BAB III
KEGIATAN SELAMA KERJA PRAKTEK

UNIT PELAKSANA TEKNIS PERPUSTAKAAN UNIKOM

3.1 Jadwal Kerja Praktek

Kerja Praktek dilaksanakan mulai tanggal 26 Agustus 2004 hingga 30 September 2004, sehingga lamanya Kerja Praktek kurang lebih 1 (satu) bulan.

3.2 Cara Kerja Praktek

Selama melakukan Praktek Kerja Lapangan, penulis ditugaskan untuk membuat aplikasi *database* Daftar riwayat Hidup Pegawai. Untuk membuat aplikasi *database* tersebut, penulis menggunakan bahasa pemrograman Visual Basic 6.0.

3.3 Landasan Teori

3.3.1 Dasar teori

Sebuah *database* merupakan kumpulan – kumpulan informasi atau file-file yang tergabung dalam *database* yang di simpan dalam suatu media perangkat keras. *Database* sendiri di susun menurut kategorinya dalam bentuk tabel dan laporan di berikan kunci induk, sebagai media relasi antara tabel-tabel yang berisikan informasi, prinsip utamanya adalah pengaturan data atau arsip, sedangkan tujuan utamanya adalah kemudahan dan kecepatan dalam pengambilan data atau arsip. *Database* juga dapat di akses dengan sistem stand alone atau *client* server.

Basisdata bukan hanya sekedar penyimpanan data secara elektronis (dengan bantuan komputer) artinya tidak semua bentuk penyimpanan data secara elektronis dapat di sebut basis data, seperti media penyimpanan dokumen berisi data dalam file teks (dengan program pengolahan kata), file Spreadsheet dan lain-lain, tetapi tidak bisa disebut basisdata didalamnya tak ada pemilihan dan pengelompokan data sesuai jenis dan fungsi data sehingga akan menyulitkan pencarian data kelak.

3.3.2 Data dan cara pengolahan

Pengolahan data adalah serangkaian operasi atas informasi yang di rencanakan untuk mencapai hasil dan tujuan yang diinginkan. Menurut George R Terry, Ph.D: "Pengolahan data adalah serangkaian operasi atas informasi yang direncanakan guna mencapai tujuan atau hasil yang diinginkan". Selanjutnya di katakan bahwa ada 8 unsur pokok pengolahan data yaitu:

1. Reading (membaca)
2. Writing, typing, cardpanching or paper tape per forting (frequently called input) (menulis, mengetik, membuat lubang atau pons atau pada kartu sering disebut masukan)
3. Recording or printing (mencatat atau mencetak)
4. Sorting (mengurutkan)
5. Transmiting (menyampaikan)
6. Calculating (menghitung)
7. Comparing (membandingkan)

8. Storing (menyimpan)

3.3.3 *Database* (basis data)

Suatu informasi manajemen menggambarkan rangkaian data yang cukup lengkap yang disimpan agar cepat menyediakan informasi dan untuk mendukung operasi manajemen dan pengambilan keputusan.

Rangkaian file data yang saling berkaitan secara logis yang dipelihara dalam sistem informasi manajemen disebut *database* (basisdata). James Martin mendefinisikan : "adalah suatu kumpulan data terhubung (*Interrelated* Data) yang disimpan secara bersamaan pada suatu media penyimpanan, tanpa ada saling tumpang tindih satu sama lain dan tidak perlu terjadi kerangkapan data (*Controlled Redudancy*) dengan cara-cara tertentu sehingga mudah untuk digunakan atau disajikan kembali; dapat digunakan satu atau lebih program aplikasi secara optimal; data disimpan tanpa mengalami ketergantungan pada program yang akan menggunakannya; data disimpan sedemikian rupa sehingga penambahan, pengambilan dan modifikasi data dapat dilakukan dengan mudah dan terkendali".

Suatu *database* terdiri dari elemen-elemen yang saling diorganisasikan kedalam record-record dan file-file dengan berbagai cara dengan maksud untuk memenuhi kebutuhan informasi bagi pemakai/user. Elemen data tersebut secara keseluruhan disebut *database* (basisdata). Dalam perancangan *database* terdapat 2 pendekatan yaitu :

a. Pendekatan Aplikasi

Digunakan untuk keperluan merancang *database* suatu program aplikasi tertentu, sehingga untuk keperluan aplikasi lainnya harus disediakan penyimpanan data yang sesuai dengan aplikasi tersebut

Dalam hal ini terdapat kemungkinan terjadinya kerangkapan data, dimana suatu data disimpan dalam beberapa file, tidak ada format baku, kurang memperhatikan hubungan logik antara elemen-elemen data.

b. Pendekatan Sistem

Database yang dirancang diusahakan mengurangi kelemahan-kelemahan diatas. *Database* dapat digunakan oleh bermacam-macam pemakai dengan bermacam-macam aplikasinya. Keuntungan dari pendekatan sistem ini adalah:

1. Dapat mengurangi pendekatan data;
2. Dapat mengurangi kerangkapan data;
3. Pembakuan format dan kemungkinan pengembangan.

