

## **KATA PENGANTAR**

Puji syukur penulis panjatkan kehadiran Tuhan Yang Maha Esa karena dengan segala rahmat dan karunia-Nya penulis dapat menyelesaikan laporan Proyek III ini. Laporan ini









































disusun dan diajukan untuk memenuhi tugas mata kuliah Proyek III di semester V (lima) pada Program Studi DIV Teknik Informatika Politeknik Pos Indonesia.

Penulis menyadari bahwa dalam penyelesaian laporan Proyek III ini, bukanlah hasil kerja keras penulis sendiri. Tetapi, laporan ini terselesaikan berkat saran dan bimbingan dari berbagai pihak. Untuk itu penulis ingin menyampaikan rasa terima kasih kepada:

1. Tuhan Yang Maha Esa yang telah melimpahkan rahmat dan karunia-Nya.
2. Kedua orang tua dan keluarga penulis yang telah memberi motivasi kepada penulis.
3. Woro Isti Rahayu S.T., M.T. selaku dosen pembimbing.

- 4.
5. M. Yusril Helmi Setyawan, S.Kom., M.Kom. selaku Ketua Program Studi DIV Teknik Informatika.
6. Cahyo Prianto, S.Pd., M.T. selaku Koordinator Proyek III Tahun 2019/2020.
7. Teman-teman dan pihak lainnya yang telah membantu dalam proses pengerjaan laporan Proyek III ini.

Penulis menyadari akan adanya kekurangan dalam penyusunan laporan Proyek III ini. Penulis mengharapkan adanya kritik dan saran yang bersifat membangun sebagai acuan dalam perbaikan pada proyek selanjutnya.

Bandung, 14 Januari 2020

## LATAR BELAKANG

Perkembangan ilmu pengetahuan dan teknologi telah mengalami percepatan yang tinggi membuat sebuah perubahan yang sangat luar biasa bagi kehidupan manusia. Salah satu pemanfaatan teknologi dalam kehidupan manusia adalah dalam pendataan penduduk di suatu daerah. Masalah kependudukan menjadi masalah yang kompleks karena pendataan data penduduk masih dilakukan secara manual, data jumlah penduduk yang belum akurat, data pertumbuhan penduduk yang belum termonitoring, data kepadatan penduduk yang terus meningkat.

Indonesia adalah Negara dengan jumlah penduduk terbesar keempat setelah China, India dan Amerika Serikat. Tingkat pertumbuhan populasi Indonesia antara tahun 2000 dan 2010 adalah sekitar 1.49 persen per tahun. Pertumbuhan tertinggi terjadi di propinsi Papua (5.46 persen), sementara pertumbuhan populasi terendah terjadi di propinsi Jawa Tengah (0.37 persen) [1]. Pertumbuhan populasi diperkirakan sebesar sekitar 1.2 persen pada tahun 2015 berdasarkan data Bank Dunia. Menurut proyeksi yang dilakukan oleh Perserikatan Bangsa-Bangsa (PBB)



dengan menilik populasi absolut Indonesia di masa depan, maka negeri ini akan memiliki penduduk lebih dari 270 juta jiwa pada tahun 2025, lebih dari 285 juta jiwa pada tahun 2035 dan 290 juta jiwa pada tahun 2045. Baru setelah 2050 populasi Indonesia akan berkurang [2]. Melihat hal tersebut, dengan banyaknya jumlah penduduk di negara kepulauan seperti Indonesia maka diperlukan adanya suatu sistem aplikasi yang berguna untuk mengolah data penduduk di setiap bagian wilayah Indonesia.

Dengan adanya suatu sistem informasi secara terkomputerisasi dapat menghasilkan data penduduk yang berdomisili secara akurat dan terkini sehingga pendataan data penduduk yang dilakukan oleh petugas instansi pemerintah menjadi lebih baik. Sistem sensus penduduk pada saat ini masih dilakukan secara manual, yaitu dengan cara petugas instansi pemerintah melakukan input data menggunakan kertas dan pulpen. Selain itu sering terjadinya kesulitan dalam menyamakan/menyesuaikan kembali data penduduk di karenakan banyaknya penduduk yang memiliki KK(kartu keluarga) double sehingga tak jarang petugas kesulitan dalam merekap data penduduk, disamping itu juga kurangnya tingkat kesadaran penduduk dalam mengurus surat - surat keterangan pindah, melahirkan, atau pun meninggal dikarenakan dengan alasan prosesnya yang terlalu berbelit - belit,dari permasalahan tersebut munculah masalah seperti sering terjadinya pendataan yang tidak sesuai. Seringkali terjadi masalah seperti penduduk yang sudah meninggal tetapi masih mendapat surat panggilan pemilu. Masalah masalah tersebut dikarena belum tersedianya aplikasi khusus untuk data kependudukan dan untuk membuat surat keterangan melahirkan, meninggal ataupun pindah yang mempermudah petugas ataupun penduduk dalam mengurus data kependudukan.

Dengan pembuatan Aplikasi Sensus Penduduk, dapat memberikan banyak kemudahan dalam proses pengelolaan data penduduk. Penduduk atau user dapat menginputkan data tanpa harus menunggu petugas sensus mendatangi rumah-rumah penduduk hal ini akan sangat memudahkan pengguna maupun petugas dalam mengikuti pelaksanaan sensus penduduk yang berlaku setiap 10 tahun dimanapun dan kapanpun tanpa harus melalui tata cara rumit dan berbelit-belit. Diharapkan dengan adanya aplikasi tersebut dapat membantu instansi pemerintah dalam mengolah data kependudukan lahir, mati, pindah dan datang sehingga dapat mempercepat proses pembuatan dan penyampaian laporan tersebut ke kecamatan. Selain itu, aplikasi ini diharapkan dapat membantu instansi pemerintah dalam memonitoring perkembangan penduduknya.

Berdasarkan latar belakang diatas maka dapat di identifikasikan beberapa masalah yang menjadi persoalan diantaranya :

1. Proses pendataan penduduk yang masih dilakukan secara manual menggunakan kertas lalu diinputkan ke komputer membutuhkan waktu cukup lama membuat sering terjadinya perbedaan data antara data petugas dan kenyataan di lapangan.
2. Kesulitan dalam menyamakan/menyesuaikan kembali data penduduk di karenakan banyaknya penduduk serta minimnya petugas.

Tujuan pembuatan Aplikasi Sensus Penduduk sebagai berikut :

1. Menghasilkan proses input data secara terkomputerisasi sehingga menghasilkan data yang tepat dan cepat.
2. Petugas dapat mengolah data penduduk dengan mudah cepat dan tepat.

## **LANDASAN TEORI**

### **Sensus**

Sensus, kadangkala juga disebut cacah jiwa adalah sebuah proses mendapatkan informasi deskriptif tentang anggota sebuah populasi (tidak hanya populasi manusia). Sensus digunakan untuk demokrasi (pemilu), pengumpulan pajak, juga digunakan dalam ilmu ekonomi. Di Indonesia terdapat beberapa macam sensus yang dilaksanakan oleh Badan Pusat Statistik. di antaranya yang terbesar: Sensus Penduduk, Sensus Pertanian dan Sensus Ekonomi. Sensus Penduduk pada umumnya dilaksanakan pada tahun yang berakhiran "0" atau dalam jangka waktu sepuluh tahun[3].

## **2.2 Penduduk**

Penduduk adalah orang - orang yang berada di dalam suatu wilayah yang terikat oleh aturan - aturan yang berlaku dan saling berinteraksi satu sama lain secara terus menerus / kontinu. Dalam sosiologi, penduduk adalah sekumpulan manusia yang menempati wilayah geografi atau ruang tertentu[4].

Penduduk atau warga suatu negara atau daerah bisa didefinisikan menjadi dua :

1. Orang yang tinggal di daerah tersebut
2. Orang yang secara hukum berhak tinggal di daerah tersebut. Dengan kata lain orang yang mempunyai surat resmi untuk tinggal di situ. Misalkan bukti kewarganegaraan, tetapi memilih tinggal di daerah lain.

### **2.3     Android**

Menurut Hermawan (2011 : 1), Android merupakan OS (Operating System) Mobile yang tumbuh ditengah OS lainnya yang berkembang dewasa ini. OS lainnya seperti Windows Mobile, i-Phone OS, Symbian, dan masih banyak lagi. Akan tetapi, OS yang ada ini berjalan dengan memprioritaskan aplikasi inti yang dibangun sendiri tanpa melihat potensi yang cukup besar dari aplikasi pihak ketiga. Oleh karena itu, adanya keterbatasan dari aplikasi pihak ketiga untuk mendapatkan data asli ponsel, berkomunikasi antar proses serta keterbatasan distribusi aplikasi pihak ketiga untuk platform mereka[5].

## **2.4 JavaScript**

JavaScript adalah bahasa pemrograman berbasis prototipe yang berjalan disisi klien. Jika kita berbicara dalam konteks web, sederhananya, kita dapat memahami JavaScript sebagai bahasa pemrograman yang berjalan khusus untuk di browser atau halaman web agar halaman web menjadi lebih hidup. Kalau dilihat dari suku katanya terdiri dari dua suku kata, yaitu Java dan Script. Java adalah Bahasa pemrograman berorientasi objek, sedangkan Script adalah serangkaian instruksi program. Secara fungsional, JavaScript digunakan untuk menyediakan akses script pada objek yang dibenamkan. Contoh sederhana dari penggunaan JavaScript adalah membuka halaman pop up, fungsi validasi pada form sebelum data dikirimkan ke server, merubah image kursor ketika melewati objek tertentu, dan lain lain[6].



## **2.5 Web**

Website ( situs web ) adalah merupakan alamat ( URL ) yang berfungsi sebagai tempat penyimpanan data dan informasi dengan berdasarkan topik tertentu. Web Page ( halaman web ) merupakan halaman khusus dari situs web tertentu yang tersimpan dalam bentuk file. Dalam web page tersimpan berbagai informasi dan link yang menghubungkan suatu informasi ke informasi lain baik itu dalam page yang sama ataupun web page lain pada website yang berbeda[7].

## **2.6 Database**

Basis data merupakan kumpulan data dari berbagai sumber yang secara logika mempunyai arti implisit. Sehingga data yang terkumpul secara acak dan tanpa mempunyai arti, tidak dapat disebut basis data [8]. *MySQL* adalah sebuah perangkat lunak sistem manajemen basis data *SQL (DBMS)* yang *multi-user* yang di implementasikan dari sistem manajemen basis data relasional[9].

## **2.7 PHP**

PHP adalah singkatan dari Hypertext Preprocessor, PHP adalah sebuah aplikasi bahasa pemrograman untuk membuat sebuah web yang sifatnya adalah server - side scripting. PHP yaitu bahasa pemrograman yang banyak digunakan secara luas dan bisa digunakan bersamaan dengan HTML. Fungsi yang utama dari

PHP adalah kemampuannya sebagai server side programming atau scripting language dalam pembuatan website atau aplikasi yang berbasis website. Server side programming atau scripting language adalah bahasa yang berjalan pada sisi server ( dalam hal ini web server ) sehingga komputer ( client / web browser ) tidak mengetahui apa yang terjadi di dalam web server[10].

# **ANALISIS DAN PERANCANGAN**









































### **3.1 Analisis**

Analisis sistem merupakan penguraian dari suatu sistem informasi yang utuh kedalam bagian-bagian komponen dengan tujuan untuk mengidentifikasi dan mengevaluasi permasalahan, kesempatan, hambatan yang terjadi dan kebutuhan yang diharapkan sehingga nantinya dapat diusulkan perbaikannya. Pada bagian ini, akan dibahas tentang analisis prosedur dan aliran dokumen sistem yang sedang berjalan yang digambarkan dalam bentuk *flowmap*, pengkodean dan analisis sistem non-fungsional yang meliputi perangkat keras dan perangkat lunak yang digunakan.

















### **3.1.1 Analisis Sistem yang sedang Berjalan**

Analisis sistem yang sedang berjalan merupakan analisis tahap awal dalam penentuan dan perancangan sistem. Di dalam analisis ini terdapat dua metode yang digunakan, yaitu: analisis prosedur (*flowmap*) dan analisis dokumen yang digunakan. Dengan demikian, aplikasi yang akan dibangun akan benar-benar sesuai dengan prosedur serta sistem kerja yang dibutuhkan.







