

**ANALISIS DAN PERANCANGAN SISTEM BASIS DATA
LABORATORIUM KOMPUTASI
JURUSAN MATEMATIKA FMIPA UNHAS**




PERPUSTAKAAN: RISAT UNIV. HASANUDDIN	
Tgl. Terbit	8 - 7 - 2002
Asal	MIPA
Banyak	1 (Satu)
Harga	Hadiah
No. Inventaris	020708102
No. Klas	

OLEH :

**AKHMAD AFFANDI B
NOMOR POKOK : H 111 97 009**



**JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS HASANUDDIN
2002**



**ANALISIS DAN PERANCANGAN SISTEM BASIS DATA
LABORATORIUM KOMPUTASI
JURUSAN MATEMATIKA FMIPA UNHAS**

SKRIPSI

*Untuk melengkapi tugas-tugas dan memenuhi syarat-syarat untuk
memperoleh gelar Sarjana Sains*

**AKHMAD AFFANDI B.
NOMOR POKOK : H 111 97 009**



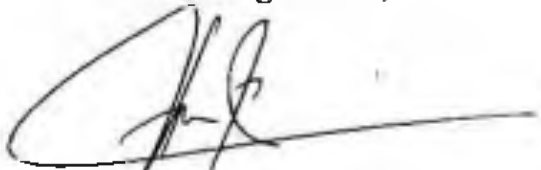
**JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS HASANUDDIN
2002**



**ANALISIS DAN PERANCANGAN SISTEM BASIS DATA
LABORATORIUM KOMPUTASI
JURUSAN MATEMATIKA FMIPA UNHAS**

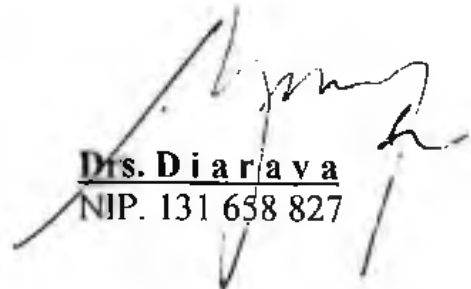
Disetujui Oleh :

Pembimbing Utama,



Drs. Muh. Hasbi, M.Sc.
NIP. 131 650 920

Pembimbing Pertama,



Drs. Diarava
NIP. 131 658 827



ABSTRAK

Penelitian ini bertujuan untuk mengatasi masalah redundansi data serta merancang dan membuat sistem basis data laboratorium komputasi jurusan Matematika FMIPA Unhas dengan sistem yang berbasis komputer. Sedangkan manfaat yang diharapkan adalah untuk mengetahui dengan mudah, cepat, tepat, dan akurat data-data praktikan, asisten, dan jadwal pemakaian laboratorium komputasi. Dalam penelitian ini, sistem yang di rancang menggunakan pendekatan model data relasional. Perancangan sistem basis data laboratorium komputasi ini menggunakan bahasa pemrograman yang berorientasi objek yaitu Borland Delphi Versi 5.0. Hasil penelitian ini menunjukkan bahwa model rancangan program yang ditawarkan mempunyai tingkat efektifitas dan efisiensi yang tinggi sehingga informasi yang dibutuhkan dapat dengan mudah diperoleh.



ABSTRACT

The aim of this research are to solve the problem of redundancy data and to make the database system at the computer laboratory of mathematic, in mathematic department, mathematic and science faculty, Hasanuddin of University based on computer system. So its advantage is to know with easily, quickly, correctly, and accurately apprentice data's, assistant, and the schedule of use the computer laboratory. In this research, the system made using approximation of the relational data model. The programming language object oriented, which used in making this computer laboratory database system is Borland Delphi 5.0 version. The conclusion about this research is to describe that the program given has high effectively and efficiency, so the information that we need can be get easily.



KATA PENGANTAR

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

السلام عليكم ورحمة الله وبركاته

Puji syukur kami panjatkan kehadiran Allah SWT. atas berkat dan karunia-Nya sehingga penulisan skripsi ini dapat terselesaikan untuk memenuhi salah satu syarat guna mencapai gelar Sarjana Sains pada Jurusan Matematika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Hasanuddin.

Penulis menyadari adanya berbagai kekurangan yang terdapat dalam skripsi ini, sebagai akibat keterbatasan pengetahuan penulis. Sehubungan dengan hal tersebut, penulis selalu membuka diri untuk menerima kritikan atau koreksi yang konstruktif dari berbagai pihak sebagai upaya penyempurnaan skripsi yang sederhana ini.

Selesainya skripsi ini atas kerja keras, bimbingan dan dorongan moril dari berbagai pihak, untuk itu pada kesempatan ini penulis ingin menghaturkan penghargaan dan terima kasih sebesar-besarnya kepada :

1. Bapak Drs. Muhammad Hasbi, MSc dan Bapak Drs. Diaraya selaku pembimbing utama dan pembimbing pertama, yang bijaksana telah meluangkan waktu dalam membenkan pengarah dan bimbingan dalam penulisan skripsi ini.
2. Bapak Prof. Dr. H.M. Noor Jalaluddin selaku Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Hasanuddin.
3. Bapak Drs. Nirwan Ilyas, M.Si. selaku Ketua Jurusan Matematika pada Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Hasanuddin.
4. Ibu Dra. Hj. Aidawayati Rangkuti, MS. sebagai Penasehat Akademik selama penulis mengikuti perkuliahan di Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Hasanuddin.



5. Para Dosen dan staf tata usaha pada Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Hasanuddin.
6. Teman-teman angkatan '97 Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Hasanuddin yang bersama dalam membagi rasa suka dan duka selama mengikuti perkuliahan, khususnya Ardi, Ajir, Agusman, Arhamdi, Sahrani, Taju, Mahmuddin, Ade, Lina, Ati, Ophy, Uchi, Temma, Vivi dan Nhanha.
7. Saudara Fajar "Jojonk" dan Alienk yang ikut membantu dalam penyelesaian listing program.
8. Sepupuku Uki, Dhader, lin, Ima, Sule, Faisal, Ilham, dan Ira atas bantuan dan dorongannya.
9. Crew laboratorium Komputasi Jurusan Matematika dan anak-anak Himatika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Hasanuddin .
10. Semua pihak yang tidak dapat disebutkan satu per satu namanya, atas segala bantuannya baik langsung maupun tidak langsung sehingga penulis dapat menyelesaikan skripsi ini.

Akhirnya terima kasih yang tak terhingga pada **Ayahanda Bakri Malla dan Ibunda St. Husnah HP** serta saudara-saudaraku Adhy, Herman, Nony, Afif dan Saleh yang telah mengasuh, mendidik dan mendoakan demi tercapainya cita-cita penulis.

Kiranya Allah SWT. selalu melimpahkan rahmat-Nya kepada kita semua.

Makassar, Juni 2002

Penulis



DAFTAR ISI

Halaman

HALAMAN JUDUL	i
HALAMAN PERSETUJUAN	ii
ABSTRAK	iii
ABSTRACT	iv
KATA PENGANTAR	v
DAFTAR ISI	vii
DAFTAR TABEL	x
DAFTAR GAMBAR	xi
DAFTAR SIMBOL	xii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Pokok Permasalahan	2
1.3 Batasan Masalah	2
1.4 Tujuan dan Manfaat Penelitian	3
1.5 Pertanyaan Penelitian	3
1.6 Sistematika Pembahasan	3
BAB II TINJAUAN PUSTAKA	5
2.1 Analisa Sistem	5
2.1.1 Bentuk dan Analisis Sistem	6

2.1.2	Struktur Analisis	8
2.2	Perancangan Sistem	8
2.2.1	Proses Perancangan	9
2.2.2	Rancangan Terstruktur	11
2.3	Sistem Basis Data	12
2.4	Sistem Informasi	14
2.5	Definisi Operasional Variabel	15
2.6	Mengenal Borland Delphi 5.0	16
BAB III PEMBANGUNAN SISTEM		18
3.1	Rancangan Sistem	18
3.1.1	Rancangan Sistem secara Umum	18
3.1.2	Rancangan Sistem Terinci	20
3.1.2.1	Rancangan Output	20
3.1.2.2	Rancangan Input	28
3.1.2.3	Rancangan Basis Data	33
3.2	Rancangan Teknologi	40
3.3	Pengendalian Aplikasi Sistem	41
3.4	Implementasi Sistem	42
BAB IV PENGUJIAN DAN HASIL PENGUJIAN SISTEM		43
4.1	Pengujian Sistem	40
4.1.1	Metode Pengujian	40
4.1.2	Populasi dan Sampel	41



4.1.3	Teknik Pengumpulan Data	41
4.1.4	Teknik Pengolahan dan Analisis Data.....	42
4.2	Hasil Pengujian Sistem.....	42
BAB V	KESIMPULAN DAN SARAN	51
5.1	Kesimpulan	51
5.2	Saran	51
DAFTAR PUSTAKA		
LAMPIRAN-LAMPIRAN		

DAFTAR TABEL

No. Tabel	Nama Tabel	Halaman
Tabel 3.1	Jadwal.Db	34
Tabel 3.2	Asisten.Db	35
Tabel 3.3	M_Kul.Db	35
Tabel 3.4	Mahasiswa	35
Tabel 3.5	Mk_Praktek.Db	35
Tabel 3.6	Praktikan.Db	36
Tabel 3.7	Aktifitas.Db	36
Tabel 4.1	Data Belum Diolah	42
Tabel 4.2	Waktu Pembuatan Laporan	43
Tabel 4.3	Waktu Pencarian Data	45
Tabel 4.4	Jumlah Tenaga Kerja Dalam Pembuatan Laporan	47
Tabel 4.5	Jumlah Tenaga Kerja Dalam Pencarian Data	48



DAFTAR GAMBAR

No. Gambar	Judul Gambar	Halaman
Gambar 2.1	Konsep Basis Data	13
Gambar 3.1	Diagram Arus Data	19
Gambar 3.2	Output Data Praktikan	22
Gambar 3.3	Output Data Asisten	22
Gambar 3.4	Output Data Mata Kuliah Praktek	23
Gambar 3.5	Output Data Aktifitas Laboratorium	23
Gambar 3.6	Output Data Nilai Praktikum	24
Gambar 3.7	Output Data Jadwal Praktikum	24
Gambar 3.8	Input Data Mahasiswa	25
Gambar 3.9	Input Data Mata Kuliah	26
Gambar 3.10	Input Data Mata Kuliah Praktek	26
Gambar 3.11	Input Data Asisten	27
Gambar 3.12	Input Data Praktikan	27
Gambar 3.13	Input Data Jadwal	28
Gambar 3.14	Input Data Aktifitas	28
Gambar 3.15	Input Data Nilai Praktikum	29
Gambar 3.16	Input Data Absensi	29
Gambar 3.17	Diagram Relasi Entitas (ERD)	33



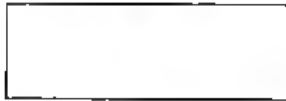
DAFTAR SIMBOL



: Simbol Awal dan Akhir Flowchart



: Simbol Decision of Logical



: Simbol Proses



: Simbol Pemberian Nilai Awal



: Simbol Input dan Output



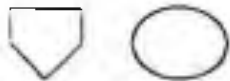
: Simbol Penyimpanan Data



: Simbol Proses Penyimpanan Data ke Disk



: Simbol Display



: Pindah Halaman



BAB I

PENDAHULUAN

1.1 Latar Belakang

Dalam era globalisasi sekarang ini di mana waktu dan informasi menjadi sangat berharga sehingga suatu informasi haruslah cepat, tepat dan akurat. Ini berarti informasi menjadi berguna bagi seseorang apabila informasi tersebut diperoleh pada saat yang tepat, jelas dan sesuai dengan kebutuhan orang tersebut.

Di dalam suatu lembaga pendidikan sangat membutuhkan informasi untuk mengambil suatu keputusan. Salah satu yang ada yaitu informasi kemahasiswaan. Informasi ini meliputi identitas mahasiswa, status mahasiswa, indeks prestasi mahasiswa, kegiatan akademik mahasiswa dan lain-lain.

Untuk beberapa informasi, sistem yang digunakan oleh pihak Universitas Hasanuddin khususnya jurusan Matematika FMIPA Unhas belum menerapkan prinsip-prinsip database yang benar. Data-data yang disimpan di beberapa tempat, sehingga organisasi data tidak begitu baik.

Penyimpanan di beberapa tempat untuk data-data yang sama (redundansi) dapat mengakibatkan pemborosan ruangan penyimpanan, biaya untuk mengakses juga lebih tinggi dan menyebabkan inkonsistensi (tidak konsisten). Dengan sistem basis data dan organisasi data yang benar maka pekerjaan tersebut dapat dengan mudah dilakukan. Akan tetapi itu juga tergantung dari desain atau sistem



yang dirancang untuk mengolah data dengan mempergunakan prinsip-prinsip database yang benar.

Dari sistem yang diterapkan sekarang ini oleh pihak jurusan Matematika tidak efisien karena waktu untuk mencari dan memproses data tersebut agak lama dan menyebabkan informasi menjadi tidak akurat. Hal ini menyebabkan informasi terlambat diperoleh dan waktu yang sebenarnya dapat digunakan untuk keperluan lain terbuang percuma.

1.2 Pokok Permasalahan

Berdasarkan latar belakang masalah yang telah dikemukakan di atas, maka fokus permasalahan yang akan di bahas dalam penelitian ini adalah bagaimana rancangan database yang sesuai dengan pengolahan data di laboratorium komputasi jurusan Matematika FMIPA Unhas.

1.3 Batasan Masalah

Melihat sistem yang ada di laboratorium komputasi jurusan Matematika agak luas cakupannya, maka penulis membatasi hanya pada perancangan sistem basis data praktikan, asisten, dan jadwal pemakaian komputer dalam laboratorium dengan pendekatan model data relasional, dengan judul : “ Analisis dan Perancangan Sistem Basis Data Laboratorium Komputasi Jurusan Matematika FMIPA Unhas ”.



1.4 Tujuan dan Manfaat Penelitian

a. Tujuan Penelitian

Adapun tujuan yang ingin dicapai dalam penelitian ini adalah untuk mengatasi masalah redundansi data serta merancang dan membuat sistem basis data yang cocok digunakan di laboratorium komputasi jurusan Matematika FMIPA Unhas.

b. Manfaat Penelitian

Hasil penelitian ini diharapkan mempunyai manfaat berupa informasi dengan mudah, cepat, tepat dan akurat data-data praktikan, asisten, dan jadwal pemakaian laboratorium, serta kepada mahasiswa lain sebagai bahan referensi untuk penelitian lain yang berminat pada bidang perancangan sistem basis data yang berbasis komputer.

1.5 Pertanyaan Penelitian

1. Apakah rancangan sistem basis data yang dibuat cocok digunakan dalam laboratorium komputasi jurusan Matematika ?
2. Apakah dengan perancangan sistem yang dirancang, pekerjaan akan lebih efektif dan efisien ?



1.6 Sistematika Pembahasan

Dalam penulisan ini sistematika pembahasan dijabarkan sebagai berikut :

Bab I Pendahuluan

Bab ini membahas latar belakang masalah, pokok permasalahan, batasan masalah, tujuan dan manfaat penelitian , pertanyaan penelitian serta sistematika pembahasan.

Bab II Tinjauan Pustaka

Bab ini membahas tentang landasan teori, analisa sistem, rancangan sistem, definisi operationan variabel dan mengenal Borland Delphi 5.0

Bab III Pembangunan Sistem

Bab ini akan dijabarkan spesifikasi kebutuhan melalui rancangan sistem, rancangan terinci, rancangan teknologi dan implementasi sistem.

Bab IV Pengujian dan Hasil Pengujian Sistem

Bab ini membahas tentang metode pengujian yang terdiri atas populasi dan sampel, teknik pengumpulan data, teknik pengolahan, analisis data., dan analisis pengujian sistem.

Bab V Kesimpulan dan Saran

Bab ini berisikan kesimpulan dan saran.

DAFTAR PUSTAKA

BAB II

TINJAUAN PUSTAKA

2.1. Analisa Sistem

Kehadiran sistem komputerisasi di tengah-tengah masyarakat telah memberikan berbagai fasilitas kemudahan, sehingga menjadikan hidup manusia lebih simpel dan praktis, terutama dalam hal pengelolaan sistem informasi

Gagasan suatu sistem informasi yang berbasis komputer tidak berarti otomatisasi total. Dalam hal manusia dan mesin membentuk suatu sistem gabungan dengan hasil yang diperoleh melalui serangkaian dialog dan interaksi antara komputer dengan manusia sebagai pengelola.

Pendekatan konsep manusia atau mesin menunjukkan bahwa perancangan sebuah sistem informasi manajemen harus memahami manusia sebagai pengelola informasi dan perilaku manusia dalam mengambil keputusan. Ini berarti bahwa kemampuan manusia dan mesin (komputer) adalah saling melengkapi.

Berdasarkan pengamatan kami, proses data dan informasi yang dilakukan pada laboratorium komputasi jurusan Matematika FMIPA Unhas masih belum menerapkan prinsip-prinsip database yang benar. Kegiatan proses di dalam laboratorium komputasi belum dilakukan dalam suatu sistem yang tepat. Dilihat dari segi tenaga dan waktu untuk memproses informasi tersebut cukup besar,

inaka pelayanan kebutuhan akan informasi yang terjadi di dalam laboratorium komputasi belum optimal.

2.1.1. Bentuk dan Analisis Sistem

Kegiatan analisis sistem sangat berbeda dengan kegiatan perancangan serta penerapan sistem informasi. Dibanding dengan kegiatan penyelidikan sistem yang lain, kegiatan analisis sangat berorientasi pada manusia dan kurang terstruktur. Kegiatan analisis mempunyai beberapa karakteristik khusus sebagai berikut :

1. Mendefinisikan apa yang seharusnya dilakukan oleh sistem, yang berhubungan dengan masa mendatang, dimana hal ini lebih berupa pendapat.
2. Melakukan negosiasi secara ekstensif, karena masing-masing anggota memiliki pendapat, oleh sebab itu selama fase analisis persetujuan tentang bentuk masalah maupun sistem baru apakah yang dipilih harus dicapai. Seorang analis sistem harus peka pada orang lain serta memiliki kemampuan diplomasi maupun teknis. Hubungan antar manusia selama masa analisis sistem sangatlah kompleks dan seringkali ditemui permusushan yang kadang-kadang ditimbulkan oleh analis.
3. Kompromi adalah kenyataan hidup dalam analisis sistem. seringkali kompromi sangat luas, sehingga meski analis telah melakukan kerja yang luar biasa namun tidak seorang yang menyenangiinya.

4. Perkiraan merupakan acara harian dalam pekerjaan analisis. Biaya diperkirakan, keperluan memory komputer yang diperkirakan, beban pemrosesan masa yang akan datang, jenis paduan transaksi diperkirakan dan bahkan persyaratan harian untuk penyelesaian proyek juga diperkirakan.
5. Hal penting dalam kegiatan analisis adalah berorientasi pada mencegah kesalahan dibanding keberhasilan total. Sebagai contoh, adalah kurang penting untuk menemukan sekumpulan spesifikasi sedemikian sehingga penyelidikan maupun penerapan sistem yang dihasilkan tidak salah. Adalah lebih penting untuk menghindari semua kesalahan kritis selama fase analisis dibanding melakukan hal lain dengan sempurna. Kesalahan kritis seperti mendefinisikan masalah, gagal berkonsultasi dengan pengguna utama sistem, atau meremehkan kapasitas sistem yang diperlukan, dapat mengakibatkan gagalnya seluruh proyek. Kesalahan proyek sistem kebanyakan diakibatkan oleh kesalahan yang dilakukan dalam fase analisis

Dengan demikian, untuk menghindari terjadinya kesalahan sistem yang akan diterapkan dalam suatu proyek, maka seorang analis diharuskan memiliki keahlian komplit. Terutama pada hal pelatihan teknis pada sistem informasi. Dan yang paling penting adalah kemampuan berhubungan dengan orang lain, keinginan untuk berkompromi dan pengetahuan tentang kesalahan kritis yang harus dihindari. Hal lain yang juga penting

adalah pemahaman tentang organisasi dan industrinya, terutama tentang fungsi-fungsi yang akan dianalisis.

2.1.2. *Struktur Analisis*

Analisis terstruktur merupakan suatu pendekatan analisis sistem yang bergantung pada pendekatan produk akhir fase analisis sistem yaitu spesifikasi persyaratan sistem. Dengan mempergunakan analisis terstruktur, maka spesifikasi persyaratan sistem diformulasikan dalam bentuk serangkaian modul dari pada sekumpulan spesifikasi monolitik.

Ada tiga tujuan modularisasi yaitu :

1. Masing-masing analisi dari suatu tim proyek dapat bekerja terpisah dan efisien pada modul yang berbeda.
2. Modul terstruktur lebih mudah dipahami oleh anggota kegiatan.
3. Modularisasi memungkinkan segenap aspek dari spesifikasi sistem diperbaharui secara terpisah dan cepat.

2.2. Perancangan Sistem

Perancangan sistem menurut **Robert J. Verzello** (Perancangan Sistem: Informasi, 1992) yaitu penentuan spesifikasi yang memenuhi kebutuhan dan persyaratan melalui tahap analisis sistem meliputi mengumpulkan data tentang sistem yang ada dan ruang lingkup pengembangan sistem yang baru melalui pengumpulan data studi para pemakai informasi dan bekerja sama dengan mereka untuk menemukan dan menentukan kebutuhan pemakai.

Perancangan sistem menurut Gergo M.Scott. (Prinsip-Prinsip Sistem Informasi Manajemen, 1995) didefinisikan sebagai suatu proses memahami sistem kemudian dilanjutkan dengan merancang sistem informasi yang berbasis komputer yang akhirnya akan menghasilkan sistem komputerisasi.

Rancangan sistem menentukan bagaimana mencapai sasaran yang ditetapkan, melibatkan pembentukan (*configurating*) perangkat lunak dan komponen perangkat keras sistem, dengan demikian sesudah pemasangan, sistem akan memenuhi spesifikasi yang dibuat pada akhir fase analisis sistem. Aspek lebih lanjut dari analisis sistem adalah membentuk sistem sehingga ia dapat diterima baik oleh pengguna sistem maupun operator. Jika sistem yang dirancang tidak dapat memenuhi spesifikasi yang dibuat dan tidak dapat diterima oleh pengguna dan operator, seperti kadang-kadang ditemui, maka kegiatan analisis sistem harus dimulai sejak awal dan spesifikasi sistem harus dimodifikasi. Keuntungan dari analisis terstruktur adalah hanya ada satu atau sedikit dari banyak modul rancangan yang masih memerlukan perbaikan (*revision*).

2.2.1. Proses Perancangan

Analisis sistem adalah kegiatan yang berorientasi pada manusia dan bersifat tidak terstruktur, yang melibatkan perkiraan (*estimates*) dan negosiasi, sedangkan perancangan sistem pada dasarnya lebih terstruktur dan teknis. Perancangan sistem membutuhkan tingkat keterampilan (*skill*) yang tinggi, sedangkan analisis sistem sangat memerlukan keterampilan antar perorangan yang tinggi. Bagaimanapun juga, interaksi diantara

anggota tim perancang (*design team*) sangat intensif, sehingga kemampuan perancang untuk bekerja sama satu sama lain merupakan syarat utama. Ada sejumlah kegiatan khusus yang dilakukan selama fase perancangan dan tim proyek (*project team*) untuk merancang proyek sistem yang besar biasanya beranggotakan para programmer, perancang file (*file designer*), ahli pengendalian masukan (*input control specialist*), ahli akuisisi perangkat keras (*hardware acquisition experts*), ahli manajemen proyek (*project management specialists*), ahli jaringan komunikasi (*telecommunication network experts*) dan konsultan ahli (*specialized consultants*), meskipun mereka tidak dilibatkan dalam waktu yang sama.

Konsultan sistem sering dilibatkan selama tahap perancangan dan penerapan, karena kedua tahap tersebut melibatkan beberapa kegiatan sangat teknis dan khusus, dan ada kemungkinan bahwa organisasi bersangkutan tidak mempunyai personil yang menguasai seluruh keahlian yang dibutuhkan. Diantara personil eksternal yang sangat sering dipekerjakan adalah konsultan pemilihan perangkat keras, pemrogram yang dikontrak, konsultan pemilihan perangkat lunak dan konsultan manajemen proyek. Perusahaan akuntan publik juga dapat diminta untuk menyakinkan bahwa pengendalian sistem yang memadai benar-benar telah dirancang dan sistem tersebut memang dapat diaudit (*auditable*).



2.2.2. Rancangan Terstruktur

Rancangan terstruktur dimulai dengan penyusunan serangkaian rancangan alternatif sistem terbesar yang memungkinkan oleh lingkup (*scope*) proyek yang ditetapkan (yang disebut alternatif rancangan makro), dan dilanjutkan dengan serangkaian kelompok alternatif yang lebih kecil hingga diperoleh alternatif rancangan mikro (*micro design alternative*) yang sepenuhnya mendefinisikan sistem secara terinci. Pendekatan berlapis yang bermula dari rancangan makro ini selanjutnya diteruskan melalui lapisan berganda hingga tercapai rancangan makro akhir, yang kadang-kadang disebut perbaikan berturut (*successive refinement*) atau rancangan dari atas ke bawah (*top-down design*).

2.3. Sistem Basis Data

Di dalam penyusunan basis data dengan komputer tidaklah harus menggunakan Sistem Manajemen Basis Data (SMBD) yang bersifat untuk tujuan umum seperti yang halnya SMBD yang banyak ditawarkan di pasaran. Dalam kasus tertentu dapat juga dibuat perangkat lunak SMBD sendiri sesuai dengan tujuan khusus penyusunan basis data.

Namun demikian, fungsi pengelolaan basis data baik itu menggunakan SMBD yang dibuat sendiri bertujuan untuk dapat memanipulasi data dari basis data sehingga diperoleh informasi yang sesuai dengan yang diinginkan. Gabungan antara basis data dengan perangkat lunak SMBD termasuk di

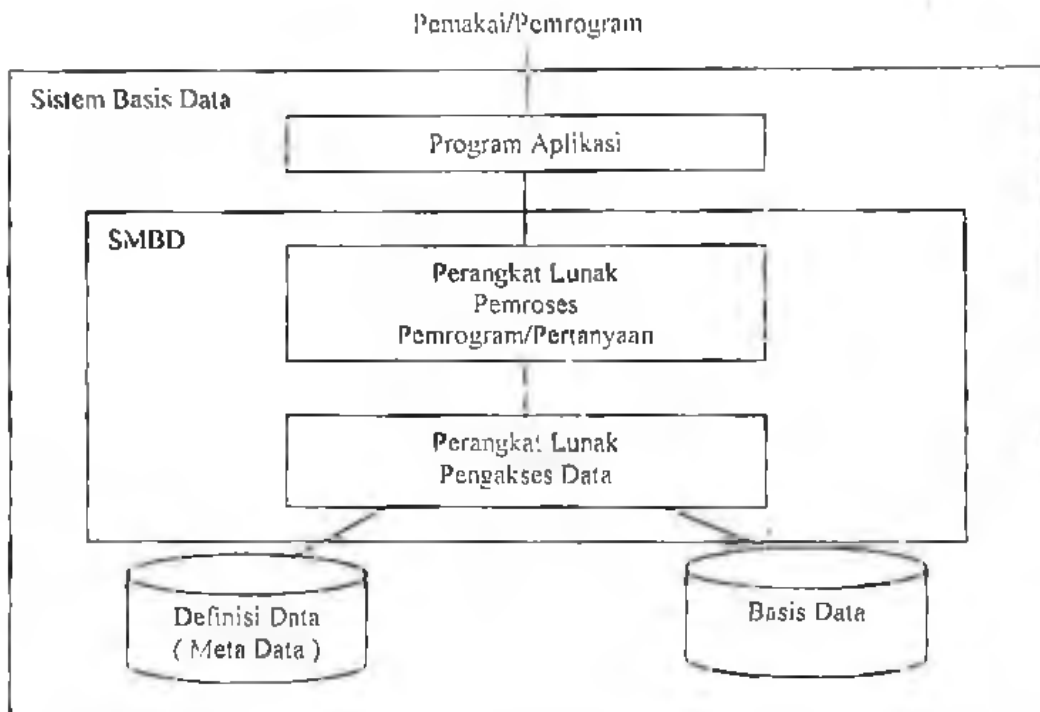
dalamnya program aplikasi yang dibuat dan bekerja dalam satu sistem, selanjutnya disebut *Sistem Basis Data*.

Menurut Waljiyanto dalam bukunya "Sistem Basis Data Analisis dan Pemodelan Data", basis data dapat diartikan sebagai kumpulan data tentang suatu benda atau kejadian yang saling berhubungan satu sama lain. Sedangkan data adalah fakta yang mewakili suatu obyek seperti manusia, hewan, peristiwa, konsep, keadaan dan sebagainya, yang dapat di catat dan mempunyai arti yang implisit. Data dapat dicatat atau direkam dalam bentuk angka, huruf, simboi, gambar, bunyi, atau kombinasinya.

Sistem Manajemen Basis Data (SMBD) adalah kumpulan program yang digunakan untuk membuat dan mengelola basis data. Suatu SMBD merupakan sistem perangkat lunak untuk melakukan pemrosesan dalam hal pendefinisian, penyusunan, dan manipulasi basis data untuk berbagai aplikasi.

Dalam sebuah sistem basis data secara lengkap terdapat komponen-komponen utama sebagai berikut :

1. Perangkat keras (Hardware)
2. Sistem operasi (Operating System)
3. Basis data (Database)
4. Sistem (Aplikasi/Perangkat Keras) Pengelola Basis Data (DBMS)
5. Pemakai (User)
6. Aplikasi (Perangkat Lunak) lain.



Gambar 2.1 Konsep Basis Data (kompilasi dari Elmasari R. dkk, 1994)

2.4. Sistem Informasi

Sistem informasi adalah suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengolahan data harian, mendukung operasi, bersifat managerial dan kegiatan strategi dari suatu organisasi serta menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan (Robert A. Leitch dan K. Roscoe Davis).