3.3.4 Model *database*

Pada suatu sistem *database* terdapat terdapat suatu cara khusus mengenai bagaimana suatu data diorganisasikan dengan suatu model dalam hal ini yang biasa disebut struktur data.

Model data merupakan struktur dari datbase management sistem. Model data tersebut menggambarkan entity-entity beserta hubungan dengan *database*. Model *database* sendiri dapat didefinisikan sebagai kumpulan perangkat konseptual/menggambarkan data, hubungan data, semantik (makna) data, dan

batasan data. Oleh karena yang ingin ditunjukkan adalah makna dari data dan hubungan dengan data lain, maka model data ini lebih tepat disebut model data logik.

Adapun sejumlah cara dalam merepresentasikan model data dalam perancangan basisdata yang secara umum dapat dibagi menjadi 2 kelompok yaitu:

1. Model logik data yang berdasarkan objek yang terdiri dari :
   a. Model ketergantungan entitas;
   b. Model berorientasi objek;
   c. Model data semantik;
   d. Model data fungsional.
2. Model logik data berdasarkan record yang terdiri dari :
   a. Model relasional;
   b. Model hirarkis;
   c. Model jaringan.

Namun demikian dari sekian cara representasi data diatas, hanya satu yang akan dibahas oleh penulis yaitu Model dat relasional yang merupakan model data yang paling populer dalam perancangan basisdata.

3.3.5 Perancangan basis data

Merancang *database* merupakan suatu hal yang sangat penting, kesulitan utama dalam merancang *database* adalah bagaimana merancang suatu *database* sehingga *database* tersebut dapat memenuhi keperluan informasi saat ini dan data yang akan datang.

Dengan perancangan model konseptual ini akan menunjukkan relasi dan entity berdasarkan proses yang diinginkan oleh suatu organisasi atau perusahaan. Pada saat menentukan entity dan relasi dibutuhkan analisis tentang informasi yang ada dalam spesifikasi dimasa mendatang.

Dengan pendekatan konseptual ini beberapa konsep pendekatan relasional digunakan namun tidak berarti konsep ini nantinya diimplementasikan ke model relasional saja, tetapi dapat saja dipakai dalam model jaringan dan hirarki.

3.3.6 Klasifikasi item data

Klasifikasi adalah proses identifikasi dan penempatan item data ke dalam kategori menurut sifat-sifat dan ciri-ciri tertentu sehingga memudahkan dalam memperoleh data-data tersebut. Klasifikasi item data yang dilakukan harus memenuhi kebutuhan pemakai dan memenuhi karakteristik operasi pengolahan dan penyimpanan.

Suatu pedoman yang dapat dilakukan untuk melakukan klasifikasi data adalah sebagai berikut :

1. Klasifikasi dilakukan memungkinkan perkembangan dan perluasan dalam penanganan item yang diidentifikasikan.
2. Klasifikasi dalam penyusunan item data harus logis.
3. Skema klasifikasi harus mencangkup kebutuhan yang luas. Bila data setelah diklasifikasikan, struktur klasifikasi telah siap diberi kode. Klasifikasi dan kode fikasi sangat penting dalam merancang suatu *database*. Kode dapat disusun dengan berbagai cara, kemudian dipilih secara seksama, tipe-tipe kode

yang akan digunakan masing-masing memiliki kekurangan dan kelebihan sehingga pemakai dapat mempertimbangkan penggunaan kode tersebut sesuai dengan kebutuhannya.

3.3.7 Perancangan *database* dengan teknik *relationship*

Database adalah kumpulan file-file yang saling terkait/berrelasi. Hubungan antar file direlasikan dengan kunci relasi yang merupakan kunci utama dari masing-masing file.

Relasi antar file/tabel dapat dikategorikan menjadi 3 macam golongan yang digunakan juga untuk menentukan kunci-kunci relasi dalam file-file tersebut yaitu:

1. One-to one (1-1)

   Hubungan ini merupakan arti bahwa setiap basisdata pada tabel pertama dihubungkan hanya kesatu baris data pada tabel kedua. Hal ini misalnya terdapat pada sebuah tabel berukuran besar yang dipecah menjadi dua tabel yang berukuran lebih kecil. Dimana kedua tabel memiliki tema yang sama, maka keduanya harus dihubungkan satu persatu (dalam hal ini fieldnya) secara klop dan tidak boleh meleset. Pembuatan relasi jenis ini ibarat menyambungkan potongan pipa yang terbelah menjadi 2.

2. One-to-many (1 - ~)

   Pada relasi ini setiap data dari tabel pertama dapat dihubungkan kesatu baris/lebih data pada tabel kedua. Pembuatan relasi jenis ini dapat

diibaratkan percabangan sebuah pipa, artinya sebuah pipa disambungkan dengan n dua buah pipa/lebih.

