LAPORAN PRAKTEK KERJA LAPANGAN

(SIPENDUK)

SISTEM INFORMASI PENGELOLAAN DATA PENDUDUK

KANTOR KELURAHAN KARANG BARU BERBASIS WEBSITE

Disusun oleh:

NURUL NADIYATUN SHOLIHAH

(F1D017069)

PROGRAM STUDI INFORMATIKA

FAKULTAS TEKNIK

UNIVERSITAS MATARAM

2020

LEMBAR PENGESAHAN

LAPORAN PRAKTEK KERJA LAPANGAN

SISTEM INFORMASI PENGELOLAAN DATA PENDUDUK KANTOR KELURAHAN KARANG BARU BERBASIS WEBSITE

Disusun oleh:

NURUL NADIYATUN SHOLIHAH
F1D017069

Telah Disetujui oleh: Tanggal:

1. Dosen Pembimbing 1. 10 Juli 2020



Ariyan Zubaidi, S.Kom., M.T.
NIP. 19860913 201504 1 001

2. Pembimbing Lapangan 2. 9 Juli 2020



Isaskar Diri
NIP. 19641024 198603 1 022

Mengetahui:
Sekretaris Program Studi Teknik Informatika
Fakultas Teknik
Universitas Mataram



Andy Hidayat Jatmika, S.T., M.Kom.
NIP. 198311302000031001

KATA PENGANTAR

Puji syukur penulis panjatkan kepada Tuhan Yang Maha Esa, karena dengan berkat, rahmat dan limpahan karunianya sehingga penulis dapat menyelesaikan Laporan Praktek Kerja Lapangan ini sebagaimana mestinya.

Adapun Praktek Kerja Lapangan (PKL) ini penulis laksanakan di Kantor Lurah Karang Baru, Kecamatan Selaparang, Kota Mataram, Nusa Tenggara Barat dengan judul "(SIPENDUK) Sistem Informasi Pengelolaan Data Penduduk Kantor Kelurahan Karang Baru Berbasis Website". Sistem Informasi ini dibuat untuk mempermudah pengelolaan data kependudukan pada Kantor Lurah Karang Baru. Hal ini meliputi pengelolaan data penduduk, data kelahiran, data kematian dan pembuatan beberapa jenis surat keterangan sehingga dengan adanya sistem informasi ini diharapkan pengelolaan data pada Kantor Lurah karang Baru menjadi lebih terorganisir. Dalam pembuatan laporan ini penulis berpedoman pada bahan kuliah, petunjuk dari pembimbing lapangan, dosen pembimbing, referensi dan literatur yang terkait dengan penulisan laporan. Tidak lupa pula penulis mengucapkan terima kasih atas bantuan dari semua pihak yang telah membantu dalam penyelesaian laporan ini.

Karena keterbatasan pengetahuan maupun pengalaman, penulis menyadari laporan ini masih jauh dari kesempurnaan. Oleh karena itu, penulis mengharapkan kritik serta saran yang bersifat membangun agar dapat menghasilkan karya yang lebih baik dimasa mendatang. Harapan penulis semoga laporan ini dapat menambah pengetahuan dan pengalaman bagi penulis dan pembaca sekalian. Terima kasih.

Mataram, 9 Juli 2020



Penulis

UCAPAN TERIMA KASIH

Pada kesempatan ini penulis tidak lupa mengucapkan terima kasih yang sebesar-besarnya kepada:

1. Allah Subhanahu Wa Ta'ala
2. Keluarga dan rekan-rekan yang telah memberikan dukungan baik materil maupun do'a kepada penulis.
3. Bapak Ariyan Zubaidi, S.Kom., M.T., selaku Dosen Pembimbing Praktek Kerja Lapangan.
4. Bapak Isaskar Diri, selaku Pembimbing Lapangan serta Kasi Sosial dan Pemberdayaan Masyarakat Kantor Lurah Karang Baru.
5. Semua pihak yang telah membantu penulis yang tidak dapat penulis sebutkan satu persatu yang telah membantu selama pelaksanaan PKL ini.

DAFTAR ISI

LEMBAR PENGESAHAN ii
KATA PENGANTAR iii
UCAPAN TERIMA KASIH..... iv
DAFTAR ISI v
DAFTAR GAMBAR vii
DAFTAR TABEL..... xi
BAB I PENDAHULUAN.....1
1.1 Latar Belakang 1
1.2 Rumusan Masalah 2
1.3 Batasan Masalah.....2
1.4 Tujuan 3
1.5 Manfaat 3
BAB II TINJAUAN INSTANSI TEMPAT PKL 4
2.1 Profil Singkat Kantor Lurah Karang Baru..... 4
2.2 Visi dan Misi Kelurahan Karang Baru 4
2.3 Fungsi Kelurahan Karang Baru 5
2.4 Struktur Organisasi Kelurahan Karang Baru..... 5
BAB III LANDASAN TEORI.....6
3.1 Sistem Informasi 6
3.2 *Database*.....6
3.3 Aplikasi *Web* 7
3.4 *Web Server* 7
3.5 Pemodelan Aplikasi dan Sistem Perangkat lunak 7
3.5.1 *Use case* Diagram 8
3.5.2 *Activity* Diagram 9
3.6 ERD (*Entity Relationship* Diagram).....10
3.7 PHP.....10
3.8 *Visual Studio Code* 11
3.9 *Framewrok CodeIgniter* dan Konsep M-V-C 11
3.10 XAMPP 12
3.11 MySQL.....12

BAB IV PEMBAHASAN.....13
4.1 Metode Pengembangan Sistem 13
4.2 Desain Sistem.....14
4.2.1 *Use case* Diagram 14
4.2.2 *Activity* Diagram 18
4.2.3 *Entity Relationship* Diagram (ERD) 52
4.3 Implementasi Sistem 56
4.3.1 Implementasi *Database* Sistem.....57
4.3.2 Implementasi *Interface* Sistem 62
4.4 Pengujian Sistem 101
BAB V PENUTUP.....103
5.1 Kesimpulan.....103
5.2 Saran 104
DAFTAR PUSTAKA 105

DAFTAR GAMBAR

Gambar 2. 1 Bagan struktur organisasi Kelurahan Karang Baru 5
Gambar 4. 1 Tahapan pengembangan perangkat lunak metode *waterfall* 14
Gambar 4. 2 *Use case* Diagram..... 14
Gambar 4. 3 *Activity* diagram untuk proses *login* admin..... 18
Gambar 4. 4 *Activity* diagram untuk proses melihat *dashboard* 19
Gambar 4. 5 *Activity* diagram untuk proses melihat halaman beranda 19
Gambar 4. 6 *Activity* diagram untuk proses melihat data penduduk 20
Gambar 4. 7 *Activity* diagram untuk proses menambahkan data penduduk..... 21
Gambar 4. 8 *Activity* diagram untuk proses pencarian data penduduk 22
Gambar 4. 9 *Activity* diagram untuk proses pembaruan data penduduk 23
Gambar 4. 10 *Activity* diagram untuk proses penghapusan data penduduk 24
Gambar 4. 11 *Activity* diagram untuk proses melihat detail informasi data penduduk 25
Gambar 4. 12 *Activity* diagram untuk proses melihat data kelahiran 25
Gambar 4. 13 *Activity* diagram untuk proses menambahkan data kelahiran 26
Gambar 4. 14 *Activity* diagram untuk proses pencarian data kelahiran 27
Gambar 4. 15 *Activity* diagram untuk proses pembaruan data kelahiran..... 28
Gambar 4. 16 *Activity* diagram untuk proses penghapusan data kelahiran..... 29
Gambar 4. 17 *Activity* diagram untuk proses melihat detail informasi data kelahiran 30
Gambar 4. 18 *Activity* diagram untuk proses melihat data kematian 30
Gambar 4. 19 *Activity* diagram untuk proses menambahkan data kematian 31
Gambar 4. 20 *Activity* diagram untuk proses pencarian data kematian 32
Gambar 4. 21 *Activity* diagram untuk proses memperbarui data kematian..... 33
Gambar 4. 22 *Activity* diagram untuk proses penghapusan data kematian 34
Gambar 4. 23 *Activity* diagram untuk proses melihat detail informasi data kematian 35
Gambar 4. 24 *Activity* diagram untuk proses melihat data layanan surat..... 35
Gambar 4. 25 *Activity* diagram untuk proses menambahkan data surat..... 36
Gambar 4. 26 *Activity* diagram untuk proses pencarian data surat..... 37
Gambar 4. 27 *Activity* diagram untuk proses pembaruan data surat 38

Gambar 4. 28 *Activity* diagram untuk proses penghapusan data surat 39
Gambar 4. 29 *Activity* diagram untuk proses pencetakan data surat 40
Gambar 4. 30 *Activity* diagram untuk proses melihat halaman data pengaturan... 40
Gambar 4. 31 *Activity* diagram untuk proses pencarian data pengaturan 41
Gambar 4. 32 *Activity* diagram untuk proses memperbarui data pengaturan 42
Gambar 4. 33 *Activity* diagram untuk proses *logout* 43
Gambar 4. 34 *Activity* diagram untuk proses *login* sekretaris 44
Gambar 4. 35 *Activity* diagram untuk proses melihat beranda 45
Gambar 4. 36 *Activity* diagram untuk proses melihat halaman data penduduk..... 45
Gambar 4. 37 *Activity* diagram untuk proses pencarian data penduduk 46
Gambar 4. 38 *Activity* diagram untuk proses melihat detail informasi data penduduk 47
Gambar 4. 39 *Activity* diagram untuk proses melihat data kelahiran..... 47
Gambar 4. 40 *Activity* diagram untuk proses pencarian data kelahiran 48
Gambar 4. 41 *Activity* diagram untuk proses melihat detail informasi data kelahiran 48
Gambar 4. 42 *Activity* diagram untuk proses melihat halaman data kematian 49
Gambar 4. 43 *Activity* diagram untuk proses pencarian data kematian 50
Gambar 4. 44 *Activity* diagram untuk proses melihat detail informasi data kematian 51
Gambar 4. 45 *Activity* diagram untuk proses *logout* 51
Gambar 4. 46 ERD sistem informasi pengelolaan data penduduk Kantor Kelurahan Karang Baru 52
Gambar 4. 47 Implementasi tabel domisili 57
Gambar 4. 48 Implementasi tabel kelahiran 57
Gambar 4. 49 Implementasi tabel kematian..... 58
Gambar 4. 50 Implementasi tabel pejabat..... 58
Gambar 4. 51 Implementasi tabel penduduk..... 59
Gambar 4. 52 Implementasi tabel sktm..... 59
Gambar 4. 53 Implementasi tabel surat_kelahiran..... 60
Gambar 4. 54 Implementasi tabel surat_kematian 60
Gambar 4. 55 Implementasi tabel usaha 61
Gambar 4. 56 Implementasi tabel *user* 61
Gambar 4. 57 Implementasi halaman *login user* 62
Gambar 4. 58 Implementasi halaman *dashboard* 62

Gambar 4. 59 Implementasi halaman beranda 63
Gambar 4. 60 Implementasi halaman data penduduk 64
Gambar 4. 61 Implementasi halaman tambah data penduduk 65
Gambar 4. 62 Implementasi halaman edit data penduduk 66
Gambar 4. 63 Implementasi halaman hapus data penduduk 67
Gambar 4. 64 Implementasi halaman informasi detail data penduduk 67
Gambar 4. 65 Implementasi halaman data kelahiran 68
Gambar 4. 66 Implementasi halaman tambah data kelahiran 69
Gambar 4. 67 Implementasi halaman edit data kelahiran 70
Gambar 4. 68 Implementasi halaman hapus data kelahiran 70
Gambar 4. 69 Implementasi halaman informasi detail data kelahiran 71
Gambar 4. 70 Implementasi halaman data kematian 72
Gambar 4. 71 Implementasi halaman tambah data kematian 73
Gambar 4. 72 Implementasi halaman edit data kematian 74
Gambar 4. 73 Implementasi halaman hapus data kematian 74
Gambar 4. 74 Implementasi halaman informasi detail data kematian 75
Gambar 4. 75 Implementasi halaman data layanan surat kelahiran 75
Gambar 4. 76 Implementasi halaman tambah data surat kelahiran 76
Gambar 4. 77 Implementasi halaman edit data surat kelahiran 77
Gambar 4. 78 Implementasi halaman hapus data surat kelahiran 78
Gambar 4. 79 Implementasi halaman cetak data surat kelahiran 78
Gambar 4. 80 Implementasi halaman data layanan surat kematian 79
Gambar 4. 81 Implementasi halaman tambah data surat kematian 80
Gambar 4. 82 Implementasi halaman edit data surat kematian 81
Gambar 4. 83 Implementasi halaman hapus data surat kematian 81
Gambar 4. 84 Implementasi halaman cetak data surat kematian 82
Gambar 4. 85 Implementasi halaman data layanan surat keterangan domisili 83
Gambar 4. 86 Implementasi halaman tambah data surat keterangan domisili 84
Gambar 4. 87 Implementasi halaman edit data surat keterangan domisili 85
Gambar 4. 88 Implementasi halaman hapus data surat keterangan domisili 85
Gambar 4. 89 Implementasi halaman cetak data surat keterangan domisili 86
Gambar 4. 90 Implementasi halaman data layanan surat keterangan tidak mampu 87
Gambar 4. 91 Implementasi halaman tambah data surat keterangan tidak mampu 88

Gambar 4. 92 Implementasi halaman edit data surat keterangan tidak mampu.... 89
Gambar 4. 93 Implementasi halaman hapus data surat keterangan tidak mampu 89
Gambar 4. 94 Implementasi halaman cetak data surat keterangan tidak mampu 90
Gambar 4. 95 Implementasi halaman data layanan surat keterangan usaha..... 91
Gambar 4. 96 Implementasi halaman tambah data surat keterangan usaha..... 92
Gambar 4. 97 Implementasi halaman edit data surat keterangan usaha 93
Gambar 4. 98 Implementasi halaman hapus data surat keterangan usaha 93
Gambar 4. 99 Implementasi halaman cetak data surat keterangan usaha 94
Gambar 4. 100 Implementasi halaman data pengaturan..... 95
Gambar 4. 101 Implementasi halaman edit data pengaturan..... 95
Gambar 4. 102 Implementasi halaman login user..... 96
Gambar 4. 103 Implementasi halaman beranda 97
Gambar 4. 104 Implementasi halaman data penduduk 97
Gambar 4. 105 Implementasi halaman informasi detail data penduduk..... 98
Gambar 4. 106 Implementasi halaman data kelahiran..... 99
Gambar 4. 107 Implementasi halaman informasi detail data kelahiran..... 99
Gambar 4. 108 Implementasi halaman data kematian 100
Gambar 4. 109 Implementasi halaman informasi detail data kematian..... 101

DAFTAR TABEL

Tabel 3.1 Simbol *use case* diagram 8
Tabel 3.2 Simbol *activity* diagram 9
Tabel 3.3 Simbol ERD 10
Tabel 4. 1 Struktur tabel domisili 52
Tabel 4. 2 Struktur tabel kelahiran 53
Tabel 4. 3 Struktur tabel kematian 53
Tabel 4. 4 Struktur tabel pejabat 54
Tabel 4. 5 Struktur tabel penduduk 54
Tabel 4. 6 Struktur tabel sktm 55
Tabel 4. 7 Struktur tabel surat_kelahiran 55
Tabel 4. 8 Struktur tabel surat_kematian 55
Tabel 4. 9 Struktur tabel usaha 56
Tabel 4. 10 Struktur tabel *user* 56
Tabel 4. 11 Persentase hasil pengujian sistem 101

BAB I
PENDAHULUAN

1.1 Latar Belakang

Perkembangan teknologi informasi disertai dengan teknologi komputer yang canggih dalam waktu yang relatif singkat telah memberikan pengaruh yang begitu besar terhadap berbagai aspek kehidupan manusia di setiap lapisan masyarakat. Perkembangan teknologi informasi dengan penggunaan komputer dapat menghasilkan pengolahan data yang cepat, tepat dan akurat sehingga lebih efektif dan efisien. Sistem Informasi merupakan salah satu bentuk penerapan dari pesatnya laju perkembangan teknologi informasi.