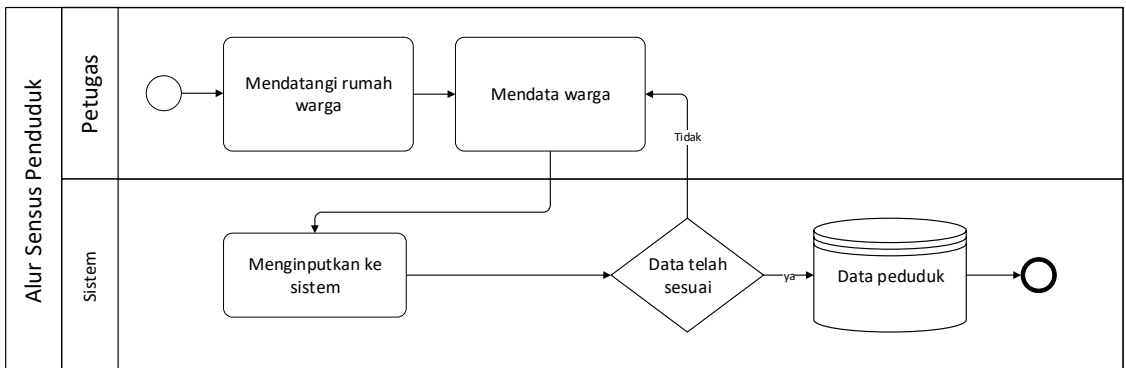
### **3.1.1.1 Analisis Prosedur/*Flow Map* Berjalan Pada Alur Sensus Penduduk**

Proses sistem pendataan benda-benda koleksi museum bisa digambarkan sebagai berikut:

1. Petugas sensus mendatangi rumah warga
2. Petugas sensus mendata masing-masing anggota keluarga
3. Petugas sensus menambahkan data diri yang telah didapat pada sistem
4. Petugas sensus mengolah data yang di dapat
5. Petugas sensus memastikan data pada sistem sesuai dengan data lapangan

Adapun *flowmap* analisis sistem yang sedang berjalan pada pendataan

benda-benda koleksi museum adalah sebagai berikut:



*Gambar 3.1 Flowmap Analisis Sistem Yang Sedang Berjalan Pada Sensus Penduduk.*

### 3.1.1.2 Analisis Dokumen yang Digunakan

Dari hasil analisis yang dilakukan, dokumen yang digunakan diantaranya adalah dokumen pendataan benda-benda koleksi museum dan daftar kunjungan museum. Adapun dokumen yang dimaksud adalah sebagai berikut:

**Tabel 3.1 Dokumen Pendataan Masyarakat**

<b>Dibuat oleh</b>	Petugas Sensus
<b>Dibuat untuk</b>	Petugas Sensus
<b>Isi</b>	Berisi data NIK, Nama, No KK, Tanggal Lahir, Status, Jenis Kelamin, Alamat, Agama, Pekerjaan, Jumlah anggota keluarga, dan status dalam keluarga
<b>Frekuensi</b>	Sesuai data yang di inputkan
<b>Tujuan</b>	Mendata penduduk yang tinggal pada suatu daerah

### 3.1.2 Analisis Sistem yang akan Dibangun

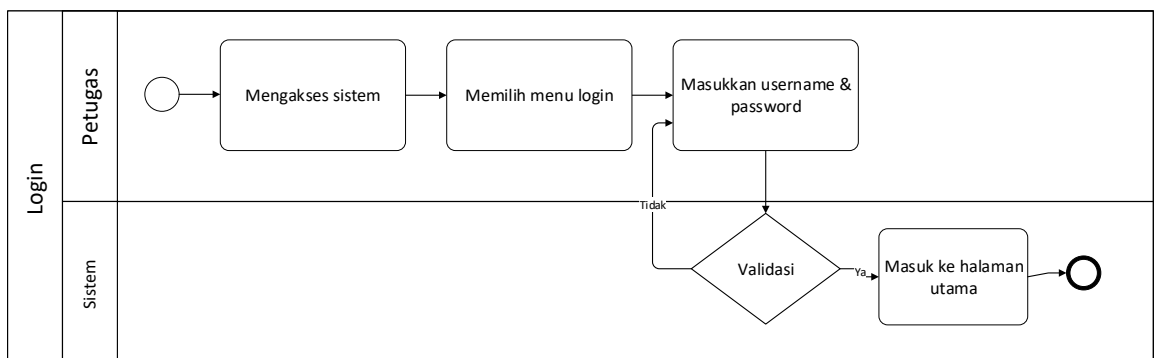
Analisis kebutuhan yang dimaksud disini berupa analisis *flowmap* mengenai sistem yang akan dibangun meliputi proses *login*, jumlah penduduk, kepadatan penduduk, persebaran penduduk, pertumbuhan penduduk,

komposisi penduduk, urbanisasi, dan *input* data penduduk. Adapun *Flowmap* yang akan dibangun dalah sebagai berikut:

### 3.1.2.1 Analisis Sistem Yang Akan Dibangun Pada *Login*

Pada proses *login* melibatkan petugas dan sistem. Adapun mekanisme kerja yang dibuat adalah sebagai berikut:

1. Petugas mengakses *form login*.
2. Petugas memasukkan *username* dan *password*.
3. Sistem akan mengecek *username* dan *password*.
4. Jika *username* dan *password* valid, maka akan masuk ke menu utama.  
Jika tidak masukkan *username* dan *password*.
5. Petugas dapat mengakses halaman utama.



Gambar 3.2 *Flowmap Analisis Sistem Yang Akan Dibangun Pada Login.*

### 3.1.2.2 Analisis Sistem Yang Akan Dibangun Pada Jumlah Penduduk

Pada proses jumlah penduduk melibatkan masyarakat, petugas (BPS) dan sistem. Adapun mekanisme kerja yang dibuat adalah sebagai berikut:

1. Masyarakat menginputkan data melalui aplikasi
2. Data tersebut akan masuk ke database sistem
3. Petugas dapat mengolah data yang telah didapat untuk menentukan jumlah penduduk pada suatu daerah













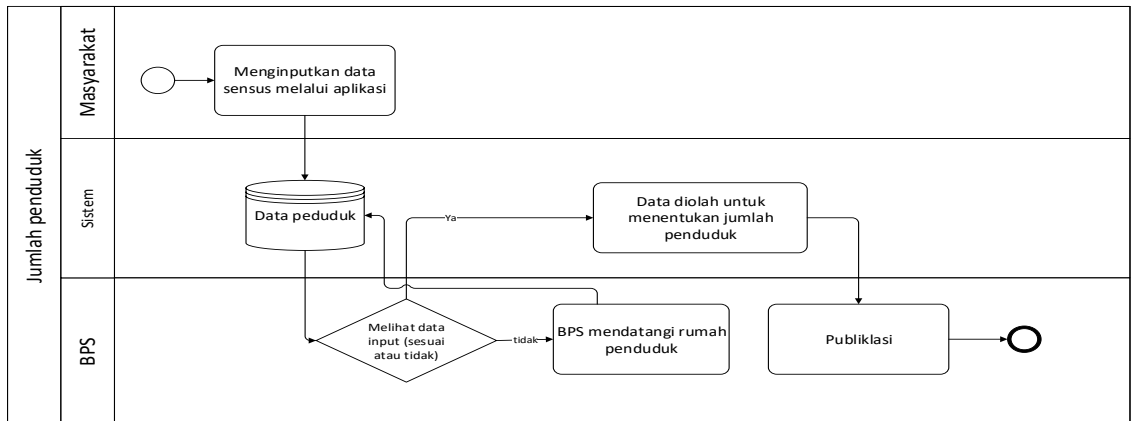








4. Petugas mendatangi rumah penduduk untuk memastikan data valid
5. Petugas mempublikasi data tersebut ke dalam website

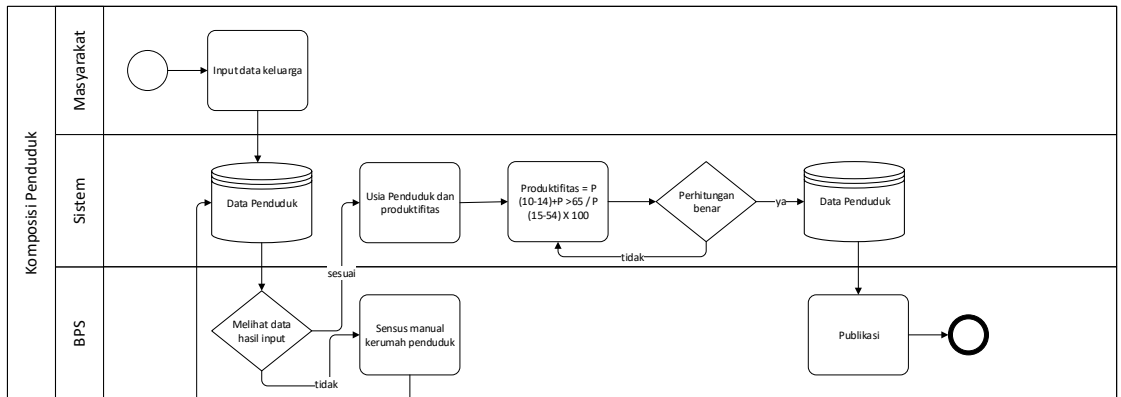


*Gambar 3.3 Flowmap Analisis Sistem Yang Akan Dibangun pada jumlah penduduk.*

### 3.1.2.3 Analisis Sistem Yang Akan Dibangun Pada Komposisi Penduduk

Pada proses komposisi penduduk melibatkan petugas, masyarakat dan sistem. Adapun mekanisme kerja yang dibuat adalah sebagai berikut:

1. Masyarakat akan menginputkan data melalui aplikasi
2. Data masuk ke database
3. Data diolah berdasarkan usia penduduk dan produktivitas
4. Jika data valid maka akan masuk ke dalam database
5. Jika tidak valid petugas akan mendatangi rumah pemilik data
6. Petugas mempublikasi data hasil perhitungan komposisi penduduk



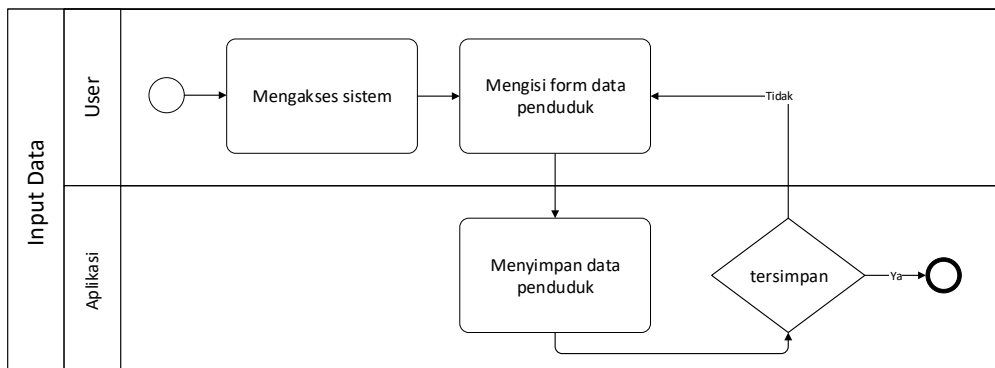
Keterangan:  
P (10-14) = Banyaknya penduduk yang tidak produktif di usia 10-14 tahun  
P > 65 = Banyaknya penduduk yang tidak produktif di usia lebih dari 65 tahun  
P (15-64) = Banyaknya penduduk yang produktif di usia 15-64 tahun

Gambar 3.4 *Flowmap Analisis Sistem Yang Akan Dibangun Pada Persebaran penduduk.*

### 3.1.2.4 Analisis Sistem Yang Akan Dibangun Pada *Input Data*

Pada proses komposisi penduduk melibatkan petugas, masyarakat dan sistem. Adapun mekanisme kerja yang dibuat adalah sebagai berikut:

1. Masyarakat akan membuka aplikasi
2. Menginputkan data pada form yang telah disediakan
3. Sistem akan menyimpan data yang dimasukkan oleh masyarakat
4. Data tersimpan



Gambar 3.5 *Flowmap Analisis Sistem Yang Akan Dibangun Pada Input Data.*



### 3.1.2.5 Kebutuhan Fungsional

Kebutuhan fungsional merupakan suatu kebutuhan yang berhubungan dengan kebutuhan sistem yang akan dibuat. Dimana menjabarkan mengenai fungsi-fungsi yang dapat mendukung jalannya sistem, adapun kebutuhan fungsional yang akan dibuat yaitu pengelolaan data terdiri dari 8 (Delapan) proses sesuai dengan urutan sebagai berikut:

1. *Login* Admin
2. Kelola data jumlah penduduk
3. Kelola data komposisi penduduk
4. Input data penduduk

Setiap proses memiliki representasi masing - masing pada sebuah tabel atau data yang terdapat pada *database* yang telah dirancang sebelumnya dan setiap proses berhubungan langsung dengan entitas atau *user*.