3. Many-to many (~ - ~)

Pada relasi ini berarti satu baris/lebih data pada tabel pertama dapat direlasikan kesatu baris/lebih data pada tabel kedua. Hal ini dapat diibaratkan seperti penggabungan 2 buah percabangan pipa.

3.3.8 *Database* Visual Basic

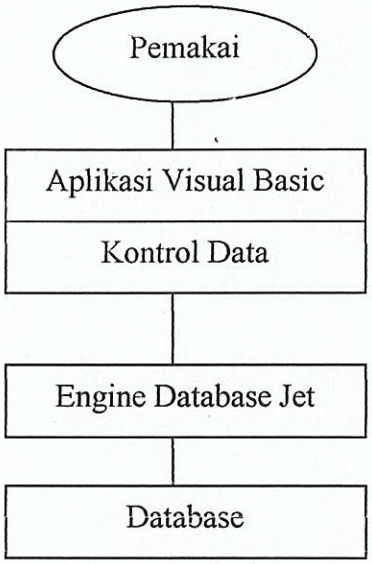
Sejak versi 3, Visuai basic sudah menyediakan tool untuk pemrograman *database*. Pertama adalah DAO *(data access object)* dalam versi 3, RDO *(remote data objects)* dalam versi 4, dan kemampuan untuk membuat komponen activeX pada versi 5.

Dalam manajemen *database*, aplikasi Visual Basic berperan sebagai *front-end* terhadap *database*, artinya aplikasi Visual Basic menyediakan antarmuka antara user dengan *database*.

Aplikasi Visual Basic tidak secara langsung berinteraksi dengan *database* karena ada dua komponen diantaranya yaitu kontrol data dengan engine *database*.

Kontrol data adalah objek Visual Basic yang menghubungkan aplikasi dengan *database* melalui engine *database*.

Engine *database* yang merupakan "jantung" dari sistem manajemen *database* Visual Basic adalah program yang mengelola informasi dalam *database*. Engine *database* yang digunakan MS Visual Basic dan juga MS Access adalah engine JET (Joint Engine Technology).



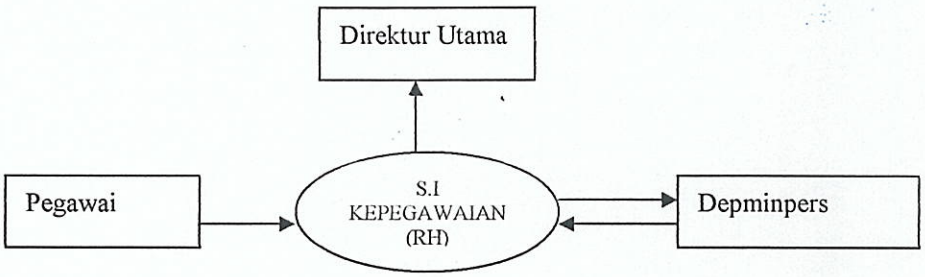
UNIT PELAKSANA TEKNIS PERPUSTAKAAN UNIKOM

Gambar 3.1. Komponen *Database* Visual Basic

3.4 Data Flow Diagram (DFD)

3.4.1 Diagram konteks/ DFD level 0

Diagram konteks ini digunakan untuk mendefinisikan serta memperlihatkan gambaran menyeluruh area study dan entitas luas yang berhubungan dengan aliran data. Diagram konteks ini akan memberikan masukan dan keluaran dari sistem sehingga data yang masuk dapat menjadi informasi sebagai keluaran yang telah melalui beberapa proses. Untuk lebih jelasnya dapat dilihat pada gambar 3.2.