Menurut Leman (Metodologi Pengembangan Sistem Informasi, 1997), sistem informasi dapat didefinisikan sebagai suatu sistem yang dibuat oleh manusia yang terdiri dari komponen-komponen dalam organisasi untuk mencapai



suatu tujuan yakni menyajikan suatu informasi. Sistem informasi meliputi dua hal yaitu komponen dan kegiatan. Adapun unsur-unsurnya meliputi :

b. Komponen, terdiri atas :

1. Perangkat keras, terdiri dari komputer, periferifal dan jaringan
2. Perangkat lunak, merupakan kumpulan dari perintah atau fungsi yang ditulis dengan aturan tertentu untuk memerintahkan komputer melaksanakan tugas tertentu.
3. Data, merupakan komponen dasar dari informasi yang akan diproses lebih lanjut untuk menghasilkan informasi
4. Manusia, yang terlibat dalam komponen manusia seperti operator, pemimpin sistem informasi dan sebagainya.
5. Prosedur, seperti dokumentasi prosedur/proses sistem, buku penuntun operasional dan teknis.

c. Kegiatan, terdiri atas :

1. Input, menggambarkan suatu kegiatan untuk menyediakan data untuk diproses.
2. Proses, menggambarkan bagaimana suatu data diproses untuk menghasilkan suatu informasi yang bernilai tambah
3. Output, suatu kegiatan untuk menghasilkan laporan dari suatu proses data.
4. Penyimpanan, suatu kegiatan untuk memelihara dan menyimpan data.
5. Kontrol, suatu aktivitas untuk menjamin bahwa sistem informasi tersebut berjalan sesuai dengan yang diharapkan.

2.5. Definisi Operasional Variabel

Variabel operasional yang akan digunakan akan kami batasi sesuai ruang lingkup pembahasan penelitian ini sebagai berikut :

- a. Informasi adalah data yang telah diolah menjadi sebuah bentuk yang berarti bagi penerimanya dan bermanfaat didalam mengambil keputusan saat ini atau mendatang. Dalam kategorinya dapat dikelompokkan menjadi :
 1. Informasi strategis, yaitu informasi yang digunakan untuk mengambil keputusan jangka panjang, mencakup informasi eksternal, rencana perluasan perusahaan dan sebagainya.
 2. Informasi taktis, informasi ini dibutuhkan untuk mengambil keputusan jangka menengah seperti informasi trend penjualan yang dapat dipakai untuk menyusun rencana-rencana penjualan
 3. Informasi teknis, informasi ini dibutuhkan untuk keperluan operasional sehari-hari.
- b. Ketepatan berarti informasi harus bebas dari kesalahan-kesalahan. Ketepatan juga berarti informasi harus jelas, kerana dari sumber informasi ke penerima informasi mungkin terjadi gangguan yang dapat merubah atau merusak informasi tersebut.
- d. Kecepatan berarti informasi yang dihasilkan harus tepat pada waktunya dimana informasi yang datang pada penerima tidak boleh terlambat.
- e. Akurat berarti bahwa informasi yang dihasilkan secara nyata dan mempunyai nilai kebenaran yang tinggi, sudah diteliti dengan cermat dan seksama.



- f. Efisiensi yang dimaksud oleh penulis adalah kemampuan menghasilkan informasi yang cepat, tepat dan mudah dipahami.
- g. Efektivitas berarti bahwa informasi yang dihasilkan dapat berdaya guna dan berhasil guna.

2.6. Mengenal Borland Delphi 5.0

Dalam pembuatan sistem basis data laboratorium komputasi jurusan Matematika, maka dirasa perlu untuk memperkenalkan bahasa pemrograman yang berorientasi object yang disebut Borland Delphi dengan versi yang digunakan sekarang adalah 5.0.

Delphi sebenarnya kelanjutan dari Turbo Pascal, yang merupakan produk Borland yang berkedudukan di Scotts Valley, California. Borland setelah melebur dengan Visigenic berganti nama menjadi Inprise Corporation pada pertengahan 1998.

Turbo Pascal untuk MS-Windows diperkenalkan tahun 1990, sampai terakhir versi dari Turbo Pascal ini adalah versi 7.0 tahun 1992. Pada tahun 1993 dimulailah suatu pekerjaan besar oleh Borland (saat itu) untuk membangun sebuah bahasa pemrograman yang lebih canggih dan lebih mudah lagi yang dinamakan Delphi.

Delphi secara resmi diumumkan pada tanggal 14 Februari 1995 dan rilisnya dilakukan pada tanggal 28 Februari tahun yang sama. Sekarang Borland

telah berganti nama menjadi Inprise setelah bergabung dengan Visigenic, dan Delphi telah memiliki versi yang terakhir yaitu Delphi 6.0:

Selain sebagai aplikasi biasa dengan mudah Delphi membuat aplikasi untuk internet. Seperti Win-CGI (Windows Common Gateway Service), ISAPI (Internet Service Application Program Interface) serta Misapi (Netscape-ISAPI). Bahkan Borland menyatakan sebagai satu langkah menuju Active X (One Step to Active X). Tentu pernyataan ini terbukti dengan mudahnya kita membuat sebuah aplikasi ActiveForm yang dapat berjalan di Internet Explorer sebagai suatu aplikasi biasa. ActiveX adalah salah satu teknologi yang diperkenalkan oleh Microsoft pada mulai pada Windows 95.



BAB III

PEMBANGUNAN SISTEM

3.1. Rancangan Sistem

Menurut **Noor Fuad** (*Analisis dan Perancangan Sistem*, 1988) sistem adalah kumpulan dari beberapa prosedur yang dirancang dan disusun sedemikian rupa untuk mencapai sasaran yang telah ditetapkan. Sedangkan menurut **G.B. Davis**, sistem adalah kesatuan unsur-unsur atau elemen yang saling berinteraksi satu sama lain untuk mencapai suatu tujuan. Dan menurut **Leman** (*Metodologi Pengembangan Sistem Informasi*, 1998) sistem terdiri dari komponen-komponen yang saling berkaitan dan bekerja sama untuk mencapai suatu tujuan.

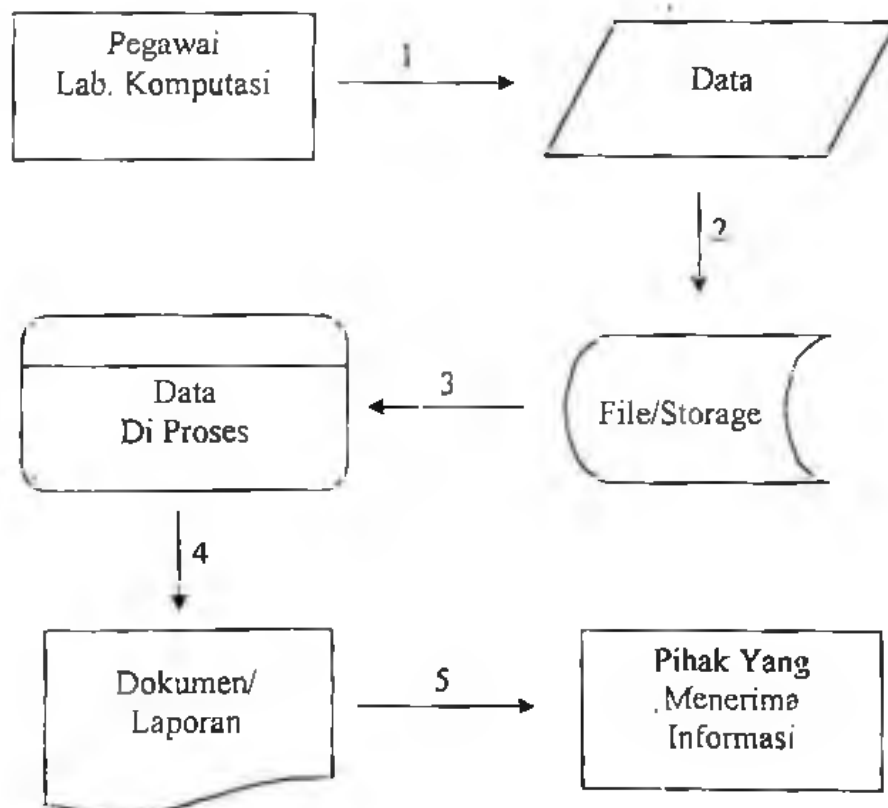
Rancangan sistem merupakan suatu kegiatan yang dilakukan untuk merancang suatu sistem dengan menggambarkan bagaimana sistem itu di bentuk. Hal ini bertujuan untuk memberi gambaran yang jelas kepada pemakai software ataupun pemrogram tentang sistem yang akan dibuat. Rancangan sistem terdiri dari dua bagian yakni rancangan sistem secara umum dan rancangan sistem secara terinci.

3.1.1. Rancangan Sistem Secara Umum

Tujuan dari rancangan sistem secara umum adalah untuk memberikan gambaran secara umum kepada pemakai tentang sistem

yang baru. Pada tahap rancangan secara umum, komponen-komponen sistem basis data yang dirancang dengan tujuan untuk dikomunikasikan kepada pemakai

Untuk lebih menggambarkan proses perancangan sistem secara umum digunakan diagram arus data (*Data Flow Diagram*). Diagram arus data tersebut digunakan untuk menggambarkan suatu sistem baru yang akan dikembangkan secara logika tanpa mempertimbangkan lingkungan fisik dimana data tersebut disimpan. Diagram arus data tersebut digambarkan sebagai berikut :



Gambar 3.1 Diagram Arus Data

Keterangan :

1. Pegawai Lab.Komputasi menerima data-data yang dibutuhkan dari mahasiswa dan dosen.
2. Data kemudian di input dan diolah dalam komputer dan disimpan dalam storage
3. Data kemudian diproses dalam system yang telah dirancang
4. Hasil proses data data, kemudian diterbitkan dalam bentuk laporan
5. Laporan/Dokumen kemudian dilihat oleh pihak yang membutuhkan informasi.

3.1.2. Rancangan Sistem Terinci

Dari penjelasan pada sub bagian sebelumnya, bahwa rancangan sistem secara umum merupakan persiapan dari rancangan terinci. Untuk rancangan sistem terinci akan dijelaskan sebagai berikut :

3.1.2.1. Rancangan Output

Output merupakan produk dari sistem informasi yang dapat dilihat. Output ini dapat berupa hasil yang dikeluarkan ke media kertas (kertas dan lain-lain) dan output yang berupa hasil dikeluarkan kemedialunak (tampilan layar)

Rancangan output ini dapat dilakukan dengan langkah-langkah sebagai berikut :

1. Menentukan kebutuhan output dari sistem yang ditawarkan. Output yang akan di desain telah dapat ditentukan dari diagram arus data sistem yang ditawarkan yang telah dibuat.
2. Menentukan parameter output. Setelah output-output yang akan di desain ditentukan, maka parameter dari output juga dapat ditentukan. Parameter ini meliputi : type dari output, format, media yang digunakan , alat output yang digunakan, dari sistem yang ditawarkan dan lain-lain.

Program aplikasi sistem basis data Laboratorium Komputasi Jurusan Matematika FMIPA Unhas, memiliki output berupa laporan, baik laporan yang dicetak pada layar monitor maupun laporan yang dicetak pada printer.

Sebanyak 6 macam output yang telah penulis rancang secara terinci. Output ini berupa laporan-laporan yang dibutuhkan. Bentuk dari output tersebut dapat dilihat pada gambar-gambar berikut :

1. Output Info/Laporan Data Mahasiswa Praktikan

The screenshot shows a window titled "Info Mahasiswa". At the top, there are search filters: "Urut Berdasarkan" (Sorted by) with options "Stambuk", "Kd Praktikan", and "Nama"; "Cari Nama" (Search Name) with a text input field; and "Cari Stambuk" (Search Stambuk) with a dropdown menu. Below the filters is a table with columns "Stambuk", "Kode Praktikan", and "Nama". The table contains 15 rows of student data. At the bottom, there is a detailed view for the selected student H11199001, showing "Nama" (ANDI RAHMA), "Jurusan" (MATEMATIKA), and "Alamat" (KOMP KODAH).

Stambuk	Kode Praktikan	Nama
H11199002	PADA0001	ABDUL HAMID
H11199001	PADA0002	ANDI RAHMA
H11199003	PADA0003	IBAR
H11199006	PADA0004	RAHMAT
H12100002	PADA0005	DESI RATNASARI
H12100009	PADA0006	ANDI ULENG
H11199007	PADA0007	FAJAR
H11199001	PANU0001	ANDI RAHMA
H11199002	PANU0002	ABDUL HAMID
H12100003	PANU0003	ANDI ULENG
H11197002	PANU0004	MUHAMMAD ARMAN
H11197003	PANU0005	ARDIANSYAH
H11197005	PANU0006	ANDI BASO

Stambuk	Nama	Jurusan	Alamat
H11199001	ANDI RAHMA	MATEMATIKA	KOMP KODAH

Gambar 3.2. Output Data Praktikan

2. Output Info/Laporan Data Asisten

The screenshot shows a window titled "Info Asisten Laboratorium - Komputasi". At the top, there are search filters: "Urut Berdasarkan" (Sorted by) with options "Stambuk", "Kd Asisten", and "Nama"; "Cari Stambuk" (Search Stambuk) with a dropdown menu; and "Cari Berdasarkan Nama" (Search by Name) with a text input field. Below the filters is a table with columns "Stambuk", "Kode Asisten", and "Nama". The table contains 15 rows of assistant data. At the bottom, there is a detailed view for the selected assistant H11197001, showing "Nama" (A.HERAWATI), "Jurusan" (MATEMATIKA), "Alamat" (JL. BUNG NG 2), and "Tanggal Lahir" (14Feb78).

Stambuk	Kode Asisten	Nama
H11197001	AADA0001	A.HERAWATI
H11197002	AADA0002	MUHAMMAD ARMAN
H11197003	AADA0003	ARDIANSYAH
H11197005	AADA0004	ANDI BASO
H12197001	AADA0005	IRNAYANTI BISMA
H12197004	AADA0006	HENDRA WIBOWO
H12197002	AADA0007	IRMA SUDJATI
H11197003	AANU0001	AFOWNSYAH
H11197005	AANU0002	ANDI BASO
H11197001	AANU0003	A.HERAWATI
H12197004	AANU0004	HENDRA WIBOWO
H11197004	AARE0001	ANTI

Stambuk	Nama	Jurusan	Alamat	Tanggal Lahir
H11197001	A.HERAWATI	MATEMATIKA	JL. BUNG NG 2	14Feb78

Gambar 3.3. Output Data Asisten



3. Output Info/Laporan Mata Kuliah Praktek

Info Mata Kuliah

Informasi Mata Kuliah

Urut Berdasarkan: Kd Mk Praktek Kd MatKul Nm Matkul

Cari Berdasarkan Nama Mata Kuliah: _____

Kd Mk Praktek	Kd Mata Kuliah	Mata Kuliah
PIK	102AM3	Pengenalan Ilmu Komputer
LMA	152AM3	Logika Matematika
ADA	171AM4	Analisa Data
ANA	253AM3	Analisa Numerik
PLI	255AM3	Program Linier
FDE	302AM4	Perencanaan Defensial
SDA	355AM3	Struktur Data
ARE	372AM4	Analisa Regresi
MPE	377AM3	Metode Penelitian

Kode_mata_kuliah	Nama_mata_kuliah	Sks
253AM3	Analisa Numerik	3

Gambar 3.4. Output Data Mata Kuliah Praktek

4. Output Laporan Data Aktifitas Laboratorium

Info Aktivitas Laboratorium Komputasi

Urut Berdasarkan: Kd Praktek Kd Aktifitas Kd Waktu Inn Akod

Kode akt_praktek	Kode_praktek	Kode_aktifitas	Kd Waktu	Tahun_ekadent
ADA	PADA0004	AADA0001	R901	Aktif 2001/2002
ADA	PADA0005	AADA0003	SN01	Aktif 2001/2002
ADA	PADA0001	AADA0004	SL01	Aktif 2001/2002
ADA	PADA0003	AADA0005	SL01	Aktif 2001/2002
ADA	PADA0007	AADA0007	SN01	Aktif 2001/2002
ANU	PANU0002	AANU0001	SN01	Aktif 2001/2002
ANU	PANU0003	AANU0001	R803	Aktif 2002/2003
ANU	PANU0006	AANU0001	SN01	Aktif 2001/2002
ANU	PANU0001	AANU0003	SL01	Aktif 2001/2002

Kode_Praktek	Nama	Kode_Aktifitas	Nama
PADA0004	LALA	AADA0001	PAJAR
PADA0004	RAHMAT	AADA0001	AJERAWATI

Gambar 3.5. Output Data Aktivitas Laboratorium

5. Output Info/Laporan Data Nilai Praktikan

The screenshot shows a window titled "Info Nilai Praktikan". At the top, there are sorting options: "Urut Berdasarkan:" with radio buttons for "Kd Praktikan" (selected), "Kd Waktu", and "Tahun Akademik". Below this is a search field "Cari Berdasarkan Kode Praktikan" with a dropdown menu and a "Bersihkan Filter" button. The main area contains a table with the following data:

Kode Mh Praktikan	Kode Praktikan	Kode Waktu	Tahun Akademik	Nilai
ADA	PADA0004	RB01	Akhir 2001/2002	45
ANU	PANU0003	RB03	Akhir 2002/2003	52
ADA	PADA0001	SL01	Akhir 2001/2002	22
ADA	PADA0003	SL01	Akhir 2001/2002	44
ANU	PANU0001	SL01	Akhir 2001/2002	11
ADA	PADA0005	SN01	Akhir 2001/2002	90
ADA	PADA0007	SN01	Akhir 2001/2002	89
ANU	PANU0002	SN01	Akhir 2001/2002	95
ANU	PANU0008	SN01	Akhir 2001/2002	89

Below the table, there are two detail boxes. The first shows "Kode Praktikan" PANU0003 and "Nama" ANQIULENG. The second shows "Kode Pr. praktik" ANU and "Mata Kuliah" Analisa Numeric.

Gambar 3.6. Output Data Nilai Praktikan

6. Output Info/Laporan Data Jadwal Praktikum

The screenshot shows a window titled "Info Jadwal Praktikum". At the top, there are sorting options: "Urut Berdasarkan:" with radio buttons for "Kd Mh Prak." (selected), "Kd Waktu", and "Thn Akademik". Below this is a search field "Cari Berdasarkan Kode Praktikan" with a dropdown menu and a "Bersihkan Filter" button. The main area contains a table with the following data:

Kode Mh Praktikan	Kode Waktu	Tahun Akademik
ADA	RB01	Akhir 2001/2002
ANU	RB03	Akhir 2002/2003
ADA	SL01	Akhir 2001/2002
ADA	SL01	Akhir 2001/2002
ANU	SL01	Akhir 2001/2002
ADA	SN01	Akhir 2001/2002
ADA	SN01	Akhir 2001/2002
ANU	SN01	Akhir 2001/2002
ANU	SN01	Akhir 2001/2002

Below the table, there are two detail boxes. The first shows "Kode Praktikan" ADA and "Mata Kuliah" Analisa Data. The second shows "Kd waktu" RB01, "Nama Hari" Rabu, and "Waktu" 08.00 - 10.00.

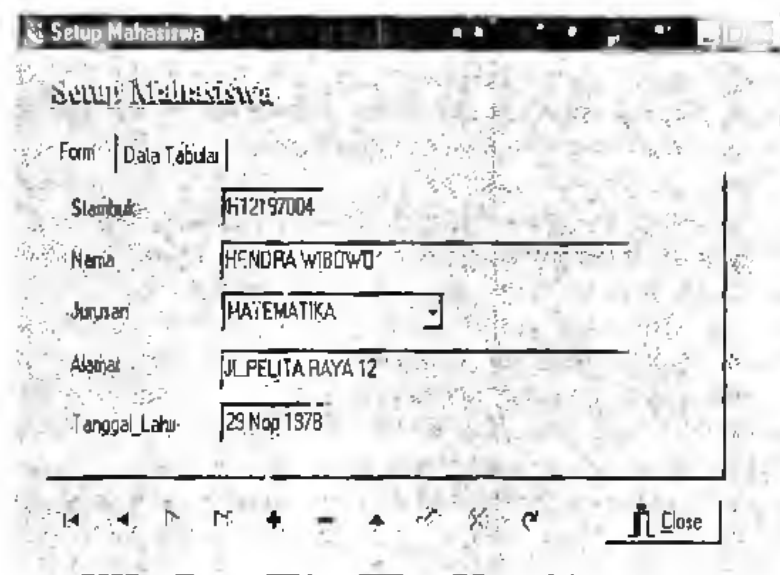
Gambar 3.7. Output Data Jadwal Praktikum

3.1.2.2. Rancangan Input

Input merupakan tahapan yang dilakukan untuk memasukkan data kedalam proses komputer melalui alat input. Ada beberapa form input data dalam penulisan tugas akhir ini, yaitu sebagai berikut :

1. Form Input Data Mahasiswa

Form input data mahasiswa ini digunakan untuk menginput data master mahasiswa dengan field kunci stambuk.



Form	Data Tabular
Stambuk	412197004
Nama	HENDRA WIBOWO
Jurusan	MATEMATIKA
Alamat	JL. PELITA RAYA 12
Tanggal_Lahir	29 Nov 1978

Gambar 3.8. Input Data Mahasiswa

2. Form Input Data Mata Kuliah

Form ini digunakan untuk menginput data-data mata kuliah yang ada di Jurusan Matematika FMIPA Unhas, dengan field kuncinya adalah kode mata kuliah.

Setup Mata Kuliah

Form | Data Tabular

Kode Mata Kuliah | 366AM3

Nama Mata Kuliah | Struktur data

S K S | 3

Close

Gambar 3.9. Input Data Mata Kuliah

3. Form Input Data Mata Kuliah Praktek

Form ini digunakan menginput jenis mata kuliah praktek yang ada dalam form mata kuliah dengan memasukkan kode mata kuliah praktek sebanyak 3 digit.

Setup Mata Kuliah Praktek

Form | Data Tabular

Nama Mata Kuliah | Metode Penelitian
 PD Persial. S mata Bata
 Pengenalan Ilmu Komputer
 Persamaan Diferensial
 Program Linier

Terpilih | Pengenalan Ilmu Komputer

Kode mata kuliah | 102AM3

Kode mata praktek | PIK

Close

Gambar 3.10. Input Data Mata Kuliah Praktek

4. Form Input Data Asisten

Form ini digunakan untuk menginput data mahasiswa yang menjadi asisten yang secara otomatis akan mendapatkan kode asisten sesuai mata kuliah praktek.

Stambuk	Nama	Jurusan	Kode Ml Kuliah Praktek	Kode Asisten
H11197003	ARDIANSYAH	MATEMATIKA	ANU	AANU0004

Gambar 3.11. Input Data Asisten

5. Form Input Data Praktikan

Form ini digunakan untuk menginput data mahasiswa yang menjadi praktikan dalam suatu mata kuliah praktek .

Stambuk	Nama	Jurusan	Kode Ml Kuliah Praktek	Kode Praktikan
H11197002	ABDUL HAMID	MATEMATIKA	ANU	PANU0006

Gambar 3.12. Input Data Praktikan



6. Form Input Data Jadwal

Form ini digunakan untuk menginput jadwal praktikum dan akan menghasilkan secara otomatis kode waktu dan merupakan field kunci.

Jadwal

Setup Jadwal

Form | Data Tabular |

Kd_waktu:

Nama_Hari:

Waktu: 07.30 - 09.10 14.00 - 15.40
 09.20 - 11.00 15.50 - 17.30
 11.10 - 12.30 Lain-lain

Toolbar: [Left Arrow] [Right Arrow] [Home] [End] [F5] [F6] [F7] [F8] [F9] [F10] [F11] [F12] [Enter] [Esc] [Keluar]

Gambar 3.13. Input Data Jadwal

7. Form Input Data Aktifitas Laboratorium

Ini digunakan untuk menginput data kegiatan yang terjadi dalam laboratorium

Input Aktivitas

Nama Mata Kuliah Praktikum	Kd praktikum	Nama Praktikan	Id pengajar	Kd waktu	Tanggal
Kelompok Data	PAD2001	ACHMEDI	ANDI MEGI	0201	April 2001/2002
Arus Data	PAD2002	YUSRI	AACHADIS	0201	April 2001/2002
Kelompok Data	PAD2003	HABIBAT	AACHADIS	0201	April 2001/2002
Kelompok Data	PAD2004	DESI BALANAYAN	AACHADIS	0201	April 2001/2002
Kelompok Data	PAD2005	FAMA	AACHADIS	0201	April 2001/2002
Kelompok Data	PAD2006	ANDI PANGRA	AACHADIS	0201	April 2001/2002
Kelompok Data	PAD2007	ABDI, HARDO	AACHADIS	0201	April 2001/2002
Kelompok Data	PAD2008	ABDI LESMI	AACHADIS	0201	April 2001/2002

Toolbar: [Left Arrow] [Right Arrow] [Home] [End] [F5] [F6] [F7] [F8] [F9] [F10] [F11] [F12] [Enter] [Esc] [Data]

Gambar 3.14. Input Data Aktivitas

8. Form Input Data Nilai Praktikum

Form ini digunakan untuk memberikan nilai praktikum kepada setiap praktikan.

Nama Praktikan	Nama Praktikan	No. Praktikan	No. Waktu	Tanggal	Nilai
Praktikan 1	PANAZIUS	0101	0101	10/01/2002	10
Praktikan 2	PANAZIUS	0102	0102	10/01/2002	10
Praktikan 3	PANAZIUS	0103	0103	10/01/2002	10

Gambar 3.15. Input Data Nilai Praktikum

9. Form Input Data Absensi

Form ini digunakan untuk mengabsensi praktikan setiap pertemuan.

Nama Praktikan	Nama Praktikan	No. Praktikan	No. Waktu	Tanggal	Absensi
Praktikan 1	DETA PANGRAGAN	0101	0101	10/01/2002	M
Praktikan 2	Andi	0102	0102	10/01/2002	M
Praktikan 3	ABDUL HAMID	0103	0103	10/01/2002	M

Gambar 3.16. Input Data Absensi

3.1.2.3. Rancangan Basis Data

Basis data merupakan kumpulan data yang saling berhubungan satu dengan yang lain, tersimpan disimpanan luar komputer kemudian dimanipulasi. Basis data merupakan salah satu komponen yang penting pada sistem informasi bagi para pemakainya. Penerapan basis data dalam sistem informasi disebut sistem basis data. Untuk tahap perancangan basis data, terlebih dahulu diidentifikasi file-file yang dibutuhkan oleh sistem informasi. Langkah-langkah rancangan basis data adalah sebagai berikut :

1. Menentukan kebutuhan file basis data untuk sistem baru yang ditawarkan. File yang dibutuhkan dapat ditentukan dari diagram arus data sistem yang telah dibuat.
2. Menentukan parameter dari file basis data. Setelah file-file yang dibutuhkan telah ditentukan, maka parameter dari file selanjutnya juga dapat ditentukan. Parameter ini dapat meliputi:

- Type file : file master, file transaksi dll.
- Media file : hardisk, disket.
- Organisasi file : file sequential, file acak dll
- Field kunci dari file.

Rancangan basis data dimaksudkan untuk mendefinisikan relasi entitas dan struktur basis data tiap-tiap file yang telah diidentifikasi.

a. Relasi entitas

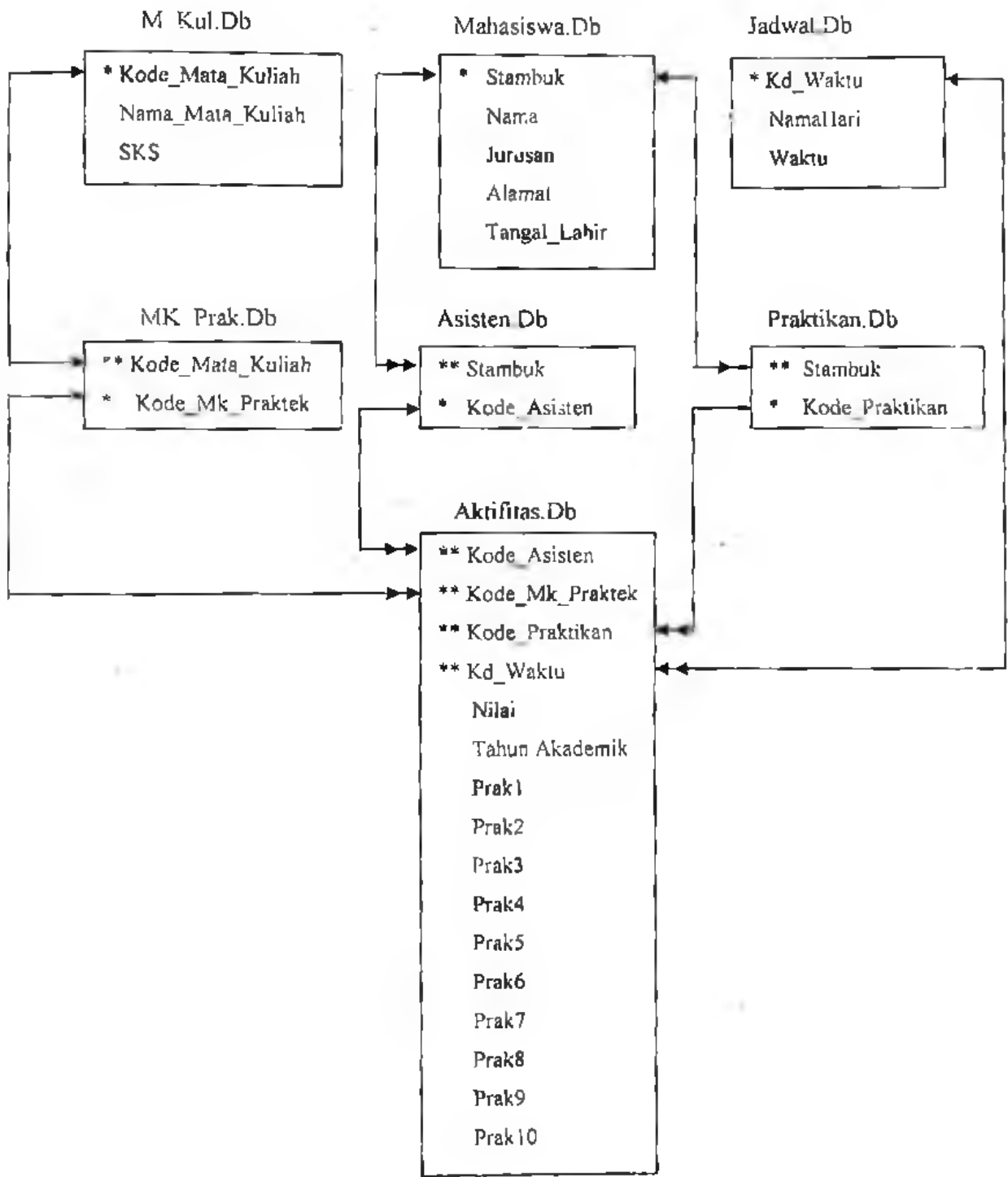
Menurut Ir. Harianto Kristanto (Konsep dan Perancangan Database. 1993 : 2) entity adalah orang, tempat, kejadian atau konsep yang informasinya direkam. Misal : asisten, praktikan, mata kuliah adalah entity dari aktivitas laboratorium. Sedangkan Ir. Fathansyah (Basis Data, 1999 : 64) bahwa entitas adalah individu yang memiliki sesuatu yang nyata yang dapat dibedakan satu dengan yang lainnya. Hubungan entitas (Entitas Relation Ship) merupakan suatu model untuk menjelaskan hubungan antara data dalam basis data berdasarkan suatu modal untuk menjelaskan hubungan antara data dalam basis data berdasarkan suatu persepsi bahwa kenyataan yang ada sebenarnya adalah obyek-obyek tersebut.

