BAB III
ANALISIS DAN PERANCANGAN SISTEM

3.1. Analisis Sistem

Pada bagian ini akan dilakukan analisis terhadap sistem yang sedang berjalan. Hasilnya digunakan untuk mengidentifikasi dan mengevaluasi berbagai kelemahan serta kekuatan yang terdapat pada sistem yang sudah dimiliki PT. Taspen. Dari proses analisis tadi akan dapat dihasilkan berbagai saran perbaikan terhadap sistem yang dapat dijadikan dasar dalam merancang sistem informasi yang akan dibangun.

3.1.1 Analisis Masalah

Dari hasil observasi, diketahui bahwa saat ini PT. Taspen Kantor Cabang Utama Bandung mengalami kesulitan dalam proses pengelolaan data kepegawaian sebagai acuan dalam perhitungan dana pensiun dan dana THT yang ada di seluruh Jawa Barat. Faktor jarak menjadi kendala utama, karena selama ini proses pendistribusian data antara kantor pusat dengan kantor cabang masih dilakukan secara berkala dengan cara mengirimkan sejumlah file data pegawai yang masih berupa file dokumen melalui *email*. Sehingga proses pendistribusian data tidak dapat berlangsung secara cepat. Hal ini menyebabkan sulitnya proses *update* data kepegawaian di setiap cabang ke kantor pusat.

Berdasarkan kelemahan-kelemahan tersebut, PT.Taspen Kantor Cabang Utama Bandung ingin melakukan perbaikan terhadap sistem terutama integrasi

antara kantor pusat dengan kantor cabang dengan membangun sistem informasi baru yang mampu berfungsi sesuai dengan kebutuhan PT. Taspen.

3.1.2 Analisis Sistem yang Sedang Berjalan

Berdasarkan hasil wawancara dengan bagian bidang pelayanan . Prosedur yang terlibat pada sistem yang akan dibangun terdiri atas dua prosedur, yaitu :

Prosedur pengajuan pembayaran klim THT dan pensiun

Di bawah ini adalah penjelasan prosedur pengajuan pembayaran klim THT dan pensiun

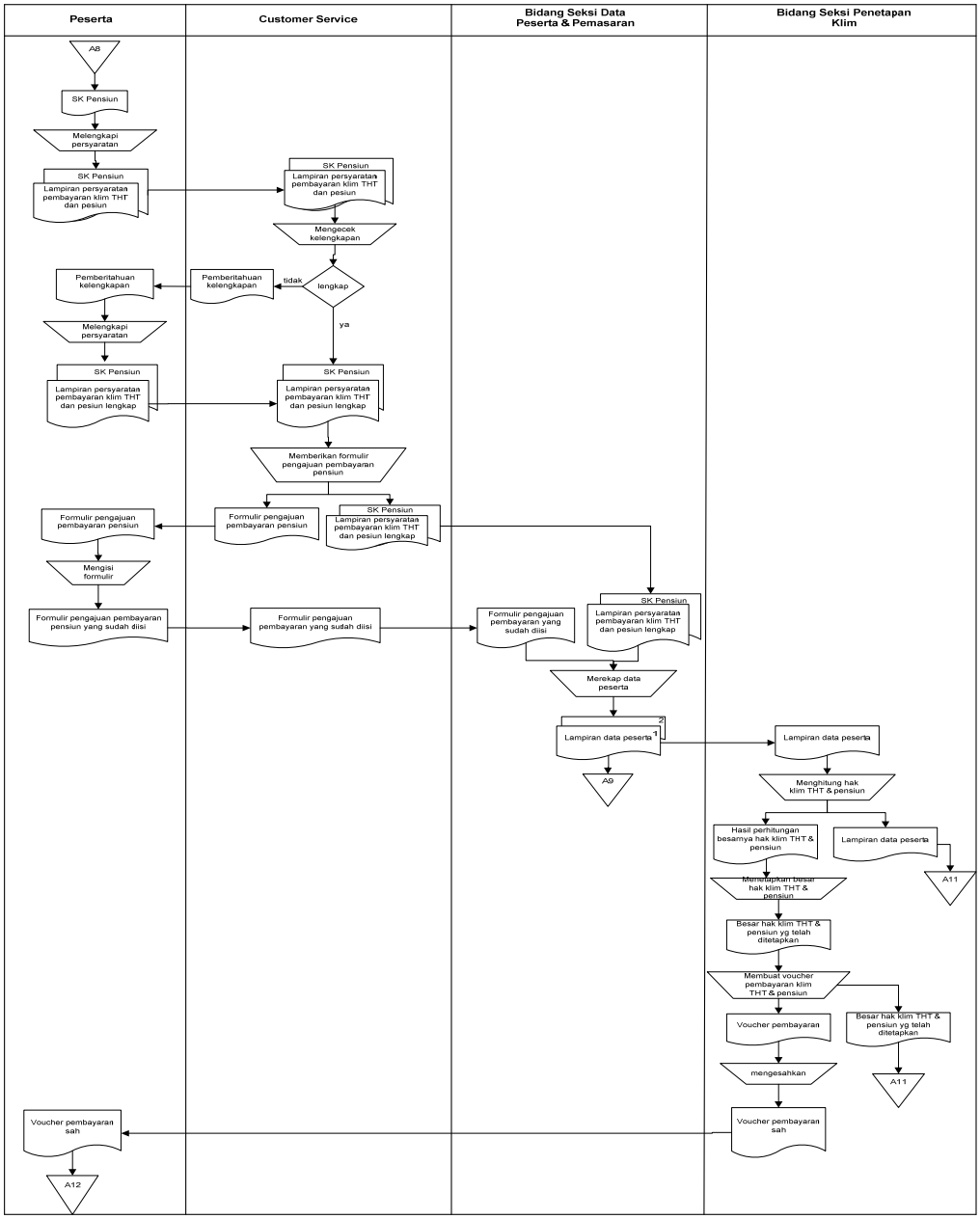
Prosedur ini melibatkan peserta, *customer service*, bidang seksi data peserta dan pemasaran, bidang seksi penetapan klim. Deskripsi prosedur pengajuan pembayaran klim THT dan pensiun adalah sebagai berikut:

1. Peserta membawa SK Pensiun dan lampiran persyaratan pembayaran klim THT dan pensiun kepada *customer service* yang ada di PT. Taspen.
2. Petugas *customer service* mengecek kelengkapan persyaratan pengajuan pembayaran klim THT dan pensiun. Jika persyaratan tidak lengkap, maka aka ada pemberitahuan kelengkapan yang diberikan oleh *customer service* kepada peserta. Sedangkan jika persyaratan sudah lengkap, *customer service* akan memberikan formulir pengajuan pembayaran pensiun untuk diisi oleh peserta.
3. Peserta mengisi formulir pengajuan pembayaran klim THT dan pensiun, kemudian formulir yang sudah diisi diberikan kembali kepada *Customer service*.

4. Formulir ini merupakan syarat untuk memproses pengajuan pembayaran klim THT dan pensiun dan berisi data-data yang berhubungan dengan status nonaktif seorang pegawai beserta data keuangan terakhir menjabat sebagai PNS.
5. Formulir yang sudah diisi akan disatukan dengan SK Pensiun, dan lampiran persyaratan pembayaran klim THT dan pensiun untuk diberikan kepada bagian seksi data peserta dan pemasaran untuk diproses lagi.
6. Formulir, SK Pensiun dan lampiran persyaratan pembayaran klim THT dan pensiun yang diberikan oleh *customer service* akan direkap oleh bagian seksi data peserta dan pemasaran.
7. Hasil rekap berupa lampiran data peserta yang akan dibuat rangkap dua, rangkap pertama akan disimpan pada bagian seksi data dan pemasaran sebagai arsip.
8. Rangkap kedua diberikan kepada bagian seksi penetapan klim untuk dilakukan perhitungan besarnya hak dana klim THT dan pensiun yang harus diterima oleh peserta.
9. Lampiran data peserta tadi akan disimpan di bagian seksi penetapan klim sebagai arsip.
10. Hasil dari perhitungan dibuat dokumen besarnya hak klim THT dan pensiun.
11. Penetapan besarnya klim THT dan dana pensiun dilakukan oleh bagian seksi penetapan klim THT dan pensiun setelah melakukan perhitungan.

12. Berkas besarnya hak klim THT dan pensiun yang telah ditetapkan disetujui oleh kepala bidang seksi penetapan klim dan berkas dijadikan arsip.
13. Bagian seksi penetapan iklim membuat voucher pembayaran klim THT dan pensiun untuk peserta.
14. Kepala bagian seksi penetapkan mengesahkan voucher pembayaran.
15. Voucher pembayaran sah akan diberikan kepada peserta sebagai bukti untuk pengambilan dana klim THT dan pensiunnya.

Prosedur pengajuan pembayaran klim THT dan pensiun dapat dlihat pada Gambar 3.1 :



Gambar 3.1 *Flowmap* Prosedur Pengajuan Pembayaran Klim THT dan Pensiun

Keterangan :

A9 : Lampiran data peserta yang diarsipkan oleh Bidang Seksi Data Peserta dan Pemasaran.

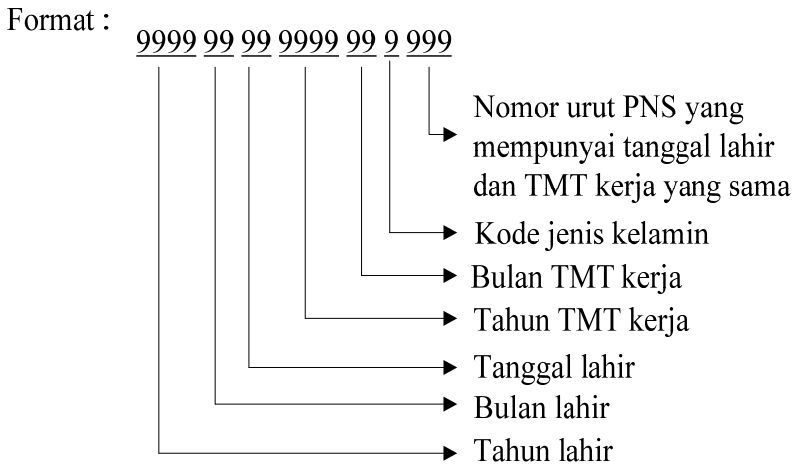
A10 : Lampiran data peserta yang diarsipkan oleh Bidang Seksi Penetapan Klim THT

A11 : Hak dana THT dan pensiun yang telah ditetapkan oleh Bidang Seksi Penetapan Klim THT

A12 : Voucher pembayaran sah yang diarsipkan oleh peserta.

3.1.3 Analisis Pengkodean

Untuk mengelola data pegawai, digunakan pengkodean berupa NIP (Nomor Induk Pegawai). NIP ini dimiliki setiap pegawai yang telah terdaftar sebagi PNS. Format NIP ini ditetapkan berupa rangkaian angka sebanyak 18 digit. Empat digit pertama merupakan tahun lahir PNS, dua digit berikutnya merupakan bulan lahir PNS, dua digit berikutnya merupakan tanggal lahir PNS, empat digit berikutnya merupakan tahun TMT kerja, dua digit berikutnya merupakan bulan TMT kerja, satu digit berikutnya merupakan kode jenis kelamin, dan tiga digit berikutnya merupakan nomor urut PNS yang mempunyai tanggal lahir sama dan TMT kerja yang sama.

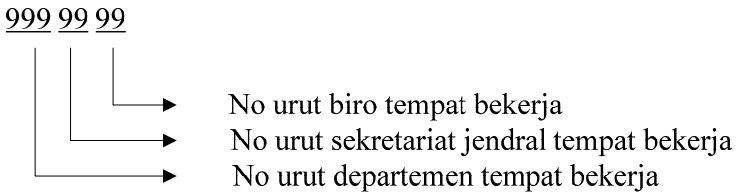


Contoh NIP 196908291999101002, berarti PNS yang mempunyai tanggal lahir 29 Agustus 1969 dengan TMT kerja pada Oktober 1999 yang berjenis kelamin laki-laki dengan no urut 2.

Pengkodean kedua adalah ID Satker. ID Satker dimiliki oleh setiap instansi kerja. Format ID Satker ini ditetapkan berupa rangkaian angka sebanyak

7digit. Tiga digit pertama menandakan urutan departemen tempat kerja, dua digit berikutnya menandakan urutan direktorat jenderal tempat bekerja dan dua digit berikutnya menandakan urutan biro instansi tempat kerja.

Format :



3.1.4 Analisis Kebutuhan Non Fungsional

Analisis dan kebutuhan non fungsional meliputi analisis dan kebutuhan pengguna,analisis dan kebutuhan perangkat keras serta analisis dan kebutuhan perangkat lunak.

3.1.4.1 Analisis dan Kebutuhan Pengguna

Analisis pengguna dimaksudkan untuk mengetahui siapa saja pengguna yang terlibat dalam proses pengolahan data pegawai siswa sehingga dapat diketahui tingkat pengalaman dan pemahaman pengguna komputer.

Dari hasil penelitian dapat dipaparkan pengguna yang terlibat dalam sistem yang sedang berjalan diantaranya adalah :

a. Pegawai Kantor Cabang Admin
    1. Minimal lulusan S1.
    2. Mempunyai keahlian dalam mengoperasikan komputer.
    3. Terbiasa menggunakan aplikasi dekstop.

b. Pegawai Kantor Cabang
    1. Minimal jurusan D3

2. Terbiasa mengoperasikan komputer.
3. Terbiasa mengolah data pegawai.

Agar sistem yang dibangun bisa digunakan secara maksimal, pengguna yang akan menggunakan aplikasi ini adalah :

a. Pegawai Kantor Cabang sebagai Admin

Admin adalah pengguna yang mengelola database yang berada di setiap cabang. Admin juga bertugas melakukan proses distribusi pengolahan data antara kantor cabang dengan kantor pusat.

Karakteristik sebagai beikut :

1. Minimal lulusan S1 Teknik Informatika dan Manajemen Informatika.
2. Mempunyai keahlian dalam mengoperasikan komputer.
3. Mempunyai keahlian dalam pemeliharaan jaringan.
4. Mempunyai keahlian dalam pemeliharaan *database*.
5. Mempunyai keahlian dalam pemeliharaan *web*.

b. Pegawai Kantor Cabang sebagai User

User adalah pengguna yang bertugas memasukkan data dan melakukan pengolahan data pegawai wilayahnya.

Karakteristiknya sebagai berikut :

1. Minimal D3 (semua jurusan).
2. Mampu mengoperasikan komputer.
3. Mempunyai pengalaman dalam proses *entry* data.
4. Terbiasa menggunakan *web browser*.

3.1.4.2 Analisis dan Kebutuhan Perangkat Keras

Adapun spesifikasi minimal perangkat keras yang digunakan sistem ini adalah:

1. Perangkat keras yang dibutuhkan pada sisi *server* pusat adalah :
    a. Prosesor IBM AIX V52
    b. RAM 2 Gb
    c. Kapasitas hardisk kosong 160 Gb
    d. CD ROM
    e. TSM *Client*
    f. Monitor SVGA 17”
    g. Jaringan untuk koneksi internet
2. Perangkat keras yang dibutuhkan pada sisi *server* cabang adalah :
    a. Prosesor IBM AIX V52
    b. RAM 1Gb
    c. Kapasitas hardisk kosong 40 Gb
    d. Monitor SVGA 15’’
    e. Jaringan untuk koneksi internet
4. Perangkat keras yang dibutuhkan pada sisi *client* cabang adalah:
    a. Processor Intel Pentium 4
    b. RAM 512 Mb
    c. Kapasitas hardisk kosong 4 Gb
    d. CD ROM
    e. Monitor SVGA 15’’

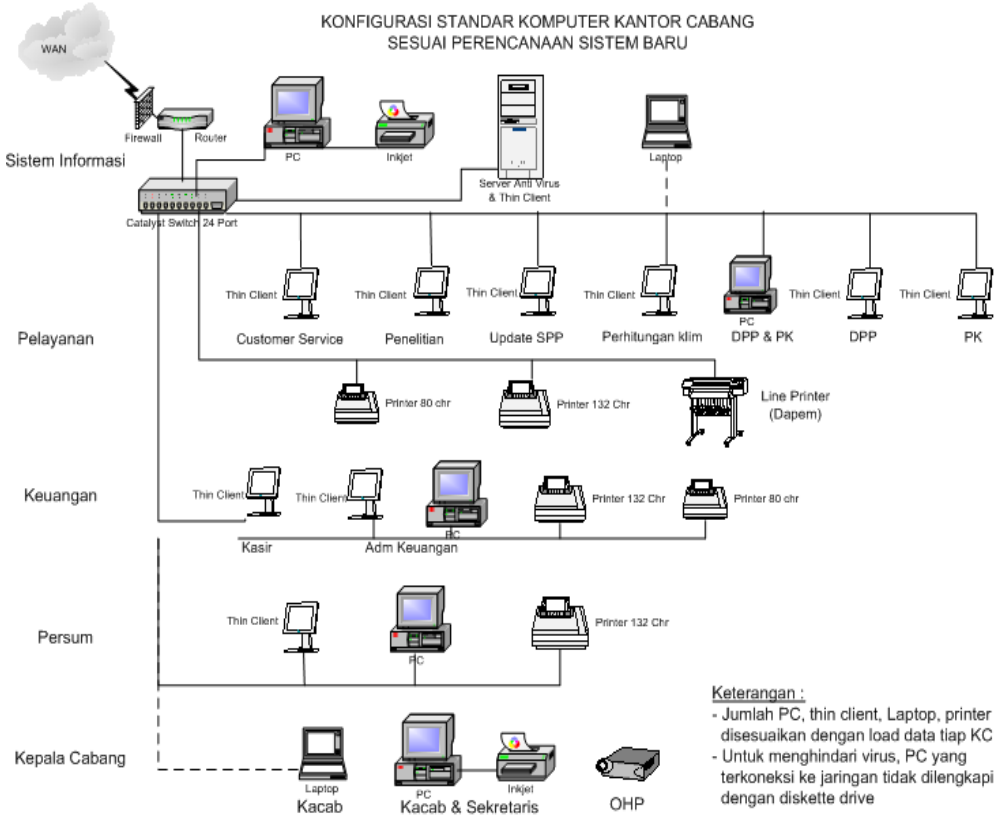
3.1.4.3 Analisis dan Kebutuhan Perangkat Lunak

Adapun spesifikasi minimal perangkat lunak yang digunakan system ini adalah:

1. Perangkat lunak yang dibutuhkan *server* pusat adalah :
    a. Microsoft Windows *Server*
    b. Xampp sebagai *webserver*
    c. PHP sebagai bahasa pemrograman
    d. MySQL sebagai system manajemen basis data
    e. *Web browser* seperti Mozilla Firefox.
2. Perangkat lunak yang dibutuhkan *server* cabang adalah :
    a. Microsoft Windows *Server*
    b. Xampp sebagai *webserver*
    c. PHP sebagai bahasa pemrograman
    d. MySQL sebagai system manajemen basis data
    e. *Web browser* seperti Mozilla Firefox.
3. Perangkat lunak yang dibutuhkan *server* cabang adalah :
    a. Microsoft Windows XP SP 2
    b. *Web browser* seperti Mozilla Firefox.

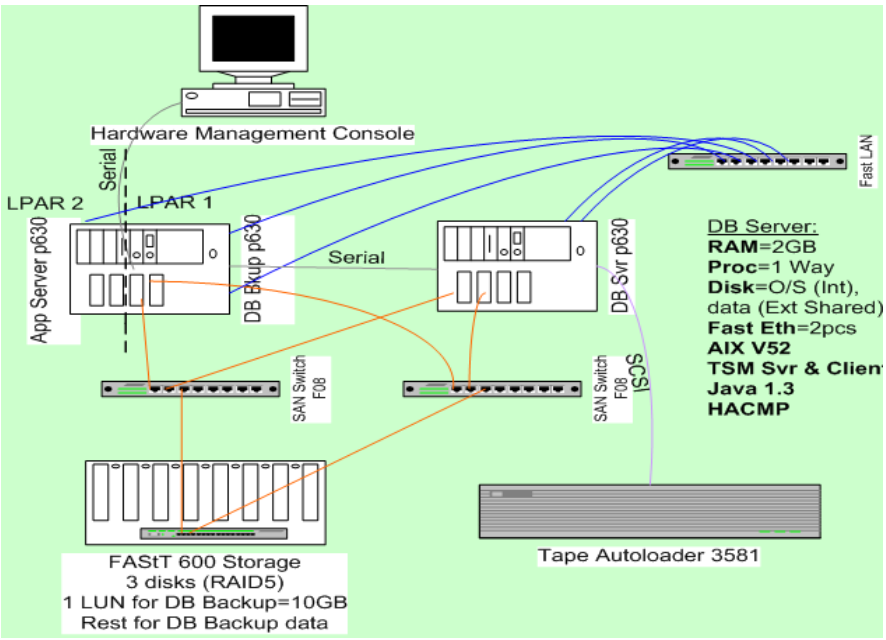
3.1.4.4 Analisis Topologi Jaringan

Topologi jaringan yang ada di setiap kantor cabang pada PT. Taspen dapat digambarkan sebagai berikut :



Gambar 3.2 Topologi Jaringan Pada PT. Taspen

Topologi yang digunakan pada kantor pusat sama halnya dengan kantor cabang, yang membedakan terletak pada sisi *server*. Dapat digambarkan sebagai berikut :



Gambar 3.3 Topologi Jaringan *Server* Pada PT.Taspen

3.1.5 Analisis Database Terdistribusi

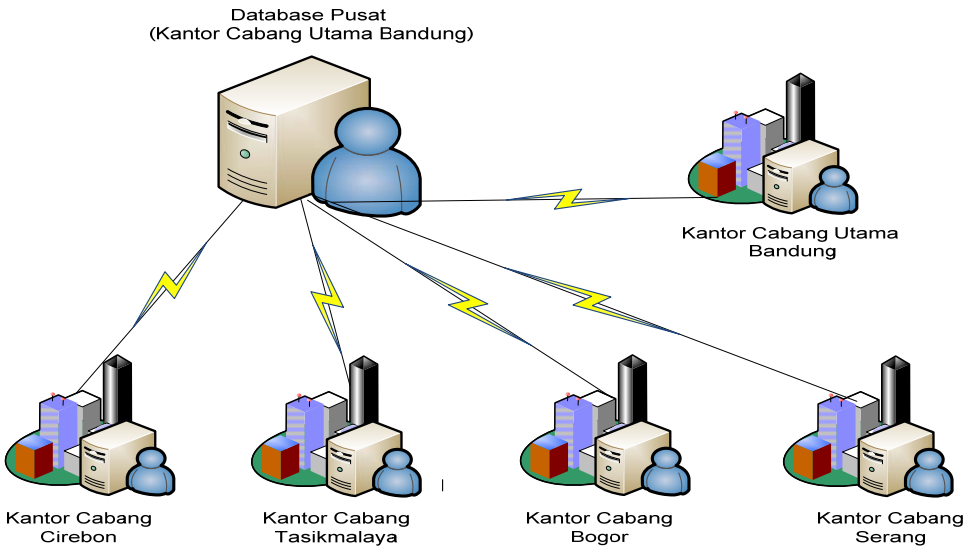
Melihat situasi dan kondisi yang terjadi pada PT. Taspen KCU Bandung yang merupakan pusat dari beberapa cabang yang dikelolanya terkadang ditemukan beberapa masalah pada saat pendistribusian data yang ada di kantor cabang ke kantor pusat. Dikarenakan pada saat ini kantor cabang hanya mengirim data-data mengenai PNS yang perlu diupdate melalui email. Sehingga jika pegawai cabang lupa mengirimkan data-data yang sudah diupdate ke kantor pusat akan menimbulkan suatu masalah jika ada data yang berhubungan dengan calon penerima dana taspen yang harus melibatkan penanganan kantor pusat. Selain itu proses pendistribusian data-data keuangan PNS dari setiap cabang diperlukan

karena kantor pusat harus mengetahui jumlah calon peserta penerima dana taspen setiap bulannya sebagai perkiraan dana yang harus dikeluarkan oleh kantor pusat untuk pembayaran dana taspen pesertanya di setiap kantor cabang. Dengan begitu kantor pusat mengetahui aliran dana yang akan dikeluarkan untuk setiap kantor cabang dalam mengelola pensiunan PNS.

Meskipun proses update data menjadi wewenang kantor cabang. Tetapi kantor pusat harus mendapat informasi mengenai semua pengelolaan data yang berkaitan dengan keuangan PNS, karena kantor pusat membawahi kantor cabang. Maka dari itu penerapan proses pendistribusian data dengan database terdistribusi merupakan salah satu cara agar proses updating bisa dilakukan antara kantor cabang dengan kantor pusat sehingga data yang ada pada database cabang sama dengan database pusat karena database cabang terintegrasi dengan database pusat.

3.1.5.1 Ars*ite*ktur Jaringan

Topologi yang akan digunakan untuk membangun database terdistribusi ini menggunakan bentuk *client-server* karena kontrol manajemen lebih terjamin. Selain itu sesuai dengan tujuan yang akan dicapai yaitu semua database di setiap cabang akan didistribusikan ke kantor pusat. Dikarenakan data yang akan didistribusikan adalah data keuangan PNS yang sangat penting, maka untuk menangani kegagalan yang mungkin terjadi, setiap kantor cabang tetap mempunyai database wilayahnya sendiri. Sehingga jika suatu saat database pusat bermasalah, kantor cabang tetap mempunyai data-data yang tersimpan di dalam databasenya.



Gambar 3.4 Arsitektur Jaringan

Keterangan:

a. Proses *entry data* akan dilakukan oleh pegawai pada setiap kantor cabang. Data yang sudah diupdate akan disimpan ke database lokalnya kemudian didistribusikan ke database pusat.

b. Database pusat berfungsi untuk menampung seluruh database cabangnya. Sedangkan database cabang hanya menampung database lokal wilayahnya saja.

c. Setiap database cabang melakukan update data, maka untuk data yang sama yang ada di database pusat otomatis akan ikut terupdate juga. Dengan begitu data pada database cabang akan sama dengan database pusat.

d. Jika pada saat tertentu database cabang bermasalah atau koneksi terputus, maka data yang akan didistribusikan akan tertampung terlebih dahulu di

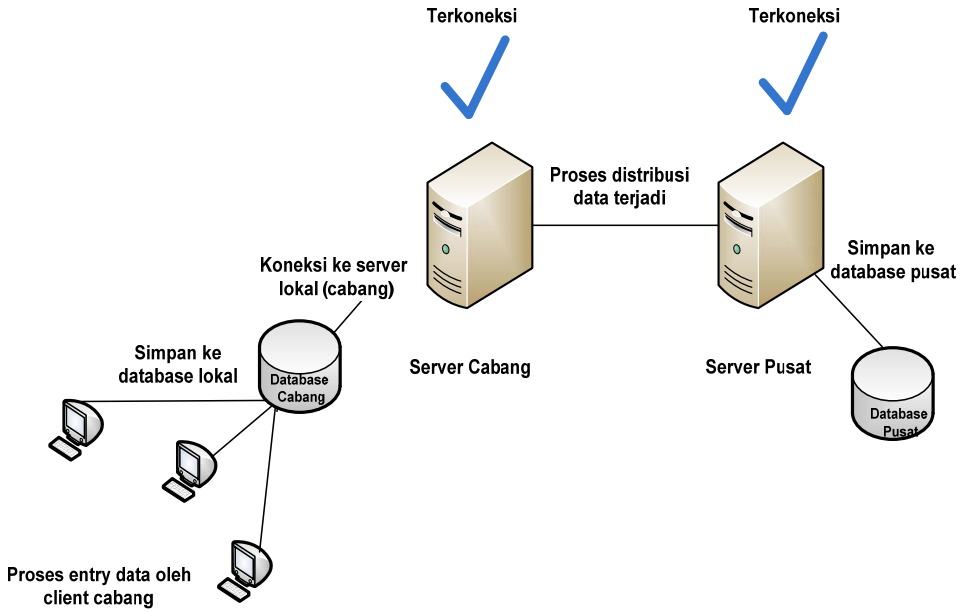
database lokal. Ketika *server* database pusat sudah kembali normal, data bisa didistribusikan kembali.

e. Pada database pusat yang menjadi *server* utama akan digunakan dua *server* yaitu *master* dan *slave*. *Slave* berfungsi untuk menyimpan *back-up* data yang berada di *master*.

Pada saat proses koneksi yang melibatkan kantor cabang dan kantor pusat terdapat 4 kondisi yang akan terjadi.

