BAB V

IMPLEMENTASI DAN PENGUJIAN SISTEM

5.1. Implementasi

Tahap implementasi adalah tahapan penerapan sistem untuk dapat dioperasikan. Pada tahapan ini dijelaskan mengenai sistem yang dirancang dan bagaimana cara penggunaannya.

5.1.1. Batasan Implementasi (optimal)

Implementasi merupakan kelanjutan dari kegiatan perancangan sistem dan dapat dipandang sebagai usaha untuk mewujudkan sistem yang dirancang. Langkah-langkah dari proses implementasi adalah urutan dari kegiatan awal sampai kegiatan akhir yang harus dilakukan dalam mewujudkan sistem yang dirancang seperti : kebutuhan akan *hardware, software*, desain input, desain output program, pengujian program.hasil dari tahapan implementasi ini adalah suatu sistem pengolahan data yang sudah dapat berjalan dengan baik.

5.1.2. Implementasi Perangkat Lunak

Untuk mendukung kelancaran sistem informasi data kepegawaian diperlukan perangkat lunak untuk mengeksekusi dari suatu program yang dirancang. Perangkat lunak digunakan sebagai pendukung informasi terdiri dari :

1. Sistem operasi windows (minimum window XP)
2. Database server menggunakan SQL server 2000, *interprice manager*

3. Laporan menggunakan Crystal Report 8.5
4. Bahasa pemograman menggunakan Visual Basic 6.0

5.1.3. Implementasi Perangkat Keras

Untuk mendukung terlaksananya penerapan aplikasi Visual Basic 6.0 yang dirancang, maka sistem pengoperasian ini membutuhkan komputer sebagai server yang memudahkan dalam menjalankan program aplikasi dengan spesifikasi sebagai berikut :

Server :

a. Menggunakan minimal processor Intel Pentium III atau sekelasnya.
b. Menggunakan RAM minimal 256 Mb.
c. Tersedianya *Hard Drive* sebagai media penyimpanan, minimal 20 Gb
d. Monitor, Mouse, Keyboard sebagai peralatan antar muka.
e. Lan Card.

Client :

a. Menggunakan minimal processor Intel Centrino atau sekelasnya.
b. Menggunakan RAM minimal 256 Mb.
c. Tersedianya *Hard Drive* sebagai media penyimpanan, minimal 40 Gb
d. Monitor, Mouse, Keyboard sebagai peralatan antar muka.
e. Lan Card.

5.1.4. Implementasi Basis Data

Sintaks SQL :

```
CREATE TABLE [dbo].[TblPegawai] (
	[nip] [varchar] (15) COLLATE SQL_Latin1_General_CP1_CI_AS NOT
NULL ,
	[nama] [varchar] (35) COLLATE SQL_Latin1_General_CP1_CI_AS
NULL ,
	[karpeg] [varchar] (15) COLLATE SQL_Latin1_General_CP1_CI_AS
NULL ,
	[tempat_lahir] [varchar] (35) COLLATE
SQL_Latin1_General_CP1_CI_AS NULL ,
	[tgl_lahir] [datetime] NULL ,
	[jenis_kelamin] [varchar] (15) COLLATE
SQL_Latin1_General_CP1_CI_AS NULL ,
	[agama] [varchar] (15) COLLATE SQL_Latin1_General_CP1_CI_AS
NULL ,
	[pendidikan_terakhir] [varchar] (50) COLLATE
SQL_Latin1_General_CP1_CI_AS NULL ,
	[gol_terakhir] [varchar] (5) COLLATE
SQL_Latin1_General_CP1_CI_AS NULL ,
	[gapok_terakhir] [float] NULL ,
	[jabatan_terakhir] [varchar] (50) COLLATE
SQL_Latin1_General_CP1_CI_AS NULL ,
```

```
	[unit_organisasi] [varchar] (50) COLLATE
SQL_Latin1_General_CP1_CI_AS NULL ,
	[alamat_rumah] [varchar] (100) COLLATE
SQL_Latin1_General_CP1_CI_AS NULL ,
	[nama_pasangan] [varchar] (35) COLLATE
SQL_Latin1_General_CP1_CI_AS NULL ,
	[tgl_lahir_pasangan] [datetime] NULL ,
	[tgl_menikah] [datetime] NULL ,
	[tmt] [datetime] NULL ,
	[masa_jabatan] [varchar] (50) COLLATE
SQL_Latin1_General_CP1_CI_AS NULL
) ON [PRIMARY]
GO
CREATE TABLE [dbo].[TblGolongan] (
	[gol] [varchar] (5) COLLATE SQL_Latin1_General_CP1_CI_AS NULL ,
	[pangkat] [varchar] (25) COLLATE SQL_Latin1_General_CP1_CI_AS
NULL ,
	[gapok] [float] NULL
) ON [PRIMARY]
GO
CREATE TABLE [dbo].[TblDetailAnak] (
	[nip] [varchar] (15) COLLATE SQL_Latin1_General_CP1_CI_AS NULL
```

```
	[nama_anak] [varchar] (35) COLLATE
SQL_Latin1_General_CP1_CI_AS NULL ,
	[tgl_lahir_anak] [datetime] NULL ,
	[nama_ayah] [varchar] (35) COLLATE
SQL_Latin1_General_CP1_CI_AS NULL ,
	[nama_ibu] [varchar] (35) COLLATE SQL_Latin1_General_CP1_CI_AS
NULL
) ON [PRIMARY]
GO
CREATE TABLE [dbo].[TblMutasi] (
	[no_sk_mutasi] [varchar] (15) COLLATE
SQL_Latin1_General_CP1_CI_AS NOT NULL ,
	[tgl_sk_mutasi] [datetime] NULL ,
	[no_mohon_mutasi] [varchar] (15) COLLATE
SQL_Latin1_General_CP1_CI_AS NULL ,
	[jabatan_awal] [varchar] (50) COLLATE
SQL_Latin1_General_CP1_CI_AS NULL ,
	[jabatan_baru] [varchar] (50) COLLATE
SQL_Latin1_General_CP1_CI_AS NULL
) ON [PRIMARY]
GO
CREATE TABLE [dbo].[TblPensiun] (
```

```
	[no_sk_pensiun] [varchar] (15) COLLATE
SQL_Latin1_General_CP1_CI_AS NOT NULL ,
	[tgl_sk_pensiun] [datetime] NULL ,
	[no_mohon_pensiun] [varchar] (15) COLLATE
SQL_Latin1_General_CP1_CI_AS NULL ,
	[gol_lama] [varchar] (5) COLLATE SQL_Latin1_General_CP1_CI_AS
NULL ,
	[gol_baru] [varchar] (5) COLLATE SQL_Latin1_General_CP1_CI_AS
NULL ,
	[gapok_lama] [float] NULL ,
	[gapok_baru] [float] NULL ,
	[masa_kerja_pensiun] [varchar] (50) COLLATE
SQL_Latin1_General_CP1_CI_AS NULL ,
	[pensiun_TMT] [datetime] NULL ,
	[pensiun_pokok] [float] NULL
) ON [PRIMARY]
GO
CREATE TABLE [dbo].[TblPermohonanKP] (
	[no_mohon_kp] [varchar] (15) COLLATE
SQL_Latin1_General_CP1_CI_AS NOT NULL ,
	[tgl_mohon_kp] [datetime] NULL ,
	[nip] [varchar] (15) COLLATE SQL_Latin1_General_CP1_CI_AS
NULL,
```

```
	[gol_lama] [varchar] (5) COLLATE SQL_Latin1_General_CP1_CI_AS
NULL ,
	[gol_baru] [varchar] (5) COLLATE SQL_Latin1_General_CP1_CI_AS
NULL
) ON [PRIMARY]
GO
CREATE TABLE [dbo].[TblPermohonanMutasi] (
	[no_mohon_mutasi] [varchar] (15) COLLATE
SQL_Latin1_General_CP1_CI_AS NOT NULL ,
	[tgl_mohon_mutasi] [datetime] NULL ,
	[nip] [varchar] (15) COLLATE SQL_Latin1_General_CP1_CI_AS
NULL,
	[unit_organisasi_awal] [varchar] (50) COLLATE
SQL_Latin1_General_CP1_CI_AS NULL ,
	[unit_organisasi_tujuan] [varchar] (50) COLLATE
SQL_Latin1_General_CP1_CI_AS NULL ,
	[alasan] [varchar] (50) COLLATE SQL_Latin1_General_CP1_CI_AS
NULL
) ON [PRIMARY]
GO
CREATE TABLE [dbo].[TblPermohonanPensiun] (
	[no_mohon_pensiun] [varchar] (15) COLLATE
SQL_Latin1_General_CP1_CI_AS NOT NULL ,
```

```
	[tgl_mohon_pensiun] [datetime] NULL ,
	[nip] [varchar] (15) COLLATE SQL_Latin1_General_CP1_CI_AS
NULL,
	[mulai_pensiun] [datetime] NULL ,
	[alamat_pensiun] [varchar] (100) COLLATE
SQL_Latin1_General_CP1_CI_AS NULL
) ON [PRIMARY]
GO
CREATE TABLE [dbo].[TblRiwayatJabatan] (
	[id_jabatan] [int] NOT NULL ,
	[nip] [varchar] (15) COLLATE SQL_Latin1_General_CP1_CI_AS
NULL,
	[jabatan] [varchar] (50) COLLATE SQL_Latin1_General_CP1_CI_AS
NULL ,
	[eselon] [varchar] (5) COLLATE SQL_Latin1_General_CP1_CI_AS
NULL ,
	[unit_organisasi] [varchar] (50) COLLATE
SQL_Latin1_General_CP1_CI_AS NULL ,
	[tmt_jabatan] [datetime] NULL
) ON [PRIMARY]
GO
CREATE TABLE [dbo].[TblRiwayatPangkat] (
	[id_gol] [int] NOT NULL ,
```

```
	[nip] [varchar] (15) COLLATE SQL_Latin1_General_CP1_CI_AS
NULL,
	[gol] [varchar] (5) COLLATE SQL_Latin1_General_CP1_CI_AS NULL ,
	[no_skkp] [varchar] (15) COLLATE SQL_Latin1_General_CP1_CI_AS
NULL ,
	[tmt_kp] [datetime] NULL
) ON [PRIMARY]
GO
CREATE TABLE [dbo].[TblRiwayatPendidikan] (
	[id_pendidikan] [int] NOT NULL ,
	[nip] [varchar] (15) COLLATE SQL_Latin1_General_CP1_CI_AS
NULL,
	[pendidikan] [varchar] (35) COLLATE
SQL_Latin1_General_CP1_CI_AS NULL ,
	[jurusan] [varchar] (25) COLLATE SQL_Latin1_General_CP1_CI_AS
NULL ,
	[sekolah] [varchar] (50) COLLATE SQL_Latin1_General_CP1_CI_AS
NULL ,
	[th_lulus] [varchar] (4) COLLATE SQL_Latin1_General_CP1_CI_AS
NULL
) ON [PRIMARY]
GO
CREATE TABLE [dbo].[TblSKKP] (
```

```
	[no_skkp] [varchar] (15) COLLATE SQL_Latin1_General_CP1_CI_AS
NOT NULL ,
	[tgl_skkp] [datetime] NULL ,
	[no_mohon_kp] [varchar] (15) COLLATE
SQL_Latin1_General_CP1_CI_AS NULL ,
	[tmt_lama] [datetime] NULL ,
	[tmt_baru] [datetime] NULL ,
	[masa_kerja_lama] [varchar] (50) COLLATE
SQL_Latin1_General_CP1_CI_AS NULL ,
	[masa_kerja_baru] [varchar] (50) COLLATE
SQL_Latin1_General_CP1_CI_AS NULL ,
	[gapok_lama] [float] NULL ,
	[gapok_baru] [float] NULL ,
	[jenis_kenaikan] [varchar] (10) COLLATE
SQL_Latin1_General_CP1_CI_AS NULL
) ON [PRIMARY]
GO
CREATE TABLE [dbo].[TblUser] (
	[user_id] [int] NOT NULL ,
	[username] [varchar] (50) COLLATE SQL_Latin1_General_CP1_CI_AS
NULL ,
	[password] [varchar] (50) COLLATE SQL_Latin1_General_CP1_CI_AS
NULL ,
```