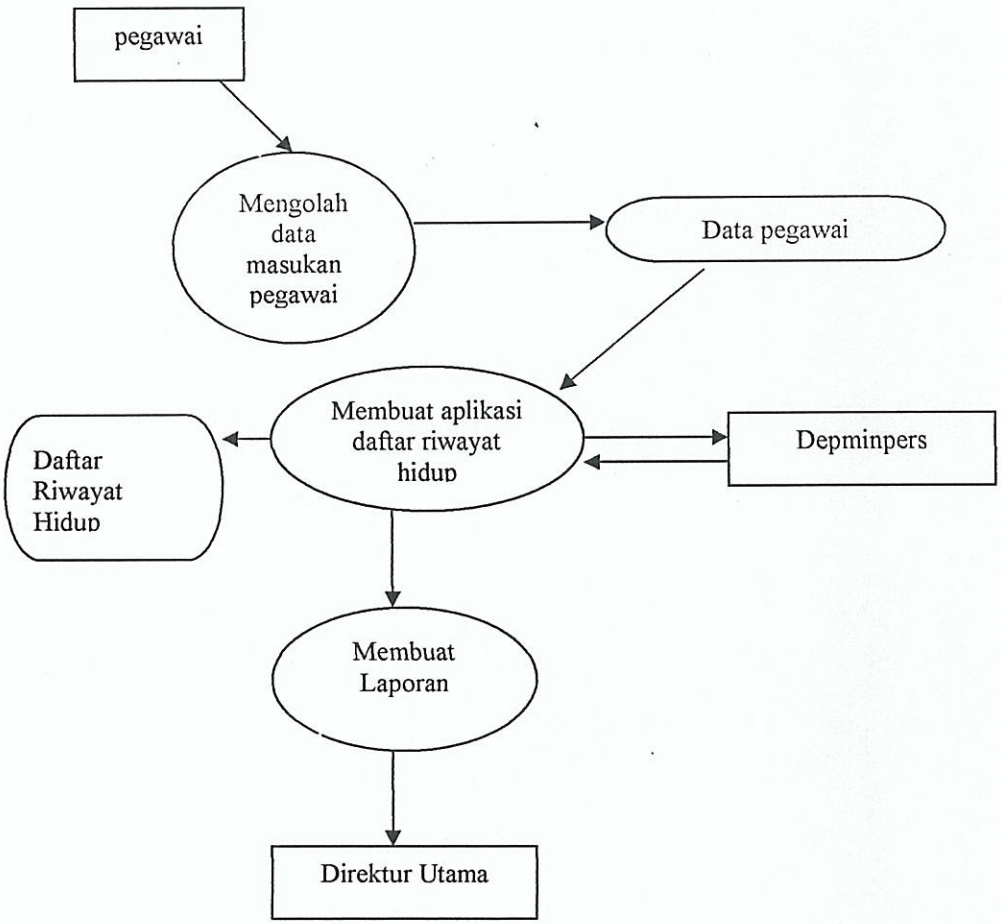
Gambar 3.2. DFD Level 0

Keterangan:

a. Data pegawai, kursus, pendidikan umum, pengalaman jabatan, penugasan, pengalaman lain, karya tulis, bahasa asing.
b. Data pegawai, kursus, pendidikan umum, pengalaman jabatan, penugasan, pengalaman lain, karya tulis, bahasa asing.
c. Aplikasi Daftar Riwayat hidup
d. Laporan

3.4.2 DFD level 1

Diagram alir data adalah suatu diagram yang menggambarkan arus data dan cara kerja sistem yang lebih detail. Diagram alir data ini menjelaskan alir data, proses dan sumber atau tujuan data yang dilakukan sistem. Untuk lebih jelasnya dapat dilihat pada gambar 3.3.



Gambar 3.3. DFD Level 1

Keterangan:

a. Biodata pegawai, kursus,pendidikan umum, pengalaman jabatan, penugasan, pengalaman lain, karya tulis, bahasa asing.

b. Data pegawai.

c. Data riwayat hidup yang belum diproses

d. Data riwayat hidup yang telah diproses

e. Data pegawai.

f. Laporan maintenance data personil

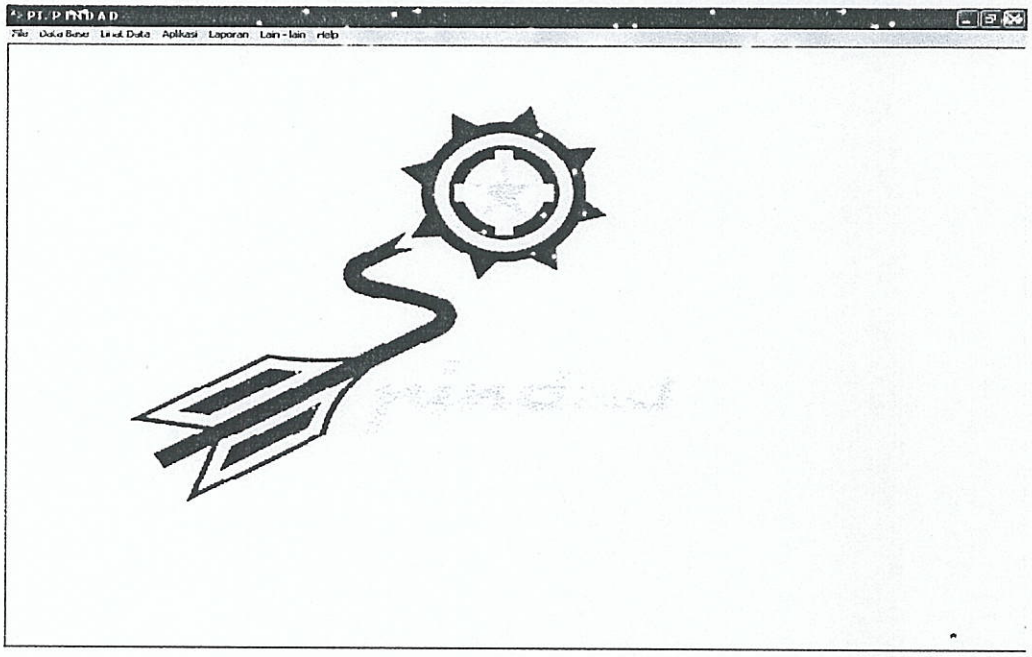
g. Daftar riwayat hidup

h. Laporan maintenance data personil

3.5 *Database* Daftar Riwayat Hidup Pegawai

3.5.1 Main menu

Terdiri dari menu pilihan File, *Database*, Lihat Data, Aplikasi, Laporan, Lain-lain, dan menu Help.