1. *Server* Cabang dan Pusat Jalan

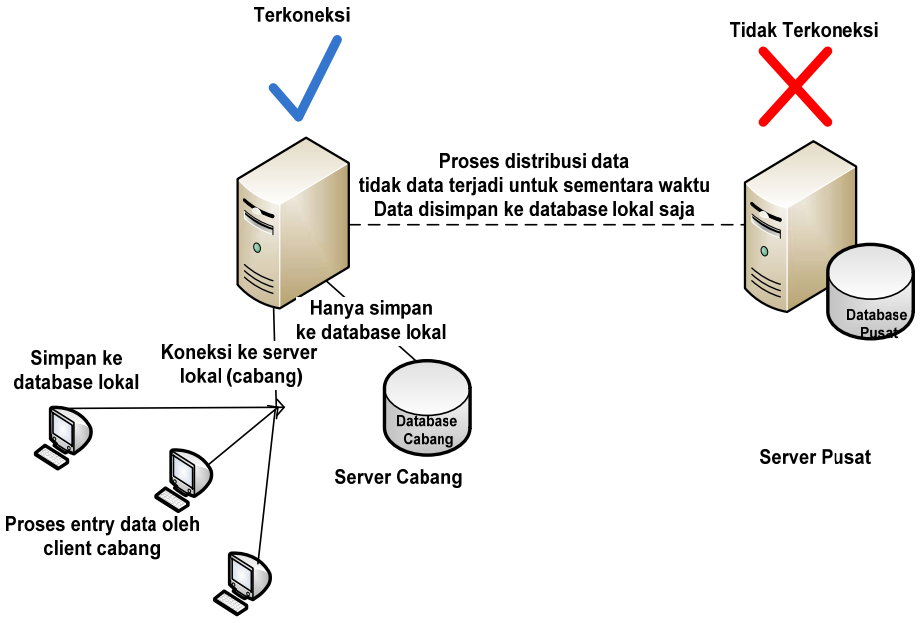
Untuk kondisi ini, proses pendistribusian dapat berjalan dengan baik dikarenakan *server* pusat dan cabang tidak bermasalah. Sehingga kondisi ini dapat dikatakan kondisi paling aman dalam proses pendistribusian barang dikarenakan kedua *server* berjalan sebagaimana mestinya.



Gambar 3.5 Kondisi *Server* Cabang dan Pusat Terhubung

2. *Server* Cabang Jalan tetapi *Server* Pusat Mati

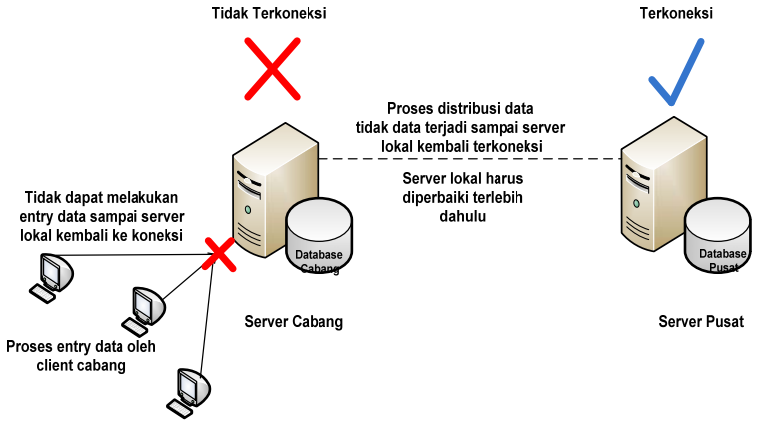
Kondisi kedua yang dapat terjadi jika *server* cabang hidup, tetapi *server* pusat mati atau ada kendala dengan internet. Untuk menangani masalah ini cabang harus melakukan proses pengolahan data pada database lokalnya dahulu. Jadi meskipun tidak terhubung dengan pusat, cabang masih bisa melakukan proses pengolahan data dengan status pengiriman data yang belum terupdate. Sehingga di dalam tabel sementara terdapat status pengiriman data ke pusat apakah sudah berhasil terupdate atau belum. Hal ini merupakan solusi jika sever pusat sedang bermasalah dan proses pengiriman data antara cabang dengan pusat tidak dapat terjadi pada saat itu.



Gambar 3.6 Kondisi *Server* Cabang Jalan dan Pusat Mati

3. *Server* Cabang Mati tetapi *Server* Pusat Hidup

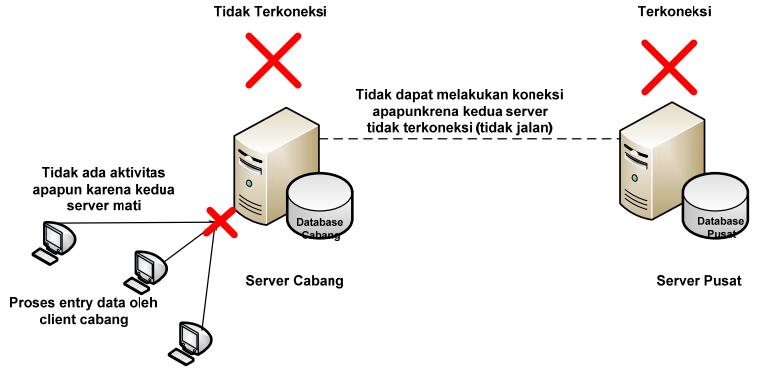
Jika kondisi ini terjadi, *client* yang berada di cabang bisa langsung melakukan proses update data langsung ke kantor pusat melalui jaringan internet.



Gambar 3.7 Kondisi *Server* Cabang Mati dan Pusat Hidup

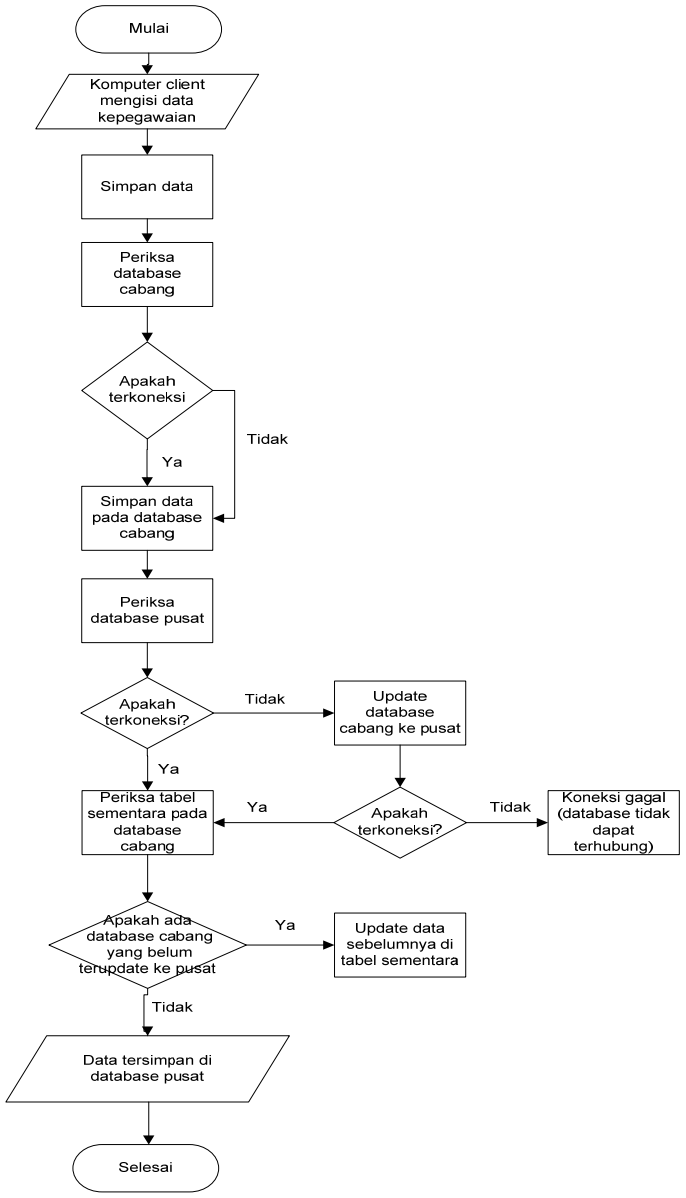
4. *Server* Cabang dan Pusat Mati

Dengan kondisi seperti ini, proses distribusi tidak dapat dilakukan. Karena *server* cabang dan pusat sedang bermasalah dan proses koneksi antar cabang tidak dapat terjadi. Proses distribusi dapat terjadi jika *server* keduanya sudah berjalan normal lagi.



Gambar 3.8 Kondisi *Server* Cabang dan Pusat Mati

Berikut bagan kerja dari proses koneksi :



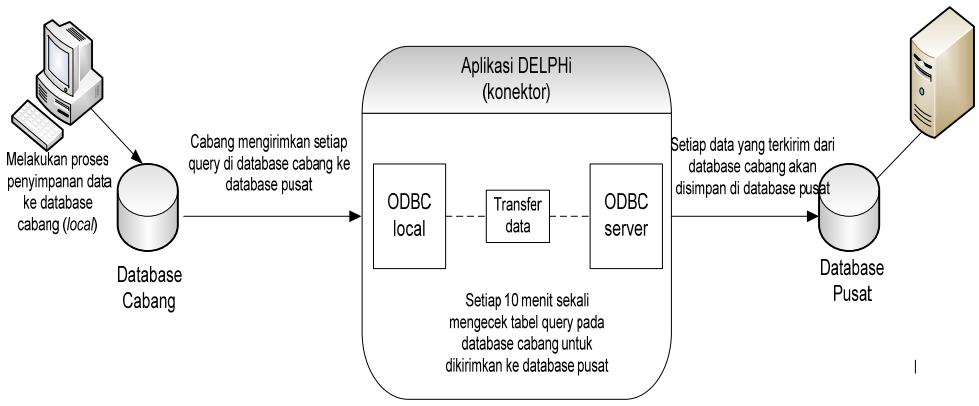
Gambar 3.9 Bagan Kerja Proses Koneksi

Untuk mempercepat proses pendistribusian data antara database kantor cabang dengan kantor pusat maka akan dibangun satu aplikasi tambahan berbasis dekstop sebagai konektor antara database cabang dengan database pusat. Aplikasi

tersebut berperan sebagai penghubung semua query di cabang yang akan disampaikan ke pusat.

3.1.5.2 Konektor Antara Database Cabang dengan Pusat

Dalam proses pengiriman data antara kantor cabang dengan kantor pusat akan dibuat aplikasi tambahan yang berfungsi sebagai konektor antara database pusat dengan database cabang. Aplikasi tersebut berupa aplikasi dekstop yang akan disimpan di setiap kantor cabang. Aplikasi tersebut bekerja secara otomatis untuk mengecek data yag tersimpan di taberl query. Hubungan antara database cabang dengan pusat melalui ODBC yang terdapat pada aplikasi delphi. Setiap per 10 menit, aplikasi itu akan bekerja mengecek ada atau tidaknya data yang belum terkirim ke database pusat. Cara kerja aplikasi tersebut dapat dilihat pada gambar berikut:



Gambar 3.10 Konektor Antara Database Cabang dengan Pusat

Keterangan :

1. Aplikasi konektor dijalankan.
2. Setiap ada proses pengolahan data pada kantor cabang, data akan disimpan ke database cabang terlebih dahulu.

3. Proses query disimpan di tabel query yang ada pada database cabang.
4. Setiap per 10 menit, aplikasi tersebut akan mengecek ada atau tidaknya data yang masih tersimpan di tabel query pada database cabang.
5. Jika ada data yang masih belum terkirim ke database pusat, maka secara otomatis aplikasi konektor akan mengirimkan data tersebut ke database pusat.
6. Proses koneksi antara database cabang dengan database pusat akan menggunakan ODBC yang ada pada aplikasi delphi tersebut.
7. Sehingga secara otomatis, ada atau tidak proses pengolahan data, aplikasi akan mengecek ke tabel query yang ada di database cabang.
8. Semua data yang berhasil dikirim dari database cabang akan disimpan di databse pusat.

Ada beberapa kondisi yang terjadi pada saat proses pendistribusian data. Hal ini berhubungan dengan jumlah record yang berada di cabang dengan pusat.

Kondisi yang dapat terjadi adalah sebagai berikut:

a. Jumlah data di database cabang samadengan jumlah data di database pusat (record cabang = record pusat)

Tidak ada data yang belum terupdate yang tersimpan di database cabang. Kondisi ini tidak bermasalah karena tidak ada perbedaan jumlah data antara database cabang dengan database pusat.

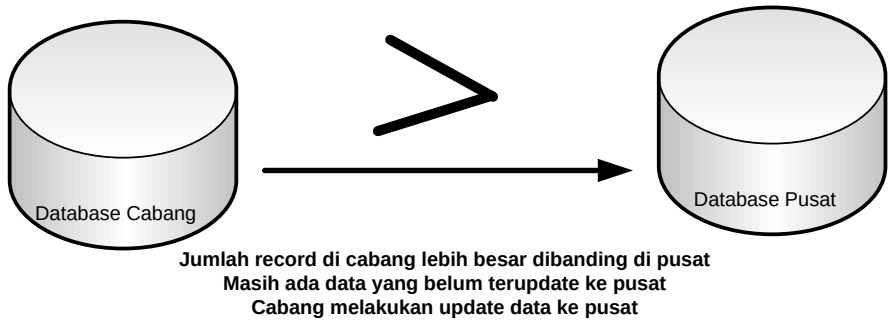


Gambar 3.11 Database Cabang samadengan Database Pusat

b. Jumlah data di database cabang lebih banyak daripada jumlah data di database pusat (record cabang > record pusat).

Kondisi ini terjadi jika data pada cabang masih ada yang belum terupdate ke database pusat. Sehingga ada beberapa data yang statusnya belum terupdate. Jadi ketika cabang dan pusat terkoneksi, cabang akan memeriksa tabel sementara terlebih dahulu yang berisi status pengiriman data. Jika ada data yang masih berstatus belum terupdate, maka cabang akan melakukan proses update data yang sebelumnya dulu, baru mengirim data yang baru diupdate.Sehingga terjadi proses

antrian data yang akan dikirim. Data yang masuk terlebih dahulu ke database cabang adalah yang akan pertama diupdate ke database pusat.



Gambar 3.12 Database Cabang lebih besar dari Database Pusat

c. Jumlah data di databse cabang lebih sedikit daripada data yang ada di database pusat (record cabang < record pusat).

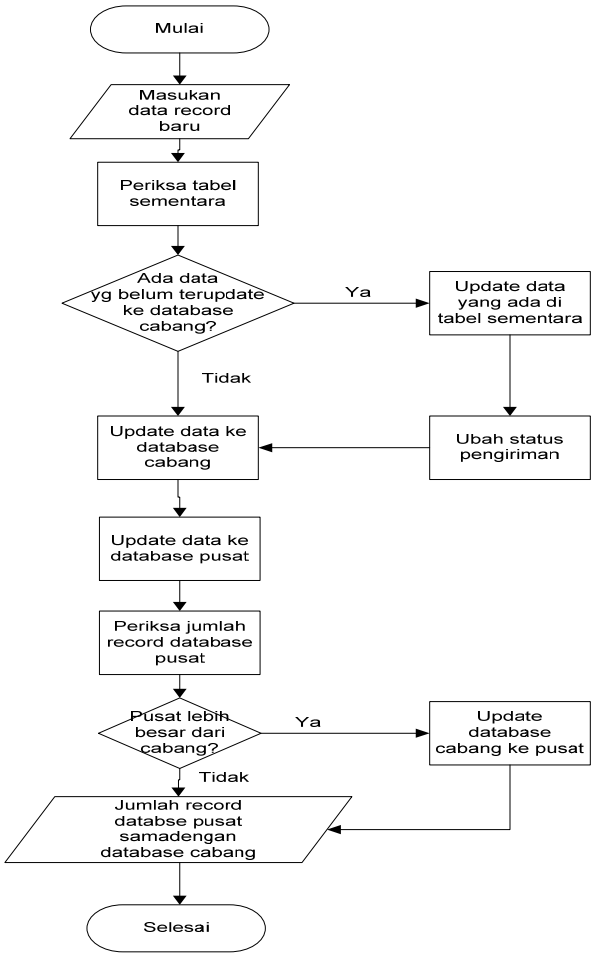
Kondisi ini terjadi jika cabang menghapus salah satu data yang ada di databasenya, tetapi tidak ada konfirmasi ke database pusat bahwa data yang dimaksud telah di hapus atau mungkin terhapus oleh user yang ada di cabang. Sehingga jumlah data yang ada di pusat lebih banyak dibanding data yang ada di cabang. Jika kondisi ini terjadi karena kesalahan user cabang yang menghapus salah satu data yang ada pada databasenya, maka pusat bisa mengirim kembali *back-up* data yang sudah tersimpan di pusat atas permintaan pihak cabang. Sehingga data yang berada di database cabang dan pusat akan sama.



Jumlah record di cabang lebih sedikit dibanding di pusat
Ada data yang terhapus di cabang
Pusat memberikan backup data ke cabang

Gambar 3.13 Database Cabang lebih sedikit dari Database Pusat

Berikut bagan kerja proses update data cabang dengan pusat :



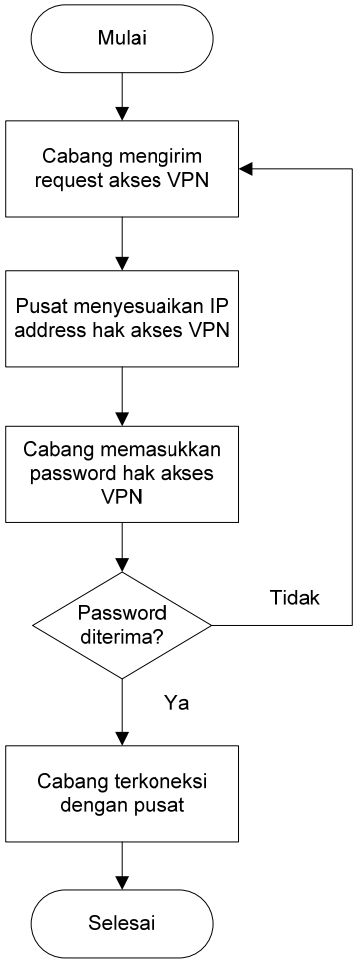
Gambar 3.14 Bagan Kerja Proses Update Database Cabang dengan Pusat

3.1.5.3 Proses Koneksi Hak Akses

Koneksi akan dilakukan menggunakan jaringan internet karena aplikasi yang dibangun berbasis *web*. Sehingga siapa saja dapat mengakses *web* ini dengan syarat terkoneksi jaringan internet. Tetapi dikarenakan data yang didistribusikan bersifat rahasia dan hak akses yang diberikan hanya kepada pegawai yang bekerja pada lingkungan PT. Taspen saja, jadi keamanan data harus sangat diperhatikan.

Oleh karena itu, untuk menjaga keamanan data, maka proses pendistribusian data akan dilakukan melalui jalur khusus yaitu jaringan VPN (*Virtual Private Network*) milik PT. Taspen. Sehingga walaupun bersifat *online*, hanya pegawai PT. Taspen saja yang mempunyai hak akses untuk melakukan pengolahan data.

Hal ini dilakukan agar proses pendistribusian data dapat terjamin kerahasiaannya dan pihak luar tidak dapat mengakses situs ini dengan bebas. Berikut kinerja proses koneksi hak akses menggunakan VPN dapat digambarkan sebagai berikut :



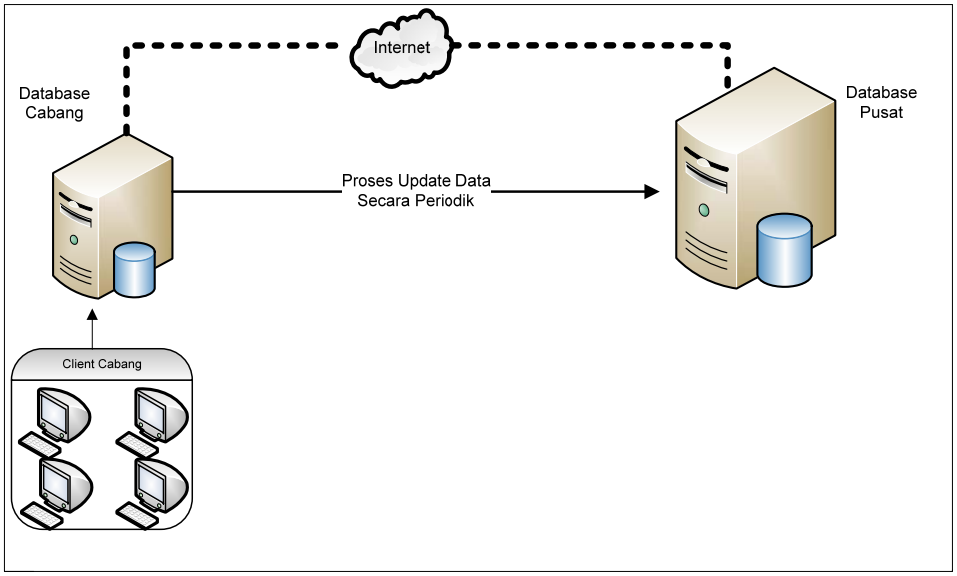
Gambar 3.15 Bagan Kerja Proses Koneksi Hak Akses

3.1.5.4 Proses Distribusi

Aplikasi ini akan menangani pendistribusian data antardatabase. Berikut merupakan karakteristik dari aplikasi yang akan dibangun :

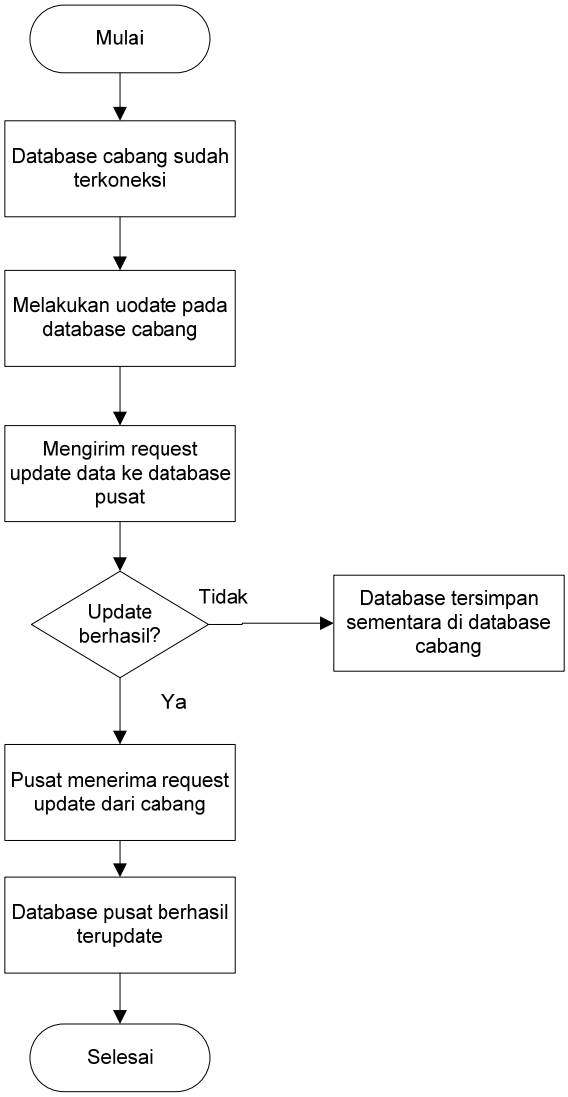
1. Sistem terdiri dari dua aplikasi yaitu aplikasi *server* dan aplikasi *client*. Aplikasi *server* dan aplikasi *server* dijalankan pada komputer dan database yang terpisah.
2. *Client* terdiri dari lima database, sedangkan *server* hanya satu. Antar*client* terpisah jarak antarkota.

3. Aplikasi *client* akan terhubung dengan aplikasi *server* dengan menggunakan koneksi jaringan internet. Kemudian aplikasi *client* akan mengirim data update yang ada pada databasenya ke database pusat.
4. Aplikasi *server* akan banyak menerima data dari banyak *client* dengan database yang berbeda-beda.
5. *Server* akan tetap menerima transaksi update data selama koneksi internet tidak terputus dan database *server* dan *client* tidak bermasalah.
6. Aplikasi merupakan sarana pendistribusian data yang berhubungan dengan keuangan PNS yang diperlukan oleh kantor pusat sebagai perkiraan dana yang akan dikeluarkan untuk melakukan pembayaran dana taspen seorang pensiunan PNS di setiap cabangnya.



Gambar 3.16 Proses Distribusi *Server* Cabang dengan *Server* Pusat

Berikut kinerja proses distribusi data dapat digambarkan sebagai berikut :



Gambar 3.17 Bagan Kerja Proses Distribusi

3.1.5.5 Proses Update data

Berdasarkan wawancara dengan pihak PT. Taspen, proses update data antara kantor cabang dengan kantor pusat akan dilakukan secara periodik. Sehingga proses update akan dilakukan pada saat-saat tertentu saja. Hal ini disesuaikan dengan kebutuhan dan kondisi yang ada di kantor cabang dan pusat

dikarenakan keterbatasan kebutuhan yang membuat proses update data akan dilakukan secara aperiodik.

3.1.5.6 Proses Replikasi

Proses replikasi akan digunakan pada saat data dari semua kantor cabang telah dikirim dan ditampung pada database pusat. Proses ini dilakukan agar database pusat mempunyai backup data pada database lain. Sehingga jika terjadi masalah terhadap isi data yang berada di database pusat, database lain masih mempunyai data yang sama dengan data yang berada di database pusat tersebut.

Melihat kondisi dan kebutuhan pada pihak PT. Taspen, maka proses replikasi akan menggunakan jenis replikasi *asynchronous*. Hal ini dilihat berdasarkan karakteristik yang dimiliki oleh replikasi jenis *asynchronous* yang memungkinkan database *master* dan *slave* tidak selalu terhubung dan dapat dihubungkan pada saat data yang tertampung sudah cukup banyak dan akan dilakukan proses replikasi dari database *master* ke database *slave*. Dikarenakan proses replikasi dilakukan setelah semua transaksi di *master* selesai dilakuakan. Sehingga tidak akan terjadi data yg *crash* pada saat proses replikasi.

Dengan melakukan proses replikasi, database pada *server* akan dibagi menjadi dua dalam database yang berbeda. Yang pertama adalah database *master* yang akan menampung semua database yang telah terkirim dari database cabang, dan yang satu lagi akan berperan sebagai *slave*. Yaitu sebagai *back-up* dari database *master*. Database *slave* akan terhubung jika dibutuhkan untuk proses replikasi dari database *master*. Proses replikasi itu sendiri terjadi sesuai kebutuhan dan berdasarkan admin yang mengelola database *server* tersebut. Jika dijabarkan

dalam hitungan waktu, proses replikasi bisa dilakukan minimal selama 1 bulan sekali. Agar kekonsistensian data bisa terjaga.

Dilihat dari karakteristik masing-masing replikasi, dapat dibedakan sesuai fungsi dan kebutuhannya. Perbedaan Replikasi *Synchronous* dengan Replikasi *Asynchronous* adalah sebagai berikut :

Replikasi *Synchronous*

1. Jika menggunakan replikasi jenis ini, database *master* dan *slave* harus selalu terhubung dan dilakukan secara real time.
2. Ketika kantor cabang melakukan proses pengiriman data, maka secara otomatis *slave* akan mengikuti aktivitas transaksi yang terjadi pada saat itu juga. Sehingga proses replikasi akan terjadi jika kedua database sedang aktif dan dapat digunakan.
3. Kelemahan dari replikasi jenis ini, jika database *slave* mati atau tidak terhubung, maka database pada *master* tidak dapat tereplikasi ke database *slave*. Sehingga tidak ada *back-up* data ke database *slave*.

Replikasi *Asynchronous*

1. Proses yang ada pada jenis *asynchronous* ini sesuai dengan kebutuhan PT. Taspen. Karena pada saat kantor cabang melakukan proses pengiriman data, pada saat yang bersamaan databasa *master* pusat tidak selalu harus terhubung dengan database *slave*nya. Sehingga ini memudahkan pihak admin untuk menunggu semua transaksi pengiriman selesai terlebih dahulu, kemudian melakukan replikasi setelah proses transaksi di *master* selesai.

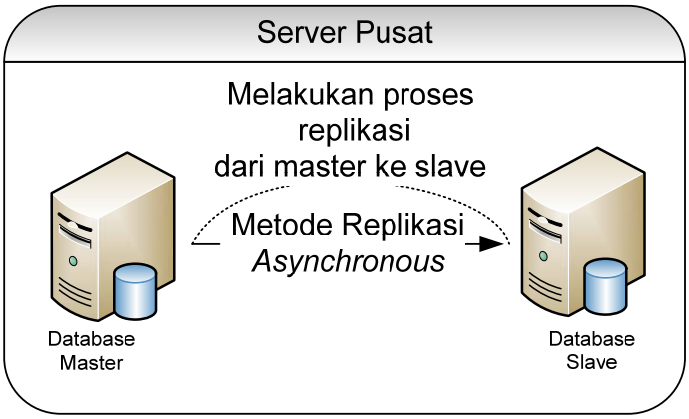
2. Kelebihan replikasi *asynchronous* yaitu jika terjadi kegagalan pada *server* tidak akan mempengaruhi operasi dari *server* lain, tetapi akan menambah pekerjaan dari aplikasi untuk memelihara integritas data.
3. Kelemahan dari replikasi *asynchronous* yaitu tidak menjamin kesinkronan data apabila salah satu *site* atau node mengalami crash saat replikasi belum selesai dilaksanakan.

Melihat karakteristik dari masing-masing jenis replikasi, maka untuk proses replikasi pada database pusat, akan dilakukan secara *asynchronous*. Karena setelah membandingkan, jenis replikasi yang paling cocok diterapkan pada prosses replikasi di PT. Taspen ini adalah replikasi *asynchronous*.