### 3.1.2.6 Kebutuhan Non-Fungsional

#### A. Kebutuhan Perangkat Lunak

Adapun spesifikasi perangkat lunak (*software*) yang dibutuhkan untuk mendukung aplikasi yang akan dibangun adalah sebagai berikut:

**Tabel 3.2 Kebutuhan Perangkat Lunak Web Administrator**

No.	Jenis Perangkat Lunak	Tools / Software
1.	Sistem Operasi	<i>Windows XP SP2, OS X 10.5.6, Ubuntu 10.04, Debian 6</i>
2.	<i>Web Server</i>	<i>PHP 5.4, Apache 2.4, Mysql 5.5, /Xampp 1.8.3</i>
3.	<i>Browser</i>	<i>Chrome, Opera, Firefox, Edge, Safari</i>

**Tabel 3.3 Kebutuhan Perangkat Lunak Aplikasi Mobile**

No.	Jenis Perangkat Lunak	Tools / Software
1.	Sistem Operasi	Android Jelly Bean 4.3

## B. Kebutuhan Perangkat Keras

Adapun spesifikasi perangkat keras (*hardware*) minimum yang dibutuhkan untuk mengimplementasikan aplikasi yang akan dibangun adalah sebagai berikut:

**Tabel 3.4 Kebutuhan Perangkat Keras Web Administrator**

No.	Jenis Perangkat Keras	Spesifikasi
1.	Harddisk	80 GB
2.	Memory	512 MB
3.	Processor	Intel Pentium 4 @ 1.8 Ghz
4.	VGA	128 MB
5.	Monitor	LCD 14,1 Inchi
6.	Mouse	Standard
7.	Keyboard	Standard
8.	Printer	Standard

**Tabel 3.5 Kebutuhan Perangkat Keras Aplikasi Mobile**

No.	Jenis Perangkat Keras	Spesifikasi
1.	Memory Eksternal	4 GB
2.	Memory Internal	1 GB
3.	Processor	Quad Core 1.2 Ghz

### **C. Analisis Pengguna**

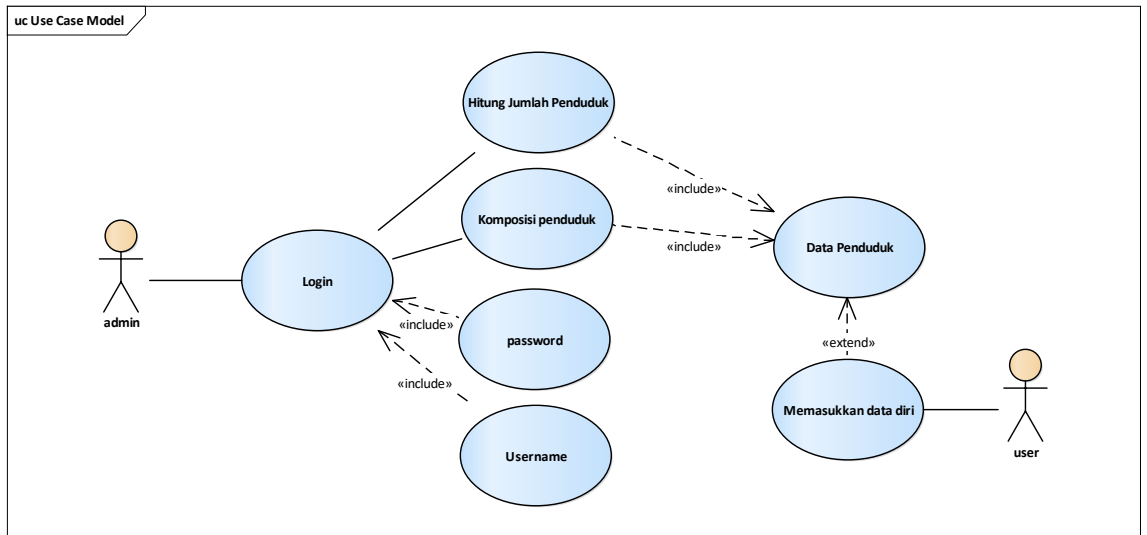
Analisis pengguna sistem dimaksudkan untuk mengetahui siapa saja aktor yang akan terlibat dalam sistem yang akan dibangun. Aplikasi pendataan penduduk ini menggunakan *platform mobile* dan *web* dimana pengguna sistem ini adalah sebagai berikut:

- a. Pengguna aplikasi pendataan penduduk yang menggunakan *platform android*. Dalam menggunakan sistem pengguna harus memiliki koneksi dengan sistem.
- b. Petugas merupakan aktor yang berperan dalam mengelola data di *server*.

## **3.2 Perancangan Sistem**

### **3.2.1 Use Case Diagram**

*Use case diagram* atau diagram *use case* merupakan pemodelan untuk kelakuan (*behavior*) suatu sistem yang akan dibuat. *Use case* diagram mendeskripsikan interaksi antara satu atau lebih aktor dengan sistem yang akan dibuat.



Gambar 3.6 *Use Case Diagram*

### 3.2.1.1 Definisi *Use Case*

### 3.2.1.2

#### 1. Definisi Aktor

**Tabel 3.6 Definisi Aktor**

No.	Aktor	Deskripsi
1.	Petugas	Aktor yang dapat melakukan proses input data penduduk pada platfrom android dan melakukan pengelolaan terhadap data penduduk.
2.	Masyarakat	Masyarakat adalah aktor yang dapat melakukan proses input data pada platform android.

#### 2. Definisi Use Case

**Tabel 3.7 Definisi Use Case**

No.	Menu Use Case	Submenu	Deskripsi
UC1	<i>Login</i>	<i>Login</i> Petugas	Merupakan proses masuk petugas untuk mengelola data pada sistem dengan memasukan <i>username</i> dan <i>password</i> .
UC2	Validasi		Merupakan proses pengecekan hak akses kepada pengguna yang berhak mengakses sistem
UC3	<i>Logout</i>	<i>Logout</i> Petugas	Merupakan proses untuk keluar dari sistem.

UC4	Halaman Hitung Jumlah Penduduk	Hitung Data	Merupakan proses menjumlahkan data penduduk keseluruhan untuk mendapatkan jumlah penduduk dari sebuah daerah.
UC5	Halaman Komposisi Penduduk	Lihat Data	Merupakan proses melihat/mengambil data komposisi penduduk pada sistem
UC6	Input Data Penduduk	Input Data	Merupakan proses dari <i>user</i> (warga/masyarakat) menginputkan data diri pada aplikasi mobile
		Lihat Data	Merupakan proses dari <i>user</i> (warga/masyarakat) melihat data yang telah mereka inputkan.



### 3.2.1.3 Skenario Use Case

Skenario *use case* mendeskripsikan urutan langkah - langkah dalam sistem yang berjalan di museum, baik yang dilakukan aktor terhadap sistem maupun yang dilakukan oleh sistem terhadap aktor. Berikut ini penjelasan dari masing -masing skenario tersebut:

#### 1. Skenario Use Case Login

**Tabel 3.8 Skenario Use Case Login**

Identifikasi	
Nomor	UC1
Nama	<i>Login</i>
Tujuan	Masuk ke dalam sistem
Deskripsi	
Aktor	Petugas
Skenario Utama	
Kondisi awal	From <i>login</i> di tampilkan
Aksi Aktor	Reaksi sistem
1. Memasukan <i>Username</i> dan <i>Password</i>	2. Mencocokan data <i>login</i> dengan data yang tersimpan di <i>database</i>
	3. Bila valid akan menampilkan halaman sesuai dengan data yang dimasukkan
Skenario Alternatif (jika gagal)	
Aksi actor	Reaksi Sistem
	1. Menampilkan pesan
2. Memasukan <i>Username</i> dan <i>Password</i>	
Kondisi akhir	Petugas dapat melakukan kegiatan pada sistem sesuai kewenangan sebagai administrator



## 2. Skenario *Use Case* Validasi

**Tabel 3.9** *Use Case* Validasi

Identifikasi	
Nomor	UC2
Nama	Validasi
Tujuan	Memastikan data yang dimasukkan telah benar, agar data yang kurang atau salah dapat diperbaiki sebelum di simpan.
Deskripsi	
Aktor	Petugas
Skenario Utama	
Kondisi Awal	Masuk <i>form login</i> pada sistem yang kemudian akan masuk ke halaman system
Aksi Aktor	Reaksi Sistem
1. Masukkan <i>Username</i> dan Password	2. Sistem akan menerima masukkan data
3. Tekan tombol <i>login</i> atau tombol simpan	4. Sistem akan memvalidasi data yang baru dimasukkan. Apabila data benar maka sistem akan memunculkan halaman selanjutnya. Jika salah atau kurang lengkap, maka sistem akan meminta mengisi kembali <i>username</i> dan <i>password</i> .
Kondisi Akhir	Jika data valid akan menampilkan halaman pengelolaan data sesuai dengan kebutuhan actor

### 3. Skenario *Use Case Logout*

**Tabel 3.10 Skenario *Use Case Logout***

Identifikasi	
Nomor	UC3
Nama	<i>Logout</i>
Tujuan	Mengeluarkan akun pengguna ketika pengguna telah selesai menggunakan system
Deskripsi	
Aktor	Petugas
Skenario Utama	
Kondisi Awal	Menekan tombol <i>logout</i>
Aksi Aktor	Reaksi Sistem
1. <i>User</i> menekan tombol <i>logout</i>	2. Sistem akan mengeluarkan akun pengguna
Kondisi Akhir	<i>User</i> akan kembali kehalaman <i>login</i>

### 4. Skenario *Use Case* Hitung Jumlah Penduduk

**Tabel 3.11 Skenario *Use Case* Hitung Jumlah Penduduk**

Identifikasi	
Nomor	UC4
Nama	Kelola Data Jumlah Penduduk

Tujuan	Untuk mengelola data penduduk dan menjumlahkan data yang didapatkan petugas
<b>Deskripsi</b>	
Aktor	Petugas
<b>Skenario Utama</b>	
Kondisi awal	<i>Login</i> terlebih dahulu lalu tampil halaman utama
Aksi actor	Reaksi system
1. Aktor memilih menu jumlah penduduk	2. Sistem akan menghitung jumlah dari data penduduk
	3. Menampilkan data jumlah penduduk
Kondisi akhir	Data jumlah penduduk terlihat

5.

**6.**

## 7. Skenario *Use Case* Komposisi Penduduk

**Tabel 3.12 Skenario *Use Case* Komposisi Penduduk**

Identifikasi	
Nomor	UC5
Nama	Kelola Data Komposisi Penduduk
Tujuan	Untuk mengelola data komposisi penduduk
Deskripsi	
Aktor	Petugas
Skenario Utama	
Kondisi awal	Login terlebih dahulu untuk menampilkan halaman utama
Aksi actor	Reaksi system
1. Aktor memilih menu komposisi penduduk	2. Menampilkan data penduduk
3. Memilih untuk menambah, mengubah, melihat atau menghapus data yang ada	4. Akan melakukan perintah yang diberikan oleh petugas
5. Menekan tombol simpan	6. Akan mengubah data didalam <i>database</i> yang telah tersimpan
7. Menekan tombol hapus dan muncul pesan lalu tekan iya	8. Akan menghapus data yang dimiliki oleh sistem
Kondisi akhir	Data terkelola