[hak_akses] [varchar] (20) COLLATE SQL_Latin1_General_CP1_CI_AS NULL

) ON [PRIMARY]

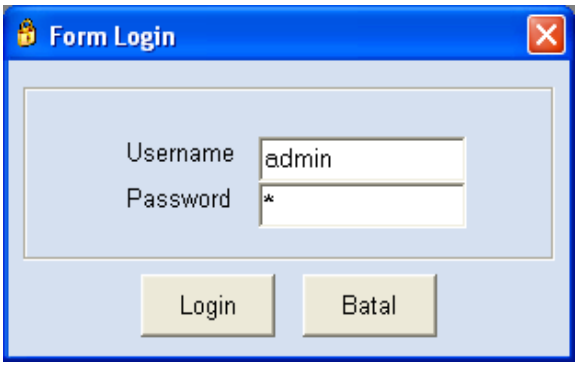
GO

5.1.5. Implementasi Antar Muka

Implementasi antar muka merupakan tahapan dalam memenuhi kebutuhan pengguna *(user)* dalam berinteraksi dengan komputer. Dalam tahapan implementasi ini sebuah program mempunyai fasilitas yang dapat membantu user dalam melakukan transaksi apabila pegawai ingin mengajukan pensiun, kenaikan pangkat atau mutasi, dalam program ini hak akses untuk admin dan user di batasi, ketika masuk login maka akan muncul form sebagai berikut:

a). Login

Login ini untuk mengamankan program agar tidak sembarang orang dapat masuk ke program ini dan untuk menjaga keamanan data agar tidak disalah gunakan.



Gambar 5.1 Tampilan Form Login

b). Form Menu Utama

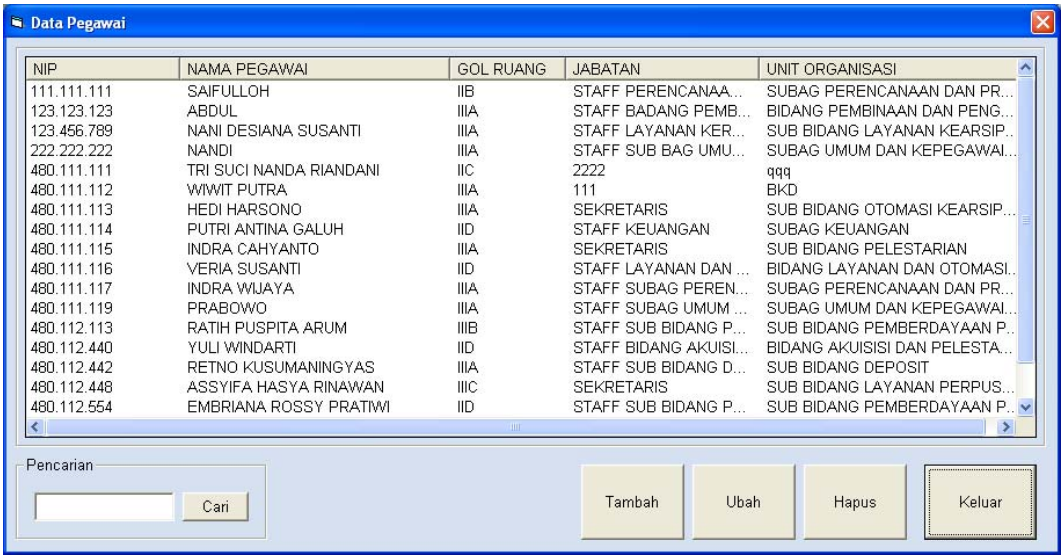
Setelah login di atur maka tampilah halaman utama yang terdiri dari tombol-tombol pada sebelah kiri atas tampilan, berfungsi untuk menampilkan dengan singkat file, data, proses, dan laporan. Gambarnya sebagai berikut untuk input data hanya admin yang dapat mengaksesnya.



Gambar 5.2 Form Menu Utama

c). Data Pegawai

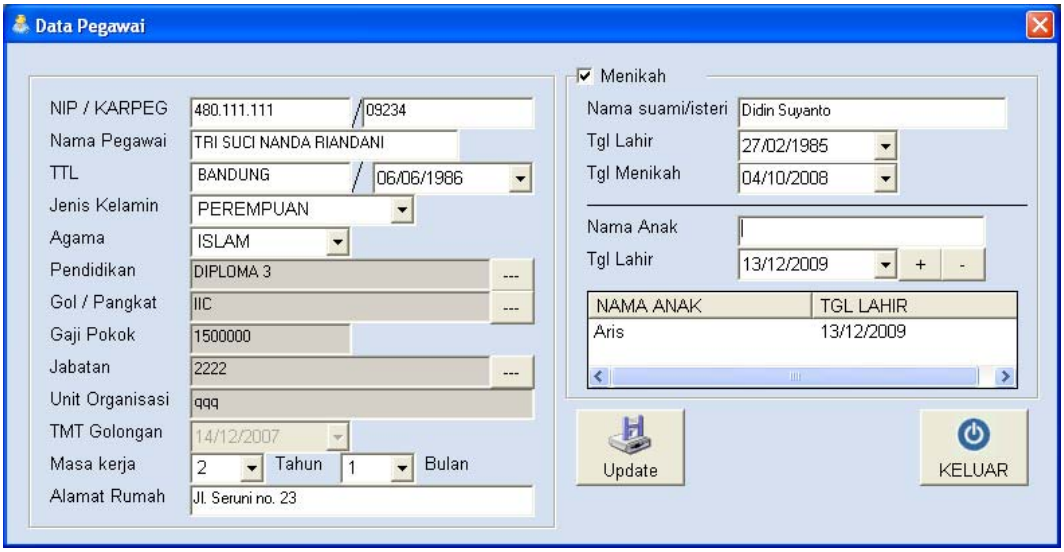
Form Data Pegawai adalah form yang menampilkan semua data pegawai. Untuk menampilkan form ini, caranya : Klik Menu **Data** >> **Pegawai** .Berikut adalah tampilan Form Data Pegawai:



