

# ANALISIS DAN PERANCANGAN APLIKASI PENGELOLAAN DATA KATEGORI KESEJAHTERAAN PENDUDUK DI DESA CIBUNUT MENGGUNAKAN ALGORITMA *DIVIDE AND CONQUER*

Dadan Zaliluddin<sup>1</sup>, Eti Rohaeti<sup>2</sup>

<sup>1</sup>Program Studi Informatika, Fakultas Teknik, Universitas Majalengka  
Jln. KH. Abdul Halim. No. 103 Majalengka 45418

E-mail: [dadanzuu@gmail.com](mailto:dadanzuu@gmail.com)

<sup>2</sup>Program Studi Informatika, Fakultas Teknik, Universitas Majalengka  
Jln. KH. Abdul Halim. No. 103 Majalengka 45418

E-mail: [etirohaeti113@gmail.com](mailto:etirohaeti113@gmail.com)

## ABSTRAK

Berkembangnya zaman menjadi tantangan negara untuk berkembang lebih baik dari waktu ke waktu. Hal ini menjadikan negara berusaha memberikan kesejahteraan sebaik mungkin kepada masyarakatnya. Masalah kemiskinan, pengangguran dan sosial merupakan beberapa permasalahan yang sering dihadapi dalam upaya peningkatan kesejahteraan. Penulis melakukan penelitian terhadap sistem yang digunakan oleh Desa Cibunut Kecamatan Argapura Kabupaten Majalengka tentang pengelolaan data kategori kesejahteraan penduduk yang masih manual yaitu berupa dokumen atau kertas dan seringnya melakukan pendataan ulang dikarenakan data hilang atau rusak. Salah satu solusi masalah ini adalah dengan menggunakan algoritma *divide and conquer* untuk mempermudah dalam pengelolaan data *real* menjadi sebuah aplikasi. Dalam pembuatan aplikasinya sendiri menggunakan pemrograman JAVA dan *tools* pendukung NetBeans 8.0. Hasil akhir dari penelitian ialah menghasilkan sebuah aplikasi berbasis dekstop yang dapat merekap data kategori kesejahteraan penduduk dan membantu dalam hal membuat laporan dengan fasilitas cetak.

## Kata Kunci

Aplikasi, Pengelolaan Data Kesejahteraan Penduduk, Algoritma *Divide and Conquer*, Dekstop

## 1. PENDAHULUAN

Berkembangnya zaman menjadi tantangan negara untuk berkembang lebih baik dari waktu ke waktu. Hal ini menjadikan negara berusaha memberikan kesejahteraan sebaik mungkin kepada masyarakat. Desa Cibunut adalah salah satu desa yang terletak di daerah pegunungan kecamatan Argapura, Kabupaten Majalengka, Jawa Barat. Penduduk Desa Cibunut terus mengalami pertumbuhan, dari tahun 2011 sebanyak 1629 jiwa sampai dengan tahun 2015 menjadi sebesar 1705 jiwa dengan luas wilayah 486,825 Ha.

Sistem pengelolaan data kategori kesejahteraan penduduk di Desa Cibunut masih memiliki kendala yaitu data masih dalam bentuk pembukuan atau arsip - arsip yang tidak dikelola dengan baik seperti arsip data yang sering hilang atau rusak, sehingga pendataan yang sudah dilakukan tidak efisien. Informasi yang sulit didapat mengakibatkan perangkat desa harus mendata ulang bahkan dalam

pembuatan laporan yang menjadi lama dan tidak akurat karena laporan hanya berupa lisan.

Algoritma *divide and conquer* merupakan algoritma yang berprinsip memecah-mecahkan permasalahan yang terlalu besar menjadi beberapa bagian kecil sehingga lebih mudah diselesaikan dan untuk meningkatkan kinerja *client* pada konfigurasi biasa, keuntungan terdapat pada berbagai kondisi implementasi *client*. Hal ini tidak hanya mengarah pemeriksaan lebih cepat.

Berdasarkan beberapa peristiwa yang terjadi, dapat disimpulkan bahwa penyimpanan data kesejahteraan penduduk di Desa Cibunut masih kurang baik. Hilangnya arsip data akan menimbulkan masalah yaitu lamanya mencari informasi dan belum akuratnya laporan karena masih berupa lisan.

Dalam penelitian ini penggunaan algoritma *divide and conquer* untuk menyelesaikan masalah-

masalah yang besar, dan dibagi (pecah) menjadi bagian yang lebih kecil dan menggunakan sebuah solusi untuk menyelesaikan problem awal, perancangan pengelolaan data kategori kesejahteraan penduduk hanya mengolah data tentang identitas kepala keluarga, identitas anggota keluarga dan keadaan rumah yang diolah secara komputerisasi berbasis *desktop* menggunakan bahasa *java* dan *tool netbeans IDE 8.0* dengan *database mysql*.

Dari hasil survei tersebut maka terbuka peluang untuk sebuah aplikasi komputer yang berupa aplikasi untuk mengelola data kategori kesejahteraan penduduk di Desa Cibunut dan memecahkan masalah dengan menggunakan algoritma *divide and conquer*. Tujuan perancangan sistem ini diharapkan mampu mempersingkat waktu kerja dalam pencarian data dan membantu dalam hal pembuatan laporan.

