

BAB III

ANALISIS DAN PERANCANGAN SISTEM

3.1 Analisis Sistem

Analisis sistem merupakan suatu analisis yang terdiri dari suatu sistem informasi yang utuh ke dalam suatu komponen dengan maksud untuk mengevaluasi permasalahan-permasalahan, kesempatan, hambatan dan kebutuhan yang diharapkan sehingga dapat diusulkan teori perbaikannya. Tahap ini bertujuan untuk mengetahui mekanisme aplikasi, proses-proses yang terlibat dalam aplikasi serta hubungan-hubungan proses. Ada tiga komponen atau perangkat yang dibutuhkan untuk membantu kinerja sistem agar tujuan dari sistem ini dapat tercapai. Perangkat tersebut adalah perangkat keras (*Hardware*) berupa komputer, perangkat lunak (*Software*) berupa program dan perangkat manusia (*Brainware*), yakni manusia. Hasil dari sistem ini adalah sebuah aplikasi sistem yang didalamnya terdapat informasi yang berguna bagi masyarakat.

3.1.1 Analisis Kebutuhan Sistem

Analisis kebutuhan sistem dilakukan untuk mengetahui fasilitas yang harus disediakan atau dimiliki oleh sistem agar dapat melayani kebutuhan pengguna sistem. Fungsi utama dari sistem ini adalah meningkatkan potensi Desa dan memberikan pelayanan administrasi kepada masyarakat yang dengan cepat dan tepat. Pada tahap ini akan di identifikasikan informasi-informasi yang diperlukan.

3.1.2 Analisis Peran Sistem

Sistem yang dibangun mempunyai peranan sebagai berikut:

1. Sistem yang dibuat dapat meng-*update* data kependudukan yang meliputi data penduduk menurut kelompok umur, pekerjaan, tingkat pendidikan dan agama.
2. Sistem ini juga memberikan *notifikasi* pada proses pengurusan administrasi.
3. Sistem ini dapat menghasilkan *output* pelayanan administrasi bagi masyarakat dengan cepat dan efisien.

3.1.3 Analisis Peran Pengguna

Pada sistem ini terdapat tiga kategori pengguna yaitu *admin*, RT dan warga. Dimana pihak desa berperan sebagai *admin* yang bertugas untuk mengelola semua data informasi, data akun RT dan mencetak surat yang diurus oleh masyarakat melalui RT dalam *web* ini. Pada *website* ini warga melakukan pemesanan surat yang ingin diurus di kantor desa Oenak. Sedangkan RT berperan sebagai perantara untuk meneruskan pesan surat dari masyarakat, agar surat tersebut dapat diproses oleh *admin*.

3.2 Sistem Perangkat Pendukung

Untuk menghasilkan *output* yang baik, maka sistem harus didukung oleh sistem perangkat keras (*hardware*) dan perangkat lunak (*software*).

3.2.1 Sistem Perangkat Keras (*Hardware*)

Perangkat keras (*hardware*) adalah perangkat fisik dari sebuah sistem komputer. Umumnya terdiri dari tiga jenis perangkat keras (*hardware*), yaitu

perangkat masukan, perangkat keluaran dan perangkat pengolah. Adapun komponen perangkat keras (*hardware*) yang digunakan dalam perancangan sistem antara lain :

- a. *Processor Intel(R) Core (TM) i3-7020U*
- b. *RAM 2 Gb.*
- c. *Flashdisk 120 GB.*
- d. *Mouse Asus.*
- e. *Printer Canon MP287 Series.*

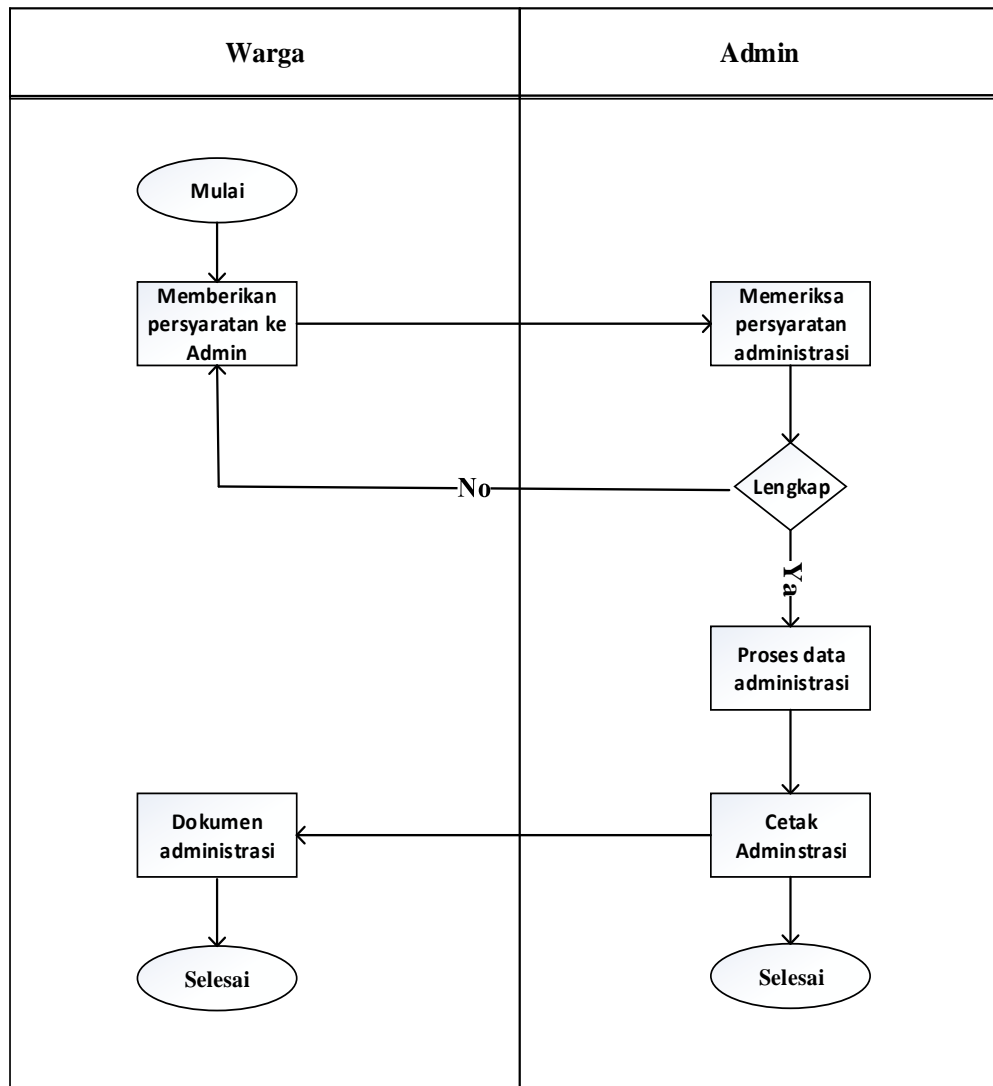
3.2.2 Sistem Perangkat Lunak (*Software*)

Adapun perangkat lunak (*software*) yang dibutuhkan dalam perancangan sistem antara lain sebagai berikut :

- a. Sistem Operasi Windows
- b. *PHP*
- c. *MySQL*
- d. *Visual Studio Code*
- e. *Visio 2013*
- f. *Ms-word 2013*

3.3 Desain Sistem

3.3.1 *Flowchart* sistem yang sedang berjalan

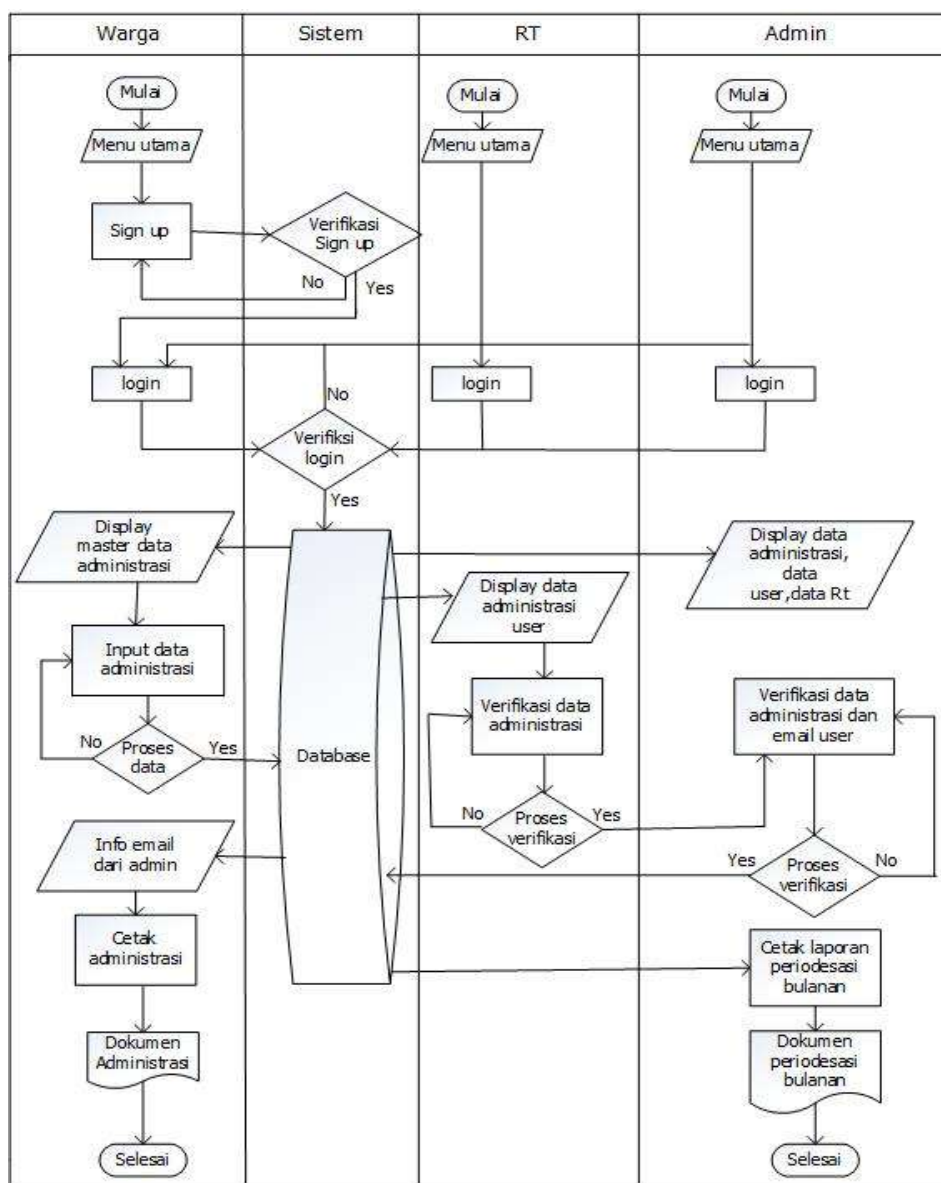


Gambar 3.1 *Flowchart* Sistem yang sedang berjalan.