Model relasi entitas yang berisi komponen-komponen himpunan entitas dan himpunan relasi yang masing-masing dilengkapi dengan atribut-atribut yang dapat digambarkan dengan lebih sistematis dengan menggunakan diagram relasi entitas.



Entity yang terdapat pada perancangan sistem basis data laboratorium komputasi jurusan matematika ada tiga yaitu mahasiswa, m_kul dan jadwal. Adapun field-fieldnya yaitu mahasiswa { Stambuk, Nama, Jurusan, Alamat, Tanggal_Lahir }, m_kul { Kode_Mata_Kuliah, Nama_Mata_Kuliah, Sks } dan jadwal { Kd_Waktu, NamaHari, Waktu }.

Sedangkan file asisten, file praktikan, file mk_praktek, dan file aktifitas adalah file transaksi dengan field-field sebagai berikut : asisten {Stambuk, Kode_Asisten }, praktikan { Stambuk, Kode_Praktikan }, mk_praktek { Kode_Mk_Praktek, Kode_Mata_Kuliah } dan aktifitas { Kode_Mk_Praktek, Kode_Praktikan, Kode_asisten, Kd_Waktu, Nilai, Tahun_Akademik, Prak1, Prak2, Prak3, Prak4, Prak5, Prak6, Prak7, Prak8, Prak9, Prak10 }. Ketujuh file-file tersebut kemudian direlasikan dengan model Entity Relation Ship sebagai berikut :



Gambar 3.17. Diagram Relasi Entitas (ERD)

b. **Struktur Basis Data**

Struktur basis data yang dimaksud terdiri atas penjelasan terperinci mengenai setiap field dalam satu data record. Dari perinciannya meliputi :

1. Field Nama untuk mengidentifikasi field.
2. Type untuk menyatakan tipe data yang dapat disimpan dalam field tersebut.
3. Size yaitu ukuran atau daya tampung suatu field
4. Key yaitu kunci suatu file yang akan berguna dalam relasi antar file.

Struktur basis data pada perancangan sistem basis data Laboratorium Komputasi Jurusan Matematika FMIPA Unhas sebagai berikut :

1. *Tabel 3.1 : Jadwal.Db*

No	Field Name	Type	Size	Key
1.	Kd_waktu	A	4	*
2.	Nama_Hari	A	7	
3.	Waktu	A	13	

2. *Tabel 3.2 : Asisten.Db*

No	Field Name	Type	Size	Key
1.	Kd_Asisten	A	8	*
2.	Stambuk	A	9	*

3. *Tabel 3.3 : M Kul.Db*

No	Field Name	Type	Size	Key
1.	Kode_mata_kuliah	A	6	*
2.	Nama_mata_kuliah	A	25	
3.	SKS	N		

4. *Tabel 3.4 : Mahasiswa.Db*

No	Field Name	Type	Size	Key
1.	Stambuk	A	9	*
2.	Nama	A	25	
3.	Jurusan	A	30	
4.	Alamat	A	30	
5.	Tanggal_lahir	D		

5. *Tabel 3.5 : Mk Praktek.Db*

No	Field Name	Type	Size	Key
1.	Kode_Mk_Praktek	A	3	*
2.	Kode_mata_kuliah	A	3	*

6. Tabel 3.6 : *Praktikan.Db*

No	Field Name	Type	Size	Key
1.	Kode_praktikan	A	8	*
2.	Stambuk	A	9	*

7. Tabel 3.7 : *Aktifitas.Db*

No	Field Name	Type	Size	Key
1.	Kode_Mk_Praktek	A	3	*
2.	Kode_Praktikan	A	8	*
3.	Kode_Asisten	A	8	*
4.	Kd_Waktu	A	4	*
5.	Prak1	A	1	
6.	Prak2	A	1	
7.	Prak3	A	1	
8.	Prak4	A	1	
9.	Prak5	A	1	
10.	Prak6	A	1	
11.	Prak7	A	1	
12.	Prak8	A	1	
13.	Prak9	A	1	
14.	Prak10	A	1	
15.	Tahun_Akademik	A	16	
16.	Nilai	N		

3.2. Rancangan Teknologi

Teknologi digunakan untuk menerima input, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirim keluaran dan membantu pengendalian sistem secara keseluruhan. Teknologi terdiri dari tiga bagian utama, yaitu perangkat keras, perangkat lunak dan Teknisi.

Untuk tahap perancangan teknologi, harus dilakukan identifikasi jenis dan jumlah tersebut adalah sebagai berikut :

1. Menentukan jenis teknologi untuk sistem baru. Untuk teknologi Perangkat Keras (*Hardware*) terlebih dahulu ditentukan jenis peralatan yang akan digunakan pada masing-masing proses dalam sistem informasi. Untuk Perangkat Lunak (*Software*), analisis sistem ditentukan terlebih dahulu jenis dan software sistem.
2. Menentukan jumlah teknologi. Untuk perangkat keras, jumlah teknologi dapat ditentukan dari waktu yang tersedia dan waktu standar masing-masing sistem operasi yang akan digunakan dalam teknologi ini. Untuk perangkat lunak, jumlah dari teknologi ini dapat ditentukan dari jumlah hardware yang dibutuhkan.

Dalam penyelesaian perancangan sistem basis data laboratorium komputasi jurusan Matematika, kami dapat uraikan spesifikasi hardware dan software sebagai berikut :

1. Spesifikasi Hardware :

- Komputer minimal prosesor Pentium 133 disarankan yang lebih tinggi.
- RAM minimal 16 MB untuk pengolahan data yng berskala kecil, disarankan lebih tinggi untuk data yang berskala besar.
- Media penyimpanan harddisk

2. Spesifikasi Software :

- Sistem Operasi minimal Windows 95 atau yang lebih tinggi.

3.3. Pengendalian Aplikasi Sistem

Pengendalian aplikasi sistem merupakan pengendalian yang diterapkan selama proses pengolahan data berlangsung. Pengendalian aplikasi ini dapat dikategorikan kedalam :

1. Pengendalian Masukan

Pengendalian masukan mempunyai tujuan untuk menyakinkan bahwa data transaksi yang valid telah lengkap, terkumpul semuanya dan bebas dari kesalahan sebelum dilakukan proses pengolahannya.

2. Pengendalian Pengolahan

Tujuan dari pengendalian pengolahan adalah untuk mencegah kesalahan-kesalahan yang terjadi selama proses pengolahan data yang dilakukan setelah data dimasukkan kedalam komputer.

3. Pengendalian Keluaran

Keluaran yang mempunyai produk dari pengolahan data disajikan dalam dua bentuk utama, yaitu dalam bentuk *hard copy* dan dalam bentuk *soft copy*. Dalam bentuk *hard copy* yang paling banyak dilakukan berbentuk laporan yang dicetak menggunakan alat cetak (*printer*) dan dalam bentuk *soft copy* yang paling umum adalah berbentuk tampilan monitor.

3.4. Implementasi Sistem

Implementasi sistem adalah prosedur yang dilakukan untuk menyelesaikan rancangan sistem yang ada didalam dokumen rancangan sistem yang disetujui dan menguji, menginstall, dan memulai menggunakan sistem baru atau sistem yang telah diperbaiki. Untuk menjalankan aplikasi sistem ini, dibutuhkan spesifikasi hardware dan software yang telah dijelaskan sebelumnya.

BAB IV

PENGUJIAN DAN HASIL PENGUJIAN SISTEM

4.1 Pengujian Sistem

4.1.1 Metode Pengujian

Metode pengujian yang dilakukan untuk mengetahui tingkat efektifitas dan efisiensi dari perangkat lunak (software) yang digunakan. Selain itu untuk memberikan kesempatan kepada pemakai sistem untuk mengoperasikan dan melakukan pengecekan terhadap laporan yang dihasilkan melalui perangkat lunak (software) ini. Hal ini dimaksudkan untuk mengantisipasi kesalahan serta kemungkinan penambahan item-item yang diinginkan oleh pihak pengelola laboratorium komputasi jurusan Matematika .

Sistem dapat berjalan dengan lancar, apabila pengisian data dilakukan secara aktif dan kontinyu agar informasi yang dihasilkan benar-benar akurat. Oleh sebab itu, untuk mengantisipasi kesalahan-kesalahan sistem diberikan kepada pemakai untuk memberikan saran atau kritikan terhadap sistem yang di uji, apakah sistem yang di buat telah sesuai dengan kebutuhan atau penambahan hal-hal lain yang dapat menunjang aplikasi tersebut.



4.1.2 Populasi dan Sampel

4.1.2.1 Populasi

Populasi dari penelitian ini adalah pihak yang terlibat dalam aktifitas laboratorium komputasi diantaranya pihak pengelola, asisten dan mantan asisten laboratorium komputasi jurusan Matematika.

4.1.2.2 Sampel

Jumlah sampel yang diambil dari populasi sebanyak 10 orang yang akan dijadikan responden, yang diambil dari pihak pengelola asisten dan mantan asisten laboratorium komputasi jurusan Matematika.

4.1.3 Teknik Pengumpulan Data

Dalam penyelesaian skripsi ini, pengumpulan data dilakukan dengan menggunakan dua metode yaitu :

1. *Library Research*, yaitu penelitian kepustakaan yang dimaksudkan untuk mengumpulkan data teori permasalahan yang di bahas.
2. *Field Research*, yaitu penelitian lapangan yang dilakukan secara observasi, wawancara, dan penyebaran kuisisioner. Observasi dilakukan dengan cara pengamatan langsung pelaksanaan kegiatan aktifitas laboratorium, wawancara dilakukan terhadap setiap orang yang terlibat langsung dalam proses pengolahan data, serta penyebaran kuisisioner yang berisi pertanyaan-pertanyaan yang berkaitan dengan permasalahan dalam skripsi ini kepada seluruh responden.

4.1.4 Teknik Pengolahan Data dan Analisa Data

Teknik pengolahan data dan analisis data yang digunakan dalam penelitian ini adalah dengan statistik deskriptif dengan prosentase menggunakan program SPSS for windows Release 6, dengan data kuantitatif diperoleh dari hasil penyebaran kuisisioner dengan jumlah responden sebanyak 10 orang.

4.2 Hasil Pengujian Sistem

Secara operasional penggunaan komputerasi pengolahan basis data laboratorium komputasi jurusan Matematika yang dibahas pada penelitian ini, memberikan berbagai kemudahan dan manfaat. Karena akan menyebabkan terjadinya efektifitas dan efisiensi penggunaan waktu, tenaga serta meningkatkan sumber daya pengelola.

Hasil dari penyebaran kuisisioner kepada 10 responden dengan jumlah pertanyaan 8 buah dapat disajikan sebagai berikut :

Tabel 4.1 Data belum diolah

No. Sampel (n)	Pertanyaan							
	1	2	3	4	5	6	7	8
1	4	4	5	5	5	5	6	6
2	2	4	4	6	5	6	6	6
3	3	4	5	6	6	6	6	6
4	2	2	4	5	5	6	6	6
5	2	4	3	4	4	6	6	6
6	3	4	4	5	5	6	6	6

7	2	4	4	5	5	6	6	6
8	4	4	4	4	5	6	6	6
9	3	4	5	5	5	6	6	6
10	3	4	4	5	5	6	6	6

Sumber : Hasil kuisioner pada laboratorium komputasi jurusan Matematika

Efektifitas dan efisiensi sistem komputerisasi pengolahan data di laboratorium komputasi jurusan Matematika tersebut dapat dilihat dari hasil pengolahan data sebagai berikut :

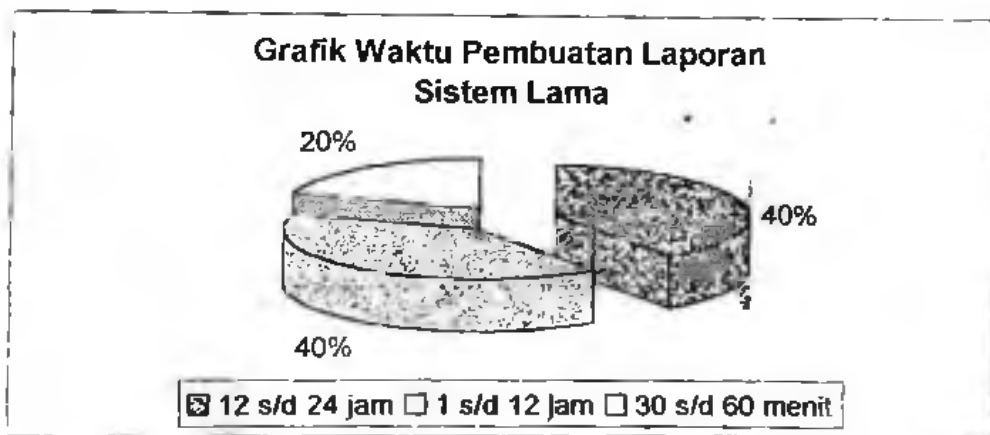
a. Waktu Pembuatan Laporan

Tabel. 4.2 Waktu Pembuatan Laporan

Waktu Pembuatan Laporan	Sistem Lama		Sistem Baru	
	Frek.	%	Frek.	%
12 jam s/d 24 jam	4	40		
1 jam s/d 12 jam	4	40		
30 menit s/d 60 menit	2	20	1	10
10 menit s/d 30 menit			8	80
kurang dari 10 menit			1	10
Total	10	100	10	100

Dari tabel 4.2 diatas menunjukkan bahwa dengan menggunakan sistem lama dalam pembuatan laporan 40 % responden menyatakan bahwa diperlukan waktu 12 – 24 jam, 40 % responden menyatakan diperlukan waktu 1 – 12 jam dan 20 % responden menyatakan 30 – 60 menit. Sedangkan dengan menggunakan sistem yang baru dalam pembuatan laporan 10 % responden

menyatakan diperlukan waktu 30 – 60 menit, 80 % responden menyatakan diperlukan waktu 10 – 30 menit dan 10 % responden menyatakan diperlukan waktu kurang dari 10 menit. Jika disajikan dalam bentuk grafik lingkaran 3-D dapat dilihat sebagai berikut :



Dari hasil analisa grafik diatas menunjukkan bahwa 80 % responden menyatakan dengan sistem lama waktu yang dibutuhkan dalam pembuatan laporan antara 1 – 24 jam dan selebihnya 20 % hanya membutuhkan waktu 30 –

60 menit. Sedangkan dengan menggunakan sistem yang baru 90 % responden menyatakan membutuhkan waktu kurang dari 30 menit dan sisanya 10 % butuh waktu 30 – 60 menit. Ini menunjukkan bahwa tingkat efektifitas dan efisiensi sistem yang baru lebih tinggi dibandingkan sistem yang lama.

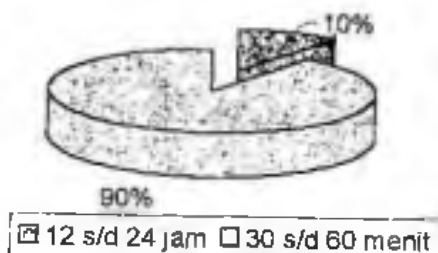
b. Waktu Pencarian Data

Tabel. 4.3 Waktu Pencarian Data

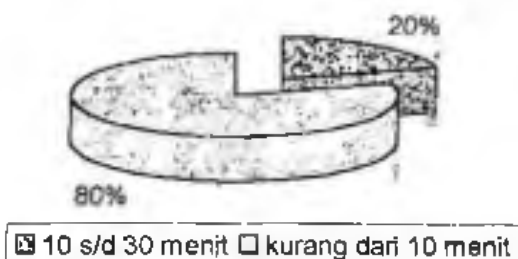
Waktu Pencarian Data	Sistem Lama		Sistem Baru	
	Frek.	%	Frek.	%
12 jam s/d 24 jam	1	10		
30 menit s/d 60 menit	9	90		
10 menit s/d 30 menit			2	20
kurang dari 10 menit			8	80
Total	10	100	10	100

Dari tabel 4.3 diatas menunjukkan bahwa dengan menggunakan sistem lama dalam pencarian data 10 % responden menyatakan bahwa diperlukan waktu 12 – 24 jam dan 90 % responden menyatakan diperlukan waktu 30 – 60 menit. Sedangkan dengan menggunakan sistem yang baru 20 % responden menyatakan diperlukan waktu 10 – 30 menit dan 80 % responden menyatakan diperlukan waktu kurang dari 10 menit. Jika disajikan dalam bentuk grafik lingkaran 3 D dapat dilihat sebagai berikut :

Grafik Waktu Pencarian Data Sistem Lama



Grafik Waktu Pencarian Data Sistem Baru



Dari hasil grafik di atas menunjukkan bahwa 100 % responden menyatakan dengan sistem lama waktu yang dibutuhkan dalam pencarian data lebih dari 30 menit sedangkan dengan sistem yang baru 80 % responden menyatakan dibutuhkan waktu kurang dari 10 menit dan sisanya 20 % butuh waktu 10 – 30 menit. Tingginya jumlah prosentase yang menyatakan sistem baru hanya butuh waktu kurang dari 10 menit dalam pencarian data menandakan sistem tersebut lebih efektif dan efisien dari sistem sebelumnya.



c. Jumlah Tenaga Kerja Dalam Pembuatan laporan

Tabel. 4.4 Jumlah Tenaga Kerja Dalam Pembuatan Laporan

Jumlah Tenaga Kerja Dalam Pembuatan Laporan	Sistem Lama		Sistem Baru	
	Frek.	%	Frek.	%
4 orang	1	10		
3 orang	6	60		
2 orang	3	30		
1 orang			10	100
Total	10	100	10	100

Dari tabel 4.4 diatas menunjukkan bahwa jumlah tenaga kerja dalam pembuatan laporan dengan menggunakan sistem lama, 10 % responden menyatakan membutuhkan tenaga 4 orang, 60 % responden menyatakan membutuhkan tenaga 3 orang dan 30 % responden menyatakan membutuhkan tenaga 2 orang. Dengan sistem baru 100 % responden menyatakan hanya membutuhkan 1 orang. Jika disajikan dalam bentuk grafik lingkaran 3 D dapat dilihat sebagai berikut :





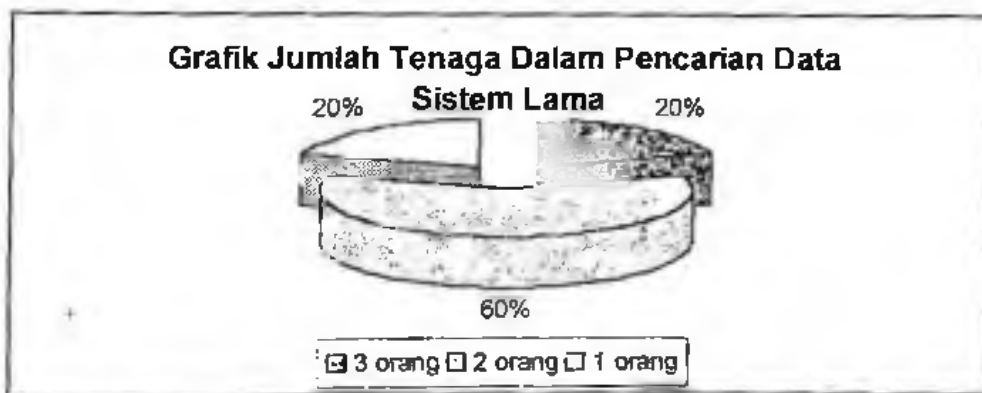
Dari hasil grafik di atas menunjukkan bahwa jumlah tenaga kerja yang dibutuhkan dalam pembuatan laporan dengan sistem yang lama 70 % responden menyatakan membutuhkan tenaga 3 - 4 orang dan 30 % responden hanya butuh tenaga 2 orang. Dengan menggunakan sistem yang baru 100 % menyatakan hanya butuh 1 orang tenaga dalam pengerjaan pembuatan laporan. Ini berarti sistem baru lebih efektif dan efisien dibandingkan sistem yang lama.

d. Jumlah Tenaga Kerja Dalam Pencarian Data

Tabel. 4.5 Jumlah Tenaga Kerja Dalam Pencarian Data

Jumlah Tenaga Kerja Dalam Pencarian Data	Sistem Lama		Sistem Baru	
	Frek.	%	Frek.	%
3 orang	2	20		
2 orang	6	60		
1 orang	2	20	10	100
Total	10	100	10	100

Dari tabel 4.5 diatas menunjukkan bahwa jumlah tenaga kerja dalam pencarian data dengan menggunakan sistem lama, 20 % responden menyatakan membutuhkan tenaga 3 orang, 60 % responden menyatakan membutuhkan tenaga 2 orang dan 20 % responden menyatakan membutuhkan tenaga 1 orang. Dengan sistem baru 100 % responden menyatakan hanya membutuhkan 1 orang. Jika disajikan dalam bentuk grafik lingkaran 3 D dapat dilihat sebagai berikut :



Dari hasil grafik di atas menunjukkan bahwa dengan menggunakan sistem lama 80 % responden menyatakan butuh tenaga 2 – 3 orang dan

selebihnya mengatakan 1 orang. Sedangkan sistem baru 100 % menyatakan hanya membutuhkan tenaga 1 orang. Ini menunjukkan bahwa tingkat efektif dan efisien dalam hal penggunaan tenaga kerja sistem yang baru lebih tinggi dibandingkan sistem sebelumnya.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Setelah melakukan penelitian tentang sistem basis data pada laboratorium komputasi jurusan Matematika FMIPA Unhas, beberapa hal yang dapat disimpulkan antara lain :

1. Pola kebutuhan atau bentuk sistem basis data yang cocok diterapkan pada laboratorium komputasi jurusan Matematika FMIPA Unhas adalah sistem yang sifatnya terus-menerus.
2. Program aplikasi sistem basis data yang telah dirancang dan diuji coba, dapat mengatasi masalah redundansi data
3. Model rancangan program aplikasi yang ditawarkan mempunyai tingkat efektifitas dan efisiensi sebesar 90 % sehingga informasi yang dibutuhkan oleh berbagai pihak dapat dengan mudah diperoleh secara cepat, tepat dan akurat.

5.2 Saran

Adapun beberapa saran terhadap hasil penelitian dan penulisan skripsi ini yaitu sebagai berikut :

1. Untuk meningkatkan kemampuan pengolahan dan penyajian informasi, maka disarankan menggunakan spesifikasi atau konfigurasi perangkat keras (*hardware*) yang terbaru.

2. Untuk menjamin keamanan data, disarankan agar dilakukan backup data secara berkala, untuk menghindari terjadinya kerusakan pada harddisk atau kehilangan data yang diakibatkan oleh virus komputer.
3. Disarankan kepada mahasiswa lain untuk dapat mengembangkan program aplikasi ini khususnya mengenai peralatan yang ada di laboratorium, sistem persuratan dalam laboratorium dan lain-lain yang masih dianggap perlu demi menyempurnakan program ini.
4. Dengan semakin meningkatnya kualitas dan kuantitas data yang diolah, maka disarankan sistem yang telah diterapkan dapat dikembangkan dalam suatu jaringan komputer (Local Area Network).

DAFTAR PUSTAKA

- Fathansyah, Ir. 1999. *Basis Data*. CV Informatika, Bandung.
- George M.Scott. 1995. *Prinsip-Prinsip Sistem Informasi Manajemen*. PT. Raja Grafindo Persada, Jakarta.
- Henry F. Kartz & Abraham Silberschatz. 1996. *Database System Concept*. Mc. Grow. Hill Book Computer.
- Iuge Martina, Ir. 2001. *Seri Aplikasi Pemrograman Database Menggunakan Delphi*. PT. Elex Media Komputindo, Jakarta.
- Jogiyanto, H.M. 1989. *Pengenalan Komputer*. ANDI Offset, Yogyakarta.
- Jayanto. 1999. *Membuat Aplikasi Database Dengan Delphi*. PT. Elex Media Komputindo, Jakarta.
- Kadir, Abdul. 2000. *Konsep & Tuntunan Praktis Basis Data*. ANDI Offset, Yogyakarta.
- Kristanto, Harianto, Ir. 1994. *Konsep dan Perancangan Database*. ANDI Offset, Yogyakarta.
- Sudjana, M.A., M.Sc., DR. Prof. 1992. *Metode Statistika Edisi Ke 5*. Tarsito, Bandung.
- Waljiyanto. 2000. *Sistem Basis Data, Analisis dan Pemodelan Data*. J & J Learning, Yogyakarta.

LAMPIRAN - LAMPIRAN



**KUISIONER ANALISIS DAN PERANCANGAN SISTEM BASIS DATA
LABORATORIUM KOMPUTASI JURUSAN MATEMATIKA FMIPA UNHAS**

No.Responden :

Beri tanda silang pada kotak jawaban yang anda rasa cocok !