Gambar 5.3 Form Data Pegawai

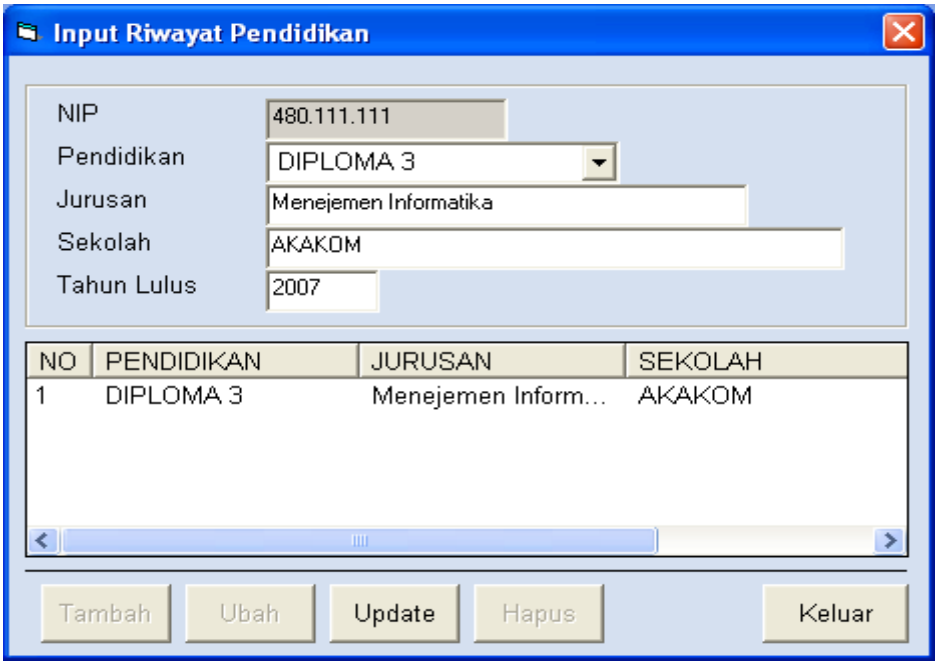
Untuk menambah data pegawai, ikuti langkah-langkah berikut:

1. Klik tombol Tambah
2. Setelah muncul form Data Pegawai, isilah data-data dengan benar.



Gambar 5.4 Form Pegawai

3. Untuk mengisi pendidikan, klik button yang ada di samping textbox pendidikan.kemudian akan muncul form seperti dibawah ini :



Gambar 5.5 Form Input Riwayat Pendidikan

4. Klik **Tambah**, dan isilah data pendidikan dengan benar. Setelah selesai, klik tombol **Simpan.**Untuk menambah data pendidikan, klik lagi tombol **Tambah**. Setelah selesai mengisi data pendidikan, klik Tombol **Keluar**.

   Catatan : Untuk Jurusan, jika kosong, isi saja dengan(-).

5. Untuk mengisi golongan dan jabatan, caranya sama dengan cara mengisi pendidikan.
6. Untuk mengisi Data Anak, isi dulu nama anak dan tanggal lahirnya, kemudian klik tombol (+) yang ada disamping Tgl Lahir. Untuk menambah data anak lagi, lakukan cara yang sama.
7. Setelah selesai mengisi data pegawai, klik **Simpan**.

Untuk mengubah data pegawai, ikuti langkah-langkah berikut:

- Klik data pegawai yang akan diubah yang ada di area grid, kemudian klik Ubah.

- Ubahlah data pegawai, dan klik update ketika selesai merubahnya.

Untuk menghapus data pegawai, ikuti langkah-langkah berikut:

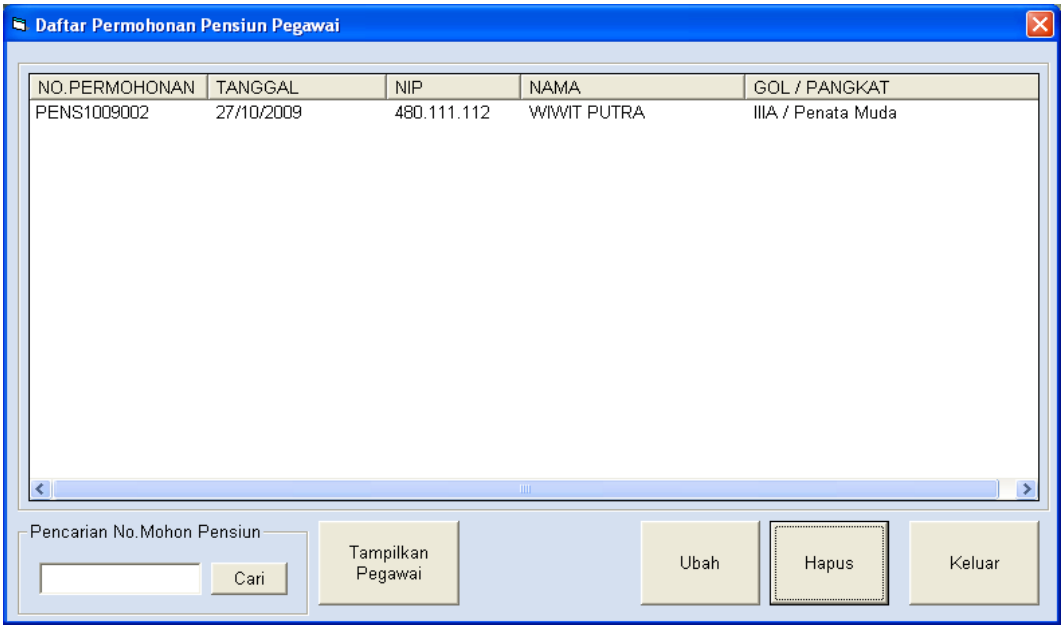
- Klik data pegawai yang akan dihapus yang ada di area grid, kemudian klik Hapus.

Untuk mencari data pegawai, ikuti langkah-langkah berikut :

- Masukkan NIP pegawai di Text Pencarian, dan klik **Cari**. Jika data yang dicari ada, maka akan tampil form data pegawai. Sebaliknya jika tidak ditemukan maka akan muncul pesan bahwa data tidak ditemukan.

d). Daftar Permohonan Pensiun Pegawai

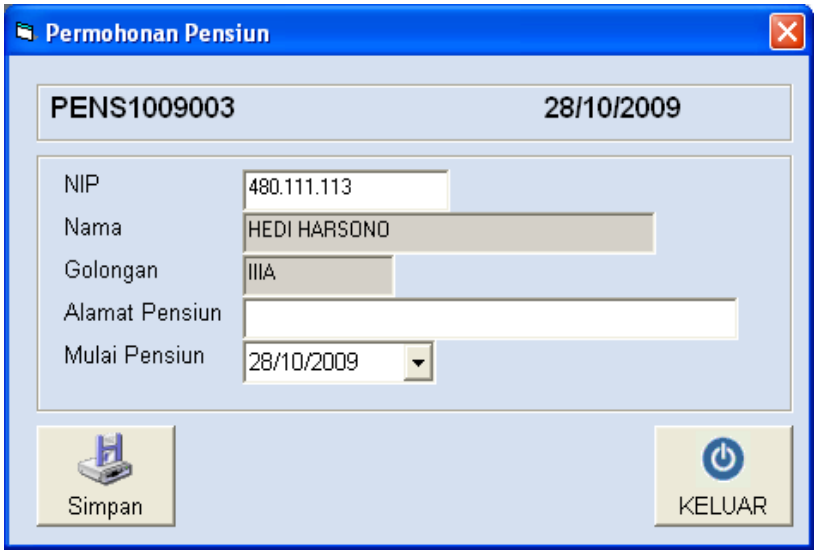
Form Daftar Permohonan Pensiun adalah form yang menampilkan semua daftar permohonan pensiun. Untuk menampilkan form ini, caranya : Klik Menu **Proses** >> **Pensiun** >> **Buat Permohonan** .Berikut adalah tampilan Form Daftar Permohonan Pensiun:



Gambar 5.6 Form Daftar Permohonan Pensiun Pegawai

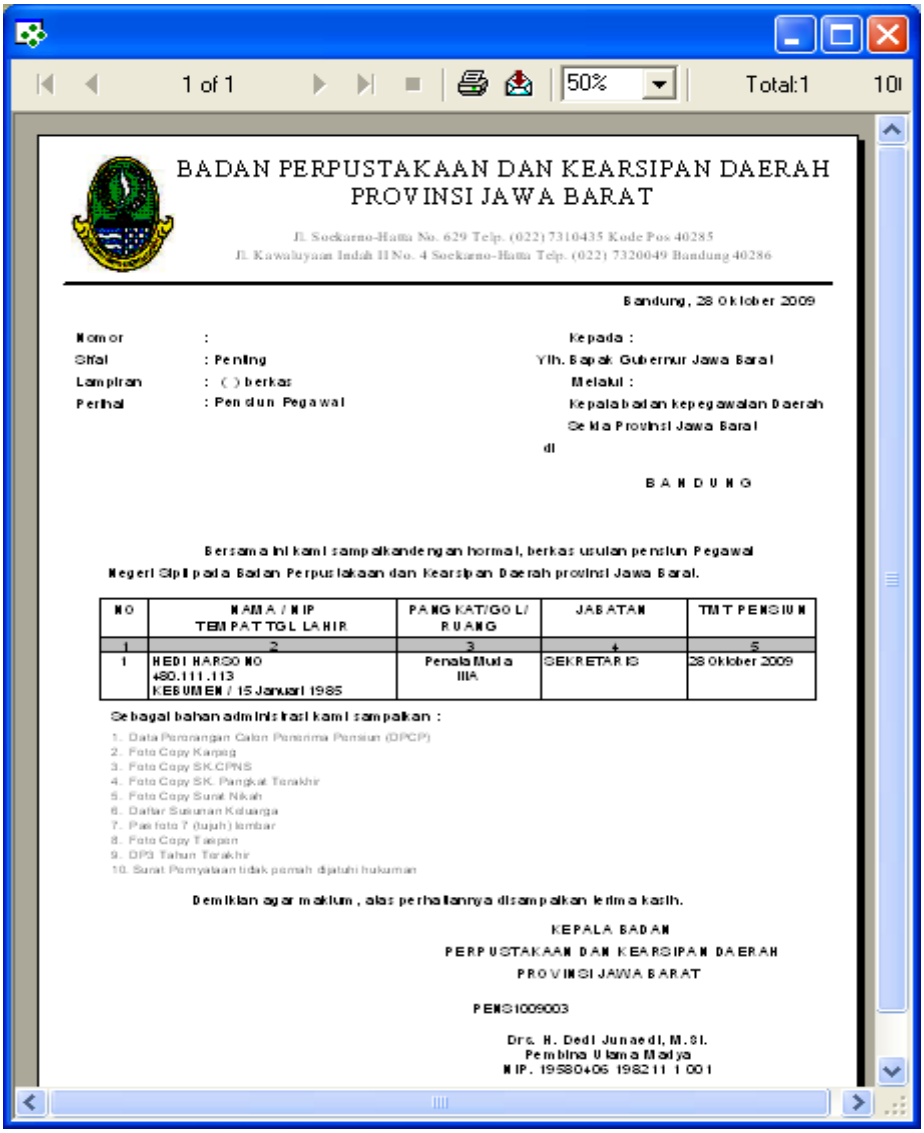
Untuk menambah data permohonan pensiun, ikuti langkah-langkah berikut:

1. Klik tombol **Tampilkan Pegawai** maka akan mucul semua data pegawai. Lalu klik pada nama pegawai.
2. Maka akan muncul form seperti dibawah ini :



Gambar 5.7 Form Permohonan Pensiun

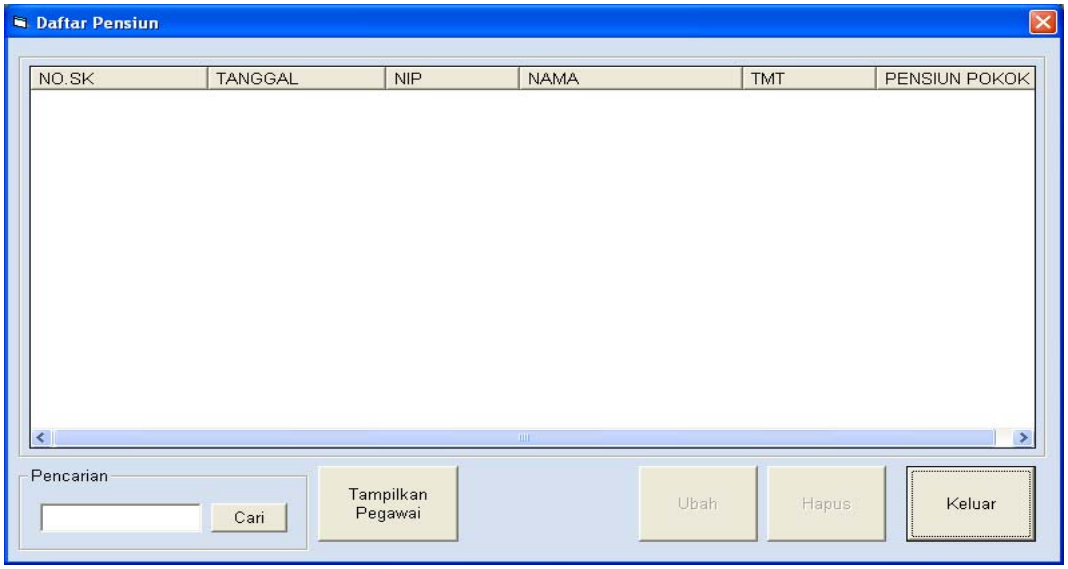
3. Hanya menambahkan data Alamat Pensiun dan Mulai Pensiun.
4. Setelah selesai mengisi data permohonan pensiun, klik **Simpan**.
5. Maka otomatis akan muncul **Surat Pengantar Pensiun** dengan tampilan sebagai berikut :



Gambar 5.8 Surat Pengantar Pensiun

e). Daftar SK Pensiun

Form Daftar SK Pensiun adalah form yang menampilkan semua daftar SK pensiun. Untuk menampilkan form ini, caranya : Klik Menu **Proses** >> **Pensiun** >> **SK Pensiun** .Berikut adalah tampilan Form Daftar SK Pensiun:



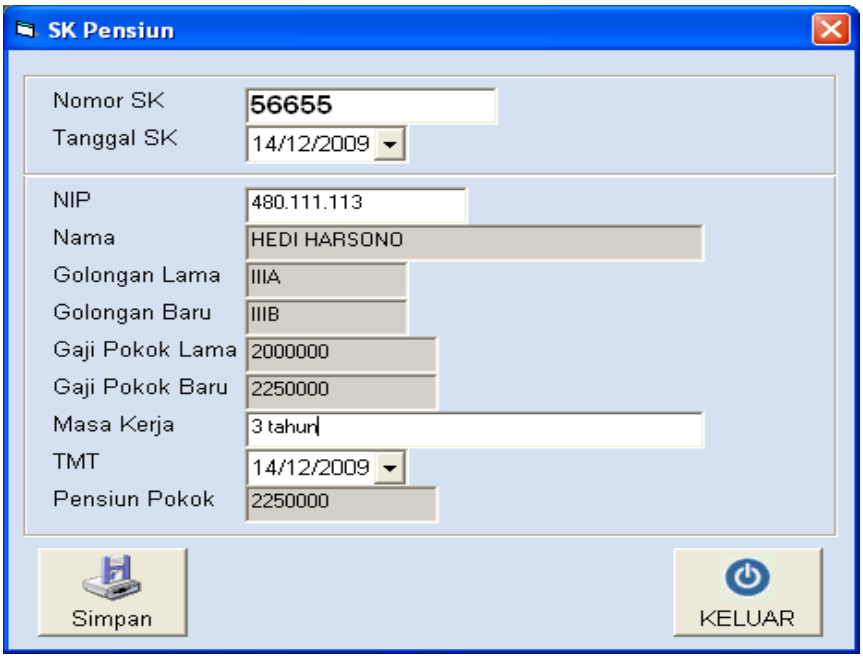
Gambar 5.9 Form Daftar SK Pensiun

Untuk menambah data SK pensiun, ikuti langkah-langkah berikut:

1. Klik tombol **Tampilkan Pegawai** maka akan mucul semua data pegawai.

   Lalu klik pada nama pegawai.