## 2. LANDASAN TEORI

### 2.1 Analisis

Analisis (*analysis*) adalah evaluasi situasi atau problem, termasuk tinjauan dari berbagai aspek dan sudut pandang. Dalam komputasi, analisis biasanya mencakup segi kontrol arus, kontrol kesalahan dan penelitian efisiensi. Seringkali problem yang lebih besar dibagi menjadi komponen-komponen yang lebih kecil sehinggalah dapat diteliti dan ditangani lebih mudah.

### 2.2 Perancangan Basis Data

Perancangan basis data diperlukan agar kita bisa memiliki basis data yang efisien dalam penggunaan ruang penyimpanan, cepat dalam pengaksesan dan mudah dalam manipulasi data.

### 2.3 Aplikasi

Aplikasi merupakan *software* yang berfungsi untuk melakukan berbagai bentuk pekerjaan atau tugas-tugas tertentu seperti penerapan, penggunaan dan penambahan data. Aplikasi dapat diartikan juga sebagai program komputer yang dibuat untuk menolong manusia dalam melakukan tugas tertentu [3]. Aplikasi adalah penggunaan dalam dalam suatu komputer, intruksi (*intruksi*) atau pernyataan (*statement*) yang disusun sedemikian rupa sehingga komputer dapat memproses *input* menjadi *output* [1].

### 2.4 Kesejahteraan

Kesejahteraan berasal dari kata “sejahtera”. Sejahtera ini mengandung pengertian dari bahasa Sansakerta “Catera” yang berarti payung. Dalam konteks ini, kesejahteraan yang terkandung dalam arti “Catera” (payung) adalah orang yang sejahtera

dimana seorang dapat memenuhi kebutuhan pokok, baik itu kebutuhan akan makanan, pakaian, tempat tinggal, air minum yang bersih serta kesempatan untuk melanjutkan pendidikan dan memiliki pekerjaan yang memadai yang dapat *Menu* njang kualitas hidupnya sehingga hidupnya bebas dari kemiskinan, kebodohan, ketakutan, atau kekhawatiran sehingga hidupnya aman tentram, baik lahir maupun batin. [2].

### 2.5 Penduduk

Penduduk adalah Orang yang tinggal, berdomisili, untuk jangka waktu yang relative lama di suatu daerah tertentu. Penduduk adalah orang yang matranya sebagai diri pribadi, anggota keluarga, anggota masyarakat, warga Negara, dan himpunan kuantitas yang bertempat tinggal di suatu tempat dalam batasan wilayah Negara pada waktu tertentu [6]

### 2.6 Algoritma Divide and Conquer

Pada algoritma *Divide and Conquer* ini memiliki tiga proses utama [4] yaitu:

#### 1. Divide

Membagi masalah menjadi beberapa sub-masalah yang memiliki kemiripan dengan masalah semula namun berukuran lebih kecil (idealnya berukuran hampir sama);

#### 1. Conquer

Memecahkan atau menyelesaikan masing-masing sub-masalah (secara rekursif);

#### 2. Combine

Menggabungkan solusi masing-masing sub-masalah sehingga membentuk solusi masalah semula.

#### a. Rational Unified Process (RUP)

RUP (*Rational Unified Process*) adalah pendekatan pengembangan perangkat lunak yang dilakukan berulang-ulang (*iterative*), fokus pada arsitektur (*architecture-centric*), lebih diarahkan berdasarkan pengguna kasus (*use case driven*). RUP merupakan proses rekayasa perangkat lunak dengan pendefinisian yang baik (*well defined*) dan penstrukturan yang baik (*well structured*). RUP menyediakan pendefinisian struktur yang baik untuk alur hidup proyek perangkat lunak. RUP adalah sebuah produk proses perangkat lunak yang dikembangkan oleh *Rational Software* yang di akuisasi oleh IBM dibulan februari 2003 [5].

Tahap atau fase yang terdapat dalam RUP adalah sebagai berikut [5]:

#### 1. Inception (permulaan)

Tahapan ini merupakan memodelkan proses bisnis yang dibutuhkan (*business modeling*) dan mendefinisikan kebutuhan akan sistem yang akan

dibuat;

## 2. *Elaboration* (perancangan)

Tahap ini lebih difokuskan pada perancangan arsitektur sistem;

## 3. *Construction* (konstruksi)

Tahap ini difokuskan pada pengembangan komponen dan fitur-fitur sistem. Tahap ini lebih pada implementasi dan pengujian sistem yang fokus pada implementasi perangkat lunak pada kode program;

## 2. *Transition* (transisi)

Tahap ini lebih pada *deployment* atau instalasi sistem agar dapat dimengerti oleh *user*. Tahap ini menghasilkan produk perangkat lunak dimana menjadi syarat dari *Initial Operational Capability Milestone* atau batas/tonggak kemampuan operasional awal.

### b. Pengenalan *Unified Modeling Language* (UML)

Pada perkembangan teknik pemrograman berorientasi objek, muncullah sebuah standarisasi bahasa pemodelan untuk pembangunan perangkat lunak yang dibangun menggunakan teknik pemrograman berorientasi objek, yaitu *Unified Modeling Language* (UML). UML muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun, dan dokumentasi dari sistem perangkat lunak. UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung [5].

#### 1. *Use Case Diagram*

*Use Case* atau diagram *use case* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut [5].