Berikut merupakan karakteristik dari proses replikasi database *master* dengan database *slave*:

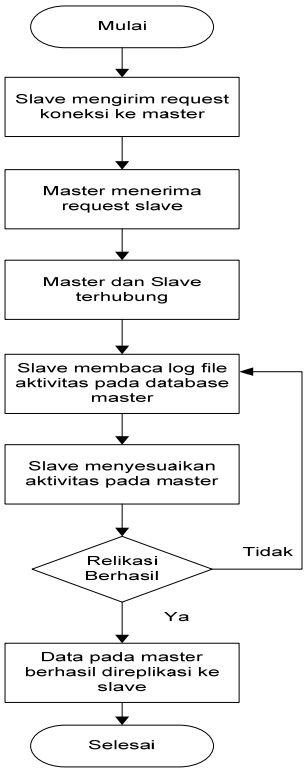
1. Menentukan database mana yang akan dijadikan *master* dan database mana yang akan dijadikan *slave.*
2. Replikasi akan dilakukan setelah transaksi di *master* selesai.
3. *Slave* akan mengirim request untuk melakukan sinkronisasi database.
4. Setelah *master* menerima permintaan dari *slave*, maka *master* dan *slave* telah berhasil terkoneksi.
5. *Slave* akan membaca setiap aktivitas yang telah terjadi di database *master* dan akan mengikuti aktivitas yang sudah terjadi di *master*.
6. *Slave* akan mulai mereplikasi data-data yang ada pada *master*.
7. Jika replikasi berhasil, maka data yang ada di database *master* sudah tereplikasi pada database *slave*.

8. Jika tidak berhasil dikarenakan koneksi terputus dll, maka *slave* akan membaca *log file master* dan kemudian menyesuaikannya kembali.



Gambar 3.18 Proses Replikasi menggunakan Metode Replikasi *Asynchronous*

Berikut kinerja proses replikasi data dapat digambarkan sebagai berikut :



Gambar 3.19 Bagan Kerja Proses Replikasi

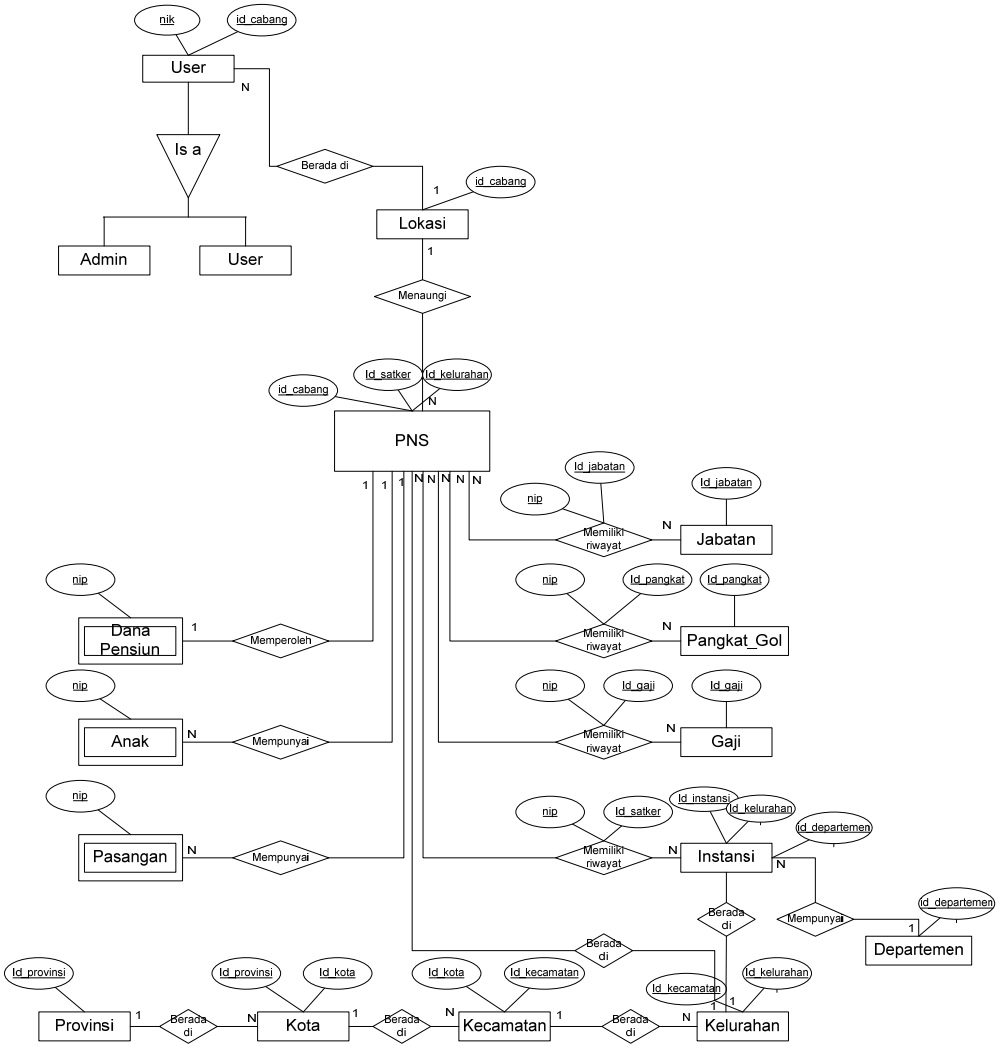
3.1.6 Analisis Basis Data

Usulan untuk diagram E-R yaitu terdapat kunci yang unik (*primary key*) pada setiap entitas (table induk) yang dapat membedakan dengan atribut lainnya sehingga tabel tersebut dapat dijadikan referensi untuk tabel lainnya.

Dari usulan diatas maka dapat dibuat diagram E-R untuk sistem yang terdiri dari beberapa table yang dibutuhkan untuk menyimpan data. Untuk melihat keterhubungan antar tabel yang ada.

3.1.6.1 *Entitiy Relationship Diagram*

ERD menggambarkan hubungan antara objek data. ERD adalah notasi yang digunakan untuk melakukan aktivitas pemodelan data. ERD pengolahan data pegawai ini dapat digambarkan seperti pada Gambar 3.3 di bawah ini :

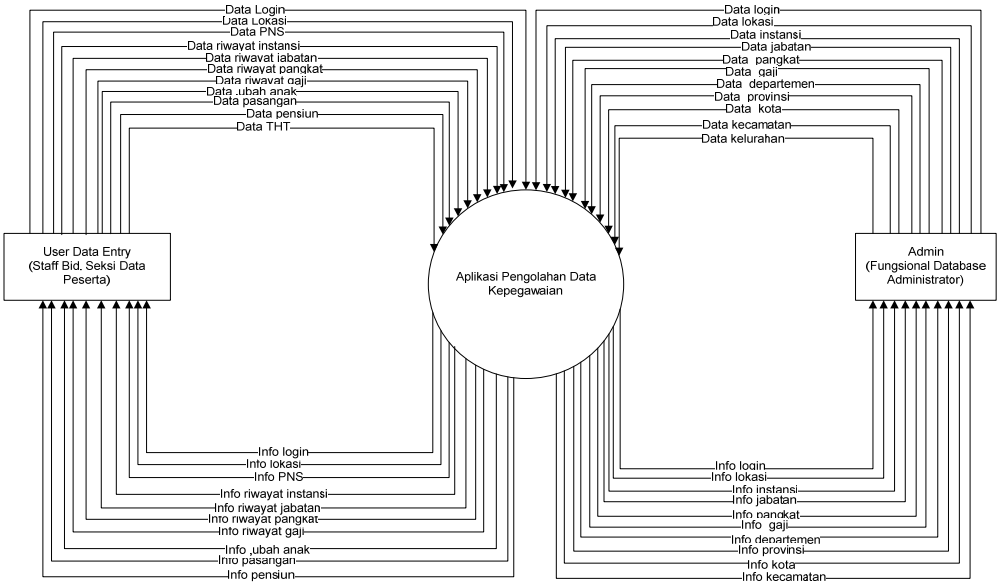


Gambar 3.20 ERD Pengolahan Data Kepegawaian

3.1.7 Analisis Kebutuhan Fungsional

3.1.7.1. Diagram Konteks

Sistem yang dibangun digambarkan secara garis besar dengan menggunakan diagram konteks. Dari diagram ini dapat dilihat secara umum mengenai alur proses yang ditangani sistem.

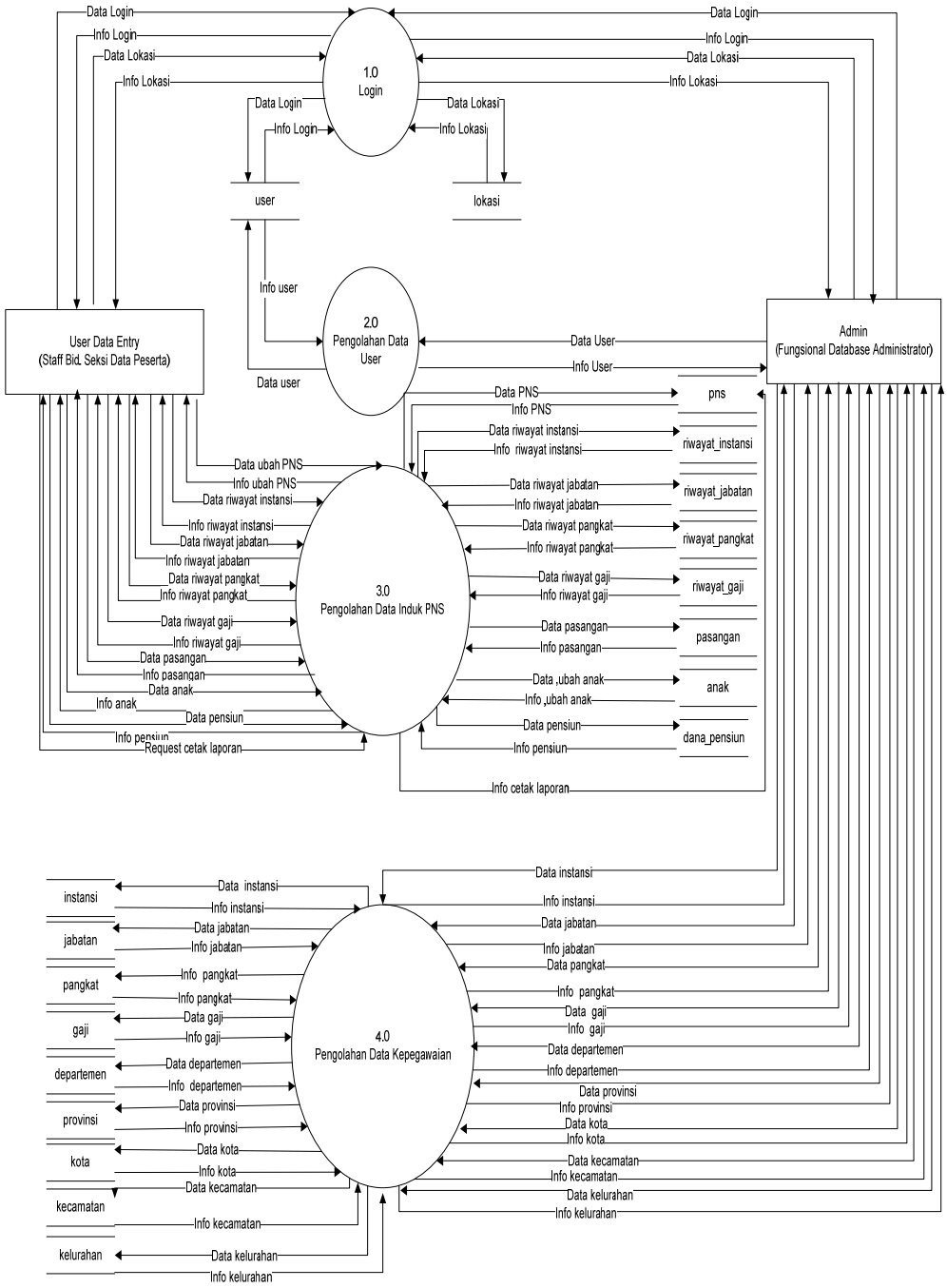


Gambar 3.21 Diagram Konteks

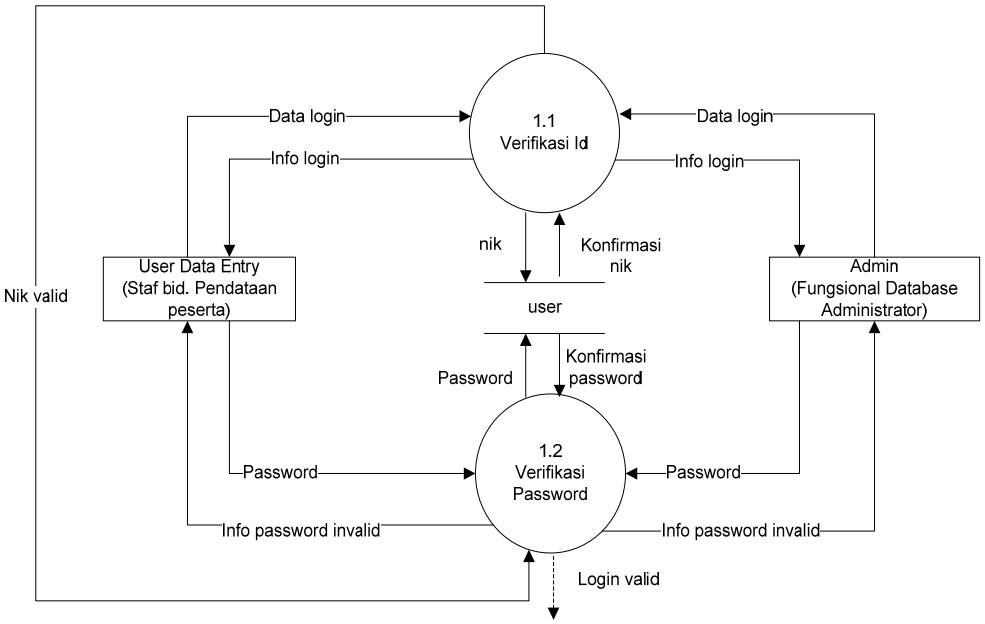
3.1.7.2. *Data Flow Diagram*

DFD sering digunakan untuk menggambarkan suatu sistem yang telah ada atau sistem baru yang akan dikembangkan secara logika tanpa mempertimbangkan lingkungan fisik dimana data tersebut mengalir atau lingkungan fisik dimana data tersebut akan disimpan. DFD dapat menggambarkan arus data di dalam sistem dengan terstruktur dan jelas.

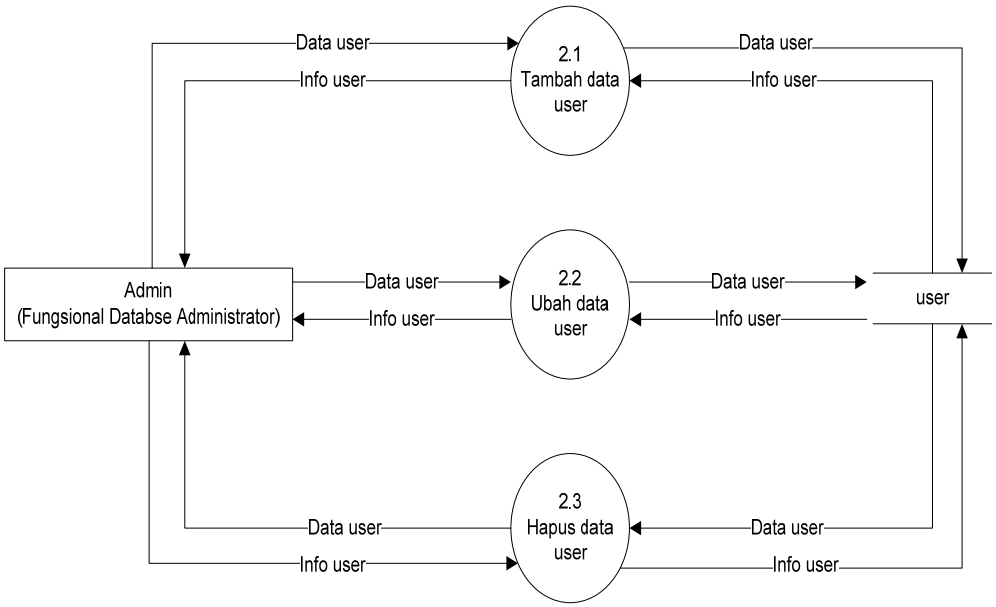
Berikut adalah DFD dari Aplikasi Pengolahan Data Pegawai Pada PT. Taspen :



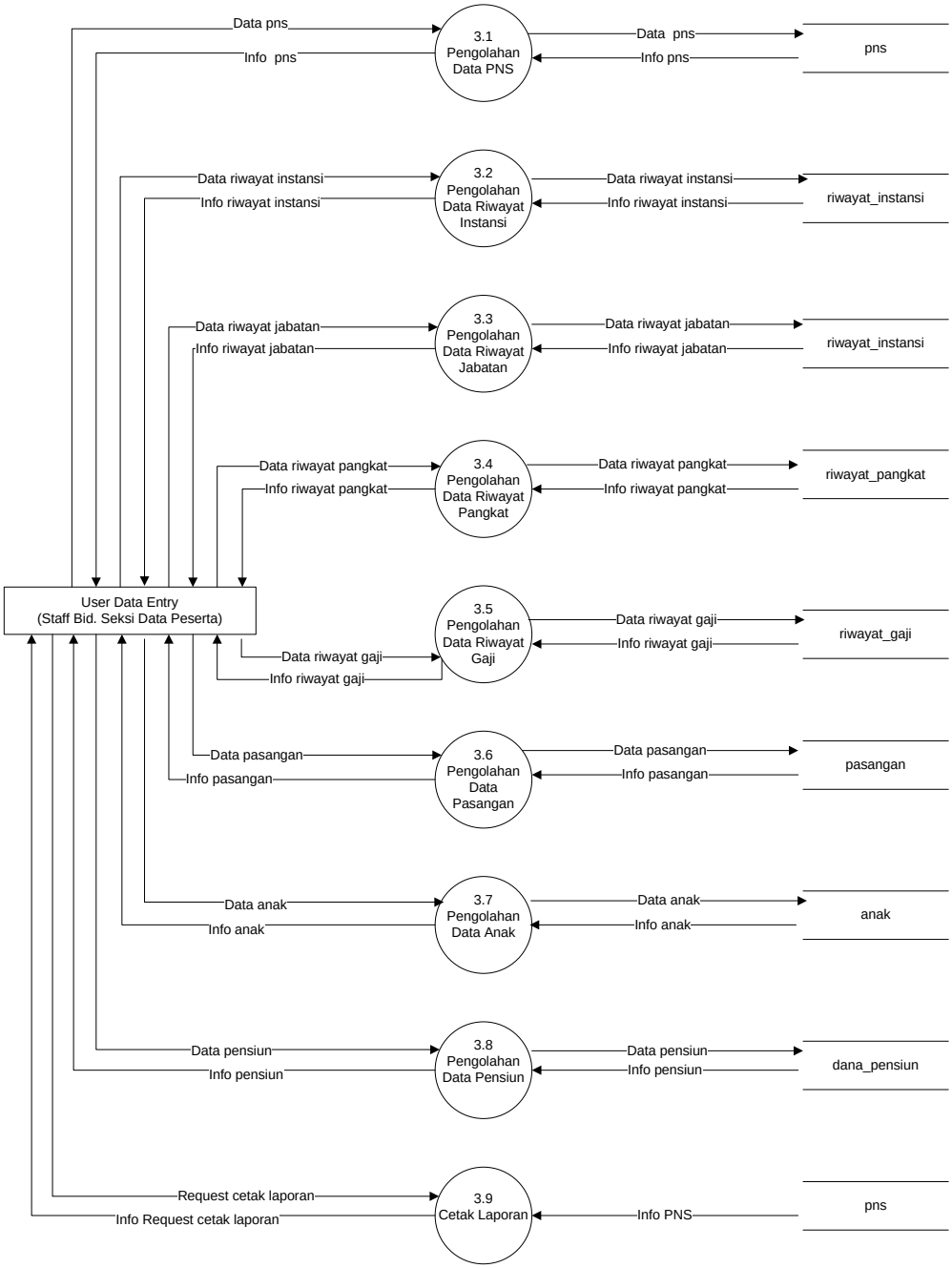
Gambar 3.22 DFD Level 1



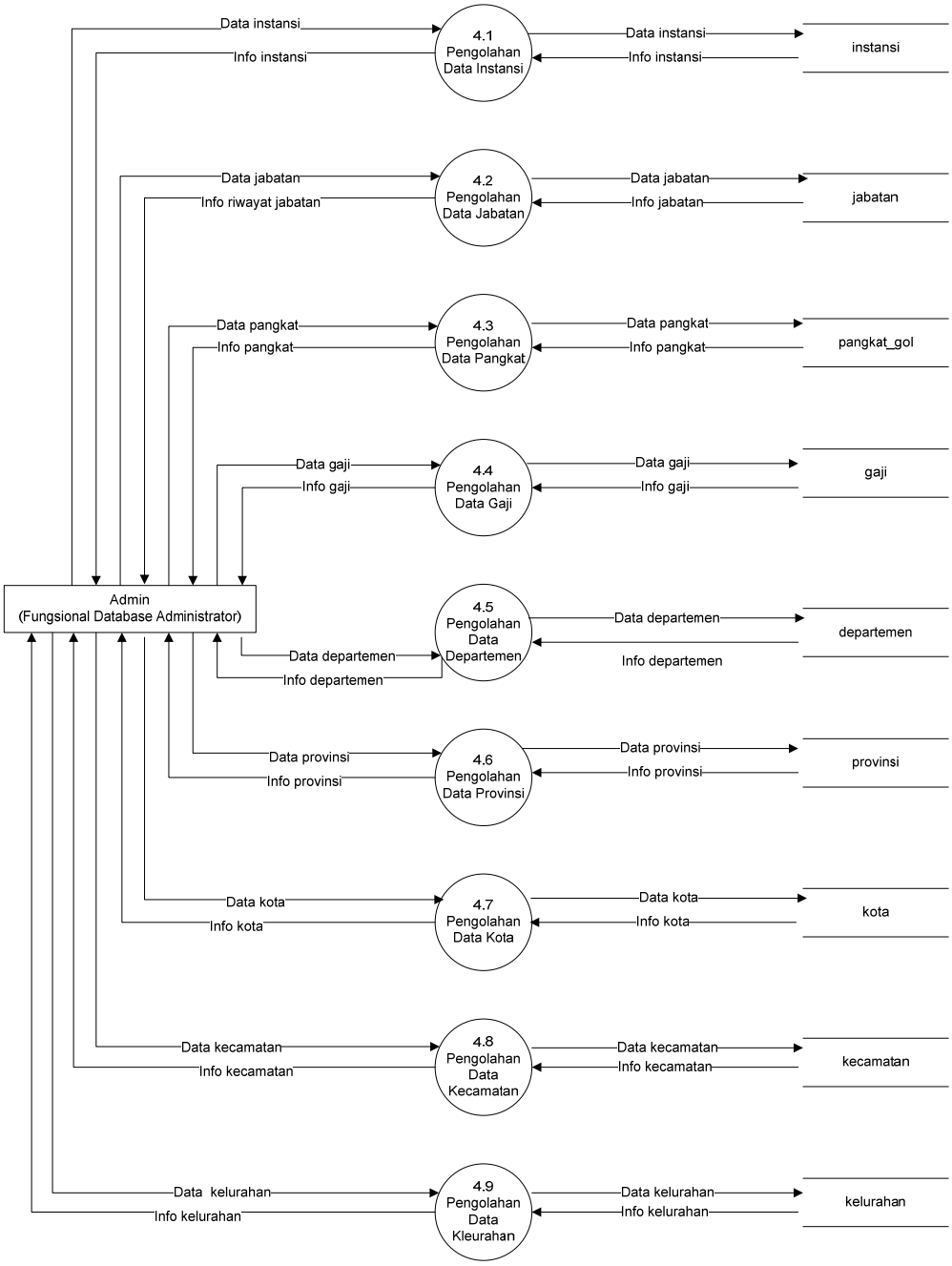
Gambar 3.23 DFD Level 2 Proses Login



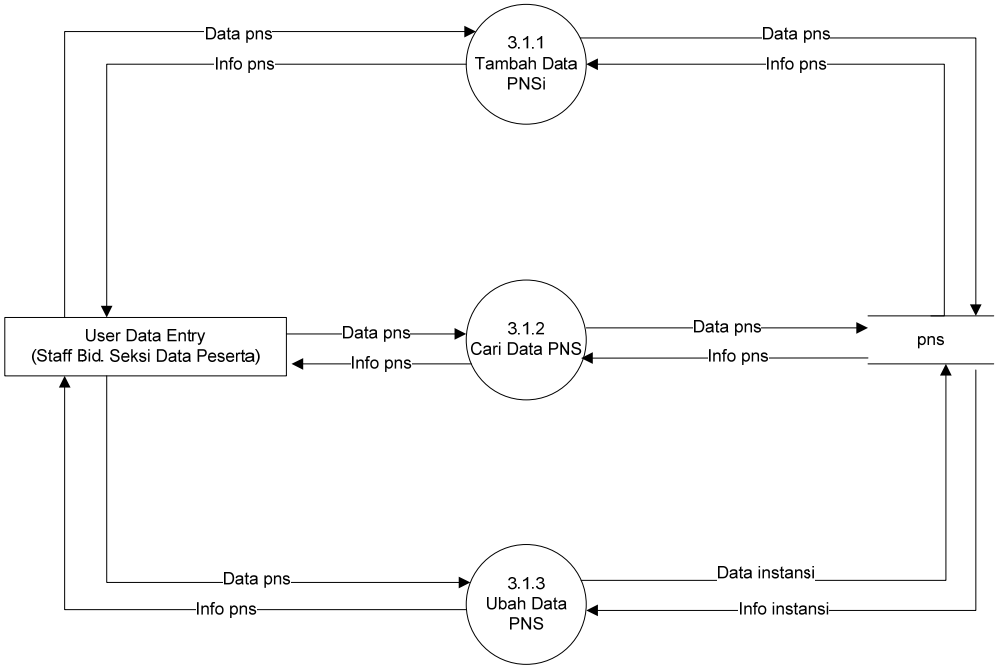
Gambar 3.24 DFD Level 2 Proses Pengolahan Data User



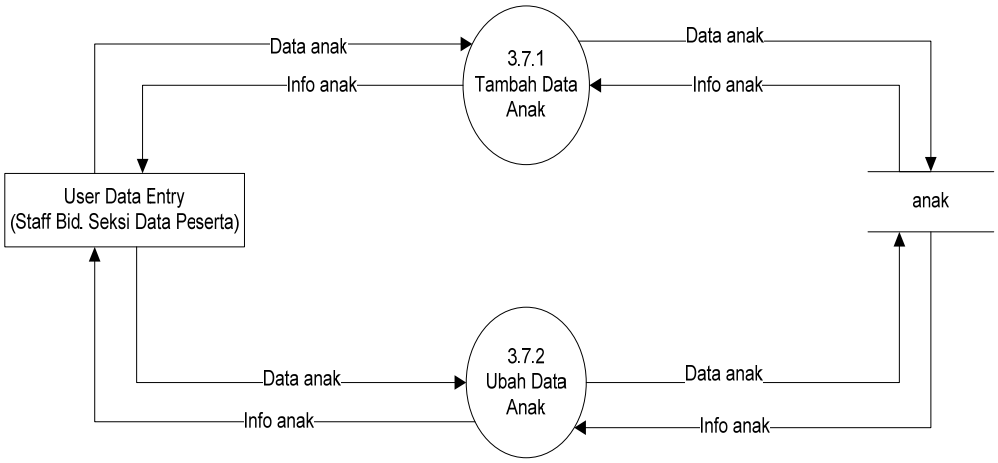
Gambar 3.25 DFD Level 2 Proses Pengolahan Data Induk PNS



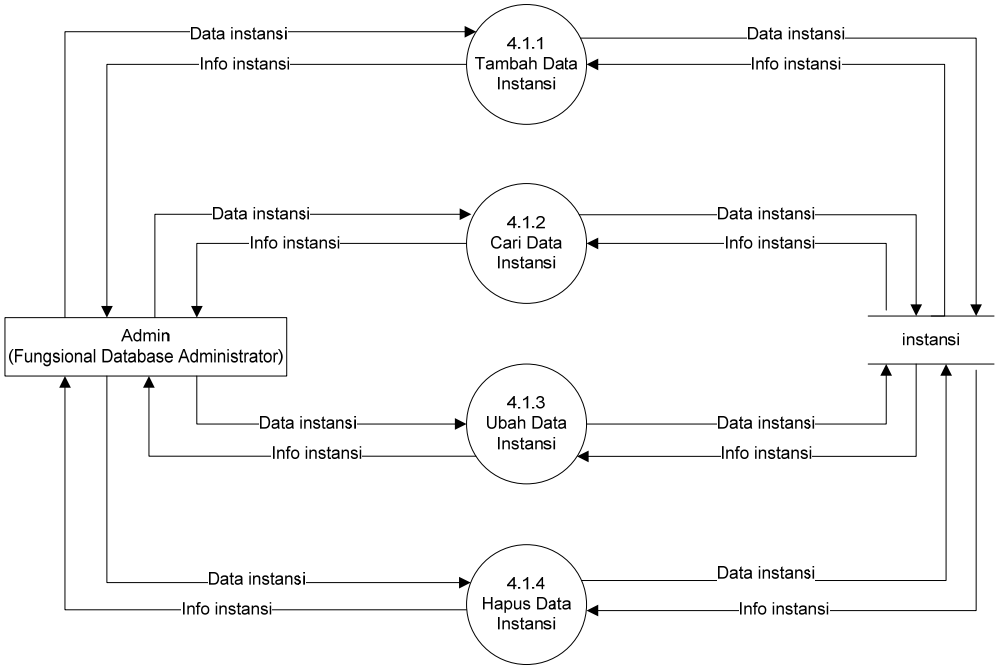
Gambar 3.26 DFD Level 2 Proses Pengolahan Data Kepegawaian