Perkembangan teknologi yang sangat cepat mengharuskan instansi mengikuti perkembangan tersebut, oleh karena itu suatu instansi membutuhkan sistem informasi yang dapat mendukung kebutuhannya dalam mencapai efisiensi dan efektifitas kerja demi meningkatkan pelayanan kepada masyarakat. Dengan suatu sistem informasi data kependudukan maka diharapkan pengolahan data akan lebih mudah dan efisien.

Pengelolaan data kependudukan merupakan tanggung jawab pemerintah kabupaten / kota, dimana pelaksanaannya diawali dari kelurahan selaku instansi paling awal untuk melakukan pendaftaran penduduk. Kantor Lurah Karang Baru memiliki tugas pokok untuk menyelenggarakan urusan pemerintahan, pembangunan dan kemasyarakatan serta urusan yang dilimpahkan oleh Walikota sesuai dengan kebutuhan Kelurahan dengan memperhatikan prinsip efisiensi dan peningkatan akuntabilitas. Kantor Lurah Karang Baru saat ini belum memiliki sistem atau basis data informasi untuk melakukan pengelolaan data kependudukan sehingga masih dilakukan secara manual dimana hal ini akan memperlambat kinerja dari staf-staf Kantor Lurah Karang Baru. Pengolahan data dan pengelolaannya sebagian sudah menggunakan teknologi komputer namun belum optimal karena masih terbatas pada pengolahan kata dan angka dengan menggunakan Microsoft Word dan Microsoft Excel.

Hal ini tentu menimbulkan berbagai kendala, adapun kendala yang dihadapi adalah sulitnya dalam melakukan pencarian data bila suatu saat data tersebut diperlukan dan kesulitan dalam penyimpanan data karena masih ada data yang menggunakan sistem penyimpanan dalam bentuk berkas fisik, terjadinya ketidak-

cocokan dan redundansi data, tidak efisiennya waktu yang digunakan untuk memproses pelayanan kependudukan dan proses pembuatan surat keterangan, surat kelahiran, surat kematian, dan sebagainya.

Sehingga penulis menyimpulkan perlunya perancangan suatu sistem informasi yang dapat mengelola data penduduk secara benar dan cepat sehingga proses pengolahan dan pengelolaan data kependudukan menjadi lebih efektif dan efisien. Oleh karena itu, penulis menawarkan solusi untuk mengatasi beberapa permasalahan tersebut yaitu dengan pembuatan "Sistem Informasi Pengelolaan Data Penduduk Kantor Kelurahan Karang Baru Berbasis *Website*" yang dilengkapi dengan *database* sebagai media penyimpanan datanya. Dengan adanya sistem informasi administrasi kependudukan ini, diharapkan proses pelayanan kepada penduduk akan meningkat dari segi waktu maupun kualitas pelayanannya baik itu pendaftaran penduduk maupun pembuatan surat keterangan lainnya yang perlu dilakukan dengan benar dan cepat agar penduduk dapat merasa mendapatkan pelayanan yang memuaskan.

1.2 Rumusan Masalah

Berdasarkan latar belakang masalah yang telah dipaparkan di atas, dapat diidentifikasi beberapa rumusan masalah sebagai berikut:

1. Bagaimana membangun Sistem Informasi Pengelolaan Data Penduduk Kantor Kelurahan Karang Baru Berbasis Website yang sesuai dengan kebutuhan di Kantor Lurah Karang Baru?
2. Bagimana merancang sistem informasi yang dapat memudahkan staf-staf Kantor Lurah karang Baru dalam pengolahan dan pengelolaan data kependudukan?

1.3 Batasan Masalah

Berdasarkan rumusan masalah yang telah dipaparkan di atas, dapat ditentukan beberapa batasan masalah untuk membatasi bahasan yang akan disampaikan, agar tidak menyimpang dari apa yang dibutuhkan. Adapun batasan masalah yang akan dibahas pada laporan ini, antara lain:

1. Sistem informasi ini hanya mencakup pengelolaan data penduduk, data kelahiran dan data kematian.
2. Sistem informasi ini hanya mengakomodir layanan untuk pembuatan surat

kelahiran, surat kematian, surat keterangan domisili, surat keterangan tidak mampu dan surat keterangan usaha.

1.4 Tujuan

Adapun tujuan dari dibuatnya Sistem Informasi Pengelolaan Data Penduduk Kantor Kelurahan Karang Baru Berbasis *Website* ini adalah sebagai berikut:

1. Merancang sistem informasi pengelolaan data penduduk Kantor Kelurahan Karang Baru berbasis *website* dengan PHP MYSQL dan *Framework CodeIgniter*.
2. Merancang suatu basis data yang dapat menyimpan data-data kependudukan Kantor Lurah Karang Baru.
3. Menghasilkan informasi dan data yang akurat sehingga dapat membantu menciptakan lingkungan kerja yang produktif di Kantor Lurah Karang Baru.
4. Meningkatkan pelayanan publik bagi penduduk Kelurahan Karang Baru.

1.5 Manfaat

Adapun manfaat yang diharapkan dari Praktek Kerja Lapangan (PKL) dalam merancang sistem informasi pengelolaan data penduduk Kantor Kelurahan Karang Baru berbasis *website* yang telah dibuat di bagi menjadi dua bagian, antara lain sebagai berikut:

1. Bagi pihak instansi
    a. Menjadikan pengelolaan data kependudukan di Kantor Lurah Karang baru lebih efektif dan efisien.
    b. Meningkatkan kinerja staf-staf Kantor Lurah Karang Baru demi memberikan pelayanan terbaiknya kepada masyarakat.
    c. Mempercepat dalam pendataan data penduduk, kelahiran penduduk dan kematian penduduk.
2. Bagi mahasiswa
    a. Memberikan pengalaman kerja pada instansi bagi mahasiswa.
    b. Memberikan pengenalan lingkungan kerja bagi mahasiswa di suatu instansi.
    c. Mengimplementasikan ilmu pengetahuan dan pembelajaran yang diperoleh pada saat perkuliahan.
    d. Menambah dan meningkatkan pemahaman mahasiswa dalam pembuatan *website*.

BAB II
TINJAUAN INSTANSI TEMPAT PKL

2.1 Profil Singkat Kantor Lurah Karang Baru

Kelurahan Karang Baru, Kecamatan Selaparang Kota Mataram merupakan salah satu dari 50 (Lima Puluh) Kelurahan Kota Mataram yang berjarak dari ± 1 Km dari pusat pemerintah Kota Mataram, awal mula terbentuknya Kelurahan Karang Baru pada tahun 1980 yang terdiri dari 9 Lingkungan dan 43 RT dengan keragaman etnis dan suku, pada tahun 2007 terjadi pemekaran Kecamatan dan Kelurahan.

Luas wilayah Kelurahan Karang Baru kurang lebihnya adalah sebesar 20.37 Hektar. Wilayah Kelurahan Karang Baru terbagi menjadi Sembilan lingkungan yaitu Karang Baru Utara, Karang Baru Selatan, Marong Jamaq Utara, Marong Pekarangan, Marong Jamaq Selatan, Marong Karang Tatah, Suradadi Barat, Suradadi Timur dan Taman. Berdasarkan hasil sensus penduduk di tahun 2017, jumlah penduduk Kelurahan Karang Baru adalah sebanyak 10.212 jiwa dengan jumlah kepala keluarga sebanyak 3.014 KK. Dengan komposisi masyarakat hetrogen yang memiliki mata pencaharian yang beragam dari PNS, TNI, Polri, Swasta, Wiraswasta, Wirausaha, Jasa, Buruh, Perdagangan dan Pensiunan. Sedangkan masyarakat yang dikategorikan kurang sejahtera mendapat bantuan raskin dan jaminan kesehatan dari Pemerintah APBD/APBN. Sampai dengan tahun 2018, terdapat 118 toko dan 5 lembaga keuangan di Kelurahan Karang Baru.

2.2 Visi dan Misi Kelurahan Karang Baru

Adapun visi dan misi dari Kelurahan Karang Baru adalah sebagai berikut:

1. Visi

   Menjadi Terdepan Sebagai Pelayan Masyarakat Untuk Meningkatkan Sumber Daya Manusia.

2. Misi
   a. Menjaga kondusifitas keamanan untuk ketertiban masyarakat dengan menggiatkan sistem keamanan swakarsa.
   b. Meningkatkan kualitas pelayanan publik.
   c. Menjaga dan memelihara sarana dan prasarana wilayah.
   d. Meningkatkan kualitas Sumber Daya Manusia melalui pendidikan dan pelatihan.
   e. Menggiatkan dan memperdayakan ekonomi rakyat berbasis potensi lokal.

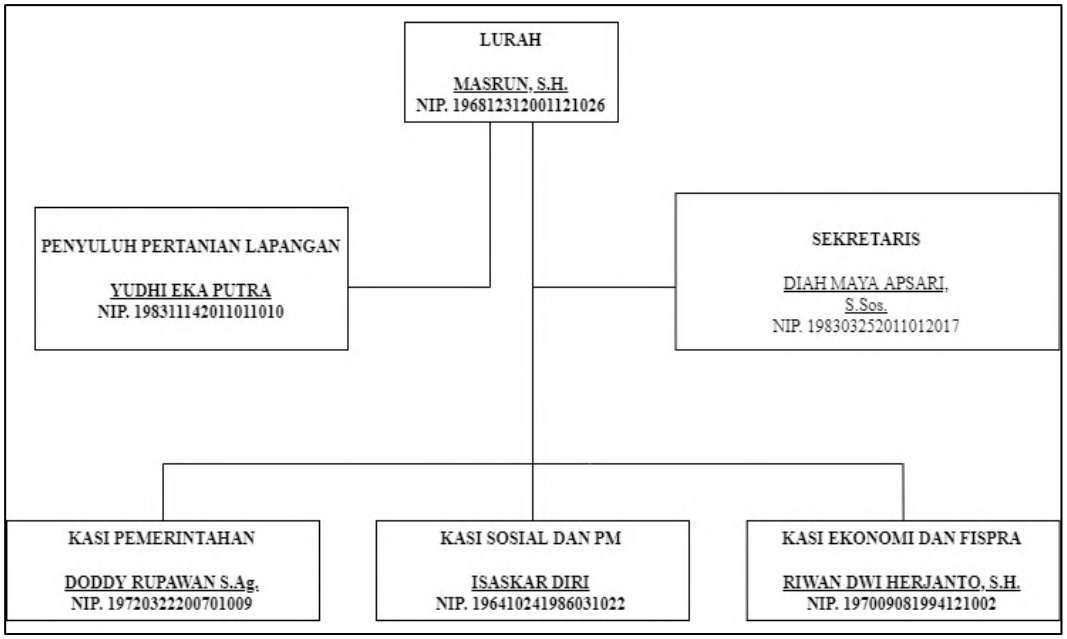
f. Menjaga dan memelihara kualitas lingkungan berbasis masyarakat.

2.3 Fungsi Kelurahan Karang Baru

a. Melaksanakan Kegiatan Pemberdayaan Masyarakat.
b. Menyelenggarakan ketertiban dan ketentraman umum.
c. Mengkoordinasikan pemeliharaan prasarana dan fasilitas umum.
d. Membina Lembaga Kemasyarakatan.
e. Membina dan mengendalikan Administrasi, Rukun Warga dan Rukun Tetangga.
f. Melaksanakan pelayanan masyarakat yang menjadi ruang lingkup tugasnya.
g. Melaksanakan tugas lain yang diberikan oleh Walikota/ Camat sesuai dengan tugas dan fungsinya.

2.4 Struktur Organisasi Kelurahan Karang Baru

Berikut ini merupakan struktur organisasi Kelurahan Karang Baru tahun 2020 dan dapat dilihat pada Gambar 2.1.



Gambar 2. 1 Bagan struktur organisasi Kelurahan Karang Baru

BAB III
LANDASAN TEORI

3.1 Sistem Informasi

Sistem informasi adalah kumpulan komponen yang saling berhubungan dalam mengumpulkan, memproses, menyimpan, menyediakan dan mendistribusikan informasi untuk mendukung pengambilan keputusan dan pengendalian di dalam organisasi [2]. Pengertian lain mengatakan bahwa sebuah Sistem Informasi merupakan sekumpulan dari perangkat keras dan perangkat lunak komputer serta perangkat manusia yang akan mengolah data menggunakan perangkat keras dan lunak tersebut. Selain itu data juga memegang peranan penting dalam sistem informasi. Data yang akan dimasukan adalah sebuah sistem infomasi yang dapat berupa formulir-formulir, prosedur-prosedur dan bentuk data lainnya [3].