#### 2. *Class Diagram*

Diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Diagram kelas dibuat agar pembuatan program atau *programmer* membuat kelas-kelas sesuai rancangan di dalam diagram kelas agar antara dokumentasi perancangan dan perangkat lunak sinkron [5].

#### 3. *Activity Diagram*

Diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau *Menu* yang ada pada perangkat lunak. Yang perlu

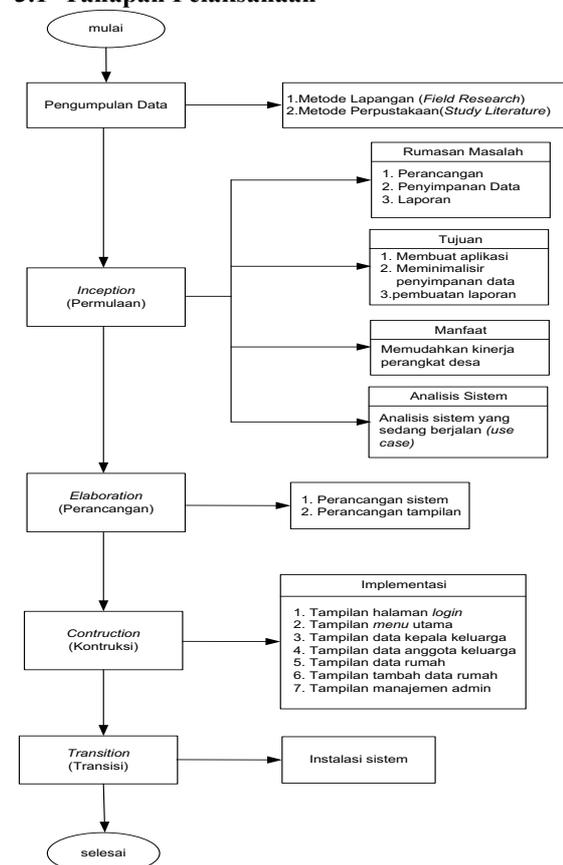
diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan sistem [5].

#### 4. *Sequence Diagram*

Diagram sekuen menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambar diagram sekuen maka harus diketahui objek-objek yang terlihat dalam *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu. Membuat diagram sekuen juga dibutuhkan untuk melihat skenario yang ada pada *use case* [5].

### 3. HASIL DAN PEMBAHASAN

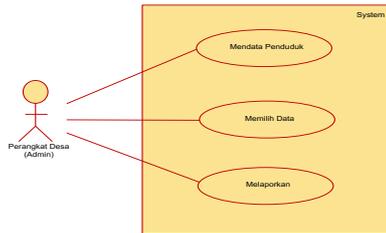
#### 3.1 Tahapan Pelaksanaan



Gambar 1 Tahapan Pelaksanaan

#### 3.2 Analisis Sistem yang Sedang Berjalan

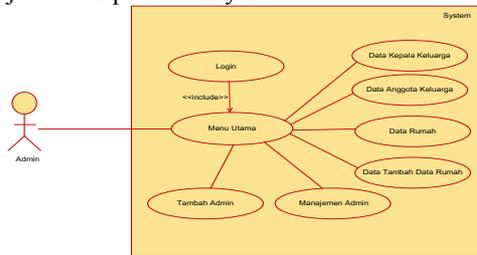
Berikut *use case* alur sistem pengelolaan data kategori kesejahteraan penduduk yang sedang berjalan di Desa Cibunut:



Gambar 2 Use Case Diagram Pengelolaan Data Kategori Kesejahteraan Penduduk yang Sedang Berjalan

### 3.3 Use Case Diagram Usulan

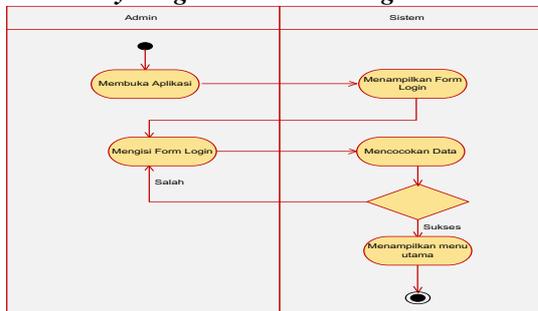
Selanjutnya dibuat *use case diagram* dari sistem yang akan dibuat. *Diagram Use case* yang diusulkan untuk pengelolaan data kategori kesejahteraan penduduk yaitu:



Gambar 3 Use Case Diagram Usulan

Dari gambar 3 dapat diketahui bahwa yang bertindak sebagai aktor dalam sistem ini adalah Perangkat Desa (Admin). Dalam sistem ini admin dapat melakukan pengolahan data kategori kesejahteraan penduduk seperti mengolah data kepala keluarga, mengolah data anggota keluarga, mengolah data rumah, mengolah tambah data rumah, mengolah data manajemen admin dan tambah admin namun admin harus *login* terlebih dahulu.