A. Sebelum Penerapan

➤ **Tingkat Kecepatan**

1. Berapa lama rata-rata waktu yang diperlukan untuk proses pembuatan Laporan selama penggunaan sistem yang lama ?

- lebih dari 24 jam (1)
- 12 s/d 24 jam (2)
- 1 s/d 12 jam (3)
- 30 s/d 60 menit (4)
- 10 s/d 30 menit (5)
- kurang dari 10 menit (6)
- (7)

2. Berapa lama rata-rata waktu yang diperlukan untuk proses pencarian data selama penggunaan sistem yang lama ?

- lebih dari 24 jam (1)
- 12 s/d 24 jam (2)
- 1 s/d 12 jam (3)
- 30 s/d 60 menit (4)
- 10 s/d 30 menit (5)
- kurang dari 10 menit (6)
- (7)

➤ **Tingkat Penggunaan Tenaga Kerja**

3. Berapa jumlah tenaga kerja yang diperlukan untuk menyelesaikan proses pembuatan Laporan ?

- lebih dari 5 orang (1)
- 5 orang (2)
- 4 orang (3)
- 3 orang (4)
- 2 orang (5)
- 1 orang (6)
- (7)

4. Berapa jumlah tenaga kerja yang diperlukan untuk menyelesaikan proses pencarian data ?

- lebih dari 5 orang (1)
- 5 orang (2)
- 4 orang (3)
- 3 orang (4)
- 2 orang (5)
- 1 orang (6)
- (7)

**KUISIONER ANALISIS DAN PERANCANGAN SISTEM BASIS DATA
LABORATORIUM KOMPUTASI JURUSAN MATEMATIKA FMIPA UNHAS**

B. Setelah Penerapan

➤ *Tingkat Kecepatan*

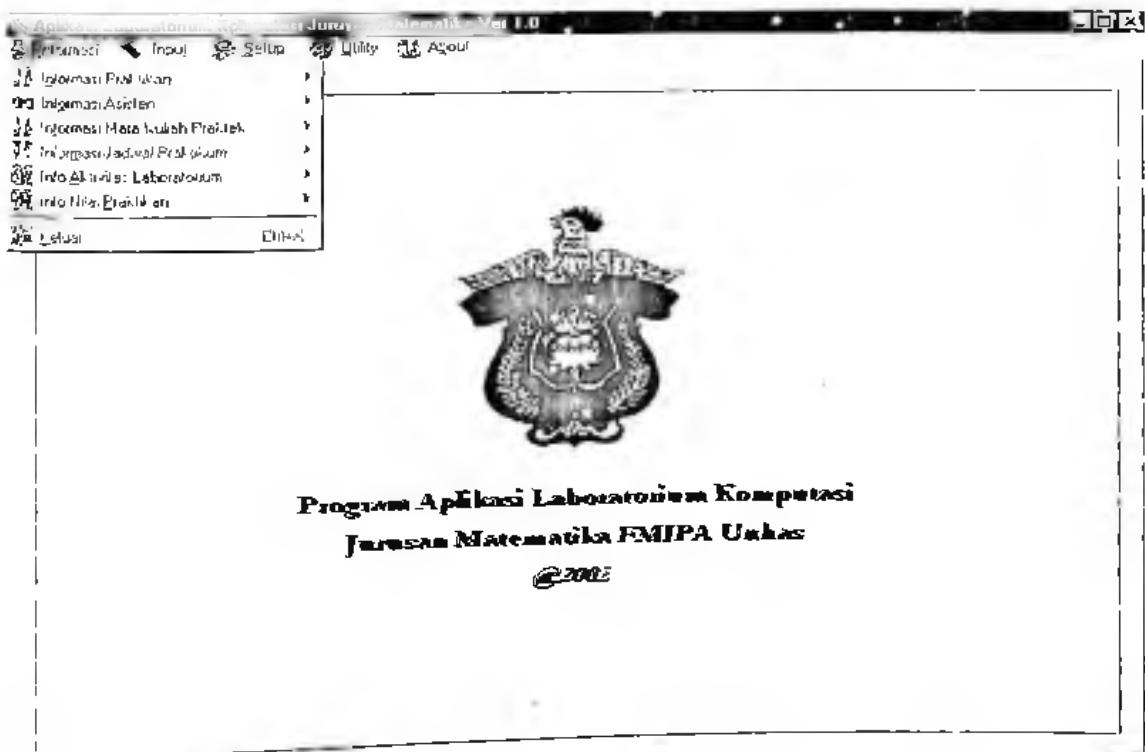
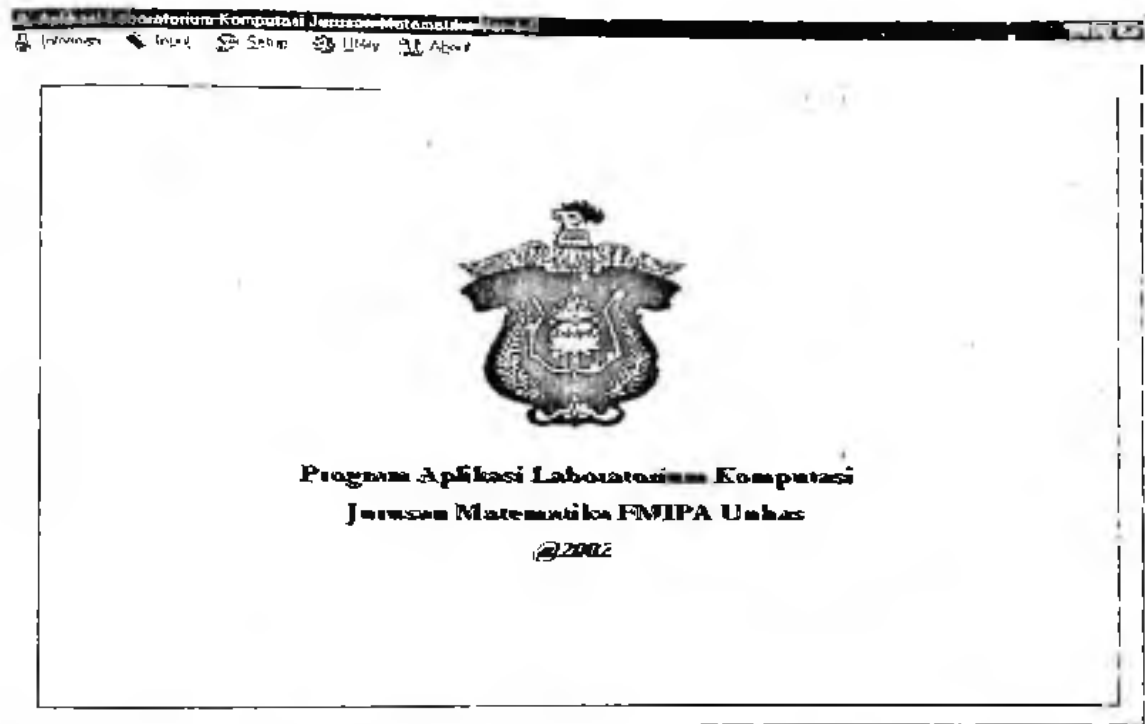
5. Berapa lama rata-rata waktu yang diperlukan untuk proses pembuatan Laporan selama penggunaan sistem yang baru ?
- lebih dari 24 jam (1)
 - 12 s/d 24 jam (2)
 - 1 s/d 12 jam (3)
 - 30 s/d 60 menit (4)
 - 10 s/d 30 menit (5)
 - kurang dari 10 menit (6)
 - (7)
6. Berapa lama rata-rata waktu yang diperlukan untuk proses pencarian data selama penggunaan sistem yang baru ?
- lebih dari 24 jam (1)
 - 12 s/d 24 jam (2)
 - 1 s/d 12 jam (3)
 - 30 s/d 60 menit (4)
 - 10 s/d 30 menit (5)
 - kurang dari 10 menit (6)
 - (7)

➤ *Tingkat Penggunaan Tenaga Kerja*

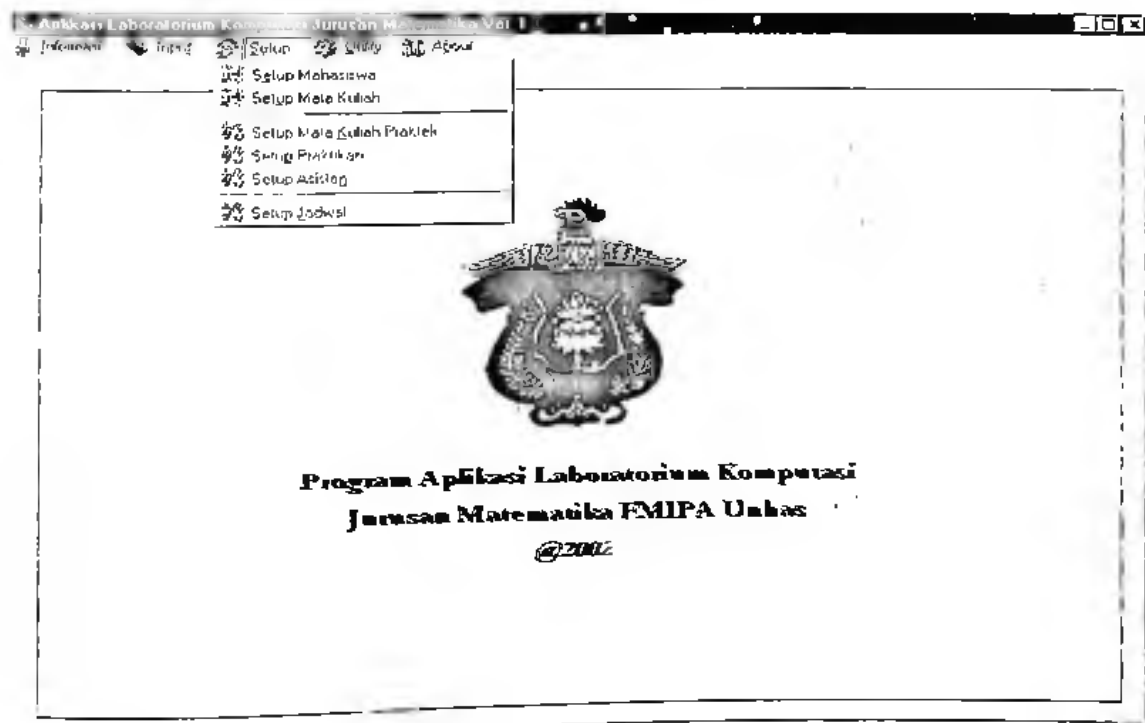
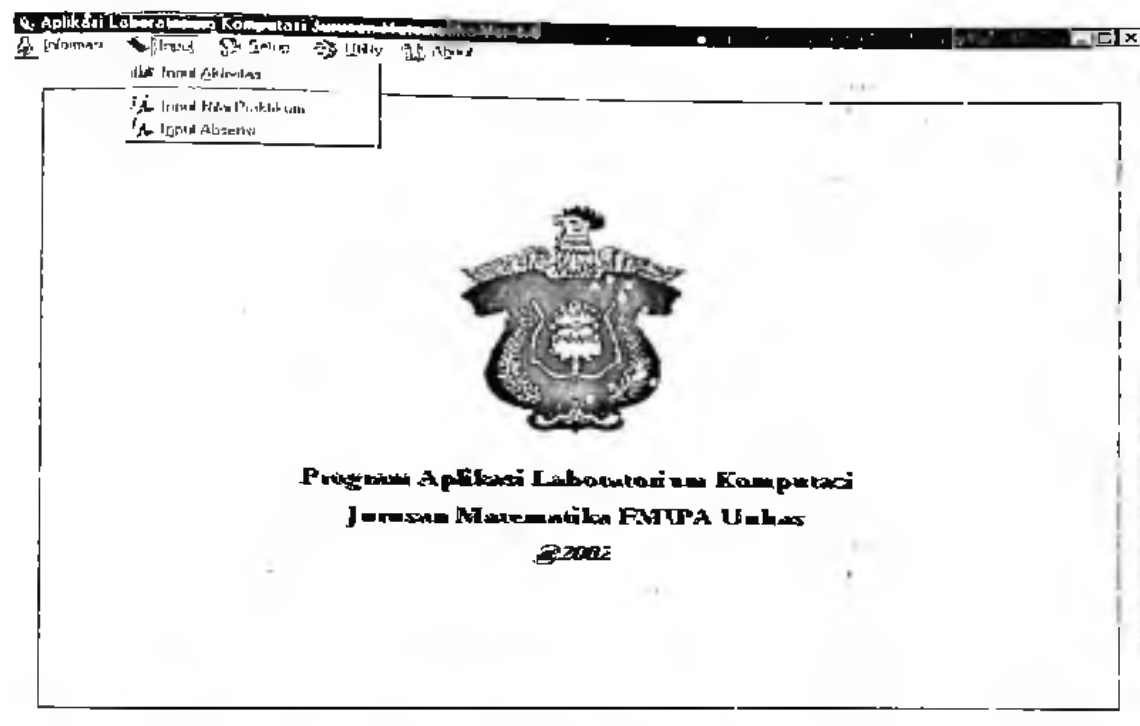
7. Berapa jumlah tenaga kerja yang diperlukan untuk menyelesaikan proses pembuatan Laporan ?
- lebih dari 5 orang (1)
 - 5 orang (2)
 - 4 orang (3)
 - 3 orang (4)
 - 2 orang (5)
 - 1 orang (6)
 - (7)
8. Berapa jumlah tenaga kerja yang diperlukan untuk menyelesaikan proses pencarian data ?
- lebih dari 5 orang (1)
 - 5 orang (2)
 - 4 orang (3)
 - 3 orang (4)
 - 2 orang (5)
 - 1 orang (6)
 - (7)

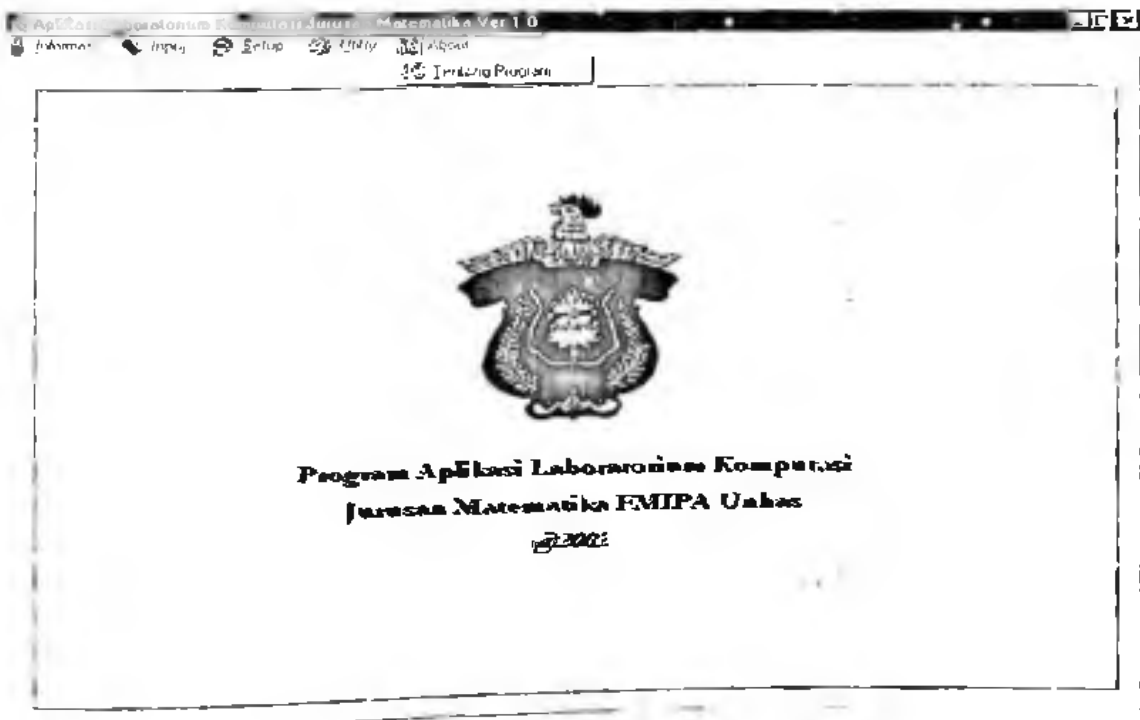
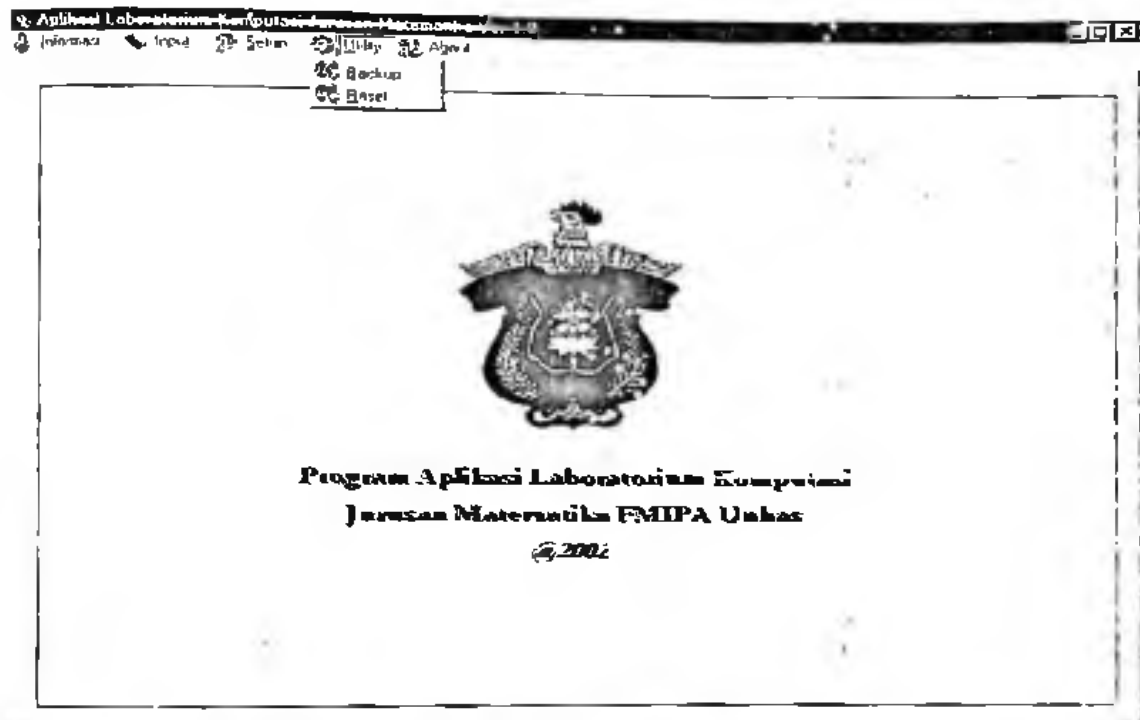
Makassar, - 2002
Responden,

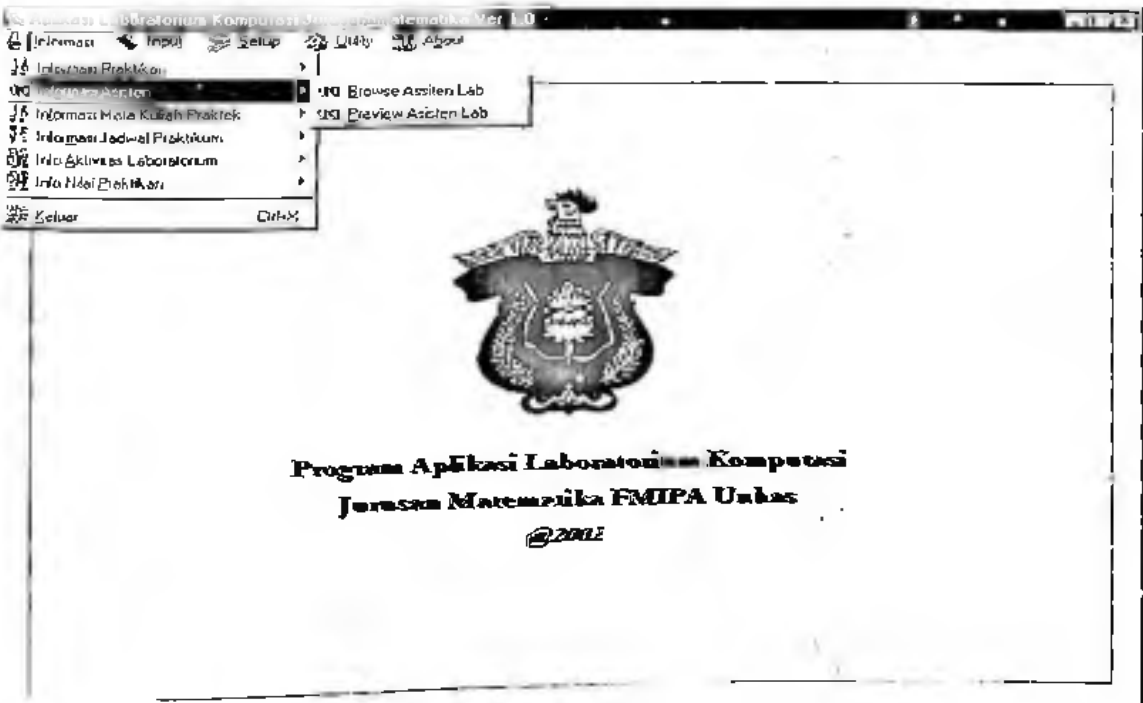
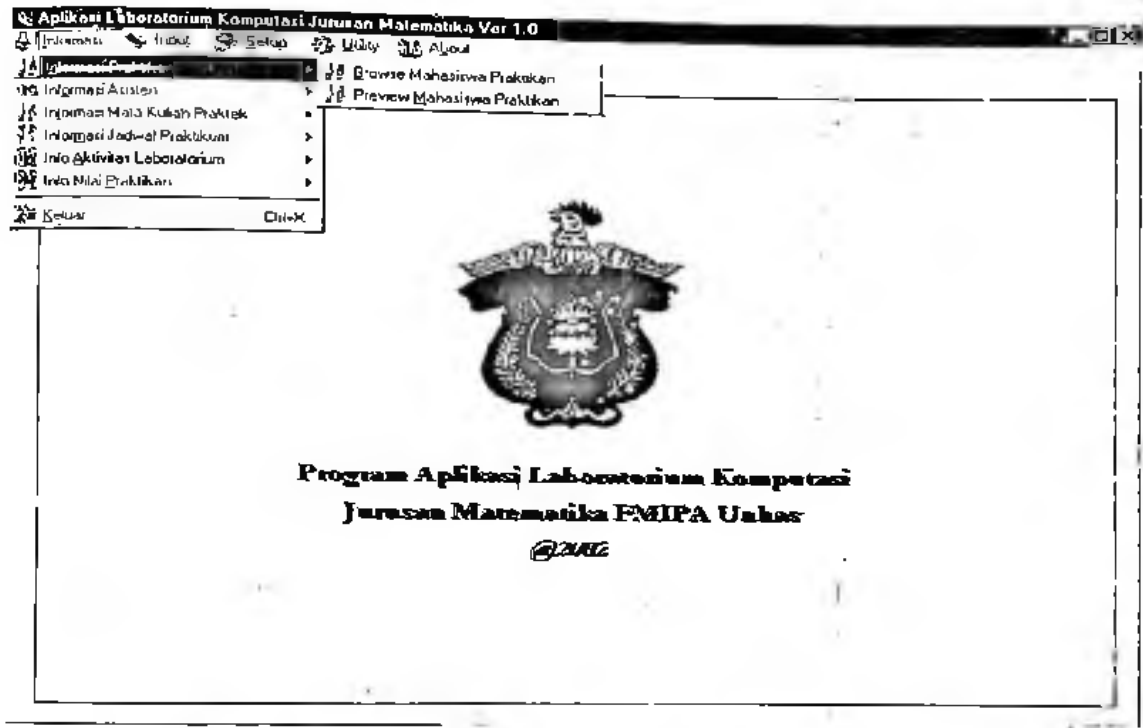
Rancangan Menu Program

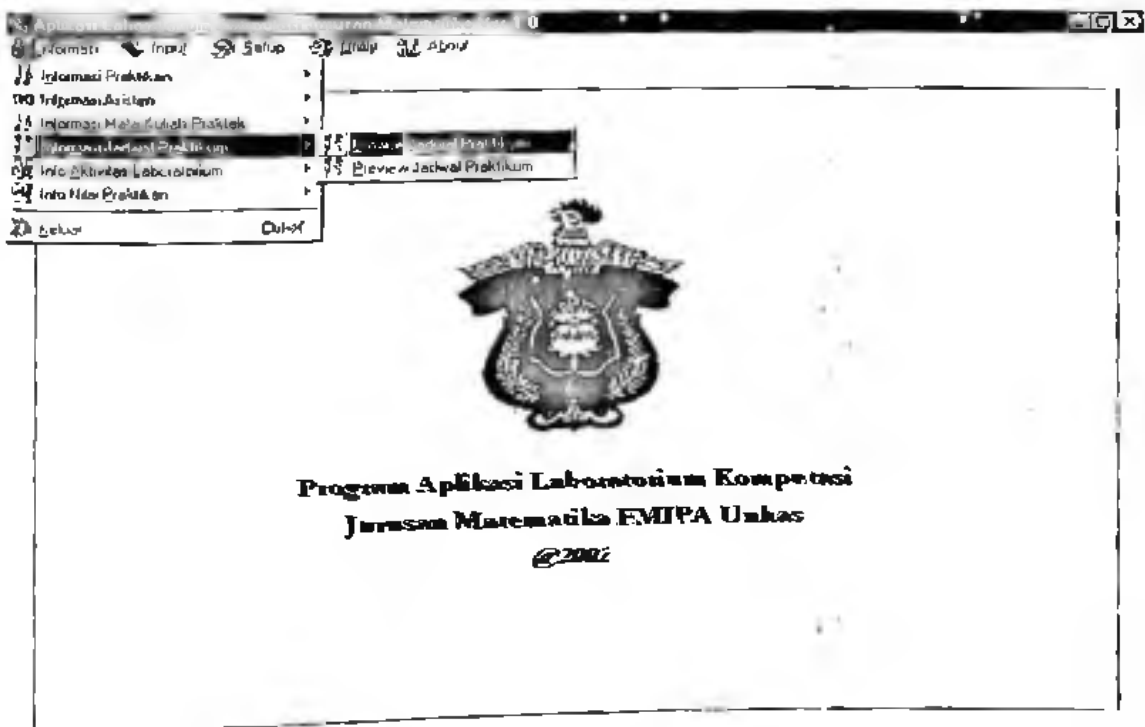
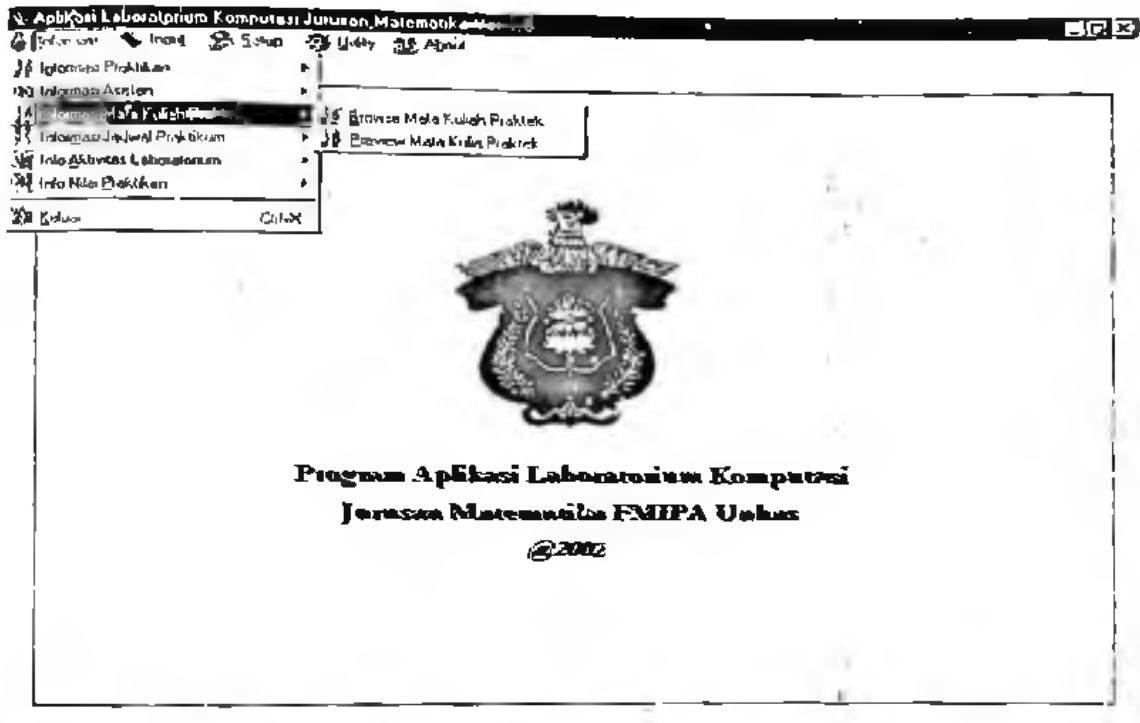
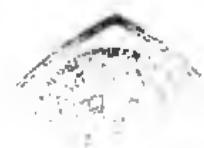


Rancangan Menu Program

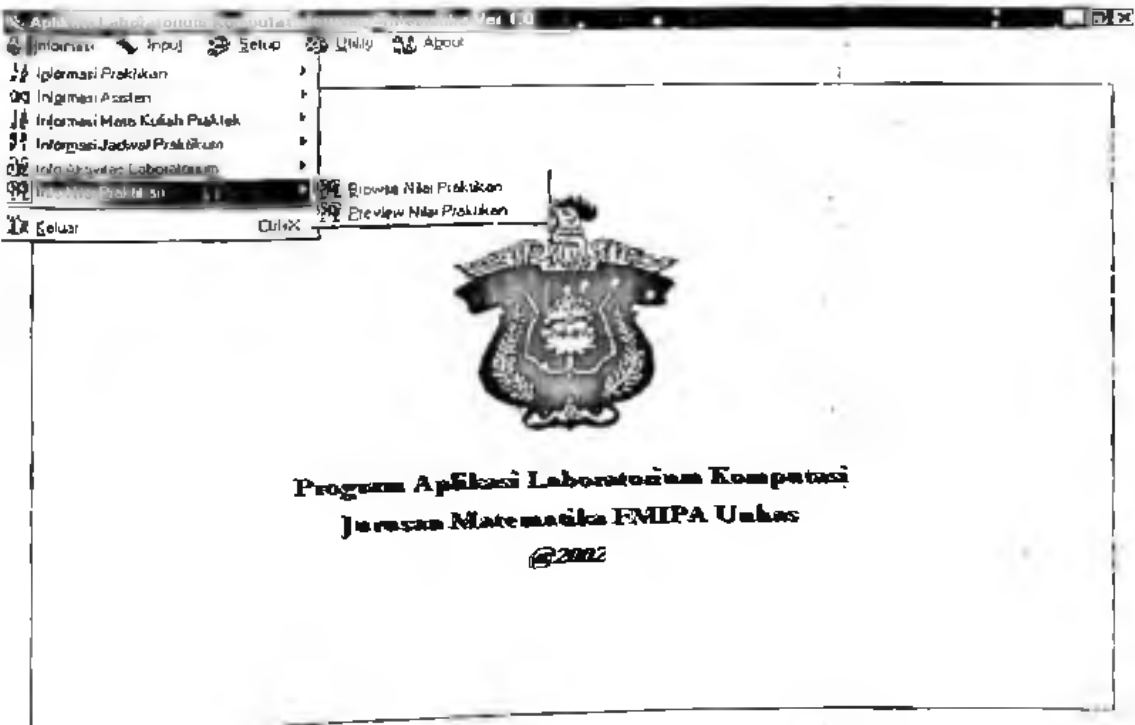
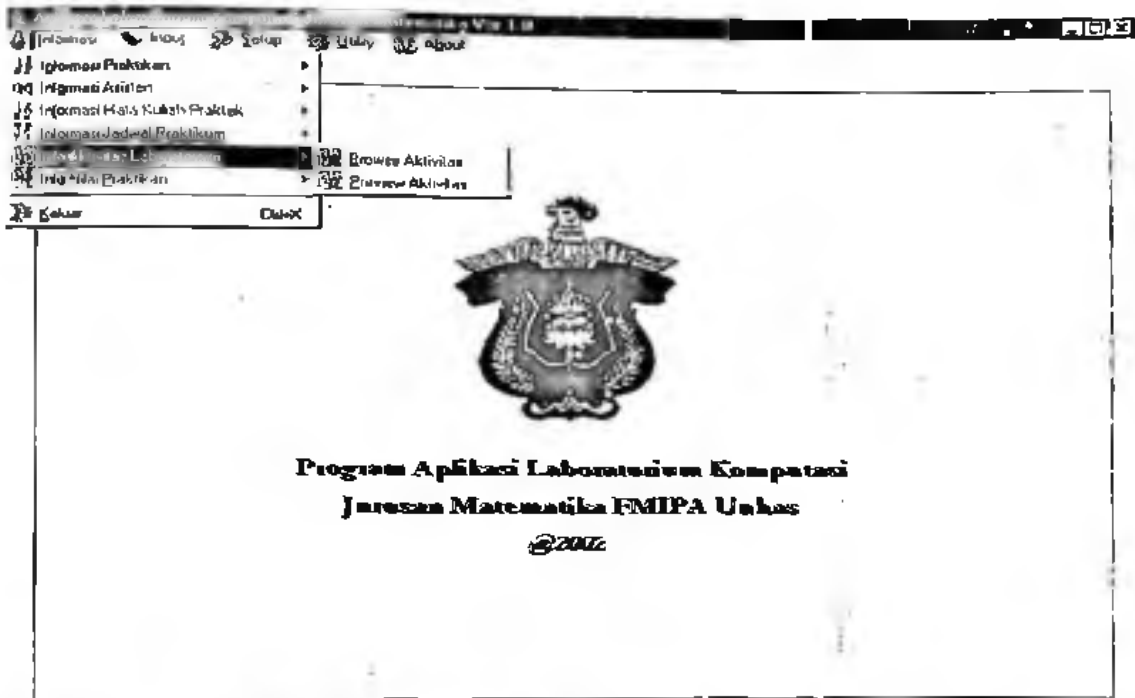