Dalam memahami pengertian sistem informasi, harus diingat keterkaitan antara data dan informasi sebagai entitas penting pembentuk sistem informasi. Informasi adalah data yang telah diolah menjadi bentuk yang lebih berarti bagi penerimanya. Data adalah kenyataan yang menggambarkan kejadian-kejadian dan kesatuan nyata. Kejadian adalah sesuatu yang terjadi pada saat tertentu. Sistem informasi adalah sistem yang menyediakan informasi dengan cara sedemikian rupa sehingga bermanfaat bagi penerima [1].

Komponen-komponen sistem informasi adalah:

a. *Hardware* (perangkat keras)
b. *Software* (perangkat lunak)
c. Sekumpulan aturan yang dipakai untuk mewujudkan pemrosesan data untuk menghasilkan *output* (prosedur).
d. Suatu pengorganisasian sekumpulan data yang saling terkait sehingga memudahkan proses pencarian informasi (Basis data).
e. Jaringan komputer dan komunikasi data.
f. Brainware

3.2 Database

Database yaitu sekumpulan data yang saling berintegrasi antara satu sama lain dan terorganisasi berdasarkan sebuah skema atau struktur tertentu dan

tersimpan pada sebuah komputer. *Database* terdiri dari beberapa tabel atau lebih dari satu tabel yang saling terorganisir. Tabel berfungsi untuk menyimpan data dan terdiri dari baris dan kolom. Data yang tersimpan pada tabel tersebut diantaranya dapat ditampilkan, dimodifikasi dan dihapus. Setiap pemakai (*user*) diberi wewenang untuk dapat melakukan akses terhadap data tersebut [9].

3.3 Aplikasi *Web*

Website dapat diartikan sebagai kumpulan halaman-halaman yang digunakan untuk mempublikasikan informasi berupa teks, gambar dan program multimedia lainnya berupa animasi (gambar gerak, tulisan gerak), suara dan atau gabungan dari semuanya itu baik yang bersifat statis maupun dinamis yang membentuk suatu rangkaian bangunan yang saling terkait antara satu halaman dengan halaman yang lain yang sering disebut sebagai *hyperlink* [7]. Terdapat dua jenis *website* yaitu *website* statis dan *website* dinamis. Pada website statis komunikaksi berjalan secara satu arah dari *server* ke *client* sedangkan website dinamis komunikasi berjalan secara dua arah dimana *user* bisa memasukan data tertentu ke dalam halaman web untuk kemudian diproses oleh *server* dan menghasilkan *output* yang diinginkan oleh *user* [8].

3.4 Web Server

Web server merupakan *software* dalam *server* yang berfungsi untuk menerima permintaan *request* berupa halaman web melalui HTTP atau HTTPS dari klien yang dikenal dengan *web browser* dan mengirimkan kembali hasilnya dalam bentuk halaman-halaman web yang umumnya berbentuk dokumen HTML [7]. Beberapa web server yang digunakan di internet antara lain:

a. Apache Web Server (Windows dan *Linux)*
b. Internet Information Services (Windows)
c. Xitami web server
d. Sun Java system web server

3.5 Pemodelan Aplikasi dan Sistem Perangkat lunak

Unified Modeling Language (UML) merupakan salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek [12].

UML menawarkan sebuah standar untuk merancang model sebuah sistem. Dengan menggunakan UML dapat membuat model untuk semua jenis aplikasi piranti lunak, dimana aplikasi tersebut dapat berjalan pada piranti keras, sistem operasi dan jaringan apapun, serta ditulis dalam bahasa pemrograman apapun [11]. Berikut merupakan beberapa diagram yang terdapat pada UML

3.5.1 *Use case* Diagram

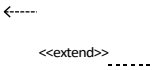
Use case adalah gambaran fungsionalitas dari suatu sistem, sehingga pengguna sistem paham dan mengerti mengenai kegunaan sistem yang akan dibangun. *Use case* digunakan untuk memodelkan dan menyatakan unit fungsi atau layanan yang disediakan oleh sistem ke pemakai.

Use case diagram dapat sangat membantu dalam menyusun *requirement* sebuah sistem, mengkomunikasikan rancangan dengan *client*, dan merancang *testcase* untuk semua *feature* yang ada pada sistem.

Sebuah *use case* dapat meng-*include* fungsionalitas *use case* lain sebagai bagian dari proses dalam dirinya. Sebuah *use case* juga dapat meng-*extend use case* lain dengan *behaviour*-nya sendiri. Sementara hubungan generalisasi antar *use case* menunjukkan bahwa *use case* yang satu merupakan spesialisasi dari yang lain [11]. Simbol penggunaan *use case* diagram ditunjukan pada tabel 3.1 berikut:

Tabel 3.1 Simbol *use case* diagram

Simbol	Nama	Keterangan
	*Actor*	*Actor* tersebut mempresentasikan seseorang atau sesuatu (seperti perangkat, sistem lain) yang berinteraksi dengan sistem.
	*Use case*	Gambaran fungsionalitas dari suatu sistem, sehingga *customer* atau pengguna sistem paham dan mengerti mengenai kegunaan sistem yang akan dibangun.
	Association	Menghubungkan *link* antar *element.*
	*Include*	Kejadian yang harus terpenuhi agar sebuah *event* dapat terjadi, di mana pada kondisi ini sebuah *usecase* adalah bagian dari *usecase* lainnya.

	*Extend*	Kejadian yang tidak harus terpenuhi agar sebuah *event* dapat terjadi, di mana pada kondisi ini sebuah kejadian dapat dilakukan atau tidak.
	Generalization	Disebut juga *inheritance* (pewarisan).

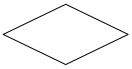
3.5.2 *Activity* Diagram

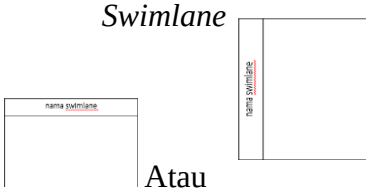
Activity diagram yaitu memberikan gambaran berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, dan bagaimana mereka berakhir. *Activity* diagram juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi.

Sebuah aktivitas dapat direalisasikan oleh satu *use case* atau lebih. Aktivitas menggambarkan proses yang berjalan, sementara *use case* menggambarkan bagaimana aktor menggunakan sistem untuk melakukan aktivitas [11].

Berikut tabel 3.2 adalah simbol-simbol yang ada pada diagram aktivitas :

Tabel 3.2 Simbol *activity* diagram

Simbol	**Deskripsi**
Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
Percabangan / *Decision* 	Asosiasi percabangan di mana jika ada pilihan aktivitas lebih dari satu.
Penggabungan / *Join*	Asosiasi penggabungan di mana lebih dari satu aktivitas digabungkan menjadi satu.
Status Akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas harus memiliki status akhir.

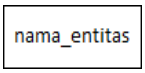
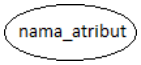
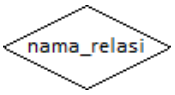
Swimlane  Atau	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.

3.6 ERD (*Entity Relationship* Diagram)

ERD (*Entity Relationship* Diagram*)* adalah sebuah diagram yang digunakan dalam merancang hubungan antar tabel-tabel dalam basis data. ERD (*Entity Relation* diagram) berfungsi untuk menggambarkan gambaran dari dunia nyata yang akan diterapkan pada suatu *database* di sebuah sistem [9].

Pada ERD terdapat beberapa simbol dengan cara penulisan di antaranya ditunjukan pada tabel 3.3 berikut:

Tabel 3.3 Simbol ERD

Simbol	Nama	Keterangan
 	Entitas / *Entitiy*	Merupakan data inti yang akan disimpan; bakal tabel pada basis data; biasanya mengacu pada benda yang terlibat dalam aplikasi yang akan dibuat.
 	Atribut	*Field* atau kolom data yang butuh disimpan dalam satu entitas.
 	Atribut kunci primer	*Field* atau kolom data yang butuh disimpan dalam suatu entitas sebagai kunci akses *record* yang diinginkan.
 	Relasi	Penghubung antar entitas; biasanya diawali dengan kata kerja.
	Asosiasi / *Association*	Penghubung antara relasi dan entitas di mana di kedua ujungnya memiliki *multiplicity* kemungkinan jumlah pemakaian.

3.7 PHP

PHP merupakan software *Open Source* yang digunakan sebagai bahasa *script server-side* dalam pengembangan *Web* yang disisipkan pada dokumen HTML [13].

PHP (*Hypertext Preprocessor*) disebut sebagai bahasa pemrograman server side karena diproses pada komputer server saja. Hal inilah yang membedakan PHP

dengan bahasa pemrograman *client-side* seperti JavaScript. Dalam pembuatan *web*, *script* PHP umumnya disisipkan pada dokumen HTML, karena kemampuan inilah PHP juga sering disebut sebagai bahasa pemrograman *script* atau *Scripting Language*. Dalam hal ini, dengan menggunakan PHP, pembuatan *web* dimudahkan dalam membuat konten di dalam sebuah halaman karena lebih efektif. Misalnya, untuk melakukan pengulangan secara lebih efektif, meng-*input* database, dan sebagainya [10].

3.8 Visual Studio Code

Visual Studio Code adalah editor *source code* yang dikembangkan oleh Microsoft untuk Windows, Linux dan MacOS. *Visual Studio Code* juga dapat digunakan sebagai *editor source code* berbagai bahasa pemrograman dari PHP, Java, Python, Javascript, Node.js, C/C++, Go, C# sampai editor untuk *script web* seperti HTML dan CSS. Fitur – fitur yang dimiliki *Visual Studio Code* antara lain adalah fitur *intellisense*, fitur *debugging*, GIT Control, fitur ekstensi, dan sebagainya.

3.9 Framework CodeIgniter dan Konsep M-V-C

CodeIgniter merupakan sebuah *framework* php yang bersifat *open source* dan menggunakan metode MVC (*Model, View, Controller*) untuk memudahkan developer atau programmer dalam membangun sebuah aplikasi berbasis web tanpa harus membuatnya dari awal. Sedangkan framework adalah kumpulan intruksi-intruksi yang dikumpulkan dalam *class* dan *function-function* dengan fungsi masing-masing untuk memudahkan developer dalam memanggilnya tanpa harus menuliskan *syntax* program yang sama berulang-ulang serta dapat menghemat waktu [6].

Codeigniter menggunakan konsep M-V-C (*Model-View-Controller*) yang memungkinkan pemisahan antara *layer application-logic* dan *presentation*. Dengan konsep ini kode PHP, *query* Mysql, Javascript dan CSS dapat saling dipisah-pisahkan sehingga ukuran file menjadi lebih kecil dan lebih mudah dalam perbaikan kedepannya atau *maintenance*. Berikut merupakan penjelasan dari komponen M-V-C [6]:

a. *Model* merupakan kode program (berupa OOP *class*) yang digunakan untuk berhubungan dengan *database* MySQL sekaligus untuk memanipulasinya (*input-edit-delete*).

b. *View* merupakan kode program berupa *template* atau PHP untuk menampilkan data pada *browser*.
c. *Controller* merupakan Kode program (berupa OOP *class*) yang digunakan untuk mengontrol aliran atau dengan kata lain sebagai pengontrol *model* dan *view*.

3.10 XAMPP

XAMPP adalah perangkat lunak bebas, yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program. Fungsinya adalah sebagai *server* yang berdiri sendiri (*localhost*), yang terdiri atas program Apache HTTP *Server*, MySQL *database*, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl. Nama XAMPP merupakan singkatan dari X (empat sistem operasi apapun), Apache, MySQL, PHP dan Perl. Program ini tersedia dalam GNU *General Public License* dan bebas, merupakan *web server* yang mudah digunakan yang dapat melayani tampilan halaman web yang dinamis [5].

3.11 MySQL

MySQL adalah sebuah program *Database server* yang mampu menerima dan mengirimkan data dengan cepat dengan menggunakan perintah – perintah SQL (*Structured Query Language*). MySQL merupakan *database* yang pertama kali didukung oleh bahasa pemrograman *script* untuk internet (PHP dan Perl). MySQL lebih sering digunakan untuk membangun aplikasi berbasis web, umumnya pengembangan aplikasinya menggunakan bahasa pemrograman *script* PHP [4].

Di dalam MySQL, terdapat 3 subbahasa yaitu *Data Definition Language* (DDL), *Data Manipulation Language* (DML), dan *Data Control Language* (DCL). DDL digunakan untuk membangun objek – objek dalam *database*, seperti tabel dan *index*. DML digunakan untuk menambah, mencari, menghapus, dan mengubah baris dalam tabel. DCL digunakan untuk menangani masalah keamanan dalam *database* [14].

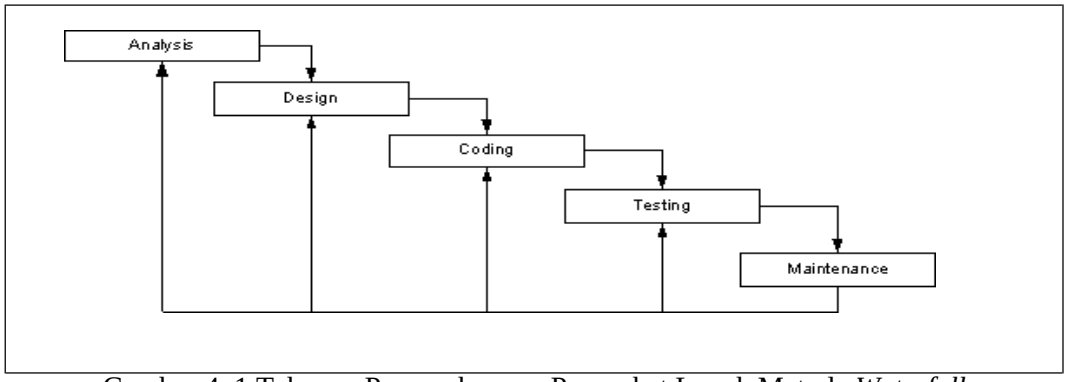
BAB IV
PEMBAHASAN

4.1 Metode Pengembangan Sistem

Pengembangan perangkat lunak Sistem Informasi Pengelolaan Data Penduduk Kantor Kelurahan Karang Baru Berbasis Website ini menggunakan SDLC (*System Development Life Cycle*) memiliki beberapa model dalam dengan model *waterfall* (*linear sequential model*). Alasan penggunaan model *waterfall* dalam pembuatan sistem informasi ini dikarenakan sistem yang dibuat tidak terlalu besar dan lingkup pengembangannya tidak begitu luas sehingga akan lebih memudahkan jika menggunakan model *waterfall* yang pendekatannya dilakukan secara sistematis dan berurutan.