Berdasarkan *flowchart* sistem yang sedang berjalan di atas seringkali warga langsung ke kantor desa Oenak untuk mengurus administrasi dengan membawa persyaratan untuk dikumpulkan ke *admin* untuk diproses menjadi sebuah dokumen administrasi.

3.3.2 Flowchart Sistem yang akan diusulkan

Flowchart system dibuat untuk menunjukkan alur kerja dari suatu sistem secara umum. Dengan adanya *flowchart* sistem, kita juga dapat mengidentifikasi permasalahan yang ada dan menganalisa kebutuhan dari sistem tersebut sebagai langkah awal dalam rancangan sistem yang akan dibentuk.



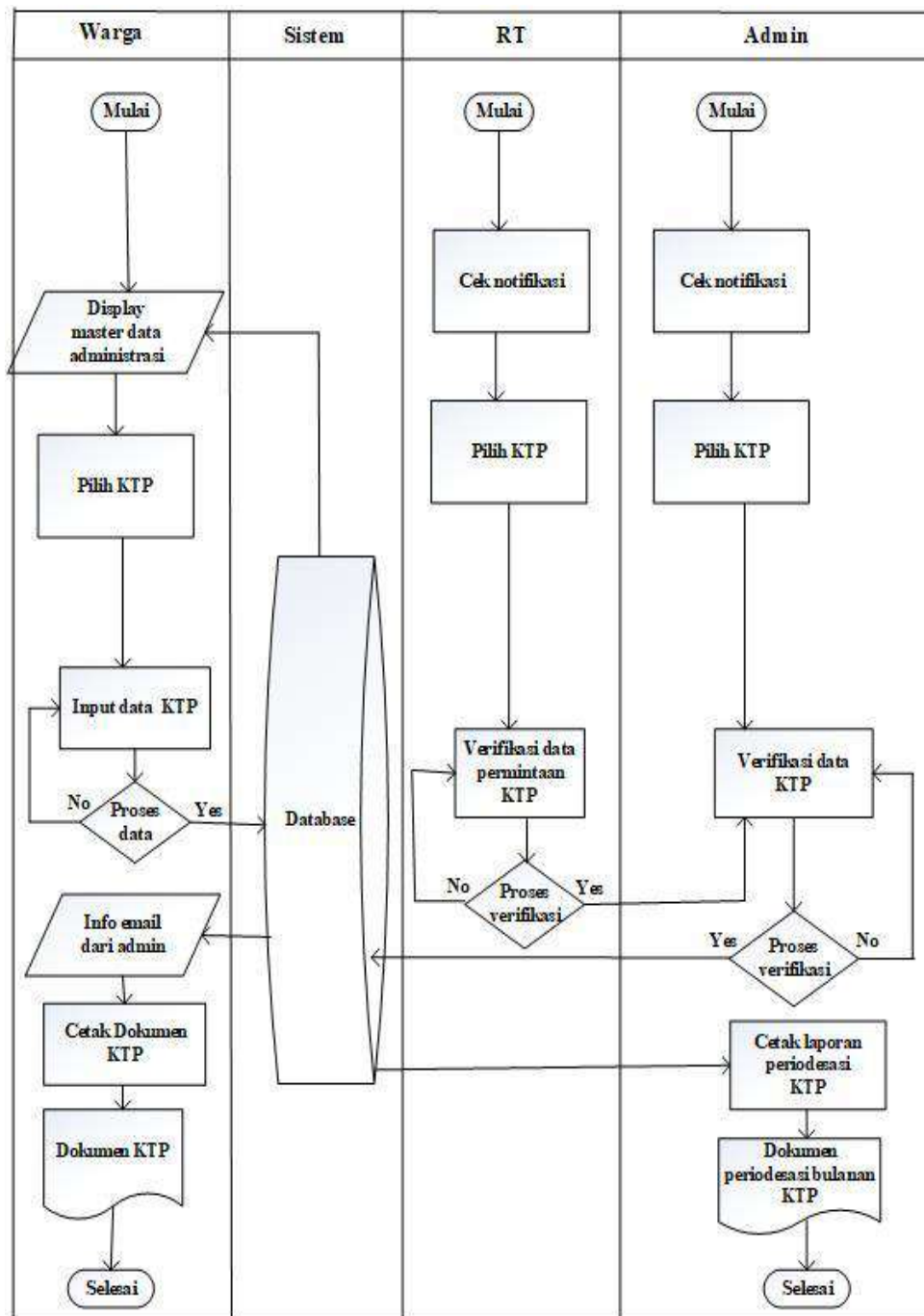
Gambar 3.2 Flowchart Sistem yang akan diusulkan.

Berdasarkan *flowchart* di atas, dapat dijelaskan bahwa warga, RT dan admin akan masuk ke *display* menu utama, lalu warga yang belum terdaftar akan melakukan proses *sign up* terlebih dahulu. Setelah data *sign up user* diverifikasi oleh RT maka warga dapat melanjutkan *login* untuk melakukan proses penginputan data administrasi untuk diverifikasi oleh RT dan admin agar dapat dicetak menjadi sebuah dokumen yang warga perlukan. Selain itu admin akan mengelolah data warga, data RT, dan melakukan proses verifikasi data administrasi warga yang telah diverifikasi sebelumnya oleh RT dan memberikan notifikasi *email* kepada warga bahwa data administrasi yang warga perlukan sudah selesai diproses.

Berikut adalah gambar *flowchart* sistem yang akan diusulkan yang dijelaskan secara terperinci setiap proses pengajuan data administrasi yang dimulai dari warga, RT dan *Admin*.

1. Surat Pengantar Pembuatan KTP

Pada Gambar 3.3, warga masuk ke tampilan master data administrasi lalu memilih menu KTP untuk menginput persyaratan pengurusan pengantar KTP yang diminta dan tekan simpan. Selanjutnya RT akan mendapatkan notifikasi lalu membuka surat keterangan pembuatan KTP dan mengecek kelengkapan dari persyaratan yang dikirim. Jika sudah lengkap maka RT melakukan konfirmasi untuk diteruskan ke *admin* untuk melakukan verifikasi data KTP untuk dikirimkan ke warga untuk dicetak dokumen administrasinya.

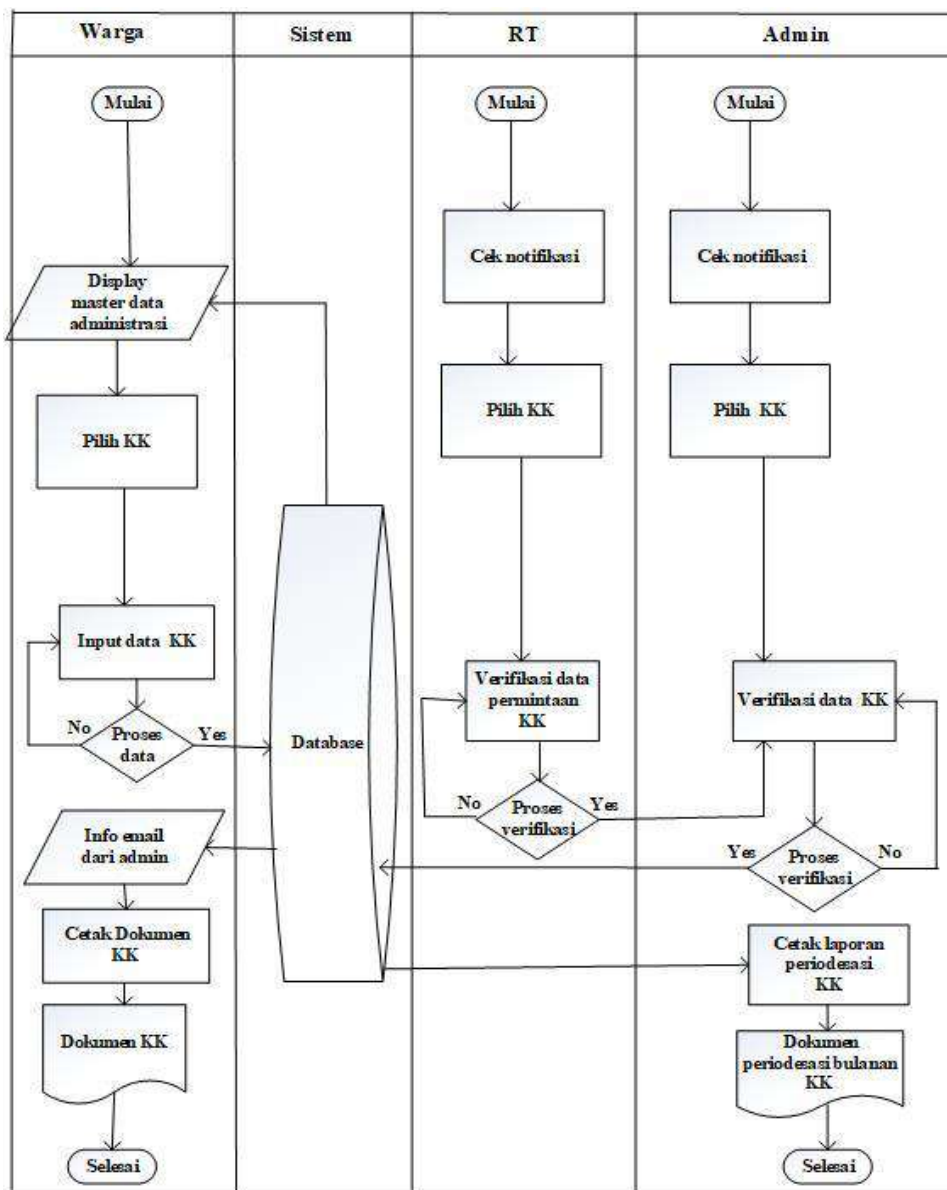


Gambar 3.3 Flowchart Surat Pengantar KTP.