Gambar 3.4. Main Menu

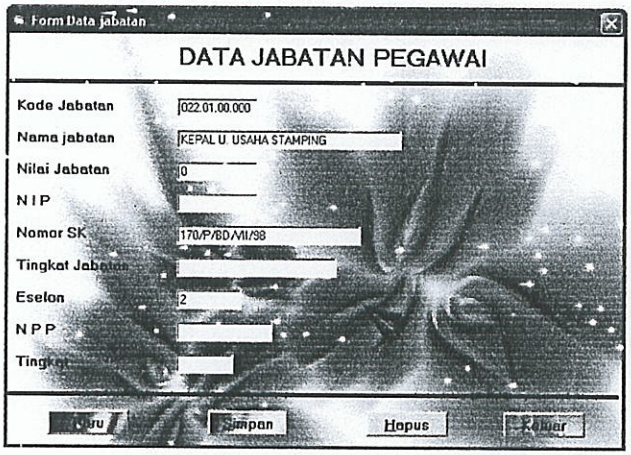
3.5.2 Menu *Database*

3.5.2.1 *Database* Jabatan Pegawai

Digunakan untuk mengisi Kode Jabatan, Nama Jabatan, Nilai Jabatan, NIP, No. SK, Tingkat Jabatan, Eselon, NPP, Tingkat.

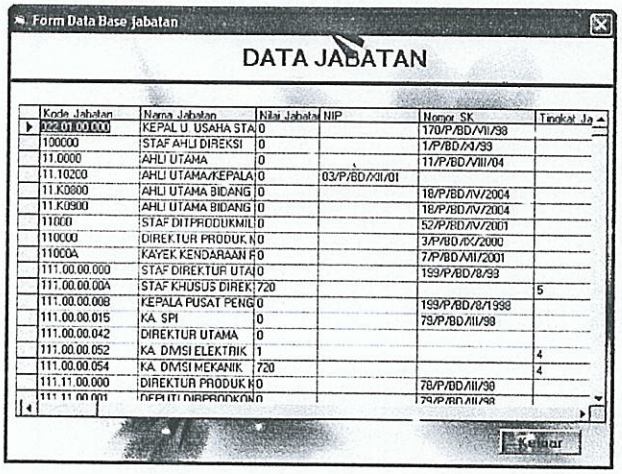
Tabel 3.1. Jabatan

Field Name	Data Type
Kode_Jabatan	Text
Nama_Jabatan	Text
Nilai_Jabatan	Text
NIP	Text
Nomor_SK	Text
Tingkat_Jabatan	Text
NPP	Text
Eselon	Text
Tingkat	Text



Gambar 3.5. Tampilan *Database* Jabatan Pegawai

Database Jabatan ini mengacu pada Data Jabatan yang berada pada menu Lihat data.



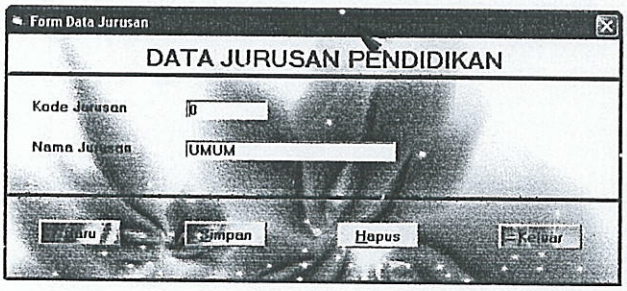
Gambar 3.6. Tampilan Data Jabatan

3.5.2.2 *Database* Jurusan

Digunakan untuk mengisi data jurusan pendidikan, yaitu Kode Jurusan dan Nama Jurusan.

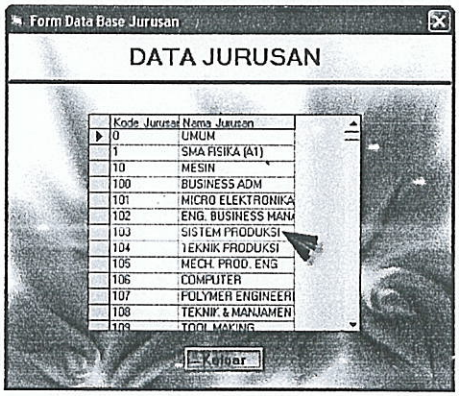
Tabel 3.2. Jurusan

Field Name	Data Type
Kode_Jurusan	Text
Nama_Jurusan	Text



Gambar 3.7. Tampilan *Database* Jurusan

Database ini mengacu pada Data Jurusan yang berada pada menu Lihat data.