Secara garis besar, metode pengembangan perangkat lunak dengan metode *waterfall* dibagi menjadi lima tahapan utama. Tahap-tahap tersebut yaitu diantaranya analisis kebutuhan sistem, perancangan sistem, implementasi sistem, pengujian sistem dan *maintenance.*

Pada tahap pertama adalah mengidentifikasi permasalahan kemudian menguraikan rincian kebutuhan sistem. Lalu pada tahap selanjutnya kebutuhan perangkat lunak dari tahap analisis sebelumnya akan diterjemahkan ke represantasi perancangan sistem sehingga dapat diimplementasikan menjadi program di tahap berikutnya sehingga penting untuk mendokumentasikan rancangan yang dihasilkan pada tahap ini. Kemudian, hasil perancangan diterjemahkan ke dalam bahasa pemrograman sehingga menghasilkan perangkat lunak yang sesuai dengan rancangan yang telah dibuat. Setelah sistem selesai, maka selanjutnya perlu dilakukan pengujian sistem untuk memastikan bahwa kebutuhan-kebutuhan sistem sudah terpenuhi dan bisa berfungsi dengan sempurna sesuai dengan tujuan pembuatannya. Tahapan terakhir merupakan *maintenance* dimana tidak menutupi kemungkinan perangkat lunak yang telah dibuat mengalami perubahan dikarenakan harus beradaptasi dengan lingkungan baru. Alur kerja dari metode pengembangan perangkat lunka dengan metode *waterfall* dapat dilihat pada gambar 4.1 berikut:

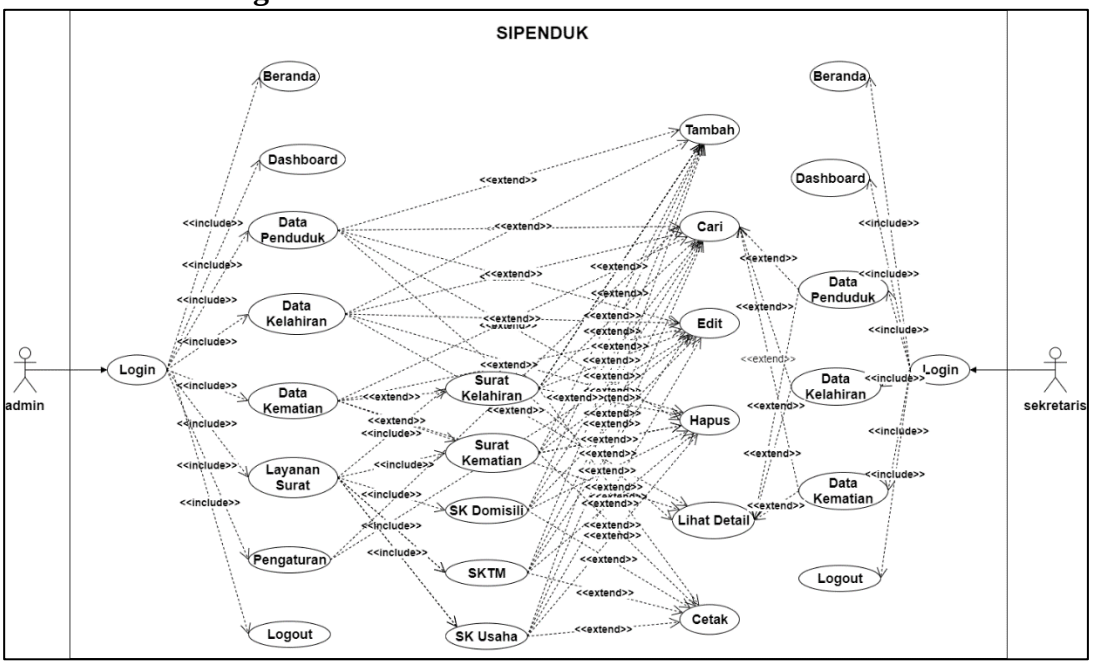


Gambar 4. 1 Tahapan Pengembangan Perangkat Lunak Metode *Waterfall*

4.2 Desain Sistem

Tahap perancangan Sistem Informasi Pengelolaan Data Penduduk Kantor Kelurahan Karang Baru Berbasis *Website* ini menggunakan tiga diagram UML seperti yang telah dijelaskan sebelumnya, yang terdiri dari *use case*, *activity*, serta ERD.

4.2.1 *Use case* Diagram



Gambar 4. 2 *Use case* Diagram

Gambar 4.2 merupakan diagram *use case* pada sistem informasi pengelolaan data penduduk Kantor Kelurahan Karang Baru. Diagram di atas menunjukkan bahwa ada 2 level pengguna yang dapat menggunakan sistem ini, yaitu staf administrasi dan sekretaris kelurahan. *Use case diagram* yang disajikan pada Gambar 4.2 mendeskripsiskan interaksi aktor, yaitu admin dan sekretaris dalam pengelolaan data. Rincian aktivitas yang dapat dilakukan oleh user setelah melakukan *login* adalah sebagai berikut:

1. *User* Admin
    a. *Login*
    b. Melihat *dashboard*
    c. Melihat beranda
    d. Melihat data penduduk
        1) Menambah data penduduk
        2) Mencari data penduduk
        3) Memperbarui data penduduk
        4) Menghapus data penduduk
        5) Melihat detail informasi data penduduk
    e. Melihat data kelahiran
        1) Menambah data kelahiran
        2) Mencari data kelahiran
        3) Memperbarui data kelahiran
        4) Menghapus data kelahiran
        5) Melihat detail informasi data kelahiran
    f. Melihat data kematian
        1) Menambah data kematian
        2) Mencari data kematian
        3) Memperbarui data kematian
        4) Menghapus data kematian
        5) Melihat detail informasi data kematian
    g. Mengakses layanan surat
        1) Melihat surat kelahiran

a) Menambah surat kelahiran

b) Mencari surat kelahiran

c) Memperbarui surat kelahiran

d) Menghapus surat kelahiran

e) Mencetak surat kelahiran

2) Melihat surat kematian

a) Menambah surat kematian

b) Mencari surat kematian

c) Memperbarui surat kematian

d) Menghapus surat kematian

e) Mencetak surat kematian

3) Melihat surat keterangan domisili

a) Menambah surat keterangan domisili

b) Mencari surat keterangan domisili

c) Memperbarui surat keterangan domisili

d) Menghapus surat keterangan domisili

e) Mencetak surat keterangan domisili

4) Melihat surat keterangan tidak mampu

a) Menambah surat keterangan tidak mampu

b) Mencari surat keterangan tidak mampu

c) Memperbarui surat keterangan tidak mampu

d) Menghapus surat keterangan tidak mampu

e) Mencetak surat keterangan tidak mampu

5) Melihat surat keterangan usaha

a) Menambah surat keterangan usaha

b) Mencari surat keterangan usaha

c) Memperbarui surat keterangan usaha

d) Menghapus surat keterangan usaha

e) Mencetak surat keterangan usaha

h. Pengaturan

1) Mencari data pengaturan

2) Memperbarui data pengaturan

i. *Logout*

2. *User* Sekretaris

a. *Login*

b. Melihat beranda

c. Melihat data penduduk

1) Mencari data penduduk

2) Melihat detail informasi data penduduk

d. Melihat data kelahiran

1) Mencari data kelahiran

2) Melihat detail informasi data kelahiran

e. Melihat data kematian

1) Mencari data kematian

2) Melihat detail informasi data kematian

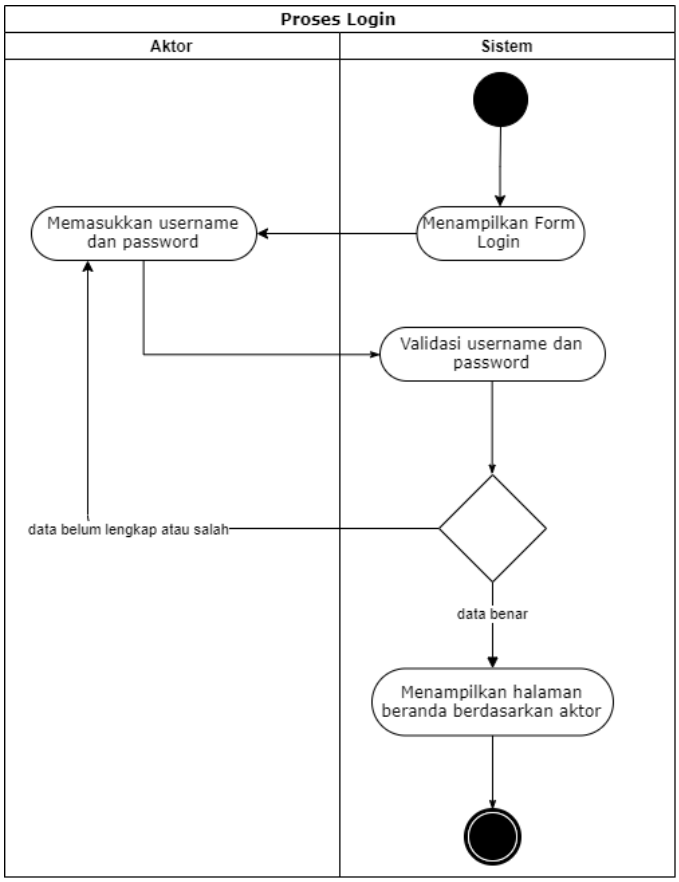
f. *Logout*

4.2.2 *Activity* Diagram

Berikut merupakan *activity diagram* dari sistem informasi pengelolaan data penduduk Kantor Kelurahan Karang Baru.

1.*User* admin:

a. Proses *login*

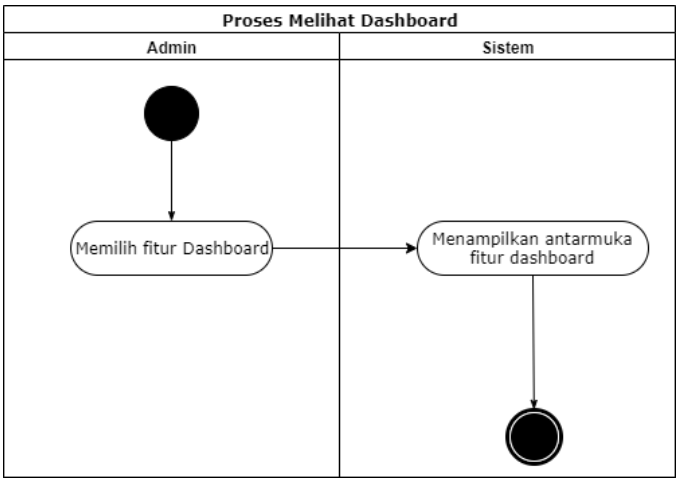


Gambar 4. 3 *Activity* diagram untuk proses *login* admin

Gambar 4.3 mengilustrasikan aktivitas untuk masuk ke dalam sistem melalui proses *login* yang dilakukan oleh *user* admin. Ketika pertama kali diakses, sistem akan menampilkan *form login* yang harus diisi oleh admin, dan selanjutnya admin akan meng-*input*-kan *username* dan *password* sesuai dengan data yang telah terdaftar pada basis data. Berikutnya, ketika admin meng-klik tombol *login* maka sistem akan memeriksa kelengkapan isi *form* yang sudah terisi, apabila terdapat *field* yang belum terisi maka sistem akan menampilkan pesan untuk melengkapi *form* terlebih dahulu. Jika *form login* sudah lengkap terisi, sistem akan melakukan validasi terhadap *username* dan *password* yang telah di-*input*-kan, jika datanya sesuai dengan data yang tersimpan pada basis data maka admin akan diarahkan ke halaman beranda sesuai level *user*-nya. Jika data yang di-

input-kan tidak sesuai, maka sistem akan menampilkan pesan kesalahan ke *user* dan kembali menampilkan halaman *login*.

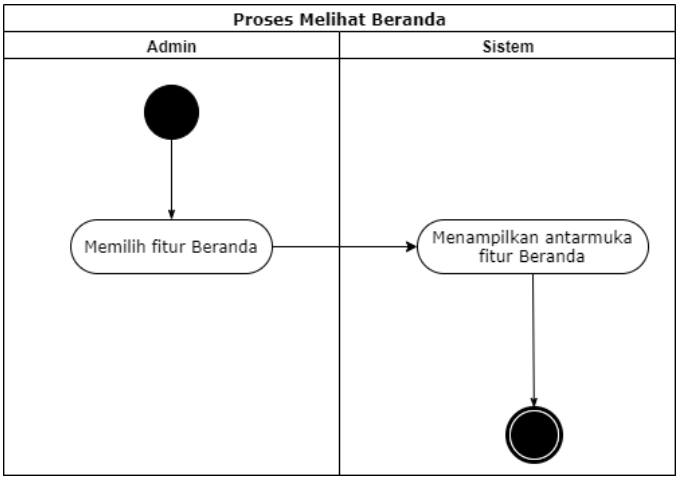
b. Proses melihat *dashboard*



Gambar 4. 4 *Activity* diagram untuk proses melihat *dashboard*

Gambar 4.4 mengilustrasikan aktivitas yang dilakukan untuk menuju ke halaman *dashboard*. Berdasarkan *activity diagram* tersebut, untuk menampilkan halaman *dashboard*, admin hanya perlu untuk memilih menu *dashboard* dan sistem akan menampilkan halaman *dashboard* yang dipilih oleh *admin*.

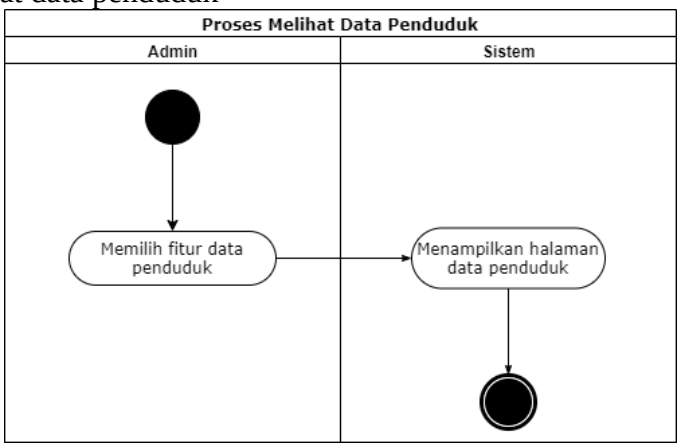
c. Proses melihat beranda



Gambar 4. 5 *Activity* diagram untuk proses melihat halaman beranda

Gambar 4.5 mengilustrasikan aktivitas yang dilakukan untuk menuju ke halaman beranda. Berdasarkan *activity diagram* tersebut, untuk menampilkan halaman beranda, admin hanya perlu untuk memilih menu beranda dan sistem akan menampilkan halaman dashboard yang dipilih oleh *admin*.