## 8.

**9.**

## 10. Skenario *Use Case Input Data*

**Tabel 3.13 Skenario *Use Case* Memasukkan Kritik Dan Saran**

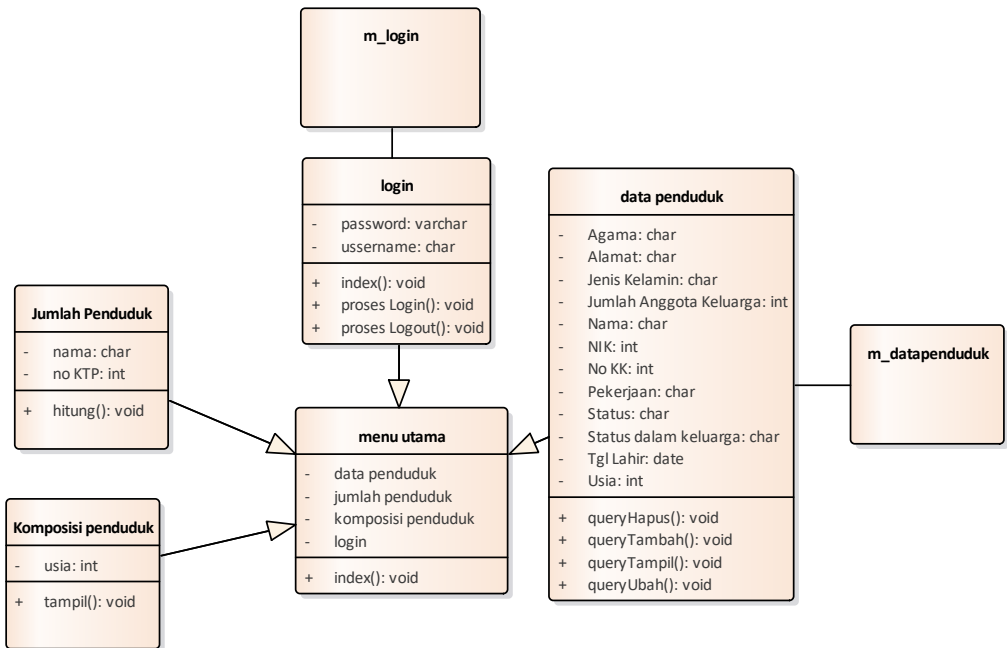
Identifikasi	
Nomor	UC6
Nama	Input Data
Tujuan	Memberikan data diri untuk kebutuhan sensus
Deskripsi	
Aktor	User
Skenario Utama	
Kondisi awal	Mengakses aplikasi sensus penduduk
Aksi actor	Reaksi system
1. User mengisikan <i>form</i> data diri	2. Menampilkan dan memberikan <i>form</i> data diri
3. Mengisi <i>form</i> data diri	4. <i>Form</i> akan disimpan ke <i>database</i>
Kondisi akhir	Data diri masuk ke <i>database</i>

### 3.2.2 *Class Diagram*

*Class diagram* atau kelas diagram merupakan diagram yang menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun suatu sistem Suatu kelas memiliki atribut dan metode atau operasi .

#### 3.2.2.1 *Class Diagram Admin*

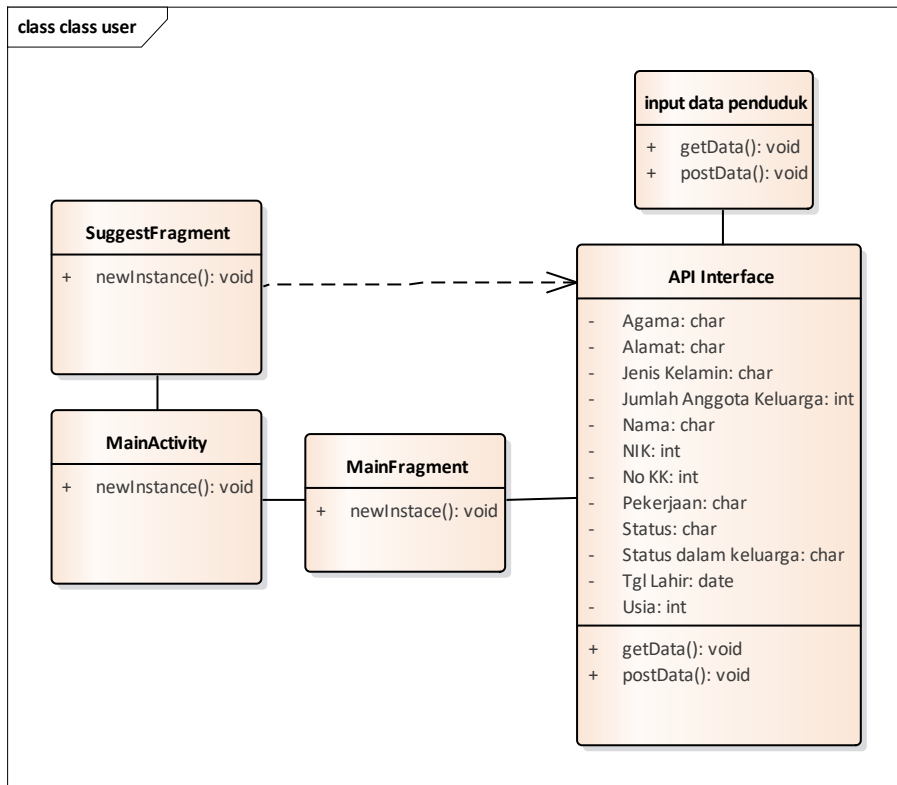
class class admin baru



Gambar 3.7 Class Diagram Admin



### 3.2.2.2 Class Diagram User



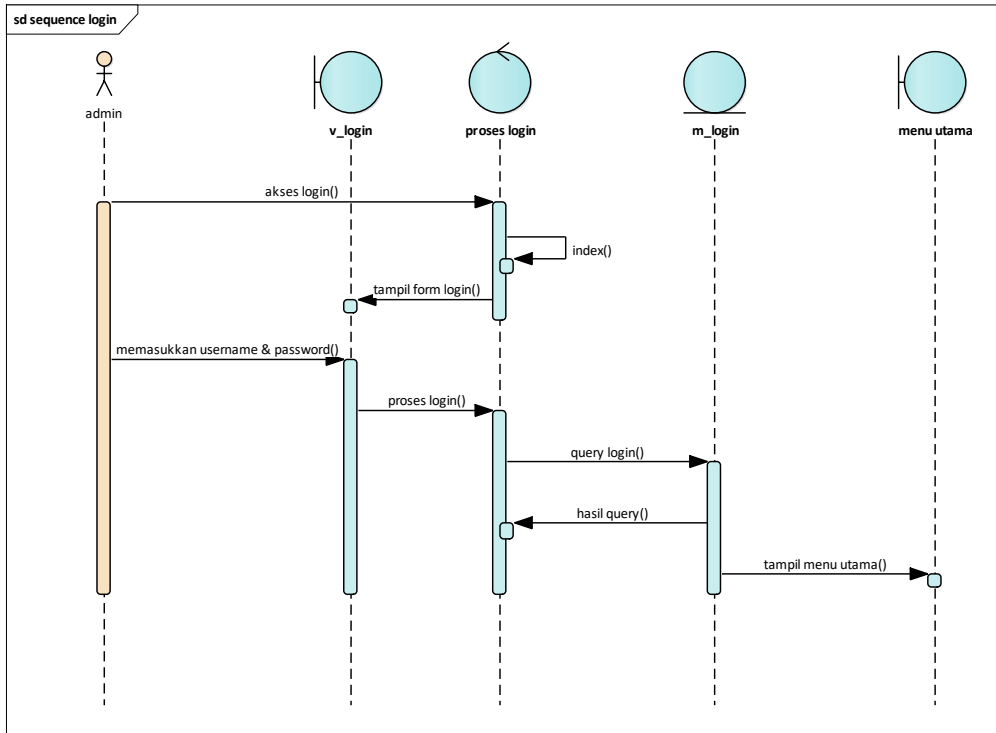
Gambar 3.8 Class Diagram User

### 3.2.3 Sequence Diagram

*Sequence diagram* atau diagram sekuen merupakan diagram yang menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan maupun yang diterima antar objek. Banyaknya diagram sekuen yang harus digambar sebanyak *usecase* yang memiliki proses sendiri atau yang penting semua *usecase* yang telah didefinisikan interaksi jalannya pesan sudah dicakup pada diagram sekuen sehingga semakin banyak *usecase* yang ada maka diagram sekuen yang dibuat semakin banyak.

### 3.2.3.1 Sequence Diagram Login

Berikut merupakan *sequence diagram login* petugas yang menjelaskan hubungan antara petugas dan *form login*.

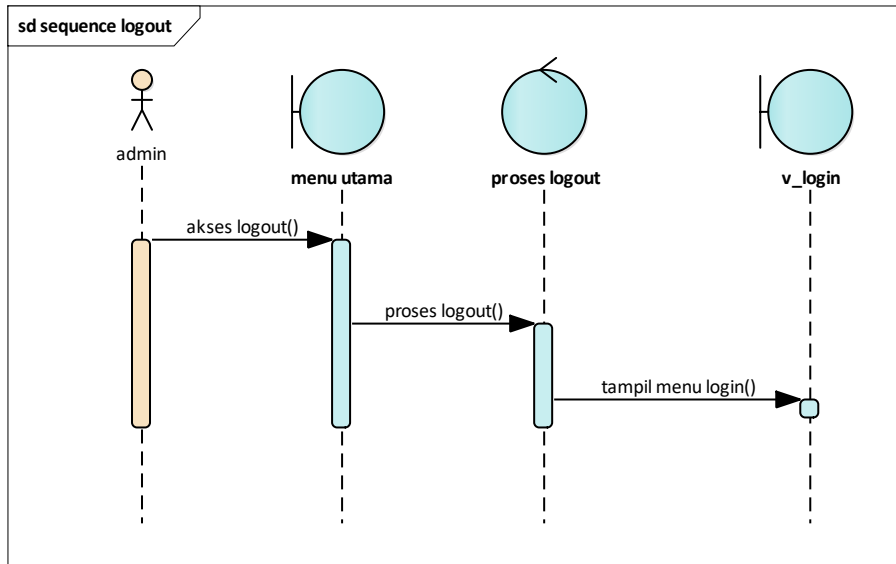


Gambar 3.9 Sequence Diagram Login

Pada *sequence diagram* ini menjelaskan proses *login* yang dilakukan oleh petugas dalam proses *login*. Aktor akan masuk kedalam sistem dengan mengakses *form login* dan memasukkan *username* dan *password*. Setelah penginputan *username* dan *password* akan mengalami pengecekan sebelum masuk kedalam sistem/menu utama.

### 3.2.3.2 Sequence Diagram Logout

Berikut merupakan *sequence diagram logout* manajer dan petugas yang menjelaskan hubungan antara manajer, petugas, menu utama, dan *form login*.

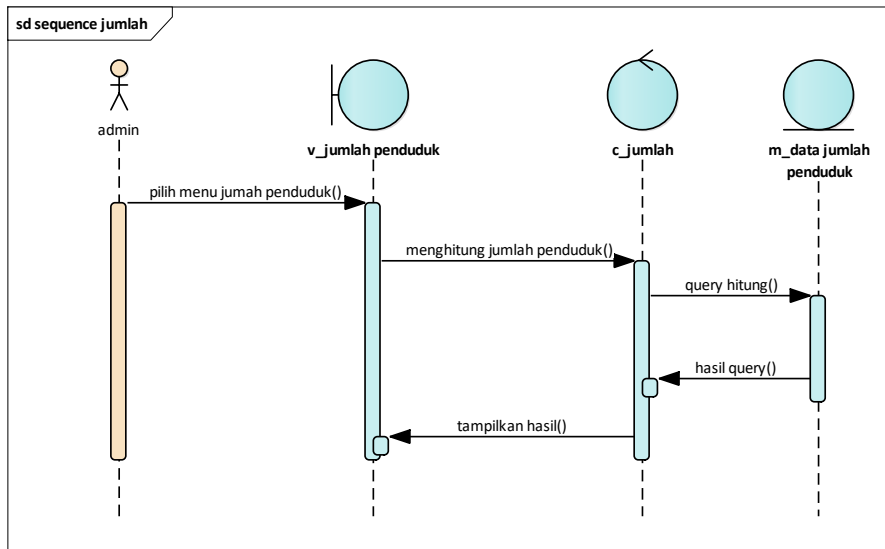


Gambar 3.10 *Sequence Diagram Logout*

Pada *sequence diagram logout* menjelaskan hubungan antara petugas dalam melakukan proses *logout*. Aktor akan memilih menu *logout* pada layar dan melakukan proses *logout* atau keluar dari sistem.

### 3.2.3.3 *Sequence Diagram Jumlah Penduduk*

Berikut merupakan *sequence diagram* mengelola data koleksi yang menjelaskan hubungan antara aktor manajer, petugas, dan *v\_jumlah penduduk*.