### 3.4 Activity Diagram Halaman Login

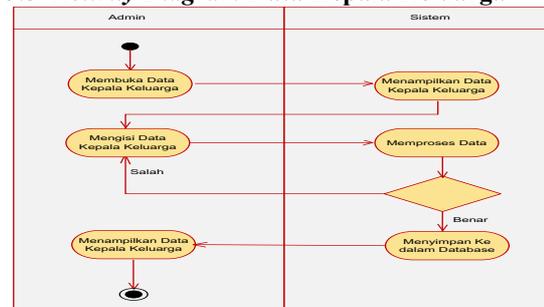


Gambar 4 Activity Diagram Login

Gambar 4 merupakan aktivitas *login* admin yang. Dimulai dengan admin membuka aplikasi data kategori kesejahteraan penduduk kemudian sistem menampilkan *form login*, selanjutnya admin dapat melakukan *login* dengan memasukkan *username* dan

*password* untuk dapat masuk kedalam sistem. Proses *login* itu sendiri diikuti juga validasi yang dilakukan oleh sistem dimana terdapat proses verifikasi *username* dan *password* apakah sukses atau salah.

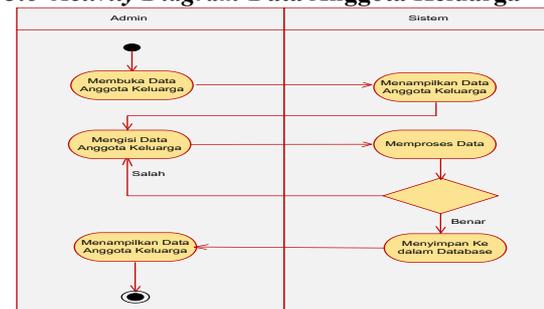
### 3.5 Activity Diagram Data Kepala Keluarga



Gambar 5 Activity Diagram Data Kepala Keluarga

Gambar 5 Menunjukkan aktivitas pengelolaan data kepala keluarga yang dilakukan oleh admin. Dimulai dari admin membuka *form* data kepala keluarga kemudian sistem menampilkan *form* data kepala keluarga, selanjutnya admin mengisi data kepala keluarga lalu sistem melanjutkan proses data yang di ikuti verifikasi yang dilakukan oleh sistem dimana terdapat proses verifikasi ketika menyimpan data apakah benar atau salah. Jika benar sistem langsung menyimpan data ke dalam *database* atau jika salah admin akan diminta mengisi data kepala keluarga kembali.

### 3.6 Activity Diagram Data Anggota Keluarga

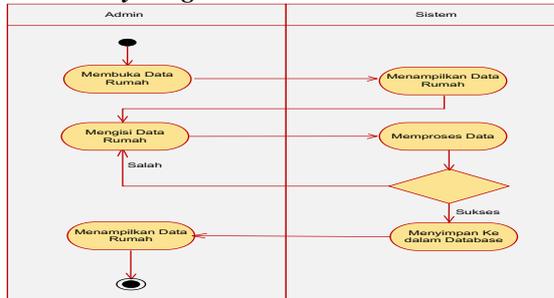


Gambar 6 Activity Diagram Data Anggota Keluarga

Gambar 6 Menunjukkan aktivitas pengelolaan data anggota keluarga yang dilakukan oleh admin. Dimulai dari admin membuka *form* data anggota keluarga kemudian sistem menampilkan *form* data anggota keluarga, selanjutnya admin mengisi data anggota keluarga lalu sistem melanjutkan proses data yang di ikuti verifikasi yang dilakukan oleh sistem dimana terdapat proses verifikasi ketika menyimpan data apakah benar atau salah. Jika benar sistem langsung menyimpan data ke dalam

database atau jika salah admin akan diminta mengisi data anggota keluarga kembali.

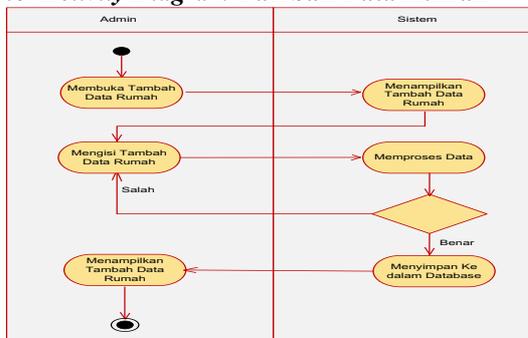
### 3.7 Activity Diagram Data Rumah



Gambar 7 Activity Diagram Data Rumah

Gambar 7 Menunjukkan aktivitas pengelolaan data rumah yang dilakukan oleh admin. Dimulai dari admin membuka *form* data rumah kemudian sistem menampilkan *form* data rumah, selanjutnya admin mengisi data rumah lalu sistem melanjutkan proses data yang di ikuti verifikasi yang dilakukan oleh sistem dimana terdapat proses verifikasi ketika menyimpan data apakah sukses atau salah. Jika sukses sistem langsung menyimpan data ke dalam *database* atau jika salah admin akan diminta mengisi data rumah kembali.

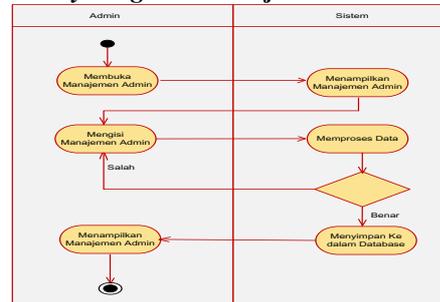
### 3.8 Activity Diagram Tambah Data Rumah



Gambar 8 Activity Diagram Tambah Data Rumah

Gambar 8 Menunjukkan aktivitas pengelolaan tambah data rumah yang dilakukan oleh admin. Dimulai dari admin membuka *form* tambah data rumah kemudian sistem menampilkan *form* tambah data rumah, selanjutnya admin mengisi tambah data rumah lalu sistem melanjutkan proses data yang di ikuti verifikasi yang dilakukan oleh sistem dimana terdapat proses verifikasi ketika menyimpan data apakah benar atau salah. Jika benar sistem langsung menyimpan data ke dalam *database* atau jika salah admin akan diminta mengisi tambah data rumah kembali.