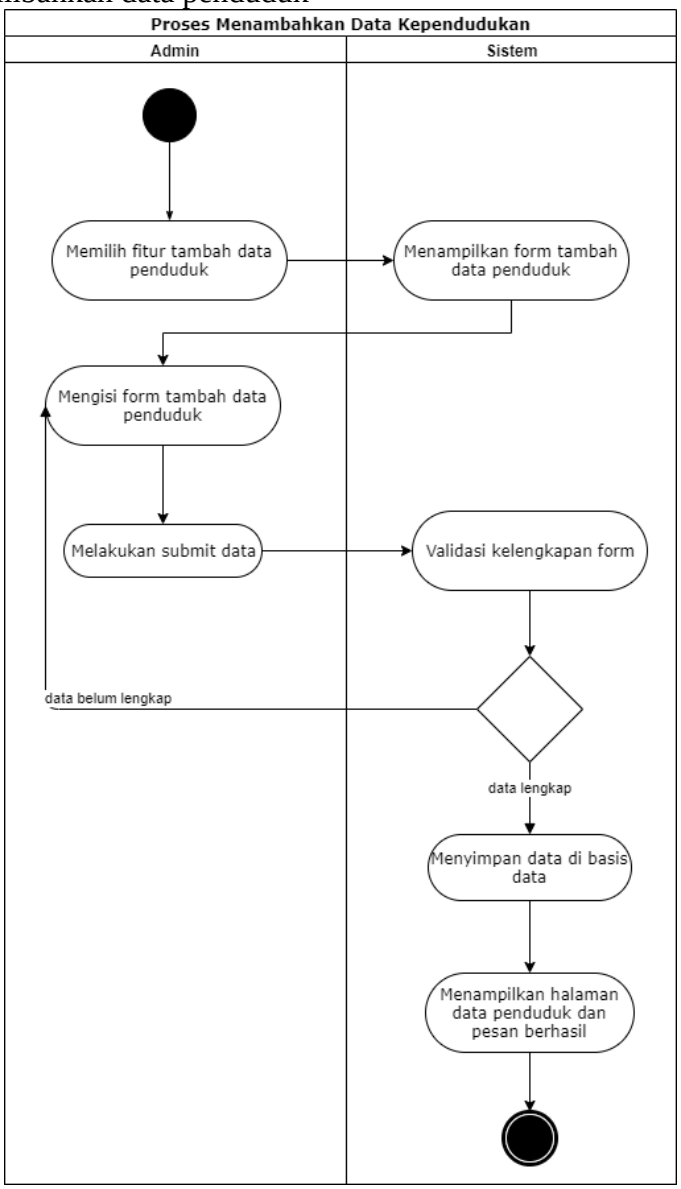
d. Proses melihat data penduduk



Gambar 4. 6 *Activity* diagram untuk proses melihat data penduduk

Gambar 4.6 mengilustrasikan aktivitas yang terjadi ketika admin memilih fitur untuk melihat data penduduk. Ketika admin mengakses halaman data penduduk maka sistem akan menampilkan daftar data penduduk yang tersimpan di sistem, jika belum terdapat data, maka sistem akan menampilkan keterangan belum ada data.

e. Proses menambahkan data penduduk

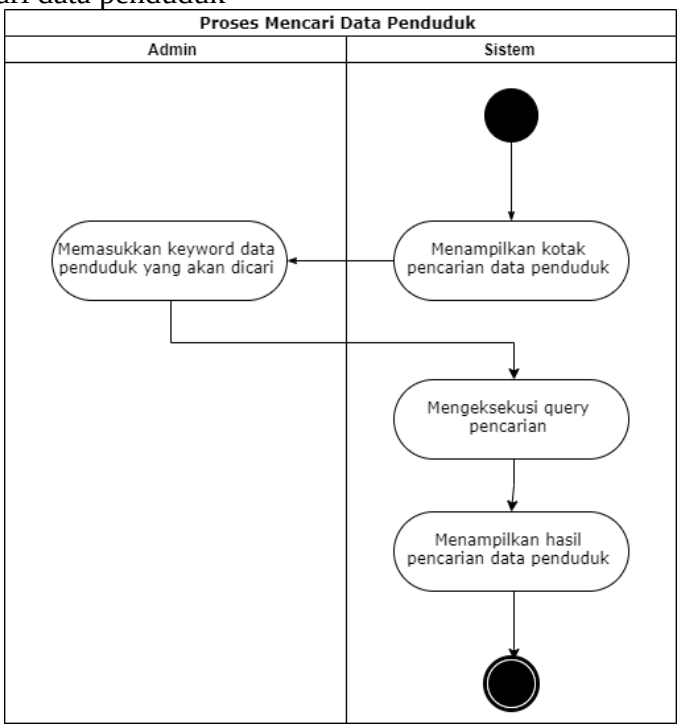


Gambar 4. 7 *Activity* diagram untuk proses menambahkan data penduduk

Gambar 4.7 mengilustrasikan aktivitas yang terjadi ketika admin menambahkan data penduduk. Untuk mengakses halaman ini admin harus terlebih dahulu membuka halaman data penduduk kemudian sistem akan menampilkan fitur untuk melakukan tambah data penduduk. Ketika admin memilih fitur tambah data penduduk maka sistem akan menampilkan *form* tambah data penduduk yang harus diisi oleh admin. Setelah admin mengisi *form* ini dan men-*submit form* sistem akan melakukan validasi kelengkapan *form*, dimana jika data yang diisikan oleh admin sudah lengkap maka data penduduk akan otomatis tersimpan pada *database* dan admin akan diarahkan kembali menuju halaman daftar penduduk serta mendapatkan pesan data penduduk berhasil ditambahkan. Sedangkan jika data yang diisikan oleh admin belum

lengkap maka sistem akan mengarahkan admin ke *field* data yang belum terisi lengkap.

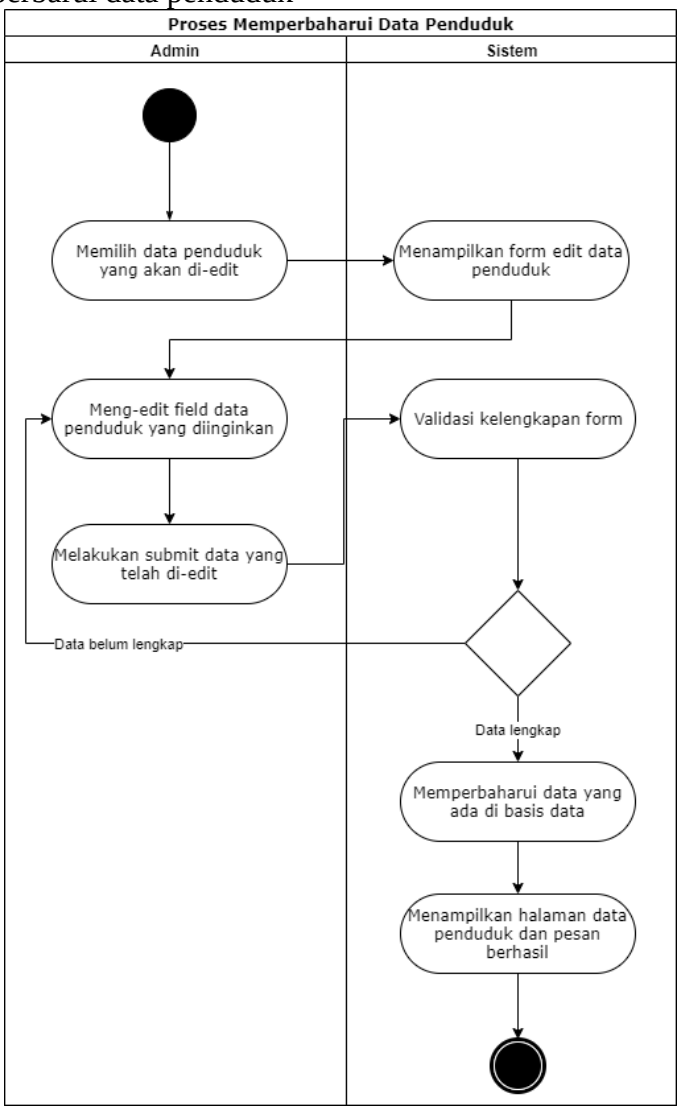
f. Proses mencari data penduduk



Gambar 4. 8 *Activity* diagram untuk proses pencarian data penduduk

Gambar 4.8 mengilustrasikan aktivitas yang terjadi ketika admin melakukan pencarian data penduduk. Untuk mengakses halaman ini admin harus terlebih dahulu membuka halaman data penduduk kemudian sistem akan menampilkan kotak pencarian untuk melakukan pencarian data penduduk. Selanjutnya admin memasukkan kata kunci dari data penduduk yang akan dicari. Lalu sistem akan mengeksekusi *query* yang diminta oleh admin dengan melakukan pengecekan pada *database*. Kemudian sistem akan menampilkan hasil pencarian yang sesuai dengan permintaan admin pada tabel daftar penduduk.

g. Proses memperbarui data penduduk

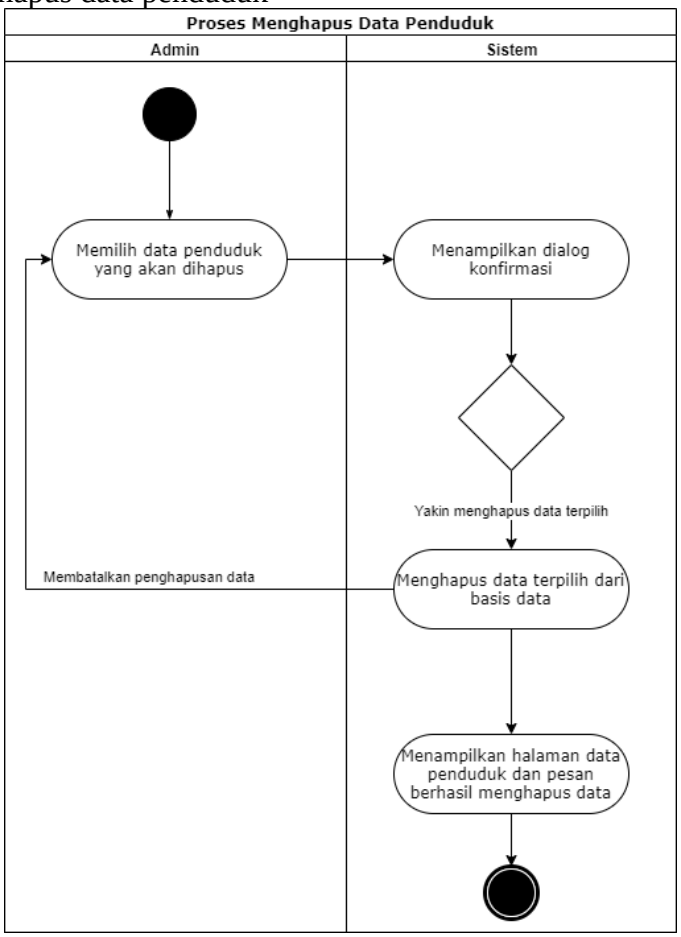


Gambar 4. 9 *Activity* diagram untuk proses pembaruan data penduduk

Gambar 4.9 mengilustrasikan aktivitas yang terjadi ketika admin melakukan pembaruan data penduduk. Untuk mengakses halaman ini admin harus terlebih dahulu membuka halaman data penduduk. Selanjutnya admin harus memilih data penduduk yang akan diperbarui dan sistem akan menampilkan *form* untuk melakukan pembaruan data penduduk. Kemudian admin dapat memilih *field* atau atribut apa yang datanya akan diperbarui dan melakukan simpan data. Ketika tombol simpan di-klik sistem akan melakukan validasi apakah semua data telah dimasukkan. Sehingga jika ada *fields* yang belum terisi atau datanya dihapus sistem akan mengarahkan admin ke *fields* tersebut untuk melengkapi datanya. Sedangkan jika semua data sudah lengkap terisi maka sistem akan memperbarui data yang ada di *database*. Kemudian sistem akan mengarahkan admin ke halaman data daftar penduduk serta mendapatkan pesan data penduduk berhasil

diperbarui.

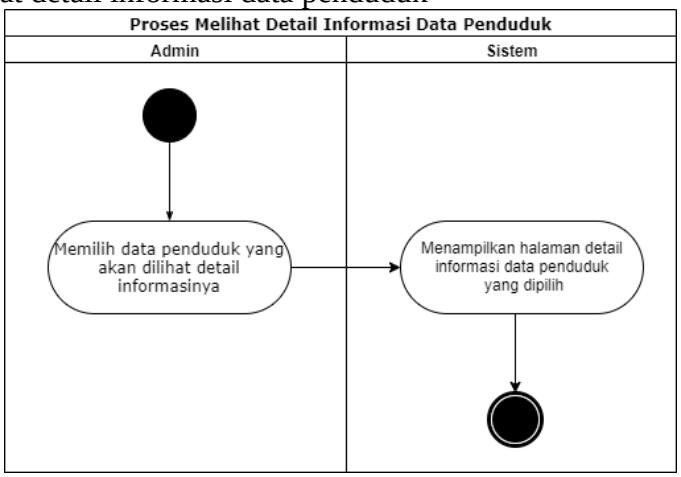
h. Proses menghapus data penduduk



Gambar 4. 10 *Activity* diagram untuk proses penghapusan data penduduk

Gambar 4.10 mengilustrasikan aktivitas yang terjadi ketika admin melakukan penghapusan data penduduk. Untuk mengakses halaman ini admin harus terlebih dahulu membuka halaman data penduduk. Selanjutnya admin harus memilih data penduduk yang akan dihapus dan sistem akan menampilkan dialog konfirmasi apakah yakin akan menghapus data penduduk. Jika admin memilih *OK* maka sistem akan menghapus data terpilih dari *database*. Kemudian sistem akan mengarahkan admin ke halaman data daftar penduduk serta mendapatkan pesan data penduduk berhasil dihapus. Sedangkan jika admin memilih *Cancel* maka data penduduk batal dihapus.