2. Maka akan muncul form seperti dibawah ini :

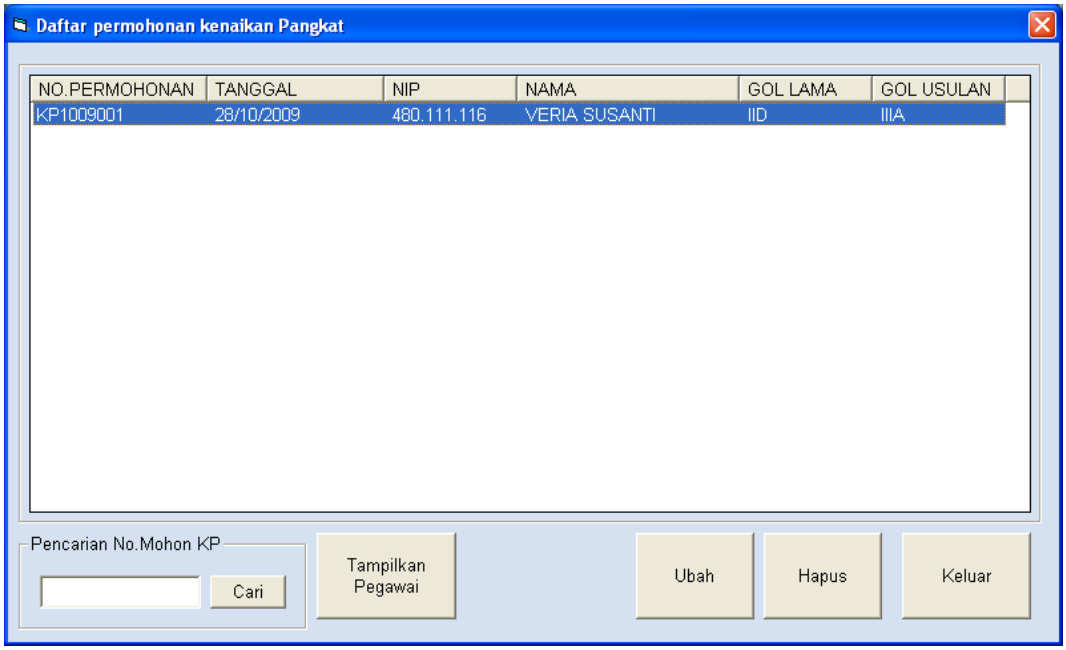


Gambar 5.10 Form SK Pensiun

3. Hanya menambahkan No SK, Tanggal SK, Lama Kerja dan TMT pensiun pegawai.
4. Setelah selesai mengisi data SK pensiun, klik **Simpan**.

f). Daftar Permohonan Kenaikan Pangkat

Form Daftar Permohonan Kenaikan Pangkat adalah form yang menampilkan semua daftar permohonan Kenaikan Pangkat. Untuk menampilkan form ini, caranya : Klik Menu **Proses** >> **Kenaikan Pangkat** >> **Buat Permohonan** .Berikut adalah tampilan Form Daftar Permohonan Kenaikan Pangkat:

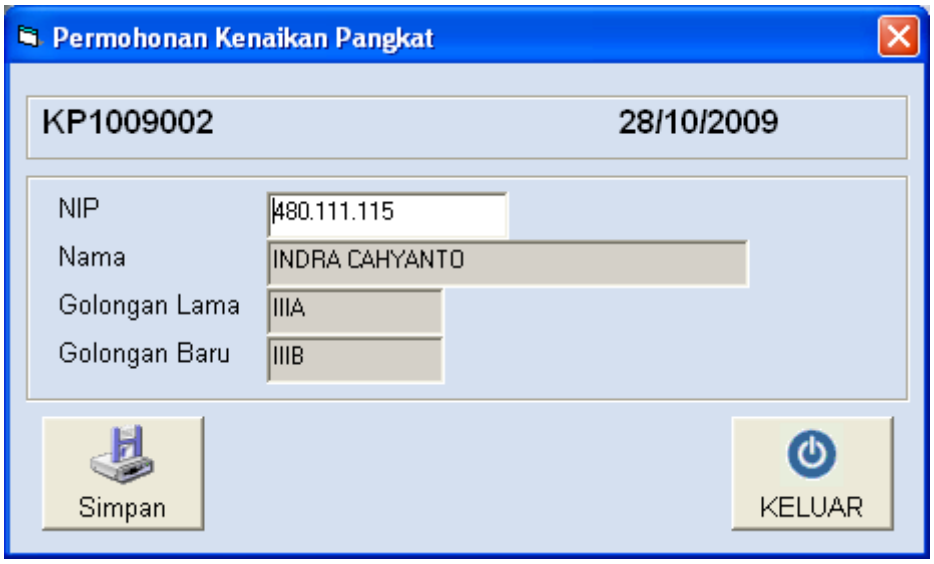


Gambar 5.11 Form Daftar Permohonan Kenaikan Pangkat

Untuk menambah data permohonan Kenaikan Pangkat, ikuti langkah-langkah berikut:

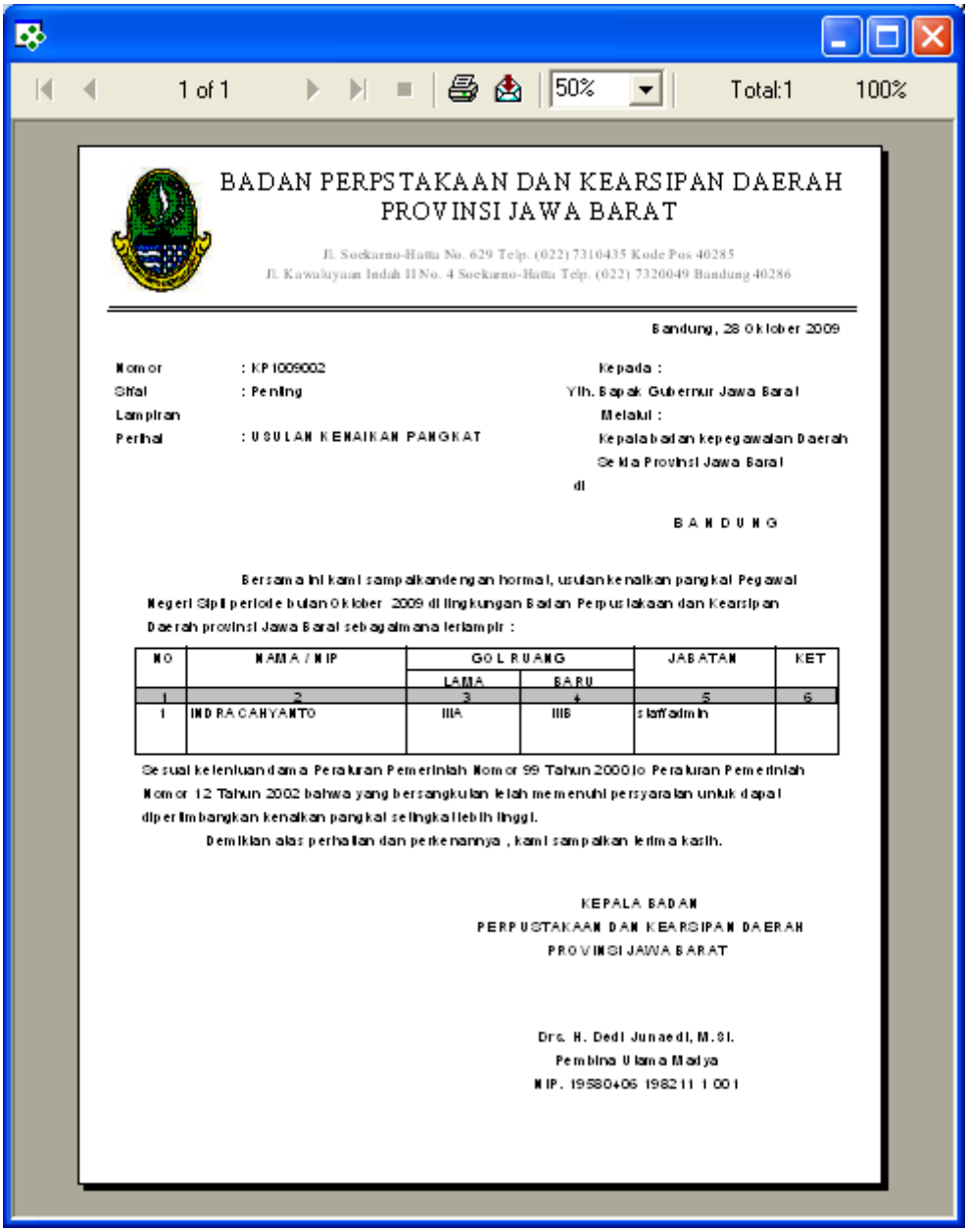
1. Klik tombol Tampilkan Pegawai maka akan mucul semua data pegawai. Lalu klik pada nama pegawai.