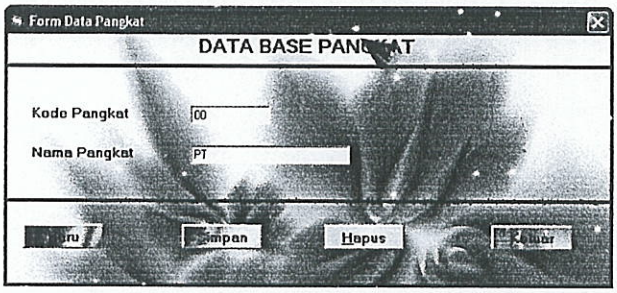
Gambar 3.8. Tampilan Data Jurusan

3.5.2.3 *Database* Pangkat

Digunakan untuk mengisi Kode Pangkat dan Nama Pangkat, yang mengacu ke Data Pangkat pada menu Lihat data.

Tabel 3.3. Pangkat

Field Name	Data Type
Kode_Pangkat	Text
Nama_Pangkat	Text



Gambar 3.9. Tampilan *Database* Pangkat



UNIT PELAKSANA TEKNIS PERPUSTAKAAN UNIKOM

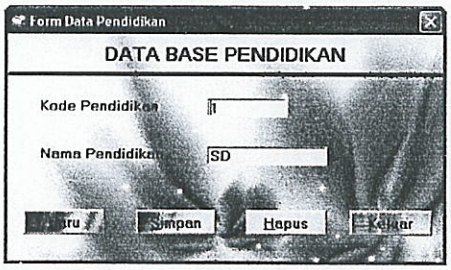
Gambar 3.10. Tampilan Data Pangkat

3.5.2.4 *Database* Pendidikan

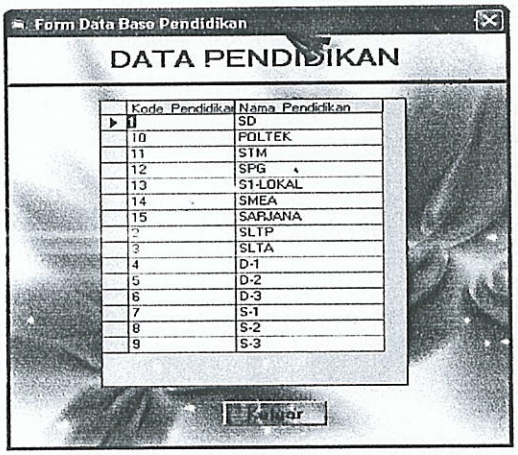
Database ini digunakan untuk mengisi data Kode Pendidikan dan Nama Pendidikan, yang mengacu pada Data Pendidikan pada menu Lihat Data.

Tabel 3.4. Pendidikan

Field Name	Data Type
Kode_Pendidikan	Text
Nama_Pendidikan	Text



Gambar 3.11. Tampilan *Database* Pendidikan



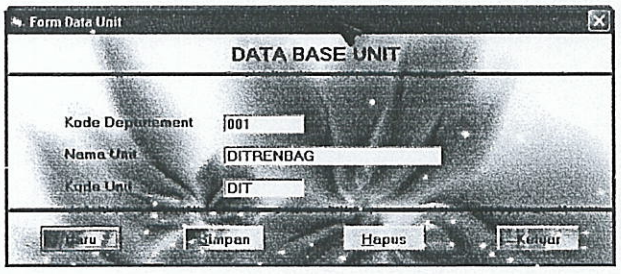
Gambar 3.12. Tampilan Data Pendidikan

3.5.2.5 *Database* Unit

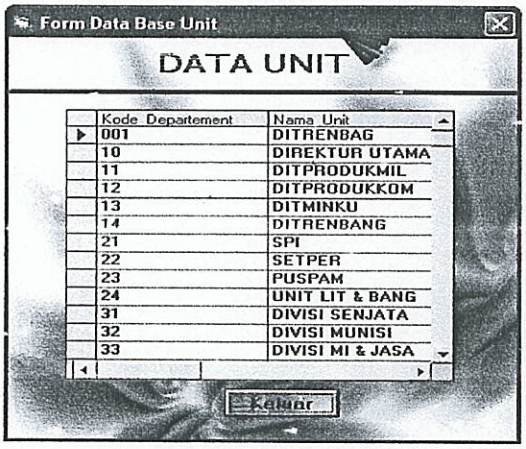
Database ini digunakan untuk mengisi data Kode Departement, Nama Unit dan Kode Unit, yang mengacu ke Data Unit pada menu Lihat Data.