Gambar 3.11 *Sequence Diagram Jumlah Penduduk*

Pada *sequence diagram* mengelola data jumlah penduduk menjelaskan hubungan petugas dan *form* jumlah penduduk. Petugas akan masuk ke menu utama dan memilih menu jumlah penduduk untuk mengolah data penduduk yang didapatkan.





























































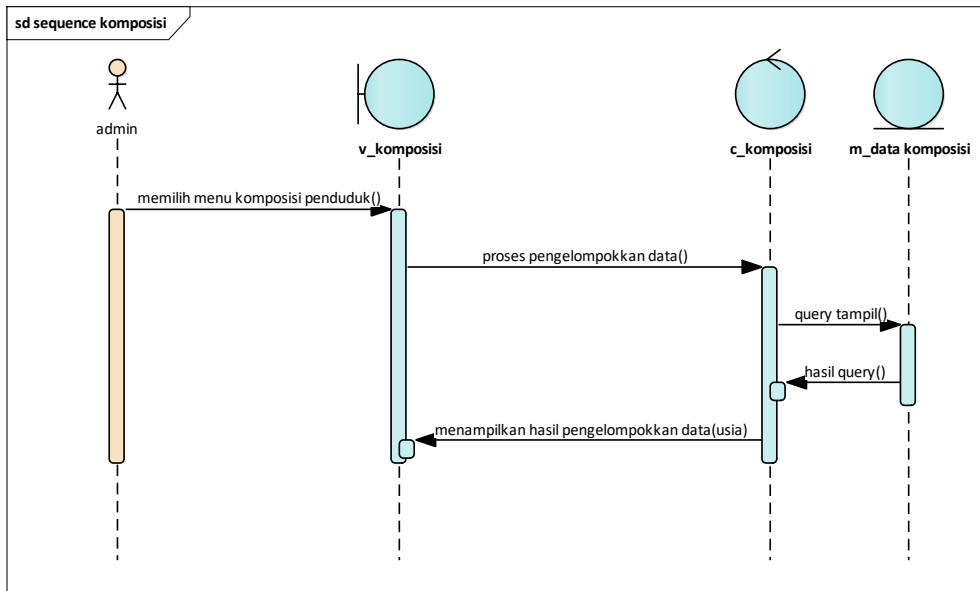






### ***Sequence Diagram Komposisi Penduduk***

Berikut merupakan *sequence diagram* mengelola data komposisi penduduk yang menjelaskan hubungan antara petugas, dan halaman komposisi penduduk.

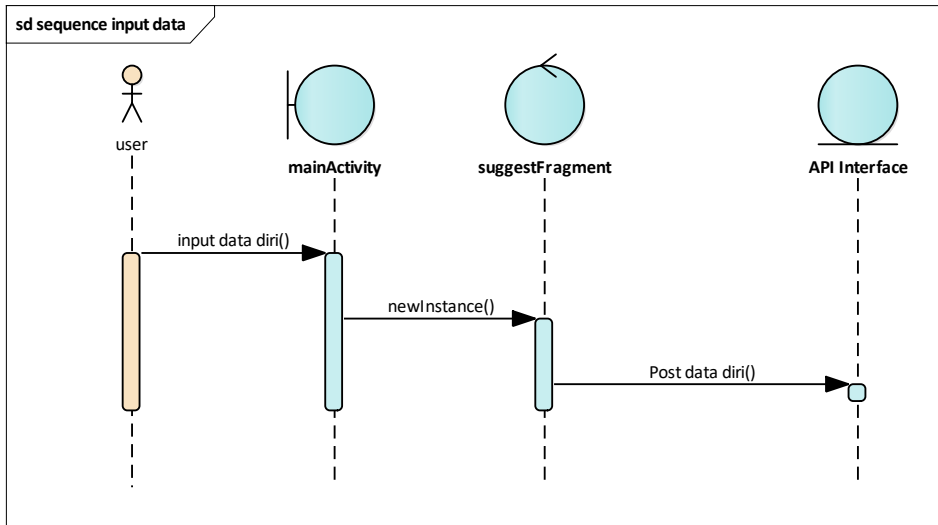


Gambar 3.12 *Sequence Diagram Komposisi Penduduk*

Pada *sequence diagram* mengelola data komposisi penduduk menjelaskan hubungan petugas dan *form* komposisi penduduk. Petugas akan masuk ke menu utama dan memilih menu komposisi penduduk untuk mengolah data penduduk yang didapatkan.

#### ***3.2.3.4 Sequence Diagram Input Data***

Berikut merupakan *sequence diagram input data* yang menjelaskan hubungan antara user, *main activity*, *main fragment*, *input data activity*, dan *api interface*.



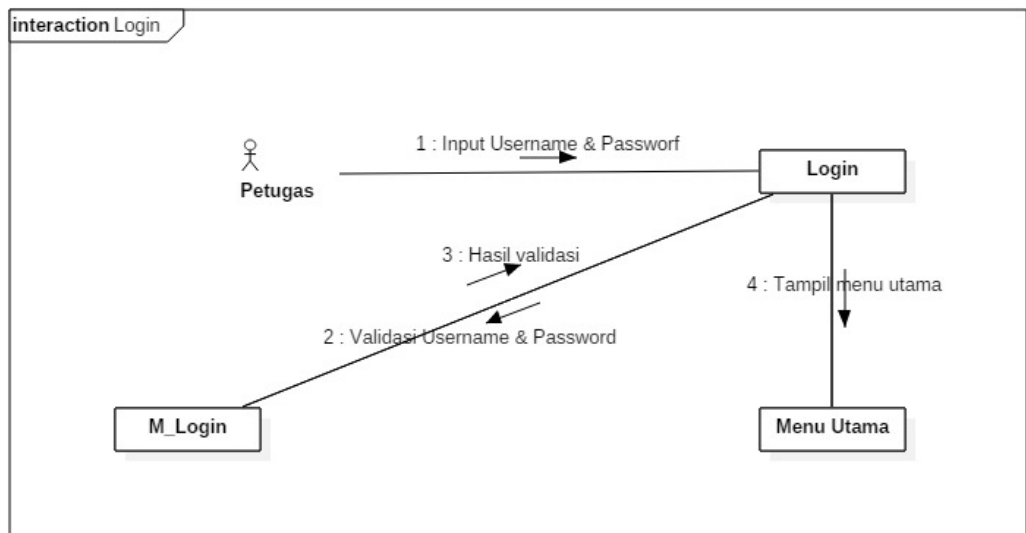
Gambar 3.13 *Sequence Diagram Input data*

Dalam *sequence diagram input data* menjelaskan dimana aktor memilih menu utama pada *main activity* dan memilih masukkan data yang akan disimpan dalam database dan dikelola oleh petugas.

### 3.2.4 *Collaboration Diagram*

*Collaboration diagram* atau diagram kolaborasi merupakan diagram yang menggambarkan interaksi antar objek atau bagian dalam bentuk urutan pengiriman pesan. Diagram kolaborasi mengelompokkan pesan pada kumpulan diagram sekuen menjadi sebuah diagram.

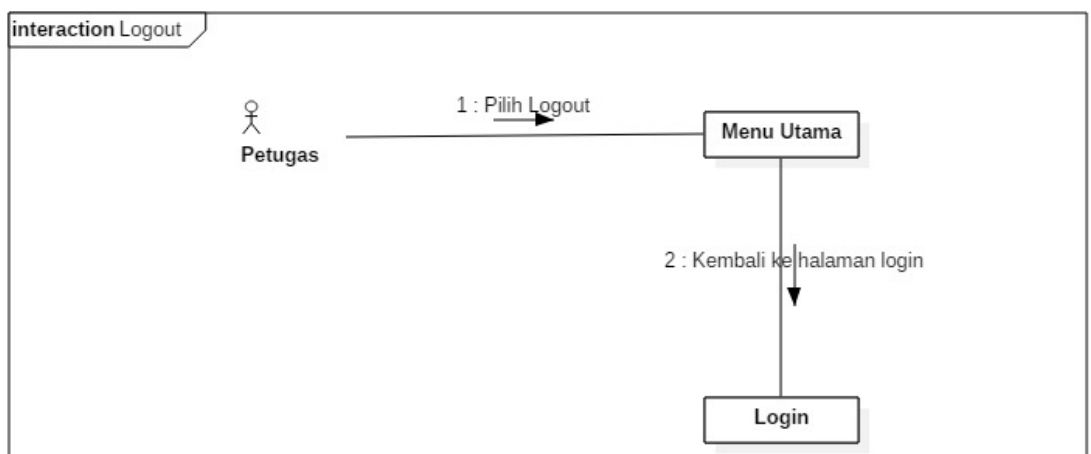
### 3.2.4.1 Collaboration Diagram Login



Gambar 3.14 Collabortaion Diagram Login

Pada *collaboration diagram login* menjelaskan mengenai proses *login*. Aktor akan mengakses *form login* dan masuk ke dalam menu utama dengan memasukkan *username* dan *password* pada *interface login* sistem akan mengecek data yang dimasukkan oleh aktor.

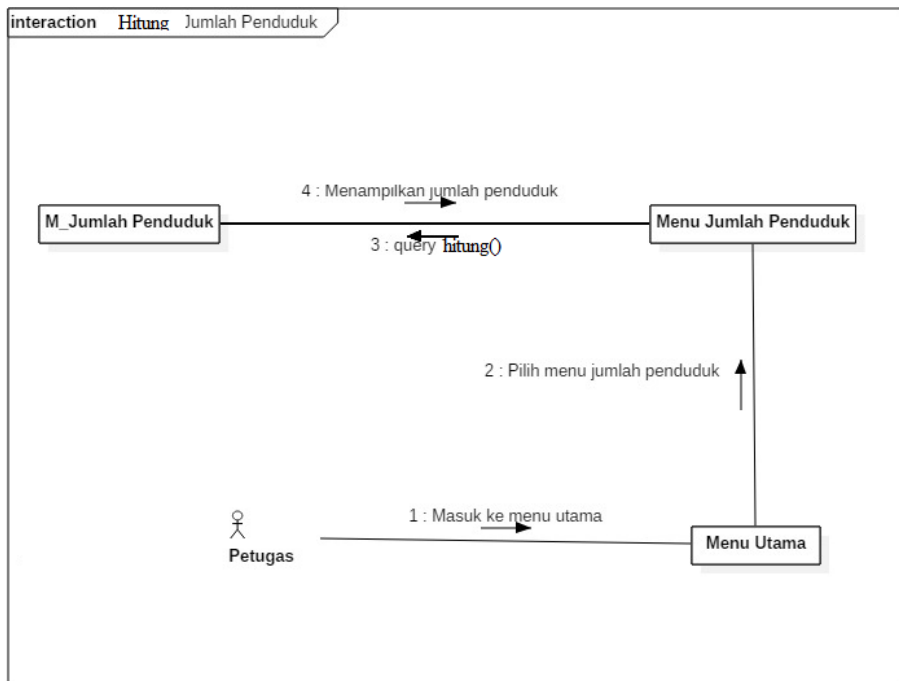
### 3.2.4.2 Collaboration Diagram Logout



Gambar 3.15 Collaboration Diagram Logout

Pada *collaboration diagram logout* menjelaskan mengenai proses *logout* atau keluar dari sistem. Aktor akan mengakses menu *logout* dalam menu utama sistem dan kemudian akan keluar dari sistem.

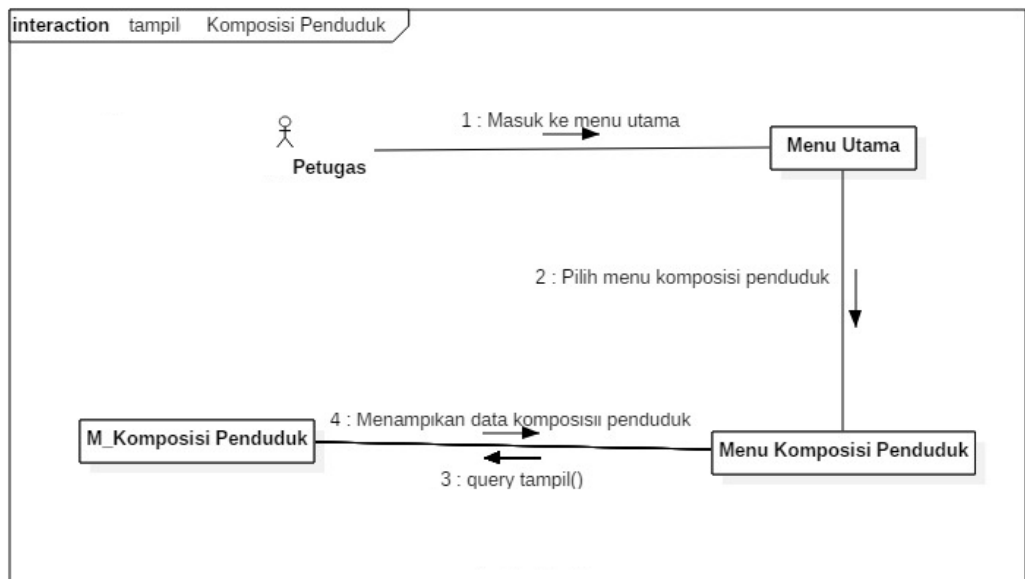
### 3.2.4.3 Collaboration Diagram Data Jumlah Penduduk



Gambar 3.16 Collaboration Diagram Jumlah Penduduk

Pada *collaboration diagram* mengelola data jumlah penduduk menjelaskan pengelolaan data jumlah penduduk yang ada di daerah tersebut. Aktor akan memilih menu jumlah penduduk pada menu utama sistem dan mengakses *database* jumlah penduduk yang memungkinkan aktor untuk melihat data keseluruhan dari jumlah penduduk di suatu daerah.