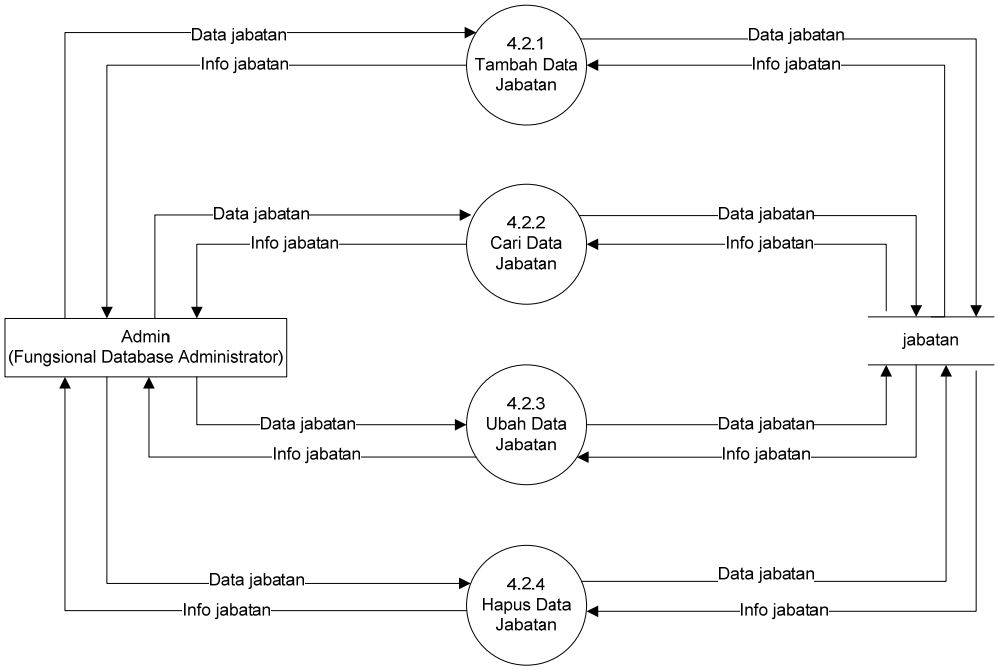
Gambar 3.27 DFD Level 3 Proses Pengolahan Data PNS



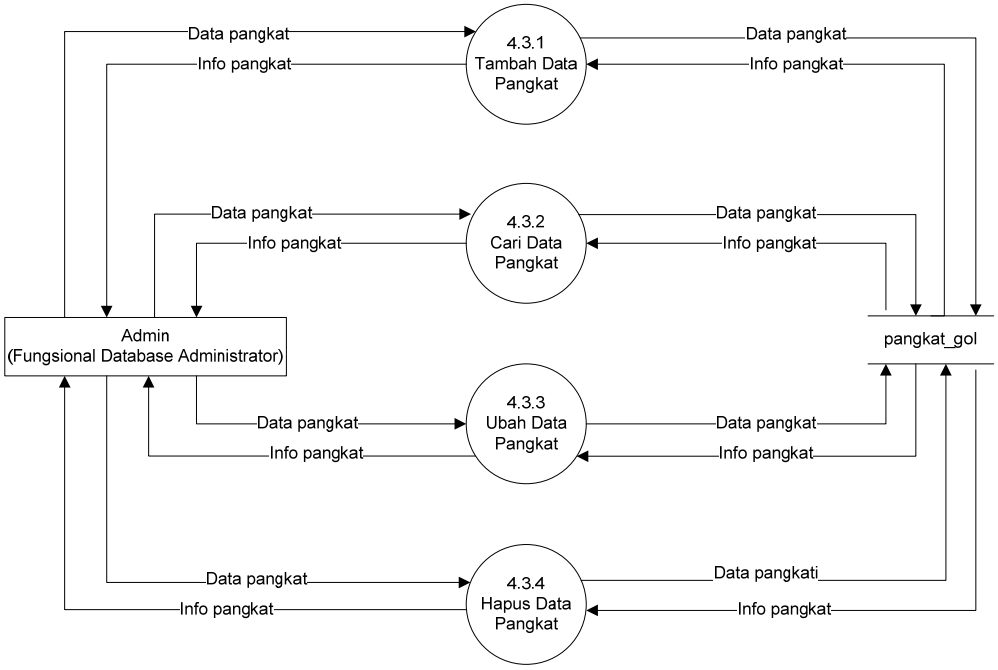
Gambar 3.28 DFD Level 3 Proses Pengolahan Data Anak



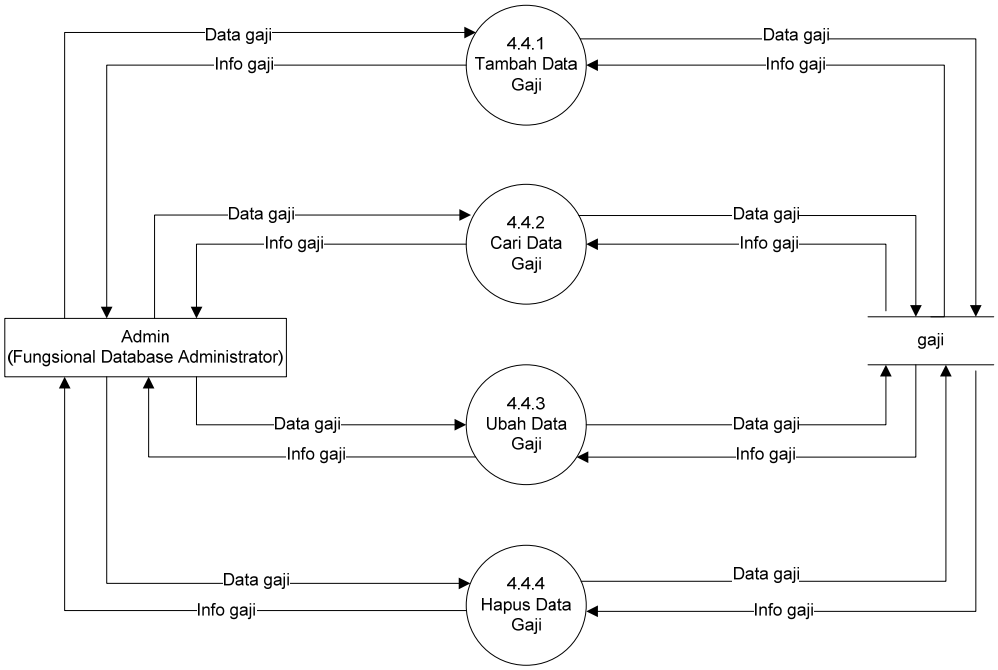
Gambar 3.29 DFD Level 3 Proses Pengolahan Data Instansi



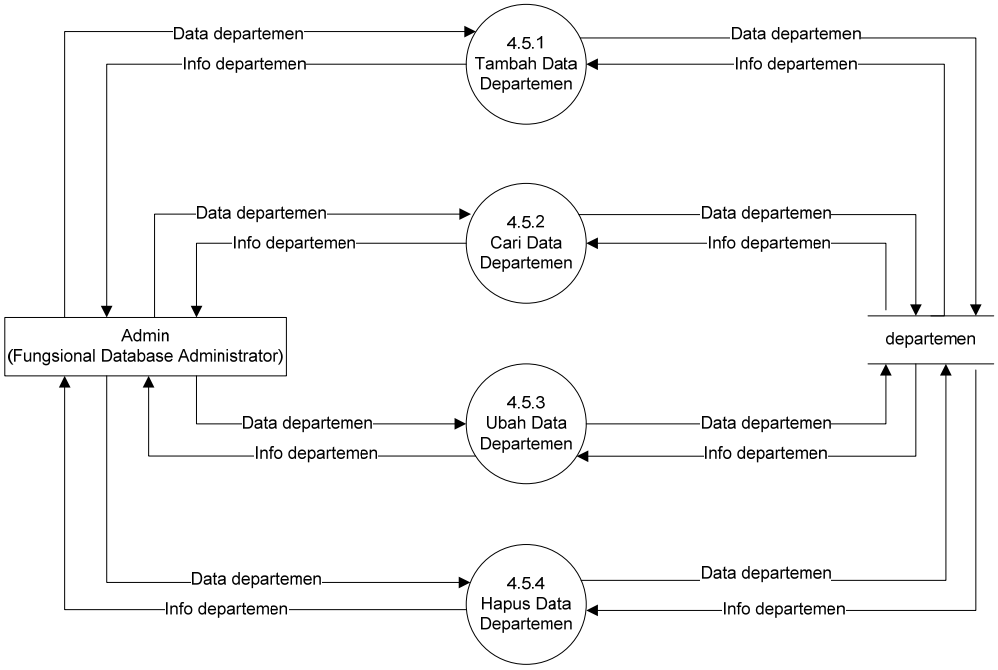
Gambar 3.30 DFD Level 3 Proses Pengolahan Data Jabatan



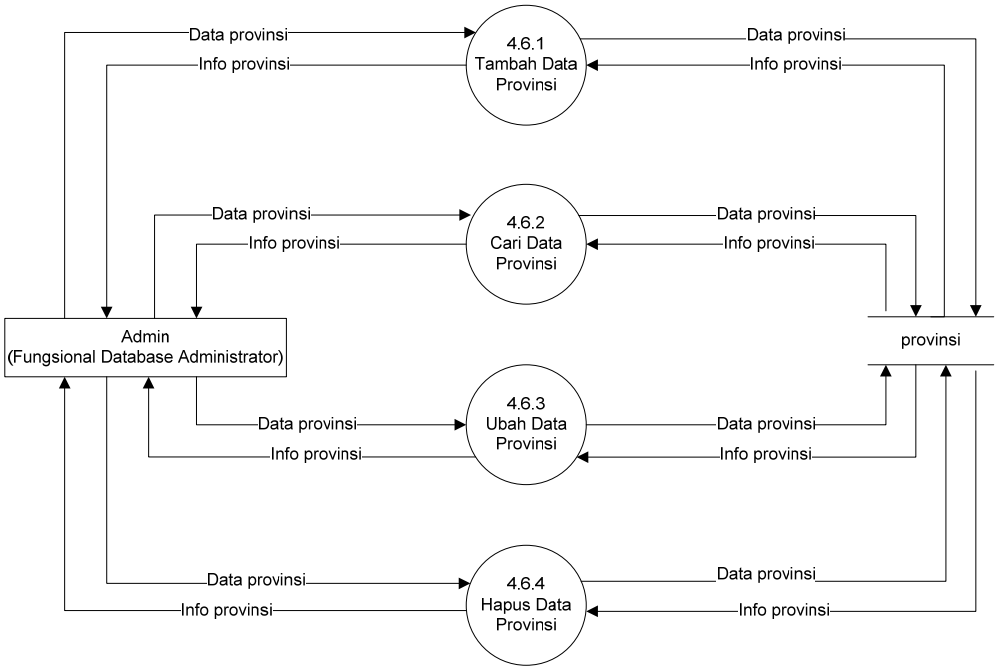
Gambar 3.31 DFD Level 3 Proses Pengolahan Data Pangkat



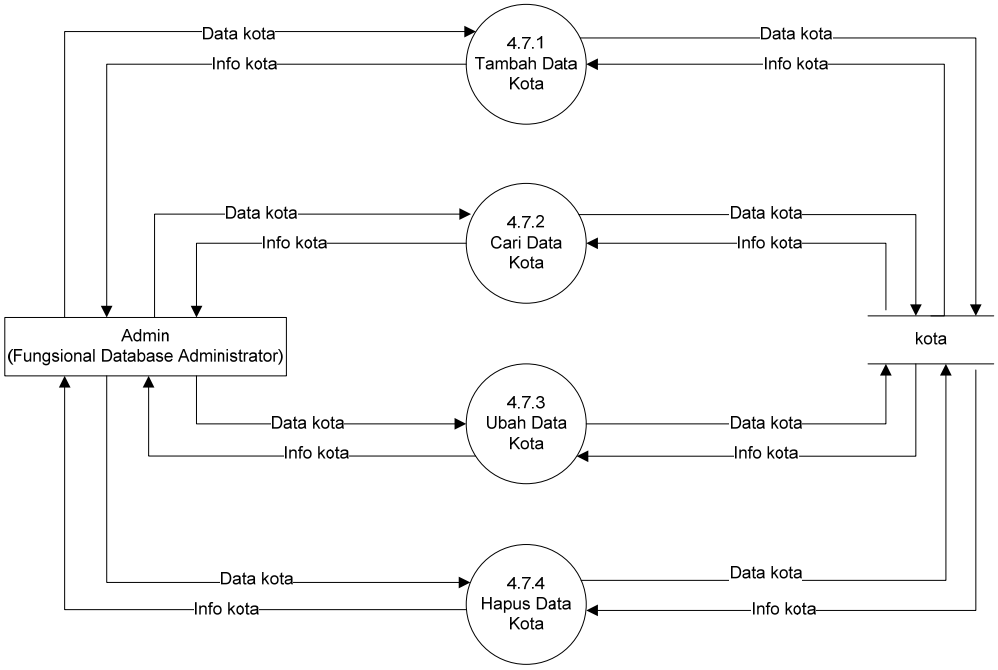
Gambar 3.32 DFD Level 3 Proses Pengolahan Data Gaji



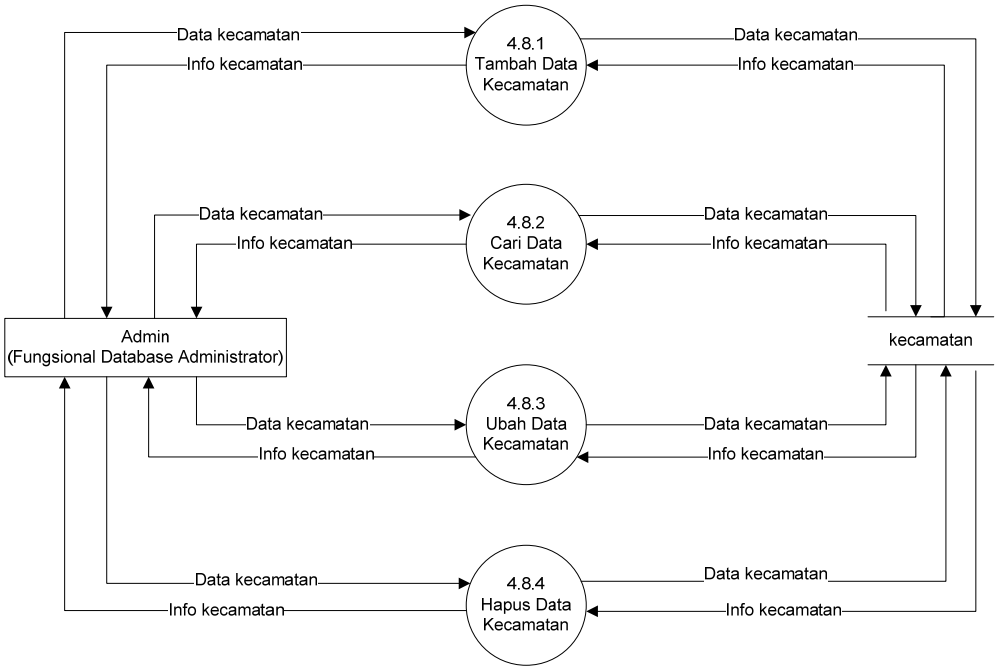
Gambar 3.33 DFD Level 3 Proses Pengolahan Data Departemen



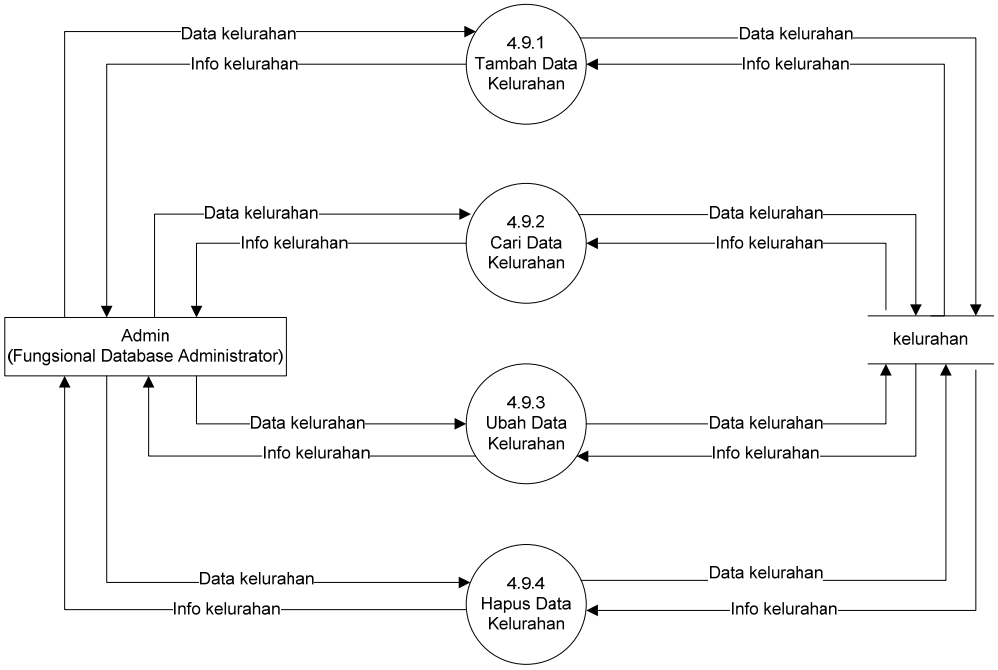
Gambar 3.34 DFD Level 3 Proses Pengolahan Data Provinsi



Gambar 3.35 DFD Level 3 Proses Pengolahan Data Kota



Gambar 3.36 DFD Level 3 Proses Pengolahan Data Kecamatan



Gambar 3.37 DFD Level 3 Proses Pengolahan Data Kelurahan

3.1.7.3. Spesifikasi Proses

Spesifikasi proses merupakan alat bantu (*tools*) sistem yang akan menjelaskan perilaku-perilaku proses yang ada dalam diagram aliran data. Berikut adalah spesifikasi proses dari Aplikasi Pengolahan Data Kepegawaian pada PT. Taspen:

Tabel 3.1 Spesifikasi Proses

No.	Proses	Keterangan
1.	No. Proses	1.0
	Nama Proses	Login
	Source	Admin, User Data Entry
	Input	Data Login
	Output	Info Loin
	Destination	Admin, User Data Entry

	Logika Proses	**begin**
		{Admin dan user data entry memasukkan nik dan password}
		ifnik ada **then**
		if Password ada **then**
		login valid
		else
		tampil informasi login invalid

		end
2.	No. Proses	1.1
	Nama Proses	Pengolahan Data User
	Source	Admin, User Data Entry
	Input	Data Login
	Output	Info Login
	Destination	Admin, User Data Entry

	Logika Proses	**begin**
		{Admin dan user data entry memasukkan nik}
		ifnik ada **then**
		periksa password
		else
		tampil informasi ID_pengguna invalid
		end

3.	No. Proses	1.2
	Nama Proses	Verifikasi Password
	Source	Admin, User Data Entry
	Input	Data Login
	Output	Informasi password invalid
	Destination	Admin, User Data Entry

	Logika Proses	**begin**
		{ Admin dan user data entry memasukkan password}
		if password ada **then**
		login valid
		else
		tampil informasi password invalid
		end

4.	No. Proses	2.0
	Nama Proses	Pengolahan Data User
	Source	Admin
	Input	Data tambah,ubah,hapus user
	Output	Info tambah,ubah,hapus user
	Destination	Admin
	Logika Proses	**begin** {Admin mengolah data user} **End**
5.	No. Proses	2.1
	Nama Proses	Tambah Data User
	Source	Admin
	Input	Data tambah user
	Output	Info tambah user
	Destination	Admin
	Logika Proses	**begin** {Admin menambah user baru} **End**
6.	No. Proses	2.2
	Nama Proses	Ubah Data User
	Source	Admin
	Input	Data ubah user
	Output	Info ubah user
	Destination	Admin
	Logika Proses	**begin** {Admin melakukan proses ubah data user} **End**

7.	No. Proses	2.3
	Nama Proses	Hapus Data User
	Source	Admin
	Input	Data hapus user
	Output	Info hapus user
	Destination	Admin
	Logika Proses	**begin** {Admin menghapus data user} **end**
8.	No. Proses	3.0
	Nama Proses	Pengolahan Data Induk PNS
	Source	User Data Entry
	Input	Data tambah,cari,ubah PNS, Data tambah riwayat instansi, Data tambah riwayat jabatan,Data tambah,ubah anak, Data ubah pasangan, Data tambah riwayat instansi, Data tambah gaji, Data ubah data pensiun, Data ubah dana pensiun
	Output	Info tambah,cari,ubah PNS, Info tambah riwayat instansi, Info tambah riwayat jabatan, Info tambah,ubah anak, Info ubah pasangan, Info tambah riwayat instansi, Info tambah gaji, Info ubah data pensiun, Info ubah dana pensiun
	Destination	User Data Entry
	Logika Proses	**begin** {Admin mengolah data induk PNS} **End**
9.	No. Proses	3.1
	Nama Proses	Pengolahan Data PNS
	Source	User Data Entry
	Input	Data tambah,cari,ubah PNS
	Output	Info tambah,cari,ubah PNS
	Destination	User Data Entry
	Logika Proses	**begin** { User data entrymengolah data PNS} **End**
10.	No. Proses	3.1.1
	Nama Proses	Tambah Data PNS
	Source	User Data Entry
	Input	Data tambah PNS
	Output	Info tambah PNS
	Destination	User Data Entry
	Logika Proses	**begin** {User data entrymenambah data PNS baru} **End**
11.	No. Proses	3.1.2
	Nama Proses	Cari Data PNS
	Source	User data entry
	Input	Data cari PNS
	Output	Info cari PNS
	Destination	User data entry

	Logika Proses	**begin**
		{User data entry mencari data PNS yang akan diubah dan dihapus }
		if data berhasil ditemukan **then**
		data diubah atau dihapus
		else

		data tidak ditemukan **end**
12.	No. Proses	3.1.3
	Nama Proses	Ubah Data PNS
	Source	User data entry
	Input	Data ubah PNS
	Output	Info ubah PNS
	Destination	User data entry

	Logika Proses	**begin**
		if data PNS benar **and** tidak ada data kosong **then**
		simpan hasil perubahan
		else
		perubahan data tidak berhasil
		end

13.	No. Proses	3.2
	Nama Proses	Pengolahan Data Riwayat Instansi
	Source	User data entry
	Input	Data tambah riwayat instansi
	Output	Info tambah riwayat instansi
	Destination	User data entry
	Logika Proses	**begin** { User data entrymenambah data riwayat instansi} **End**
14.	No. Proses	3.3
	Nama Proses	Pengolahan Data Riwayat Jabatan
	Source	User data entry
	Input	Data tambah riwayat jabatan
	Output	Info tambah riwayat jabatan
	Destination	User data entry
	Logika Proses	**begin** { User data entry menambah data riwayat jabatan} **End**
15.	No. Proses	3.4
	Nama Proses	Pengolahan Data Riwayat Pangkat
	Source	User data entry
	Input	Data tambah riwayat pangkat
	Output	Info tambah riwayat pangkat
	Destination	User data entry
	Logika Proses	**begin** { User data entrymenambah data riwayat pangkat} **End**
16.	No. Proses	3.5
	Nama Proses	Pengolahan Data Riwayat Gaji
	Source	User data entry
	Input	Data tambah riwayat gaji
	Output	Info tambah riwayat gaji
	Destination	User data entry
	Logika Proses	**begin** { User data entrymengolah data riwayat gaji} **End**
17.	No. Proses	3.6
	Nama Proses	Pengolahan Data Pasangan
	Source	User data entry
	Input	Data ubahpasangan

	Output	Info ubah pasangan
	Destination	User data entry
	Logika Proses	**begin** { User data entrymengubah data pasangan} **End**
18.	No. Proses	3.7
	Nama Proses	Pengolahan Data Anak
	Source	User data entry
	Input	Data tambah,ubah data anak
	Output	Info tambah,ubah data anak
	Destination	User data entry
	Logika Proses	**begin** { User data entrymengolah data dana THT} **End**
19.	No. Proses	3.7.1
	Nama Proses	Tambah Data Anak
	Source	User data entry
	Input	Data tambah dana THT
	Output	Info tambah dana THT
	Destination	User data entry
	Logika Proses	**begin** {User data entrymenambah data THT } **End**
20.	No. Proses	3.7.2
	Nama Proses	Ubah Data Anak
	Source	User data entry
	Input	Data ubah anak
	Output	Info ubah anak
	Destination	User data entry
	Logika Proses	**Begin** {User data entrymengubah jenis tunjangan anak } **End**
21.	No. Proses	3.8
	Nama Proses	Pengolahan Data Pensiun
	Source	User data entry
	Input	Data ubah data pensiun
	Output	Info ubah data pensiun
	Destination	User data entry
	Logika Proses	**Begin** { User data entrymengubah data pensiun} **End**
22.	No. Proses	3.9
	Nama Proses	Cetak Laporan
	Source	User data entry
	Input	Request cetak laporan
	Output	Info request cetak laporan
	Destination	User data entry
	Logika Proses	**Begin** { User data entry menceak laporan} **End**
23.	No. Proses	4.0
	Nama Proses	Pengolahan Data Kepegawaian
	Source	Admin
	Input	Data tambah,cari,ubah hapus instansi, Data tambah,cari,ubah

		hapus jabatan, Data tambah,cari,ubah hapus pangkat, Data tambah,cari,ubah hapus gaji, Data tambah,cari,ubah hapus departemen, Data tambah,cari,ubah hapus provinsi, Data tambah,cari,ubah hapus kota, Data tambah,cari,ubah hapus kecamatan, Data tambah,cari,ubah hapus kelurahan.
	Output	Info tambah,cari,ubah hapus instansi, Info tambah,cari,ubah hapus jabatan, Info tambah,cari,ubah hapus pangkat, Info tambah,cari,ubah hapus gaji, Info tambah,cari,ubah hapus departemen, Info tambah,cari,ubah hapus provinsi, Info tambah,cari,ubah hapus kota, Info tambah,cari,ubah hapus kecamatan, Info tambah,cari,ubah hapus kelurahan.
	Destination	Admin
	Logika Proses	**Begin** { Admin mengolah data kepegawaian} **End**
24.	No. Proses	4.1
	Nama Proses	Pengolahan Data Instansi
	Source	Admin
	Input	Data tambah,cari,ubah,hapus instansi
	Output	Info tambah,cari,ubah,hapus instansi
	Destination	Admin
	Logika Proses	**Begin** { Adminmengolah data instansi} **End**
25.	No. Proses	4.1.1
	Nama Proses	Tambah Data Instansi
	Source	Admin
	Input	Data tambah instansi
	Output	Info tambah instansi
	Destination	Admin
	Logika Proses	**Begin** {Adminmenambah data instansi} **End**
26.	No. Proses	4.1.2
	Nama Proses	Cari Data Instansi
	Source	Admin
	Input	Data cari instansi
	Output	Info cari instansi
	Destination	Admin

	Logika Proses	**begin**
		{Admin mencari data instansi }
		if data ada **then** data berhasil ditemukan
		else
		data tidak ditemukan
		end

27.	No. Proses	4.1.3
	Nama Proses	Ubah Data Instansi
	Source	Admin
	Input	Data ubah instansi
	Output	Info ubah instansi
	Destination	Admin

	Logika Proses	**Begin**
		if data instansibenar **and** tidak ada data kosong **then**
		simpan hasil perubahan

		else perubahan data tidak berhasil **end**
28.	No. Proses	4.1.4
	Nama Proses	Hapus Data Instansi
	Source	Admin
	Input	Data hapus instansi
	Output	Info hapus instansi
	Destination	Admin

	Logika Proses	**Begin**
		if data akan yakin dihapus **then**
		data terhapus
		else
		data tidak terhapus
		end

29.	No. Proses	4.2
	Nama Proses	Pengolahan Data Jabatan
	Source	Admin
	Input	Data tambah,cari,ubah,hapus jabatan
	Output	Info tambah,cari,ubah,hapus jabatan
	Destination	Admin
	Logika Proses	**Begin** { Adminmengolah data jabatan} **End**
30.	No. Proses	4.2.1
	Nama Proses	Tambah Data Jabatan
	Source	Admin
	Input	Data tambah jabatan
	Output	Info tambah jabatan
	Destination	Admin
	Logika Proses	**Begin** {Adminmenambah data jabatan} **End**
31.	No. Proses	4.2.2
	Nama Proses	Cari Data Jabatan
	Source	Admin
	Input	Data cari jabatan
	Output	Info cari jabatan
	Destination	Admin

	Logika Proses	**begin**
		{Admin mencari data jabatan }
		if data ada **then** data berhasil ditemukan
		else
		data tidak ditemukan
		end