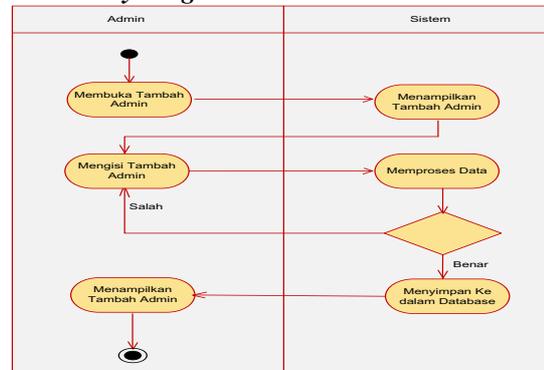
### 3.9 Activity Diagram Manajemen Admin



Gambar 9 Activity Diagram Data Admin

Gambar 9 Menunjukkan aktivitas pengelolaan manajemen admin yang dilakukan oleh admin. Dimulai dari admin membuka *form* manajemen admin kemudian sistem menampilkan *form* manajemen admin, selanjutnya admin mengisi ubah *password* atau ubah *username* lalu sistem melanjutkan proses data yang di ikuti verifikasi yang dilakukan oleh sistem dimana terdapat proses verifikasi ketika menyimpan data apakah benar atau salah. Jika benar sistem langsung menyimpan data ke dalam *database* atau jika salah admin akan diminta mengisi data ubah *password* atau data ubah *username* kembali.

### 3.10 Activity Diagram Tambah Admin

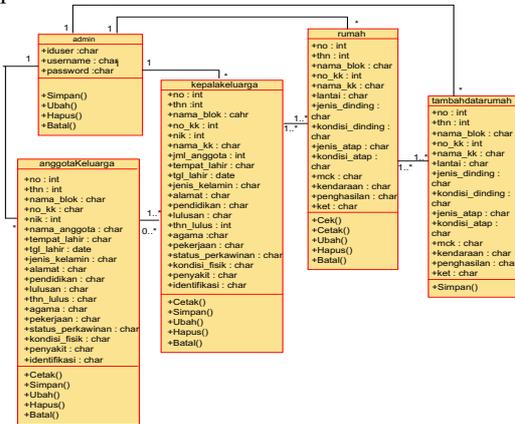


Gambar 10 Activity Diagram Tambah Admin

Gambar 10 Menunjukkan aktivitas pengelolaan tambah admin yang dilakukan oleh admin. Dimulai dari admin membuka *form* tambah admin kemudian sistem menampilkan *form* tambah admin, selanjutnya admin mengisi data admin lalu sistem melanjutkan proses data yang di ikuti verifikasi yang dilakukan oleh sistem dimana terdapat proses verifikasi ketika menyimpan data apakah benar atau salah. Jika benar sistem langsung menyimpan data ke dalam *database* atau jika salah admin akan diminta mengisi data admin kembali.

### 3.11 Class Diagram

Diagram kelas menggambarkan kelas-kelas yang saling berinteraksi. Admin berinteraksi dengan data rumah, tambah data rumah, data kepala keluarga, data anggota keluarga, dan data admin. Terlihat pada Gambar 11.

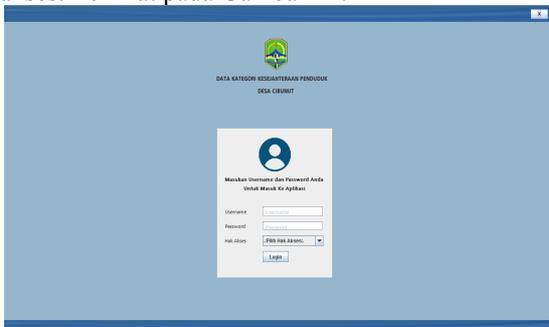


Gambar 11 Class Diagram

## 4. TAMPILAN PADA LAYAR

### 4.1 Tampilan Halaman Login

Tampilan halaman login merupakan tampilan dari form login di mana terdapat perintah untuk memasukkan *username*, *password* dan memilih hak akses. Terlihat pada Gambar 12:



Gambar 12 Tampilan Halaman Login

Tabel 1 adalah tabel hasil pengecekan terhadap halaman login yang telah dilakukan.