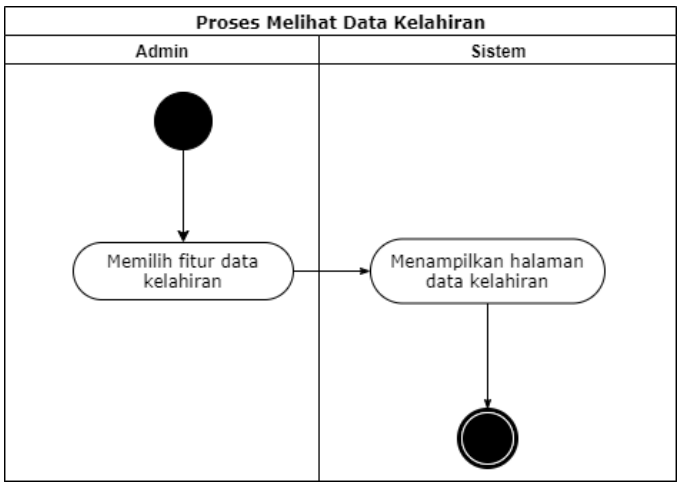
i. Proses melihat detail informasi data penduduk



Gambar 4. 11 *Activity* diagram untuk proses melihat detail informasi data penduduk

Gambar 4.11 mengilustrasikan aktivitas yang terjadi ketika admin memilih fitur untuk melihat informasi data penduduk. Untuk mengakses halaman ini admin harus terlebih dahulu membuka halaman data penduduk. Selanjutnya admin harus memilih data penduduk yang akan dilihat rincian informasinya dan sistem akan menampilkan rincian informasi tentang data penduduk yang dipilih.

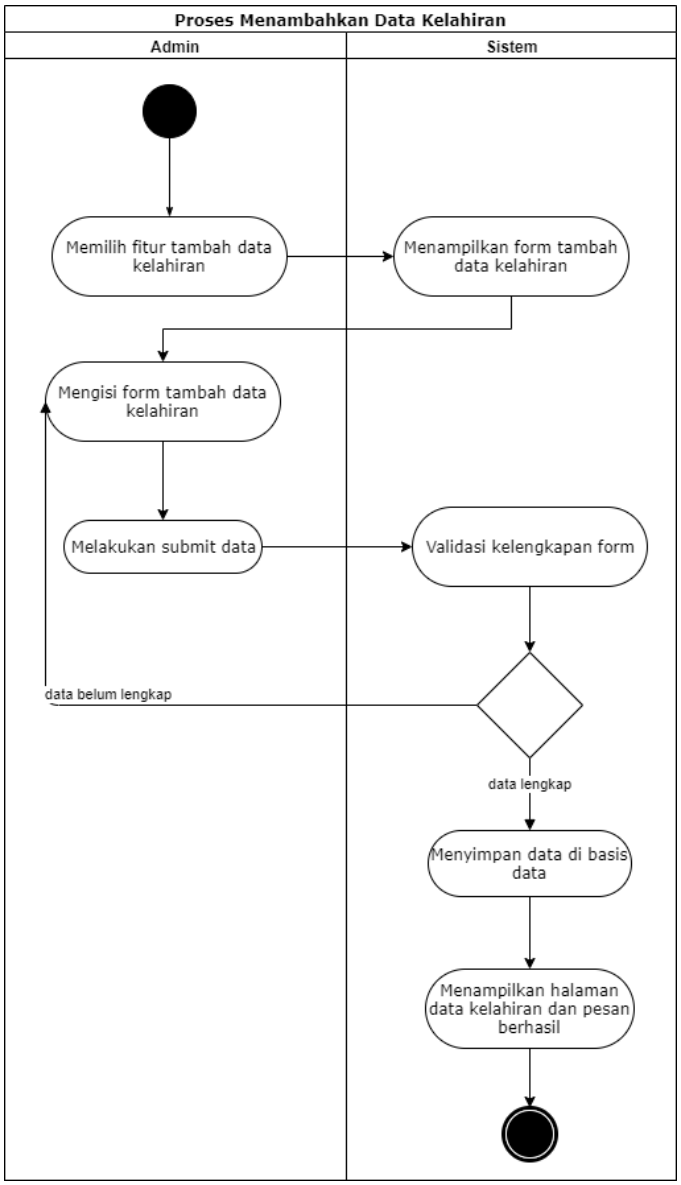
j. Proses melihat data kelahiran



Gambar 4. 12 *Activity* diagram untuk proses melihat data kelahiran

Gambar 4.12 mengilustrasikan aktivitas yang terjadi ketika admin memilih fitur untuk melihat data kelahiran. Ketika admin mengakses halaman data kelahiran maka sistem akan menampilkan daftar data kelahiran yang tersimpan di sistem, jika belum terdapat data, maka sistem akan menampilkan keterangan belum ada data.

k. Proses menambahkan data kelahiran

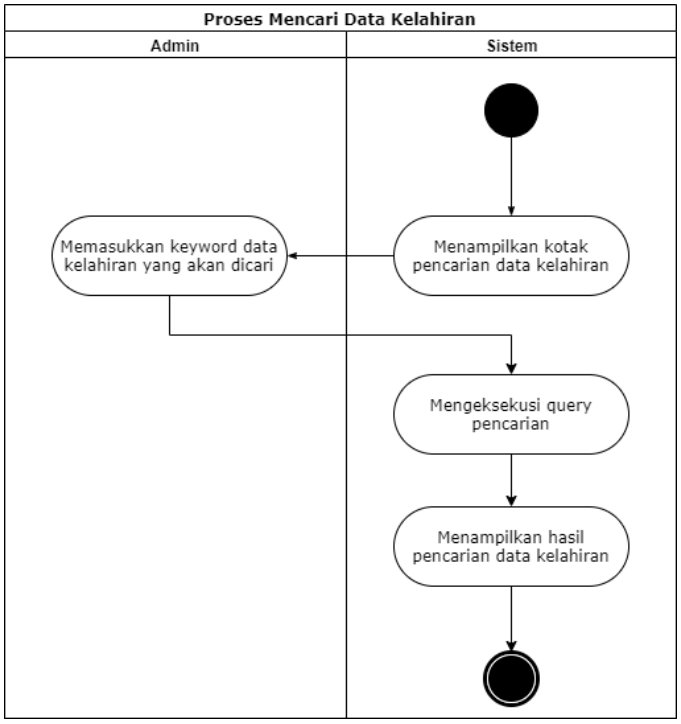


Gambar 4. 13 *Activity* diagram untuk proses menambahkan data kelahiran

Gambar 4.13 mengilustrasikan aktivitas yang terjadi ketika admin menambahkan data kelahiran. Untuk mengakses halaman ini admin harus terlebih dahulu membuka halaman data kelahiran kemudian sistem akan menampilkan fitur untuk melakukan tambah data kelahiran. Ketika admin memilih fitur tambah data kelahiran maka sistem akan menampilkan *form* tambah data kelahiran yang harus diisi oleh admin. Setelah admin mengisi *form* ini dan men-*submit form* sistem akan melakukan validasi kelengkapan *form*, dimana jika data yang diisikan oleh admin sudah lengkap maka data kelahiran akan otomatis tersimpan pada *database* dan admin akan diarahkan kembali menuju halaman daftar kelahiran serta mendapatkan pesan data kelahiran berhasil ditambahkan. Sedangkan jika data yang diisikan oleh admin belum lengkap maka sistem akan mengarahkan

admin ke *field* data yang belum terisi lengkap.

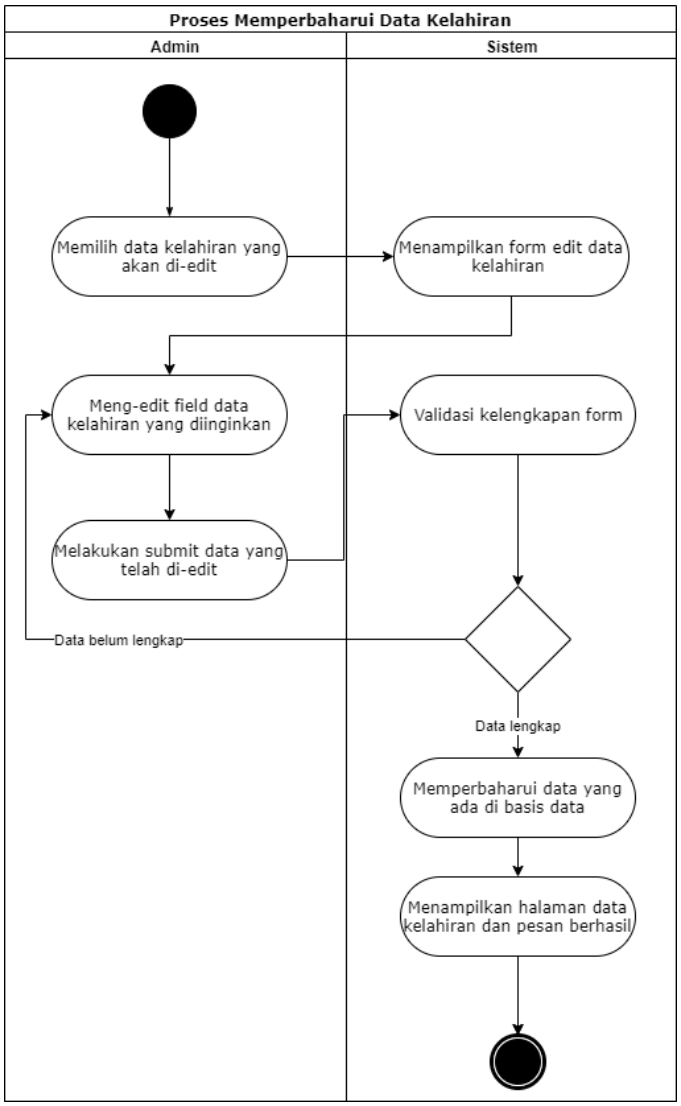
l. Proses mencari data kelahiran



Gambar 4. 14 *Activity* diagram untuk proses pencarian data kelahiran

Gambar 4.14 mengilustrasikan aktivitas yang terjadi ketika admin melakukan pencarian data kelahiran. Untuk mengakses halaman ini admin harus terlebih dahulu membuka halaman data kelahiran kemudian sistem akan menampilkan kotak pencarian untuk melakukan pencarian data kelahiran. Selanjutnya admin memasukkan kata kunci dari data kelahiran yang akan dicari. Lalu sistem akan mengeksekusi *query* yang diminta oleh admin dengan melakukan pengecekan pada *database*. Kemudian sistem akan menampilkan hasil pencarian yang sesuai dengan permintaan admin pada tabel daftar kelahiran.

m. Proses memperbarui data kelahiran

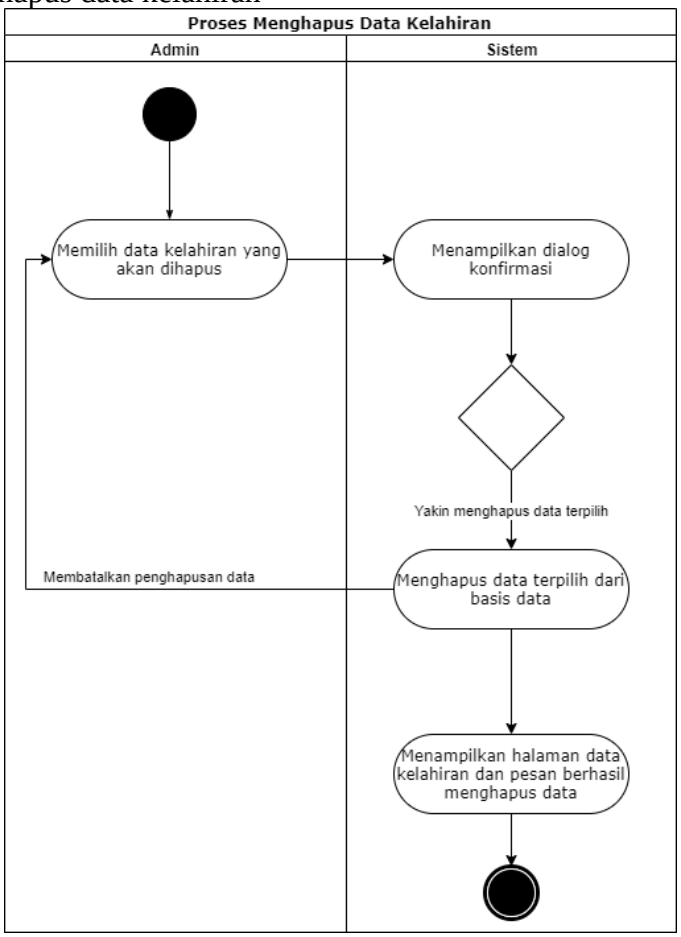


Gambar 4. 15 *Activity* diagram untuk proses memperbarui data kelahiran

Gambar 4.15 mengilustrasikan aktivitas yang terjadi ketika admin melakukan pembaruan data kelahiran. Untuk mengakses halaman ini admin harus terlebih dahulu membuka halaman data kelahiran. Selanjutnya admin harus memilih data kelahiran yang akan diperbarui dan sistem akan menampilkan *form* untuk melakukan pembaruan data kelahiran. Kemudian admin dapat memilih *field* atau atribut apa yang datanya akan diperbarui dan melakukan simpan data. Ketika tombol simpan di-klik sistem akan melakukan validasi apakah semua data telah dimasukkan. Sehingga jika ada *fields* yang belum terisi atau datanya dihapus sistem akan mengarahkan admin ke *fields* tersebut untuk melengkapi datanya. Sedangkan jika semua data sudah lengkap terisi maka sistem akan memperbarui data yang ada di *database*. Kemudian sistem akan mengarahkan admin ke halaman data daftar kelahiran serta mendapatkan pesan data kelahiran berhasil

diperbarui.