#### 3.2.4.4 Collaboration Diagram Mengelola Data Komposisi Penduduk



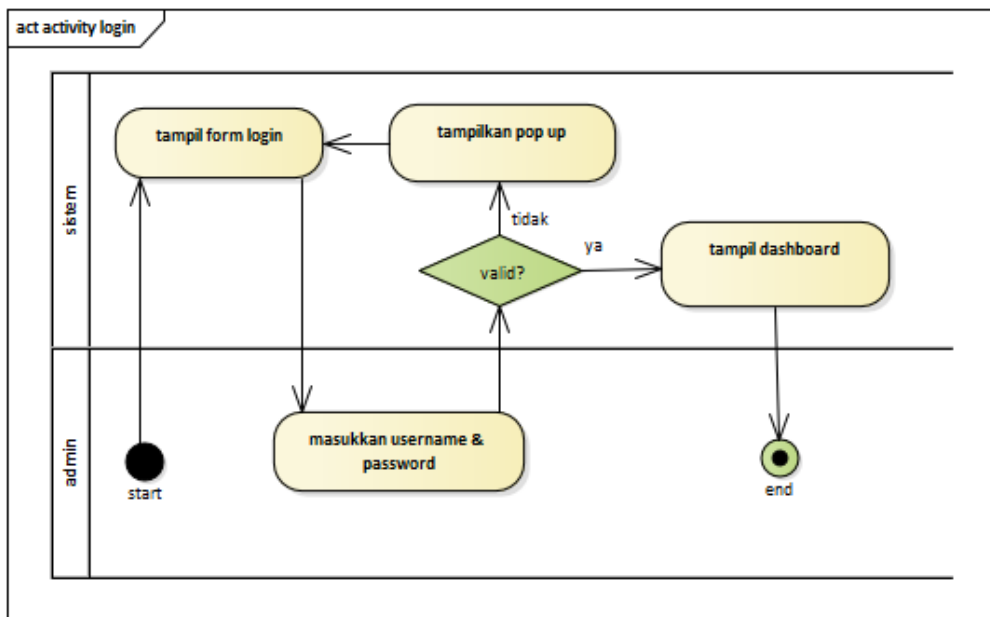
Gambar 3.17 Collaboration Diagram Mengelola Data Komposisi Penduduk

Pada *collaboration diagram* mengelola data komposisi penduduk menjelaskan pengelolaan data komposisi penduduk yang ada di daerah tersebut. Aktor akan memilih menu komposisi penduduk pada menu utama sistem dan mengakses *database* komposisi penduduk yang memungkinkan aktor melihat data komposisi penduduk yang telah dikelompokkan berdasarkan usia penduduk.

#### 3.2.5 Activity Diagram

*Activity* diagram atau diagram aktivitas merupakan diagram yang menggambarkan aliran kerja atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang terdapat pada perangkat lunak. Diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan oleh aktor.

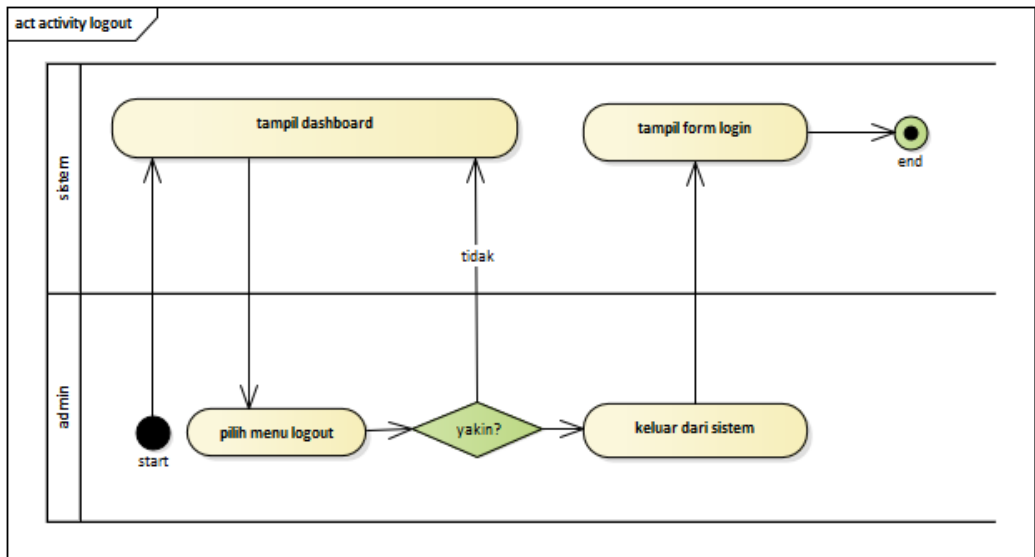
### 3.2.5.1 Activity Diagram Login



Gambar 3.18 Activity Diagram Login

Aktor (admin, petugas) akan masuk ke dalam sistem dengan mengakses *form login* dan memasukkan *username dan password* apabila data yang dimasukkan valid maka aktor akan masuk ke dalam sistem, namun jika data yang dimasukkan tidak valid maka aktor harus memasukkan kembali data yang valid.

### 3.2.5.2 Activity Diagram Logout

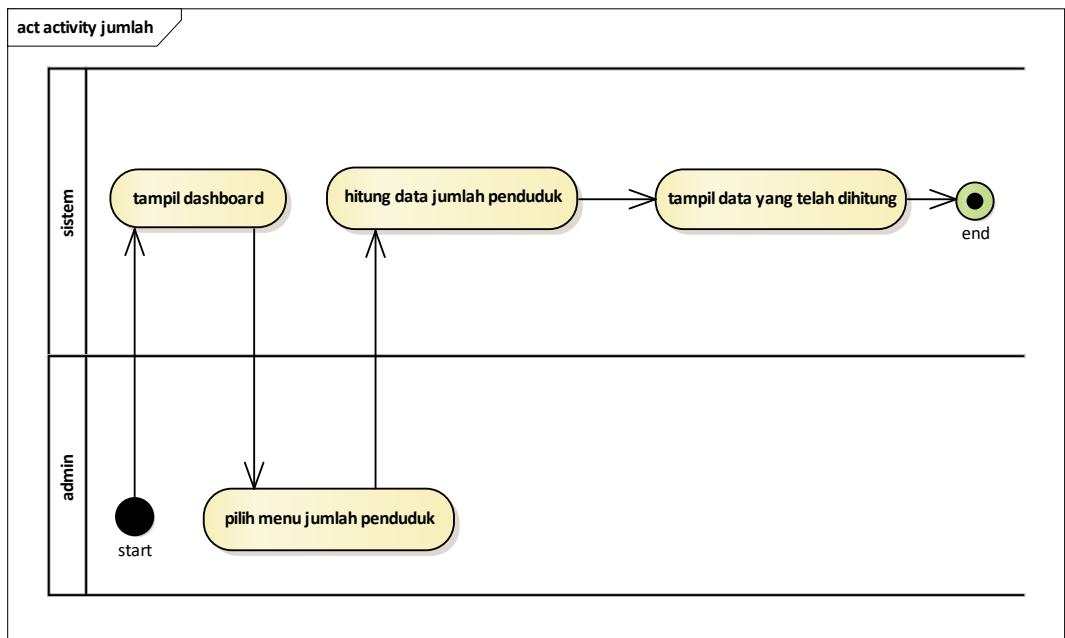


Gambar 3.19 Activity Diagram Logout

Aktor akan masuk ke dalam sistem dan mengakses menu *logout* pada menu utama apabila aktor ingin keluar maka ia akan keluar dari sistem dan menuju tampilan *form login*, jika tidak maka aktor akan tetap berada di dalam sistem.



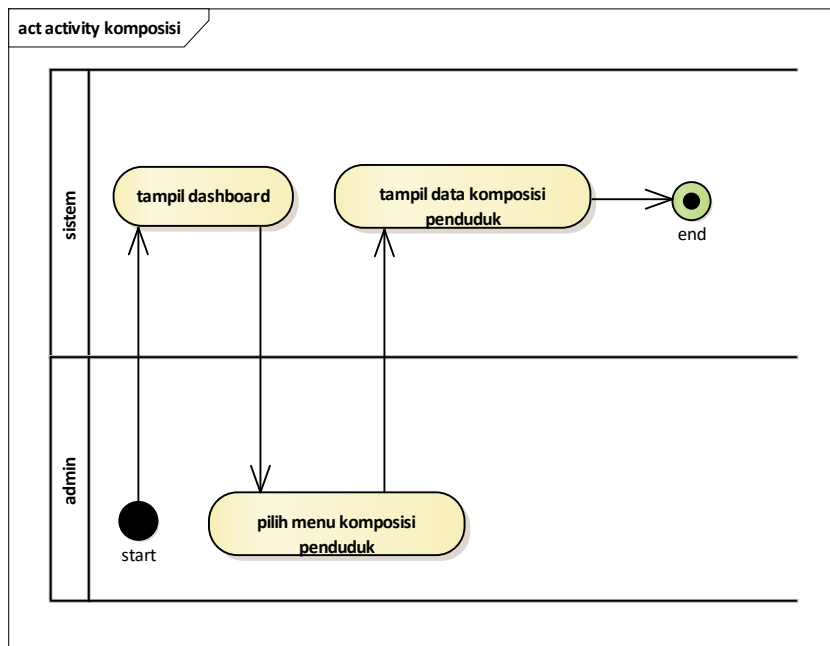
### 3.2.5.3 Activity Diagram Hitung Jumlah Penduduk



Gambar 3.20 Activity Diagram Mengelola Data Jumlah Penduduk

Aktor akan memilih menu jumlah penduduk pada sistem dan sistem akan menampilkan data jumlah penduduk yang diinginkan.

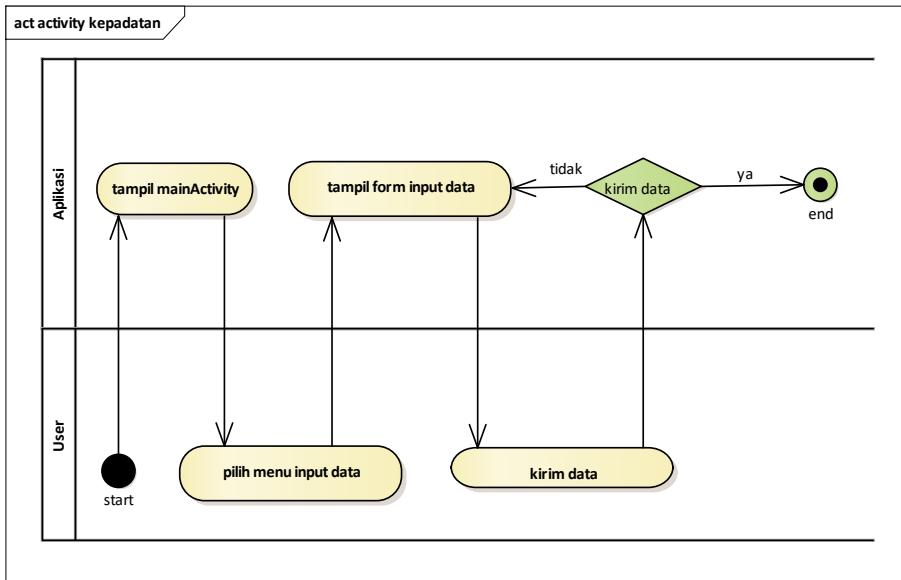
### 3.2.5.4 Activity Diagram Mengelola Data Komposisi Penduduk



Gambar 3.21 Activity Diagram Mengelola Data Komposisi Penduduk

Aktor akan memilih menu komposisi penduduk pada sistem dan sistem akan menampilkan data komposisi penduduk yang diinginkan.

