

## BAB III

### METODELOGI PENELITIAN

#### 3.1 Tempat dan Waktu Penelitian

Penelitian ini mengambil tempat di Balai desa Panunggalan. Adapun penelitian akan dilaksanakan mulai bulan Maret 2015 sampai dengan Oktober 2015.

#### 3.2 Alat dan Bahan Penelitian

Penelitian ini membutuhkan piranti-piranti untuk mendukung berjalannya perancangan dan implementasi aplikasi menggunakan laptop dengan spesifikasi sebagai berikut :

1. Intel core I 5
2. RAM 4Gb
3. Hardisk 500Gb

Perangkat lunak pendukung yang diperlukan dalam pelaksanaan penelitian di Balai Desa Panunggalan yaitu:

1. *Microsoft Visual Studio 2012* digunakan untuk merancang interface dan juga *coding script* aplikasi.
2. *MySql 5.6* digunakan untuk server database.
3. *Xamp* di gunakan sebagai penyambung antara *MySql* agar data base dapat di kelola melalui *localhost*.
4. *Crome* digunakan untuk mengelola database dengan menggunakan *interface*.
5. *Corel draw* digunakan untuk mendesain gambar rancangan interface aplikasi.
6. *Microsoft Visio 2010* digunakan untuk membuat diagram.

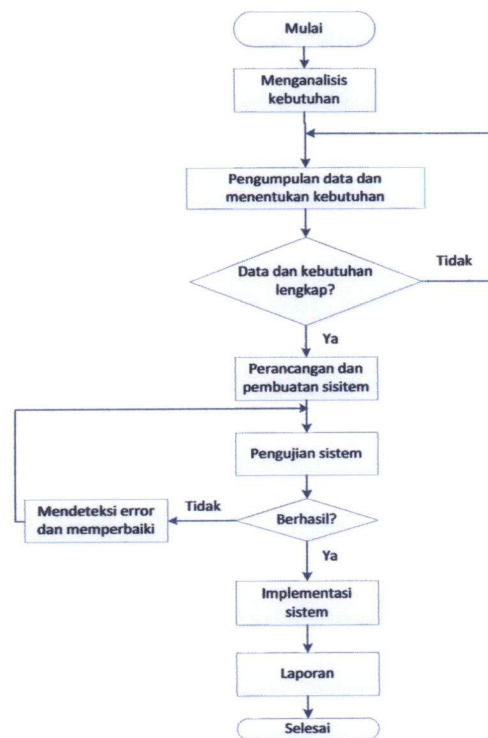
fungsi sebagai berikut:

1. Menambah data penduduk
2. Mengubah data penduduk
3. Menghapus data penduduk
4. Mencetak surat keterangan kelahiran
5. Mencetak surat keterangan kematian
6. Mencetak surat permohonan KTP
7. Mencetak surat keterangan domisili
8. Mencetak surat keterangan usaha
9. Mencetak surat keterangan tidak mampu
10. Mencetak surat ijin keramaian
11. Menambah pegawai
12. Mengubah data kepegawaian
13. Menghapus pegawai
14. Mengubah profil pengguna
15. Menambah pengguna
16. Pengarsipan surat

### **3.3 Alur Penelitian**

Tahap pada alur penelitian di atas digambarkan dalam *flow chart* pada gambar

3.1



**Gambar 3.1** Alur penelitian

Penelitian Program Aplikasi Administrasi Persuratan di Balai Desa Panunggalan menggunakan metode SDLC, hal ini bertujuan apabila dalam perjalanan sistem tersebut terdapat kesalahan, kerusakan ataupun *error* maka dilakukan analisis kebutuhan kembali dari awal memperbaiki sistem.

Alur penelitian penulis dilakukan dalam beberapa tahap sebagai berikut:

#### 1. Menganalisis kebutuhan

Analisis kebutuhan pada dasarnya merupakan tahap merancang dan membangun sebuah sistem informasi. Analisis kebutuhan menyangkup kebutuhan *software* dan kebutuhan *hardware*, analisis kebutuhan isi dan interaksi menu pada aplikasi.

Sebelum penulis membuat program aplikasi persuratan, terlebih dahulu melakukan wawancara terhadap petugas kantor Balai Desa Panunggalan selaku obyek dan juga selaku pengguna aplikasi ini. Wawancara ini dilakukan untuk

administrasi persuratan.

Berdasarkan dari hasil wawancara yang telah penulis lakukan, maka penulis dapat menarik kesimpulan bahwa kebutuhan dikategorikan dalam dua kelompok yaitu:

1. Kebutuhan fungsional sistem adalah aktifitas dan pelayanan yang harus dimiliki oleh sebuah sistem berupa *input*, proses, *output*, maupun penyimpanan data. Berdasarkan kebutuhan sistem secara fungsional, aplikasi yang dirancang harus mampu memenuhi kebutuhan fungsional sebagai berikut:

- a. Pengguna memasukkan *input* dalam pencarian dapat berupa nama, dusun, RT, RW, jenis kelamin.
- b. Pengguna memasukkan *input* dalam penambahan penduduk desa dengan mengisikan data diri dari penduduk baru.
- c. Pengguna memasukkan *input* dalam penambahan surat masuk dengan memasukan data surat.
- d. *Input* dapat berupa masukan langsung maupun secara otomatis saat pembuatan surat.

2. Kebutuhan nonfungsional sistem adalah karakteristik atau batasan yang menentukan kepuasan sebuah sistem seperti kinerja, kemudahan penggunaan, biaya, dokumentasi, kontrol, dan kemampuan sistem bekerja tanpa mengganggu fungsional sistem lainnya.

a. Sisi performa, sistem yang dirancang memiliki :

- Antarmuka (*interface*) yang sederhana dan menarik.
- Sistem dilengkapi dengan informasi *sample*, yaitu cara menggunakan aplikasi.
- Waktu saat sistem bekerja relatif singkat.

b. Sisi kemudahan penggunaan, sistem yang dirancang memiliki :

- Tampilan antarmuka yang di rancang sedemikian rupa sehingga mirip dengan antarmuka perangkat lunak ternama pada umumnya, seperti *layout memubar*.



huruf, warna, dan *layout* dari antarmuka.

- Bahasa yang digunakan sederhana dan mudah dimenerti.

c. Sisi ekonomi, sistem yang dirancang memiliki:

- Instalasi perangkat lunak dan keras tidak memerlukan biaya lebih.

d. Sisi dokumentasi, sistem memiliki kemampuan sebagai berikut:

- Sistem dapat menyimpan *input* yang dimasukkan pengguna dalam database dan juga dokumen.

e. Sisi kontrol, sistem yang dirancang memiliki kemampuan :

- Sistem dapat menampilkan pesan kesalahan jika *input* yang dimasukkan pengguna tidak tepat.

f. Sisi kualitas, sistem yang dirancang memiliki :

- Kemampuan mengantisipasi apabila pengguna lupa akan kata sandi atau nama pengguna, yaitu dengan menggunakan pertanyaan keamanan yang diisi dan di jawab oleh pengguna aplikasi.
- Kemampuan ganda saat membuat surat kematian dengan otomatis sistem menghapus data penduduk yang meninggal.