Tabel 1 Pengecekan halaman Login

No	Input	Hasil yang diharapkan	Pengamatan	Hasil /Uji
1	Mengkosongkan form kemudian klik simpan	Muncul peringatan <i>username</i> atau <i>password</i> salah	Muncul peringatan <i>username</i> atau <i>password</i> salah	OK / sesuai
2	Memasukkan <i>username</i> , <i>password</i> dan hak akses yang tidak ada dalam database, klik tombol login	Muncul pesan peringatan <i>username</i> , <i>password</i> dan hak akses tidak sesuai	Muncul pesan peringatan <i>username</i> , <i>password</i> dan hak akses tidak sesuai	OK / sesuai
3	Memasukkan <i>username</i> , <i>password</i> dan hak akses yang	<i>login</i> berhasil	<i>login</i> berhasil	OK / sesuai

No	Input	Hasil yang diharapkan	Pengamatan	Hasil /Uji
4	sudah ada di database, klik tombol login Klik tombol X	Keluar dari aplikasi	Keluar dari aplikasi	OK / sesuai

### 4.2 Tampilan Menu Utama

Tampilan *Menu* utama merupakan tampilan dimana terdapat media untuk pengoperasian sistem yaitu terdapat tombol data kepala keluarga, tombol data anggota keluarga, tombol data rumah, tombol tambah data rumah, tombol manajemen admin, tombol keluar dan disebelah kiri terdapat uraian gambaran umum Desa cibunut. Terlihat seperti Gambar 13:



Gambar 13 Tampilan Menu Utama

Tabel 2 adalah tabel hasil pengecekan terhadap menu utama yang telah dilakukan.

Tabel 2 Pengecekan menu utama

No	Input	Hasil yang diharapkan	Pengamatan	Hasil /Uji
1	Klik tombol data kepala keluarga	Muncul tampilan entry data kepala keluarga dengan tombol Simpan, ubah, hapus,batal,cetak, dan selengkapnya	Muncul tampilan entry data kepala keluarga dengan tombol Simpan, ubah, hapus,batal,cetak, dan selengkapnya	OK / sesuai
2	Klik tombol data anggota keluarga	Muncul tampilan entry data anggota keluarga dengan tombol Simpan, ubah, hapus,batal,cetak, dan cetak	Muncul tampilan entry data anggota keluarga dengan tombol Simpan, ubah, hapus, batal, dan cetak	OK / sesuai
3	Klik tombol data rumah	Muncul tampilan data rumah dengan tombol cek, cetak, ubah, hapus, batal	Muncul tampilan data rumah dengan tombol cek, cetak, ubah,hapus,batal	OK / sesuai
4	Klik tombol tambah data rumah	Muncul tampilan entry tambah data rumah dengan tombol simpan	Muncul tampilan entry tambah data rumah dengan tombol simpan	OK / sesuai
5	Klik tombol manajemen admin	Muncul tampilan ubah <i>username</i> dan <i>password</i> dengan tombol ubah	Muncul tampilan ubah <i>username</i> dan ubah <i>password</i> dengan tombol ubah	OK / sesuai
6	Klik tombol tambah admin	Muncul tampilan data tambah admin dengan tombol simpan, ubah, hapus,dan batal	Muncul tampilan data tambah admin dengan tombol simpan, ubah, hapus,dan batal	OK / sesuai

No	Input	Hasil yang diharapkan	Pengamatan	Hasil /Uji
7	Klik tombol keluar	Muncul tampilan pemberitahuan dengan tombol ya dan tidak lalu akan keluar dari aplikasi	Muncul tampilan pemberitahuan dengan tombol ya dan tidak lalu akan keluar dari aplikasi	OK / sesuai

### 4.3 Tampilan Menu Data Kepala Keluarga

Menu data kepala keluarga merupakan tampilan pengoperasian *input* (masukan) data kepala keluarga. Terlihat pada Gambar 14:



Gambar 14 Tampilan Menu Data Kepala Keluarga

Tabel 3 adalah tabel hasil pengecekan terhadap *menu* data kepala keluarga yang telah dilakukan.

Tabel 3 Pengecekan *menu* data kepala keluarga

No	Input	Hasil yang diharapkan	Pengamatan	Hasil /Uji
1	Mengosongkan <i>form</i> kemudian klik simpan	Muncul pesan peringatan data belum lengkap	Muncul pesan peringatan data belum lengkap	OK / sesuai
2	Isi <i>field</i> periode, no kk, nik, jumlah anggota dan tahun lulus dengan karakter	Muncul pesan data hanya angka	Muncul pesan data hanya angka	OK / sesuai
3	Isi <i>field</i> pada kolom yang tersedia, klik tombol Simpan	Muncul pesan Data Berhasil Disimpan	Muncul pesan Data Berhasil Disimpan	OK / sesuai
4	Ubah data pada <i>field</i> , klik tombol ubah	Muncul pesan Data Berhasil Diubah	Muncul pesan Data Berhasil Diedit	OK / sesuai
5	Klik tombol hapus	Muncul pesan Data Berhasil Dihapus	Muncul pesan Data Berhasil Dihapus	OK / sesuai
6	Klik tombol kembali	Muncul tampilan <i>menu</i> utama	Muncul tampilan <i>menu</i> kepala utama	OK / sesuai
7	Isi <i>field</i> periode dan no kk, klik tombol cetak	Muncul properti print	Muncul properti print	OK / sesuai
8	Isi <i>field</i> masukan kata kunci	Muncul data kepala keluarga sesuai kata kunci	Muncul data kepala keluarga sesuai kata kunci	OK / sesuai

### 4.4 Tampilan Menu Anggota Keluarga

Menu data anggota keluarga merupakan tampilan pengoperasian *input* (masukan) data anggota keluarga. Terlihat pada Gambar 15:



Gambar 15 Tampilan Menu Data Anggota Keluarga

Tabel 4 adalah tabel hasil pengecekan terhadap *menu* data anggota keluarga yang telah dilakukan.