2. Surat Pengantar Pembuatan Kartu Keluarga (KK)

Pada Gambar 3.4, warga masuk ke tampilan master data administrasi lalu memilih menu KK untuk menginput persyaratan

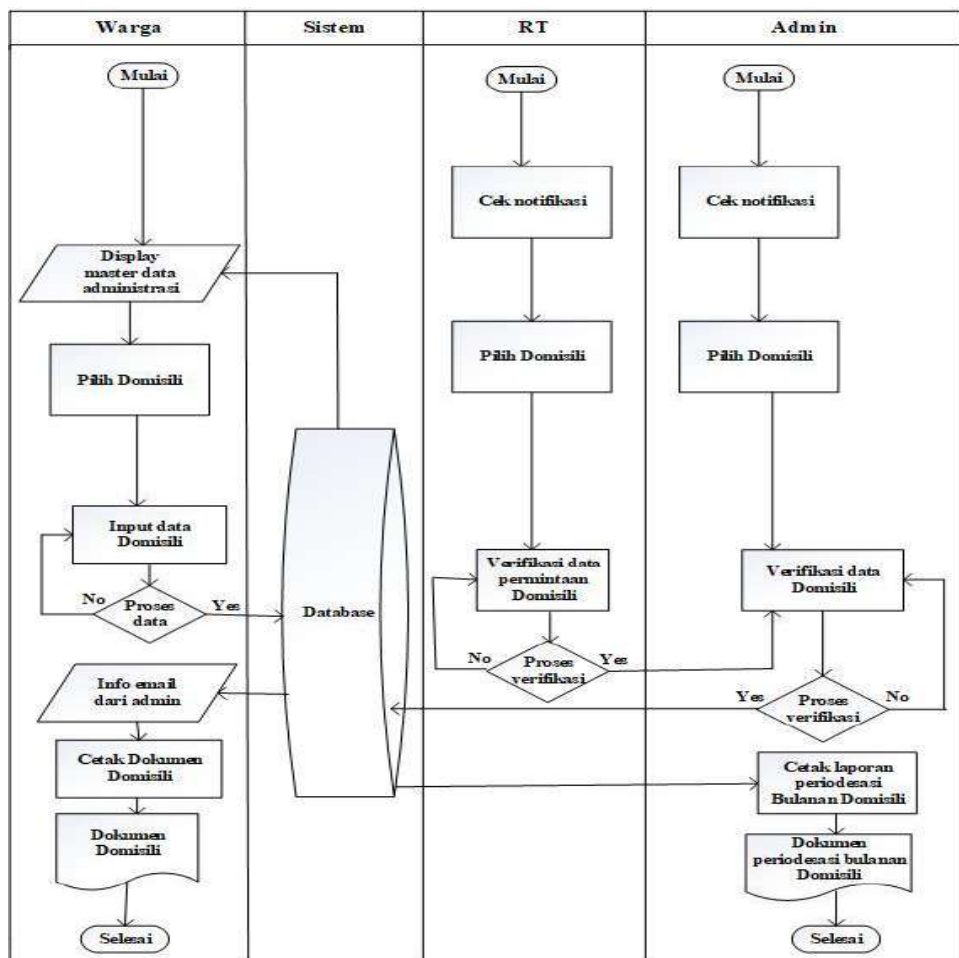
pengurusan rekomendasi KK yang diminta dan tekan simpan. Selanjutnya RT akan mendapatkan notifikasi lalu membuka surat keterangan pembuatan KK dan mengecek kelengkapan dari persyaratan yang dikirim. Jika sudah lengkap maka RT melakukan konfirmasi untuk diteruskan ke *admin* untuk melakukan verifikasi data KK untuk dikirimkan ke warga untuk dicetak dokumen administrasinya.



Gambar 3.4 *Flowchart* Surat Pengantar Kartu Keluarga (KK).

3. Surat Keterangan Domisili

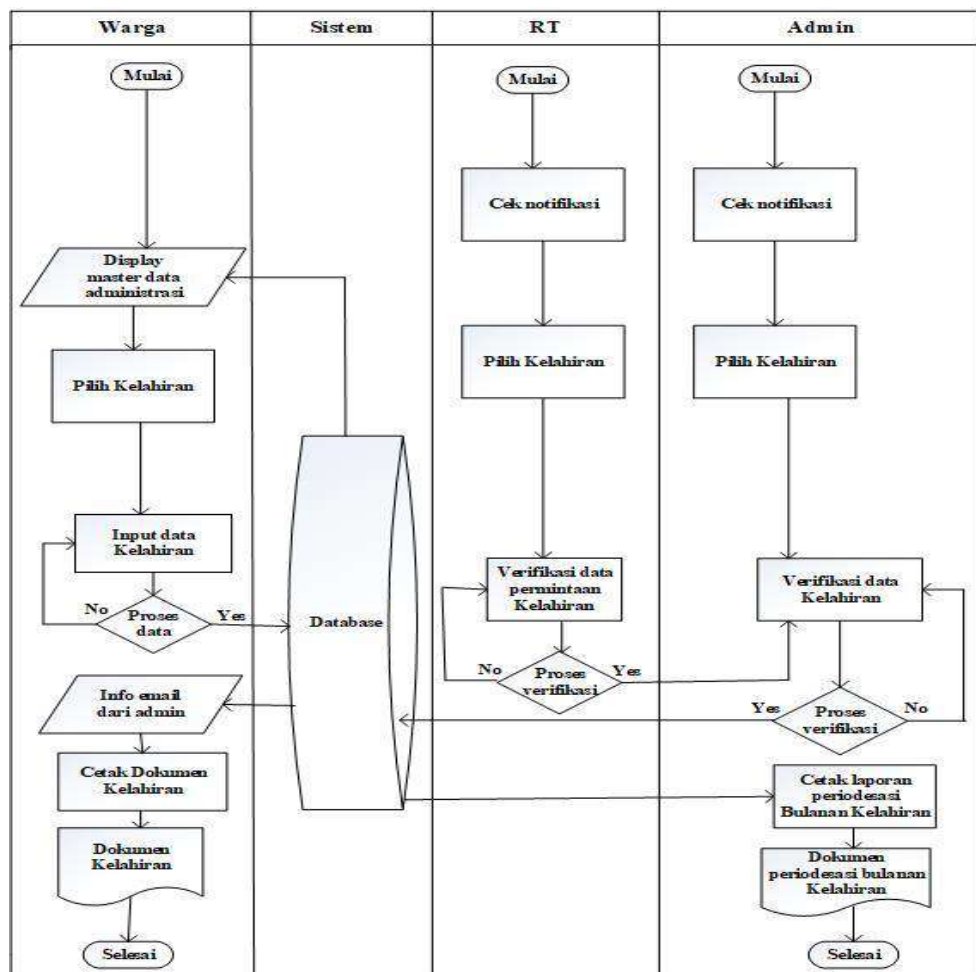
Pada Gambar 3.5, warga masuk ke tampilan master data administrasi lalu memilih menu domisili untuk meng-*input* persyaratan pengurusan domisili yang diminta dan tekan simpan. Selanjutnya RT akan mendapatkan notifikasi lalu membuka menu domisili dan mengecek kelengkapan dari persyaratan yang dikirim. Jika sudah lengkap, maka RT melakukan konfirmasi untuk diteruskan ke *admin* untuk melakukan verifikasi data domisili untuk dikirimkan ke warga untuk dicetak dokumen administrasinya.



Gambar 3.5 Surat Keterangan Domisili.