32.	No. Proses	4.2.3
	Nama Proses	Ubah Data Jabatan
	Source	Admin
	Input	Data ubah jabatan
	Output	Info ubah jabatan
	Destination	Admin

	Logika Proses	**begin**
		if data jabatanbenar **and** tidak ada data kosong **then**
		simpan hasil perubahan

		else perubahan data tidak berhasil **end**
33.	No. Proses	4.2.4
	Nama Proses	Hapus Data Jabatan
	Source	Admin
	Input	Data hapus jabatan
	Output	Info hapus jabatan
	Destination	Admin

	Logika Proses	**begin**
		if data akan yakin dihapus **then**
		data terhapus
		else
		data tidak terhapus
		end

34.	No. Proses	4.3
	Nama Proses	Pengolahan Data Pangkat
	Source	Admin
	Input	Data tambah,cari,ubah,hapus pangkat
	Output	Info tambah,cari,ubah,hapus pangkat
	Destination	Admin
	Logika Proses	**begin** { Adminmengolah data pangkat } **End**
35.	No. Proses	4.3.1
	Nama Proses	Tambah Data Pangkat
	Source	Admin
	Input	Data tambah pangkat
	Output	Info tambah pangkat
	Destination	Admin
	Logika Proses	**begin** {Adminmenambah data pangkat} **End**
36.	No. Proses	4.3.2
	Nama Proses	Cari Data Pangkat
	Source	Admin
	Input	Data cari pangkat
	Output	Info cari pangkat
	Destination	Admin

	Logika Proses	**begin**
		{Admin mencari data jabatan }
		if data ada **then** data berhasil ditemukan
		else
		data tidak ditemukan
		end

36.	No. Proses	4.3.3
	Nama Proses	Ubah Data Pangkat
	Source	Admin
	Input	Data ubah pangkat
	Output	Info ubah pangkat
	Destination	Admin

	Logika Proses	**begin**
		if data pangkatbenar **and** tidak ada data kosong **then**
		simpan hasil perubahan

		else perubahan data tidak berhasil **end**
38.	No. Proses	4.3.4
	Nama Proses	Hapus Data Pangkat
	Source	Admin
	Input	Data hapus pangkat
	Output	Info hapus pangkat
	Destination	Admin

	Logika Proses	**begin**
		if data akan yakin dihapus **then**
		data terhapus
		else
		data tidak terhapus
		end

39.	No. Proses	4.4
	Nama Proses	Pengolahan Data Gaji
	Source	Admin
	Input	Data tambah,cari,ubah,hapus gaji
	Output	Info tambah,cari,ubah,hapus gaji
	Destination	Admin
	Logika Proses	**begin** { Adminmengolah data gaji} **End**
40	No. Proses	4.4.1
	Nama Proses	Tambah Data Gaji
	Source	Admin
	Input	Data tambah gaji
	Output	Info tambah gaji
	Destination	Admin
	Logika Proses	**begin** {Adminmenambah gaji} **End**
41.	No. Proses	4.4.2
	Nama Proses	Cari Data Gaji
	Source	Admin
	Input	Data cari gaji
	Output	Info cari gaji
	Destination	Admin

	Logika Proses	**begin**
		{Admin mencari data gaji}
		if data ada **then** data berhasil ditemukan
		else
		data tidak ditemukan
		end

42.	No. Proses	4.4.3
	Nama Proses	Ubah Data Gaji
	Source	Admin
	Input	Data ubah gaji
	Output	Info ubah gaji
	Destination	Admin

	Logika Proses	**begin**
		if data gajibenar **and** tidak ada data kosong **then**
		simpan hasil perubahan

		else perubahan data tidak berhasil **end**
43.	No. Proses	4.4.4
	Nama Proses	Hapus Data Gaji
	Source	Admin
	Input	Data hapus gaji
	Output	Info hapus gaji
	Destination	Admin

	Logika Proses	**begin**
		if data akan yakin dihapus **then**
		data terhapus
		else
		data tidak terhapus
		end

44.	No. Proses	4.5
	Nama Proses	Pengolahan Data Departemen
	Source	Admin
	Input	Data tambah,cari,ubah,hapus departemen
	Output	Info tambah,cari,ubah,hapus departemen
	Destination	Admin
	Logika Proses	**begin** { Adminmengolah data departemen } **End**
45.	No. Proses	4.5.1
	Nama Proses	Tambah Data Departemen
	Source	Admin
	Input	Data tambah departemen
	Output	Info tambah departemen
	Destination	Admin
	Logika Proses	**begin** {Adminmenambah departemen } **End**
46.	No. Proses	4.5.2
	Nama Proses	Cari Data Departemen
	Source	Admin
	Input	Data cari departemen
	Output	Info cari departemen
	Destination	Admin

	Logika Proses	**begin**
		{Admin mencari data departemen }
		if data ada **then** data berhasil ditemukan
		else
		data tidak ditemukan
		end

47.	No. Proses	4.5.3
	Nama Proses	Ubah Data Departemen
	Source	Admin
	Input	Data ubah departemen
	Output	Info ubah departemen
	Destination	Admin

	Logika Proses	**begin**
		if data departemen benar **and** tidak ada data kosong **then**
		simpan hasil perubahan

		else perubahan data tidak berhasil **end**
48.	No. Proses	4.5.4
	Nama Proses	Hapus Data Departemen
	Source	Admin
	Input	Data hapus departemen
	Output	Info hapus departemen
	Destination	Admin

	Logika Proses	**begin**
		if data akan yakin dihapus **then**
		data terhapus
		else
		data tidak terhapus
		end

49.	No. Proses	4.6
	Nama Proses	Pengolahan Data Provinsi
	Source	Admin
	Input	Data tambah,cari,ubah,hapus provinsi
	Output	Info tambah,cari,ubah,hapus provinsi
	Destination	Admin
	Logika Proses	**begin** { Adminmengolah data provinsi } **End**
50.	No. Proses	4.6.1
	Nama Proses	Tambah Data Provinsi
	Source	Admin
	Input	Data tambah provinsi
	Output	Info tambah provinsi
	Destination	Admin
	Logika Proses	**begin** {Adminmenambah data provinsi } **End**
51.	No. Proses	4.6.2
	Nama Proses	Cari Data Provinsi
	Source	Admin
	Input	Data cari provinsi
	Output	Info cari provinsi
	Destination	Admin

	Logika Proses	**begin**
		{Admin mencari data provinsi }
		if data ada **then** data berhasil ditemukan
		else
		data tidak ditemukan
		end

52.	No. Proses	4.6.3
	Nama Proses	Ubah Data Provinsi
	Source	Admin
	Input	Data ubah kota
	Output	Info ubah kota
	Destination	Admin

	Logika Proses	**begin**
		if data provinsibenar **and** tidak ada data kosong **then**
		simpan hasil perubahan

		else perubahan data tidak berhasil **end**
53.	No. Proses	4.6.4
	Nama Proses	Hapus Data Provinsi
	Source	Admin
	Input	Data hapus provinsi
	Output	Info hapus provinsi
	Destination	Admin

	Logika Proses	**begin**
		if data akan yakin dihapus **then**
		data terhapus
		else
		data tidak terhapus
		end

54.	No. Proses	4.7
	Nama Proses	Pengolahan Data Kota
	Source	Admin
	Input	Data tambah,cari,ubah,hapus kota
	Output	Info tambah,cari,ubah,hapus kota
	Destination	Admin
	Logika Proses	**begin** { Adminmengolah data kota} **End**
55.	No. Proses	4.7.1
	Nama Proses	Tambah Data Kota
	Source	Admin
	Input	Data tambah kota
	Output	Info tambah kota
	Destination	Admin
	Logika Proses	**begin** {Adminmenambah data kota} **End**
56.	No. Proses	4.7.2
	Nama Proses	Cari Data Kota
	Source	Admin
	Input	Data cari kota
	Output	Info cari kota
	Destination	Admin

	Logika Proses	**begin**
		{Admin mencari data kota }
		if data ada **then** data berhasil ditemukan
		else
		data tidak ditemukan
		end

57.	No. Proses	4.7.3
	Nama Proses	Ubah Data Kota
	Source	Admin
	Input	Data ubah kota
	Output	Info ubah kota
	Destination	Admin

	Logika Proses	**begin**
		if data kecamatan benar **and** tidak ada data kosong **then**
		simpan hasil perubahan

		else perubahan data tidak berhasil **end**
58.	No. Proses	4.7.4
.	Nama Proses	Hapus Data Kota
	Source	Admin
	Input	Data hapus kota
	Output	Info hapus kota
	Destination	Admin

	Logika Proses	**begin**
		if data akan yakin dihapus **then**
		data terhapus
		else
		data tidak terhapus
		end

59.	No. Proses	4.8
	Nama Proses	Pengolahan Data Kecamatan
	Source	Admin
	Input	Data tambah,cari,ubah,hapus kecamatan
	Output	Info tambah,cari,ubah,hapus kecamatan
	Destination	Admin
	Logika Proses	**begin** { Adminmengolah data kecamatan } **End**
60.	No. Proses	4.8.1
	Nama Proses	Tambah Data Kecamatan
	Source	Admin
	Input	Data tambah kecamatan
	Output	Info tambah kecamatan
	Destination	Admin
	Logika Proses	**begin** {Adminmenambah data kecamatan} **End**
61.	No. Proses	4.8.2
	Nama Proses	Cari Data Kecamatan
	Source	Admin
	Input	Data cari kecamatan
	Output	Info cari kecamatan
	Destination	Admin

	Logika Proses	**begin**
		{Admin mencari data kecamatan}
		if data ada **then** data berhasil ditemukan
		else
		data tidak ditemukan
		end

62.	No. Proses	4.8.3
	Nama Proses	Ubah Data Kecamatan
	Source	Admin
	Input	Data ubah kecamatan
	Output	Info ubah kecamatan
	Destination	Admin

	Logika Proses	**begin**
		if data kecamatanbenar **and** tidak ada data kosong **then**
		simpan hasil perubahan

		else perubahan data tidak berhasil **end**
63.	No. Proses	4.8.4
	Nama Proses	Hapus Data Kecamatan
	Source	Admin
	Input	Data hapus kecamatan
	Output	Info hapus kecamatan
	Destination	Admin

	Logika Proses	**begin**
		if data akan yakin dihapus **then**
		data terhapus
		else
		data tidak terhapus
		end

64.	No. Proses	4.9
	Nama Proses	Pengolahan Data Kelurahan
	Source	Admin
	Input	Data tambah,cari,ubah,hapus kelurahan
	Output	Info tambah,cari,ubah,hapus kelurahan
	Destination	Admin
	Logika Proses	**begin** { Adminmengolah data kecamatan } **End**
65.	No. Proses	4.9.1
	Nama Proses	Tambah Data Kelurahan
	Source	Admin
	Input	Data tambah kelurahan
	Output	Info tambah kelurahan
	Destination	Admin
	Logika Proses	**begin** {Adminmenambah data kelurahan } **End**
66.	No. Proses	4.9.2
	Nama Proses	Cari Data Kelurahan
	Source	Admin
	Input	Data cari kelurahan
	Output	Info cari kelurahan
	Destination	Admin

	Logika Proses	**begin**
		{Admin mencari data kelurahan }
		if data ada **then** data berhasil ditemukan
		else
		data tidak ditemukan
		end

67.	No. Proses	4.9.3
	Nama Proses	Ubah Data Kelurahan
	Source	Admin
	Input	Data ubah kelurahan
	Output	Info ubah kelurahan
	Destination	Admin

	Logika Proses	**begin**
		if data kelurahan benar **and** tidak ada data kosong **then**
		simpan hasil perubahan

		else perubahan data tidak berhasil **end**
68.	No. Proses	4.9.4
	Nama Proses	Hapus Data Kelurahan
	Source	Admin
	Input	Data hapus kelurahan
	Output	Info hapus kelurahan
	Destination	Admin

	Logika Proses	**begin**
		if data akan yakin dihapus **then**
		data terhapus
		else
		data tidak terhapus
		end

3.1.7.4. Kamus Data

Kamus data merupakan deskripsi formal mengenai seluruh elemen yang tercakup dalam DFD. Kamus data untuk DFD Aplikasi PengolahanData Kepegawaian pada PT. Taspen sebagai berikut:

Tabel 3.2 Kamus Data

Komponen	Keterangan
Nama	**Data *Login***
Where used/how used	User Data Entry-1.0 Admin-1.0 1.0- User
Deskripsi	Berisi data *login*
Struktur data	NIK+Password
ID Password	[A...Z|a...z|0...9] [A...Z|a...z|0...9]
Nama	**Data user**
Where used/how used	User Data Entry-2.0 Admin-2.0 2.0-user
Deskripsi	Berisi data user
Struktur data	Nik+nama+username+password+level+id_lokasi
ID Password	[A...Z|a...z|0...9] [A...Z|a...z|0...9]
Nama	**Data PNS**
Where used/how used	User Data Entry – 3.0 3.0 – pns
Deskripsi	Berisi data pns
Struktur data	nip+nama+tempat_lahir+tgl_lahir+jk+agama+status_kepegawaian+jenis_kepegawaian+kedudukan+status_kawin+alamat+id1+id2+id3+id4+gol_darah+no_karpeg+no_taspen+no_

	askes+no_karis+no_ktp+no_npwp+id_kelurahan+id_lokasi
Nip	[0...9]
nama	[A...Z|a...z]
tempat_lahir	[A...Z|a...z]
tgl_lahir	[A...Z|a...z|0...9]
jk	[A...Z|a...z]
agama	[A...Z|a...z]
status_kawin	[A...Z|a...z]
alamat	[A...Z|a...z|0...9]
id1	[0...9]
id2	[0...9]
id3	[0...9]
id4	[0...9]
gol_darah	[0...9]
no_karpeg	[0...9]
no_taspen	[0...9]
no_askes	[0...9]
no_karis	[0...9]
no_ktp	[0...9]
mo_npwp	[0...9]
id_kelurahan	[0...9]
id_lokasi	[0...9]
Nama	**Data riwayat instansi**
Where used/how used	User data entry 3.0 3.0 – riwayat_instansi
Deskripsi	Berisi data riwayat instansi
Struktur data	nip+id_satker+tgl_masuk
nip	[0...9]
tgl_masuk	[A...Z|a...z|0...9]
id_satker	[0...9]
Nama	**Data riwayat jabatan**
Where used/how used	User data entry- 3.0 3.0-riwayat_jabatan
Deskripsi	Berisi data mata pelajaran
Struktur data	nip+id_jabatan+tmt_jabatan+tgl_berakhir
nip	[0...9]
id_jabatan	[0...9]
tmt_jabatan	[A...Z|a...z|0...9]
tgl_berakhir	[A...Z|a...z|0...9]
Nama	**Data riwayat pangkat/gol**
Where used/how used	User data entry 3.0 3.0 – riwayat_pangkat_gol
Deskripsi	Berisi data riwayat pangkat/gol
Struktur data	nip+id_pangkat+no_sk+tmt_pangkat
nip	[0...9]
id_pangkat	[0...9]
no_sk	[A...Z|a...z|0...9]
tmt_pangkat	[A...Z|a...z|0...9]
Nama	**Data riwayat gaji**
Where used/how used	User data entry 3.0 3.0 – riwayat_gaji
Deskripsi	Berisi data riwayat gaji
Struktur data	nip+id_gaji+tmt_gaji
Nip	[0...9]

id_gaji tmt_gaji	[0...9] [A...Z|a...z|0...9]
Nama	**Data dana pensiun**
Where used/how used	User data entry 3.0 3.0 – dana_pensiun
Deskripsi	Berisidana pensiun
Struktur data	Nip+tmt_pensiun+nama_penerima+alamat+hub_kel
Nip tmt_pensiun nama_penerima alamat hub_kel	[0...9] [A...Z|a...z|0...9] [A...Z|a...z] [A...Z|a...z|0...9] [A...Z|a...z]
Nama	**Cetak Laporan**
Where used/how used	User data entry 3.0 3.0 – pns
Deskripsi	Berisi data dana tht
Struktur data	Nip+nama+tempat/tanggal lahir+lokasi+besar dana pensiun
Nip Nama Tempat/Tanggal lahir Lokasi Besar Dana Pensiun	[0...9] [0...9] [A...Z|a...z|0...9] [A...Z|a...z] [0...9]
Nama	**Data provinsi**
Where used/how used	Admin 4.0 4.0 – provinsi
Deskripsi	Berisi data provinsi
Struktur data	id_provinsi+nama_provinsi
id_provinsi nama_provinsi	[0...9] [A...Z|a...z]
Nama	**Data kota**
Where used/how used	Admin 4.0 4.0 – kota
Deskripsi	Berisi data kota
Struktur data	id_kota+nama_kota+id_provinsi
id_kota nama_kota id_provinsi	[0...9] [A...Z|a...z] [0...9]
Nama	**Data kecamatan**
Where used/how used	Admin 4.0 4.0 – kecamatan
Deskripsi	Berisi data kecamatan
Struktur data	id_kecamatan+nama_kecamatan+id_kota
id_kecamatan nama_kecamatan id_kota	[0...9] [A...Z|a...z] [0...9]
Nama	**Data kelurahan**
Where used/how used	Admin 4.0 4.0 – kelurahan
Deskripsi	Berisi data kelurahan
Struktur data	Id_kelurahan+nama_kelurahan+id_kecamatan
id_kelurahan nama_kelurahan id_kecamatan	[0...9] [A...Z|a...z] [0...9]
Nama	**Data departemen**

Where used/how used	Admin 4.0 4.0 – departemen
Deskripsi	Berisi data departemen
Struktur data	Id_departemen+nama_departemen
id_departemenn nama_departemen	[0...9] [A...Z|a...z]
Nama	**Data gaji**
Where used/how used	Admin 4.0 4.0 – gaji
Deskripsi	Berisi data gaji
Struktur data	Masker+tmt_gapok+gol+nilai
Masker mt_gapok gol nilai	[0...9] [A...Z|a...z] [A...Z|a...z|0...9] [A...Z|a...z]
Nama	**Data lokasi**
Where used/how used	Admin 4.0 4.0 – lokasi
Deskripsi	Berisi data lokasi cabang
Struktur data	id_lokasi+nama_cabang
id_lokasi nama_cabang	[0...9] [A...Z|a...z]
Nama	**Data instansi**
Where used/how used	Admin 4.0 4.0 – instansi
Deskripsi	Berisi data instansi
Struktur data	id_satker+id_departemen+unit_kerja+alamat+id_kelurahan
id_satker id_departemen unit_kerja alamat id_kelurahan	[0...9] [0...9] [A...Z|a...z|0...9] [A...Z|a...z|0...9] [0...9]
Nama	**Data jabatan**
Where used/how used	Admin 4.0 4.0 – jabatan
Deskripsi	Berisi data jabatan
Struktur data	id_jabatan+jenis_jabatan+eselon+nama_jabatan
id_jabatan jenis_jabatan eselon nama_jabatan	[0...9] [A...Z|a...z] [A...Z|a...z|0...9] [A...Z|a...z]
Nama	**Data Pangkat/Gol**
Where used/how used	Admin 4.0 4.0 – pangkat_gol
Deskripsi	Berisi data pangkat/gol
Struktur data	id_pangkat+nmgol1+nmgol2+golgapok
id_pangkat nmgol1 nmgol2 golgapok	[0...9] [A...Z|a...z|0..9] [A...Z|a...z|0...9] [A...Z|a...z|0..9]
Nama	**Data Anak**
Where used/how used	User data entry 3.0 3.0 – anak
Deskripsi	Berisi data anak

Struktur data	nip+nama+tempat_lahir+tgl_lahir+jenis_kelamin+pendidikan+status_tunjangan
Nip nama tempat_lahir tgl_lahir jenis_kelamin pendidikan status_tunjangan	[0...9] [A...Z|a...z] [A...Z|a...z] [A...Z|a...z|0...9] [A...Z|a...z] [A...Z|a...z|0...9] [A...Z|a...z]
Nama	**Data Pasangan**
Where used/how used	User data entry 3.0 3.0 – pasangan
Deskripsi	Berisi data pasangan
Struktur data	nip+nama_pasangan+tempat_lahir+tgl_lahir+tgl_kawin+pekerjaan+pend_terakhir
Nip nama_pasangan tempat_lahir tgl_lahir tgl_kawin pekerjaan pend_terakhir	[0...9] [A...Z|a...z] [A...Z|a...z] [A...Z|a...z|0...9] [A...Z|a...z|0...9] [A...Z|a...z] [A...Z|a...z|0...9]

3.2. Perancangan Sistem

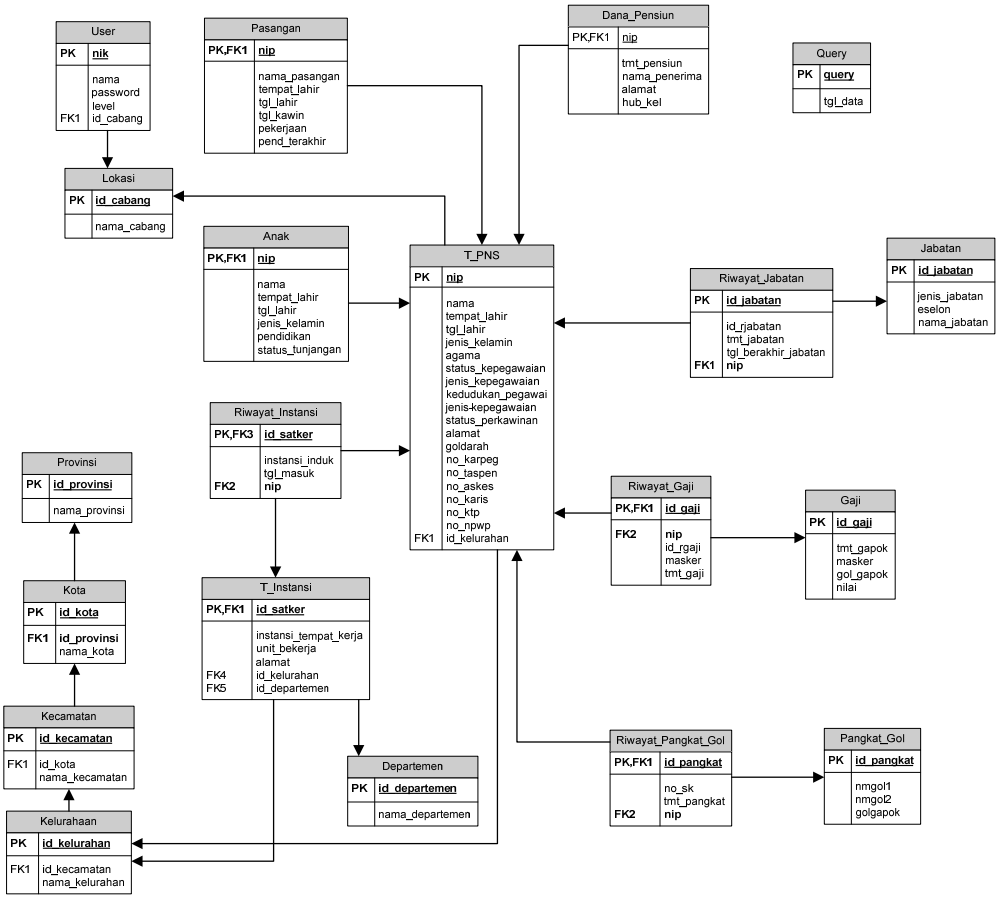
Perancangan akan dimulai setelah tahap analisis terhadap sistem selesai dilakukan. Perancangan dapat didefinisikan sebagai proses aplikasi berbagai teknik dan prinsip bagi tujuan pendefinisian suatu perangkat, suatu proses atau sistem dalam detail yang memadai untuk memungkinkan realisasi fisiknya. Perancangan digambarkan sebagai proses multi-langkah dimana representasi struktur data, struktur program, karakteristik *interface*, dan detail prosedur, disintesis dari persyaratan informasi.

3.2.1 Perancangan Basis Data

Perancangan data mentransformasi model domain informasi yang dibuat selama analisis ke dalam struktur data yang akan diperlukan untuk mengimplementasi perangkat lunak.

3.2.1.1. Skema Relasi

Proses relasi antar file merupakan gabungan antar file yang mempunyai kunci utama yang sama, sehingga file-file tersebut menjadi satu kesatuan yang dihubungkan oleh *field* kunci tersebut. Pada proses ini elemen-elemen data dikelompokkan menjadi satu file database beserta entitas dan hubungannya. Berikut adalah skema relasi Aplikasi Pengolahan Data Kepegawaian Pada PT. Taspen:



Gambar 3.38 Skema Relasi

3.2.1.2. Struktur Tabel

Tabel adalah sekumpulan data atau informasi spesifik tentang subjek tertentu yang disusun dalam bentuk kolom dan baris. Tabel adalah komponen utama dan pertama dari sebuah *database*. Struktur tabel untuk setiap tabel yang terdapat dalam *database* Aplikasi Pengolahan Data Kepegawaian Pada PT. Taspen:

1. Tabel User

Tabel user digunakan untuk mencatat data penting yang merupakan identitas para user. Secara umum, struktur tabel yang akan digunakan untuk mencatat data user dapat dilihat pada tabel 3.3 :

Tabel 3.3 Tabel User

Nama field	Tipe	Size	Keterangan
nik	varchar	18	Primary key
nama	varchar	30	
password	varchar	14	
Level	enum	‘user’,’admin’	
id_lokasi	char	5	Foreign Key

2. Tabel Lokasi

Tabel lokasi digunakan untuk menentukan user yang sedang login ada di kantor cabang mana. Secara umum, struktur tabel yang akan digunakan untuk mencatat data lokasi dapat dilihat pada tabel 3.4:

Tabel 3.4 Tabel Lokasi

Nama field	Tipe	Size	Keterangan
id_lokasi	int	11	Primary Key
nama_cabang	varchar	30	

3. Tabel Anak

Tabel anak digunakan untuk mencatat data identitas anak dari seorang PNS. Secara umum, struktur tabel yang akan digunakan untuk mencatat data anak dapat dilihat pada tabel 3.5:

Tabel 3.5 Tabel Anak

Nama field	Tipe	Size	Keterangan
nip	varchar	18	Primary Key
nama	varchar	30	
tempat_lahir	varchar	30	
tgl_lahir	date	1	
jenis_kelamin	enum	‘laki-laki’,’perempuan’	
pendidikan	enum	‘SD’,’SMP’,’SMA’,’Universitas’	
status_tunjangan	enum	‘masih’,’tidak	

4. Tabel Pasangan

Tabel pasangan digunakan untuk mencatat data identitas suami/istri dari seorang PNS. Secara umum, struktur tabel yang akan digunakan untuk mencatat data pasangan dapat dilihat pada tabel 3.6:

Tabel 3.6 Tabel Pasangan

Nama field	Tipe	Size	Keterangan
nip	varchar	18	Primary Key
nama_pasangan	varchar	30	
tempat_lahir	varchar	20	
tgl_lahir	date	1	
tgl_kawin	date	1	
pekerjaan	varchar	30	
pend_terakhir	varchar	30	

5. Tabel PNS

Tabel pns digunakan untuk mencatat data *master* PNS. Secara umum, struktur tabel yang akan digunakan untuk mencatat data pns dapat dilihat pada tabel 3.7 :

Tabel 3.7 Tabel PNS

Nama field	Tipe	Size	Keterangan
Nip	varchar	18	Primary key
Nama	varchar	50	
tempat_lahir	varchar	20	
tgl_lahir	date	1	
jenis_kelamin	enum	‘laki-laki’,’perempuan’	
Agama	enum	‘islam’,’katolik’,’protestan’,’budha’,’hindu’,’konghucu’	
status_kepegawaian	enum	‘pns’,’pejabat negara’,dprd’	
jenis_kepegawaian	varchar	100	
kedudukan_pegawai	varchar	100	
status_perkawinan	varchar	‘kawin’,belum kawin’	
Alamat	text		
gol_darah	varchar	‘0’,’A’,’B,’AB’	
no_karpeg	varchar	20	
no_taspen	varchar	20	
no_askes	varchar	20	
no_karis	varchar	20	
no_ktp	varchar	20	
no_npwp	varchar	20	
id_kelurahan	int	11	Foreign Key
id_lokasi	int	11	Foreign Key
Hapus	enum	‘ya’,’tidak’	

6. Tabel Riwayat_Gaji

Tabel riwayat_gaji digunakan untuk mencatat data riwayat gaji PNS. Secara umum, struktur tabel yang akan digunakan untuk mencatat data riwayat gaji dapat dilihat pada tabel 3.8 :

Tabel 3.8 Tabel Riwayat Gaji

Nama field	Tipe	Size	Keterangan
Nip	varchar	18	Primary Key
id_gaji	int	11	Foreign Key
tmt_gaji	date	1	

7. Tabel Riwayat_Instansi

Tabel riwayat_instansi digunakan untuk mencatat data riwayat instansi PNS. Secara umum, struktur tabel yang akan digunakan untuk mencatat data riwayat instansi dapat dilihat pada tabel 3.9 :

Tabel 3.9 Tabel Riwayat_Instansi

Nama field	Tipe	Size	Keterangan
Nip	varchar	18	Primary Key
id_satker	varchar	7	Foreign Key
tgl_masuk	date	1	