Tabel 4 Pengecekan *menu* data anggota keluarga

No	Input	Hasil yang diharapkan	Pengamatan	Hasil /Uji
1	Mengosongkan <i>form</i> kemudian klik simpan	Muncul pesan peringatan data belum lengkap	Muncul pesan peringatan data belum lengkap	OK / sesuai
2	Isi <i>field</i> periode, no kk, nik dan tahun lulus dengankarakter	Muncul pesan data hanya angka	Muncul pesan data hanya angka	OK / sesuai
3	Isi <i>field</i> pada kolom yang tersedia, klik tombol Simpan	Muncul pesan Data Berhasil Disimpan	Muncul pesan Data Berhasil Disimpan	OK / sesuai
4	Ubah data pada <i>field</i> , klik tombol ubah	Muncul pesan Data Berhasil Diubah	Muncul pesan Data Berhasil Diubah	OK / sesuai
5	Klik tombol hapus	Muncul pesan Data Berhasil Dihapus	Muncul pesan Data Berhasil Dihapus	OK / sesuai
6	Klik tombol batal	<i>field</i> yang sudah terisi akan kosong	<i>field</i> yang sudah terisi akan kosong	OK / sesuai
7	Isi <i>field</i> periode dan no kk, klik tombol cetak	Muncul properti print	Muncul properti print	OK / sesuai
8	Isi <i>field</i> masukan kata kunci	Muncul data kepala keluarga sesuai kata kunci	Muncul data kepala keluarga sesuai kata kunci	OK / sesuai

### 4.5 Tampilan Menu Data Rumah

Menu data rumah merupakan tampilan pengoperasian data rumah. Terlihat pada Gambar 16:



Gambar 16 Tampilan *Menu* Data Rumah

Tabel 5 adalah tabel hasil pengecekan terhadap *menu* data rumah yang telah dilakukan.

Tabel 5 Pengecekan *menu* data rumah

No	Input	Hasil yang diharapkan	Pengamatan	Hasil /Uji
1	Mengosongkan <i>form</i> kemudian klik ubah	Muncul pesan peringatan data belum lengkap	Muncul pesan peringatan data belum lengkap	OK / sesuai
2	Isi <i>field</i> masukan no kk, periode, no kk dan nik dengan karakter	Muncul pesan data hanya angka	Muncul pesan data hanya angka	OK / sesuai
3	Ubah data pada <i>field</i> , klik tombol ubah	Muncul pesan Data Berhasil Diubah	Muncul pesan Data Berhasil Diubah	OK / sesuai
4	Klik tombol hapus	Muncul pesan Data Berhasil Dihapus	Muncul pesan Data Berhasil Dihapus	OK / sesuai
5	Klik tombol batal	<i>field</i> yang sudah terisi akan kosong	<i>field</i> yang sudah terisi akan kosong	OK / sesuai
6	Isi <i>field</i> masukan no kk, klik tombol cek	Muncul peringatan data di temukan, muncul tampilan tambah data rumah	Muncul peringatan data di temukan, muncul tampilan tambah data rumah	OK / sesuai
7	Isi <i>field</i> periode dan kategori, klik tombol cetak	Muncul propertis print	Muncul propertis print	OK / sesuai
8	Isi <i>field</i> masukan kata kunci	Muncul data kepala keluarga sesuai kata kunci	Muncul data kepala keluarga sesuai kata kunci	OK / sesuai

#### 4.6 Tampilan *Menu* Tambah Data Rumah

*Menu* tambah data rumah merupakan tampilan pengoperasian *input* (masukan) data rumah. Terlihat pada Gambar 17:



Gambar 17 Tampilan *Menu* Tambah Data Rumah

Tabel 6 adalah tabel hasil pengecekan terhadap *menu* tambah data rumah yang telah dilakukan.

Tabel 6 Pengecekan *menu* tambah data rumah

No	Input	Hasil yang diharapkan	Pengamatan	Hasil /Uji
1	Mengosongkan <i>form</i> kemudian klik simpan	Muncul pesan peringatan data belum lengkap	Muncul pesan peringatan data belum lengkap	OK / sesuai
2	Isi <i>field</i> periode, no kk, dan nik dengan karakter	Muncul pesan data hanya angka	Muncul pesan data hanya angka	OK / sesuai
3	Isi <i>field</i> pada kolom yang tersedia, klik tombol Simpan	Muncul pesan Data Berhasil Disimpan	Muncul pesan Data Berhasil Disimpan	OK / sesuai

#### 4.6 Tampilan *Menu* Manajemen Admin

*Menu* manajemen admin merupakan tampilan untuk mengubah *username* atau *password*. Terlihat pada Gambar 18:



Gambar 18 Tampilan *Menu* Manajemen Admin

Tabel 7 adalah tabel hasil pengecekan terhadap *menu* manajemen admin yang telah dilakukan.