Tabel 3.5. Unit

Field Name	Data Type
Kode_Departement	Text
Nama_Unit	Text
Kode_Unit	Text



Gambar 3.13. Tampilan *Database* Unit



Gambar 3.14. Tampilan Data Unit

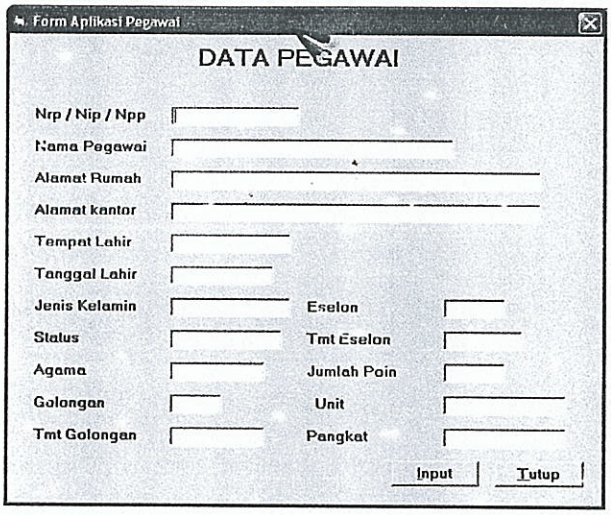
3.5.3 Menu Aplikasi

Menu aplikasi ini digunakan untuk pengisian form yang telah disediakan, yang nantinya akan ditampilkan pada menu Laporan.

3.5.3.1 Aplikasi Pegawai

Tabel 3.6. Pegawai

Field Name	Data Type
Nrp	Text
Nama_Pegawai	Text
Alamat_Rumah	Text
Alamat_Kantor	Text
Tempat_Lahir	Text
Tanggal_Lahir	Date/Time
Jenis_Kelamin	Text
Status	Text
Agama	Text
Golongan	Text
Tmt_Gol	Date/Time
Eselon	Text
Tmt_Eselon	Date/Time
Jumlah_Poin	Text
Unit	Text
Pangkat	Text

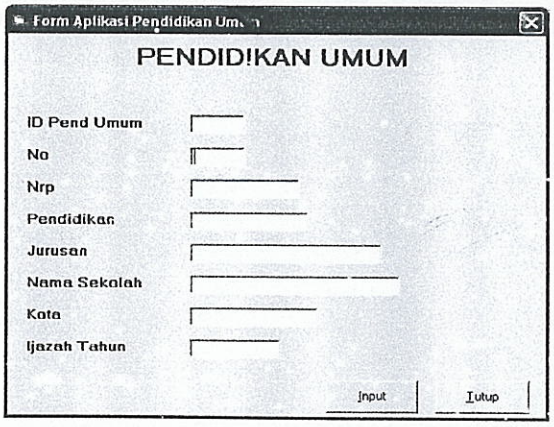


Gambar 3.15. Tampilan Aplikasi Pegawai

3.5.3.2 Aplikasi Pendidikan umum

Tabel 3.7. Umum

Field Name	Data type
ID_Umum	Text
No	Text
Nrp	Text
Pendidikan	Text
Jurusan	Text
Nama_Sekolah	Text
Kota	Text
Ijazah_Thn	Date/Time

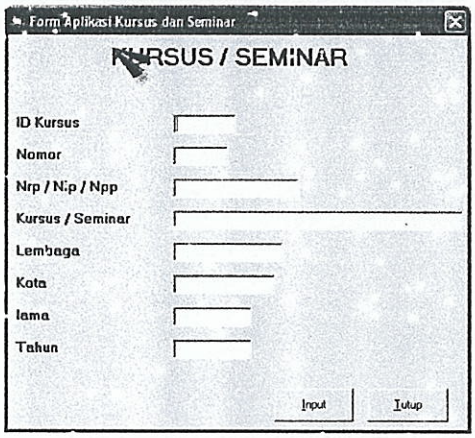


Gambar 3.16. Tampilan Aplikasi Pendidikan Umum

3.5.3.3 Aplikasi Kursus dan Seminar

Tabel 3.8. Kursus

Field Name	Data Type
ID_Kursus	Text
No	Text
Nrp	Text
Kursus	Text
Lembaga	Text
Kota	Text
Lama	Text
Tahun	Text

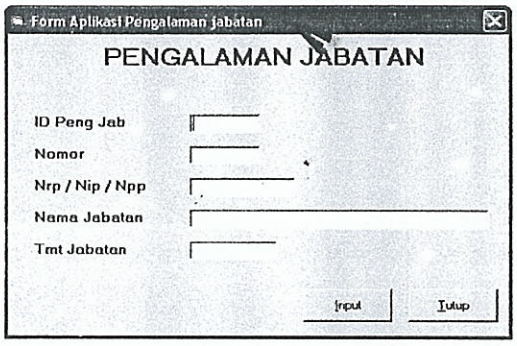


Gambar 3.17. Tampilan Aplikasi Kursus dan Seminar

3.5.3.4 Apliikasi Pengalaman Jabatan

Tabel 3.9. Pengalaman Jabatan

Field Name	Data Type
ID_Pengjab	Text
No	Text
Nrp	Text
Nama_Jabatan	Text
Tmt_Jabatan	Text