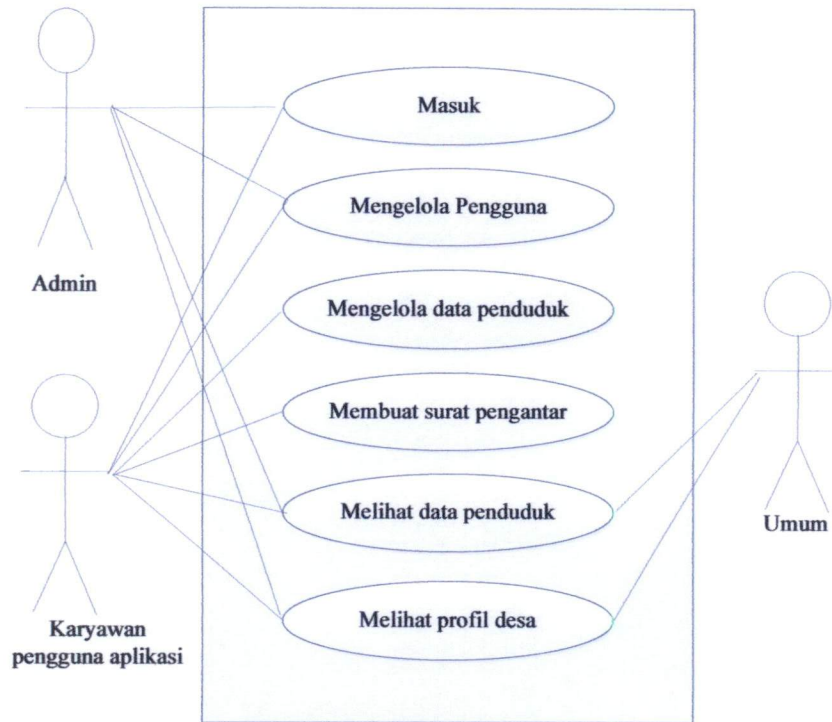
## 2. Pengumpulan data dan menentukan kebutuhan

Tahap ini merupakan tahap pengumpulan data–data yang diperlukan untuk terbangunnya penelitian program aplikasi administrasi persuratan di Balai Desa Panunggalan. Serta menentukan kebutuhan kebutuhan yang diperlukan untuk membuat penelitian tersebut. Baik dari segi kebutuhan *software*, *hardware* ataupun materi untuk pembuatan aplikasi dan *database*.

## 3. Perancangan dan pembuatan sistem

Sebelum pembuatan program aplikasi administrasi persuratan dilakukan, maka sebelumnya diperlukan perancangan sistem terlebih dahulu agar mempunyai gambaran tentang bagaimana sistem yang akan dibuat nantinya. Setelah tahap perancangan selesai maka akan dilanjutkan pada tahap pembuatan sistem. Pembuatan sistem ini dapat dilakukan setelah data–data dan materi yang diperlukan telah diperoleh semua.

balai desa Panunggalan baru dimulai oleh proses *requirement Analysis* yaitu mempelajari dan mempersiapkan kebutuhan untuk sistem baru. Proses-proses tersebut melibatkan aktor. Aktor yang terlibat dalam penelitian ini adalah admin, dimana aktor tersebut memiliki beberapa aktifitas. *Use case* yang akan dibuat dalam sistem secara umum seperti dijelaskan pada gambar 3.2.



**Gambar 3. 2** *Use case actor*

Berdasarkan *Use case* Gambar 3.2 dapat dilihat ketiga aktor dapat melakukan kegiatan berikut:

a. Admin

Admin mempunyai peran aktifitas dalam program aplikasi administrasi persuratan di Balai Desa Panunggalan yaitu menambahkan pengguna berdasarkan karyawan kantor.

sebagai berikut :

#### 1. Masuk

Dilakukan oleh admin agar dapat masuk ke dalam *form* tambah pengguna pada program aplikasi administrasi persuratan di balai desa Panunggalan.

#### 2. Mengelola pengguna

Dilakukan oleh admin untuk dapat mengelola data admin itu sendiri setelah dapat masuk kedalam halaman tambah pengguna aplikasi administrasi persuratan.

#### 3. Melihat data penduduk

Dilakukan oleh admin untuk melihat data penduduk didalam aplikasi administrasi persuratan.

#### 4. Melihat profil desa

Dilakukan oleh admin untuk melihat tentang profil desa Panunggalan didalam aplikasi administrasi persuratan.

#### b. Karyawan Pengguna Aplikasi

Admin mempunyai peran untuk mengelola data yang terdapat di dalam program aplikasi administrasi persuratan di Balai Desa Panunggalan. Dilihat dari *Use case* gambar 3.2 karyawan pengguna aplikasi dapat melakukan beberapa kegiatan sebagai berikut :

#### 1. Masuk

Dilakukan oleh pengguna agar dapat masuk ke dalam *form* utama program administrasi persuratan di Balai Desa Panunggalan.

#### 2. Mengelola pengguna

Dilakukan oleh pengguna untuk dapat mengelola profil pengguna itu sendiri setelah dapat masuk kedalam halaman utama aplikasi administrasi persuratan.

#### 3. Mengelola data penduduk

Dilakukan oleh pengguna untuk mengelola data penduduk yang terdapat dalam aplikasi administrasi persuratan.

#### 4. Membuat surat pengantar

Dilakukan oleh pengguna untuk membuat surat pengantar apabila ada pemohon yang mengajukan.

administrasi persuratan.

#### 6. Melihat profil desa

Dilakukan oleh pengguna untuk melihat tentang profil desa Panunggalan didalam aplikasi administrasi persuratan.

#### c. Umum

Peran umum pada program aplikasi administrasi persuratan di balai desa Panunggalan adalah petugas lain yang terdapat di kantor Balai Desa panunggalan akan tetapi bukan merupakan petugas utama pembuatan surat lampiran. Menurut *Use case* pada gambar 3.2 aktor umum memiliki dua kegiatan yaitu :

##### 1. Melihat data penduduk

Dilakukan oleh aktor umum untuk melihat data penduduk didalam aplikasi administrasi persuratan tanpa harus memasukan nama pengguna dan kata sandi terlebih dahulu.

##### 2. Melihat profil desa

Dilakukan oleh aktor umum untuk melihat tentang profil desa Panunggalan didalam aplikasi administrasi persuratan tanpa harus memasukan nama pengguna dan kata sandi terlebih dahulu.

#### c. *Use Case Definition*

Pendefinisian *use case* dari sistem yang dibuat sebagai berikut:

##### 1. *Input* akses masuk admin

Digunakan oleh admin untuk masuk ke *form* utama program aplikasi administrasi persuratan di Balai Desa Panunggalan.

##### 2. Mengelola pengguna

Dilakukan oleh admin setelah berhasil masuk kedalam *form* utama aplikasi untuk mengubah data admin itu sendiri mengenai nama pengguna, kata sandi pengguna, pertanyaan keamanan, jawaban pertanyaan keamanan.

##### 3. Mengelola data

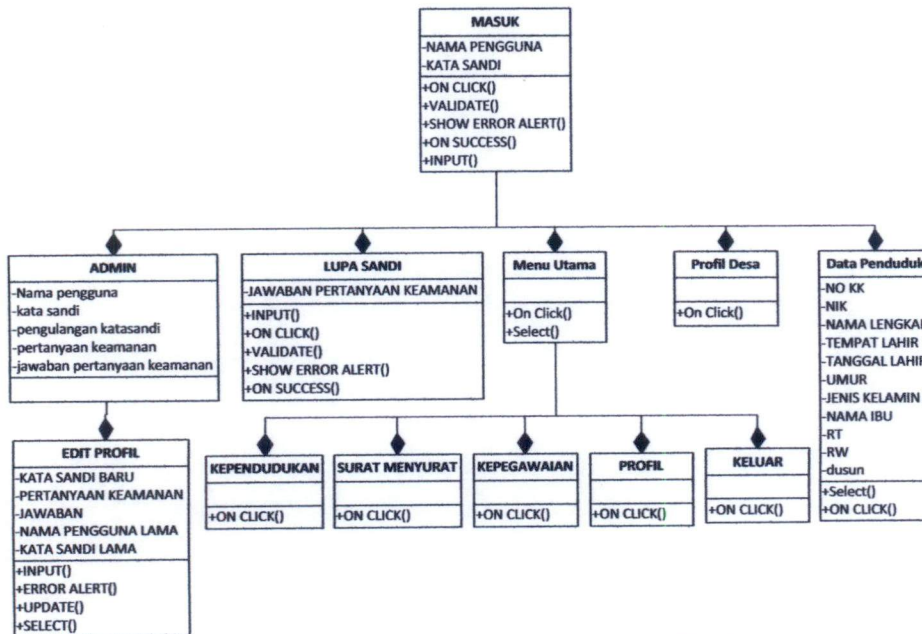
Digunakan oleh admin untuk dapat menambahkan, mengurangi maupun mengubah data penduduk Desa panunggalan.



penduduk, surat kelahiran, surat kematian, surat ijin keramaian maupun surat yang lain apabila ada warga desa Panunggalan selaku pemohon ingin membuatnya.