4. Surat Keterangan Kelahiran

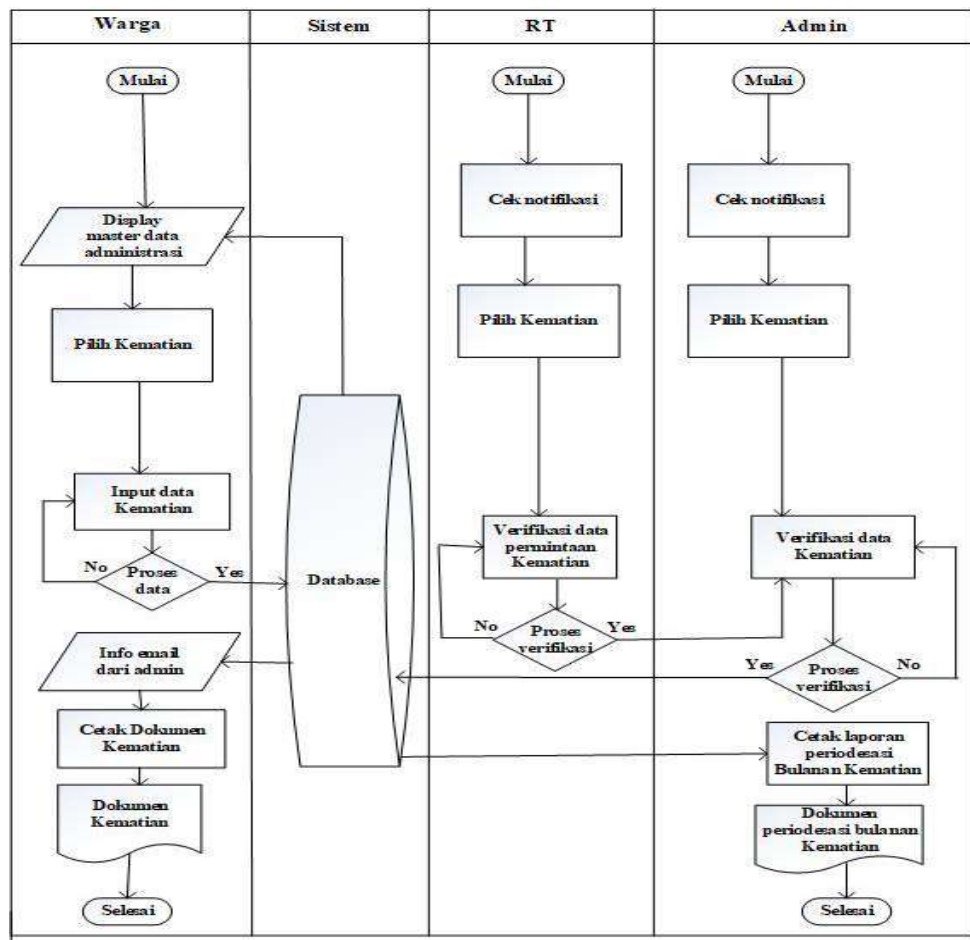
Pada Gambar 3.6, warga masuk ke tampilan master data administrasi lalu memilih menu kelahiran untuk menginput persyaratan pengurusan kelahiran yang diminta dan tekan simpan. Selanjutnya RT akan mendapatkan notifikasi lalu membuka menu kelahiran dan mengecek kelengkapan dari persyaratan yang dikirim. Jika sudah lengkap, maka RT melakukan konfirmasi untuk diteruskan ke *admin* untuk melakukan verifikasi data kelahiran untuk dikirimkan ke warga untuk dicetak dokumen administrasinya.



Gambar 3.6 Flowchart Surat Keterangan Kelahiran.

5. Surat Keterangan Kematian

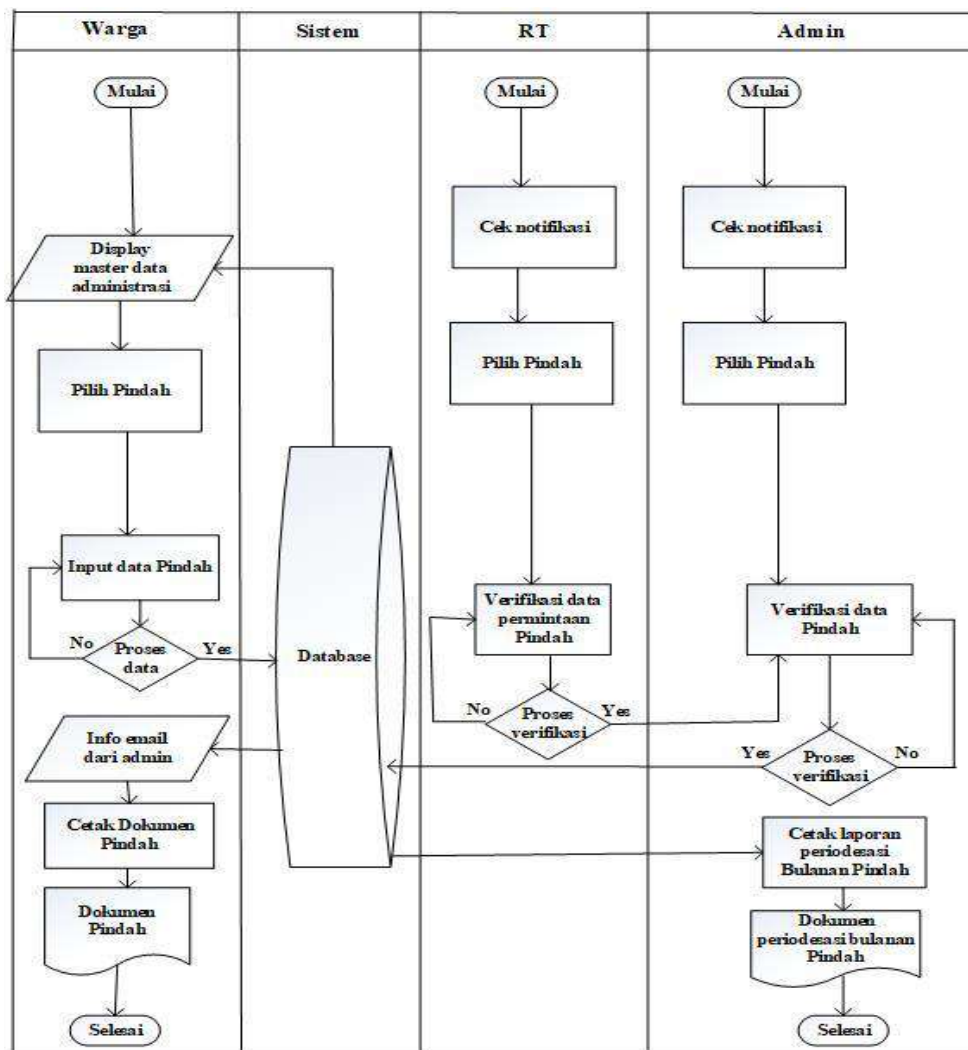
Pada Gambar 3.7, warga masuk ke tampilan master data administrasi lalu memilih menu kematian untuk menginput persyaratan pengurusan kematian yang diminta dan tekan simpan. Selanjutnya RT akan mendapatkan notifikasi lalu membuka menu kematian dan mengecek kelengkapan dari persyaratan yang dikirim. Jika sudah lengkap, maka RT melakukan konfirmasi untuk diteruskan ke *admin* untuk melakukan verifikasi data kematian untuk dikirimkan ke warga untuk dicetak dokumen administrasinya.



Gambar 3.7 Flowchart Surat Keterangan Kematian.

6. Surat Keterangan Pindah Penduduk

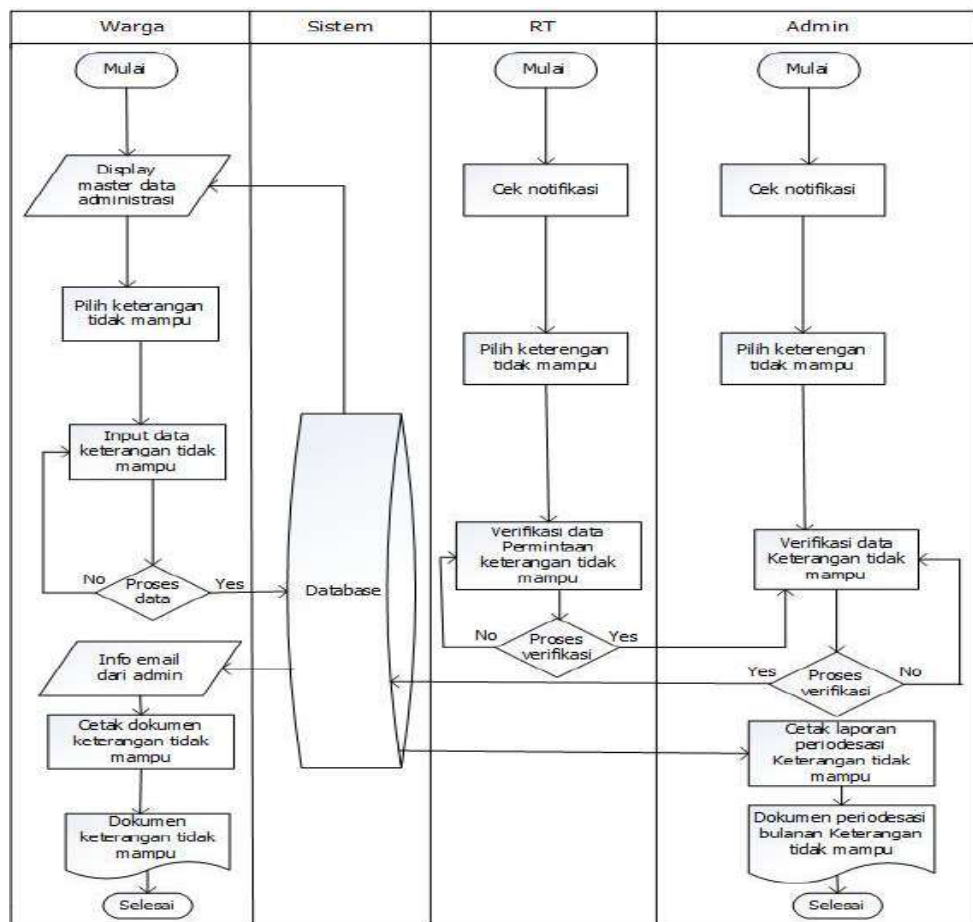
Pada Gambar 3.8, warga masuk ke tampilan master data administrasi lalu memilih menu pindah untuk menginput persyaratan pengurusan pindah yang diminta dan tekan simpan. Selanjutnya RT akan mendapatkan notifikasi lalu membuka menu pindah dan mengecek kelengkapan dari persyaratan yang dikirim. Jika sudah lengkap, maka RT melakukan konfirmasi untuk diteruskan ke *admin* untuk melakukan verifikasi data pindah untuk dikirimkan ke warga untuk dicetak dokumen administrasinya.



Gambar 3.8 Surat Keterangan Pindah.