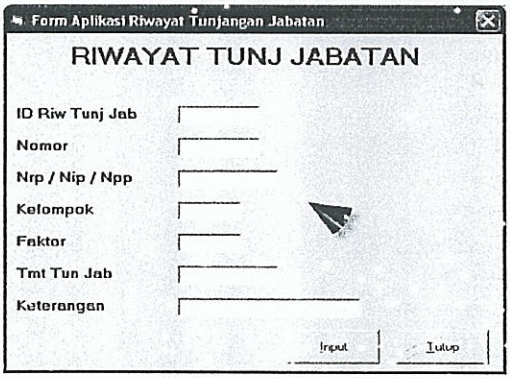


Gambar 3.18. Tampilan Aplikasi Pengalaman Jabatan

3.5.3.5 Aplikasi Riwayat Tunjangan Jabatan

Tabel 3.10 Riwayat Tunjangan Jabatan (Riw_Tun_Jab)

Field Name	Data Type
ID_RiwTunJab	Text
No	Text
Nrp	Text
Kelompok	Text
Faktor	Text
Tmt_Tunjab	Date/Time
Keterangan	Text

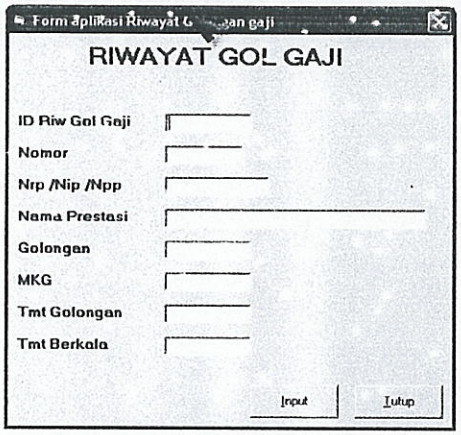


Gambar 3.19. Tampilan Aplikasi Riwayat Tunjangan Jabatan

3.5.3.6 Aplikasi Riwayat Golongan Gaji

Tabel 3.11. Riwayat Golongan Gaji (Riw_Gol_Gaji)

Field name	Data Type
ID_RiwGolGaji	Text
No	Text
Nrp	Text
Nama_Prestasi	Text
Gol	Text
MKG	Text
Tmt_Gol	Date/time
Tmt_Berkala	Date/time

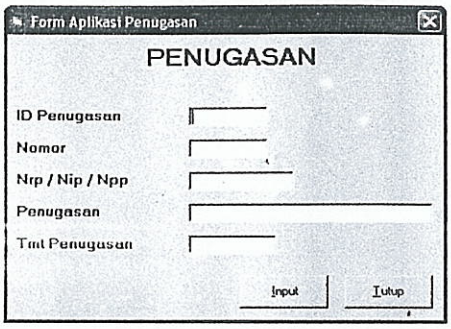


Gambar 3.20. Tampilan Aplikasi Riwayat Golongan Gaji

3.5.3.7 Apiikasi Penugasan

Tabel 3.12. Penugasan

Field Name	Data Type
ID_Penugasan	Text
No	Text
Nrp	Text
Penugasan	Text
Tmt_Penugasan	Date/Time



Gambar 3.21. Tampilan Aplikasi Penugasan

3.5.3.8 Aplikasi Pengalaman Lain

Tabel 3.13. Pengalaman Lain (Peng_Lain)

Field Name	Data Type
ID_PengLain	Text
No	Text
Nrp	Text
Nama_jabatan	Text
Tahun	Text

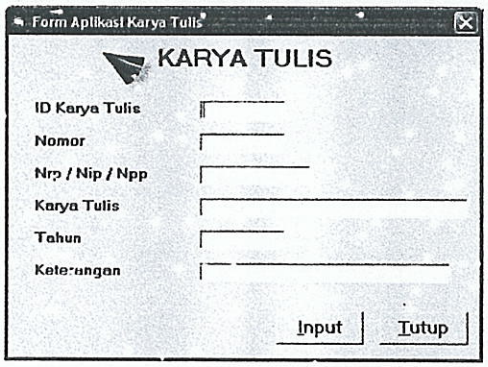


Gambar 3.22. Tampilan Aplikasi Pengalaman lain:

3.5.3.8 Aplikasi Karya Tulis

Tabel 3.14. Karya_Tulis

Field Name	Data Type
ID_Karya_Tulis	Text
No	Text
Nrp	Text
Karya_Tulis	Text
Tahun	Text
Keterangan	Text

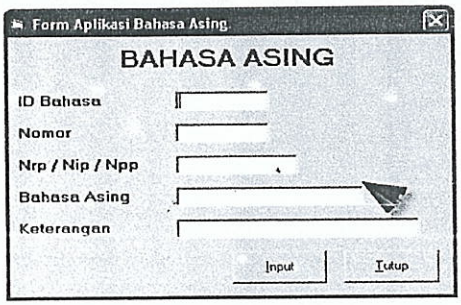


Gambar 3.23. Aplikasi Karya_Tulis

3.5.3.10Aplikasi Bahasa Asing

Tabel 3.15. Bahasa

Field Name	Data Type
ID_Bahasa	Text
No	Text
Nrp	Text
Bahasa	Text
Keterangan	Text



Gambar 3.24. Tampilan Aplikasi Bahasa Asing


UNIVERSITAS KOMPUTER INDONESIA
UNIKOM