### 3.3.2 Class Diagram

*Class diagram* pada program aplikasi administrasi persuratan ini menjelaskan tentang kumpulan dari kelas dan hubungan yang struktural. *Class diagram* untuk perancangan program aplikasi persuratan berbasis desktop dapat dilihat pada gambar 3.3 sampai dengan gambar 3.7



**Gambar 3. 3 Class Diagram Aplikasi**

Pada gambar 3.3 menjelaskan bahwa aplikasi dibagi atas beberapa *class* terdiri dari:

Masuk : adalah *class* yang meliputi kegiatan masuk atau *login*. *Class* masuk memiliki nama pengguna dan kode sandi sebagai atribut sedangkan *on click*, *validate*, *show error alert*, *on success* dan *input* sebagai *method*.

Lupa sandi: adalah *class* yang meliputi kegiatan masuk apabila lupa nama pengguna atau kata sandi dengan cara menjawab pertanyaan. *Class* lupa sandi



*validate, error alert, on success dan input sebagai method.*

*Form Utama* : adalah *class* yang yang berisi menu-menu utama seperti kependudukan, surat menyurat, kepegawaian dan profil. Pengguna juga dapat melihat data penduduk.

*Profil Desa* : adalah *class* agar prngguna dapat melihat profil desa Panunggalan.

*Data penduduk*: adalah *class* agar pengguna dapat melihat data penduduk tanpa harus mengisikan nama pengguna dan kata sandi. Dalam *class* data penduduk terdapat no, nokk, nik, nama lengkap, tempat lahir, tanggal lahir, jenis kelamin, nama ibu, rt, rw dan dusun sebagai atribut dan input, *on click*, dan *select* sebagai *method*.

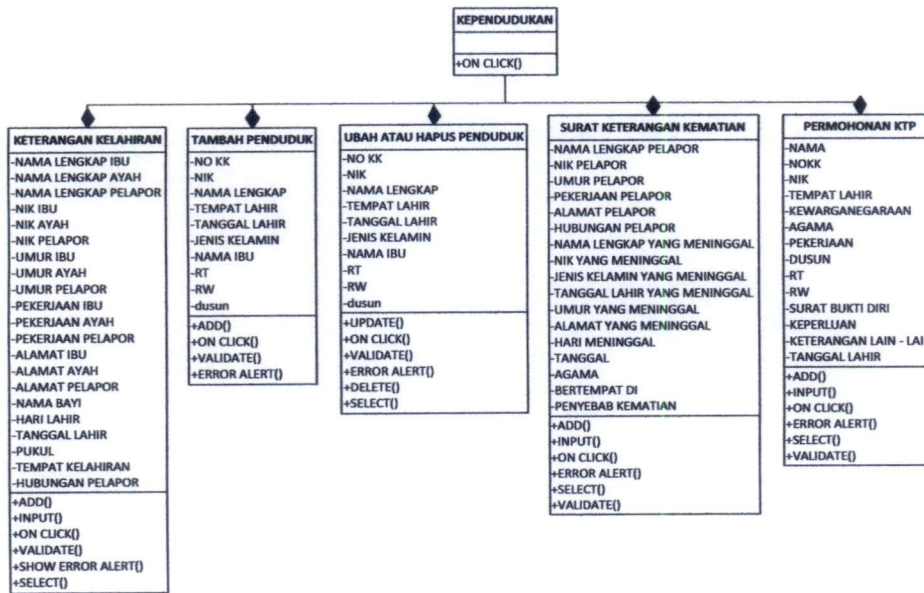
*Kependudukan* : adalah *class* yang meliputi kegiatan untuk membuka kumpulan *submenu* kependudukan. *Class* kependudukan memiliki beberapa *class* turunan yang akan dijelaskan secara lebih detail pada gambar 3.4 *Class* kependudukan tidak memiliki atribut tetapi memiliki *on click* sebagai *method*.

*Surat menyurat* : adalah *class* yang meliputi kegiatan untuk membuka kumpulan *submenu* dari menu surat menyurat. *Class* surat menyurat memiliki beberapa *class* turunan yang akan dijelaskan secara lebih detail pada gambar 3.5 *Class* kependudukan tidak memiliki atribut tetapi memiliki *on click* sebagai *method*.

*Kepegawaian* : adalah *class* yang meliputi kegiatan untuk membuka kumpulan *submenu* dari menu kepegawaian. *Class* kepegawaian memiliki beberapa *class* turunan yang akan dijelaskan secara lebih detail pada gambar 3.6 *Class* kependudukan tidak memiliki atribut tetapi memiliki *on click* sebagai *method*.

*Profil* : adalah *class* yang meliputi kegiatan untuk membuka kumpulan *submenu* dari menu profil. *Class* profil memiliki beberapa *class* turunan yang akan dijelaskan secara lebih detail pada gambar 3.7 *Class* kependudukan tidak memiliki atribut tetapi memiliki *on click* sebagai *method*.

dialihkan ke menu masuk. *Class* keluar tidak memiliki atribut tetapi memiliki *on click* sebagai *method*.



**Gambar 3. 4** *Class Diagram* Kependudukan

Pada *class* kependudukan memiliki beberapa *class* turunan yang akan di jelaskan sebagai berikut :

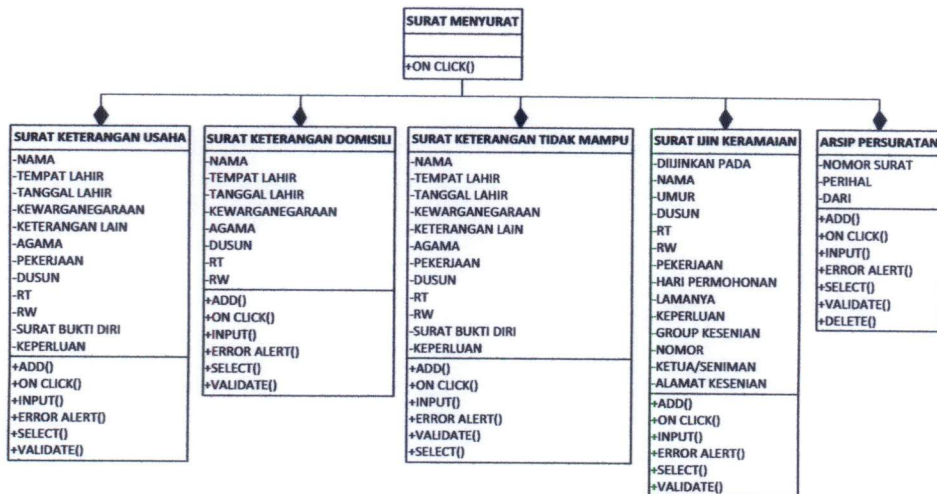
Keterangan kelahiran : merupakan *class* yang digunakan pengguna untuk membuat surat keterangan kelahiran. Pada kelas keterangan kelahiran memiliki nama lengkap ibu, nama lengkap ayah, nama lengkap pelapor, nik ibu, nik ayah, nik pelapor, umur ibu, umur ayah, umur pelapor, pekerjaan ibu, pekerjaan ayah, pekerjaan pelapor, alamat ibu, alamat pelapor, alamat ayah, nama bayi, hari lahir, tanggal lahir, pukul, tempat kelahiran dan hubungan pelapor sebagai atribut dan input, *on click*, *validate*, *select* dan *show error alert* sebagai *method*.