7. Surat Keterangan Tidak Mampu

Pada Gambar 3.8, warga masuk ke tampilan master data administrasi lalu memilih menu keterangan tidak mampu untuk menginput persyaratan pengurusan surat yang diminta dan tekan simpan. Selanjutnya RT akan mendapatkan notifikasi lalu membuka menu keterangan tidak mampu dan mengecek kelengkapan dari persyaratan yang dikirim. Jika sudah lengkap, maka RT melakukan konfirmasi untuk diteruskan ke *admin* untuk melakukan verifikasi data keterangan tidak mampu untuk dikirimkan ke warga untuk dicetak dokumen administrasinya.



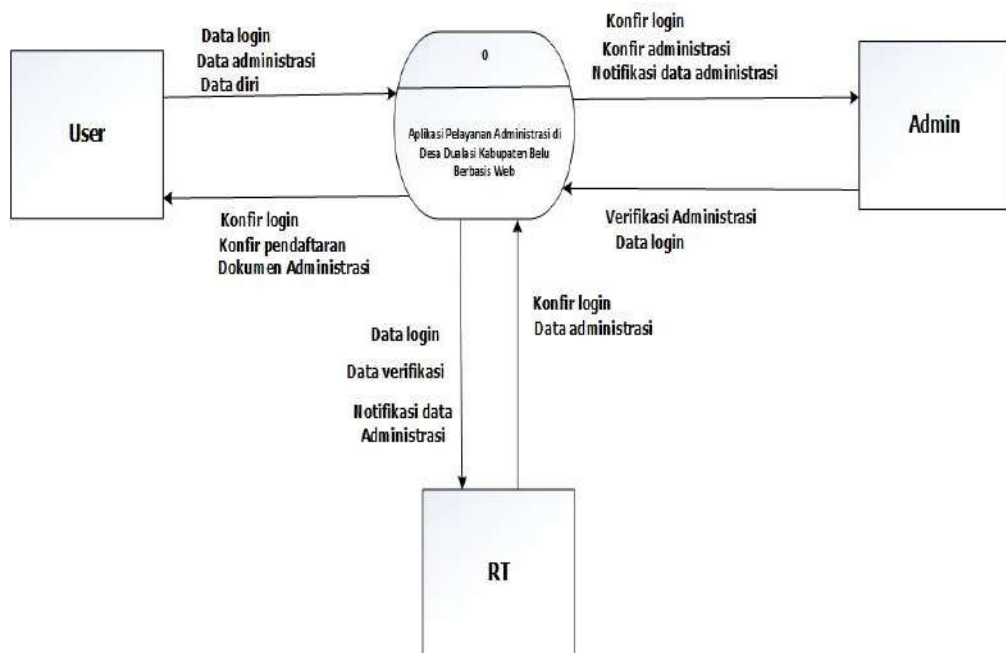
Gambar 3.9 Flowchart Surat Keterangan Tidak Mampu.

3.3.3 Diagram Alir Data (DAD)

3.3.3.1 Diagram konteks

Diagram konteks merupakan level tertinggi dari *diagram* arus data (DAD) yang menggambarkan hubungan sistem dengan lingkungannya. *Diagram* ini menunjukkan secara umum hubungan dari *input*, proses dan *output*.

Berikut adalah *diagram* konteks dari sistem :



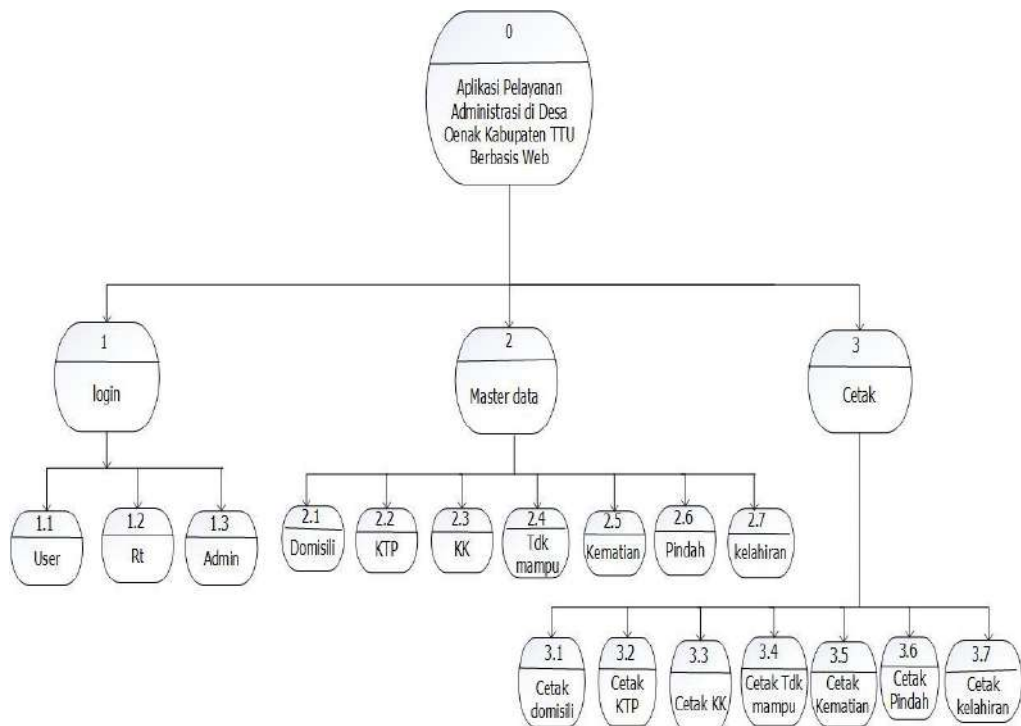
Gambar 3.10 Diagram Konteks.

Berdasarkan *diagram* konteks di atas, *user* memberikan data *login*, data administrasi dan data diri kepada sistem lalu sistem memberikan data *login* dan data verifikasi kepada RT, setelah itu RT memberikan data administrasi *user* yang sudah terverifikasi untuk diteruskan pada *admin*. *Admin* menerima data *login* dan konfirmasi data administrasi untuk diproses dan memberikan kembali verifikasi data administrasi pada sistem untuk

dicetak menjadi dokumen administrasi kepada *user*.

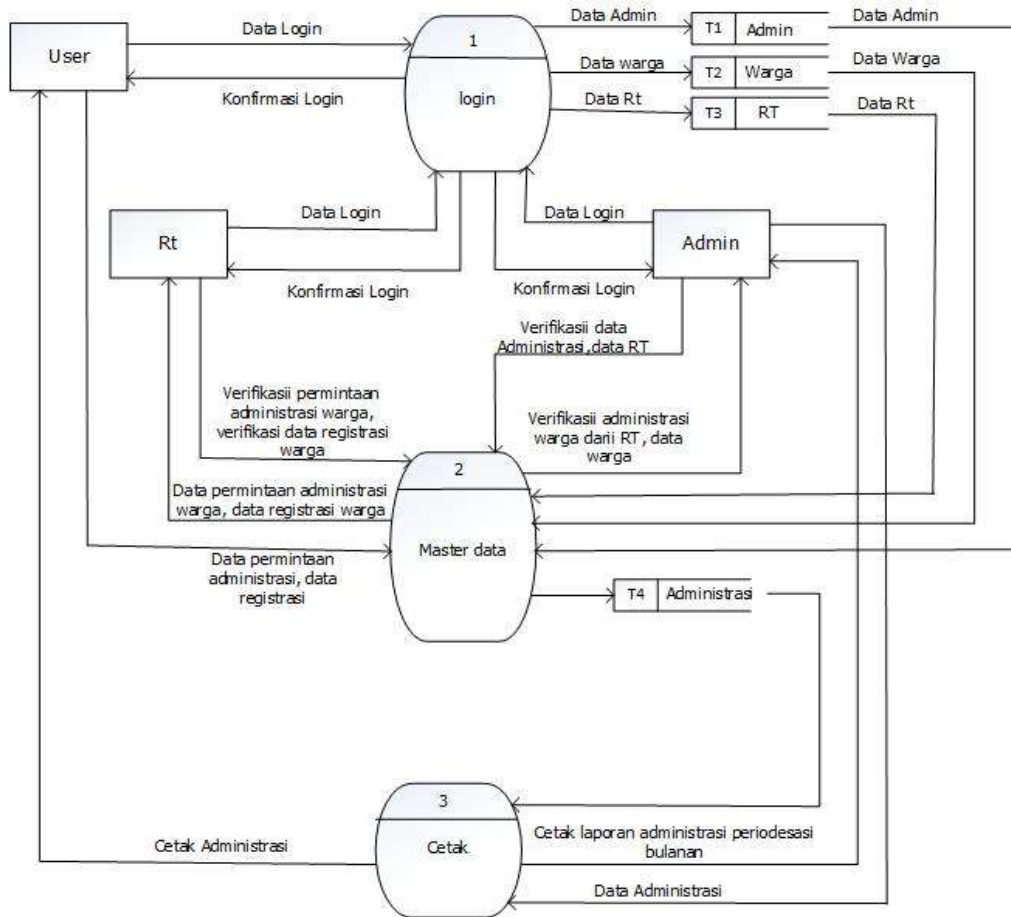
3.3.3.2 Diagram berjenjang

Diagram berjenjang adalah *diagram* yang digunakan untuk mempersiapkan penggambaran *diagram* arus data ke level - level bawah. *Diagram* berjenjang dapat digambarkan dengan menggunakan notasi proses pada data *flow diagram*. Adapun *diagram* berjenjang pada sistem yang dapat dilihat pada Gambar di bawah ini.



Gambar 3.11 *Diagram* Berjenjang.