8. Tabel Riwayat_Jabatan

Tabel riwayat_jabatan digunakan untuk mencatat data riwayat jabatan. Secara umum, struktur tabel yang akan digunakan untuk mencatat data riwayat jabatan dapat dilihat pada tabel 3.10:

Tabel 3.10 Tabel Riwayat_Jabatan

Nama field	Tipe	Size	Keterangan
Nip	varchar	18	Primary Key
id_jabatan	int	11	Foreign Key
tmt_jabatan	date	1	
tgl_berkhir_jabatan	date	1	

9. Tabel Riwayat_Pangkat_Gol

Tabel riwayat_pangkat_gol digunakan untuk mencatat data riwayat pangkat/gol PNS. Secara umum, struktur tabel yang akan digunakan untuk mencatat data pangkat/gol dapat dilihat pada tabel 3.11:

Tabel 3.11 Tabel Riwayat_Pangkat_Gol

Nama field	Tipe	Size	Keterangan
Nip	varchar	18	Primary Key
id_pangkat	varchar	2	Foreign Key
no_sk	varchar	20	
tmt_pangkat	date	1	

10. Tabel Gaji

Tabel gaji digunakan untuk mencatat data gaji. Secara umum, struktur tabel yang akan digunakan untuk mencatat data gaji dapat dilihat pada tabel 3.12:

Tabel 3.12 Tabel Gaji

Nama field	Tipe	Size	Keterangan
id_gaji	Int	11	Primary Key
masker	Int	11	
tmt_gapok	Date	1	
Gol	varchar	2	
Nilai	Int	11	

11. Tabel Instansi

Tabel instansi digunakan untuk mencatat data instansi. Secara umum, struktur tabel yang akan digunakan untuk mencatat data siswa perjurusan dapat dilihat pada tabel 3.13:

Tabel 3.13 Tabel Instansi

Nama field	Tipe	Size	Keterangan
id_satker	varchar	7	Primary Key
id_departemen	varchar	50	
unit_kerja	varchar	20	
alamat	text		
id_kelurahan	int	11	Foreign Key

12. Tabel Jabatan

Tabel jabatan digunakan untuk mencatat data jabatan. Secara umum, struktur tabel yang akan digunakan untuk mencatat data jabatan dapat dilihat pada tabel 3.14:

Tabel 3.14 Tabel Jabatan

Nama field	Tipe	Size	Keterangan
id_jabatan	varchar	11	Primary Key
jenis_jabatan	varchar	30	
Eselon	varchar	20	
nama_jabatan	varchar	30	

13. Tabel Pangkat_Gol

Tabel pangkat_gol digunakan untuk mencatat data pangkat/gol. Secara umum, struktur tabel yang akan digunakan untuk mencatat data pangkat/gol dapat dilihat pada tabel 3.15:

Tabel 3.15 Tabel Pangkat_Gol

Nama field	Tipe	Size	Keterangan
id_pangkat	varchar	2	Primary Key
nmgol1	varchar	5	
nmgol2	varchar	60	
golgapok	varchar	2	

14. Tabel Departemen

Tabel departemen digunakan untuk mencatat data departemen. Secara umum, struktur tabel yang akan digunakan untuk mencatat data departemen dapat dilihat pada tabel 3.16:

Tabel 3.16 Tabel Departemen

Nama field	Tipe	Size	Keterangan
id_departemen	varchar	3	Primary Key
nama_departemen	varchar	50	

15. Tabel Provinsi

Tabel provinsi digunakan untuk mencatat data provinsi. Secara umum, struktur tabel yang akan digunakan untuk mencatat data provinsi dapat dilihat pada tabel 3.17:

Tabel 3.17 Tabel Provinsi

Nama field	Tipe	Size	Keterangan
id_provinsi	int	11	Primary Key
nama_provinsi	varchar	65	

16. Tabel Kota

Tabel kota digunakan untuk mencatat data kota. Secara umum, struktur tabel yang akan digunakan untuk mencatat data kota dapat dilihat pada tabel 3.18:

Tabel 3.18 Tabel Kota

Nama field	Tipe	Size	Keterangan
id_kota	int	11	Primary Key
nm_kota	varchar	65	
id_provinsi	int	11	Foreign Key

17. Tabel Kecamatan

Tabel kecamatan digunakan untuk mencatat data kecamatan. Secara umum, struktur tabel yang akan digunakan untuk mencatat data kecamatan dapat dilihat pada tabel 3.19:

Tabel 3.19 Tabel Kecamatan

Nama field	Tipe	Size	Keterangan
id_kecamatan	int	11	Primary Key
nama_kecamatan	varchar	65	
id_kota	int	11	Foreign Key

18. Tabel Kelurahan

Tabel kelurahan digunakan untuk mencatat data kelurahan. Secara umum, struktur tabel yang akan digunakan untuk mencatat data kelurahan dapat dilihat pada tabel 3.20:

Tabel 3.20 Tabel Kelurahan

Nama field	Tipe	Size	Keterangan
id_kelurahan	int	11	Primary Key
nama_kelurahan	varchar	65	
id_kecamatan	int	11	Foreign Key

19. Tabel Dana_Pensiun

Tabel dana_pensiun digunakan untuk mencatat data dana pensiun PNS. Secara umum, struktur tabel yang akan digunakan untuk mencatat data dana pensiun dapat dilihat pada tabel 3.21:

Tabel 3.21 Tabel Dana_Pensiun

Nama field	Tipe	Size	Keterangan
Nip	varchar	18	Primary Key
tmt_pensiun	date	1	
nama_penerima	varchar	30	
Alamat	text		
hub_kel	varchar	25	

20. Tabel Query

Tabel query digunakan untuk menyimpan semua aktivitas transaksi query yang telah dilakukan. Secara umum tabel yang digunakan untuk mencatat data query dapat dilihat pada tabel 3.22:

Tabel 3.22 Tabel Query

Nama field	Tipe	Size	Keterangan
Query	text		
tgl_data	date	1	

3.2.2 Cara Replikasi Pada *MySQL*

Replikasi memungkinkan dari satu database MySQL (yang disebut *master*) dapat direplikasi untuk satu atau lebih database MySQL *server* (*slave*). Replikasi ini sifatnya *asynchronous* – mesin *slave* replikasi kita tidak perlu dihubungkan secara permanen untuk menerima update dari *master*, yang berarti bahwa pembaruan dapat terjadi melalui sambungan jarak jauh dan bahkan solusi sementara seperti layanan dial-up. Tergantung pada konfigurasi, kita dapat mereplikasi semua database, database yang dipilih, atau bahkan hanya tabel-table tertentu saja yang berada dalam sauatu database.

1. Proses Replikasi

Saat replikasi berjalan, semua SQL statement yang dilakukan *server*, dicatat MySQL dalam log biner (bin.log) ditandai dengan nomor identifikasi log. Pada gilirannya *server slave* secara berulang dan sering membaca log biner *master* ini dan memeriksa apa ada perubahan. Jika ditemukan perubahan, *slave* akan meng-copy statement baru ini pada log relay (relay.log). Kemudian *slave* mencatat posisi baru nomor identifikasi dalam suatu file (*master*.info) di *server slave*.

Untuk menjamin replikasi berjalan lancar, hanya diperlukan penambahan beberapa baris di file konfigurasi (yaitu my.ini) pada *server master* dan *server slave*. Jika *server* masih baru, hanya diperlukan peng-copy-an database pada *server master* ke *server slave*. Kemudian mengaktifkan *slave* untuk mulai melakukan replikasi.

2. Pengguna Replikasi (Replication User)

Diperlukan beberapa tahapan untuk melakukan setting replikasi. Tahap pertama adalah melakukan set up pengguna (*user account*) yang hanya digunakan untuk replikasi. Sebaiknya tidak menggunakan pengguna yang saat ini sudah terdaftar, demi alasan keamanan. Untuk melakukan ini, jalankan statement SQL berikut ini di *server master*, masuk sebagai root atau user yang mempunyai privilege GRANT OPTION:

```
GRANT REPLICATION SLAVE *.*
TO 'server2'@'192.168.0.5' IDENTIFIED BY 'replikasi';
```

Dalam statement SQL di atas, pengguna replicant diberikan hak untuk melakukan replikasi. Nama pengguna bisa apa saja. Nama host (atau alamat IP) diberikan diantara satu tanda petik. Setelah itu harus menjalankan statement yang sama di *server slave* dengan nama pengguna dan *password* yang sama, tetapi dengan nama *host master* atau alamat IP *master*.

3. Setting keadaan di kedua database dengan posisi yang sama

Keadaan database pada saat pertama akan dilakukan replikasi. Untuk mengecek agar database dalam keadaan tidak melakukan transaksi, jalankna statement SQL di bwaha ini:

```
Flush tables with read lock;
```

Untuk mengecek status *master*, jalankan statement dibawh ini :

```
show master status;
```

4. Buka file my.ini

Nomor identifikasi *server* dapat sembarang untuk membedakan *server master*. Nomor yang lain harus diberikan pada *server slave* sehingga keduanya dapat dibedakan. Seperti 1 untuk *master*, dan 2 untuk *slave*.

```
server_id=1
```

Sedangkan pada *slave*

```
server_id=2
```

5. Pada *slave*

Pada database *slave*, tambahkan baris di bawah ini :

```
binlog-do-db=taspen
binlog-ignore-db=mysql
binlog-ignore-db=information_schema
binlog-ignore-db=phpmyadmin
```

Kemudian save settingan diatas, dan restart databse *slave*.

6. Settingan pada *master*

Eksport database di *master* ke *slave*. Proses eksport dari database *master* ke *slave* bisa dilakukan dengan cara mengkopi data pada database *master* ke database *slave*. Atau bisa jalankan statement ini :

```
mysqldump taspen –r dump.sql –u root –p
```

7. Pada *Slave*

Masuk ke mysql cmd. Jalankan statement ini :

```
create database taspen;

-q;
```

Masuk lagi ke console mysql. Jalankan statement ini:

```
stop slave;

change master to

master_host='192.168.0.42',

master_user='server2',

master-password='replikasi',

master_log_file='mysql-bin.000001',

master_log_post=106;
```

Kemudian :

```
Show slave status\G
```

Untuk mengetahui status pasa databse *slave*. Sehingga dapat diketahui apakah database *master* bs mereplikasikan data ke database *slave*.

Status pada *slave* dapat diketahui dengan melihat :

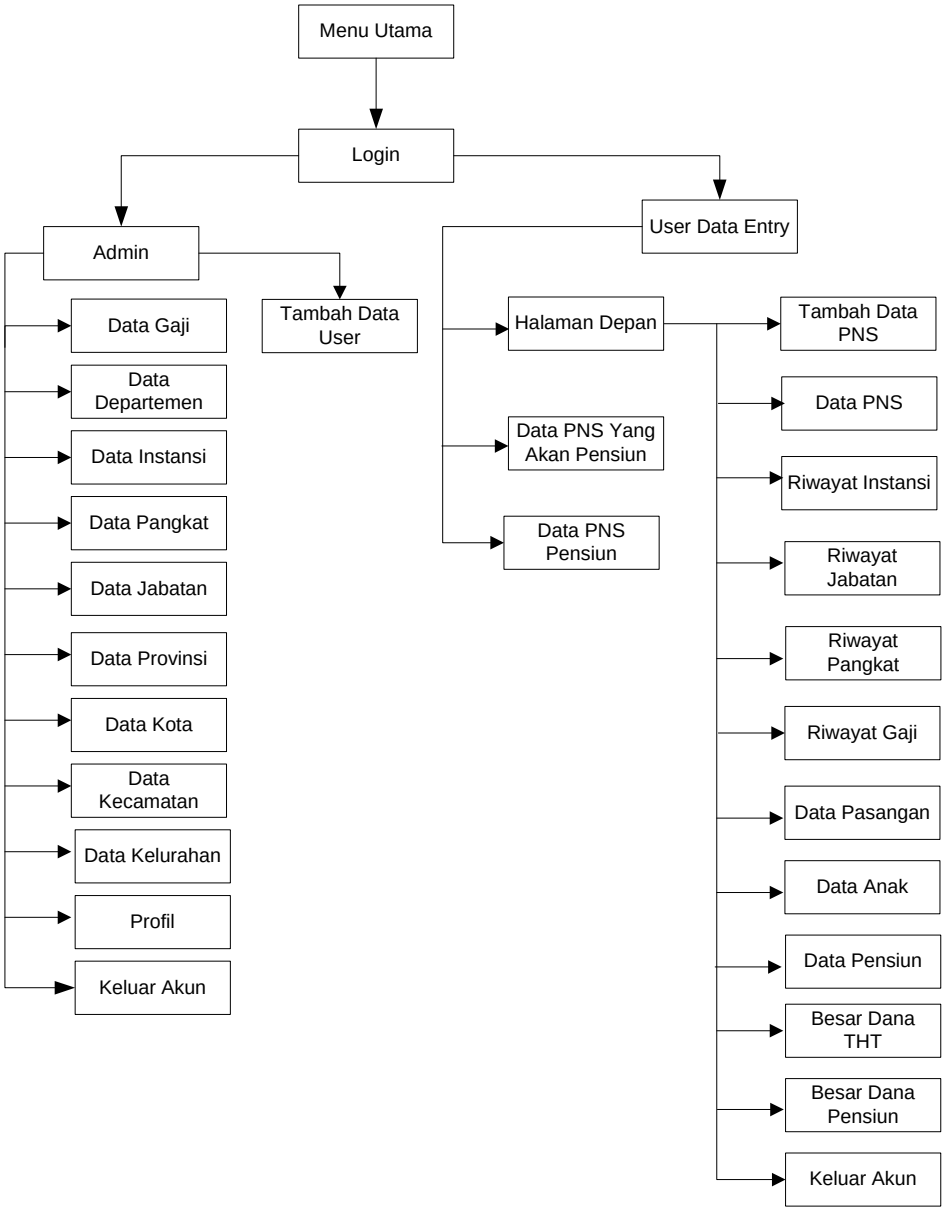
```
slave_10_running=yes

slave_sql_running=yes
```

Itu berarti prose repliaksi bisa dilakukan. Karena *slave* telah bisa menerima data dari *master*.

3.2.3 Perancangan Ar*site*ktur

Perancangan arstitektur merupakan hubungan di antara elemen-elemen struktural utama dari program. Perancangan ar*site*ktur dapat memberikan gambaran mengenai struktur program. Gambar 3.44 adalah ar*site*ktur perangkat lunak dari Aplikasi Pengolahan Data Kepegawaian Pada PT. Taspen yang berbentuk hirarki:

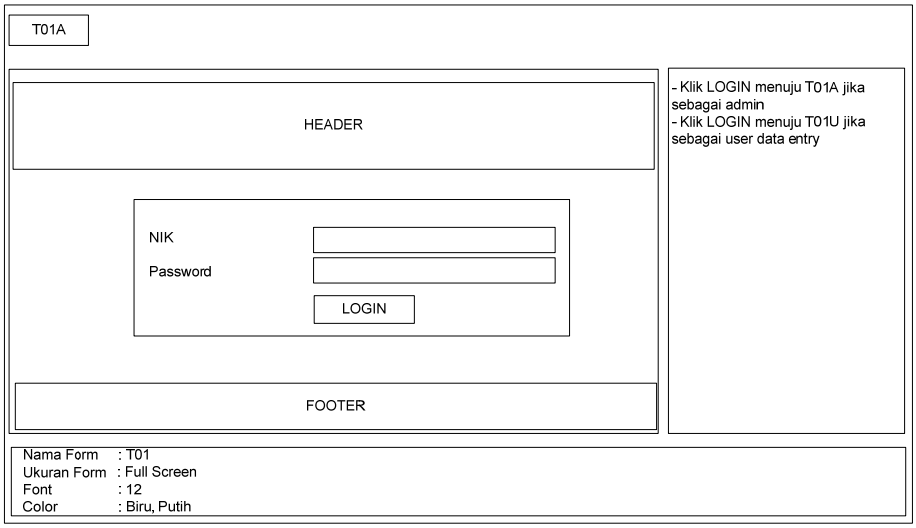


Gambar 3.39 Struktur Menu Aplikasi Pengolahan Data Kepegawaian

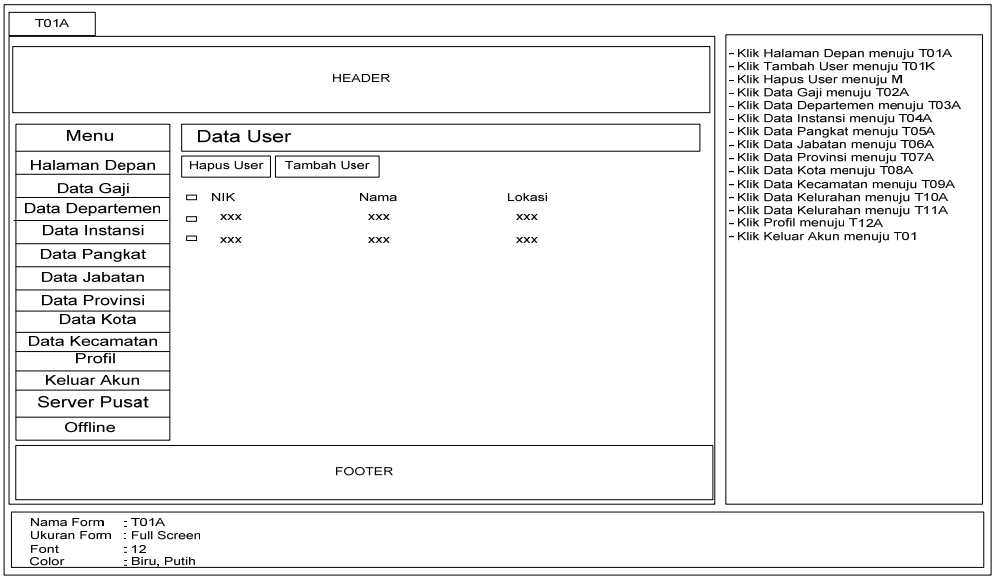
3.2.4 Perancangan Antarmuka (*Interface*)

Perancangan antarmuka (*interface*) menggambarkan bagaimana perangkat lunak berkomunikasi dengan dirinya sendiri, dengan sistem yang berinteroperasi dengannnya, dan dengan manusia yang menggunakannya.

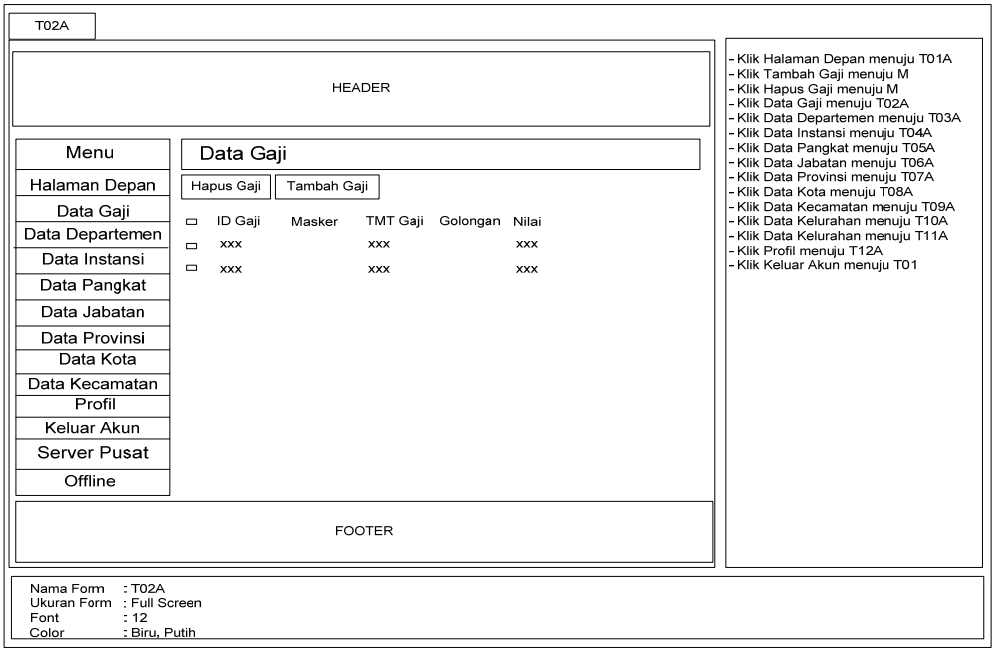
3.2.4.1. Perancangan Tampilan



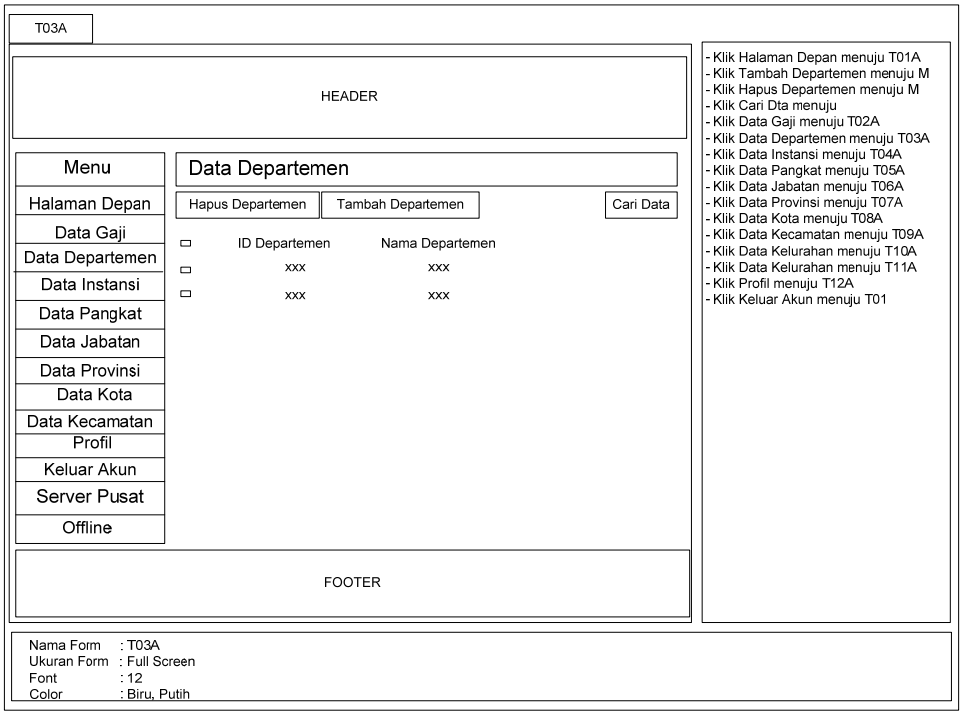
Gambar 3.40 Tampilan Halaman Login



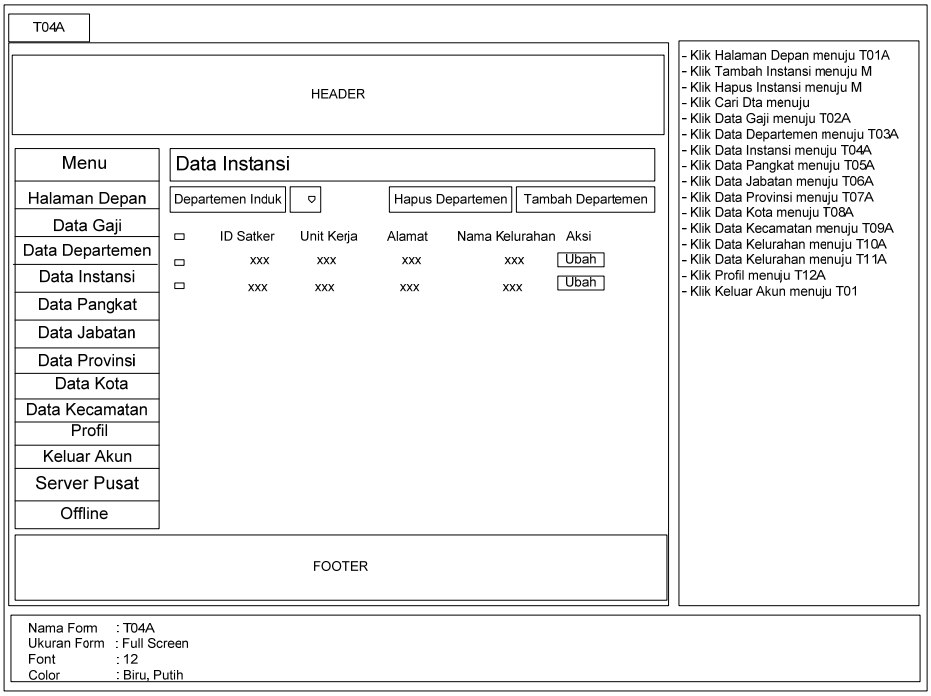
Gambar 3.41 Tampilan Halaman Depan Admin