Tambah penduduk : merupakan *class* yang digunakan pengguna untuk menambah penduduk. Pada kelas tambah penduduk memiliki nokk, nik, nama lengkap, tempat lahir, tanggal lahir, jenis kelamin, nama ibu, rt, rw dan dusun sebagai atribut dan input, *on click*, *validate*, *add* dan *error alert* sebagai *method*.

untuk mengubah data penduduk ataupun menghapus penduduk. Pada kelas ubah atau hapus penduduk memiliki nokk, nik, nama lengkap, tempat lahir, tanggal lahir, jenis kelamin, nama ibu, rt, rw dan dusun sebagai atribut dan input, *on click*, *validate*, *update*, *delete*, *select* dan *error alert* sebagai *method*.

Surat keterangan kematian : merupakan *class* yang digunakan pengguna untuk membuat surat keterangan kematian. Pada kelas keterangan kematian memiliki nama lengkap pelapor, nik pelapor, pekerjaan pelapor, alamat pelapor, umur pelapor, hubungan pelapor, nama lengkap meninggal, nik meninggal, jenis kelamin meninggal, tanggal lahir meninggal, jenis kelamin meninggal, alamat meninggal, hari meninggal, tanggal, agama, bertempat di, penyebab kematian sebagai atribut dan *input*, *select*, *on click*, *validate* dan *error alert* sebagai *method*

Permohonan ktp : merupakan *class* yang digunakan pengguna untuk membuat surat permohonan ktp. Pada kelas permohonan ktp memiliki nama, nokk, nik, tempat lahir, tanggal lahir, kewarganegaraan, dusun, agama, rt, rt, surat bukti diri, keterangan lain – lain dan keperluan sebagai atribut *input*, *on click*, *select*, *validate* dan *error alert* sebagai *method*.



Gambar 3. 5 Class Diagram Surat Menyurat

Pada *class* surat menyurat terdapat beberapa *class* turunan yang akan di jelaskan sebagai berikut :



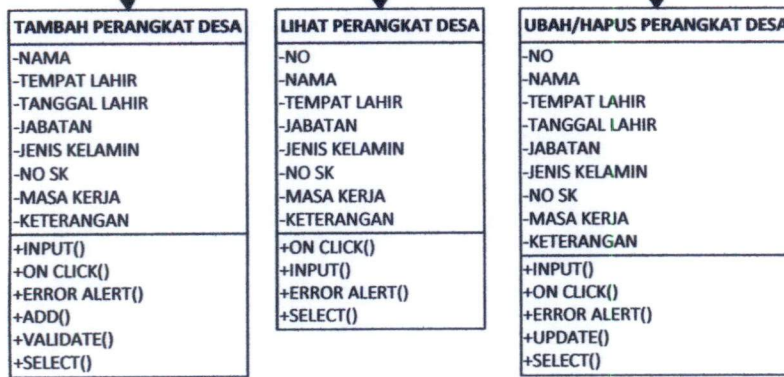
membuat surat pengantar keterangan usaha. Pada kelas surat keterangan usaha memiliki nama, tempat lahir, tanggal lahir, kewarganegaraan, dusun, agama, rt, rw, surat bukti diri, keterangan lain – lain, pekerjaan dan keperluan sebagai atribut *input, on click, validate, select* dan *error alert* sebagai *method*.

Surat keterangan tidak mampu : merupakan *class* yang digunakan pengguna untuk membuat surat pengantar keterangan tidak mampu. Pada kelas surat keterangan tidak mampu memiliki nama, tempat lahir, tanggal lahir, kewarganegaraan, dusun, agama, rt, rw, surat bukti diri, keterangan lain – lain, pekerjaan dan keperluan sebagai atribut *input, on click, validate, select* dan *error alert* sebagai *method*.

Surat keterangan domisili : merupakan *class* yang digunakan pengguna untuk membuat surat pengantar keterangan domisili. Pada kelas surat keterangan domisili memiliki nama, tempat lahir, tanggal lahir, kewarganegaraan, dusun, agama, rt dan rw sebagai atribut *input, on click, validate, select* dan *error alert* sebagai *method*.

Surat Keterangan Ijin Keramaian : merupakan *class* yang digunakan pengguna untuk membuat surat pengantar ijin keramaian. Pada kelas surat keterangan ijin keramaian memiliki nama, umur, dusun, rt, rw, pekerjaan, hari permohonan, lamanya, keperluan, group kesenian, nomor, ketua/seniman, alamat kesenian sebagai atribut *input, on click, validate, select* dan *error alert* sebagai *method*.

Arsip Persuratan: merupakan *class* yang digunakan pengguna untuk membuat arsip surat yang telah masuk ataupun surat yang telah dikeluarkan. Pada kelas Arsip Persuratan memiliki nomor surat, perihal dan dari sebagai atribut, sedangkan *input, on click, validate, select* dan *error alert* sebagai *method*.



**Gambar 3. 6** *Class Diagram* Turunan Kepegawaian

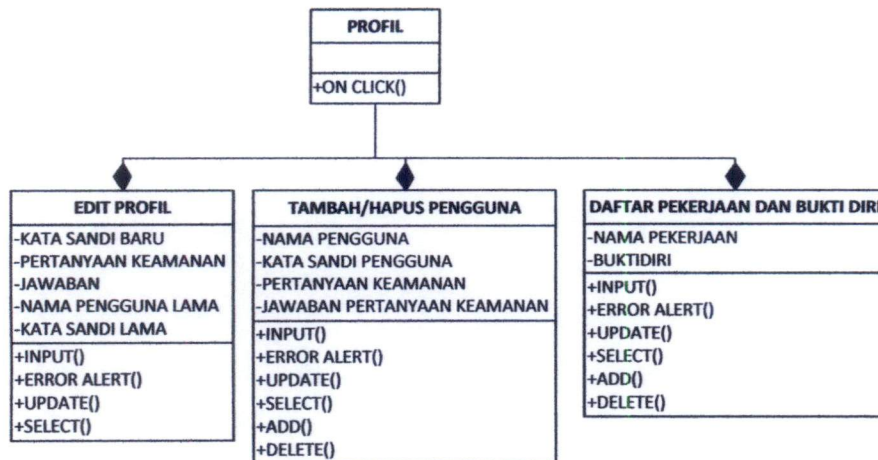
Pada *class* kepegawaian terdapat beberapa *class* turunan yang akan di jelaskan sebagai berikut :

Tambah perangkat desa : adalah *class* yang berfungsi untuk pengguna agar dapat menambah data perangkat desa di desa Panunggalan. Pada kelas tambah perangkat desa memiliki nama, tempat lahir, tanggal lahir, jabatan, jenis kelamin, no sk, masa kerja dan keterangan sebagai atribut sedangkan *input*, *on click*, *add*, *validate*, *select* dan *error alert* sebagai *method*.

Lihat perangkat desa : adalah *class* yang berfungsi untuk pengguna agar dapat melihat data perangkat desa di desa Panunggalan. Pada kelas lihat perangkat desa memiliki nama, tempat lahir, tanggal lahir, jabatan, jenis kelamin, no sk, masa kerja dan keterangan sebagai atribut sedangkan *input*, *on click*, *select* dan *error alert* sebagai *method*.