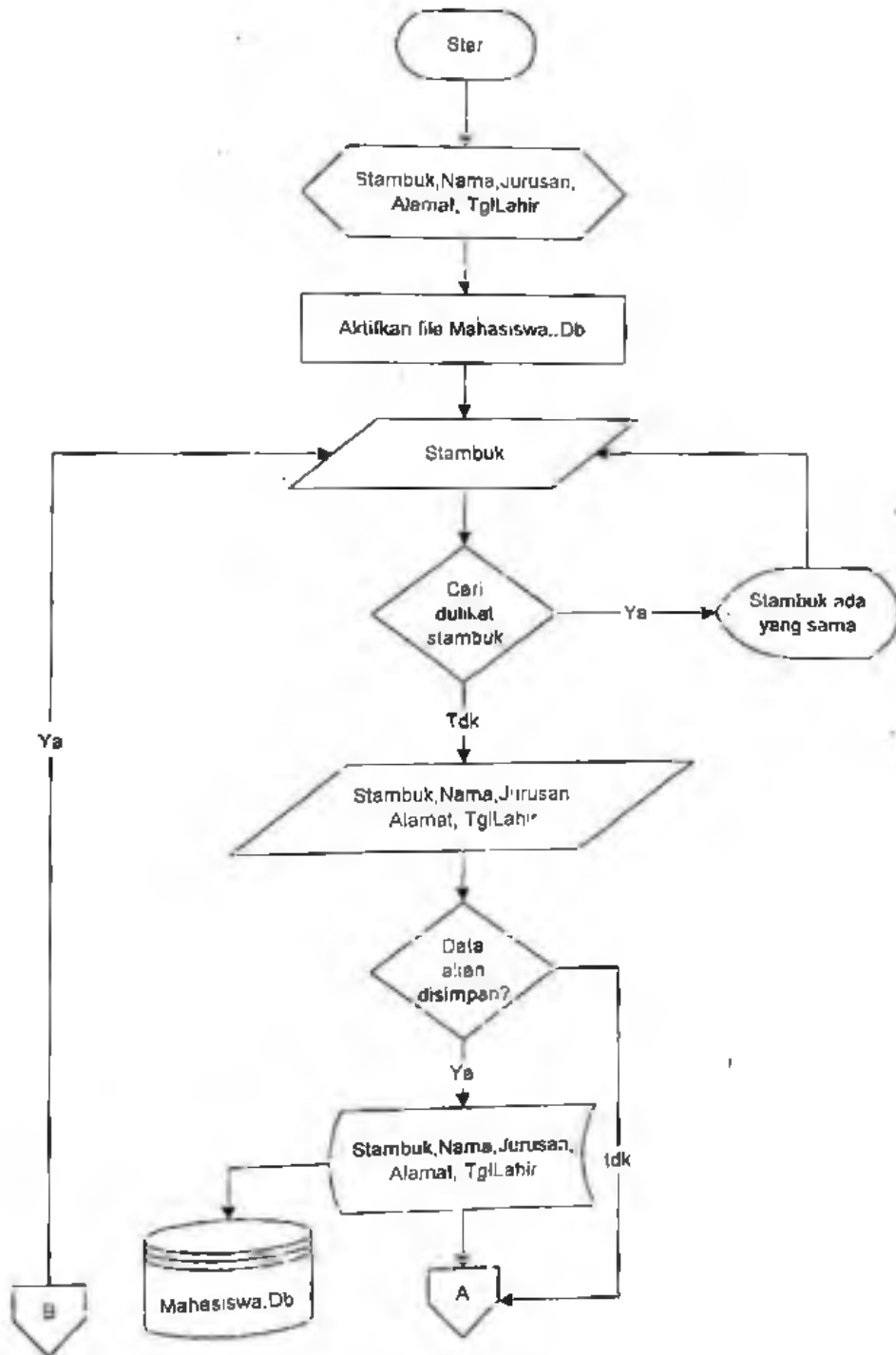




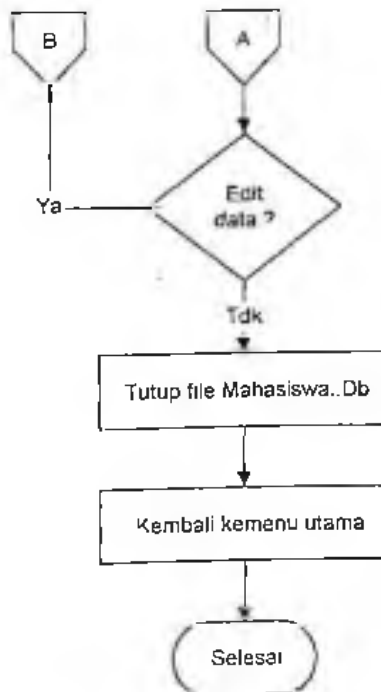
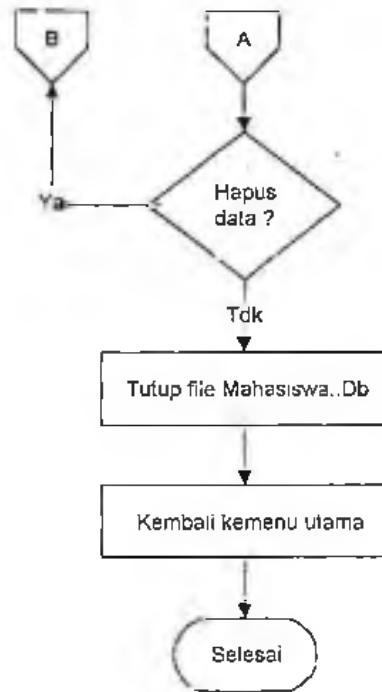
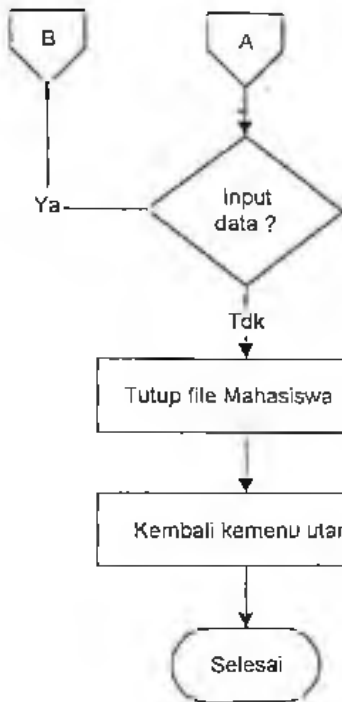
Rancangan Menu Program



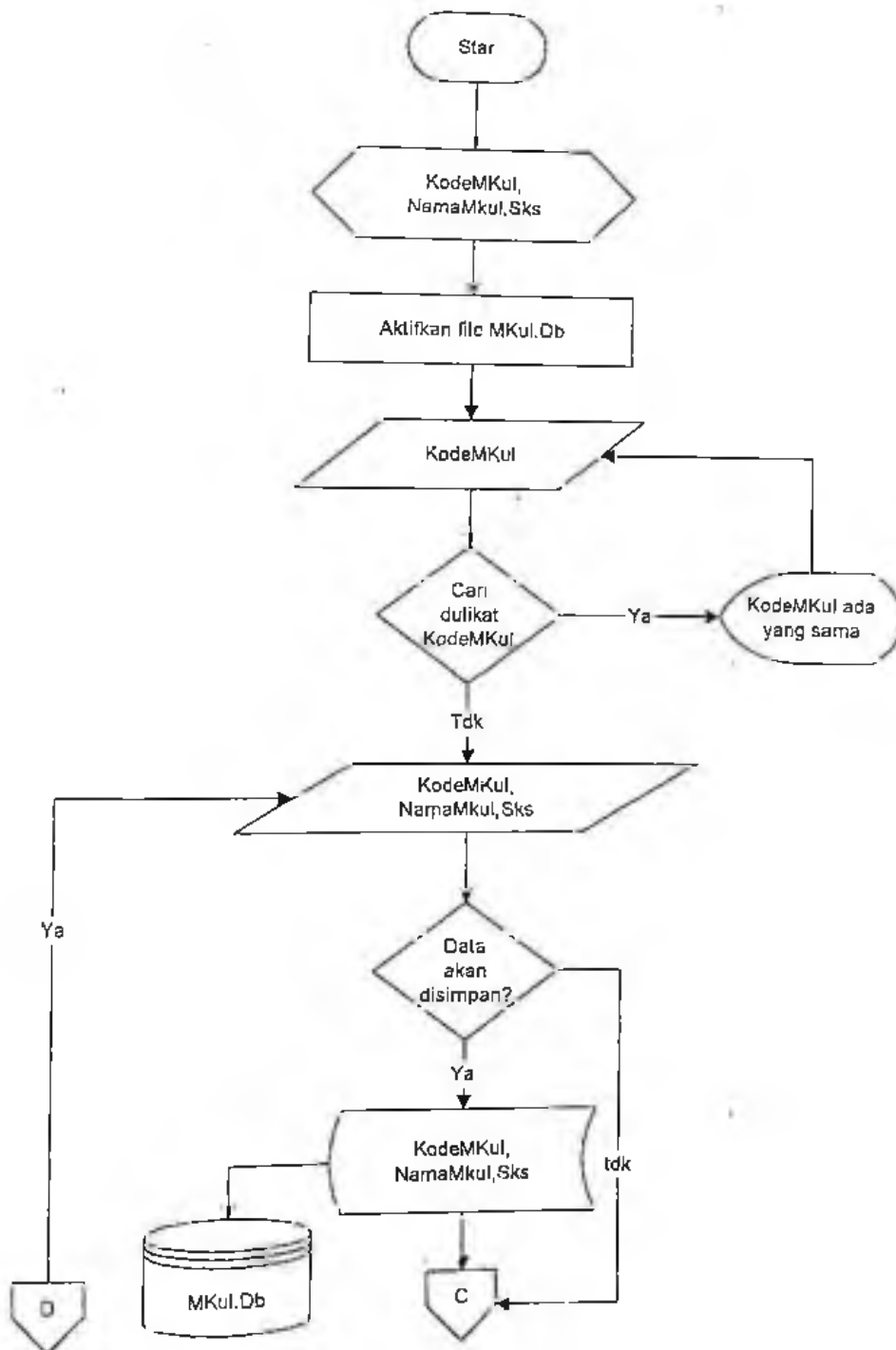
FlowChart Data Mahasiswa



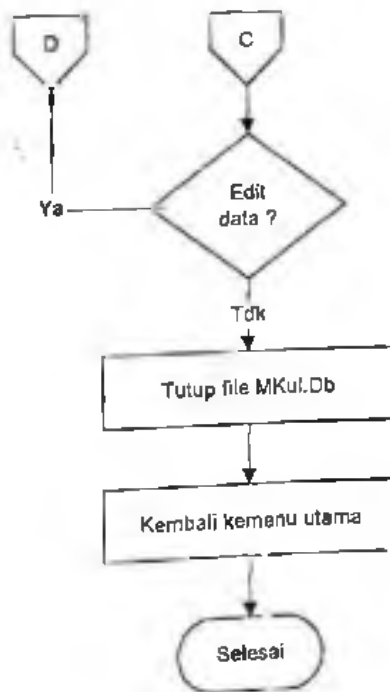
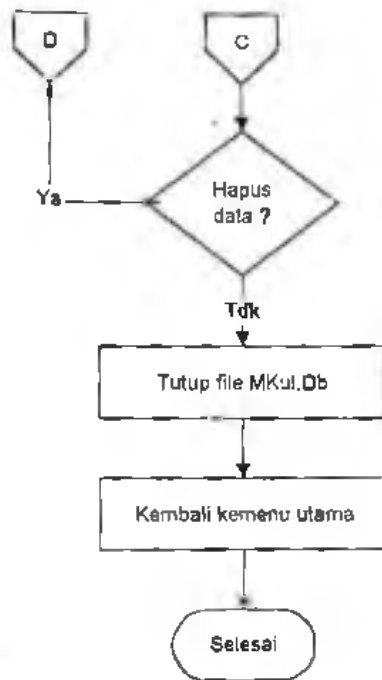
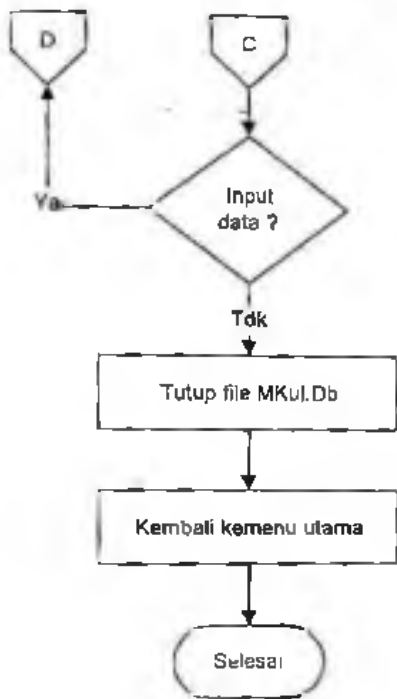
FlowChart Program



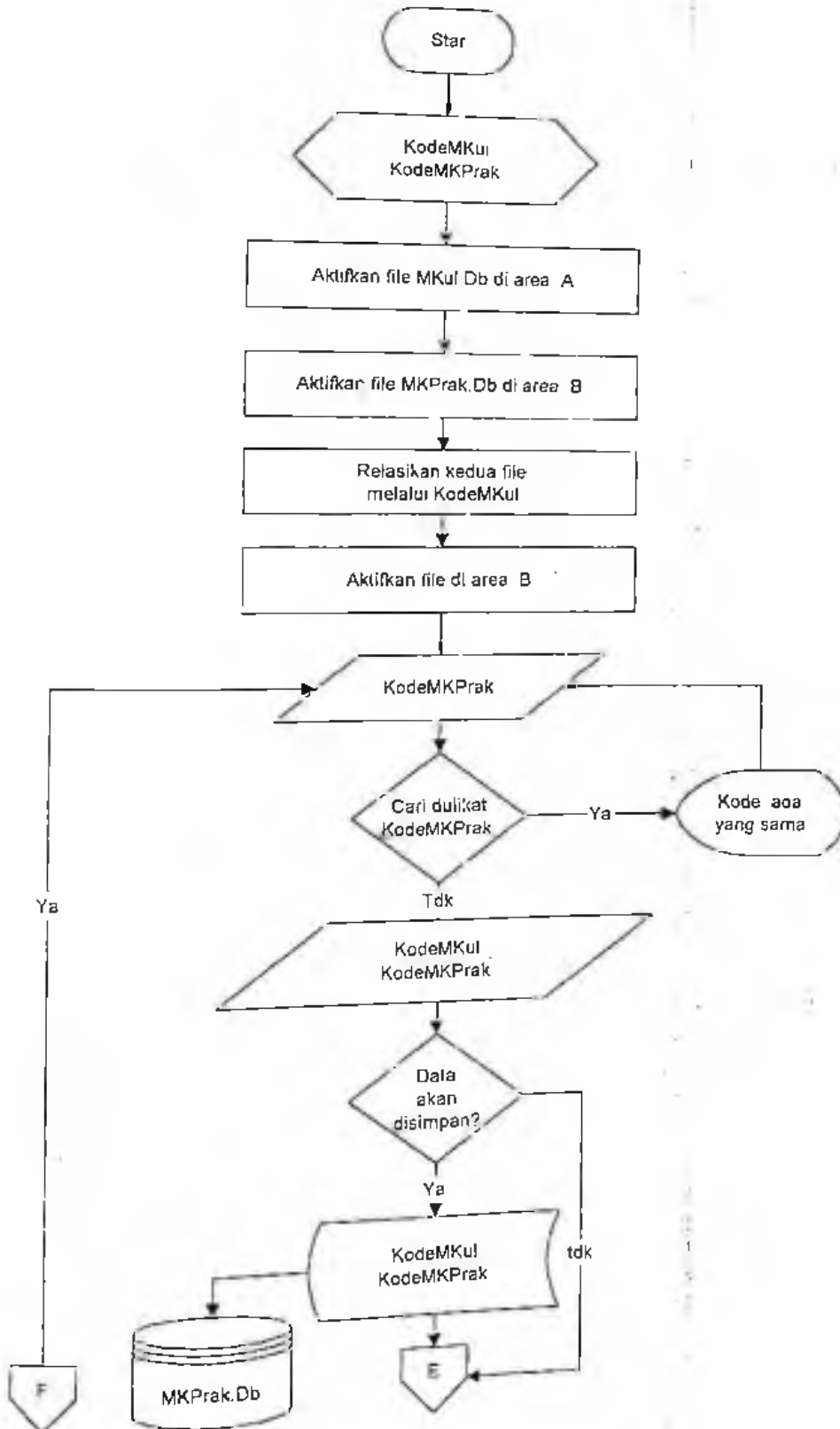
FlowChart Data Mata Kuliah



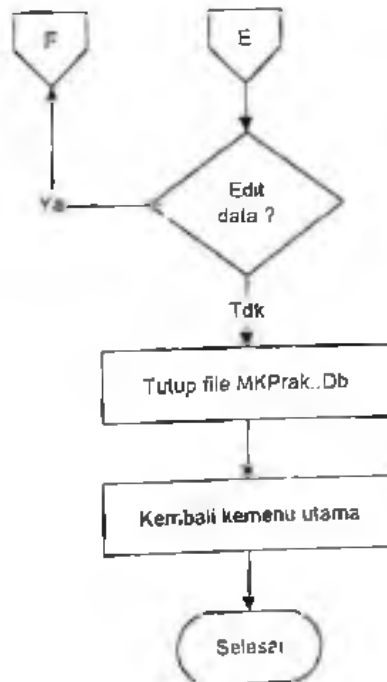
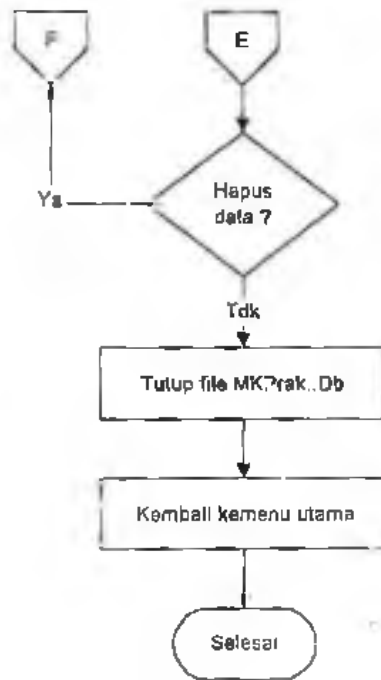
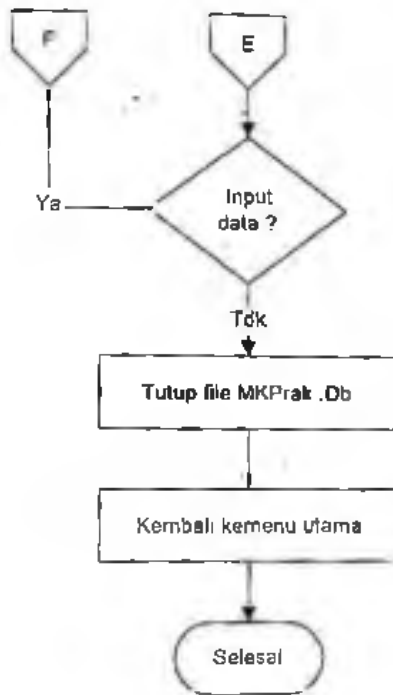
FlowChart Program



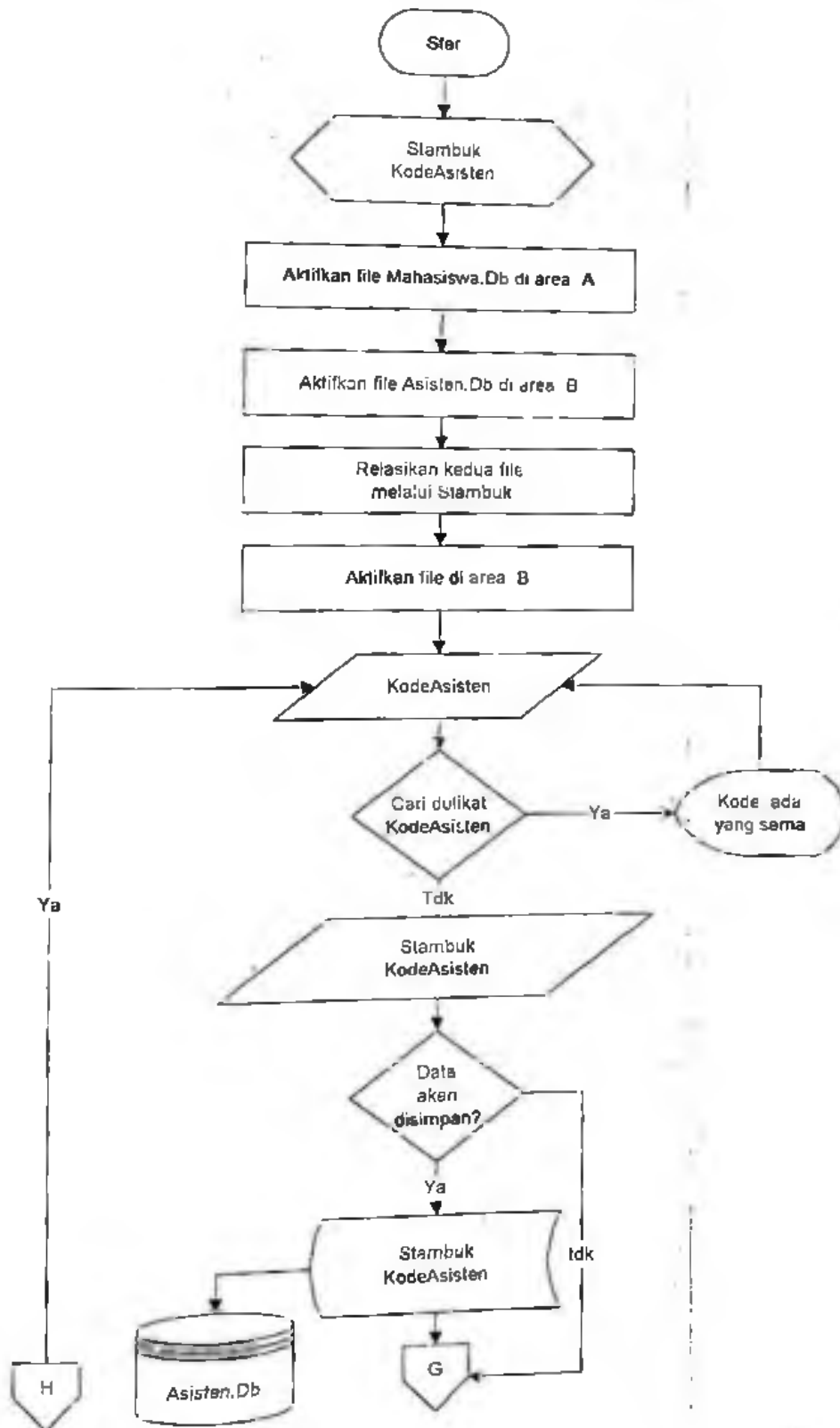
FlowChart Data Mata Kuliah Praktek



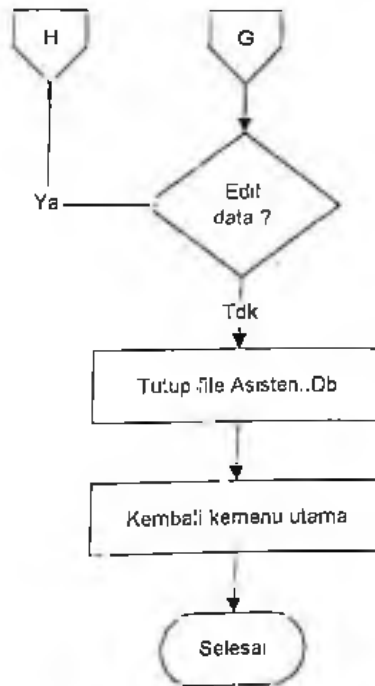
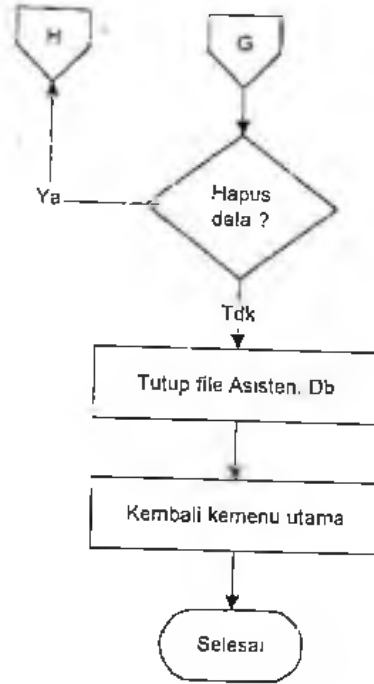
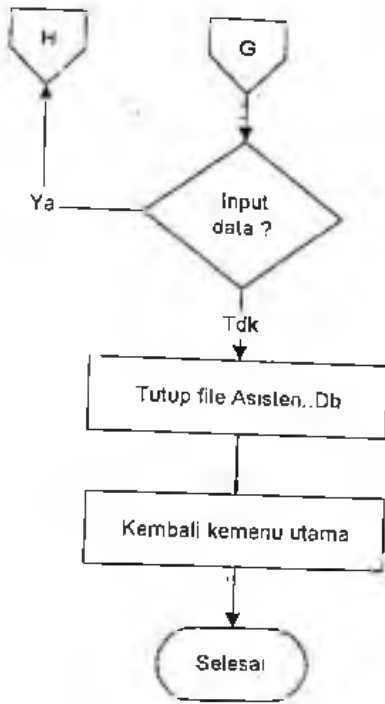
FlowChart Program



FlowChart Data Asisten

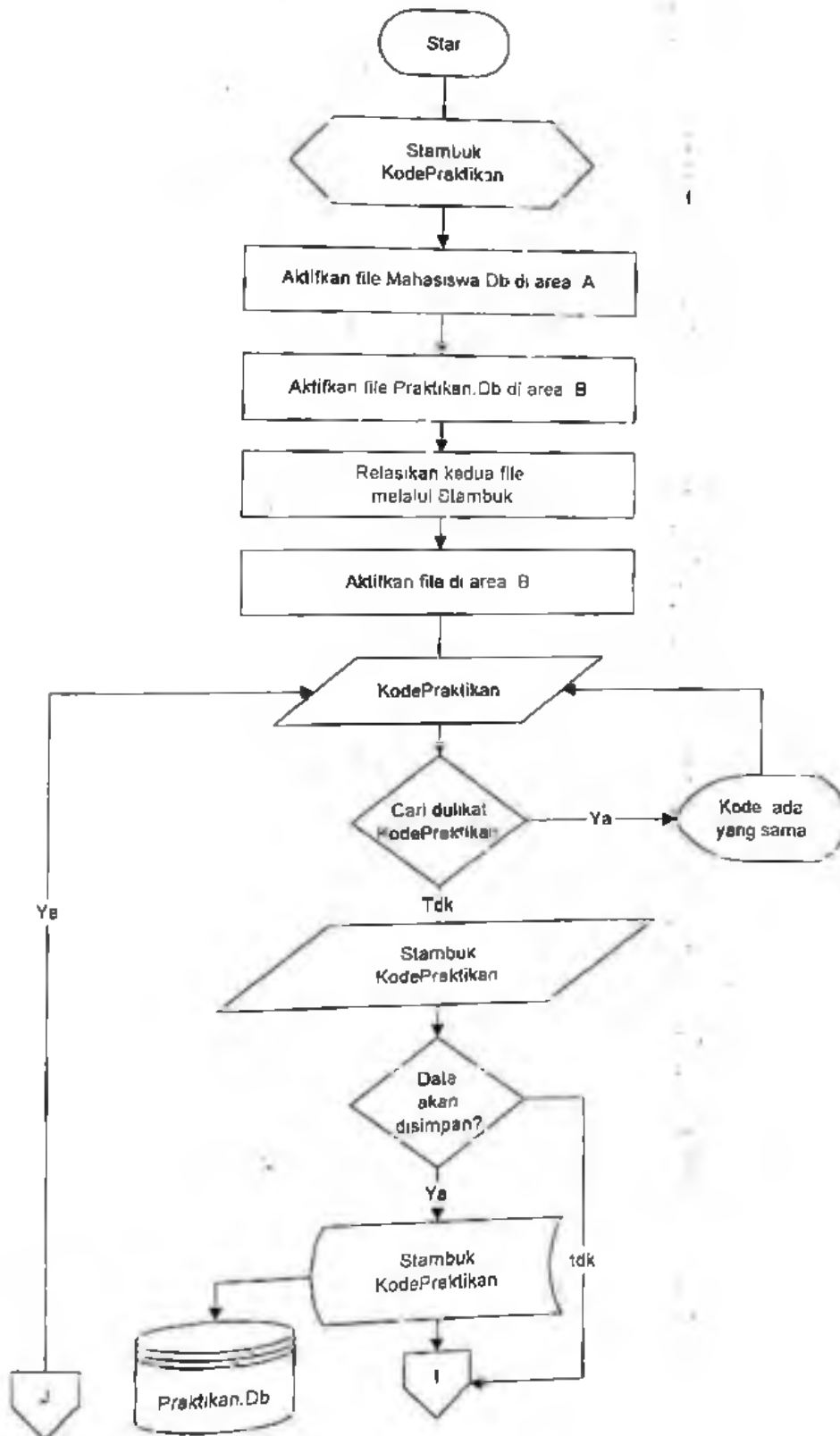


FlowChart Program

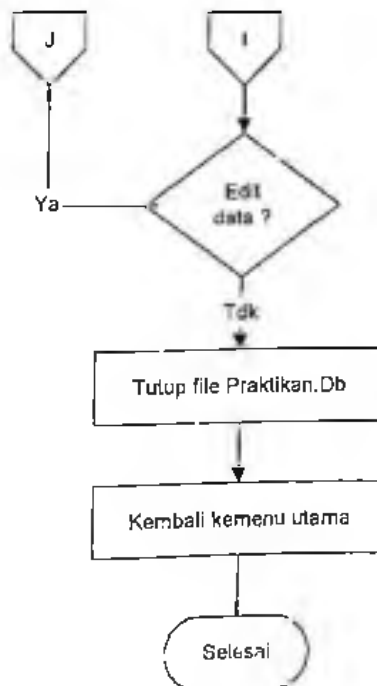
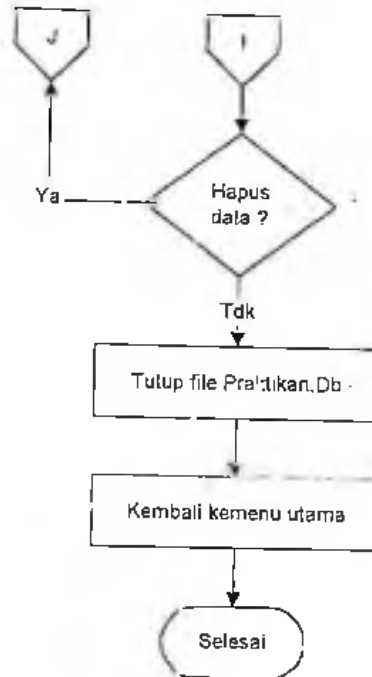
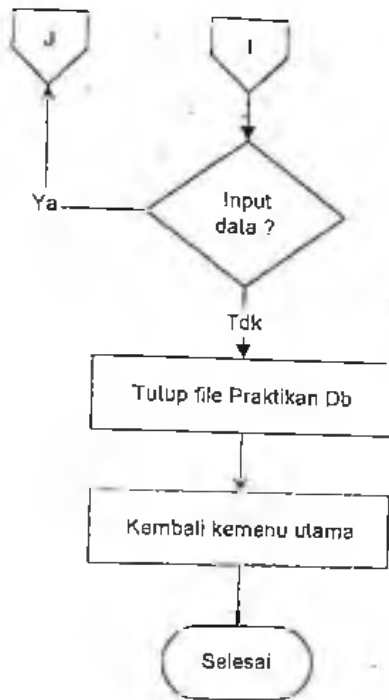