### 3.2.5.5 Activity Diagram Input Data Penduduk



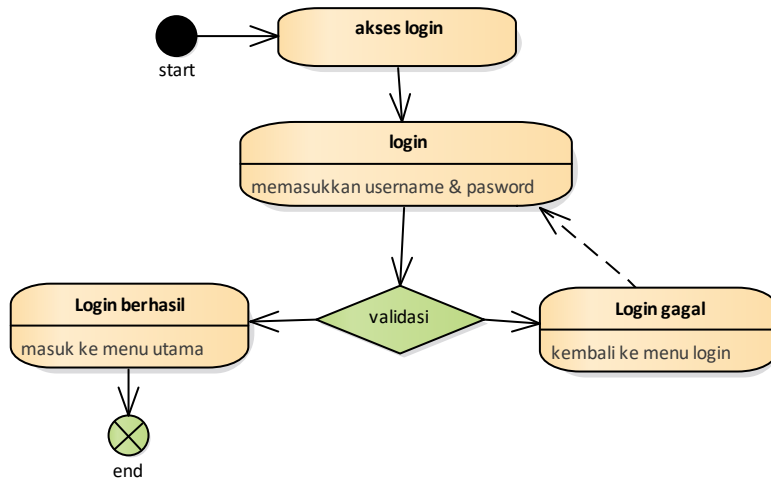
Gambar 3.22 Activity Diagram Input Data Penduduk.

Aktor akan memilih menu input data pada aplikasi kemudian mengisikan data yang diminta. Setelah itu aktor akan menekan tombol kirim dan data akan masuk ke dalam database.

### 3.2.6 Statechart Diagram

*Statechart diagram* atau diagram *statechart* merupakan diagram yang digunakan untuk menggambarkan perubahan status atau transisi status dari sebuah sistem atau objek [31]. Perubahan tersebut digambarkan dalam suatu graf berarah. Adapun *state chart diagram* pada aplikasi pemandu museum ini adalah sebagai berikut:

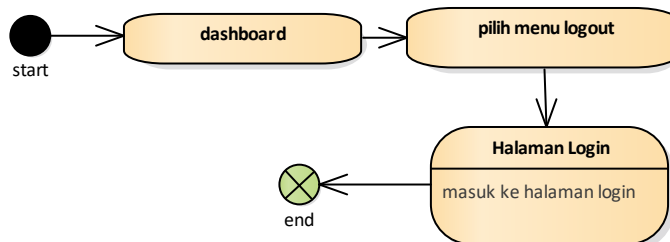
### 3.2.6.1 Statechart Diagram Login



Gambar 3.23 Statechart Diagram Login

Pada menu *login* aktor akan menjalankan aplikasi dan mengakses *form login* apabila data valid maka *login* berhasil, jika tidak *login* gagal.

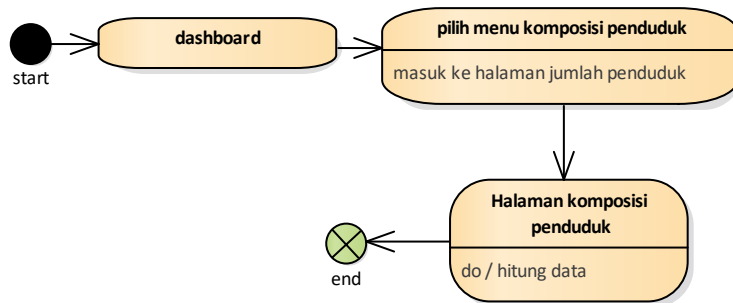
### 3.2.6.2 Statechart Diagram Logout



Gambar 3.24 Statechart Diagram Logout

Pada menu *logout* aktor akan memilih menu *logout* dan kemudian mengakses *form login*.

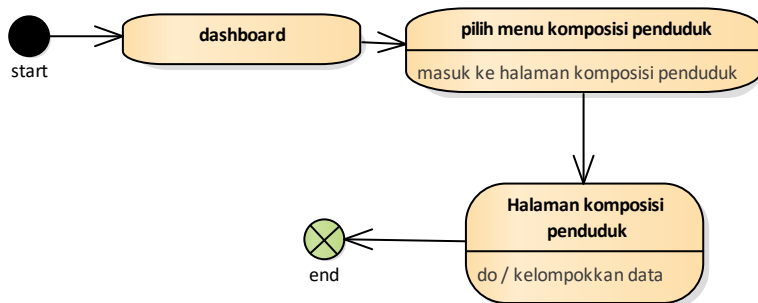
### 3.2.6.3 Statechart Diagram Jumlah Penduduk



Gambar 3.25 Statechart Diagram Mengelola Data Jumlah Penduduk

Pada menu mengelola data jumlah penduduk aktor akan memilih jumlah penduduk dan masuk ke dalam halaman data jumlah penduduk.

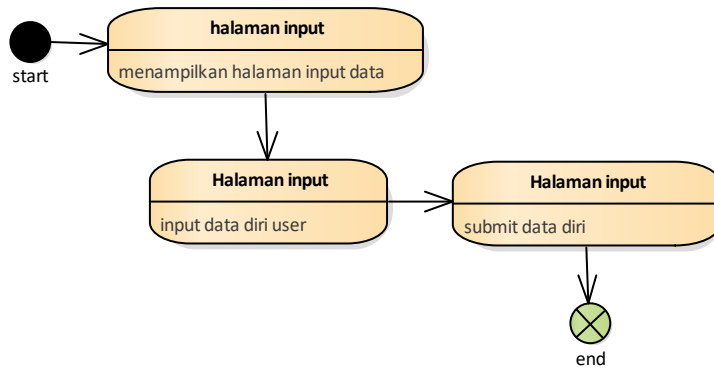
### 3.2.6.4 Statechart Diagram Komposisi Penduduk



Gambar 3.26 StateChart Diagram Mengelola Data Komposisi Penduduk

Pada menu mengelola data komposisi penduduk penduduk aktor akan memilih komposisi penduduk dan masuk ke dalam halaman data komposisi penduduk.

### 3.2.6.5 Statcechart Diagram Input Data Penduduk



Gambar 3.27 *StateChart Diagram Mengelola Data Komposisi Penduduk*

Pada menu mengelola data komposisi penduduk penduduk aktor akan memilih komposisi penduduk dan masuk ke dalam halaman data komposisi penduduk.







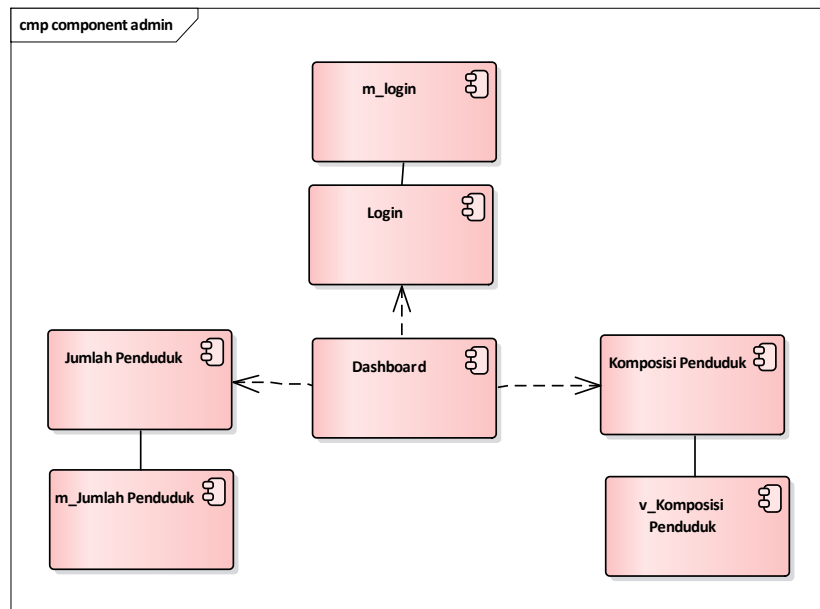




### **3.2.7 *Component Diagram***

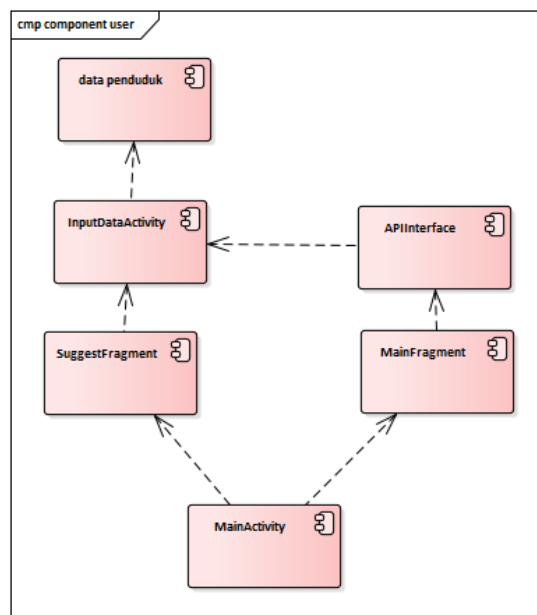
*Component diagram* atau diagram komponen merupakan diagram yang dibuat untuk menunjukkan sebuah organisasi atau ketergantungan diantara kumpulan komponen pada suatu sistem. Diagram komponen fokus pada komponen sistem yang dibutuhkan di suatu sistem. *Component diagram* ditunjukkan pada gambar di bawah ini:

### 3.2.7.1 Component Diagram Admin



Gambar 3.28 Component Diagram Admin.

### 3.2.7.2 Component Diagram User

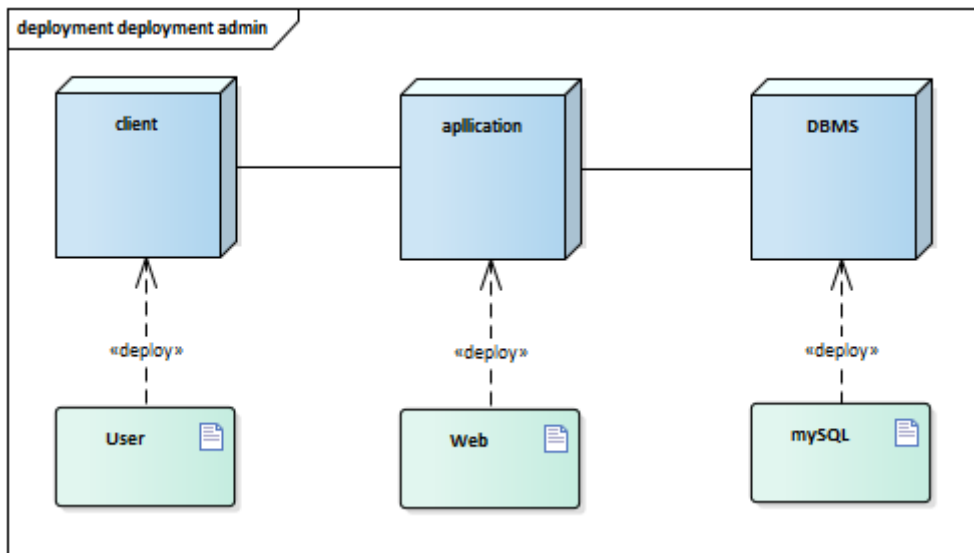


Gambar 3.29 Component Diagram User.

### 3.2.8 Deployment Diagram

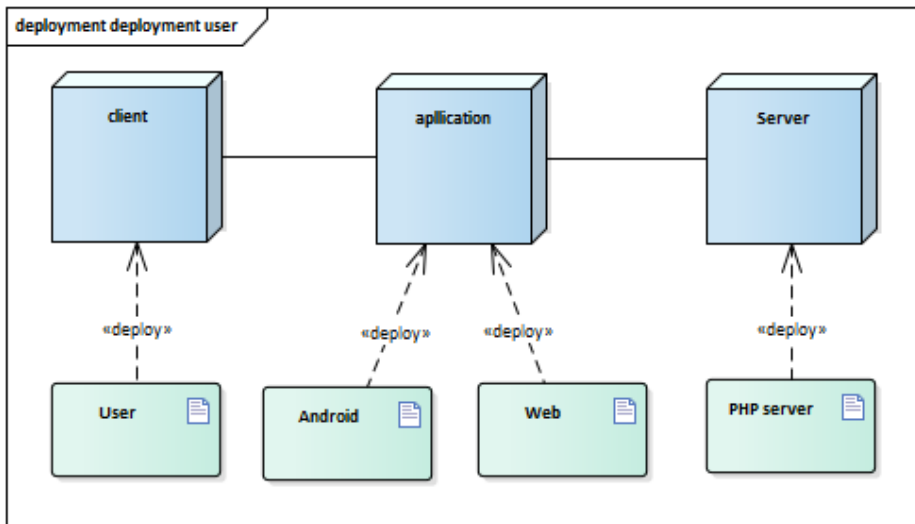
*Deployment* diagram atau diagram deployment merupakan diagram yang menunjukkan konfigurasi komponen dalam proses eksekusi aplikasi [31]. Berikut *deployment diagram* yang menggambarkan susunan fisik dan perangkat lunak dan sistem.