Ubah/Hapus perangkat desa : adalah *class* yang berfungsi untuk pengguna agar dapat mengubah ataupun menghapus data perangkat desa di desa Panunggalan. Pada kelas ubah perangkat desa memiliki no, nama, tempat lahir, tanggal lahir, jabatan, jenis kelamin, no sk, masa kerja dan keterangan sebagai atribut sedangkan *input*, *on click*, *update* dan *error alert*, *select* sebagai *method*.





**Gambar 3. 7** Class Diagram Profil

Pada *class* kepegawaian terdapat *class* turunan yaitu *edit* profil, *Tambah/hapus* pengguna dan *daftar pekerjaan dan bukti diri*. *Class – class* tersebut dijelaskan lebih detail seperti dibawah ini :

*Edit* profil adalah kelas agar pengguna dapat mengubah nama pengguna, kata sandi, pertanyaan keamanan, jawaban pertanyaan keamanan. Di dalam *edit* profil memiliki nama pengguna baru, kata sandi baru, pertanyaan keamanan, jawaban, nama pengguna lama, kata sandi lama sebagai atribut sedangkan *input*, *error alert* dan *update* sebagai *method*.

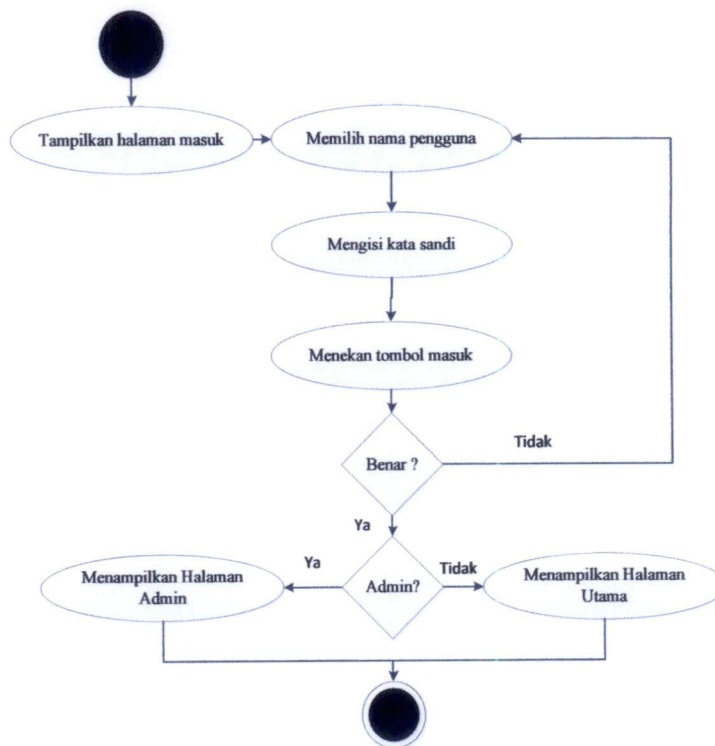
*Tambah/hapus* pengguna merupakan *class* agar pengguna dapat menambahkan ataupun menghapus *user* pada aplikasi. *Class* *Tambah/hapus* pengguna terdapat nama pengguna, kata sandi, pertanyaan keamanan, jawaban pertanyaan keamanan sebagai atribut, sedangkan *input*, *error alert* dan *update*, *delete*, *add* sebagai *method*.

*Daftar pekerjaan dan bukti diri* merupakan *class* agar pengguna dapat menambahkan, mengubah ataupun menghapus daftar nama pekerjaan dan bukti diri. Dalam *class* *daftar pekerjaan dan bukti diri* terdapat nama pekerjaan, *buktidiri* sebaai atribut dan terdapat *input*, *error alert* dan *update*, *delete*, *add* sebagai *method*.

dengan gambar 3.8 sampai 3.17.

#### a. Activity Diagram Masuk

Activity diagram masuk menggambarkan bagaimana proses pengguna login dengan cara memilih nama pengguna dan memasukan kata sandi pada form masuk agar dapat mengakses sistem. Apabila pengguna login menggunakan user admin, form admin akan ditampilkan sedangkan apabila dalam proses login pengguna menggunakan user selain admin yang akan di tampilkan adalah form utama. Gambar activity diagram masuk seperti gambar 3.8



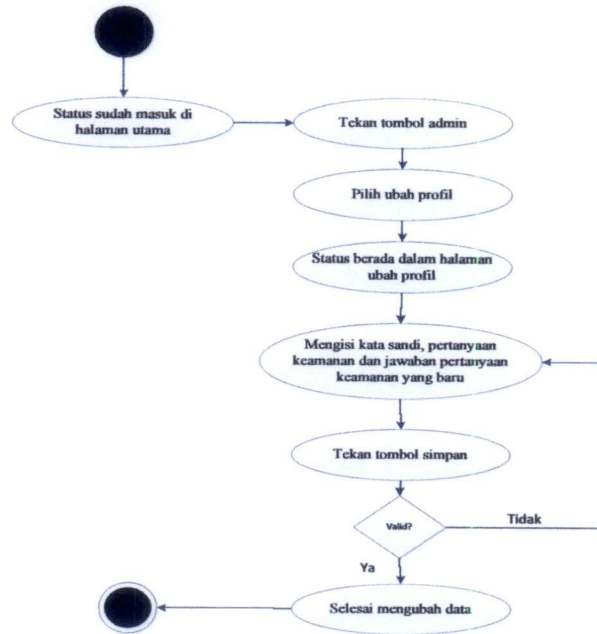
**Gambar 3. 8** Activity Diagram Masuk

#### b. Activity Diagram Keluar

Activity Diagram Keluar menggambarkan bagaimana proses keluar dari sistem yang di lakukan oleh user. Gambar Activity Diagram keluar ditunjukkan pada gambar 3.9

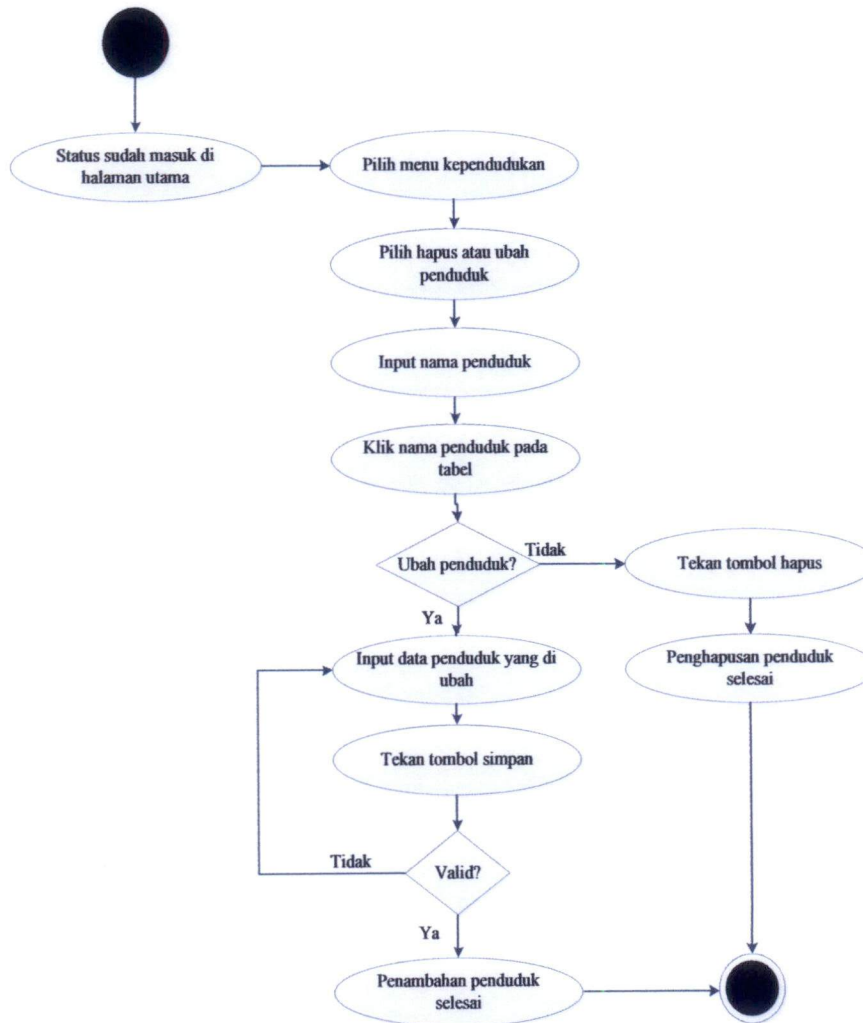
user mengubah kata sandi, pertanyaan keamanan, jawaban pertanyaan keamanan.