FlowChart Data Praktikan

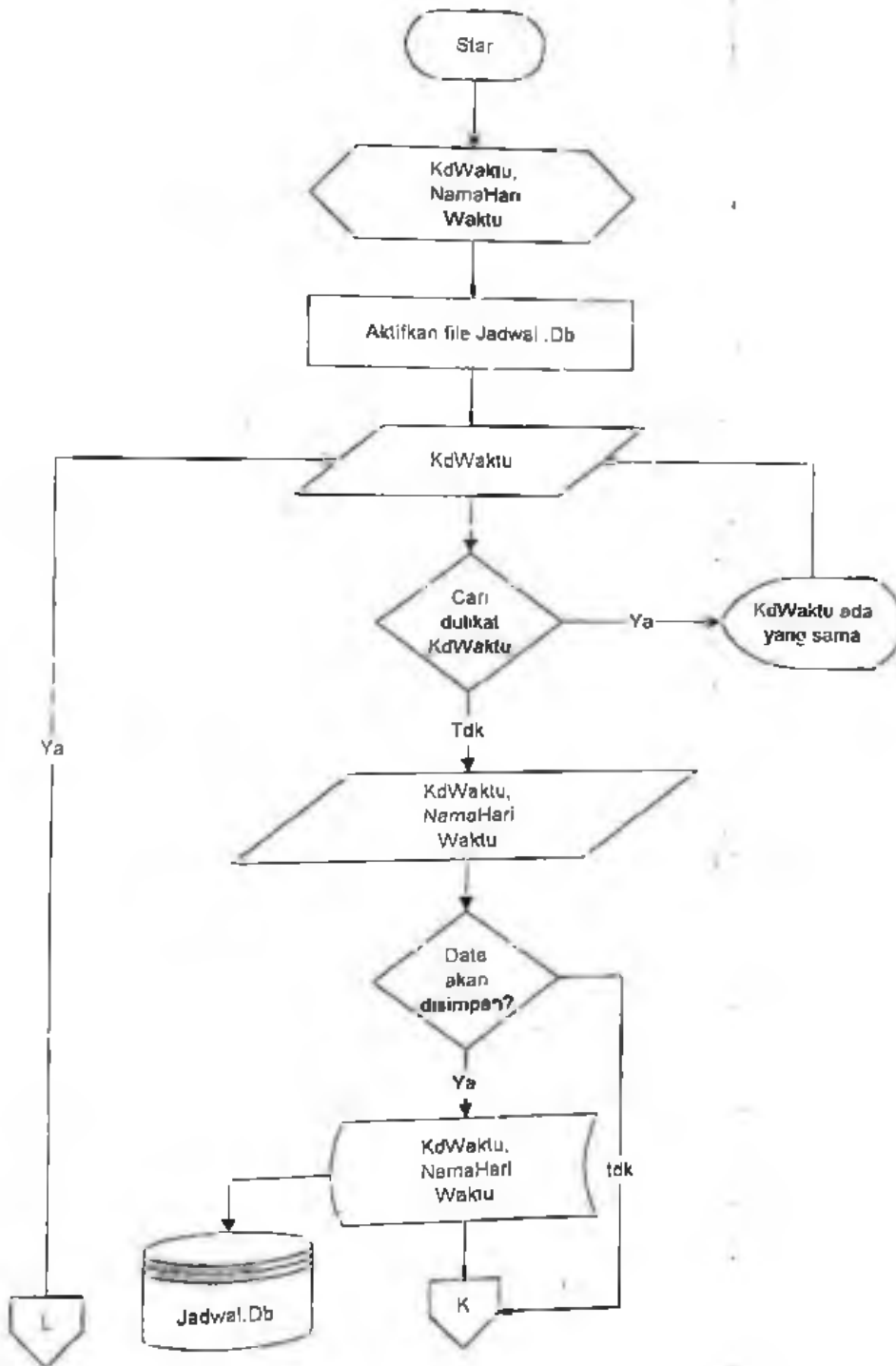


FlowChart Program

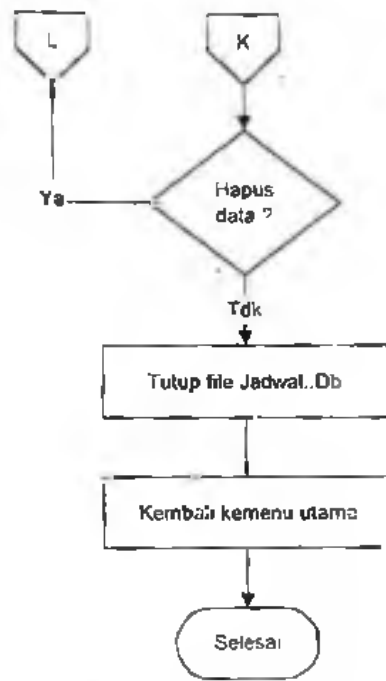
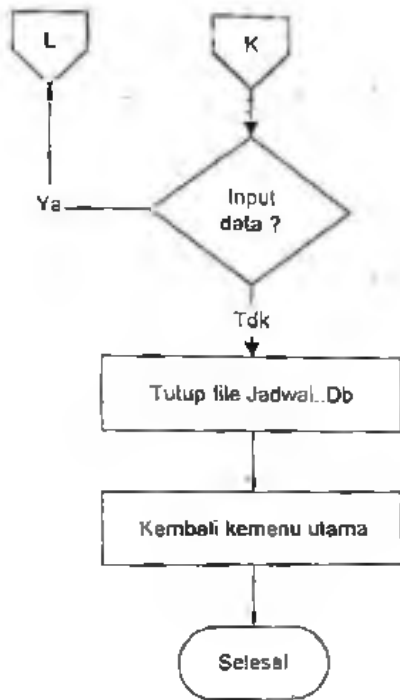




FlowChart Data Jadwal

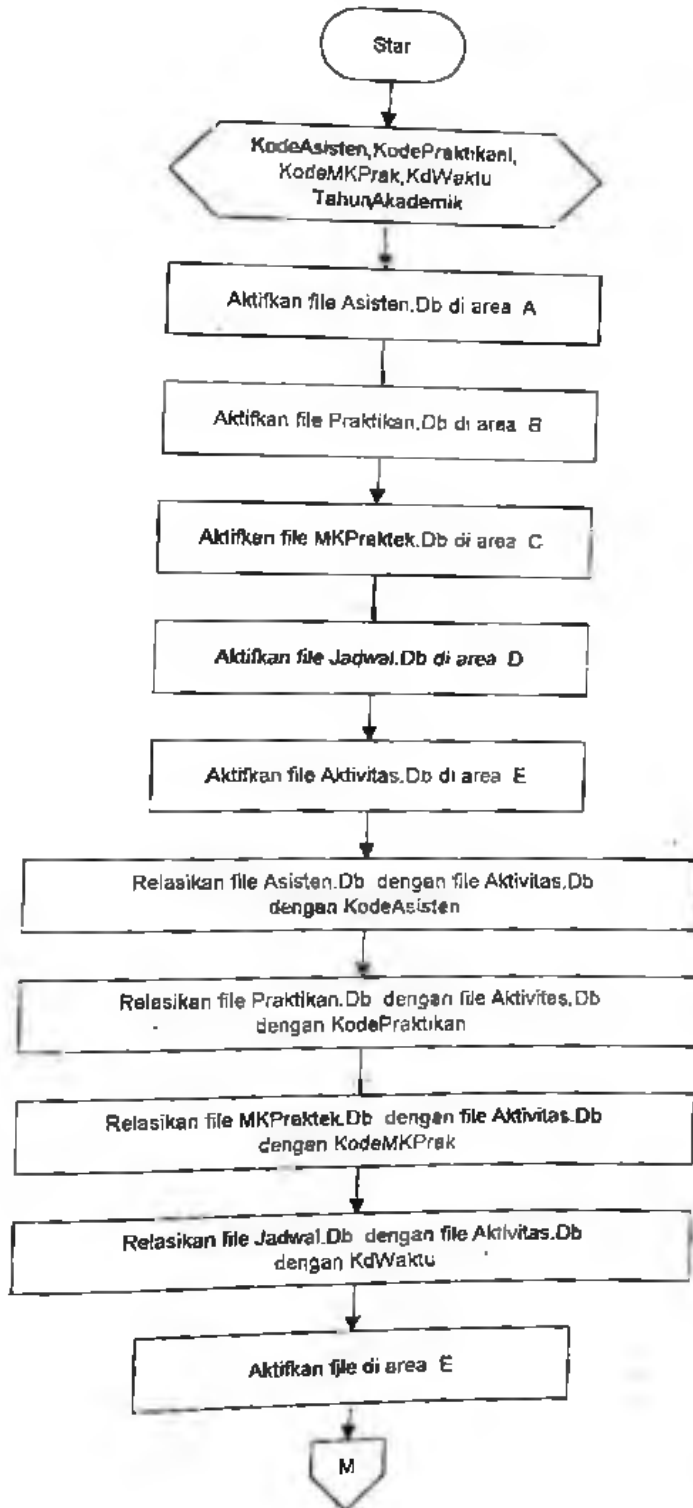


FlowChart Program

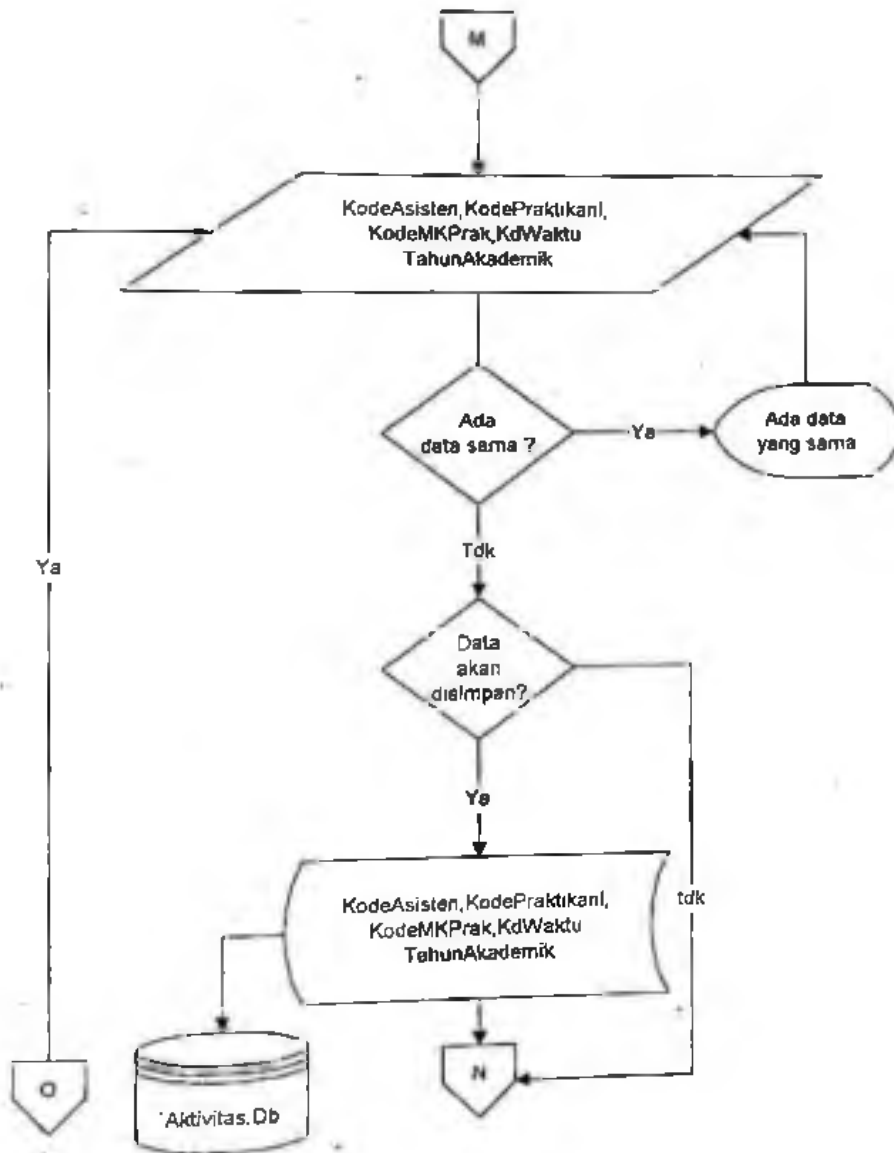




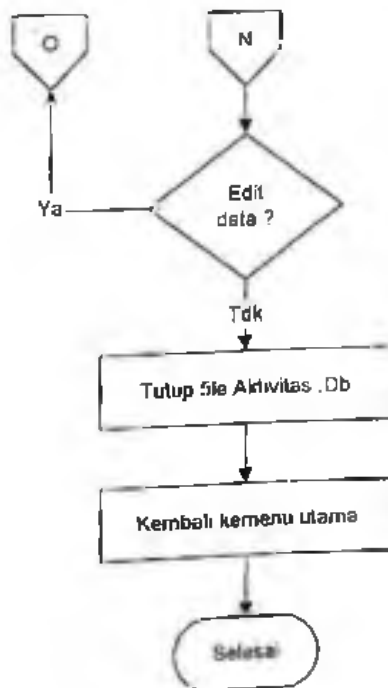
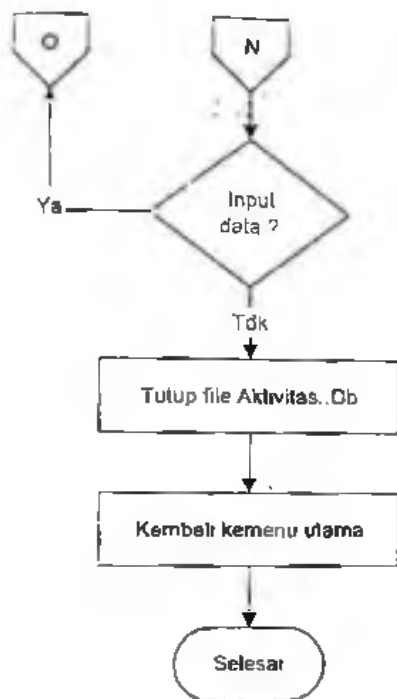
FlowChart Data Aktivitas



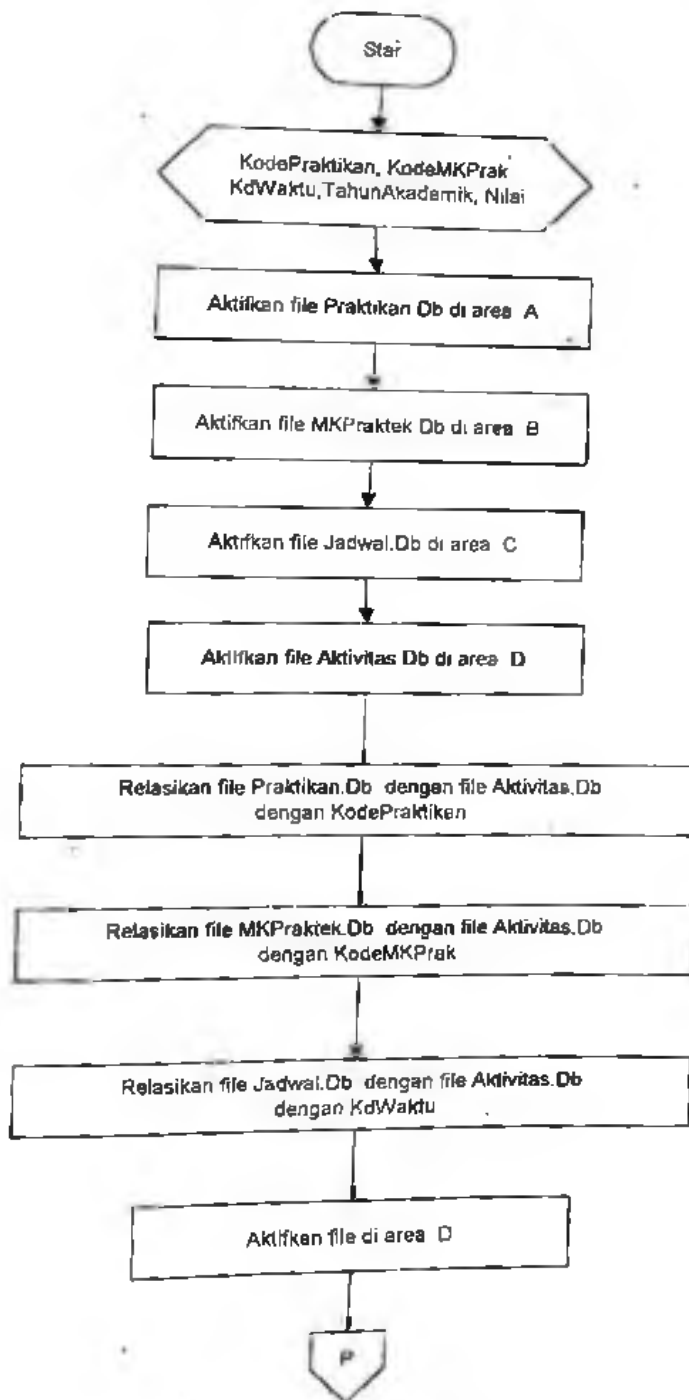
FlowChart Program



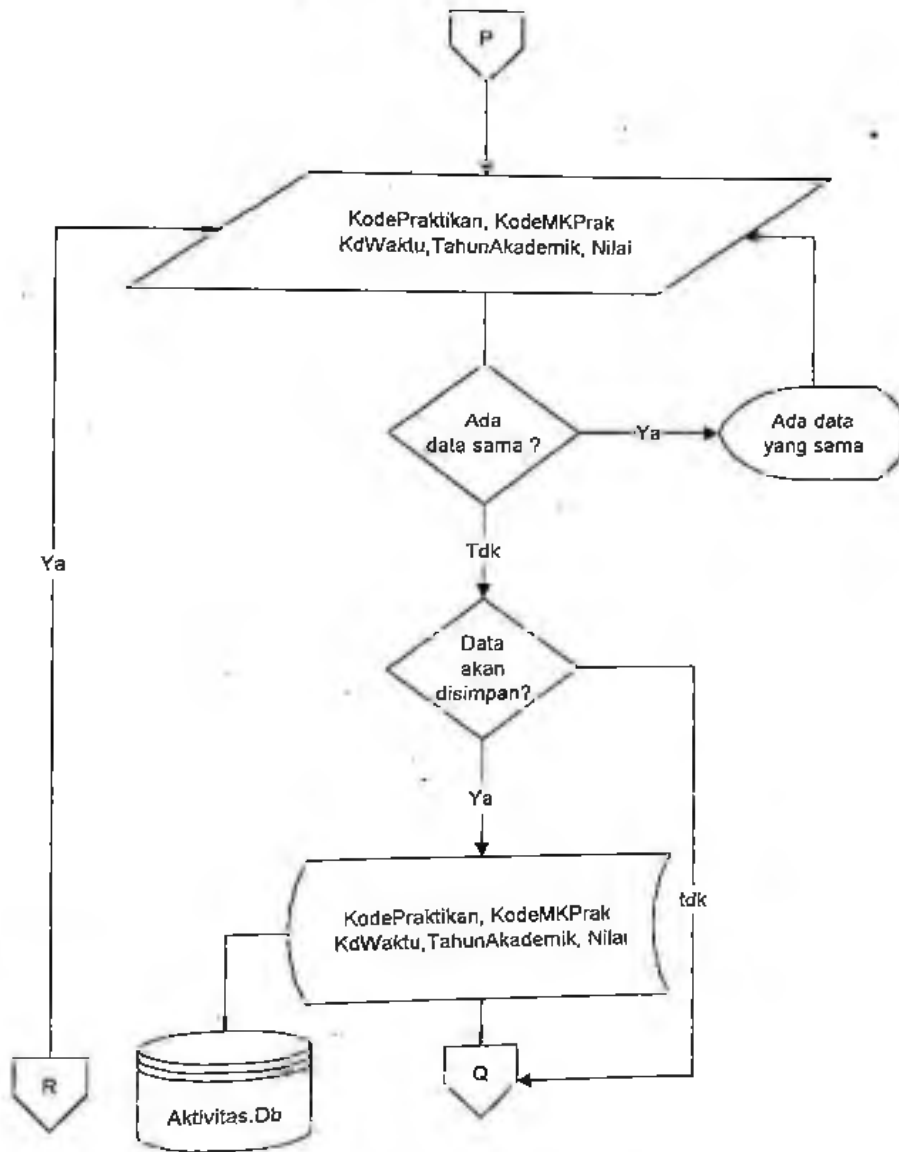
FlowChart Program



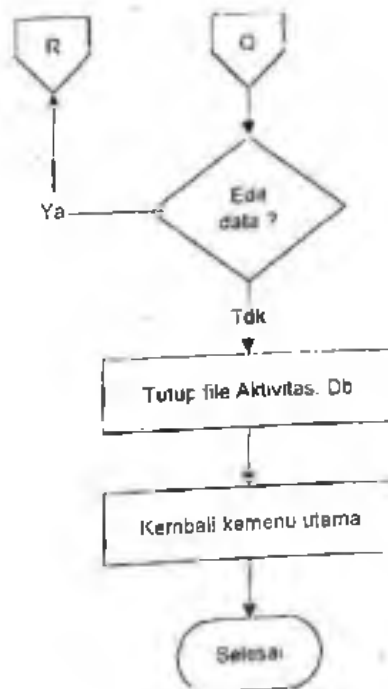
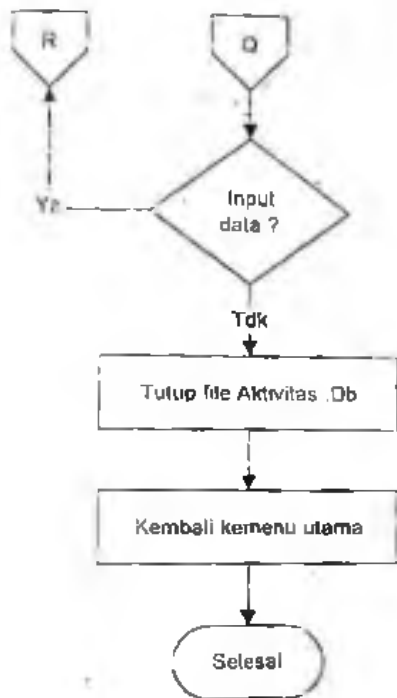
FlowChart Data Nilai



FlowChart Program



FlowChart Program



Program Lab_Mipa;

unit utama;

interface

uses

Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
Menus, ExtCtrls, StdCtrls, ComCtrls;

type

```
TFrm_utama = class(TForm)
  MainMenu1: TMainMenu;
  Informasi1: TMenuItem;
  Setup1: TMenuItem;
  SetupMahasiswa1: TMenuItem;
  SetupMataKuliah1: TMenuItem;
  Setmkpraktek: TMenuItem;
  SetupPraktikan1: TMenuItem;
  SetupAsisten1: TMenuItem;
  Set_jadwal: TMenuItem;
  Reset1: TMenuItem;
  SatuPriode5Tahun1: TMenuItem;
  About1: TMenuItem;
  AboutProgram1: TMenuItem;
  InformasiMahasiswa1: TMenuItem;
  informasiMataKuliahPraktek1: TMenuItem;
  InformasiAsisten1: TMenuItem;
  InformasiJadwal1: TMenuItem;
  NI: TMenuItem;
  Keluar1: TMenuItem;
  Input1: TMenuItem;
  InputNilai1: TMenuItem;
  InputAktivitas2: TMenuItem;
  InputAbsensi2: TMenuItem;
  BrowseMataKuliahPraktek1: TMenuItem;
  PreviewMataKuliaPraktek1: TMenuItem;
  BrowseMahasiswaPraktikan1: TMenuItem;
  PreviewMahasiswaPraktikan1: TMenuItem;
  BrowseAssitenLab1: TMenuItem;
  PreviewAsistenLab1: TMenuItem;
  InfoAktivitaslaboratorium1: TMenuItem;
  InfoNilaiPraktikan1: TMenuItem;
  BrowseAktivitas1: TMenuItem;
  Preview1: TMenuItem;
  BrowseNilaiPraktikan1: TMenuItem;
  Preview2: TMenuItem;
```

Listing Program

```
BrowseJadwal1: TMenuItem;
Preview3: TMenuItem;
N2: TMenuItem; N3: TMenuItem; N4: TMenuItem;
Reset2: TMenuItem;
Panel1: TPanel;
HeaderControl1: THeaderControl;
GroupBox1: TGroupBox;
Image1: TImage; Label7: TLabel; Label8: TLabel;
Label9: TLabel; Label10: TLabel; Label11: TLabel; Label12: TLabel;
procedure Keluar1Click(Sender: TObject);
procedure AboutProgram1Click(Sender: TObject);
procedure SetupMahasiswa1Click(Sender: TObject);
procedure FormKeyPress(Sender: TObject; var Key: Char);
procedure SetupMataKuliah1Click(Sender: TObject);
procedure SetmkpraktekClick(Sender: TObject);
procedure Set_jadwalClick(Sender: TObject);
procedure SetupPraktikan1Click(Sender: TObject);
procedure SetupAsisten1Click(Sender: TObject);
procedure FormActivate(Sender: TObject);
procedure FormDb1Click(Sender: TObject);
procedure InputAktivitas1Click(Sender: TObject);
procedure InputAbsensi1Click(Sender: TObject);
procedure InputNilaiPraktikum1Click(Sender: TObject);
procedure InputNilai1Click(Sender: TObject);
procedure InputAktivitas2Click(Sender: TObject);
procedure InputAbsensi2Click(Sender: TObject);
procedure BrowseMataKuliahPraktek1Click(Sender: TObject);
procedure PreviewMataKuliaPraktek1Click(Sender: TObject);
procedure browse1Click(Sender: TObject);
procedure BrowseMahasiswaPraktikan1Click(Sender: TObject);
procedure PreviewMahasiswaPraktikan1Click(Sender: TObject);
procedure BrowseAssitenLab1Click(Sender: TObject);
procedure PreviewAsistenLab1Click(Sender: TObject);
procedure BrowseAktivitas1Click(Sender: TObject);
procedure Preview1Click(Sender: TObject);
procedure BrowseNilaiPraktikan1Click(Sender: TObject);
procedure Preview2Click(Sender: TObject);
procedure BrowseJadwal1Click(Sender: TObject);
procedure Preview3Click(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;
```

Listing Program

```
var
  Frm_utama: TFrm_utama;
  status : boolean;

implementation

  uses about, set_mahasiswa, mata_kuliah, cari, set_mkpraktek, set_jadwal,
  set_praktikan, set_asisten, set_aktivitas_1, set_absensi, absensi,
  inp_nilai, info_matkul, lap_info_matkul, info_mahasiswa, lap_praktikan,
  info_asisten, lap_asisten, info_aktivitas, lap_aktivitas, info_nilai,
  lap_Nilai_Praktikan, lap_Jadwal, info_Jadwal;

{$R *.DFM}

procedure TFrm_utama.Keluar1Click(Sender: TObject);
begin
  application.terminate
end;

procedure TFrm_utama.AboutProgram1Click(Sender: TObject);
begin
  frm_about.showmodal;
end;

procedure TFrm_utama.SetupMahasiswa1Click(Sender: TObject);
begin
  frm_Mahasiswa.showmodal;
end;

procedure TFrm_utama.FormKeyPress(Sender: TObject; var Key: Char);
begin
  If key = #13 then
  begin
    key := #0;
    PostMessage(Handle, WM_NEXTDLGCTL, 0, 0);
  end,
end;

procedure TFrm_utama.SetupMataKuliah1Click(Sender: TObject);
begin
  frmset_mkul.showmodal;
end;
```


Listing Program

```
procedure TFrm_utama.SetmkpraktekClick(Sender: TObject);
begin
  frmset_mkpraktek.showmodal;
end;

procedure TFrm_utama.Set_jadwalClick(Sender: TObject);
begin
  frm_jadwal1.showmodal;
end;

procedure TFrm_utama.SetupPraktikan1Click(Sender: TObject);
begin
  Frmset_praktikan.showmodal;
end;

procedure TFrm_utama.SetupAsisten1Click(Sender: TObject);
begin
  Frmset_asisten.showmodal;
end;

procedure TFrm_utama.FormActivate(Sender: TObject);
begin
  status := True;
end;

procedure TFrm_utama.FormDb1Click(Sender: TObject);
begin
  showmessage('masuk');
  Frm_utama.ClientHeight := -2;
end;

procedure TFrm_utama.InputAktivitas1Click(Sender: TObject);
begin
  frm_aktivitas.show;
end;

procedure TFrm_utama.InputAbsensi1Click(Sender: TObject);
begin
  frm_absensi1.show;
end;

procedure TFrm_utama.InputNilaiPraktikum1Click(Sender: TObject);
begin
  frm_inp_nilai.show;
end;
```

Listing Program

```
procedure TFrm_utama.InputNilai1Click(Sender: TObject);  
begin  
    frm_inp_nilai.show;  
end;
```

```
procedure TFrm_utama.InputAktivitas2Click(Sender: TObject);  
begin  
    frm_aktivitas.show;  
end;
```

```
procedure TFrm_utama.InputAbsensi2Click(Sender: TObject);  
begin  
    frm_absensi1.show;  
end;
```

```
procedure TFrm_utama.BrowseMataKuliahPraktek1Click(Sender: TObject);  
begin  
    frm_info_mkPraktek.show;  
end;
```

```
procedure TFrm_utama.PreviewMataKuliaPraktek1Click(Sender: TObject);  
begin  
    qr_Info_matkul.preview;  
end;
```

```
procedure TFrm_utama.browse1Click(Sender: TObject);  
begin  
    frm_info_mahasiswa.show;  
end;
```

```
procedure TFrm_utama.BrowseMahasiswaPraktikan1Click(Sender: TObject);  
begin  
    frm_info_mahasiswa.show;  
end;
```

```
procedure TFrm_utama.PreviewMahasiswaPraktikan1Click(Sender: TObject);  
begin  
    Qr_Praktikan.preview;  
end;
```

```
procedure TFrm_utama.BrowseAssitenLab1Click(Sender: TObject);  
begin  
    frm_info_asisten.show;  
end;
```

```
procedure TFrm_utama.PreviewAsistenLab1Click(Sender: TObject);
begin
  Qr_info_asisten.preview;
end;
```

```
procedure TFrm_utama.BrowseAktivitas1Click(Sender: TObject);
begin
  frm_info_aktivitas.show;
end;
```

```
procedure TFrm_utama.Preview1Click(Sender: TObject);
begin
  Qr_Info_Aktivitas.preview;
end;
```

```
procedure TFrm_utama.BrowseNilaiPraktikan1Click(Sender: TObject);
begin
  frm_Nilai_Praktikan.show;
end;
```

```
procedure TFrm_utama.Preview2Click(Sender: TObject);
begin
  Qr_Info_Nilai.preview;
end;
```

```
procedure TFrm_utama.BrowseJadwal1Click(Sender: TObject);
begin
  frm_info_Jadwal.show;
end;
```

```
procedure TFrm_utama.Preview3Click(Sender: TObject);
begin
  Qr_Info_Jadwal.preview;
end;
```

end.