3.3.3.3 Diagram alir data level 1

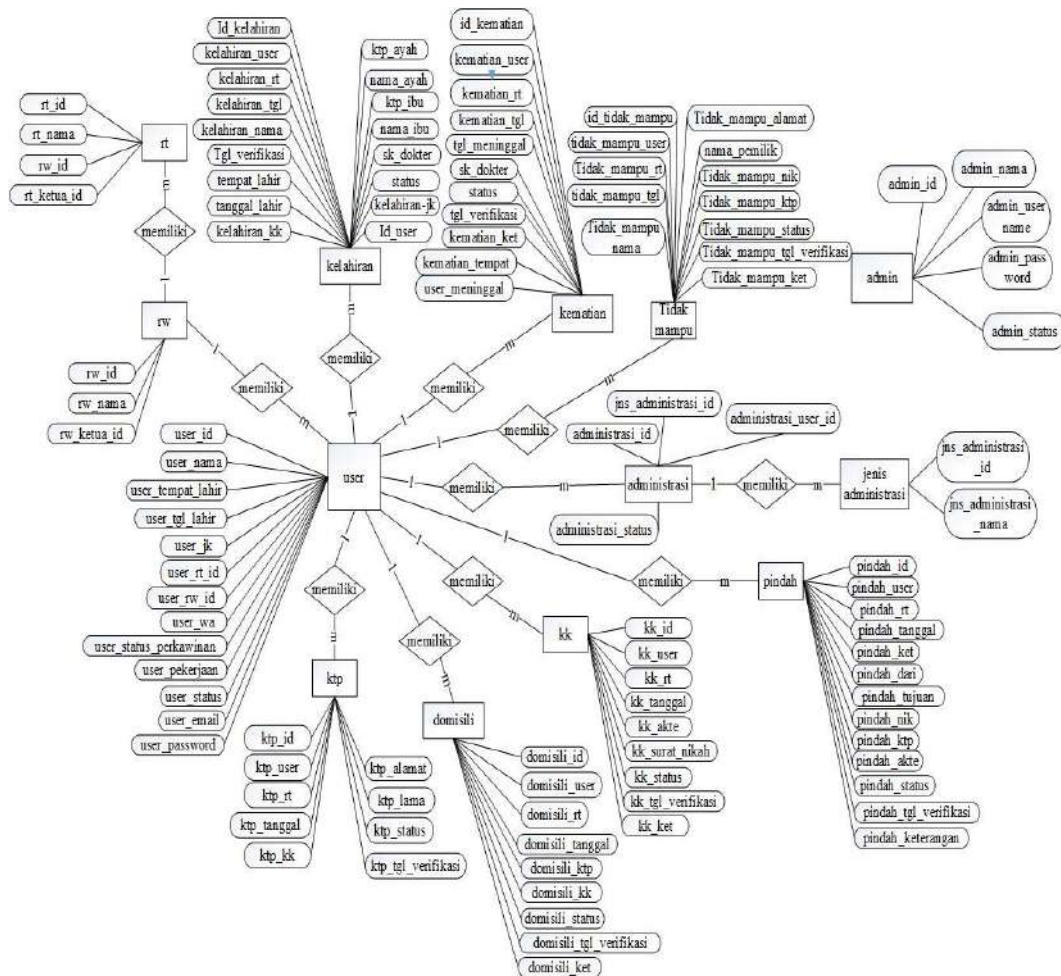


Gambar 3.12 Diagram Alir Data level 1.

Berdasarkan *diagram* di atas, warga, admin dan RT akan melakukan *login*, kemudian warga mengajukan permintaan administrasi yang dibutuhkan. *Admin* bertugas mengelolah data RT serta memverifikasi data administrasi warga untuk dicetak.

3.3.4 Entity Relationship Diagram (ERD)

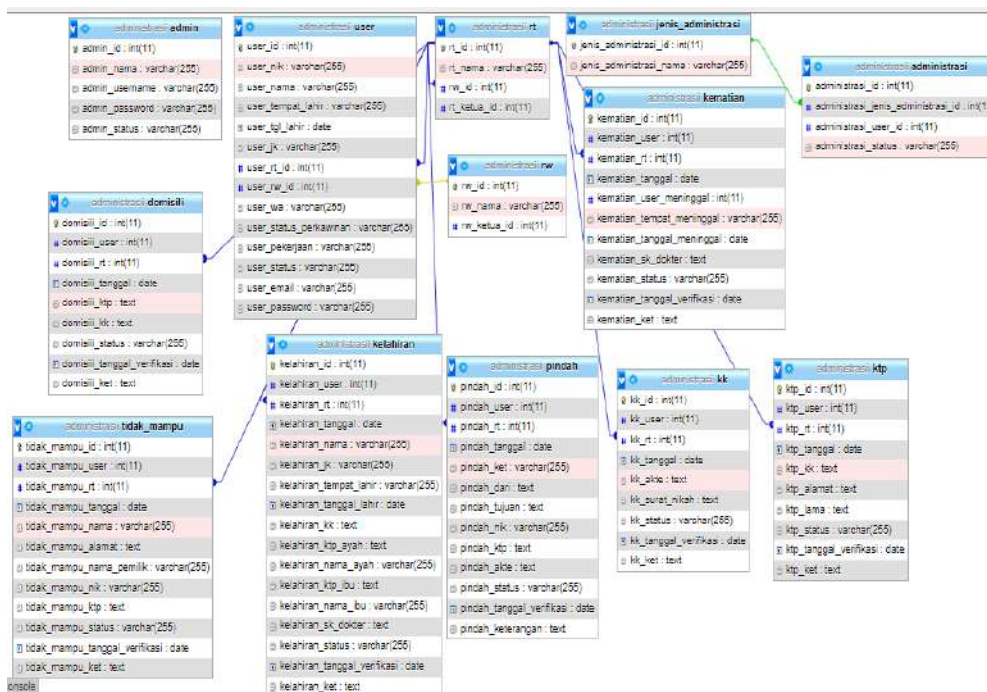
Entity Relationship Diagram (ERD) merupakan suatu model untuk menjelaskan hubungan antar data dalam basis data berdasarkan objek-objek dasar data yang mempunyai hubungan antar relasi. Pada ERD sistem ini terdiri dari 13 entitas yaitu entitas admin, entitas administrasi, entitas domisili, entitas tidak_mampu, tabel jenis_administrasi, entitas kelahiran, entitas kematian, entitas KK, entitas KTP, entitas pindah, entitas RT, entitas RW dan entitas user. Adapun Entity Relation Diagram (ERD) pada sistem yang dapat dilihat pada Gambar di bawah ini.



Gambar 3.13 Entity Relation Diagram (ERD).

3.3.5 Relasi Antar Tabel

Relasi merupakan hubungan yang terjadi pada satu tabel dengan tabel lainnya yang mempresentasikan hubungan antar obyek di dunia nyata dan berfungsi untuk mengatur operasi suatu *database*. Adapun Relasi Antar Tabel dapat dilihat pada Gambar berikut.



Gambar 3.14 Relasi Antar Tabel.

3.3.6 Perancangan Tabel

Dalam merancang *Database* suatu aplikasi perlu adanya perancangan tabel *Database* sebagai berikut :

1. Tabel *admin*

Tabel *admin* memiliki 4 *field* dengan *primary key*-nya adalah *id_admin*. Tabel *admin* dapat dilihat pada Tabel 3.1.

Tabel 3.1 Tabel *admin*.

<i>Field</i>	Tipe data	<i>size</i>	ket
admin_id	<i>int</i>	2	<i>Primary Key</i>
admin_nama	<i>varchar</i>	25	
admin_username	<i>varchar</i>	25	
admin_password	<i>varchar</i>	10	
admin_status	<i>varchar</i>	25	

2. Tabel *user*

Tabel *user* memiliki 13 *field* dengan *primary key*-nya adalah *id_user*. Tabel *user* dapat dilihat pada Tabel 3.2.

Tabel 3.2 Tabel *user*.

Field	Tipe data	Size	Ket
user_id	<i>int</i>	10	<i>Primary Key</i>
User_nik	<i>Varchar</i>	16	
user_nama	<i>Varchar</i>	25	
user_tempat_lahir	<i>Varchar</i>	15	
user_tanggal_lahir	<i>date</i>		
user_Jk	<i>Varchar</i>	15	
user_rt_id	<i>int</i>	11	<i>Foreign Key</i>
user_rw_id	<i>int</i>	11	<i>Foreign Key</i>
user_wa	<i>Varchar</i>	12	
user_status_perkawinan	<i>Varchar</i>	15	
user_pekerjaan	<i>Varchar</i>	15	
user_status	<i>Varchar</i>	15	
user_email	<i>Varchar</i>	25	
user_password	<i>Varchar</i>	10	

3. Tabel RT

Tabel RT memiliki 4 *field* dengan *primary key*-nya adalah *id_rt*. Tabel ini dapat dilihat pada Tabel 3.3.

Tabel 3.3 Tabel RT.

<i>Field</i>	Tipe data	<i>Size</i>	Ket
rt_id	<i>int</i>	10	<i>Primary Key</i>
rt_nama	<i>Varchar</i>	3	
rw_id	<i>int</i>	10	<i>Foreign Key</i>
rt_ketua_id	<i>int</i>	10	

4 Tabel RW

Tabel Rw memiliki 3 *field* dengan *primary key*-nya adalah id_rw.

Tabel ini dapat dilihat pada Tabel 3.4 di bawah ini.

Tabel 3.4 Tabel RW.

<i>Field</i>	Tipe data	<i>Size</i>	Ket
rw_id	<i>int</i>	10	<i>Primary Key</i>
rw_nama	<i>Varchar</i>	3	
rw_ketua_id	<i>int</i>	10	

5 Tabel kartu keluarga (KK)

Tabel kartu keluarga (KK) memiliki 9 *field* dengan *primary key*-nya adalah id_kk. Tabel ini dapat dilihat pada Tabel 3.5.

Tabel 3.5 Tabel KK.