Activity Diagram mengelola pengguna ditunjukkan pada gambar 3.10

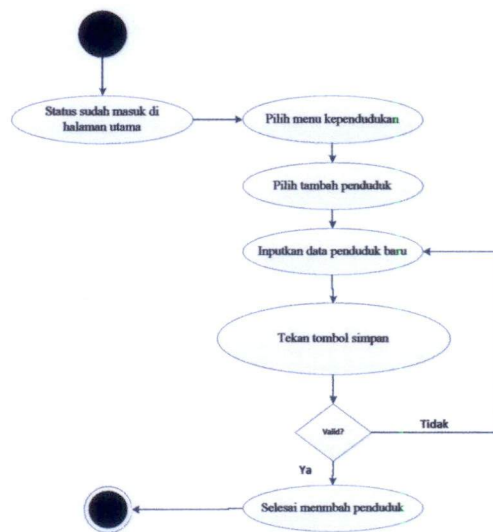


**Gambar 3. 10** Activity Diagram Mengelola Pengguna

*Activity diagram* mengelola data penduduk menggambarkan proses *user* melakukan penambahan, perubahan ataupun penghapusan penduduk. *Activity diagram* mengelola data penduduk akan ditunjukkan pada gambar berikut 3.11 dan 3.12



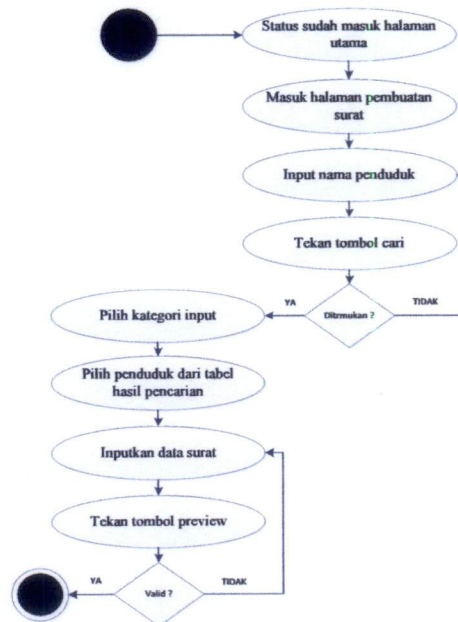
**Gambar 3. 11** *Activity Diagram* Mengubah atau Menghapus Data Penduduk



**Gambar 3. 12** Activity Diagram Menambah Data Penduduk

e. Activity Diagram Membuat Surat

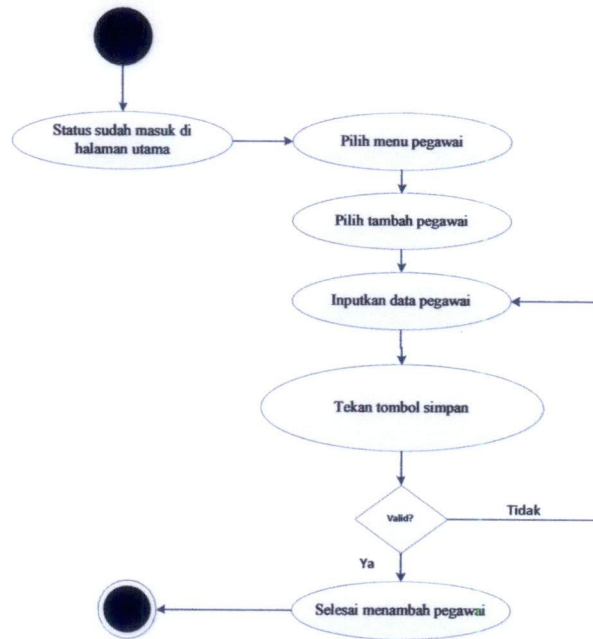
Activity diagram membuat surat adalah menggambarkan proses secara umum tentang pembuatan surat pengantar. Activity diagram membuat surat akan di tunjukan melalui gambar 3.13.



**Gambar 3. 13** Activity Diagram Membuat Surat



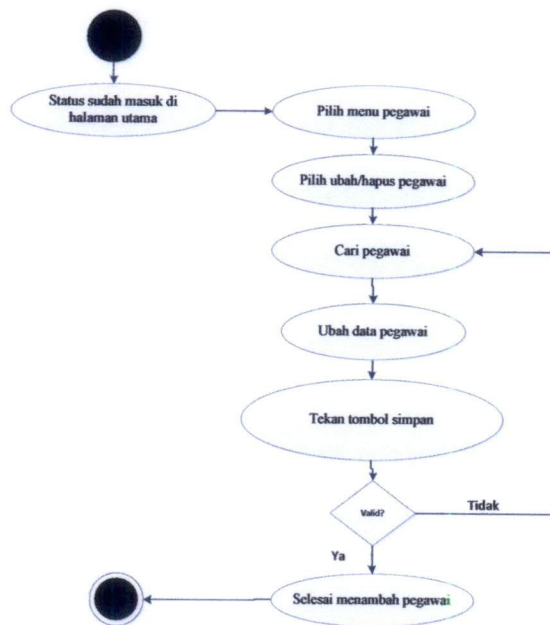
umum *user* tentang menambah pegawai kantor balai desa Pannggalan. *Activity* diagram menambah pegawai akan dijelaskan melalui gambar 3.14.



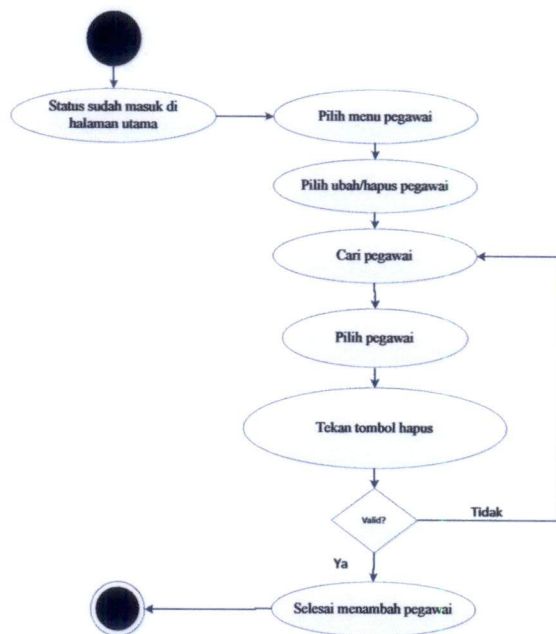
**Gambar 3. 14** *Activity Diagram* Menambah Pegawai

g. *Activity Diagram* Ubah/Hapus Pegawai

*Activity diagram* ubah pegawai adalah menggambarkan proses secara umum tentang mengubah maupun menghapus data pegawai. *Activity diagram* ubah/hapus data pegawai akan di tunjukan melalui gambar 3.15 dan gambar 3.16.