Maka akan muncul form seperti dibawah ini :



Gambar 5.12 Form Permohonan Kenaikan Pangkat

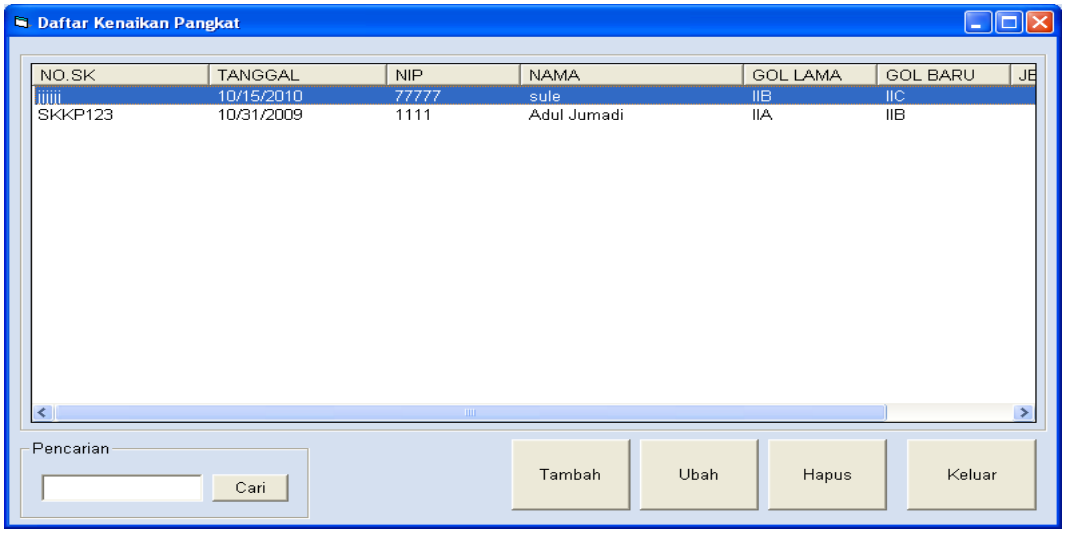
- Kemudian klik **Simpan**.
- Maka otomatis akan muncul Surat Pengantar Kenaikan Pangkat dengan tampilan sebagai berikut :



Gambar 5.13 Surat Pengantar Kenaikan Pangkat

g). Daftar SK Kenaikan Pangkat

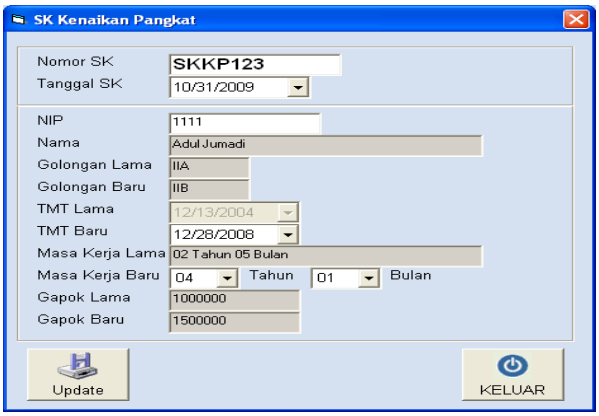
Form Daftar SK Kenaikan Pangkat adalah form yang menampilkan semua daftar SK Kenaikan Pangkat. Untuk menampilkan form ini, caranya : Klik Menu **Proses** >> **Kenaikan Pangkat >> SK Kenaikan Pangkat** .Berikut adalah tampilan Form Daftar SK Kenaikan Pangkat:



Gambar 5.14 Form Daftar Kenaikan Pangkat

Untuk menambah data SK Kenaikan Pangkat, ikuti langkah-langkah berikut:

- Klik tombol Tambah
- Setelah muncul form SK Kenaikan Pangkat, isilah data-data dengan benar.



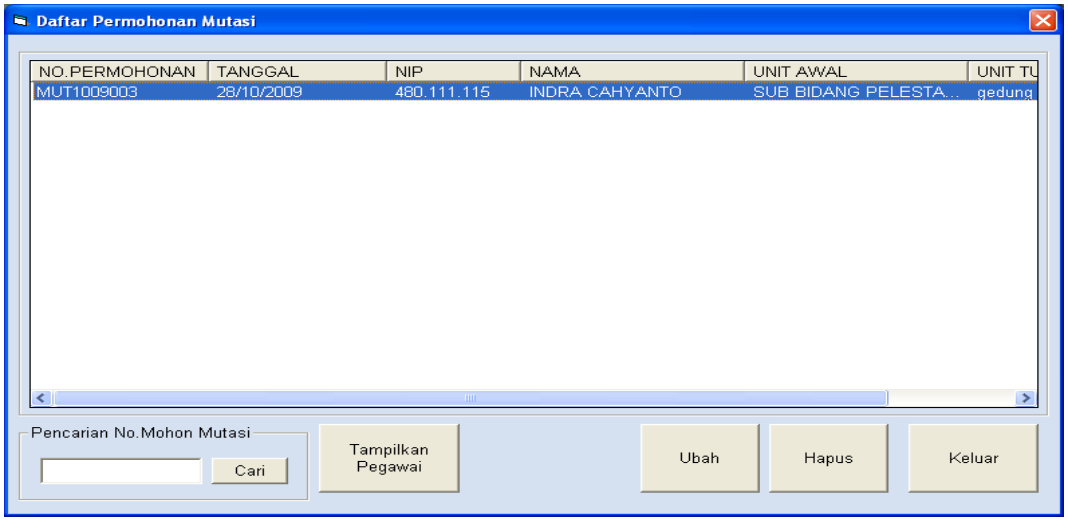
Gambar 5.15 Form SK Kenaikan Pangkat

- Dimulai dengan mengisi nomor SK yang ada di SK Kenaikan Pangkat yang dikeluarkan oleh Badan yang berwenang, kemudian isi tanggal SK sesuai dengan yang tertera di dalam SK Kenaikan Pangkat. Lalu isi NIP Pegawai dan tekan **enter** , kemudian isi TMT Baru pegawai.
- Selanjutnya isi masa kerja baru pegawai.

- Terakhir isilah jenis kenaikan pegawai.
- Setelah selesai mengisi data SK Kenaikan Pangkat, klik **Simpan**.

h). Daftar Permohonan Mutasi

Form Daftar Permohonan Mutasi adalah form yang menampilkan semua daftar permohonan Mutasi. Untuk menampilkan form ini, caranya : Klik Menu **Proses** >> **Mutasi >> Buat Permohonan**. Berikut adalah tampilan Form Daftar Permohonan Mutasi:



Gambar 5.16 Form Daftar Permohonan Mutasi

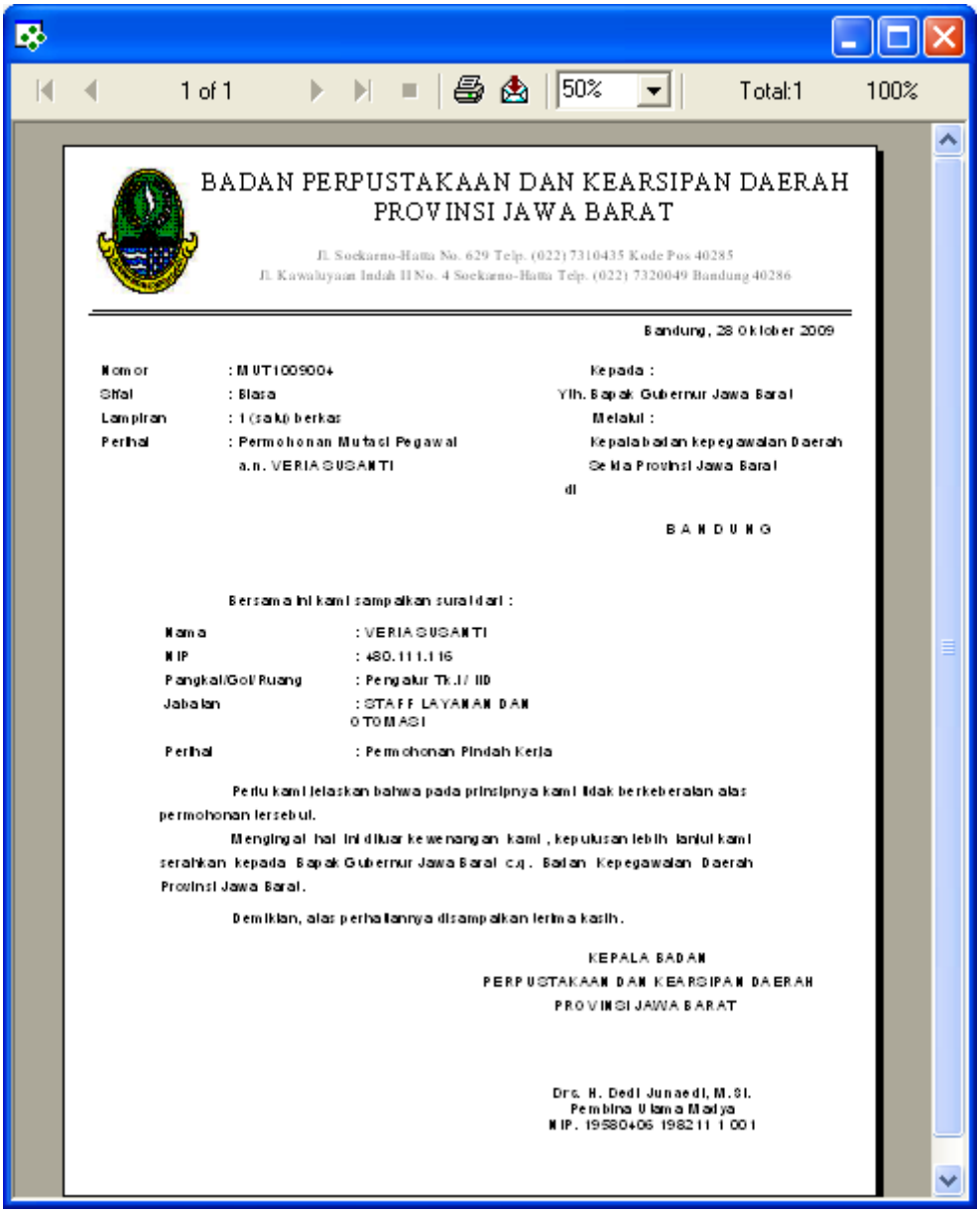
Untuk menambah data permohonan Mutasi, ikuti langkah-langkah berikut:

