Q4.Params[0].Value:= n2+1;
          end;
        3: begin
          Q4.SQL.Add('UPDATE nilai SET n1= :nb WHERE nomer=:no
AND kd_aspek=:ks');
          Q4.Params[0].Value:= n1+1;
          end;
      end;
      Q4.Params[1].Value:=urut;
      Q4.Params[2].Value:=kaspek;
      Q4.ExecSQL;
    end;
  end;

procedure TForm2.FormCreate(Sender: TObject);
var perintah:string;
    i:byte;
begin
  perintah:='SELECT * FROM pertanyaan WHERE kd_aspek=1 and
pertanyaan<>''''';
  select2(perintah);
  kaspek:=1;
  if (Q2.RecordCount <> 0) then
    begin
      perintah:='select naspek,pilihan from topik where
kd_aspek=1';
      select1(perintah);
      Panel2.Caption:=Q1.fieldbyname('naspek').AsString;
    end;
end;

```

```

        hilang;
        for i:=1 to Q2.RecordCount do
            begin
                tampilan(Q1.FieldName('pilihan').Value);
                seting(i,      IntToStr(i)      +      '      '      +
Q2.fieldbyname('pertanyaan').AsString);
                Q2.next;
            end;
        end;
    end;

procedure TForm2.SpeedButton1Click(Sender: TObject);
var validasi: boolean;
    perintah: string;
    i: byte;
begin
    validasi:= true;
    If ((R1.Visible = true) and (R1.ItemIndex = -1)) then validasi:=
false;
    If ((R2.Visible = true) and (R2.ItemIndex = -1)) then validasi:=
false;
    If ((R3.Visible = true) and (R3.ItemIndex = -1)) then validasi:=
false;
    If ((R4.Visible = true) and (R4.ItemIndex = -1)) then validasi:=
false;
    If ((R5.Visible = true) and (R5.ItemIndex = -1)) then validasi:=
false;
    If ((R6.Visible = true) and (R6.ItemIndex = -1)) then validasi:=
false;
    If ((R7.Visible = true) and (R7.ItemIndex = -1)) then validasi:=
false;
    If ((R8.Visible = true) and (R8.ItemIndex = -1)) then validasi:=
false;
    If ((R9.Visible = true) and (R9.ItemIndex = -1)) then validasi:=
false;
    If ((R10.Visible = true) and (R10.ItemIndex = -1)) then
validasi:= false;
    If (validasi= false) then MessageDlg('Anda belum menjawab semua
pertanyaan', mtWarning, [mbOK], 0)
    Else
        begin
            if (messagedlg('Anda yakin dengan opsi pilihan untuk
topik ini?', mtconfirmation, [mbyes,mbno], 0) = mryes) then
                begin
                    masukkannilai(R1,1,kaspek);
                    masukkannilai(R2,2,kaspek);
                    masukkannilai(R3,3,kaspek);
                    masukkannilai(R4,4,kaspek);
                    masukkannilai(R5,5,kaspek);
                    masukkannilai(R6,6,kaspek);
                    masukkannilai(R7,7,kaspek);
                    masukkannilai(R8,8,kaspek);
                    masukkannilai(R9,9,kaspek);
                    masukkannilai(R10,10,kaspek);
                    inc(kaspek);
                    perintah:='SELECT * FROM pertanyaan WHERE kd_aspek='
+IntToStr(kaspek) + ' and pertanyaan<>''''';
                    select2(perintah);
                end;
            end;
        end;
end;

```

```

if (Q2.RecordCount<>0) then
begin
perintah:='select naspek,pilihan from topik where
kd_aspek=' + IntToStr(kaspek);
select1(perintah);
Panel2.Caption:=Q1.fieldbyname('naspek').AsString;
hilang;
for i:=1 to Q2.RecordCount do
begin
tampilan(Q1.FieldByName('pilihan').Value);
seting(i, IntToStr(i) + ' ' +
Q2.fieldbyname('pertanyaan').AsString);
Q2.Next;
end;
end;
else
begin
MessageDlg('Terima kasih atas partisipasi Anda
dalam mengisi kuisioner ini', mtInformation, [mbOk], 0);
close();
end;
end;
end;

procedure TForm2.SpeedButton2Click(Sender: TObject);
begin
close();
end;

procedure TForm2.FormClose(Sender: TObject; var Action:
TCloseAction);
begin
action := cfFree;
end;

end.

```

LISTING PROGRAM PEMBUAT

```

unit Unit3;

interface

uses
Windows, Messages, SysUtils, Variants, Classes, Graphics,
Controls, Forms,
Dialogs, Buttons, jpeg, ExtCtrls;

type
TForm3 = class(TForm)
SpeedButton1: TSpeedButton;
Image1: TImage;
procedure SpeedButton1Click(Sender: TObject);
end;

```

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

var
  Form3: TForm3;

implementation

{$R *.dfm}

procedure TForm3.SpeedButton1Click(Sender: TObject);
begin
  close;
end;

end.
```