#### 3.2.8.1 Deployment Diagram Admin



Gambar 3.30 Deployment Diagram Admin.

### 3.2.8.2 Deployment Diagram User

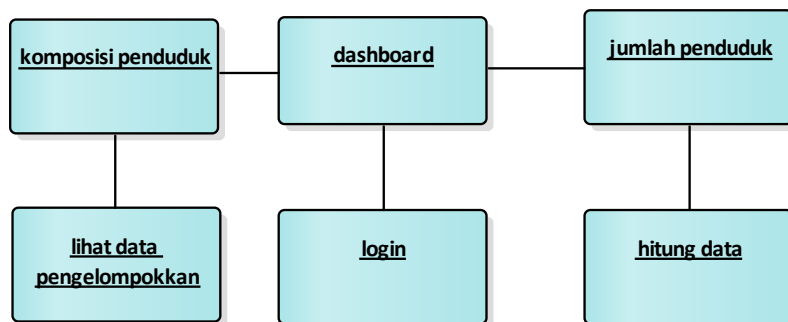


Gambar 3.31 Deployment Diagram Masyarakat

### 3.2.9 Struktur Menu

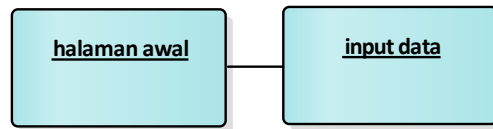
Adapun Struktur menu dari aplikasi sensus penduduk ini yaitu sebagai berikut :

#### 3.2.9.1 Struktur Menu Admin



Gambar 3.32 Struktur Menu Admin

### 3.2.9.2 Struktur Menu User



Gambar 3.33 *Struktur Menu User*

### 3.2.10 Perancangan Antar Muka

#### 3.2.10.1 Perancangan Antar Muka Admin

##### a. Perancangan Antar Muka *Login*

http://sensus-penduduk.co.id

Selamat datang di Web Sensus

Login (1)

username : masukkan username (2)

password : masukkan password (3)

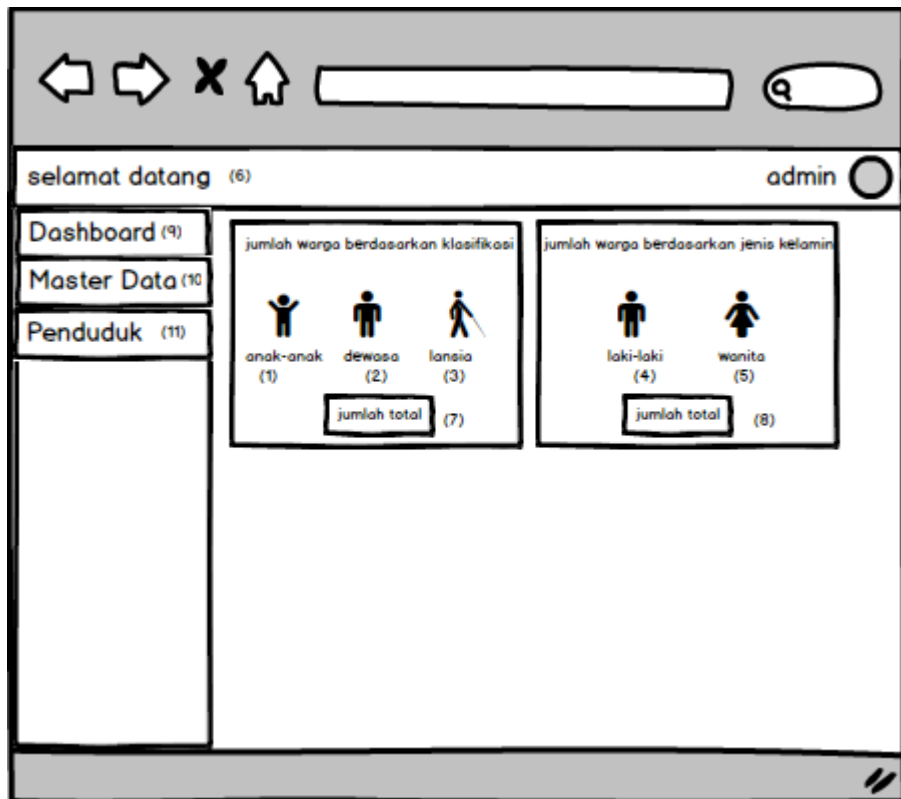
login (4)

Gambar 3.34 *Antar Muka Login*

- (1) *label* : Login
- (2) *textbox* : masukkan *username*
- (3) *textbox* : masukkan *password*
- (4) *button* : login



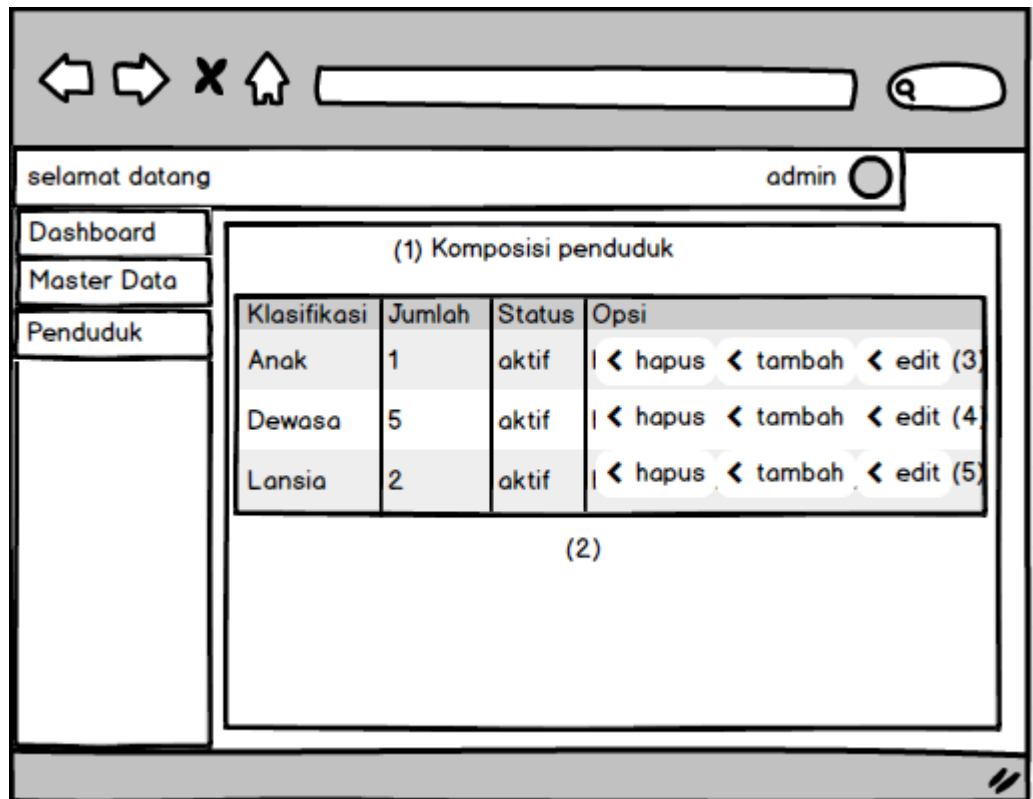
**b. Perancangan Antar Muka Halaman Utama**



*Gambar 3.35 Antar Muka Halaman Utama.*

- (1) *textview* : anak-anak
- (2) *textview* : dewasa
- (3) *textview* : lansia
- (4) *textview* : laki-laki
- (5) *textview* : wanita
- (6) *label* : selamat datang
- (7) *textview* : jumlah penduduk
- (8) *textview* : jumlah penduduk
- (9) *button* : dashboard
- (10) *button* : master data
- (11) *button* : penduduk

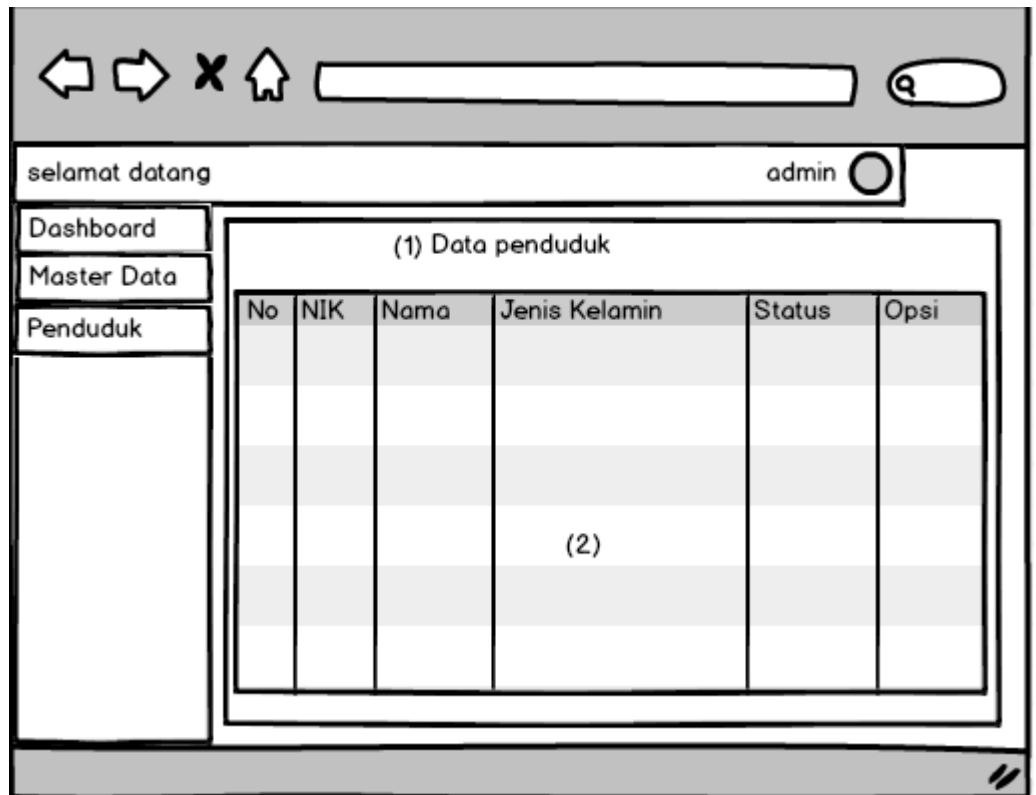
c. Perancangan Antar Muka Klasifikasi Penduduk



Gambar 3.36 Antar Muka Klasifikasi Penduduk.

- (1) label : komposisi penduduk
- (2) table : klasifikasi penduduk
- (3) button : hapus, tambah, edit
- (4) button : hapus, tambah, edit
- (5) button : hapus, tambah, edit

d. Perancangan Antar Muka Data Penduduk



Gambar 3.37 Antar Muka Data Penduduk.

(1) label : Data Penduduk

(2) table : Data Penduduk

### 3.2.10.2 Perancangan Antar Muka *User*

#### a. Perancangan Antar Muka *Input Data*

input data(1)

(2) NIK

(3) Nama

(4) Tanggal Lahir

status

(5) ☐ menikah  
☐ belum menikah

jenis kelamin

(6) ☐ laki-laki  
☐ perempuan

(7)

Gambar 3.38 *Antar Muka Input Data.*

- (1) label : input data
- (2) text input : nik
- (3) text input : nama
- (4) text input : tanggal lahir
- (5) radiogroup : status
- (6) radiogroup : jenis kelamin
- (7) button : simpan

## b. Perancangan Antar Muka Lihat Data

The image shows a mobile application interface for viewing resident data. At the top, there is a status bar with a Wi-Fi icon, signal strength bars, and the time 7:12. Below the status bar, the title "data penduduk(1)" is centered. The main content area contains a list of six items, each with a number in parentheses and a text input field:

- (2) tampil NIK
- (3) tampil Nama
- (4) tampil Tanggal Lahir
- (5) tampil status
- (6) tampil jenis kelamin

*Antar Muka Lihat Data.*

- (1) *label* : data penduduk
- (2) *textview* : tampil nik
- (3) *textview* : tampil nama
- (4) *textview* : tampil tanggal lahir
- (5) *textview* : tampil status
- (6) *textview* : tampil jenis kelamin