<i>Field</i>	Tipe data	<i>Size</i>	Ket
kk_id	<i>int</i>	10	<i>Primary Key</i>
kk_user	<i>int</i>	10	<i>Foreign Key</i>
kk_rt	<i>int</i>	10	<i>Foreign Key</i>
kk_tanggal	<i>date</i>		
kk_akte	<i>text</i>		
kk_surat_nikah	<i>tex</i>		
kk_status	<i>varchar</i>	15	
kk_tanggal_verifikasi	<i>date</i>		
kk_ket	<i>text</i>		

6 Tabel Kartu Tanda Penduduk (KTP)

Tabel Kartu Tanda Penduduk (KTP) memiliki 10 *field* dengan *primary key*-nya adalah id_ktp. Tabel ini dapat dilihat pada Tabel 3.6.

Tabel 3.6 Tabel KTP.

<i>Field</i>	<i>Tipe data</i>	<i>Size</i>	<i>Ket</i>
ktp_id	<i>int</i>	10	<i>Primary Key</i>
ktp_user	<i>int</i>	10	<i>Foreign Key</i>
ktp_rt	<i>int</i>	10	<i>Foreign Key</i>
ktp_tanggal	<i>date</i>		
ktp_kk	<i>text</i>		
ktp_alamat	<i>text</i>		
ktp_lama	<i>text</i>		
ktp_status	<i>varchar</i>	15	
ktp_tgl_verifikasi	<i>date</i>		
ktp_ket	<i>text</i>		

7. Tabel Pindah

Tabel pindah memiliki 13 *field* dengan *primary key*-nya adalah *id_pindah*. Tabel ini dapat dilihat pada Tabel 3.7.

Tabel 3.7 Tabel Pindah.

<i>Field</i>	<i>Tipe data</i>	<i>Size</i>	<i>Ket</i>
pindah_id	<i>int</i>	10	<i>Primary Key</i>
pindah_user	<i>int</i>	10	<i>Foreign Key</i>
pindah_rt	<i>int</i>	10	<i>Foreign Key</i>
pindah_tanggal	<i>date</i>		
pindah_ket	<i>varchar</i>	25	
pindah_dari	<i>text</i>		
pindah_tujuan	<i>text</i>		
pindah_nik	<i>varchar</i>	16	
pindah_ktp	<i>text</i>		
pindah_akte	<i>text</i>		
pindah_status	<i>varchar</i>	15	
pindah_tanggal_verifikasi	<i>date</i>		
pindah_ket	<i>text</i>		

8. Tabel Tidak Mampu

Tabel tidak mampu memiliki 13 *field* dengan *primary key*-nya adalah *id_tidak_mampu*. Tabel ini dapat dilihat pada Tabel 3.8.

Tabel 3.8 Tabel Tidak Mampu.

<i>Field</i>	Tipe data	<i>Size</i>	<i>Ket</i>
tidak_mampu_id	<i>Int</i>	10	<i>Primary Key</i>
tidak_mampu_user	<i>varchar</i>	25	
tidak_mampu_rt	<i>Int</i>	10	<i>Foreign Key</i>
tidak_mampu_tanggal	<i>Date</i>		
tidak_mampu_nama	<i>varchar</i>	20	
tidak_mampu_alamat	<i>Text</i>		
tidak_mampu_nama_pemilik	<i>varchar</i>	25	
tidak_mampu_nik	<i>varchar</i>	16	
tidak_mampu_ktp	<i>Text</i>		
tidak_mampu_status	<i>varchar</i>	15	
tidak_mampu_tanggal_verifikasi	<i>date</i>		
tidak_mampu_ket	<i>Text</i>		

9. Tabel domisili

Tabel domisili memiliki 9 *field* dengan *primary key*-nya adalah *id_domisili*. Tabel ini dapat dilihat pada Tabel 3.9.

Tabel 3.9 Tabel Domisili.

<i>Field</i>	Tipe data	<i>Size</i>	<i>Ket</i>
domisili_id	<i>int</i>	10	<i>Primary Key</i>
domisili_user	<i>int</i>	10	<i>Foreign Key</i>
domisili_rt	<i>int</i>	10	<i>Foreign Key</i>
domisili_tanggal	<i>date</i>		
domisili_ktp	<i>text</i>		
domisili_kk	<i>text</i>		
domisili_status	<i>varchar</i>	15	
domisili_tanggal_verifikasi	<i>date</i>		
domisili_ket	<i>text</i>		

10. Tabel kematian

Tabel kematian ini memiliki 11 *field* dengan *primary key*-nya adalah *id_kematian*. Tabel ini dapat dilihat pada Tabel 3.10.

Tabel 3.10 Tabel Kematian.

<i>Field</i>	<i>Tipe data</i>	<i>Size</i>	<i>Ket</i>
kematian_id	<i>int</i>	5	<i>Primary key</i>
kematian_user	<i>int</i>	10	<i>Foreign Key</i>
kematian_rt	<i>int</i>	10	<i>Foreign Key</i>
kematian_tanggal_kematian	<i>date</i>		
kematian_user_meninggal	<i>int</i>		
kematian_tempat_meninggal	<i>varchar</i>	15	
kematian_anggal_meninggal	<i>date</i>		
kematian_sk_dokter	<i>text</i>		
kematian_status	<i>varchar</i>	15	
tanggal_verifikasi	<i>date</i>		
kematian_ket	<i>text</i>		

11. Tabel Kelahiran

Tabel kelahiran memiliki 17 *field* dengan *primary key*-nya adalah *id_kelahiran*. Tabel ini dapat dilihat pada Tabel 3.11.

Tabel 3.11 Tabel Kelahiran.

<i>Field</i>	<i>Tipe data</i>	<i>Size</i>	<i>Ket</i>
kelahiran_id	<i>Int</i>	5	<i>Primary Key</i>
kelahiran_user	<i>Int</i>	10	<i>Foreign Key</i>
kelahiran_rt	<i>int</i>	10	<i>Foreign Key</i>
kelahiran_tanggal	<i>date</i>		
kelahiran_nama	<i>varchar</i>	25	
kelahiran_jk	<i>varchar</i>	15	
kelahiran_tempat_lahir	<i>varchar</i>	15	
kelahiran_tgl_lahir	<i>date</i>		
kelahiran_kk	<i>Text</i>		
kelahiran_ktp_ayah	<i>Text</i>		
kelahiran_nama_ayah	<i>varchar</i>	25	
kelahiran_ktp_ibu	<i>Text</i>		
kelahiran_nama_ibu	<i>varchar</i>	25	
kelahiran_sk_dokter	<i>Text</i>		
kelahiran_status	<i>varchar</i>	15	
kelahiran_tgl_verifikasi	<i>Date</i>		
kelahiran_ket	<i>Text</i>		

12. Tabel Administrasi

Tabel administrasi memiliki 3 *field* dengan *primary key*-nya adalah *id_administrasi*. Tabel ini dapat dilihat pada Tabel 3.12.

Tabel 3.12 Tabel Administrasi.

<i>Field</i>	<i>Tipe data</i>	<i>Size</i>	<i>Ket</i>
<i>administrasi_id</i>	<i>int</i>	10	<i>Primary Key</i>
<i>administrasi_jenis_administrasi_id</i>	<i>int</i>	10	<i>Foreign Key</i>
<i>administrasi_user_id</i>	<i>int</i>	10	<i>Foreign Key</i>
<i>administrasi_status</i>	<i>varchar</i>	15	

13. Tabel Jenis Administrasi

Tabel jenis administrasi memiliki 2 *field* dengan *primary key*-nya adalah *id_jns_administrasi*. Tabel ini dapat dilihat pada Tabel 3.13.

Tabel 3.13 Tabel Jenis Administrasi.

<i>Field</i>	<i>Tipe data</i>	<i>Size</i>	<i>Ket</i>
<i>jns_administrasi_id</i>	<i>int</i>	10	<i>Primary Key</i>
<i>jns_administrasi_nama</i>	<i>varchar</i>	25	

3.3.6 Perancangan Antarmuka

Perancangan antarmuka dibuat agar pengguna dan computer dapat saling berinteraksi sehingga pengguna merasakan adanya kemudahan dari sistem komputer. Adapun desain *interface* dari *web* administrasi adalah sebagai berikut :

1. Desain Halaman Utama



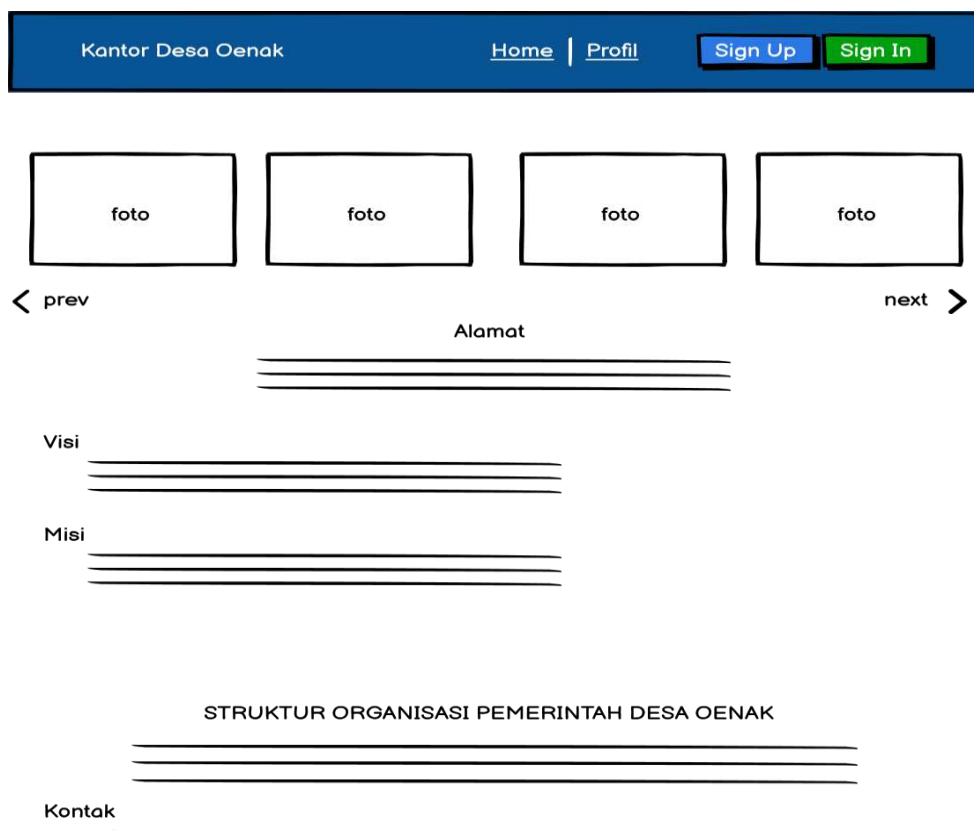
Welcome . .