Tabel 7 Pengecekan *menu* manajemen admin

No	Input	Hasil yang diharapkan	Pengamatan	Hasil /Uji
1	Mengosongkan <i>form</i> ubah <i>username</i>	Muncul pesan peringatan	Muncul pesan peringatan	OK / sesuai

	kemudian klik ubah	data belum lengkap	data belum lengkap	
2	Memasukkan <i>username</i> baru dan <i>password</i> lama yang tidak ada dalam <i>database</i> , klik tombol ubah	Muncul pesan peringatan <i>password</i> lama tidak cocok	Muncul pesan peringatan <i>password</i> lama tidak cocok	OK / sesuai
3	Memasukkan <i>username</i> baru dan <i>password</i> lama yang sudah ada dalam <i>database</i> , klik tombol ubah	Muncul pesan peringatan ubah <i>username</i> berhasil	Muncul pesan peringatan ubah <i>username</i> berhasil	OK / sesuai
4	Mengosongkan <i>form</i> ubah <i>password</i> kemudian klik ubah	Muncul pesan peringatan data belum lengkap	Muncul pesan peringatan data belum lengkap	OK / sesuai
5	Memasukkan <i>password</i> lama yang tidak ada dalam <i>database</i>	Muncul pesan peringatan <i>password</i> lama tidak cocok	Muncul pesan peringatan <i>password</i> lama tidak cocok	OK / sesuai
6	Memasukkan <i>password</i> lama yang sudah ada dalam <i>database</i> , memasukan <i>password</i> baru dan ulangi <i>password</i> baru yang tidak sesuai, klik tombol ubah	Muncul pesan peringatan mohon ulangi <i>password</i> baru	Muncul pesan peringatan mohon ulangi <i>password</i> baru	OK / sesuai
7	Memasukkan <i>password</i> lama yang sudah ada dalam <i>database</i> , memasukan <i>password</i> baru dan ulangi <i>password</i> baru yang sesuai, klik tombol ubah	Muncul pesan peringatan ubah <i>password</i> berhasil	Muncul pesan peringatan ubah <i>password</i> berhasil	OK / sesuai

#### 4.7 Tampilan Menu Tambah Admin

*Menu* tambah admin merupakan tampilan pengoperasian *input* (masukan) data admin. Terlihat pada Gambar 19



Gambar 19 Tampilan *Menu* Tambah Admin

Tabel 8 adalah tabel hasil pengecekan terhadap *menu* tambah admin yang telah dilakukan.

Tabel 8 Pengecekan *menu* tambah admin

No	<i>Input</i>	Hasil yang diharapkan	Pengamatan	Hasil /Uji
1	Mengosongkan <i>form</i> kemudian klik simpan	Muncul pesan peringatan data belum lengkap	Muncul pesan data belum lengkap	OK / sesuai
2	Isi <i>field</i> pada kolom yang tersedia, klik tombol Simpan	Muncul pesan Data Berhasil Disimpan	Muncul pesan Data Berhasil Disimpan	OK / sesuai
3	Ubah data pada <i>field</i> , klik tombol ubah	Muncul pesan Data Berhasil Diubah	Muncul pesan Data Berhasil Diubah	OK / sesuai
4	Klik tombol hapus	Muncul pesan Data Berhasil Dihapus	Muncul pesan Data Berhasil Dihapus	OK / sesuai
5	Klik tombol batal	<i>field</i> yang sudah terisi akan kosong	<i>field</i> yang sudah terisi akan kosong	OK / sesuai

## 5 KESIMPULAN DAN SARAN

Berdasarkan dari keseluruhan yang sudah dibahas seperti pada hasil dan pembahasan serta tampilan pada layar dapat ditarik kesimpulan bahwa hasil penelitian yang dilaksanakan di Desa Cibunut, menghasilkan aplikasi pengelolaan data kategori kesejahteraan penduduk dalam melakukan pengolahan data kepala keluarga, data anggota keluarga dan data rumah serta dapat membantu perangkat desa (admin) membuat laporan kategori kesejahteraan penduduk, namun dalam *form* anggota keluarga belum dapat menampilkan informasi secara lengkap (kepala keluarga dengan anggota keluarga) dan penginputan NIK masih manual.

Untuk pengembangan selanjutnya agar lebih baik yaitu sistem yang lebih kompleks dan dapat dilengkapi dengan fasilitas statistik data untuk mempermudah dalam mengetahui perkembangan kesejahteraan penduduk.



## 9<sup>th</sup> Industrial Research Workshop and National Seminar



Peran Penelitian dan Inovasi di Era Industri 4.0 Dalam Mewujudkan  
Pembangunan Berkelanjutan Menuju Kemandirian Bangsa

### DAFTAR PUSTAKA

- [1] Budiman, A., & Mulyani, A. (2016). Rancang Bangun Aplikasi Sistem Informasi Persediaan Barang Di TB.Indah Jaya Berbasis Dekstop. *Jurnal Algoritma*, 374-378.
- [2] Fahrudin, A. (2012). *Pengantar Kesejahteraan Sosia*. Bandung: PT Refika Aditama.
- [3] Muthohari, A., Bunyamin, & Rahayu, S. (2016). Pengembangan Aplikasi Kasir Pada Sistem Informasi Rumah Makan Padang Ariung. *Jurnal Algoritma*, 157-163.
- [4] Ponidi, & Sandy, F. (2015). Perancangan Sistem Informasi Pendataan Penduduk Berbasis Web Menggunakan Metode Waterfall Pada Kecamatan Gadingrejo. *Technology Acceptance Model*, 69.
- [5] Rosa, A. S., & Shalahuddin, M. (2015). *Rekayasa Perangkat Lunak Menggunakan UML dan JAVA*. Bandung: Informatika Bandung.
- [6] Sinandar, D., Mardjoko, B. P., & Tanudjaja, H. (2017). Penerapan Algoritma Divide and Conquer pada Perancangan Sistem Identitas Penduduk. *TESLA*, 45-46.