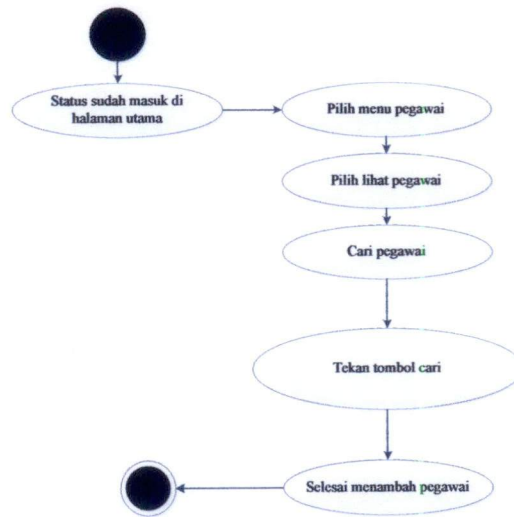


**Gambar 3. 15** Activity Diagram Ubah Pegawai



**Gambar 3. 16** Activity Diagram Hapus Pegawai

tentang melihat data pegawai. *Activity* diagram lihat data pegawai akan di tunjukan melalui gambar 3.17.



**Gambar 3. 17** *Activity Diagram* Lihat Pegawai

### 3.3.4 Perancangan Basis Data

Perancangan *database* pada Program Aplikasi Administrasi Persuratan Desa berasal dari input data pada transaksi yang terjadi dalam sistem administrasi kantor balai desa Panunggalan.

Perancangan basis data merupakan hasil dari wawancara yang telah dilakukan kemudian hasilnya disebut perancangan fisik yang diperlukan karena dalam tahap ini dilakukan penyesuaian agar data-data yang telah dirancang memenuhi kriteria efisiensi dalam menggunakan ruang penyimpanan, serta dapat memenuhi kriteria dalam pemograman beorientasi objek.

Dalam perancangan basis data diperlukan pembuatan desain tabel yang disertai dengan strukturnya. Maka dari itu untuk merancang program aplikasi persuratan desa, dibutuhkan tabel – tabel yang diperlukan untuk merancang *database*. Diantaranya dijelaskan pada tabel 3.1 dan 3.3.

penduduk desa Panunggalan. Pada tabel panunggalan tanggal lahir menggunakan tipe *varchar* dikarenakan data penduduk yang telah ada tidak memungkinkan disimpan di *database* dalam bentuk *date*. Struktur tabel panunggalan ditunjukkan pada tabel 3.1

**Tabel 3. 1** Struktur Tabel Panunggalan

No	Field Name	Key Type	Null	Data Type	Max. Length
1	NO	PK	NN	Int	10
2	NO_KK		NN	Bigint	25
3	NIK		NN	Bigint	25
4	NAMA LENGKAP		NN	Varchar	80
5	TEMPAT LAHIR		NN	Varchar	30
6	TANGGAL LAHIR		NN	Varchar	15
7	JENIS KELAMIN		NN	Varchar	5
8	NAMA_IBU		NN	Varchar	80
9	DUSUN		NN	Varchar	30
10	RT		NN	Int	5
11	RW		NN	Int	5

## 2. Tabel admin

Tabel admin merupakan tabel yang digunakan untuk menyimpan data *user* program aplikasi persuratan. Pada tabel admin terdapat kolom *idpegawai* sebagai *foreign key* yang bersumber dari *primary key* tabel kepeawaian. Struktur tabel admin ditunjukkan pada tabel 3.2

**Tabel 3. 2** Struktur Tabel Admin

No	Field Name	Key Type	Null	Data Type	Max. Length
1	Id	PK	NN	Int	2
2	Username		NN	Varchar	20
3	Password		NN	Varchar	20
4	Pertanyaan		N	Varchar	100
5	Jawaban		N	Varchar	100
6	idpegawai	FK	N	Int	3

pegawai dibalai desa Panunggalan. Pada tabel kepegawaian terdapat kolom idpenduduk sebaai *foreign key* yang bersumber dari *primary key* tabel panunggalan. Struktur tabel kepegawaian ditunjukkan pada tabel 3.3

**Tabel 3. 3** Struktur Tabel Kepegawaian

No	Field Name	Key Type	Null	Data Type	Max. Length
1	NO	PK	NN	Int	3
2	NAMA		NN	Varchar	100
3	TEMPATLAHIR		NN	Varchar	40
4	TANGGALLAHIR		NN	Varchar	10
5	JABATAN		NN	Varchar	20
6	JENISKELAMIN		NN	Varchar	10
7	NOMORSK		NN	Varchar	30
8	MASAKERJA		NN	Varchar	20
9	KETERANGAN		N	Varchar	100
10	idpenduduk	FK	NN	Int	25

#### 4. Tabel Bukti Diri

Tabel bukti diri adalah tabel yang berisi *database* tentang apa saja yang dapat dijadikan bukti dalam pembuatan surat. Tabel buktidiri akan dijelaskan dengan lebih detail pada tabel 3.4

**Tabel 3. 4** Tabel Bukti Diri

NO	Field Name	Key Type	Null	Data Type	Max. Length
1	Id	PK	NN	Int	2
2	buktidiri		NN	Varchar	15

#### 5. Tabel pekerjaan

Tabel pekerjaan adalah tabel yang berisi *database* tentang pekerjaan apa saja yang tersimpan di *database* untuk pembuatan surat. Tabel pekerjaan akan dijelaskan dengan lebih detail pada tabel 3.5



NO	Field Name	Key Type	Null	Data Type	Max. Length
1	id	PK	NN	Int	3
2	pekerjaan		NN	Varchar	20

## 6. Tabel Surat Keluar

Tabel suratkeluar adalah tabel yang berisi pengarsipan surat yang telah dikeluarkan atau dibuat oleh aplikasi. Tabel suratkeluar dijelaskan lebih detail pada tabel 3.6.

**Tabel 3. 6** Tabel Surat Keluar

NO	Field Name	Key Type	Null	Data Type	Max. Length
1	Id	PK	NN	Int	5
2	Kodesurat		NN	Varchar	5
3	Nomorsurat		NN	Int	7
4	Tanggalpembuatan		NN	Datetime	-
5	Idpenduduk		N	Int	5
6	Pemohon		NN	Varchar	35
7	Idpetugas		N	Int	3
8	Petugas		NN	Varchar	35

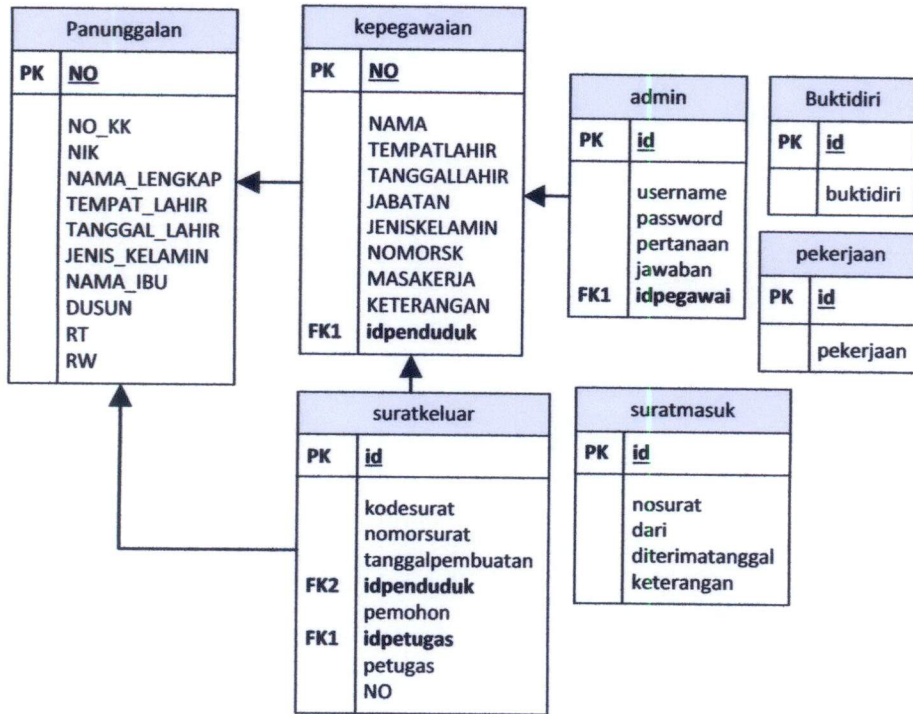
## 7. Tabel Surat Masuk

Tabel suratmasuk adalah tabel penyimpanan surat apa saja yang telah diterima kantor balai desa Panungglan. Tabel suratmasuk akan dijelaskan lebih detail pada tabel 3.7.