- Klik tombol Tampilkan Pegawai kemudian klik nama pegawai. Maka akan muncul tampilan seperti gambar dibawah ini:



Gambar 5.17 Form Permohonan Mutasi

- Kemudian isi unit tujuan mutasi dan alasan mutasi.
- Setelah selesai mengisi data permohonan Mutasi, klik **Simpan** maka otomatis akan muncul **Surat Pengantar Mutasi** . Seperti gambar dibawah ini:



BADAN PERPUSTAKAAN DAN KEARSIPAN DAERAH
PROVINSI JAWA BARAT

Jl. Soekarno-Hatta No. 629 Telp. (022) 7310435 Kode Pos 40285
Jl. Kawaluyaan Indah II No. 4 Soekarno-Hatta Telp. (022) 7320049 Bandung 40286

Bandung, 28 Oktober 2009

Nomor : MUT1009004
Sifat : Biasa
Lampiran : 1 (satu) berkas
Perihal : Permohonan Mutasi Pegawai a.n. VERIA SUSANTI

Kepada :
Yth. Bapak Gubernur Jawa Barat
Melalui :
Kepala Badan kepegawaian Daerah
Setia Provinsi Jawa Barat
di

BANDUNG

Bersama ini kami sampaikan surat dari :

Nama : VERIA SUSANTI
NIP : 480.111.116
Pangkat/Gol/Ruang : Pengatur Tk.I / IID
Jabatan : STAFF LAYANAN DAN OTOMASI
Perihal : Permohonan Pindah Kerja

Perlu kami jelaskan bahwa pada prinsipnya kami tidak berkeberatan atas permohonan tersebut.

Mengingat hal ini diluar kewenangan kami , kepukusan lebih lanjut kami serahkan kepada Bapak Gubernur Jawa Barat c.q. Badan Kepegawaian Daerah Provinsi Jawa Barat.

Demikian, atas perhatiannya disampaikan terima kasih.

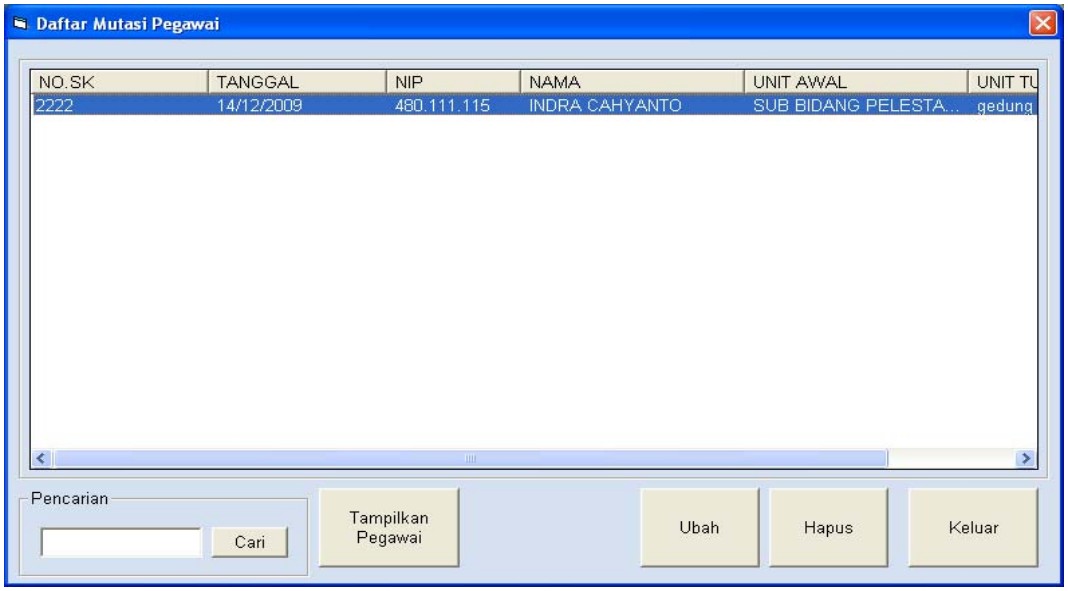
KEPALA BADAN
PERPUSTAKAAN DAN KEARSIPAN DAERAH
PROVINSI JAWA BARAT

Drs. H. Dedi Junaedi, M.Si.
Pembina Utama Madya
NIP. 19580405 198211 1 001

Gambar 5.18 Surat Pengantar Mutasi

i). Daftar SK Mutasi

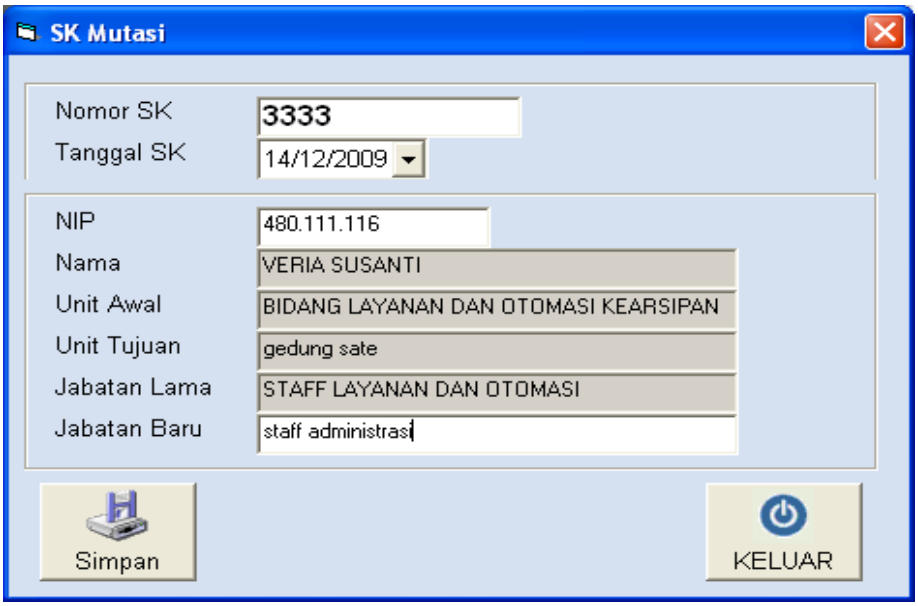
Form Daftar SK Mutasi adalah form yang menampilkan semua daftar SK Mutasi. Untuk menampilkan form ini, caranya : Klik Menu **Proses** >> **Mutasi** >> **SK Mutasi**. Berikut adalah tampilan Form Daftar SK Mutasi:



Gambar 5.19 Form Daftar Mutasi

Untuk menambah data SK Mutasi, ikuti langkah-langkah berikut:

- Klik tombol Tampilkan Pegawai maka akan muncul semua daftar pegawai, kemudian klik nama pegawai maka akan muncul tampilan sebagai berikut:



Gambar 5.20 Form SK Mutasi

- Hanya mengisi No SK, Tanggal SK dan Jabatan baru pegawai dan klik **Simpan**.

j). Laporan

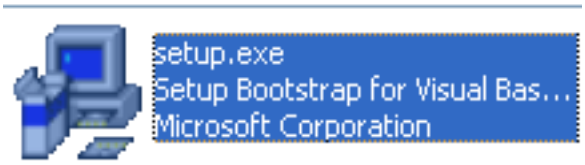
Pada form ini ditampilkan seluruh data laporan kepegawaian. Untuk menampilkan form ini, caranya : Klik Menu **Laporan** >> **Master Data.** Berikut adalah tampilan Form Laporan