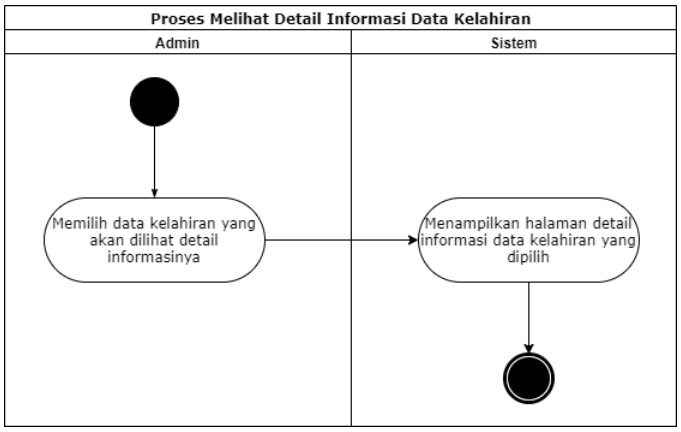
n. Proses menghapus data kelahiran



Gambar 4. 16 *Activity* diagram untuk proses penghapusan data kelahiran

Gambar 4.16 mengilustrasikan aktivitas yang terjadi ketika admin melakukan penghapusan data kelahiran. Untuk mengakses halaman ini admin harus terlebih dahulu membuka halaman data kelahiran. Selanjutnya admin harus memilih data kelahiran yang akan dihapus dan sistem akan menampilkan dialog konfirmasi apakah yakin akan menghapus data kelahiran. Jika admin memilih *OK* maka sistem akan menghapus data terpilih dari *database*. Kemudian sistem akan mengarahkan admin ke halaman data daftar kelahiran serta mendapatkan pesan data kelahiran berhasil dihapus. Sedangkan jika admin memilih *Cancel* maka data kelahiran batal dihapus.

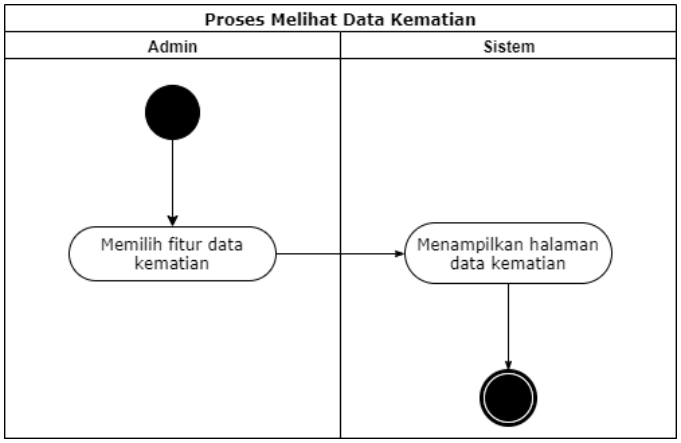
o. Proses melihat detail informasi data kelahiran



Gambar 4. 17 *Activity* diagram untuk proses melihat detail informasi data kelahiran

Gambar 4.17 mengilustrasikan aktivitas yang terjadi ketika admin memilih fitur untuk melihat informasi data kelahiran. Untuk mengakses halaman ini admin harus terlebih dahulu membuka halaman data kelahiran. Selanjutnya admin harus memilih data kelahiran yang akan dilihat rincian informasinya dan sistem akan menampilkan rincian informasi tentang data kelahiran yang dipilih.

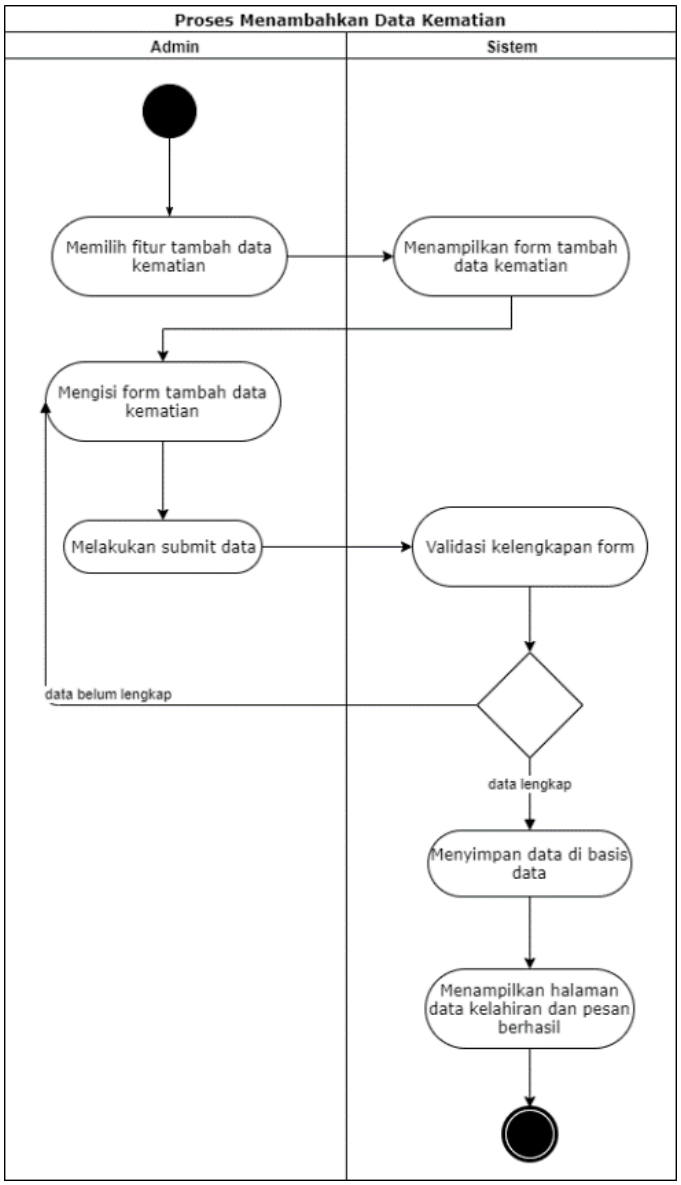
p. Proses melihat data kematian



Gambar 4. 18 *Activity* diagram untuk proses melihat data kematian

Gambar 4.18 mengilustrasikan aktivitas yang terjadi ketika admin memilih fitur untuk melihat data kematian. Ketika admin mengakses halaman data kematian maka sistem akan menampilkan daftar data kematian yang tersimpan di sistem, jika belum terdapat data, maka sistem akan menampilkan keterangan belum ada data.

q. Proses menambahkan data kematian

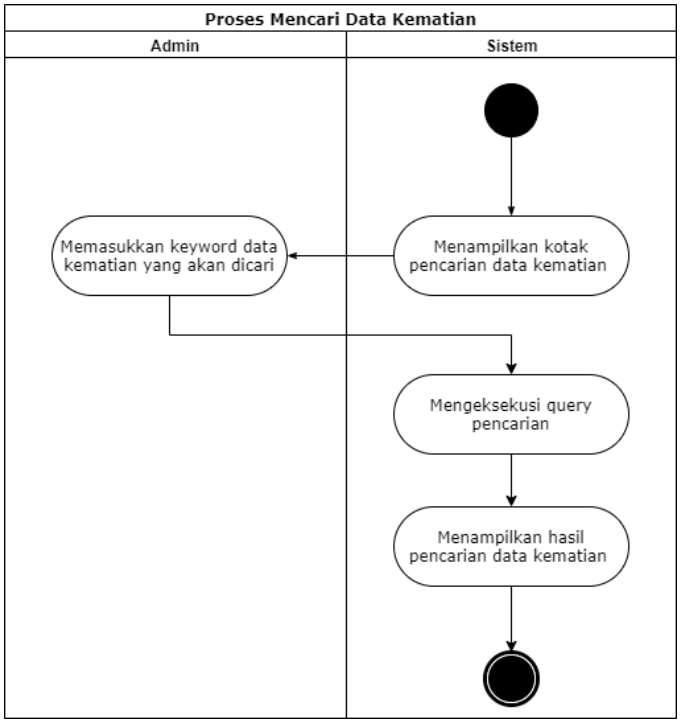


Gambar 4. 19 *Activity* diagram untuk proses menambahkan data kematian

Gambar 4.19 mengilustrasikan aktivitas yang terjadi ketika admin menambahkan data kematian. Untuk mengakses halaman ini admin harus terlebih dahulu membuka halaman data kematian kemudian sistem akan menampilkan fitur untuk melakukan tambah data kematian. Ketika admin memilih fitur tambah data kematian maka sistem akan menampilkan *form* tambah data kematian yang harus diisi oleh admin. Setelah admin mengisi *form* ini dan men-*submit form* sistem akan melakukan validasi kelengkapan *form*, dimana jika data yang diisikan oleh admin sudah lengkap maka data kematian akan otomatis tersimpan pada *database* dan admin akan diarahkan kembali menuju halaman daftar kematian serta mendapatkan pesan data kematian berhasil ditambahkan. Sedangkan jika data yang diisikan oleh admin belum lengkap maka sistem akan mengarahkan

admin ke *field* data yang belum terisi lengkap.

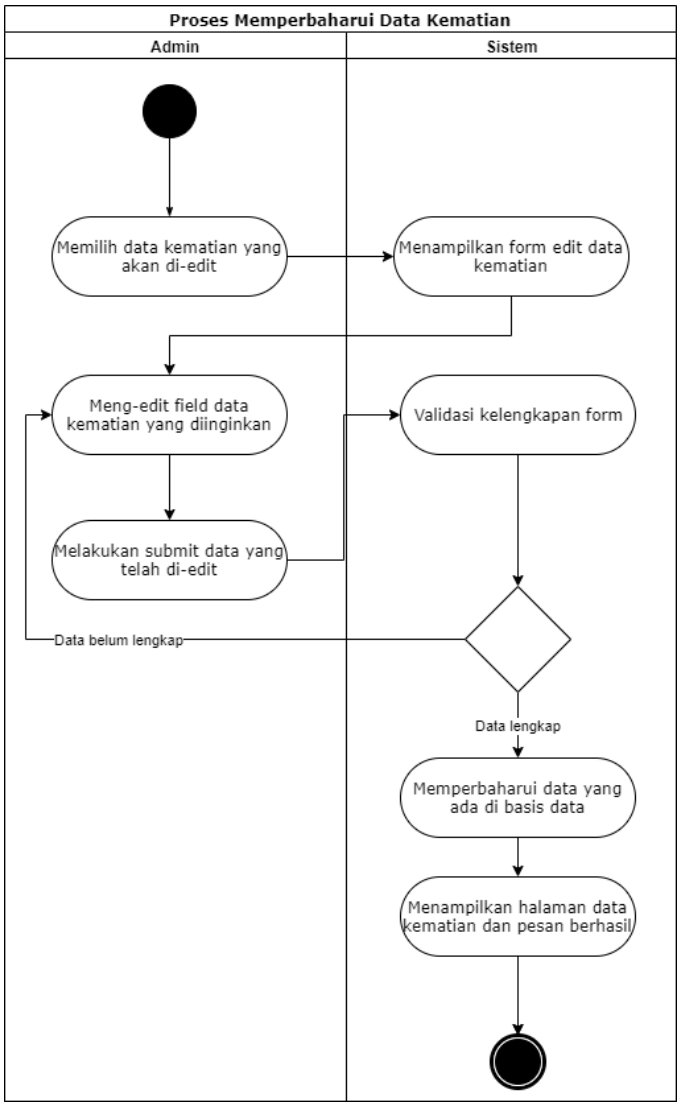
r. Proses mencari data kematian



Gambar 4. 20 *Activity* diagram untuk proses pencarian data kematian

Gambar 4.20 mengilustrasikan aktivitas yang terjadi ketika admin melakukan pencarian data kematian. Untuk mengakses halaman ini admin harus terlebih dahulu membuka halaman data kematian kemudian sistem akan menampilkan kotak pencarian untuk melakukan pencarian data kematian. Selanjutnya admin memasukkan kata kunci dari data kematian yang akan dicari. Lalu sistem akan mengeksekusi *query* yang diminta oleh admin dengan melakukan pengecekan pada *database*. Kemudian sistem akan menampilkan hasil pencarian yang sesuai dengan permintaan admin pada tabel daftar kematian.

s. Proses memperbarui data kematian

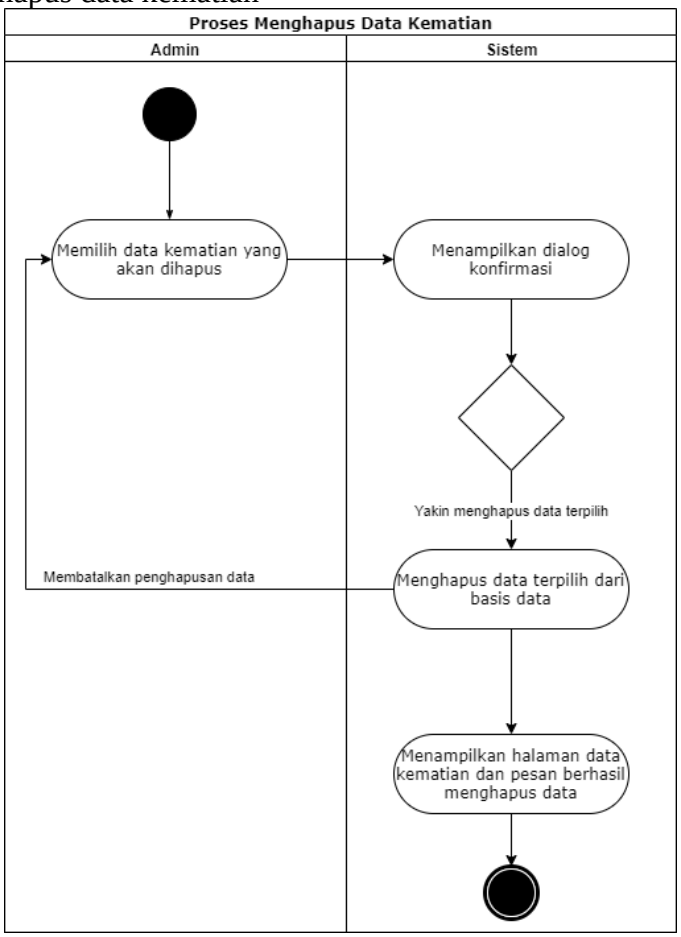


Gambar 4. 21 *Activity* diagram untuk proses memperbarui data kematian

Gambar 4.21 mengilustrasikan aktivitas yang terjadi ketika admin melakukan pembaruan data kematian. Untuk mengakses halaman ini admin harus terlebih dahulu membuka halaman data kematian. Selanjutnya admin harus memilih data kematian yang akan diperbarui dan sistem akan menampilkan *form* untuk melakukan pembaruan data kematian. Kemudian admin dapat memilih *field* atau atribut apa yang datanya akan diperbarui dan melakukan simpan data. Ketika tombol simpan di-klik sistem akan melakukan validasi apakah semua data telah dimasukkan. Sehingga jika ada *fields* yang belum terisi atau datanya dihapus sistem akan mengarahkan admin ke *fields* tersebut untuk melengkapi datanya. Sedangkan jika semua data sudah lengkap terisi maka sistem akan memperbarui data yang ada di *database*. Kemudian sistem akan mengarahkan admin ke halaman data daftar kematian serta mendapatkan pesan data kematian berhasil

diperbarui.