**Tabel 3. 7** Tabel Surat Masuk

NO	Field Name	Key Type	Null	Data Type	Max. Length
1	Id	PK	NN	Int	10
2	nosurat		NN	Varchar	20
3	dari		NN	Varchar	40
4	diterimatanggal		NN	Datetime	-
5	keterangan		N	Varchar	50

beberapa tabel yang tidak memiliki relasi. Relasi antar tabel akan dijelaskan lebih detail menggunakan gambar relasi tabel basis data pada gambar 3.18.



Gambar 3. 18 Relasi Antar Tabel

### 3.3.6 Perancangan Tampilan Antar Muka

Pada tahap ini dilakukan perancangan bentuk *interface* program aplikasi administrasi persuratan di balai desa Panunggalan, dengan tujuan supaya pemakai mudah mengerti (*user friendly*). Perancangan *interface* ini adalah gambaran secara umum mengenai tampilan aplikasi. Tampilan *form* selengkapnya terlampir. Beberapa perancangan tampilan *form* aplikasi yaitu:

#### 1. Tampilan *form* masuk

*Form* masuk adalah *form* dimana pengguna diharuskan mengimputkan nama pengguna dan kata sandi pengguna agar dapat masuk kedalam *form* utama aplikasi. Desain *interface form* masuk dapat dilihat seperti gambar 3.19.

Gambar	Label informasi pembuat
<p>Label: Selamat Datang  Label : Program Administrasi Persuratan Desa  Label interaksi</p> <p>Label nama pengguna    <input type="text" value="Textbox nama pengguna"/>    <input type="button" value="Button Data Penduduk"/></p> <p>Label kata sandi        <input type="text" value="Textbox kata sandi"/>        <input type="button" value="Button Profil Desa"/></p> <p>Link: Lupa kata sandi</p> <p><input type="button" value="Button masuk"/>        <input type="button" value="Button Keluar"/></p>	

**Gambar 3. 19** *Form Masuk*

## 2. Tampilan *Form* Utama

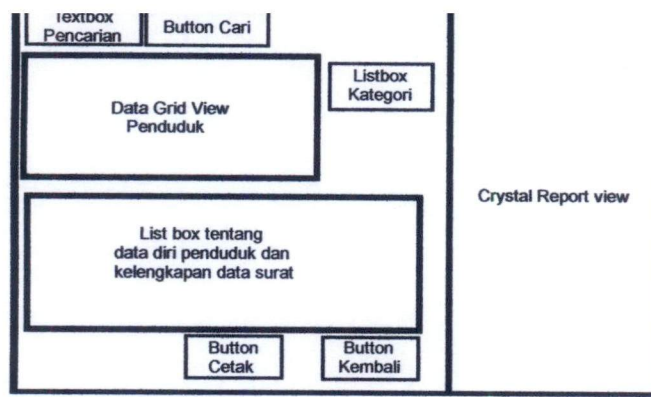
*Form* utama adalah *form* inti aplikasi. Desain *interface form* utama dapat dilihat seperti gambar 3.20.

Menu Bar		
Gambar		
<input type="text" value="Text box kata kunci pencarian"/>	<input type="list" value="List box kategori"/>	<input type="button" value="Button Cari"/>
Data Grid view Penduduk		

**Gambar 3. 20** *Form* Utama

## 3. Tampilan *Form* Membuat Surat

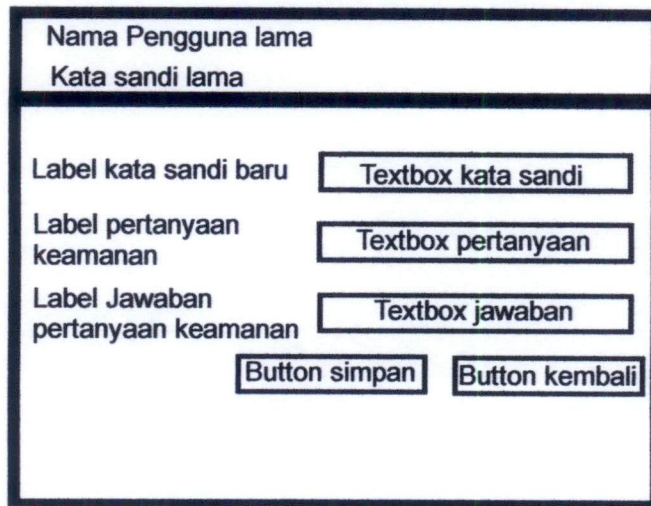
Desain *interface form* membuat surat dapat dilihat seperti gambar 3.21



**Gambar 3. 21** Desain *Form* Surat Kelahiran

#### 4. Tampilan *Form* Ubah Profil

Desain interface *form* ubah profil dapat dilihat seperti gambar 3.22.



**Gambar 3. 22** *Form* Ubah Profil

#### 5. Tampilan *Form* Tambah Penduduk

Desain interface *form* tambah penduduk dapat dilihat seperti gambar 3.23.



**Gambar 3. 23** *Form* Tambah Penduduk

#### 6. *Form* Ubah atau Hapus Penduduk

Desain interface *form* ubah atau hapus penduduk dapat dilihat seperti gambar 3.24.

**Gambar 3. 24** *Form* Ubah atau Hapus Penduduk

### 3.4 Metode Pengujian

Dalam pegujian aplikasi penulis menggunakan metode *blackbox*. Dimana aplikasi akan diuji fungsi pada setiap *form*. Penulis menggunakan metode *blackbox testing* untuk pengujian aplikasi dikarenakan dinilai memiliki beberapa ke unggulan yaitu :

- Bisa memilih subset test secara efektif dan efisien.
- Dapat menemukan cacat atau *error*.

*blackbox* akan menguji semua *form* yang berada dalam aplikasi. *Form – form* yang akan diuji adalah :

- *Form* Masuk
  - *Form* Lupa Sandi
  - *Form* Profil Desa
  - *Form* Data Penduduk
  - *Form* Utama
  - Surat Keterangan Kelahiran
  - *Form* Tambah Penduduk
  - *Form* Ubah atau Hapus Penduduk
  - Surat Keterangan Kematian
  
  - *Form* Edit Profil
  - *Form* Arsip Surat
- *Form* Permohonan KTP
  - Surat Keterangan Usaha
  - Surat Keterangan Tidak Mampu
  - Surat Keterangan Domisili
  - Surat Ijin Keramaian
  - *Form* Tambah Perangkat Desa
  - *Form* Ubah\ Hapus Perangkat Desa
  - *Form* Lihat Perangkat Desa
  - *Form* Daftar pekerjaan dan bukti diri
  
  - *Form* Admin
  - *Form* Tambah/Hapus Pengguna

Hasil pengujian aplikasi berupa tabel yang akan di jelaskan lebih detail pada bab IV. Sementara cara menyimpulkan hasil pengujian dengan mencoba setiap fungsi yang ada dalam aplikasi dan apabila semua fungsi aplikasi berfungsi dengan benar maka pengujian dikatakan berhasil.