{ Mahasiswa }

unit set_mahasiswa;

interface

uses
Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
StdCtrls, Buttons, ExtCtrls, DBCtrls, Grids, DBGrids, Db, Mask, ComCtrls,
DBTables;

Listing Program

```
type
Tfrm_mahasiswa = class(TForm)
  Label1: TLabel; Label2: TLabel; set_mahasiswa: TTable;
  PageControl1: TPageControl; TabSheet1: TTabSheet; TabSheet2: TTabSheet;
  set_mahasiswaStambuk: TStringField; set_mahasiswaNama: TStringField;
  set_mahasiswaJurusan: TStringField; set_mahasiswaAlamat: TStringField;
  set_mahasiswaTanggal_Lahir: TDateField; Label3: TLabel;
  DBEdit1: TDBEdit; DataSource1: TDataSource; Label4: TLabel;
  DBEdit2: TDBEdit; Label5: TLabel; Label6: TLabel; DBEdit4: TDBEdit;
  Label7: TLabel; DBEdit5: TDBEdit; DBGrid1: TDBGrid;
  DBNavigator1: TDBNavigator; BitBtn1: TBitBtn;
  DBComboBox1: TDBComboBox;
  procedure BitBtn1Click(Sender: TObject);
  procedure FormKeyPress(Sender: TObject; var Key: Char);
  procedure set_mahasiswaPostError(DataSet: TDataSet; E: EDatabaseError;
    var Action: TDataAction);
  procedure DBEdit2KeyPress(Sender: TObject; var Key: Char);
  procedure set_mahasiswaBeforePost(DataSet: TDataSet);
private
public
end;

var
  frm_mahasiswa: Tfrm_mahasiswa;
implementation

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

procedure Tfrm_mahasiswa.FormKeyPress(Sender: TObject; var Key: Char);
begin
  if key = #13 then
  begin
    key := #0;
    PostMessage(Handle, WM_NEXTDLGCTL, 0, 0);
  end;
end;

procedure Tfrm_mahasiswa.set_mahasiswaPostError(DataSet: TDataSet;
  E: EDatabaseError; var Action: TDataAction);
begin
  if (E is EDBEngineError) then
```

```
if (E as EDBEngineError).Errors[0].Errorcode = 9729 then
begin
  MessageDlg('Tidak bisa direkam, Kode Double', mtWarning, [mbOK], 0);
  Abort;
end;
end;

procedure Tfrm_mahasiswa DBEdit2KeyPress(Sender: TObject; var Key: Char);
begin
  If key = #13 then
  begin
    key := #0;
    PostMessage(Handle, WM_NEXTDLGCTL, 0, 0);
  end;
end;

{ Mata Kuliah }
unit mata_kuliah;
interface
uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  ExtCtrls, DBCtrls, StdCtrls, Buttons, Grids, DBGrids, Db, Mask, ComCtrls,
  DBTables;

type
  Tfrmset_mkul = class(TForm)
    Label2: TLabel; Label1: TLabel; Table1: TTable;
    PageControl1: TPageControl; TabSheet1: TTabSheet; TabSheet2: TTabSheet;
    Table1Kode_mata_kuliah: TStringField; Table1Nama_mata_kuliah: TStringField;
    Table1Sks: TFloatField; Label3: TLabel; DBEdit1: TDBEdit;
    DataSource1: TDataSource; Label4: TLabel; DBEdit2: TDBEdit;
    Label5: TLabel; DBEdit3: TDBEdit; DBGrid1: TDBGrid; BitBtn1: TBitBtn;
    DBNavigator1: TDBNavigator; procedure BitBtn1Click(Sender: TObject);
    procedure FormKeyPress(Sender: TObject; var Key: Char);
  private
  public
  end;

var
  frmset_mkul: Tfrmset_mkul;
implementation

procedure Tfrmset_mkul.BitBtn1Click(Sender: TObject);
begin
```

Listing Program

```
close;  
end;
```

```
procedure Tfrmset_mkul.FormKeyPress(Sender: TObject; var Key: Char);  
begin  
  If key = #13 then  
    begin  
      key := #0;  
      PostMessage(Handle, WM_NEXTDLGCTL, 0, 0);  
    end;  
  end;  
end.
```

{ Jadwal }

```
unit set_jadwal;  
interface  
uses  
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,  
  StdCtrls, ExtCtrls, DBCtrls, Mask, Db, DBTables, Grids, DBGrids, ComCtrls;
```

```
type  
TFrm_Jadwal = class(TForm)  
  Label2: TLabel; Label1: TLabel; PageControl1: TPageControl;  
  TabSheet1: TTabSheet; Label3: TLabel; Label4: TLabel;  
  TabSheet2: TTabSheet; DBRadioGroup1: TDBRadioGroup;  
  set_jadwal: TTable; DataSource1: TDataSource;  
  set_jadwalKd_waktu: TStringField; set_jadwalNamaHari: TStringField;  
  set_jadwalWaktu: TStringField; DBComboBox1: TDBComboBox;  
  Label5: TLabel; DBEdit1: TDBEdit; DBNavigator1: TDBNavigator;  
  Button1: TButton; DBGrid1: TDBGrid;  
  procedure Button1Click(Sender: TObject);  
  procedure DBRadioGroup1Change(Sender: TObject);  
  procedure set_jadwalPostError(DataSet: TDataSet; E: EDatabaseError;  
    var Action: TDataAction);
```

```
private
```

```
public
```

```
end;
```

```
var
```

```
  Frm_Jadwal1: TFrm_Jadwal1;
```

```
implementation
```

```
  procedure TFrm_Jadwal1.Button1Click(Sender: TObject);
```

```
  begin
```

```
    close;
```

```
  end;
```

```
procedure TFrm_Jadwal.DBRadioGroup1Change(Sender: TObject);
var
  kd, nm_hr : string;
  ke : string[2];

begin
  if DbComboBox1.Text='Senin' then
    nm_hr := 'sn'
  else if DbComboBox1.Text='Selasa' then
    nm_hr := 'sl'
  else if DbComboBox1.Text='Rabu' then
    nm_hr := 'rb'
  else if DbComboBox1.Text='Kamis' then
    nm_hr := 'km'
  else if DbComboBox1.Text='Jumat' then
    nm_hr := 'jm'
  else if DbComboBox1.Text='Minggu' then
    nm_hr := 'mg'
  else if DbComboBox1.Text='Sabtu' then
    nm_hr := 'sb';

  case dbRadioGroup1.ItemIndex of
    0 : ke := '01';
    1 : ke := '02';
    2 : ke := '03';
    3 : ke := '04';
    4 : ke := '05';
    5 : ke := '06';
  end;
  kd := nm_hr+ke;
  dbEdit1.Text := kd;
end;

procedure TFrm_Jadwal.set_jadwalPostError(DataSet: TDataSet,
  E: EDatabaseError; var Action: TDataAction);
begin
  if (E is EDBEngineError) then
    begin
      if (E as EDBEngineError).Errors[0].ErrorCode = 9729 then
        begin
          showMessage('Kode duplikasi');
          abort;
        end;
    end;
end;
```

Listing Program



end;
end.

{ Mata Kuliah Praktek Praktek }

unit set_mkpraktek;

interface

uses

Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
StdCtrls, Buttons, ExtCtrls, DBCtrls, Grids, DBGrids, Db. Mask, DBTables,
ComCtrls;

type

Tfrmset_mkpraktek = class(TForm)

Label1: TLabel; Label2: TLabel; PageControl1: TPageControl;
TabSheet1: TTabSheet; TabSheet2: TTabSheet; Label3: TLabel;
Label4: TLabel; DBEdit2: TDBEdit; DBGrid1: TDBGrid;
DBNavigator1: TDBNavigator; BitBtn1: TBitBtn;
DBLookupListBox1: TDBLookupListBox; ds_mk_praktek: TDataSource;
tb_mk_praktek: TTable; ds_matkul: TDataSource;
tb_matkul: TTable; StringField1: TStringField; StringField2: TStringField;
FloatField1: TFloatField; tb_mk_praktekKode_mata_kuliah: TStringField;
Label5: TLabel; DBText1: TDBText; Label6: TLabel;
tb_mk_praktekKode_mk_praktek: TStringField;
DBLookupComboBox1: TDBLookupComboBox; DBEdit1: TDBEdit;
tb_mk_praktekMata_kuliah: TStringField;
procedure BitBtn1Click(Sender: TObject);
procedure FormKeyPress(Sender: TObject; var Key: Char);
procedure DBEdit1Exit(Sender: TObject);
procedure tb_mk_praktekPostError(DataSet: TDataSet; E: EDatabaseError;
var Action: TDataAction);
procedure DBLookupListBox1Click(Sender: TObject);
procedure DBEdit1Enter(Sender: TObject);
procedure tb_mk_praktekBeforePost(DataSet: TDataSet);

private

public

end;

var

frmset_mkpraktek: Tfrmset_mkpraktek;

implementation

procedure Tfrmset_mkpraktek.BitBtn1Click(Sender: TObject);

begin

close;

end;


```
procedure Tfrmset_mkpraktek.FormKeyPress(Sender: TObject; var Key: Char);
begin
  if key = #13 then
  begin
    key := #0;
    PostMessage(Handle, WM_NEXTDLGCTL, 0, 0);
  end;
end;
```

```
procedure Tfrmset_mkpraktek.tb_mk_praktekPostError(DataSet: TDataSet;
  E: EDatabaseError; var Action: TDataAction);
begin
  if (E is EDBEngineError) then
  if (E as EDBEngineError).Errors[0].ErrorCode = 9729 then
  begin
    showmessage('Ada kode yang sama');
    abort;
  end;
end;
```

```
procedure Tfrmset_mkpraktek.DBLookupListBox1Click(Sender: TObject);
begin
  DbLookupcomboBox1.Visible := False;
  DbEdit1.Visible := True;
end;
```

```
procedure Tfrmset_mkpraktek.DBEdit1Enter(Sender: TObject);
begin
  dbEdit1.Visible := false;
  DbLookupCOMboBox1.Visible := true;
end;
```

```
procedure Tfrmset_mkpraktek.tb_mk_praktekBeforePost(DataSet: TDataSet);
begin
  tb_mk_praktek.FieldName('Mata_Kuliah').AsString :=
  DbLookupListBox1.SelectedItem;
end;
end.
```

{ Asisten }

unit set_asisten;

interface

uses

Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,

Listing Program

ExtCtrls, DBCtrls, StdCtrls, Buttons, Grids, DBGrids, Db. Mask, DBTables, ComCtrls;

```
type
TFrmset_Asisten = class(TForm)
  Label1: TLabel; Label2: TLabel; PageControl1: TPageControl;
  TabSheet1: TTabSheet; TabSheet2: TTabSheet; tb_asisten: TTable,
  Label3: TLabel; ds_praktikan: TDataSource; Label4: TLabel;
  tb_mahasiswa: TTable; tb_mahasiswaStambuk: TStringField;
  tb_mahasiswaNama: TStringField; tb_mahasiswaJurusan: TStringField;
  tb_mahasiswaAlamat: TStringField; tb_mahasiswaTanggal_Lahir: TDateField;
  Label6: TLabel; ds_mahasiswa: TDataSource; Label7: TLabel;
  DBGrid1: TDBGrid; BitBtn1: TBitBtn; DBNavigator1: TDBNavigator;
  DBLookupListBox1: TDBLookupListBox; tb_mk_praktek: TTable.
  ds_mk_praktek: TDataSource; tb_mk_praktekKode_mk_praktek: TStringField;
  tb_mk_praktekKode_mata_kuliah: TStringField; Label5: TLabel;
  DBLookupComboBox1: TDBLookupComboBox; Edit1: TEdit;
  tb_matkul: TTable; ds_matkul: TDataSource;
  tb_matkulKode_mata_kuliah: TStringField;
  tb_matkulNama_mata_kuliah: TStringField;
  tb_matkulSks: TFloatField; DBText1: TDBText; DBText2: TDBText;
  DBText3: TDBText; tb_asistenNo: TAutoIncField;
  tb_asistenKode_Asisten: TStringField; tb_asistenStambuk: TStringField;
  tb_asistenNama: TStringField;
  procedure BitBtn1Click(Sender: TObject);
  procedure DBLookupComboBox1Click(Sender: TObject);
  procedure tb_asistenFilterRecord(DataSet: TDataSet;
    var Accept: Boolean);
  procedure tb_asistenAfterPost(DataSet: TDataSet);
  procedure tb_asistenBeforePost(DataSet: TDataSet);
  procedure tb_asistenAfterInsert(DataSet: TDataSet);
  procedure Edit1Change(Sender: TObject);
  procedure tb_asistenNewRecord(DataSet: TDataSet);
  procedure DBLookupListBox1Exit(Sender: TObject);
  procedure PageControl1Change(Sender: TObject);
  procedure tb_asistenAfterScroll(DataSet: TDataSet);
private
public
end;

var
  Frmset_Asisten: TFrmset_Asisten;
  nm_terpilih, stb_terpilih : string;
implementation
```

Listing Program

```
procedure TFrmset_Asisten.BitBtn1Click(Sender: TObject);
begin
  application.Terminate;
end;

procedure TFrmset_Asisten.DBLookupComboBox1Click(Sender: TObject);
var
  s_pos, s_urut : string;
  pjg, pos, kode, n_urut : integer;
begin
  case tb_asisten.State of
  dsInsert : begin
    tb_asisten.last;
    s_urut := trim(copy(tb_asistenKode_Asisten.Value,5,4));
    val(s_urut,n_urut,kode);
    pos := n_urut + 1;
    s_pos := intToStr(pos);
    pjg := Length(s_pos);
    if
tb_asisten.Locate('stambuk;kode_asisten',VarArrayOf([stb_terpilih,'A'+dbLookupCo
mboBox1.Text]),[loPartialKey]) then
      begin
        if MessageDlg('Mahasiswa '+stb_terpilih+' sudah terdaftar'#15'Batalan
saja?',
          mtConfirmation, [mbYes, mbNo], 0) = mrYes then
          exit;
        end;
        tb_asisten.insert;
        case Pjg of
          1 : edit1.Text := 'A'+dbLookupComboBox1.Text+'000'+s_pos;
          2 : edit1.Text := 'A'+dbLookupComboBox1.Text+'00'+s_pos;
          3 : edit1.Text := 'A'+dbLookupComboBox1.Text+'0'+s_pos;
        end;
        tb_asisten.FieldName('Kode_asisten').AsString := Edit1.Text;
        tb_asisten.FieldName('stambuk').AsString := Stb_Terpilih;
        dbLookupListBox1.KeyValue :=
tb_Asisten.fieldbyname('Stambuk').AsString;
      end;
    dsBrowse : begin
      tb_asisten.last;
    end;
  end;
end;
end;
```

Listing Program

```
procedure TFrmset_Asisten.tb_asistenFilterRecord(DataSet: TDataSet;
  var Accept: Boolean);
begin
  Accept := copy(tb_asistenKode_Asisten.Value,2,3)=dbLookupComboBox1.Text;
end;

procedure TFrmset_Asisten.tb_asistenBeforePost(DataSet: TDataSet);
begin
  showmessage(eDIT1.TEXT);
  tb_asisten.fieldByName('Kode_asisten').AsString := Edit1.Text;
  tb_asisten.Fieldbyname('Nama').AsString := nm_terpilih;
end;

procedure TFrmset_Asisten.tb_asistenAfterInsert(DataSet: TDataSet);
begin
  edit1.Text := "";
end;

procedure TFrmset_Asisten.Edit1Change(Sender: TObject);
begin
  tb_asisten.Last;
  cari := tb_mk_praktek.fieldbyname('kode_mata_kuliah').AsString;
  if not tb_matkul.FindKey([cari]) then
    showmessage('tidak ditemukan');
  s_urut := copy(tb_asistenKode_asisten.Value,5,4);
  val(s_urut,n_urut,kode);
  pos := n_urut + 1;
  s_pos := intToStr(pos);
  Edit1.Text := 'A'+dbLookupComboBox1.Text+s_pos;
  tb_asisten.edit;
end;

procedure TFrmset_Asisten.DBLookupListBox1Exit(Sender: TObject);
begin
  sb_terpilih := DbLookupListBox1.SelectedItem;
  nm_terpilih := tb_mahasiswa.fieldByName('Nama').AsString;
end;

procedure TFrmset_Asisten.PageControl1Change(Sender: TObject);
begin
  if PageControl1.ActivePage = TabSheet2 then
    DhNavigator1.VisibleButtons :=
[nbFirst,nbPrior,nbNext,nbLast,nbDelete,nbRefresh]
  else
```

Listing Program

```
DbNavigator1.VisibleButtons :=  
[nbFirst,nbPrior,nbNext,nbLast,nbInsert,nbDelete,nbPost,nbCancel,nbRefresh];  
end;
```

```
procedure TFrmset_Asisten.tb_asistenAfterScroll(DataSet: TDataSet);  
begin  
  dbLookupListBox1.KeyValue := tb_asisten.fieldbyname('Stambuk').AsString;  
  Edit1.Text := tb_asisten.fieldbyname('kode_asisten').AsString;  
end;  
end.
```

{ Praktikan }

```
unit set_praktikan;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,  
ExtCtrls, DBCtrls, StdCtrls, Buttons, Grids, DBGrids, Db, Mask, DBTables,  
ComCtrls;
```

```
type
```

```
TFrmset_praktikan = class(TForm)
```

```
  Label1: TLabel; Label2: TLabel; PageControl1: TPageControl;
```

```
  TabSheet1: TTabSheet; TabSheet2: TTabSheet; tb_praktikan: TTable;
```

```
  Label3: TLabel; ds_praktikan: TDataSource; Label4: TLabel;
```

```
  tb_mahasiswa: TTable; tb_mahasiswaStambuk: TStringField;
```

```
  tb_mahasiswaNama: TStringField; tb_mahasiswaJurusan: TStringField;
```

```
  tb_mahasiswaAlamat: TStringField; tb_mahasiswaTanggal_Lahir: TDateField;
```

```
  Label6: TLabel; ds_mahasiswa: TDataSource; Label7: TLabel;
```

```
  DBGrid1: TDBGrid; BitBtn1: TBitBtn; DBNavigator1: TDBNavigator;
```

```
  DBLookupListBox1: TDBLookupListBox; tb_mk_praktek: TTable;
```

```
  ds_mk_praktek: TDataSource; tb_mk_praktekKode_mk_praktek: TStringField;
```

```
  tb_mk_praktekKode_mata_kuliah: TStringField;
```

```
  DBLookupComboBox1: TDBLookupComboBox; Edit1: TEdit; Label5: TLabel;
```

```
  tb_matkul: TTable; ds_matkul: TDataSource;
```

```
  tb_matkulKode_mata_kuliah: TStringField;
```

```
  tb_matkulNama_mata_kuliah: TStringField; tb_matkulSks: TFloatField;
```

```
  DBText1: TDBText; DBText2: TDBText; DBText3: TDBText;
```

```
  tb_praktikanKode_Praktikan: TStringField;
```

```
  tb_praktikanStambuk: TStringField; tb_praktikanNo: TAutoIncField;
```

```
  tb_praktikanNama: TStringField;
```

```
  procedure BitBtn1Click(Sender: TObject);
```

```
  procedure DBLookupComboBox1Click(Sender: TObject);
```

```
  procedure tb_praktikanFilterRecord(DataSet: TDataSet;
```

```
    var Accept: Boolean);
```

Listing Program

```
procedure tb_praktikanAfterPost(DataSet: TDataSet);
procedure tb_praktikanBeforePost(DataSet: TDataSet);
procedure tb_praktikanAfterInsert(DataSet: TDataSet);
procedure EditIChange(Sender: TObject);
procedure tb_praktikanNewRecord(DataSet: TDataSet);
procedure DBLookupListBox1Exit(Sender: TObject);
procedure PageControl1Change(Sender: TObject);
procedure tb_praktikanAfterScroll(DataSet: TDataSet);
private
public
end;

var
  Frmset_praktikan: TFrmset_praktikan;
  nm_terpilih, stb_terpilih : string;

implementation
procedure TFrmset_praktikan.BitBtn1Click(Sender: TObject);
begin
  application.Terminate;
end;

procedure TFrmset_praktikan.DBLookupComboBox1Click(Sender: TObject);
var
  s_pos, s_urut : string;
  pjg, pos, kode, n_urut : integer;
begin
  case TB_Praktikan.State of
  dsinsert : begin
    Tb_praktikan.last;
    s_urut := trim(copy(Tb_praktikanKode_praktikan.Value,5,4));
    val(s_urut,n_urut,kode);
    pos := n_urut + 1;
    s_pos := intToStr(pos);
    pjg := Length(s_pos);
    if
tb_praktikan.Locate('stambuk;kode_praktikan',VarArrayOf([stb_terpilih,'P'+dbLookup1
pComboBox1.Text]),[loPartialKey]) then
      begin
        if MessageDlg('Mahasiswa '+stb_terpilih+' sudah pernah ambil mata
kuliah praktek tsb'#13'Batalan' saja ?',
          mtConfirmation, [mbYes, mbNo], 0) = mrYes then
          exit;
        end;
      end;
  end;
```

```
tb_praktikan.insert;
case Pjg of
  1 : edit1.Text := 'P'+dbLookupComboBox1.Text+'000'+s_pos;
  2 : edit1.Text := 'P'+dbLookupComboBox1.Text+'00'+s_pos;
  3 : edit1.Text := 'P'+dbLookupComboBox1.Text+'0'+s_pos;
end;
tb_praktikan.FieldName('Kode_praktikan').AsString := Edit1.Text;
tb_praktikan.FieldName('stambuk').AsString := Stb_Terpilih;
dbLookupListBox1.KeyValue :=
tb_praktikan.fieldbyname('Stambuk').AsString;
end;
dsBrowse : begin
  tb_praktikan.last;
end;
end;
end;

procedure TFrmset_praktikan.tb_praktikanFilterRecord(DataSet: TDataSet;
var Accept: Boolean);
begin
  Accept :=
copy(Tb_PraktikanKode_Praktikan.Value,2,3)=dbLookupCComboBox1.Text;
end;

procedure TFrmset_praktikan.tb_praktikanArterPost(DataSet: TDataSet);
begin
  dbLookupListBox1.KeyValue := tb_praktikan.fieldbyname('Stambuk').AsString;
end;

procedure TFrmset_praktikan.tb_praktikanBeforePost(DataSet: TDataSet);
begin
  tb_praktikan.fieldByName('Kode_Praktikan').AsString := Edit1.Text;
  tb_praktikan.FieldName('Nama').AsString := nm_terpilih;
end;

procedure TFrmset_praktikan.tb_praktikanAfterInsert(DataSet: TDataSet);
begin
  edit1.Text := "";
end;

procedure TFrmset_praktikan.Edit1Change(Sender: TObject);
begin
  tb_praktikan.Last;
  cari := tb_mk_praktek.fieldbyname('kode_mata_kuliah').AsString;
```

```
if not tb_matkul.FindKey([cari]) then
  showmessage('tidak ditemukan');

s_urut := copy(Tb_praktikanKode_praktikan.Value,5,4);
val(s_urut,n_urut,kode);
pos := n_urut + 1;
s_pos := intToStr(pos);
Edit1.Text := 'P'+dbLookupComboBox1.Text+s_pos;
tb_praktikan.edit;}
end;

procedure TFrmset_praktikan.DBLookupListBox1Exit(Sender: TObject);
begin
  stb_terpilih := DbLookupListBox1.SelectedItem;
  nm_terpilih := tb_mahasiswa.FieldByName('nama').AsString;
end;

procedure TFrmset_praktikan.PageControl1Change(Sender: TObject);
begin
  if PageControl1.ActivePage = TabSheet2 then
    DbNavigator1.VisibleButtons :=
[nbFirst,nbPrior,nbNext,nbLast,nbDelete,nbRefresh]
  else
    DbNavigator1.VisibleButtons :=
[nbFirst,nbPrior,nbNext,nbLast,nbInsert,nbDelete,nbPost,nbCancel,nbRefresh];
end;

procedure TFrmset_praktikan.tb_praktikanAfterScroll(DataSet: TDataSet);
begin
  dbLookupListbox1.KeyValue := tb_praktikan.fieldbyname('Stambuk').AsString;
  Edit1.Text := tb_praktikan.fieldbyname('kode_praktikan').AsString;
end;
end.

{ Aktivitas }
unit set_aktivitas_1;
interface
uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  Grids, DBGrids, Db, DBTables, ExtCtrls, StdCtrls, DBCtrls, Buttons;

type
  Tfrm_aktivitas = class(TForm)
    Panel1: TPanel; Panel2: TPanel; Panel3: TPanel;
```



```
DBNavigator1: TDBNavigator; BitBtn1: TBitBtn; DBGrid1: TDBGrid;
dsAktivitas: TDataSource; tb_aktivitas: TTable; ds_praktek: TDataSource;
tb_praktek: TTable; tb_aktivitasKode_mk_praktek: TStringField;
tb_aktivitasL.Kode_mk_praktek: TStringField; ds_praktikan: TDataSource;
tb_praktikan: TTable; tb_aktivitasKode_praktikan: TStringField;
tb_aktivitasLKode_Praktikan: TStringField; ds_asisten: TDataSource;
tb_asisten: TTable; tb_aktivitasKode_asisten: TStringField;
tb_aktivitasLKodeAsisten: TStringField; ds_jadwal: TDataSource;
tb_jadwal: TTable; tb_aktivitasKd_Waktu: TStringField;
tb_aktivitaslih_kd_waktu: TStringField;
tb_aktivitasTahun_akademik: TStringField;
procedure BitBtn1Click(Sender: TObject);
procedure FormActivate(Sender: TObject);
procedure FormClose(Sender: TObject; var Action: TCloseAction);
procedure tb_asistenFilterRecord(DataSet: TDataSet;
  var Accept: Boolean);
procedure tb_praktikanFilterRecord(DataSet: TDataSet;
  var Accept: Boolean);
procedure DBGrid1DBClick(Sender: TObject);
private
public
end;
var
  frm_aktivitas: Tfrm_aktivitas;
  Kembali : string;

implementation
uses pil_sem;

procedure Tfrm_aktivitas.tb_asistenFilterRecord(DataSet: TDataSet;
  var Accept: Boolean);
begin
  Accept :=
copy(tb_asisten.fieldbyname('Kode_Asisten').AsString,2,3)=tb_aktivitas.fieldbyname
('kode_mk_praktek').AsString
end;

procedure Tfrm_aktivitas.tb_praktikanFilterRecord(DataSet: TDataSet;
  var Accept: Boolean);
begin
  Accept :=
trim(copy(tb_praktikan.fieldbyname('Kode_Praktikan').AsString,2,3))=trim(tb_aktivi
tas.fieldbyname('kode_mk_praktek').AsString)
end;
```



```
procedure Tfrm_aktivitas.DBGrid1DbClick(Sender: TObject);
begin
  if DbGrid1.SelectedIndex = 7 then
  begin
    frm_pil_sem.showmodal;
    Tb_Aktivitas.Edit;
    Tb_Aktivitas.FieldName('Tahun_akademik').AsString := Kembali;
  end;
end;
end.
```