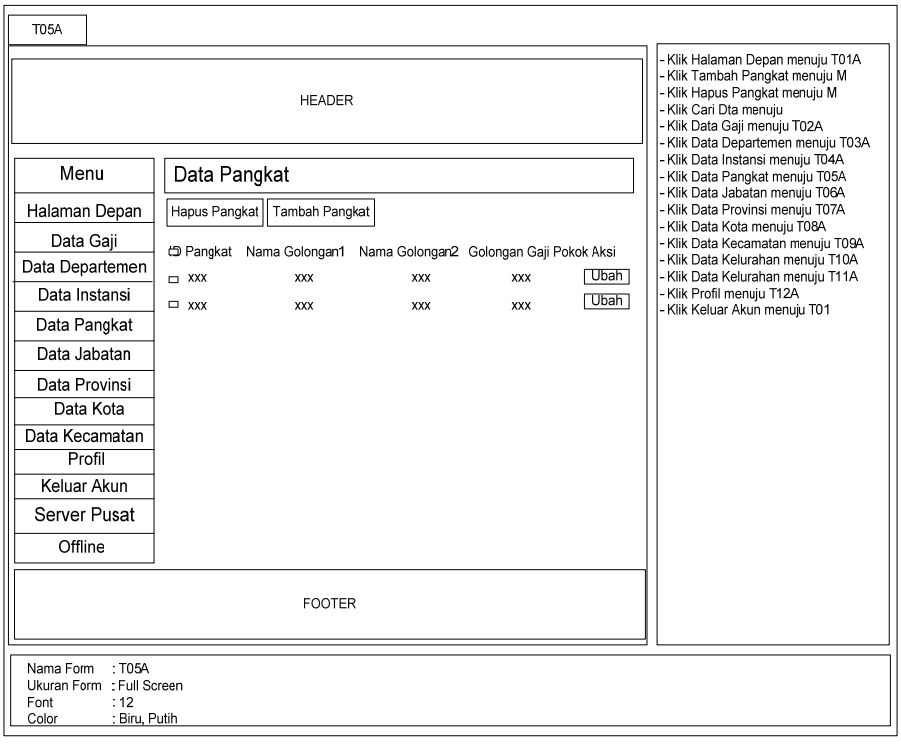
Gambar 3.42 Tampilan Data Gaji



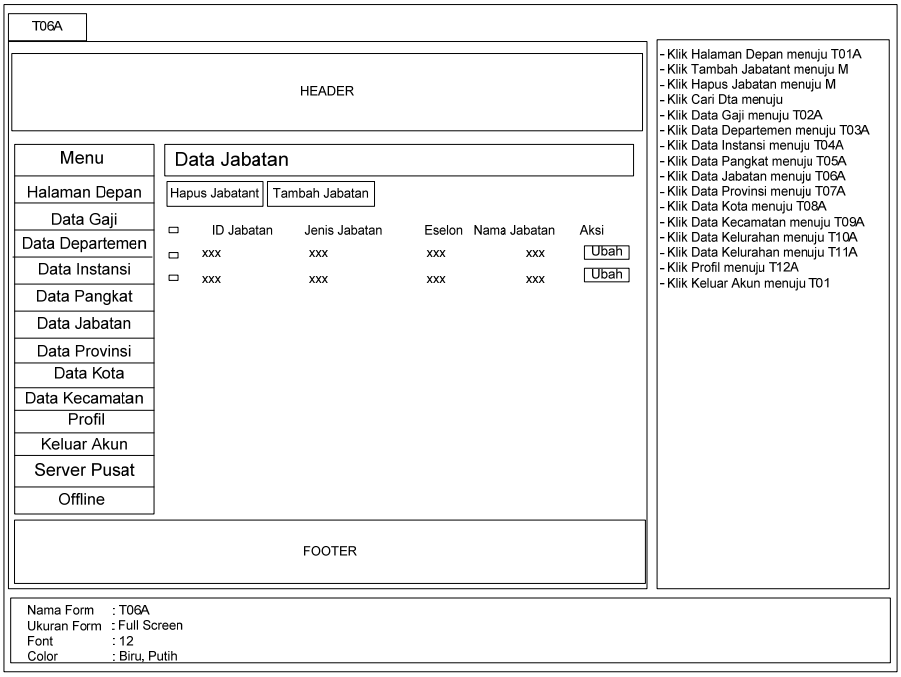
Gambar 3.43 Tampilan Data Departemen



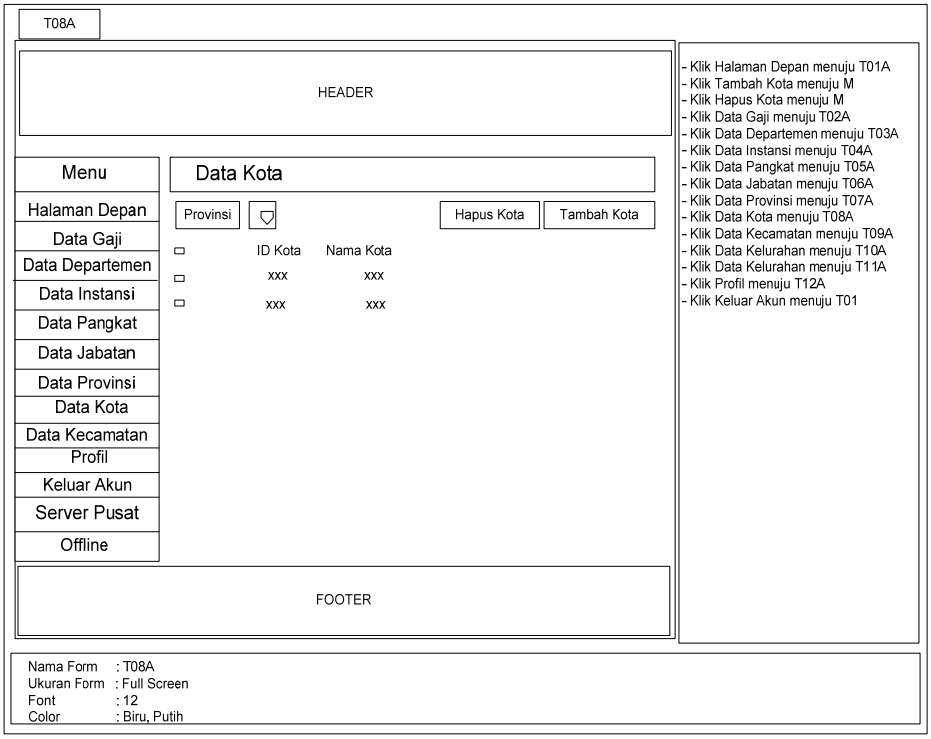
Gambar 3.44 Tampilan Data Instansi



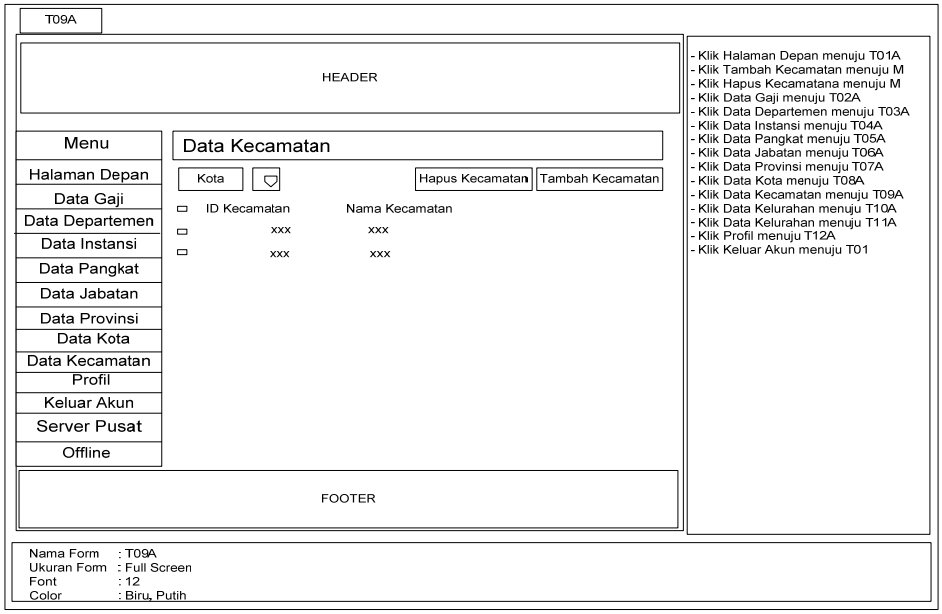
Gambar 3.45 Tampilan Data Pangkat



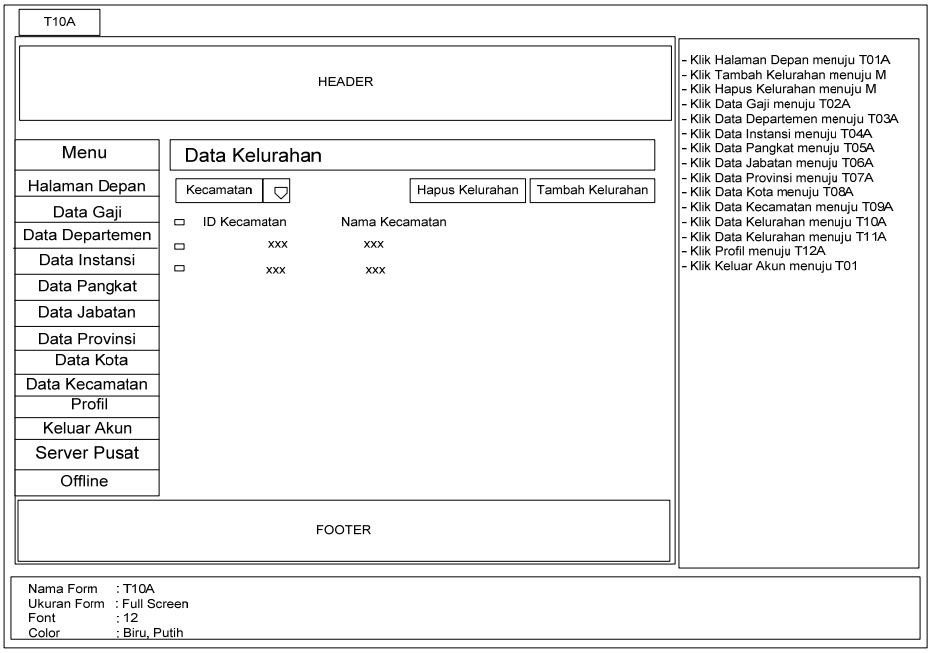
Gambar 3.46 Tampilan tampil data jabatan



Gambar 3.47 Tampilan tambah jabatan

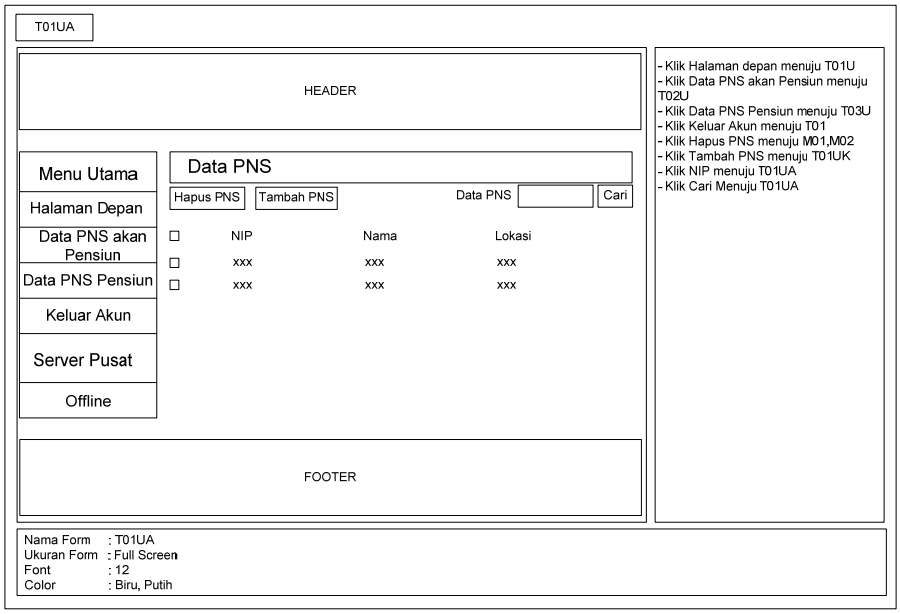


Gambar 3.48 Tampilan data keacmatan

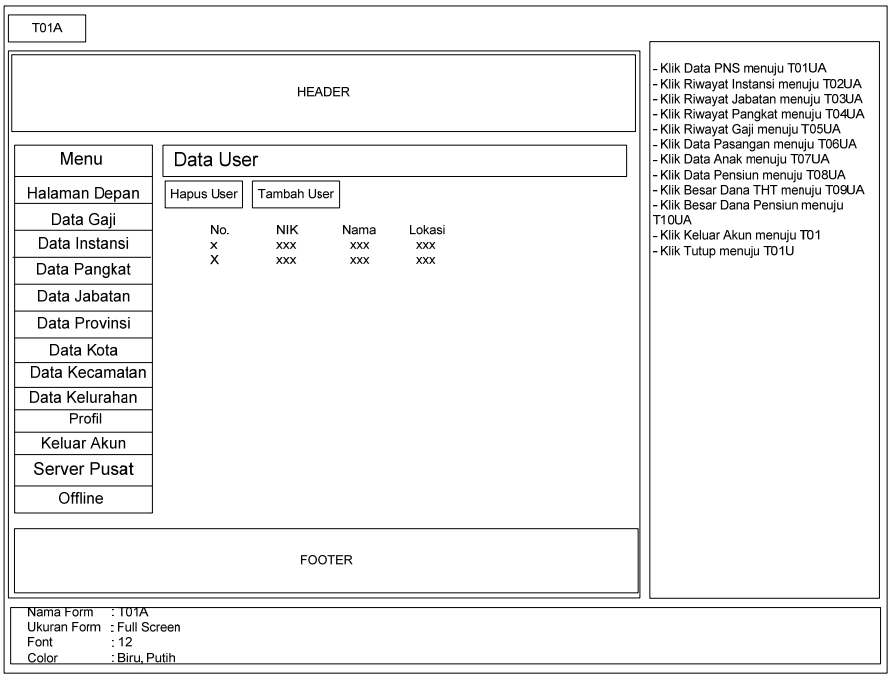


Gambar 3.49 Tampilan Data Kelurahan

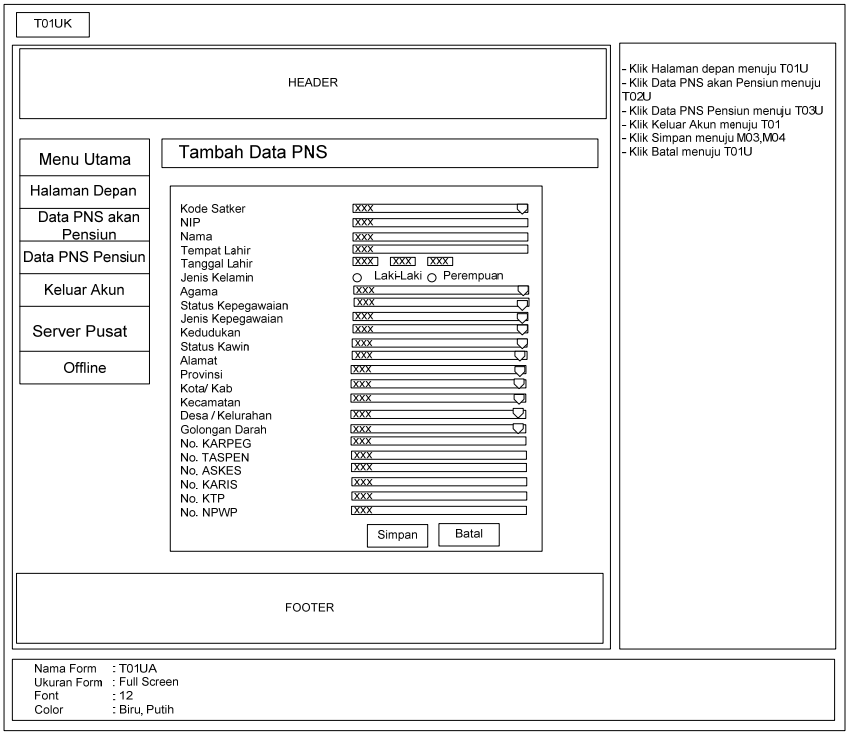
Tampilan Menu User Data Entry



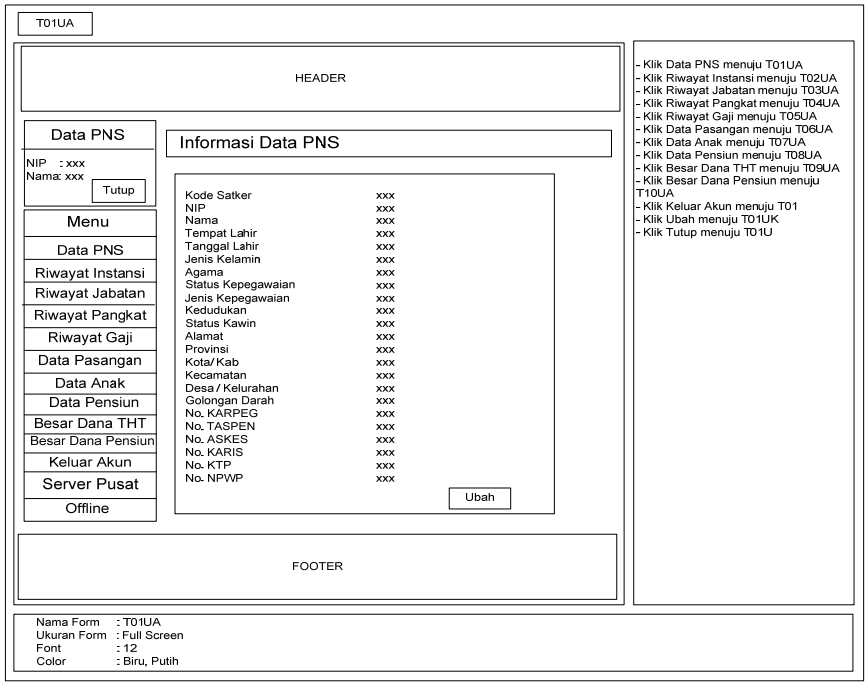
Gambar 3.50 Tampilan Halaman User Data Entry



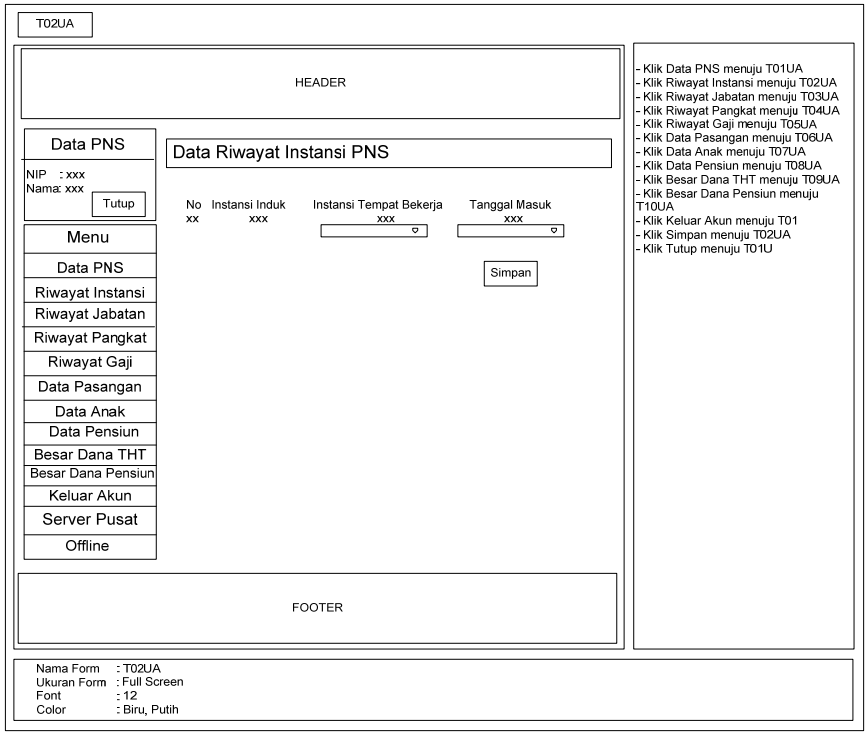
Gambar 3.51 Tampilan Data User



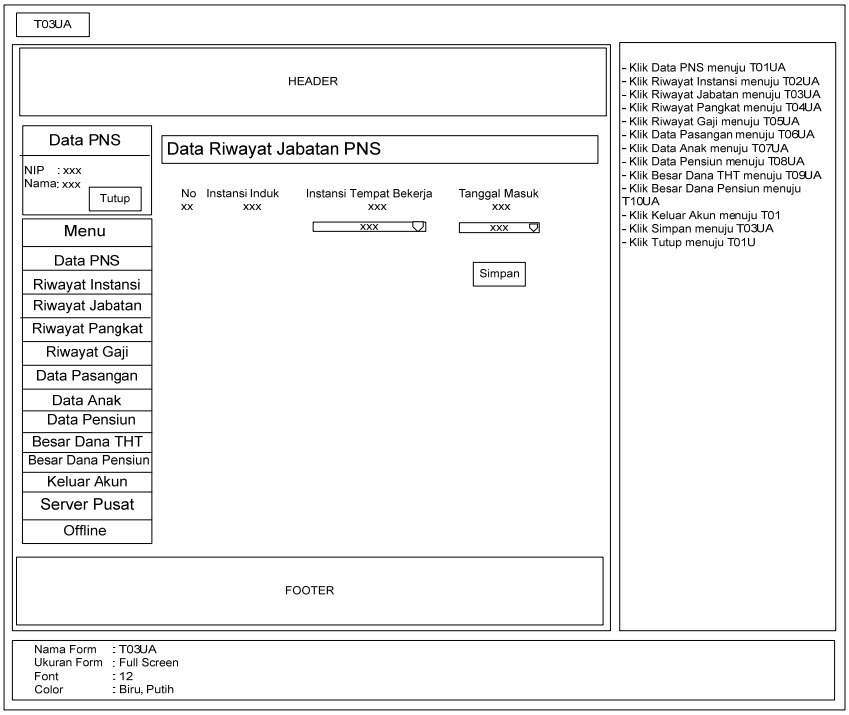
Gambar 3.52 Tampilan Isi Tambah PNS



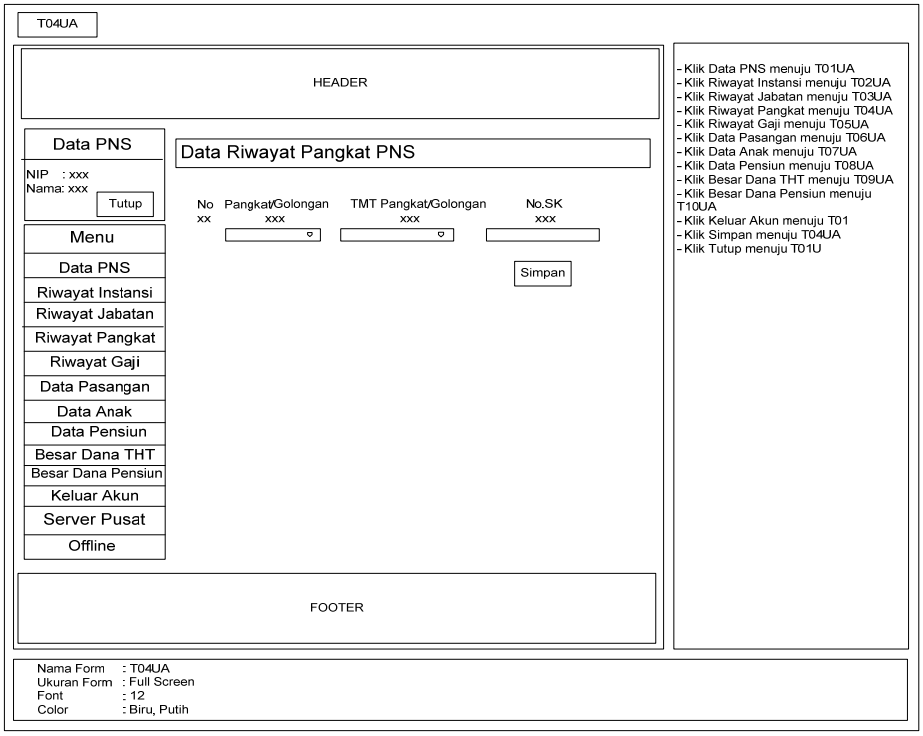
Gambar 3.53 Tampilan Data PNS



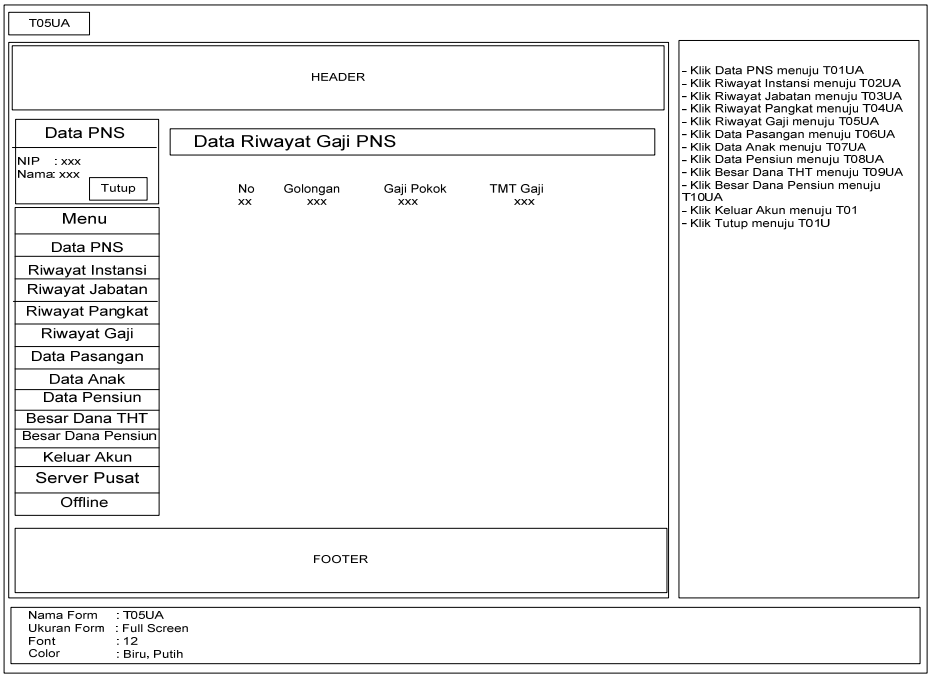
Gambar 3.54 Tampilan Data Riwayat Instansi



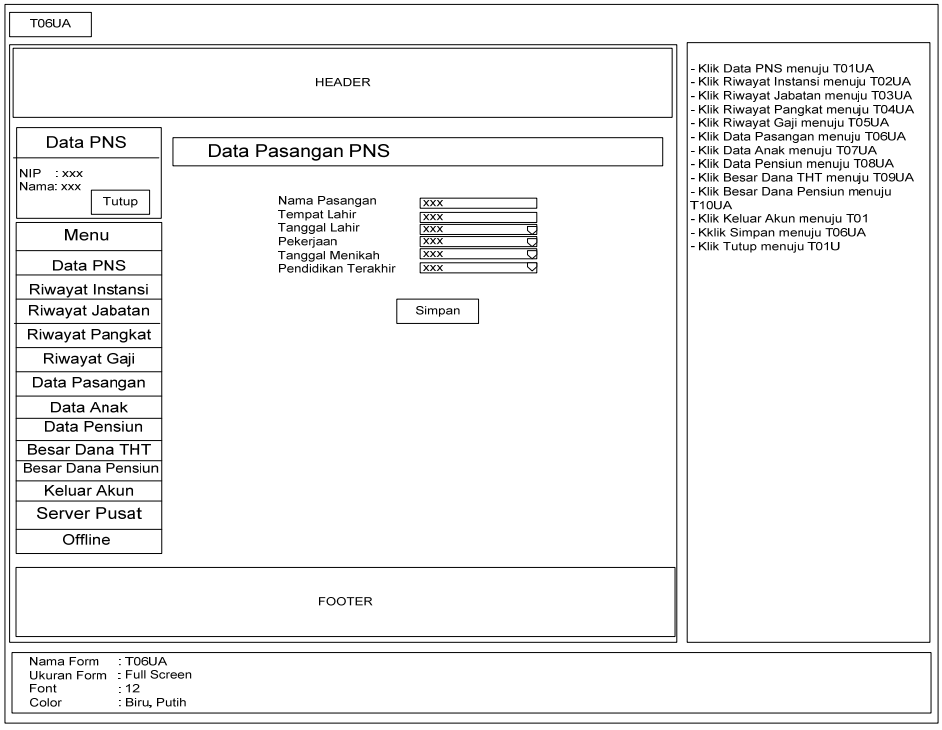
Gambar 3.55 Tampilan Data Riwayat Jabatan



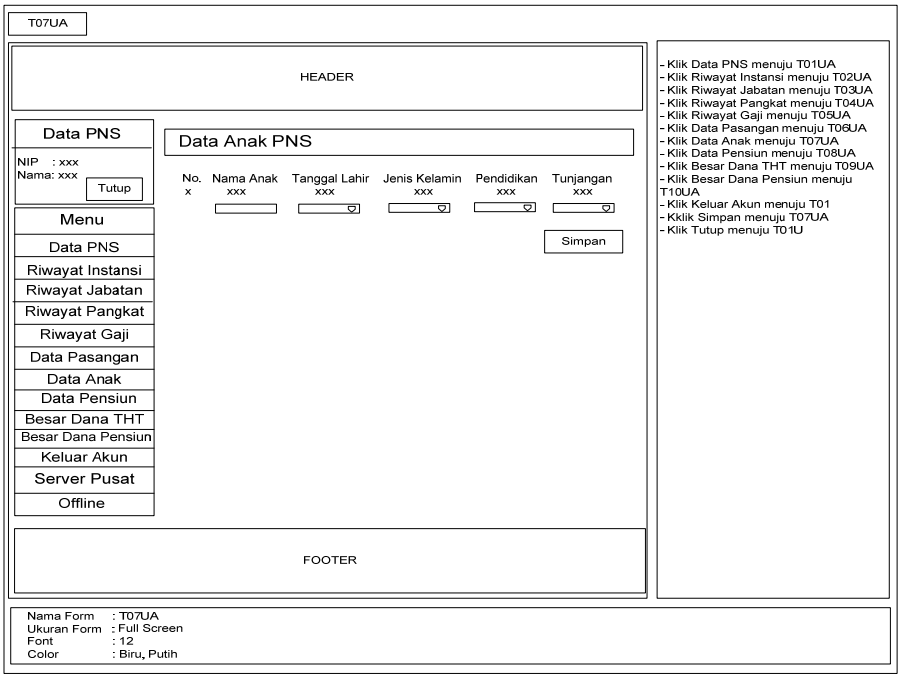
Gambar 3.56 Tampilan Data Riwayat Pangkat



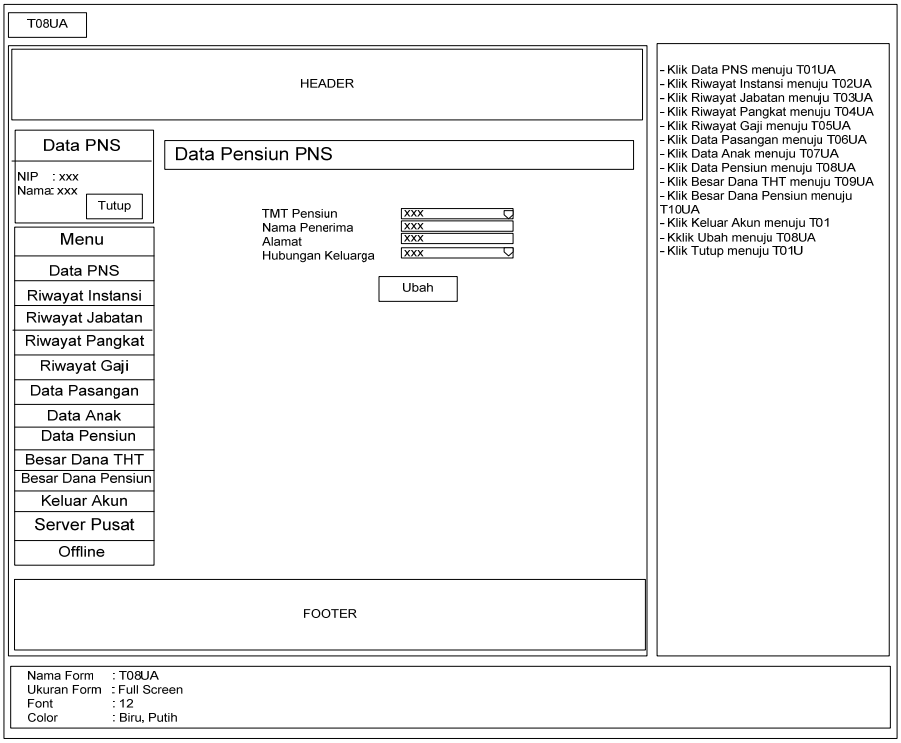
Gambar 3.57 Tampilan Data Riwayat Gaji



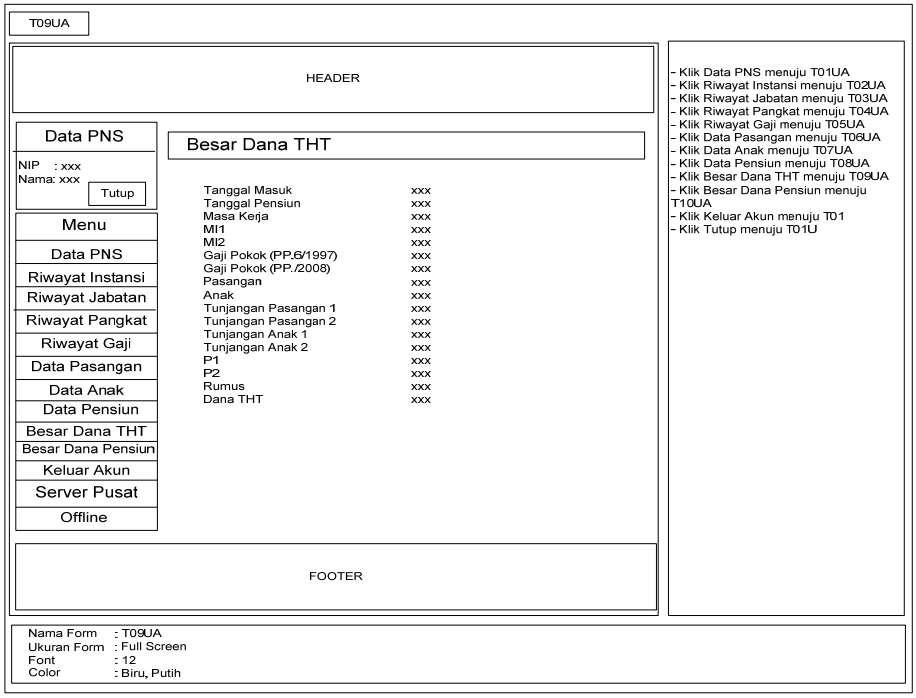
Gambar 3.58 Tampilan Data Riwayat Pasangan