Kantor Desa Oenak

Gambar 3.15 Desain Halaman Utama.

Pada halaman utama terdapat menu *home*, *profil*, *sign-up* dan *sign-in* serta ucapan selamat datang kantor desa Oenak.

2. Desain Halaman Profil



Gambar 3.16 Desain Halaman Profil.

Pada halaman halaman profil terdapat beberapa informasi berupa foto, alamat kantor, visi dan misi, struktur organisasi serta info kontak.

3. Desain halaman *sign-up*

Daftar

Nama Lengkap

Tempat Tanggal lahir

Jenis Kelamin

Alamat

Nomor Whatsapp

Status Perkawinan

Pekerjaan

Email Address

Password

Gambar 3.17 Desain halaman *sign-up*.

Pada desain halaman *sign-up* terdapat menu registrasi atau pendaftaran *user*, bagi *user* yang belum terdaftar.

4. Desain halaman *sign in*

Login

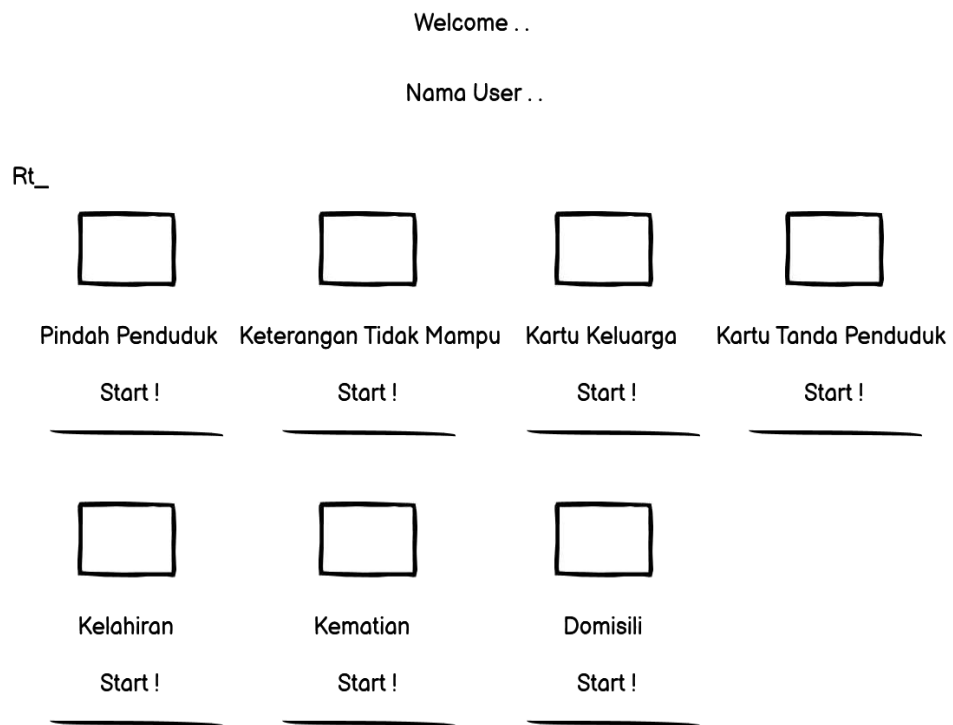
Email / Telepon

Password

Gambar 3.18 Desain Halaman *Sign-In*.

Pada desain halaman ini RT dan warga (warga yang sudah berstatus aktif) diminta untuk *login* dengan memasukkan *email* dan *password* akun.

5. Desain Halaman Warga



Gambar 3.19 Desain Halaman Warga.

Pada desain halaman warga terdapat menu *home*, *progres* dan *logout*. Menu *progress* berisikan data-data administrasi *user*. Pada halaman ini juga terdapat nama RT dan ucapan selamat datang kepada *user* yang sedang aktif. Bagian bawah halaman *user* tersedia jenis-jenis layanan administrasi yang siap untuk diakses oleh warga.

6. Desain halaman RT



Gambar 3.20 Desain Halaman RT.

Desain halaman RT berisi menu *home*, pendaftaran, administrasi, warga dan *logout*. Pada halaman ini juga terdapat ucapan selamat datang untuk ketua RT. Pada bagian bawah halaman RT terdapat tabel data warga RT, baik itu warga yang aktif maupun yang masih dinonaktifkan oleh RT.

7. Tampilan Permintaan Warga

Rt_ Home Pendaftaran Administrasi Warga [logout](#)

Data Rt:_Rw:_

Pemberitahuan! data sedang menunggu verifikasi X

Kartu tanda penduduk

Kartu keluarga

Domisili

Pindah penduduk

Kelahiran

Kematian

Keterangan tidak mampu

Gambar 3.21 Tampilan Permintaan Warga.

Desain tampilan permintaan warga berisi notifikasi dari warga kepada RT bahwa ada proses pemesanan administrasi yang harus segera diperiksa dan diverifikasi oleh RT.

8. Desain Tampilan *Login Admin*

Login

Username

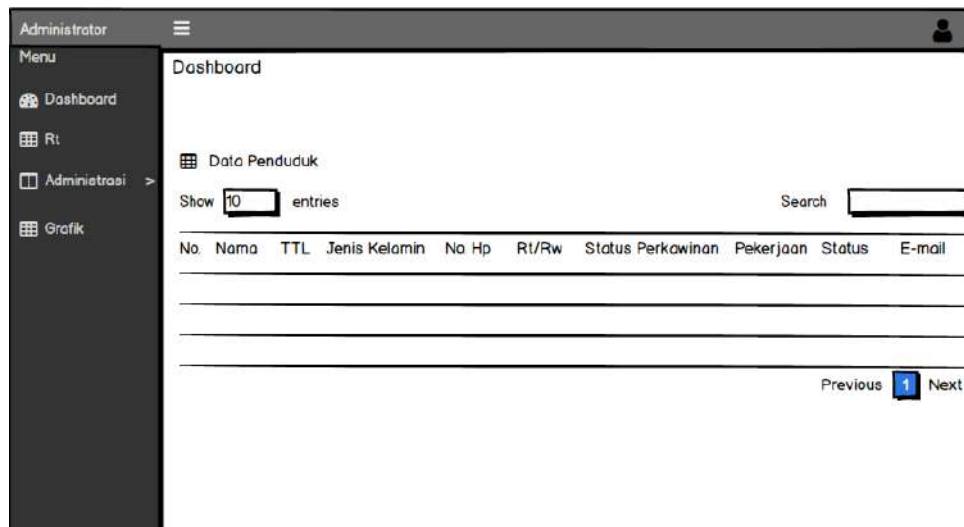
Password

Login

Gambar 3.22 Desain Tampilan *Login Admin*.

Pada halaman ini *admin* akan melakukan proses *login* ke sistem dengan memasukkan *username* dan *password*.

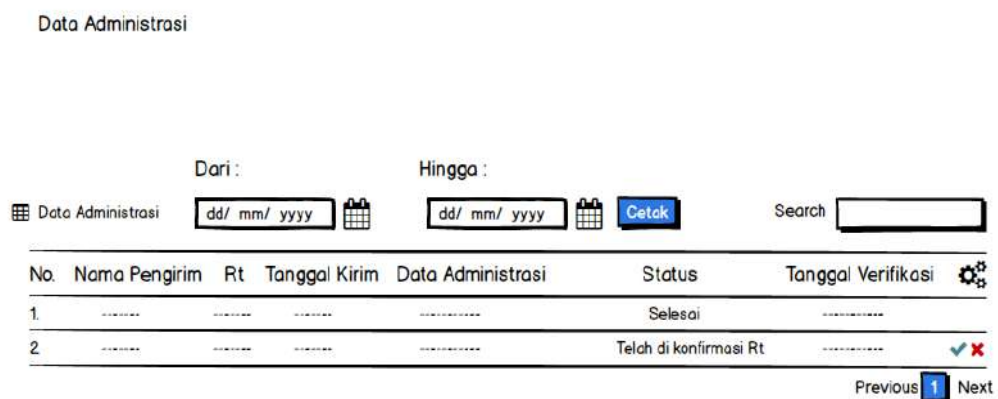
9. Desain Halaman *Admin*



Gambar 3.23 Desain Halaman *Admin*.

Halaman *admin* berisi tabel data penduduk pada tampilan *dashboard*, data RT, data administrasi dan berisi grafik pelayanan administrasi.

10. Tampilan Verifikasi Administrasi *Admin*



Gambar 3.24 Tampilan Verifikasi Administrasi *Admin*.

Setelah RT menyetujui data administrasi dari warga maka sistem akan meruskan data tersebut ke *admin* untuk melakukan proses verifikasi

agar bisa di cetak oleh warga yang bersangkutan menjadi sebuah dokumen administrasi.

11. Halaman Cetak Laporan Administrasi



Gambar 3.25 Halaman Cetak Laporan Administrasi.

Halaman ini berisikan data administrasi yang sudah selesai diproses oleh *admin*, sehingga warga dapat mencetaknya menjadi sebuah dokumen administrasi yang siap digunakan.