{ Pilih Semester }

```
unit pil_sem;
interface
uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  StdCtrls, ComCtrls, Mask;
```

```
type
  Tfrm_pil_sem = class(TForm)
    GroupBox1: TGroupBox; RadioButton1: TRadioButton;
    RadioButton2: TRadioButton; Edit1: TEdit; UpDown1: TUpDown;
    Button1: TButton; Label1: TLabel; Label2: TLabel;
    procedure Button1Click(Sender: TObject);
    procedure FormActivate(Sender: TObject);
    procedure Edit1Change(Sender: TObject);
  private
  public
  end;
```

```
var
  frm_pil_sem: Tfrm_pil_sem;
  tahun : string;
  Int_tahun : integer;
```

```
implementation
uses set_aktivitas_1;
procedure Tfrm_pil_sem.Button1Click(Sender: TObject);
begin
  Kembali := Label2.Caption;
  close;
end;
```

```
procedure Tfrm_pil_sem.FormActivate(Sender: TObject);
var
  tgl,bln,thn : word;
begin
  DecodeDate(Now,thn,bln,tgl);
  Edit1.Text := trim(copy(IntToStr(thn),1,1)+'.'+copy(IntToStr(thn),2,3));
end;
```

```
procedure Tfrm_pil_sem.Edit1Change(Sender: TObject);
var
  thn2 : integer;
begin
  tahun := copy(edit1.text,1,1)+copy(edit1.text,3,3);
  thn2 := StrToInt(tahun)+1;
  if Radiobutton1.Checked then
    Label2.Caption := 'Awal '+tahun+'/' + IntToStr(thn2)
  else
    Label2.Caption := 'Akhir '+tahun+'/' + IntToStr(thn2);
end;
end.
```

{ Laporan Praktikan }

unit lap_praktikan;

interface

uses Windows, SysUtils, Messages, Classes, Graphics, Controls,
StdCtrls, ExtCtrls, Forms, Quickrpt, QRCtrls, Db, DBTables;

type

TQr_Praktikan = class(TQuickRep)

TitleBand1: TQRBand; ColumnHeaderBand1: TQRBand;

DetailBand1: TQRBand; SummaryBand1: TQRBand; QRLabel1: TQRLabel;

ds_praktikan: TDataSource; tb_praktikan: TTable; tb_mahasiswa: TTable;

ds_mahasiswa: TDataSource; QRDBText1: TQRDBText;

QRDBText2: TQRDBText; QRDBText3: TQRDBText;

QRDBText4: TQRDBText; QRDBText5: TQRDBText;

QRDBText6: TQRDBText; QRLabel2: TQRLabel;

QRLabel3: TQRLabel; QRSysData1: TQRSysData;

QRLabel4: TQRLabel; QRLabel5: TQRLabel; QRLabel6: TQRLabel;

QRLabel7: TQRLabel; QRLabel8: TQRLabel; QRSysData2: TQRSysData;

private

public

end;

```
var
  Qr_Praktikan: TQr_Praktikan;
implementation
end.

{ Laporan Nilai Praktikan }
unit lap_Nilai_Praktikan;
interface
uses Windows, SysUtils, Messages, Classes, Graphics, Controls,
  StdCtrls, ExtCtrls, Forms, Quickrpt, QRCtrls, Db, DBTables;

type
  TQr_Info_Nilai = class(TQuickRep)
    TitleBand1: TQRBand; ColumnHeaderBand1: TQRBand;
    DetailBand1: TQRBand; SummaryBand1: TQRBand; tb_aktivitas: TTable;
    ds_aktivitas: TDataSource; ds_praktikan: TDataSource; tb_praktikan: TTable;
    tb_mk_prak: TTable; ds_mk_prak: TDataSource; QRLabel1: TQRLabel;
    QRLabel2: TQRLabel; QRLabel3: TQRLabel; QRLabel4: TQRLabel;
    QRLabel5: TQRLabel; QRLabel6: TQRLabel; QRLabel7: TQRLabel;
    QRDBText1: TQRDBText; QRDBText2: TQRDBText;
    QRDBText3: TQRDBText; QRDBText4: TQRDBText;
    QRDBText5: TQRDBText; QRDBText6: TQRDBText;
    tb_aktivitasKode_mk_praktek: TStringField;
    tb_aktivitasKode_praktikan: TStringField;
    tb_aktivitasKode_asisten: TStringField; tb_aktivitasKd_Waktu: TStringField;
    tb_aktivitasTahun_akademik: TStringField;
    tb_aktivitasPrak1: TStringField; tb_aktivitasPrak2: TStringField;
    tb_aktivitasPrak3: TStringField; tb_aktivitasPrak4: TStringField;
    tb_aktivitasPrak5: TStringField; tb_aktivitasPrak6: TStringField;
    tb_aktivitasPrak7: TStringField; tb_aktivitasPrak8: TStringField;
    tb_aktivitasPrak9: TStringField; tb_aktivitasPrak10: TStringField;
    tb_aktivitasPrak11: TStringField; tb_aktivitasPrak12: TStringField;
    tb_aktivitasPrak13: TStringField; tb_aktivitasPrak14: TStringField;
    tb_aktivitasPrak15: TStringField; tb_aktivitasNilai: TFloatField;
    tb_praktikanNo: TAutoIncField; tb_praktikanKode_Praktikan: TStringField;
    tb_praktikanStambuk: TStringField; tb_praktikanNama: TStringField;
    tb_mk_prakKode_mk_praktek: TStringField;
    tb_mk_prakKode_mata_kuliah: TStringField;
    tb_mk_prakMata_kuliah: TStringField; QRSysData1: TQRSysData;
    QRLabel8: TQRLabel; QRSysData2: TQRSysData;
  private
  public
end;
```



```
var
  Qr_Info_Nilai: TQr_Info_Nilai;

implementation
end.

{ Laporan Asisten }
unit lap_asisten;
interface

uses Windows, SysUtils, Messages, Classes, Graphics, Controls,
  StdCtrls, ExtCtrls, Forms, Quickrpt, QRCtrls, Db, DBTables;

type
  TQr_info_asisten = class(TQuickRep)
    Qr_info_asisten: TQRBand; ColumnHeaderBand1: TQRBand;
    DetailBand1: TQRBand; SummaryBand1: TQRBand; QRLabel1: TQRLabel;
    tb_asisten: TTable; ds_asisten: TDataSource; tb_mahasiswa: TTable;
    ds_mahasiswa: TDataSource; QRSysData1: TQRSysData;
    QRLabel2: TQRLabel; QRLabel3: TQRLabel; QRLabel4: TQRLabel;
    QRDBText1: TQRDBText; QRDBText2: TQRDBText;
    QRDBText3: TQRDBText; QRDBText4: TQRDBText;
    QRDBText5: TQRDBText; QRDBText6: TQRDBText;
    QRLabel5: TQRLabel; QRLabel6: TQRLabel;
    QRLabel7: TQRLabel; QRLabel8: TQRLabel; QRSysData2: TQRSysData;
  private
  public
  end;

var
  Qr_info_asisten: TQr_info_asisten;
implementation
end.

{ Laporan Aktivitas }
unit lap_aktivitas;
interface

uses Windows, SysUtils, Messages, Classes, Graphics, Controls,
  StdCtrls, ExtCtrls, Forms, Quickrpt, QRCtrls, Db, DBTables;

type
  TQr_Info_Aktivitas = class(TQuickRep)
    TitleBand1: TQRBand; ColumnHeaderBand1: TQRBand;
    DetailBand1: TQRBand; SummaryBand1: TQRBand; tb_aktivitas: TTable;
  end;
```

Listing Program

```
ds_aktivitas: TDataSource; tb_praktikan: TTable; ds_praktikan: TDataSource;
ds_asisten: TDataSource; tb_asisten: TTable; QRLabel1: TQRLabel;
QRLabel2: TQRLabel; QRLabel3: TQRLabel; QRLabel4: TQRLabel;
QRLabel5: TQRLabel; QRLabel6: TQRLabel; QRLabel7: TQRLabel;
QRDBText1: TQRDBText; QRDBText2: TQRDBText;
QRDBText3: TQRDBText; QRDBText4: TQRDBText;
QRDBText5: TQRDBText; QRDBText6: TQRDBText;
QRSysData1: TQRSysData; QRLabel8: TQRLabel;
QRSysData2: TQRSysData;
private
public
end;
var
  Qr_Info_Aktivitas: TQr_Info_Aktivitas;
implementation
end

{ Laporan Jadwal }
unit lap_Jadwal;

interface
uses Windows, SysUtils, Messages, Classes, Graphics, Controls,
  StdCtrls, ExtCtrls, Forms, Quickrpt, QRCtrls, Db, DBTables;

type
  TQr_Info_Jadwal = class(TQuickRep)
  TitleBand1: TQRBand; ColumnHeaderBand1: TQRBand;
  DetailBand1: TQRBand; SummaryBand1: TQRBand;
  tb_aktivitas: TTable; ds_aktivitas: TDataSource; tb_jadwal: TTable;
  ds_jadwal: TDataSource; tb_mk_prak: TTable; ds_mk_prak: TDataSource;
  QRLabel1: TQRLabel; QRLabel2: TQRLabel; QRLabel3: TQRLabel;
  QRLabel4: TQRLabel; QRDBText1: TQRDBText;
  QRDBText2: TQRDBText; QRDBText3: TQRDBText;
  QRLabel5: TQRLabel; QRDBText4: TQRDBText;
  QRSysData1: TQRSysData; QRLabel6: TQRLabel; QRSysData2: TQRSysData;
private
public
end;

var
  Qr_Info_Jadwal: TQr_Info_Jadwal;
implementation
end.
```

```
{ Absensi }
unit absensi;
interface
uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  StdCtrls, Buttons, Grids, DBGrids, DBCtrls, Db, DBTables, ExtCtrls;

type
  Tfrm_absensi1 = class(TForm)
    Panel1: TPanel; Panel2: TPanel; Panel3: TPanel; Label1: TLabel;
    tb_absensi: TTable; ds_absensi: TDataSource; DBNavigator1: TDBNavigator;
    DBGrid1: TDBGrid; BitBtn1: TBitBtn; tb_aktivitas: TTable;
    ds_aktivitas: TDataSource; tb_aktivitasKode_mk_praktek: TStringField;
    tb_aktivitasKode_praktikan: TStringField; tb_aktivitasKode_asisten: TStringField;
    tb_aktivitasKd_Waktu: TStringField; tb_aktivitasTahun_akademik: TStringField;
    tb_aktivitasPrak1: TStringField; tb_aktivitasPrak2: TStringField;
    tb_aktivitasPrak3: TStringField; tb_aktivitasPrak4: TStringField;
    tb_aktivitasPrak5: TStringField; tb_aktivitasPrak6: TStringField;
    tb_aktivitasPrak7: TStringField; tb_aktivitasPrak8: TStringField;
    tb_aktivitasPrak9: TStringField; tb_aktivitasPrak10: TStringField;
    tb_aktivitasPrak11: TStringField; tb_aktivitasPrak12: TStringField;
    tb_aktivitasPrak13: TStringField; tb_aktivitasPrak14: TStringField;
    tb_aktivitasPrak15: TStringField; ds_praktikan: TDataSource;
    lb_praktikan: TTable; tb_aktivitasLih_praktikan: TStringField;
    DBLookupComboBox1: TDBLookupComboBox;
    ds_jadwal: TDataSource; tb_jadwal: TTable; Label2: TLabel;
    ds_mk_praktek: TDataSource; tb_mk_praktek: TTable;
    tb_aktivitasLih_matkul: TStringField;
    procedure BitBtn1Click(Sender: TObject);
    procedure tb_aktivitasFilterRecord(DataSet: TDataSet;
      var Accept: Boolean);
    procedure DBLookupComboBox1Click(Sender: TObject);
  private
  public
  end;

var
  frm_absensi1: Tfrm_absensi1;

implementation
procedure Tfrm_absensi1.BitBtn1Click(Sender: TObject);
begin
  close;
end;
```

Listing Program

```
procedure Tfrm_absensi.tb_aktivitasFilterRecord(DataSet: TDataSet;  
  var Accept: Boolean);  
begin  
  Accept := uppercase(Tb_Aktivitas.FieldName('Kd_waktu').AsString) =  
  DBLookupComboBox1.Text;  
end;
```

```
procedure Tfrm_absensi.DBLookupComboBox1Click(Sender: TObject);  
begin  
  Tb_aktivitas.First;  
end;  
end.
```

{ Input Nilai }

unit inp_nilai;

interface

uses

Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
StdCtrls, Buttons, Grids, DBGrids, DBCtrls, Db, DBTables, ExtCtrls;

type

```
Tfrm_inp_nilai = class(TForm)  
  Panel1: TPanel; Panel2: TPanel; Panel3: TPanel; Label1: TLabel;  
  tb_absensi: TTable; ds_absensi: TDataSource; DBNavigator1: TDBNavigator;  
  DBGrid1: TDBGrid; BitBtn1: TBitBtn; tb_aktivitas: TTable;  
  ds_aktivitas: TDataSource; tb_aktivitasKode_mk_praktek: TStringField;  
  tb_aktivitasKode_praktikan: TStringField; tb_aktivitasKode_asisten: TStringField;  
  tb_aktivitasKd_Waktu: TStringField; tb_aktivitasTahun_akademik: TStringField;  
  tb_aktivitasPrak1: TStringField; tb_aktivitasPrak2: TStringField;  
  tb_aktivitasPrak3: TStringField; tb_aktivitasPrak4: TStringField;  
  tb_aktivitasPrak5: TStringField; tb_aktivitasPrak6: TStringField;  
  tb_aktivitasPrak7: TStringField; tb_aktivitasPrak8: TStringField;  
  tb_aktivitasPrak9: TStringField; tb_aktivitasPrak10: TStringField;  
  tb_aktivitasPrak11: TStringField; tb_aktivitasPrak12: TStringField;  
  tb_aktivitasPrak13: TStringField; tb_aktivitasPrak14: TStringField;  
  tb_aktivitasPrak15: TStringField; ds_praktikan: TDataSource;  
  tb_praktikan: TTable; tb_aktivitasLih_praktikan: TStringField;  
  DBLookupComboBox1: TDBLookupComboBox;  
  ds_jadwal: TDataSource; tb_jadwal: TTable; Label2: TLabel;  
  ds_mk_praktek: TDataSource; tb_mk_praktek: TTable;  
  tb_aktivitasLih_matkul: TStringField;  
  tb_aktivitasNilai: TFloatField; procedure BitBtn1Click(Sender: TObject);  
  procedure tb_aktivitasFilterRecord(DataSet: TDataSet;
```




```
var Accept: Boolean);  
procedure DBLookupComboBox1Click(Sender: TObject);  
private  
public  
end;
```

```
var  
  Frm_inp_nilai: TFrm_inp_nilai;
```

```
implementation  
procedure TFrm_inp_nilai.BitBtn1Click(Sender: TObject);  
begin  
  close;  
end;
```

```
procedure TFrm_inp_nilai.tb_aktivitasFilterRecord(DataSet: TDataSet;  
  var Accept: Boolean);  
begin  
  Accept := uppercase(Tb_Aktivitas.FieldName('Kd_waktu').AsString) =  
  DBLookupComboBox1.Text;  
end;
```

```
procedure TFrm_inp_nilai.DBLookupComboBox1Click(Sender: TObject);  
begin  
  Tb_aktivitas First;  
end;  
end.
```

{ Info Mahasiswa }

```
unit info_mahasiswa;
```

```
interface
```

```
uses
```

```
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,  
  DBCtrls, Db, DBTables, Grids, DBGrids, FxtCtrls, StdCtrls;
```

```
type
```

```
TFrm_info_mahasiswa = class(TForm)  
  Panel1: Tpanel; Panel2: TPanel; Panel3: TPanel; DBGrid1: TDBGrid;  
  tb_mahasiswa: TTable; ds_mahasiswa: TDataSource; tb_praktikan: TTable;  
  ds_praktikan: TDataSource; DBNavigator1: TDBNavigator;  
  DBGrid2: TDBGrid; RadioGroup1: TRadioGroup;  
  DBLookupComboBox1: TDBLookupComboBox;  
  DBLookupComboBox2: TDBLookupComboBox; Label1: TLabel;
```

Listing Program

```
Label2: TLabel; procedure RadioGroup1Click(Sender: TObject);  
private  
public  
end;
```

```
var  
frm_info_mahasiswa: Tfrm_info_mahasiswa;  
implementation
```

```
procedure Tfrm_info_mahasiswa.RadioGroup1Click(Sender: TObject);  
begin  
case RadioGroup1.ItemIndex of  
0 : Tb_Praktikan.IndexFieldNames := 'Stambuk';  
1 : Tb_Praktikan.IndexFieldNames := 'Kode_Praktikan';  
2 : Tb_Praktikan.IndexFieldNames := 'Nama';  
end;  
end;  
end.
```

{ Info Mata Kuliah Praktek }

```
unit info_matkul;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,  
DBCtrls, Grids, DBGrids, Db, DBTables, ExtCtrls, StdCtrls;
```

```
type
```

```
Tfrm_info_mkPraktek = class(TForm)
```

```
Panel1: TPanel; Panel2: TPanel; Panel3: TPanel; ds_matkul: TDataSource;
```

```
tb_matkul: TTable; DBNavigator1: TDBNavigator;
```

```
DataSource1: TDataSource; Table1: TTable; DBGrid2: TDBGrid;
```

```
DBGrid1: TDBGrid; RadioGroup1: TRadioGroup;
```

```
DBLookupComboBox1: TDBLookupComboBox; Label1: TLabel;
```

```
procedure RadioGroup1Click(Sender: TObject);
```

```
private
```

```
public
```

```
end;
```

```
var
```

```
frm_info_mkPraktek: Tfrm_info_mkPraktek;
```

```
implementation
```

```
procedure Tfrm_info_mkPraktek.RadioGroup1Click(Sender: TObject);  
begin
```

Listing Program

```
case RadioGroup1.ItemIndex of
0 : Table1.IndexFieldNames := 'Kode_Mk_Praktek';
1 : Table1.IndexFieldNames := 'Kode_Mata_kuliah';
2 : Table1.IndexFieldNames := 'Mata_Kuliah';
end;
end;
end.

{ Info Aktivitas }
unit info_aktivitas;
interface
uses
Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
Grids, DBGrids, DBCtrls, Db, DBTables, ExtCtrls, StdCtrls;

type
Tfrm_info_aktivitas = class(TForm)
Panel1: TPanel; Panel2: TPanel; Panel3: TPanel; tb_aktivitas: TTable;
ds_aktivitas: TDataSource; DBNavigator1: TDBNavigator; DBGrid1: TDBGrid;
DBGrid2: TDBGrid; DBGrid3: TDBGrid; tb_praktikan: TTable;
ds_praktikan: TDataSource; ds_asisten: TDataSource; tb_asisten: TTable;
RadioGroup1: TRadioGroup; procedure RadioGroup1Click(Sender: TObject);
private
public
end;

var
frm_info_aktivitas: Tfrm_info_aktivitas;
implementation

procedure Tfrm_info_aktivitas.RadioGroup1Click(Sender: TObject);
begin
case RadioGroup1.ItemIndex of
0 : Tb_Aktivitas.IndexFieldNames := 'Kode_Praktikan';
1 : Tb_Aktivitas.IndexFieldNames := 'Kode_Asisten';
2 : Tb_Aktivitas.IndexFieldNames := 'Kd_Waktu';
3 : Tb_Aktivitas.IndexFieldNames := 'Tahun_Akademik';
end;
end;
end.
```

Listing Program

```
{ Info Asisten }
```

```
unit info_asisten;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,  
DBCtrls, Db, DBTables, Grids, DBGrids, ExtCtrls, StdCtrls;
```

```
type
```

```
Tfm_info_asisten = class(TForm)
```

```
  Panel1: TPanel;
```

```
  Panel2: TPanel; Panel3: TPanel; DBGrid2: TDBGrid; ds_asisten: TDataSource;
```

```
  tb_asisten: TTable; ds_mahasiswa: TDataSource; tb_mahasiswa: TTable;
```

```
  DBGrid1: TDBGrid; DBNavigator1: TDBNavigator;
```

```
  RadioGroup1: TRadioGroup; DBLookupComboBox1: TDBLookupComboBox;
```

```
  DBLookupComboBox2: TDBLookupComboBox; Label1: TLabel;
```

```
  Label2: TLabel;
```

```
  procedure RadioGroup1Click(Sender: TObject);
```

```
private
```

```
public
```

```
end;
```

```
var
```

```
  fm_info_asisten: Tfm_info_asisten;
```

```
implementation
```

```
procedure Tfm_info_asisten.RadioGroup1Click(Sender: TObject);
```

```
begin
```

```
  case RadioGroup1.ItemIndex of
```

```
  0: Tb_Asisten.IndexFieldNames := 'Stambuk';
```

```
  1: Tb_Asisten.IndexFieldNames := 'Kode_Asisten';
```

```
  2: Tb_Asisten.IndexFieldNames := 'Nama';
```

```
  end;
```

```
end;
```

```
end;
```

```
{ Info Mata Kuliah }
```

```
unit lap_info_matakul;
```

```
interface
```

```
uses Windows, SysUtils, Messages, Classes, Graphics, Controls,  
  StdCtrls, ExtCtrls, Forms, QuickRep, DBCtrls, Db, DBTable;
```

```
type
```

```
  Tfp_info_matakul = class(TQuickRep)
```

```
TitleBand1: TQRBand; ColumnHeaderBand1: TQRBand;
DetailBand1: TQRBand; QRLabel1: TQRLabel; QRDBText1: TQRDBText;
QRLabel2: TQRLabel; QRLabel3: TQRLabel; QRDBText2: TQRDBText;
QRLabel4: TQRLabel; QRDBText3: TQRDBText;
Table1: TTable; DataSource1: TDataSource; QRDBText4: TQRDBText;
QRLabel5: TQRLabel; Table2: TTable, DataSource2: TDataSource;
TableIKode_mk_praktek: TStringField; TableIKode_mata_kuliah: TStringField;
TableIMata_kuliah: TStringField; SummaryBand1: TQRBand;
QRSysData1: TQRSysData; QRLabel6: TQRLabel; QRSysData2: TQRSysData;
private
public
end;

var
  qr_info_matkul: Tqr_info_matkul;
implementation
end.

{ Info Jadwal }
unit info_jadwal;
interface
uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  Grids, DBGrids, DBCtrls, Db, DBTables, ExtCtrls, StdCtrls;

type
  Tfrm_info_jadwal = class(TForm)
    Panel1: TPanel; Panel2: TPanel; Panel3: TPanel; tb_aktivitas: TTable;
    ds_aktivitas: TDataSource; tb_mk_prak: TTable; ds_mk_prak: TDataSource;
    tb_jadwal: TTable; ds_jadwal: TDataSource; DBNavigator1: TDBNavigator;
    DBGrid1: TDBGrid; DBGrid2: TDBGrid; DBGrid3: TDBGrid;
    RadioGroup1: TRadioGroup; procedure RadioGroup1Click(Sender: TObject);
  private
  public
  end;

var
  frm_info_jadwal: Tfrm_info_jadwal;
implementation

procedure Tfrm_info_jadwal.RadioGroup1Click(Sender: TObject);
begin
  case RadioGroup1.ItemIndex of
  0 : Tb_Aktivitas.IndexFieldNames := 'Kode_mk_praktek';
```

Listing Program



```
1 : Tb_Aktivitas.IndexFieldNames := 'Kd_waktu';
2 : Tb_Aktivitas.IndexFieldNames := 'Tahun_Akademik';
end;
end;
end.
```

{ Info Nilai }

```
unit info_nilai;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
DBCtrls, Grids, DBGrids, Db, DBTables, ExtCtrls, StdCtrls;
```

```
type
```

```
Tfrm_Nilai_Praktikan = class(TForm)
```

```
Panel1: TPanel; Panel2: TPanel; Panel3: TPanel; tb_aktivitas: TTable;
ds_aktivitas: TDataSource; tb_praktikan: TTable; ds_praktikan: TDataSource;
```

```
DBGrid1: TDBGrid; DBNavigator1: TDBNavigator; tb_mk_prak: TTable;
```

```
ds_mk_prak: TDataSource; DBGrid2: TDBGrid; DBGrid3: TDBGrid;
```

```
RadioGroup1: TRadioGroup; GroupBox1: TGroupBox;
```

```
DBLookupComboBox1: TDBLookupComboBox; Label1: TLabel;
```

```
procedure RadioGroup1Click(Sender: TObject);
```

```
private
```

```
public
```

```
end;
```

```
var
```

```
frm_Nilai_Praktikan: Tfrm_Nilai_Praktikan;
```

```
implementation
```

```
procedure Tfrm_Nilai_Praktikan.RadioGroup1Click(Sender: TObject);
```

```
begin
```

```
case RadioGroup1.ItemIndex of
```

```
0 : Tb_Aktivitas.IndexFieldNames := 'Kode_Praktikan';
```

```
1 : Tb_Aktivitas.IndexFieldNames := 'Kd_waktu';
```

```
end;
```

```
end;
```

```
end.
```

{ About }

```
unit about;
```

```
interface
```

```
uses
```

Listing Program

Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs, jpeg, ExtCtrls, StdCtrls, Buttons:

```
type
  TFrm_About = class(TForm)
    BitBtn1: TBitBtn;
    Label1: TLabel; Label2: TLabel; Memo1: TMemo; Image1: TImage;
  procedure BitBtn1Click(Sender: TObject);
  private
  public
  end;

var
  Frm_About: TFrm_About;

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

{ Program Utama }
uses
  Forms,
  utama in 'utama.pas' {Frm_utama},
  about in 'about.pas' {Frm_About},
  mata_kuliah in 'mata_kuliah.pas' {frmset_mku?},
  set_aktivitas_1 in 'set_aktivitas_1.pas' {frm_aktivitas},
  set_asisten in 'set_asisten.pas' {Frmset_Asisten},
  set_mahasiswa in 'set_mahasiswa.pas' {frm_mahasiswa},
  set_mkpraktek in 'set_mkpraktek.pas' {frmset_mkpraktek},
  set_praktikan in 'set_praktikan.pas' {Frmset_praktikan},
  set_jadwal in 'set_jadwal.pas' {Frm_Jadwal},
  pil_sem in 'pil_sem.pas' {frm_pil_sem},
  inp_nilai in 'inp_nilai.pas' {Frm_inp_nilai},
  absensi in 'absensi.pas' {frm_absensi},
  info_matkul in 'info_matkul.pas' {frm_info_mkPraktek},
  lap_info_matkul in 'lap_info_matkul.pas' {qr_Info_matkul: TQuickRep},
  info_mahasiswa in 'info_mahasiswa.pas' {frm_info_mahasiswa},
  lap_praktikan in 'lap_praktikan.pas' {Qr_Praktikan: TQuickRep},
  info_asisten in 'info_asisten.pas' {frm_info_asisten},
  lap_asisten in 'lap_asisten.pas' {Qr_info_asisten: TQuickRep},
  info_aktivitas in 'info_aktivitas.pas' {frm_info_aktivitas},
```

Listing Program

```
lap_aktivitas in 'lap_aktivitas.pas' {Qr_Info_Aktivitas: TQuickRep},  
info_nilai in 'info_nilai.pas' {frm_Nilai_Praktikan},  
lap_Nilai_Praktikan in 'lap_Nilai_Praktikan.pas' {Qr_Info_Nilai: TQuickRep},  
info_Jadwal in 'info_Jadwal.pas' {frm_info_jadwal},  
lap_Jadwal in 'lap_Jadwal.pas' {Qr_Info_Jadwal: TQuickRep};
```

begin

```
Application.Initialize;  
Application.CreateForm(TFrm_utama, Frm_utama);  
Application.CreateForm(Tfrm_info_aktivitas, frm_info_aktivitas);  
Application.CreateForm(Tfrm_Nilai_Praktikan, frm_Nilai_Praktikan);  
Application.CreateForm(Tfrm_info_asisten, frm_info_asisten);  
Application.CreateForm(Tfrm_info_mahasiswa, frm_info_mahasiswa);  
Application.CreateForm(TFrm_About, Frm_About);  
Application.CreateForm(Tfrmset_mkul, frmset_mkul);  
Application.CreateForm(Tfrm_aktivitas, frm_aktivitas);  
Application.CreateForm(TFrmset_Asisten, Frmset_Asisten);  
Application.CreateForm(Tfrm_mahasiswa, frm_mahasiswa);  
Application.CreateForm(Tfrmset_mkpraktek, frmset_mkpraktek);  
Application.CreateForm(TFrmset_praktikan, Frmset_praktikan);  
Application.CreateForm(TFrm_Jadwal1, Frm_Jadwal1);  
Application.CreateForm(Tfrm_pil_sem, frm_pil_sem);  
Application.CreateForm(TFrm_inp_nilai, Frm_inp_nilai),  
Application.CreateForm(Tfrm_absensi1, frm_absensi1);  
Application.CreateForm(Tfrm_info_mkPraktek, frm_info_mkPraktek);  
Application.CreateForm(Tqr_Info_matkul, qr_Info_matkul);  
Application.CreateForm(TQr_Praktikan, Qr_Praktikan);  
Application.CreateForm(TQr_info_asisten, Qr_info_asisten);  
Application.CreateForm(TQr_Info_Aktivitas, Qr_Info_Aktivitas);  
Application.CreateForm(TQr_Info_Nilai, Qr_Info_Nilai);  
Application.CreateForm(Tfrm_info_jadwal, frm_info_jadwal);  
Application.CreateForm(TQr_Info_Jadwal, Qr_Info_Jadwal);  
Application.Run;
```

end.