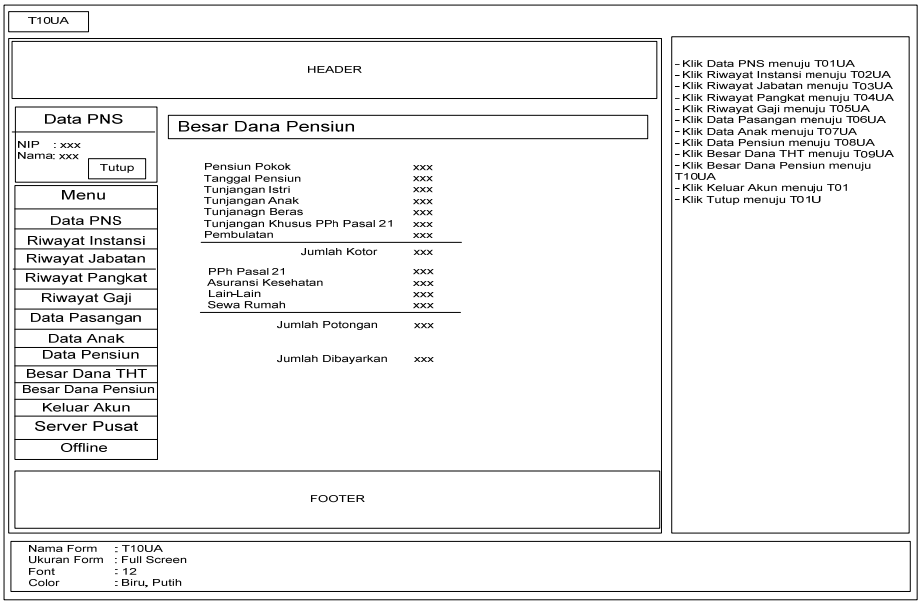
Gambar 3.59 Tampilan Data Riwayat Anak



Gambar 3.60 Tampilan Data Pensiun



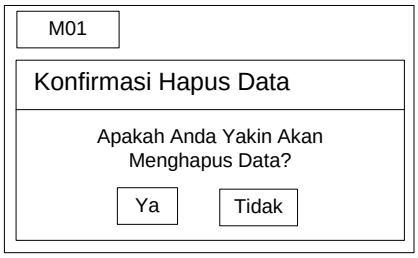
Gambar 3.61 Tampilan Besar Dana THT



Gambar 3.62 Tampilan Besar Dana Pensiun

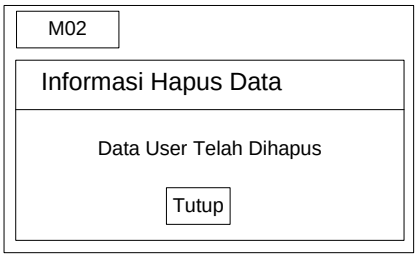
3.2.4.2. Perancangan Pesan

M01



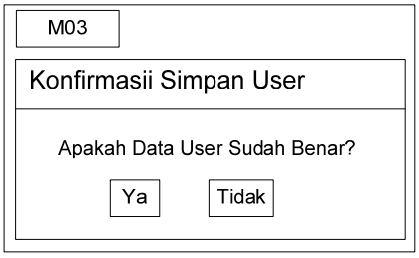
Gambar 3.63 Tampilan Pesan M01

M02



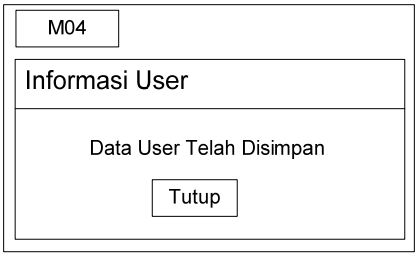
Gambar 3.64 Tampilan Pesan M02

M03



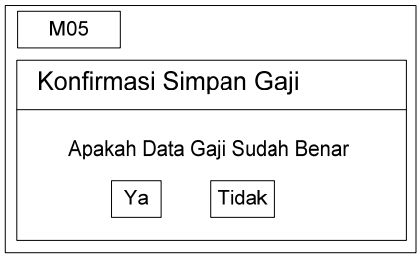
Gambar 3.65. Tampilan Pesan M03

M04



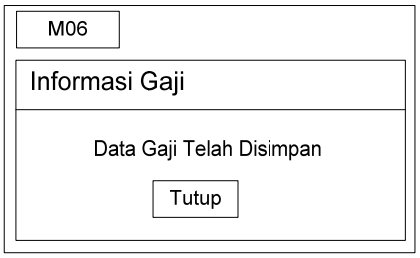
Gambar 3.66. Tampilan Pesan M04

M05



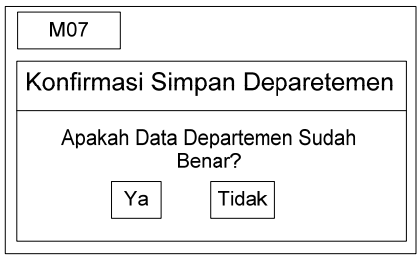
Gambar 3.67. Tampilan Pesan M05

M06



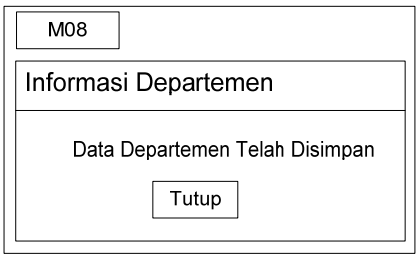
Gambar 3.68. Tampilan Pesan M06

M07



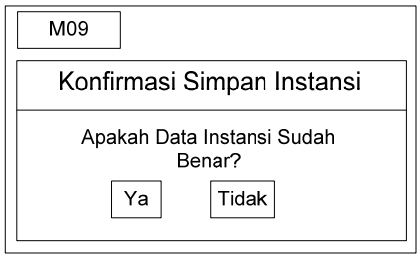
Gambar 3.69. Tampilan Pesan M07

M08



Gambar 3.70. Tampilan Pesan M08

M09



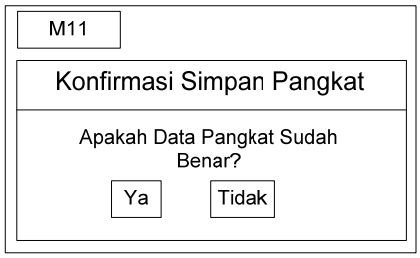
Gambar 3.71. Tampilan Pesan M09

M10



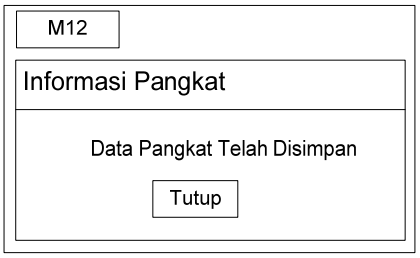
Gambar 3.72. Tampilan Pesan M10

M11



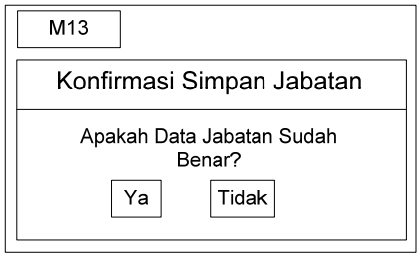
Gambar 3.73. Tampilan Pesan M11

M12



Gambar 3.74. Tampilan Pesan M12

M13



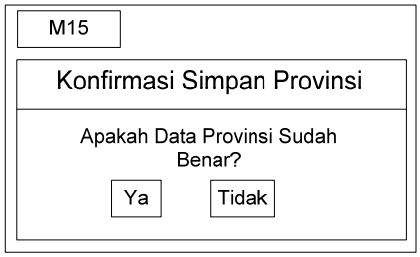
Gambar 3.75. Tampilan Pesan M13

M14



Gambar 3.76. Tampilan Pesan M14

M15



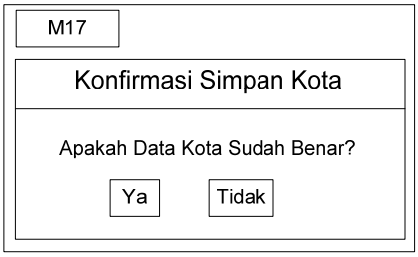
Gambar 3.77. Tampilan Pesan M15

M16



Gambar 3.78. Tampilan Pesan M16

M17



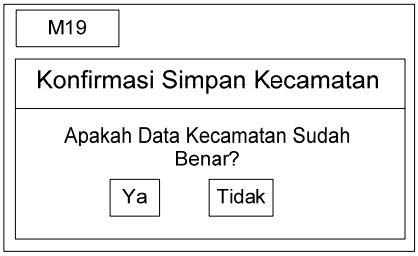
Gambar 3.79. Tampilan Pesan M17

M18



Gambar 3.80. Tampilan Pesan M18

M19



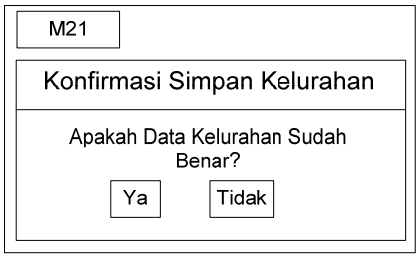
Gambar 3.81. Tampilan Pesan M19

M20



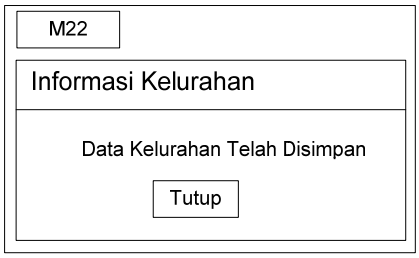
Gambar 3.82. Tampilan Pesan M20

M21



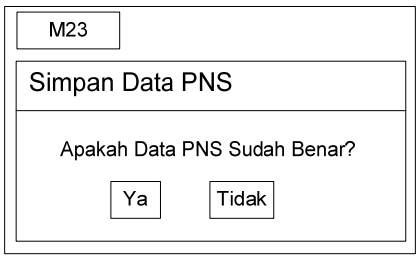
Gambar 3.83. Tampilan Pesan M21

M22



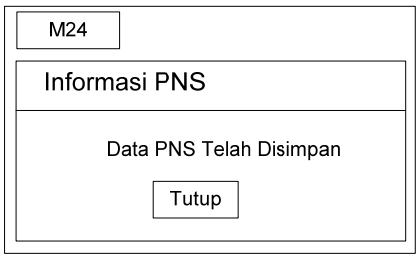
Gambar 3.84. Tampilan Pesan M22

M23



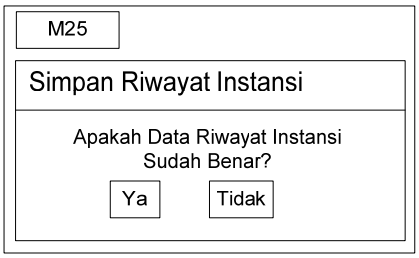
Gambar 3.85. Tampilan Pesan M23

M24



Gambar 3.86. Tampilan Pesan M24

M25



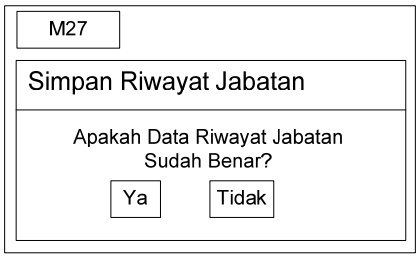
Gambar 3.87. Tampilan Pesan M25

M26



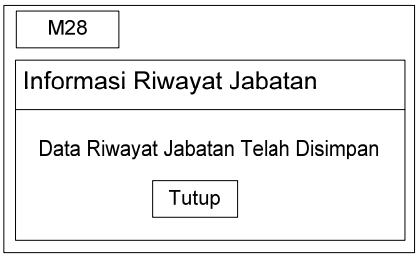
Gambar 3.88. Tampilan Pesan M26

M27



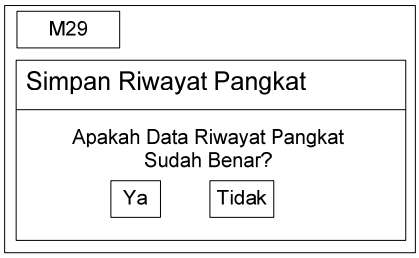
Gambar 3.89. Tampilan Pesan M27

M28



Gambar 3.90. Tampilan Pesan M28

M29



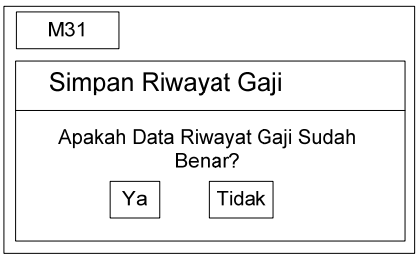
Gambar 3.91. Tampilan Pesan M29

M30



Gambar 3.92. Tampilan Pesan M30

M31



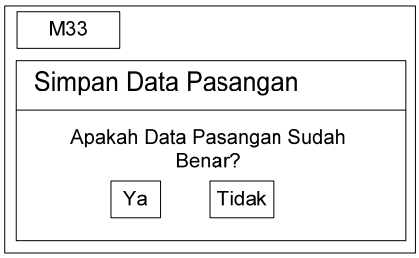
Gambar 3.93. Tampilan Pesan M31

M32



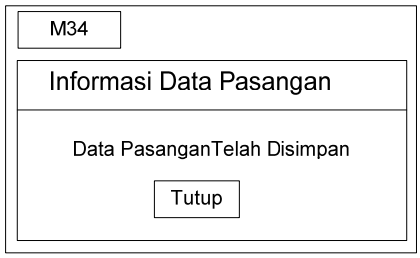
Gambar 3.94. Tampilan Pesan M32

M33



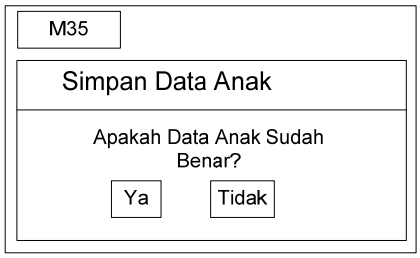
Gambar 3.95. Tampilan Pesan M33

M34



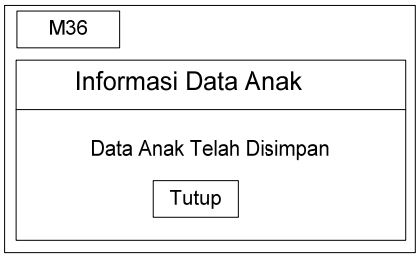
Gambar 3.96. Tampilan Pesan M34

M35



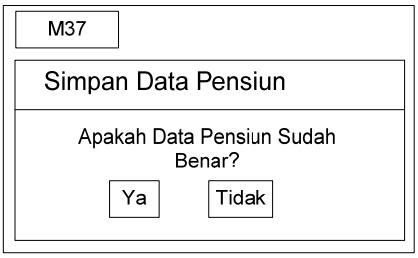
Gambar 3.97. Tampilan Pesan M35

M36



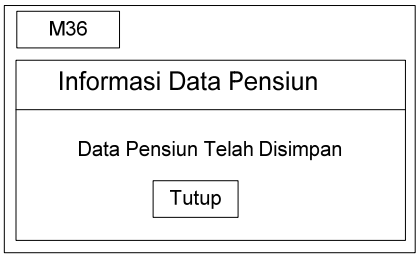
Gambar 3.98. Tampilan Pesan M36

M37



Gambar 3.99. Tampilan Pesan M37

M38

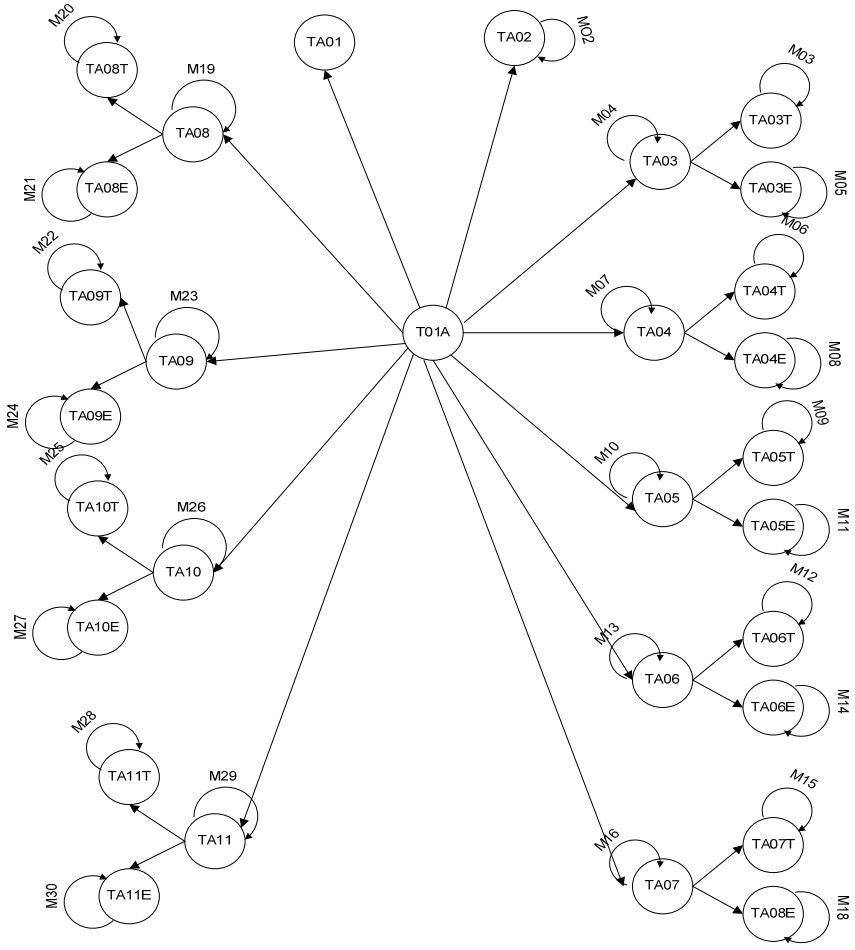


Gambar 3.100. Tampilan Pesan M38

3.2.4.3. Jaringan Semantik

Jaringan semantik pada Aplikasi Pengolahan Data Kepegawaian adalah sebagai berikut:

Jaringan Semantik

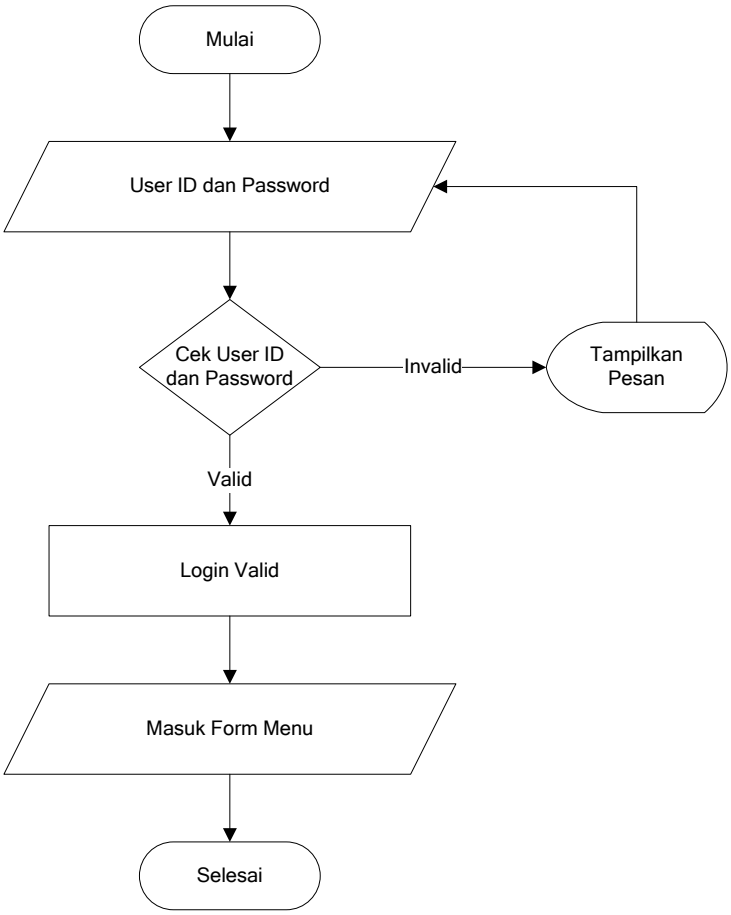


Gambar 3.101 Jaringan Semantik

3.2.5 Perancangan Prosedural

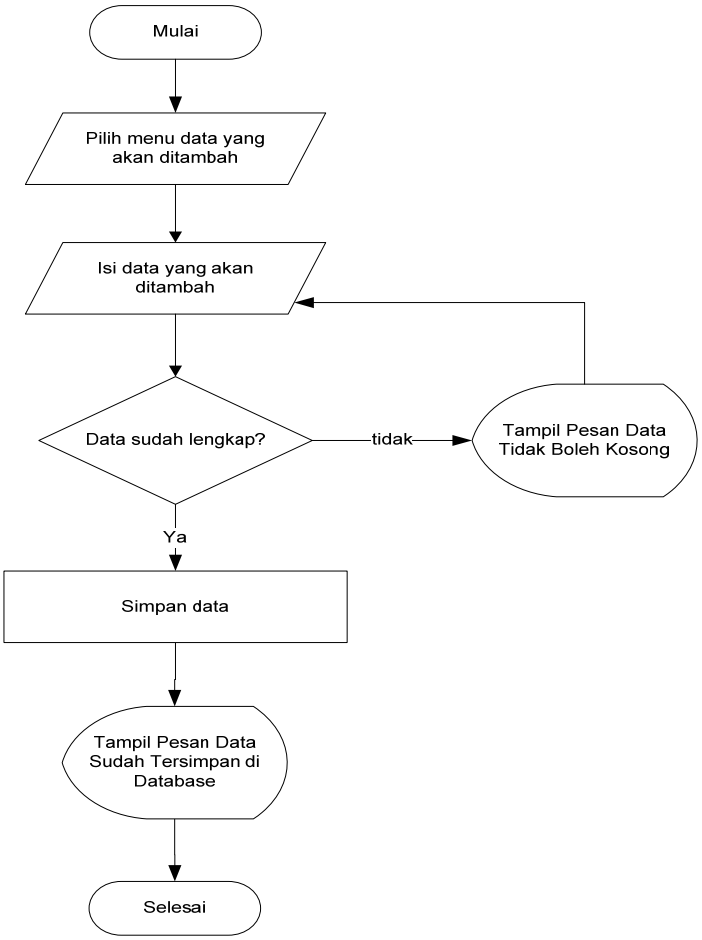
Perancangan prosedural mentransformasi elemen-elemen struktural dari ar*site*ktur program ke dalam suatu deskripsi prosedural dari komponen-komponen perangkat lunak. Adapun perancangan prosedural untuk aplikasi yang akan dibangun adalah sebagai berikut:

1. Prosedural *login*, merupakan prosedur yang terjadi ketika admin atau guru akan mengakses aplikasi. Prosedural *login* dapat dilihat pada gambar di bawah ini :



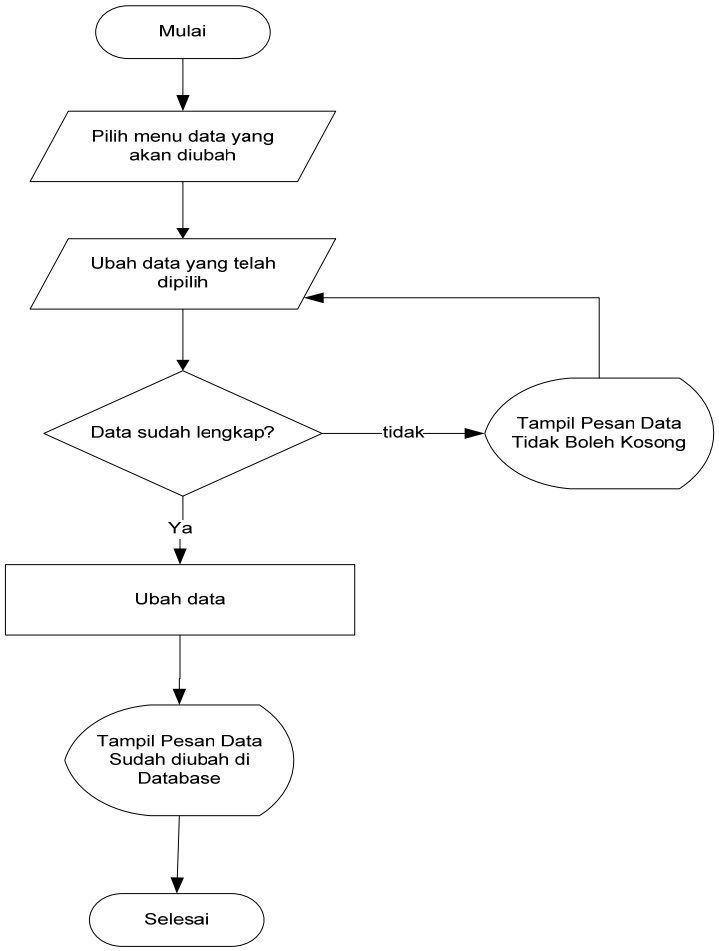
Gambar 3.102 Prosedural *Login*

2. Prosedural tambah, prosedur ini dilakukan ketika penggunaakan melakukan penambahan data. Prosedural tambahdapat dilihat pada gambar di bawah ini :



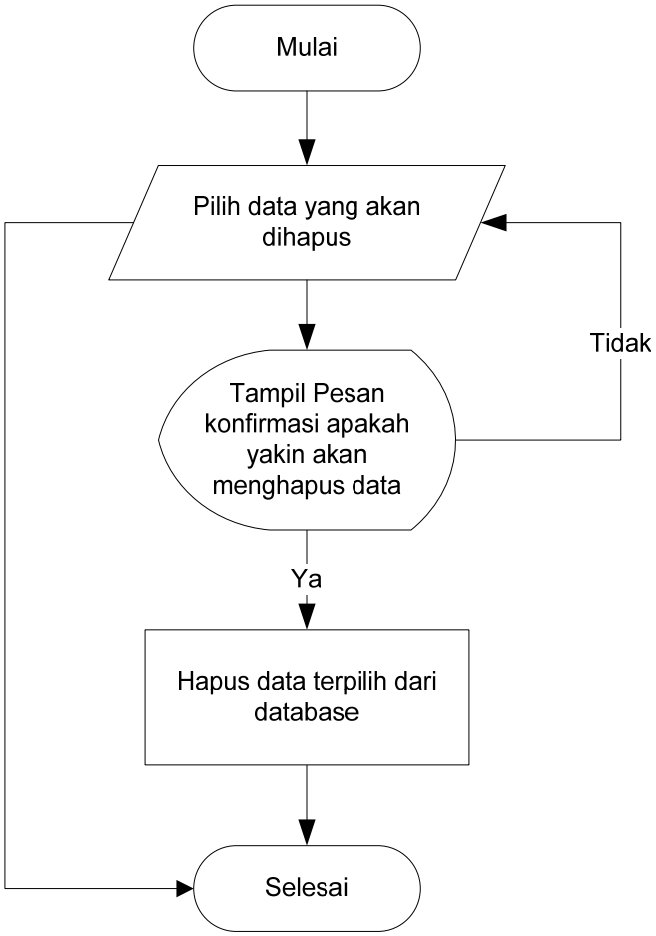
Gambar 3.103. Prosedural Tambah

3. Prosedural ubah, prosedur ini dilakukan ketika penggunaakan melakukan perubahan data. Prosedural ubahdapat dilihat pada gambar di bawah ini :



Gambar 3.104 Prosedural Ubah

4. Prosedural hapus, prosedur ini dilakukan ketika *user*akan melakukan penghapusan data. Prosedural hapusdapat dilihat pada gambar di bawah ini :



Gambar 3.105 Prosedural Hapus