Gambar 5.21 Form Laporan Data Pegawai

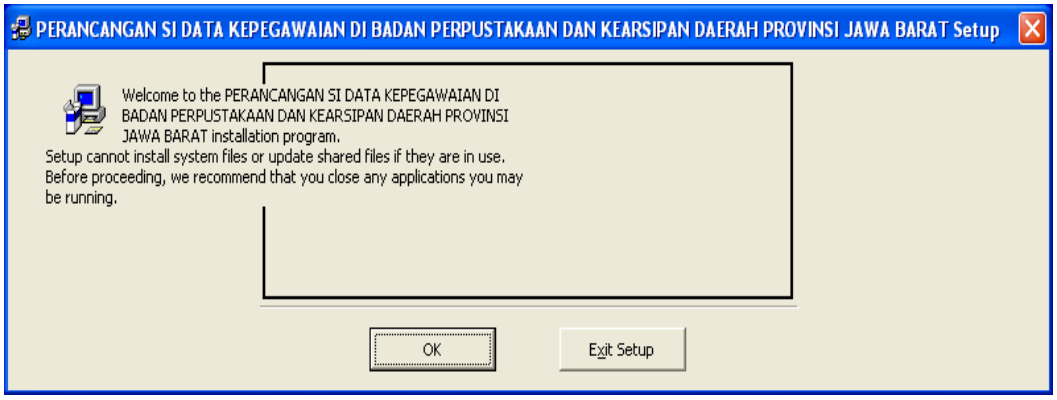
4.1.6. Implementasi Instalasi Program

1. Klik 2 kali atau klik kanan >> open pada file setup.exe master aplikasi Kepegawaian seperti pada gambar di bawah ini :



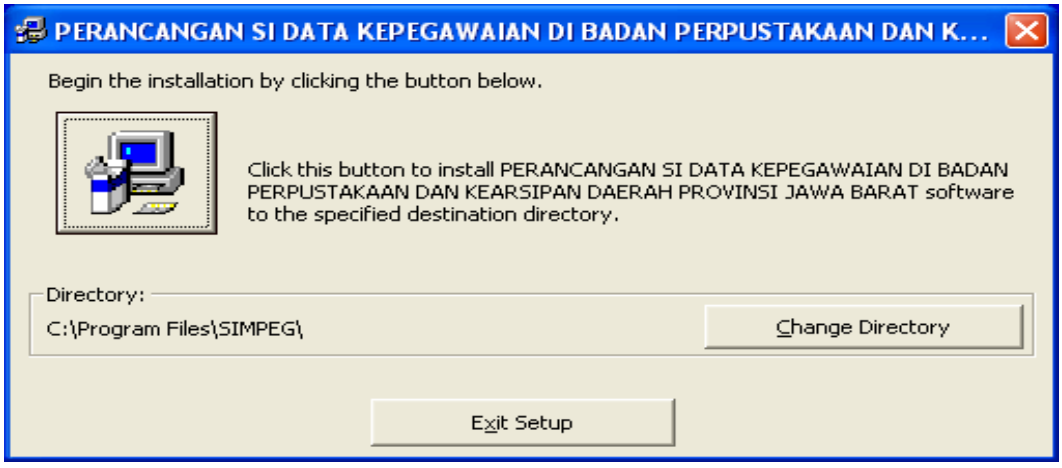
Gambar 5.22 Tahap Pertama Instalasi Aplikasi Kepegawaian

2. Kemudian akan tampil awal proses instalasi seperti pada gambar dibaah ini :



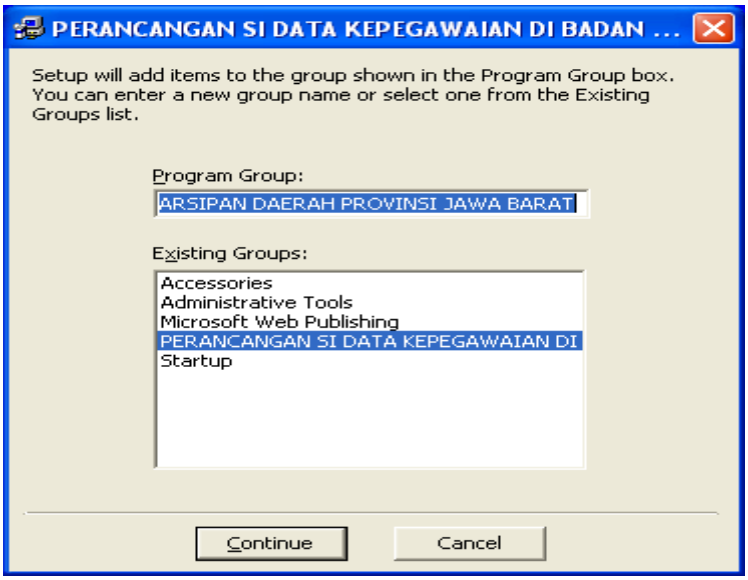
Gambar 5.23 Tahap Kedua Instalasi Aplikasi Kepegawaian

3. Klik tombol OK maka akan terlihat tampilan sebagai berikut :



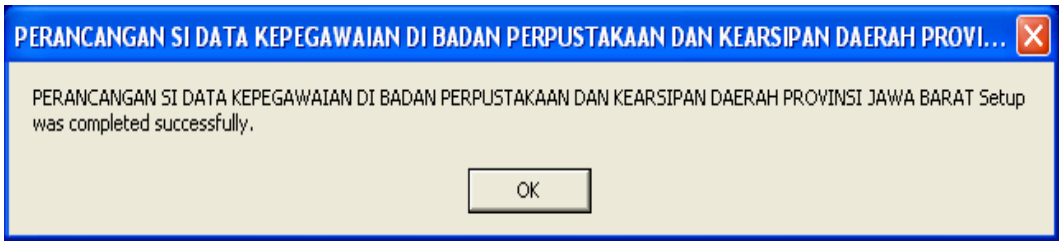
Gambar 5.24 Tahap Ketiga Instalasi Aplikasi Kepegawaian

4. Kemudian klik tombol instalasi untuk melanjutkan instalasi, maka akan muncul tampilan sebagai berikut :



Gambar 5.25 Tahap Keempat Instalasi Aplikasi Kepegawaian

5. Setelah muncul tampilan diatas maka pilih tombol *continue*, maka akan tampil *message box* sebagai berikut :



Gambar 5.26 Tahap Kelima Instalasi Aplikasi Kepegawaian

5.1.7. Penggunaan Program

Sistem informasi kepegawaian berbasis client server hakekatnya untuk mempermudah proses-proses yang terjadi dalam kegiatan proses pensiun,

kenaikan pangkat dan mutasi di Badan Perpustakaan dan Kearsipan Daerah Provinsi Jawa Barat.

1. Admin (bagian kepegawaian) berbeda hak aksesnya dengan Kepala BAPUSIPDA.
2. Setelah masuk program admin akan memasukan data pegawai apabila pegawai tersebut belum terdapat di dalam database apabila sudah ada di dalam database maka langsung dapat menggunakan program dan melakukan proses.
3. Kemudian setelah data diisi maka tekan tombol simpan maka data yang diinput akan tersimpan pada database pegawai.
4. Pada proses permohonan dilakukan satu persatu.
5. Kepala BAPUSIPDA hanya dapat melihat laporan saja.

5.2. Pengujian

Pengujian merupakan bagian yang penting dalam siklus membangun perangkat lunak. Pengujian dilakukan untuk menjamin kulaitas dan untuk mengetahui kelemahan dari perangkat lunak. Tujuan dari pengujian perangkat lunak adalah untuk menjamin perangkat lunak yang dibangun memilki kualitas yang handal, yaitu mampu mempresentasikan kajian pokok dari analisis, perancangan dan pengkodean dari perangkat lunak itu sendiri

5.2.1. Rencana Pengujian

Pengujian sistem informasi data kepegawaian berbasis client server berikut menggunkan data uji berupa sebuah data.

Tabel 5.1 Rencana Pengujian

Kelas Uji	**Butir Uji**	**Tingkat Pengujian**	**Jenis**
Pengisian Data pegawai	Proses permohonan	Modul	Blackbox

5.2.2. Kasus dan Hasil Pengujian

Pengujian Login

a. Kasus dan hasil uji (data normal)

Tabel 5.2. Pengujian *login*

Data Masuk	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Data login admin *username* : admin *Password*: 1	Tercantum pada textbox username dan password admin	Dapat mengisi login sesuai dengan yang diharapkan	[x] diterima [ ]ditolak
Klik ok	Maka akan masuk ke form utama	Tombol login dapat berfungsi sesuai dengan yang diharapkan	[x] diterima [ ] ditolak

b. Kasus dan hasil uji (data salah)

Tabel 5.3 kasus dan hasil uji

Data Masuk	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Data login admin *username* : xxxxx *Password* : xxxxx	Tidak dapat masuk ke program	Admin tidak dapat masuk	[x] diterima [ ]ditolak

2. Pengujian Proses

Tabel 5.4. Pengujian proses

Data Masuk	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi data lengkap	*File* yang tersimpan di database db_pegawai.mdf	Dapat menyimpan data yang masuk	Disimpan [x] diterima [ ] ditolak
Klik tombol tambah untuk menambah data	Maka akan langsung akan mengisi data pegawai	Dapat berfungsi merekam data yang masuk ke file pegawai	[x] diterima [ ] ditolak

5.2.3. Kesimpulan Hasil Pengujian

Berdasarkan hasil pengujian dengan kasus dan hasil uji, maka dapat diambil kesimpulan bahwa proses login dan pengisian data pegawai dapat dilakukan dengan sukses serta secara fungsional sistem dapat menghasilkan *output* yang dihasilkan.