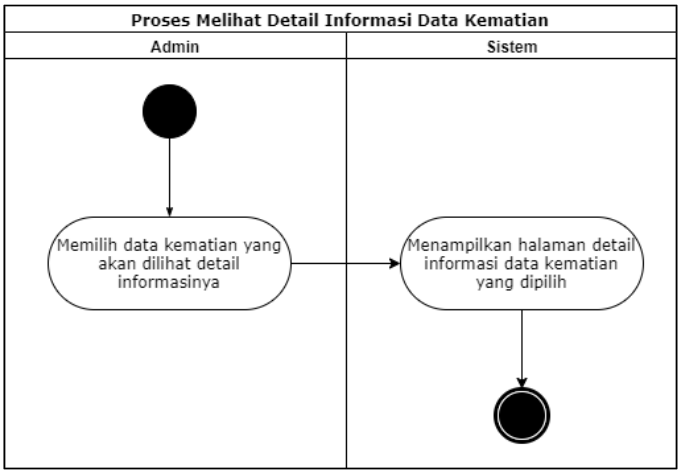
t. Proses menghapus data kematian



Gambar 4. 22 *Activity* diagram untuk proses penghapusan data kematian

Gambar 4.22 mengilustrasikan aktivitas yang terjadi ketika admin melakukan penghapusan data kematian. Untuk mengakses halaman ini admin harus terlebih dahulu membuka halaman data kematian. Selanjutnya admin harus memilih data kematian yang akan dihapus dan sistem akan menampilkan dialog konfirmasi apakah yakin akan menghapus data kematian. Jika admin memilih *OK* maka sistem akan menghapus data terpilih dari *database*. Kemudian sistem akan mengarahkan admin ke halaman data daftar kematian serta mendapatkan pesan data kematian berhasil dihapus. Sedangkan jika admin memilih *Cancel* maka data kematian batal dihapus.

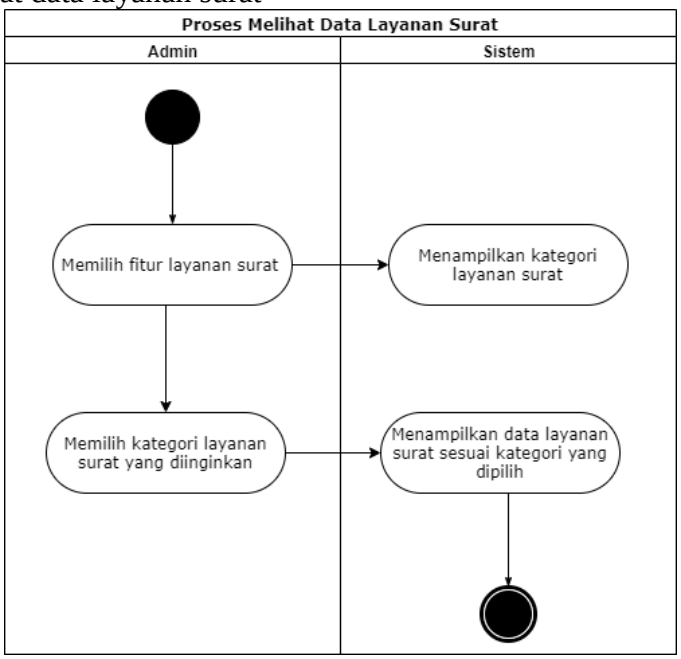
u. Proses melihat detail informasi data kematian



Gambar 4. 23 *Activity* diagram untuk proses melihat detail informasi data kematian

Gambar 4.23 mengilustrasikan aktivitas yang terjadi ketika admin memilih fitur untuk melihat informasi data kematian. Untuk mengakses halaman ini admin harus terlebih dahulu membuka halaman data kematian. Selanjutnya admin harus memilih data kematian yang akan dilihat rincian informasinya dan sistem akan menampilkan rincian informasi tentang data kematian yang dipilih.

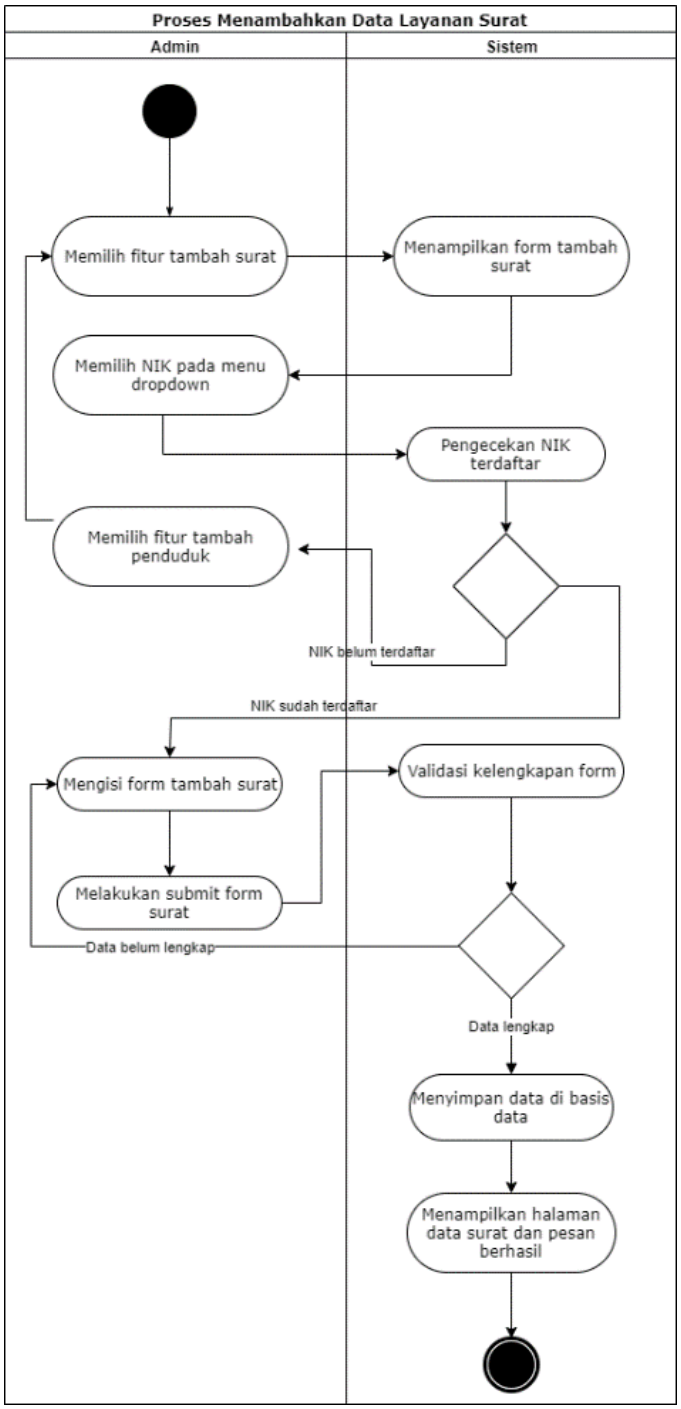
v. Proses melihat data layanan surat



Gambar 4. 24 Activity diagram untuk proses melihat data layanan surat

Gambar 4.24 mengilustrasikan aktivitas yang terjadi ketika admin memilih fitur untuk melihat data layanan surat. Ketika admin mengakses halaman data layanan surat maka sistem akan menampilkan daftar data layanan surat yang tersimpan di sistem, jika belum terdapat data, maka sistem akan menampilkan keterangan belum ada data.

w. Proses menambahkan data surat



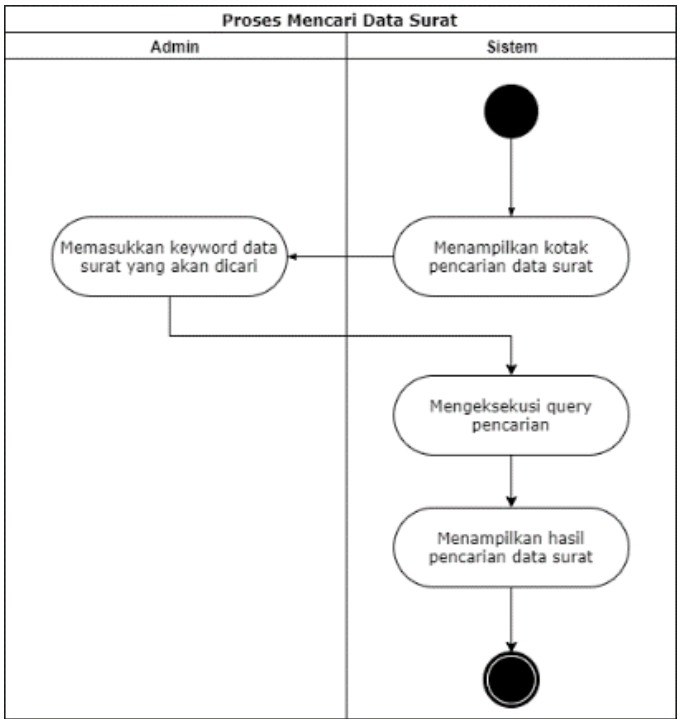
Gambar 4. 25 Activity diagram untuk proses menambahkan data surat

Gambar 4.25 mengilustrasikan aktivitas yang terjadi ketika admin menambahkan data layanan surat. Untuk mengakses halaman ini admin harus terlebih dahulu membuka halaman data layanan surat yang diinginkan kemudian sistem akan menampilkan fitur untuk melakukan tambah data layanan surat sesuai dengan jenis layanan surat yang dipilih. Ketika admin memilih fitur tambah data surat maka sistem akan menampilkan *form* tambah data surat yang harus diisi oleh admin. Untuk dapat menambahkan surat, syaratnya adalah NIK

penduduk yang ingin dibuatkan surat harus terlebih dahulu terdaftar dalam sistem, namun terdapat pengecualian untuk layanan tambah data surat kelahiran dimana yang harus terdaftar dalam sistem adalah NIK ayah dan NIK ibu dari anak yang baru lahir. Jika NIK sudah terdaftar admin bisa langsung memilih NIK dari penduduk yang bersangkutan pada menu *dropdown* kemudian melengkapi *form* surat. Sedangkan jika NIK penduduk yang bersangkutan belum terdaftar pada sistem, admin dapat memilih menu tambah data penduduk terlebih dahulu untuk mendaftarkan NIK penduduk yang bersangkutan pada sistem.

Setelah admin mengisi *form* tambah data surat dan men-*submit form*, sistem akan melakukan validasi kelengkapan *form*, dimana jika data yang diisikan oleh admin sudah lengkap maka data surat akan otomatis tersimpan pada *database* dan admin akan diarahkan kembali menuju halaman daftar surat serta mendapatkan pesan data surat berhasil ditambahkan. Sedangkan jika data yang diisikan oleh admin belum lengkap maka sistem akan mengarahkan admin ke *field* data yang belum terisi lengkap.

x. Proses mencari data surat

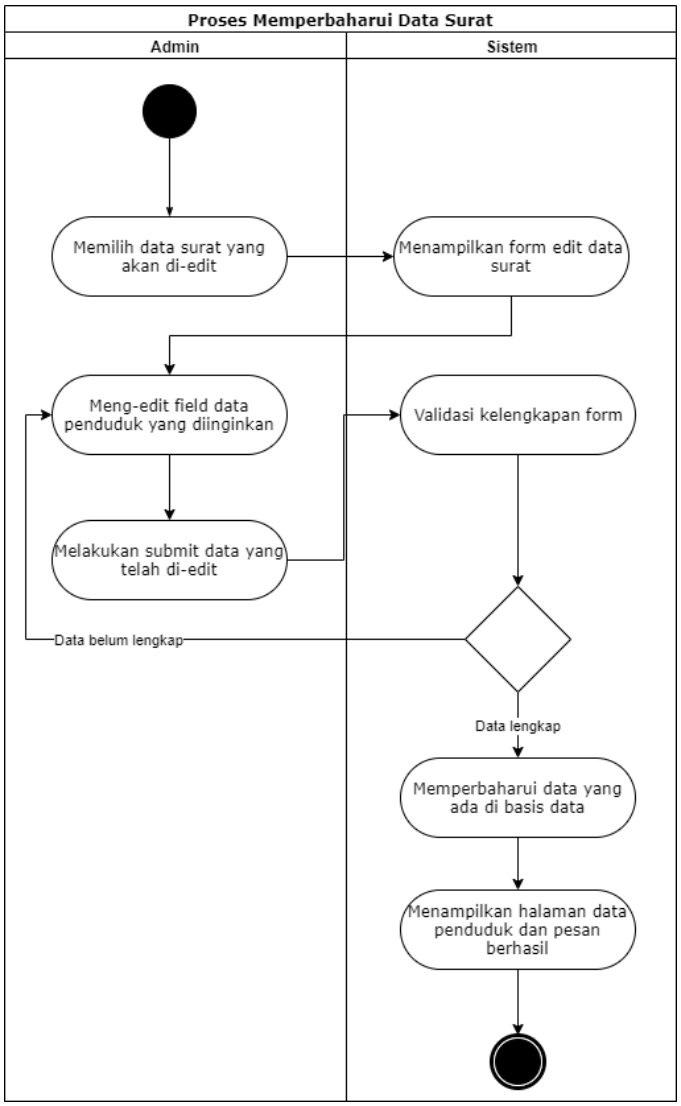


Gambar 4. 26 *Activity* diagram untuk proses pencarian data surat

Gambar 4.26 mengilustrasikan aktivitas yang terjadi ketika admin melakukan pencarian data surat. Untuk mengakses halaman ini admin harus terlebih dahulu membuka halaman data layanan surat kemudian sistem akan menampilkan kotak

pencarian untuk melakukan pencarian data surat. Selanjutnya admin memasukkan kata kunci dari data surat yang akan dicari. Lalu sistem akan mengeksekusi *query* yang diminta oleh admin dengan melakukan pengecekan pada *database*. Kemudian sistem akan menampilkan hasil pencarian yang sesuai dengan permintaan admin pada tabel daftar surat.

y. Proses memperbarui data surat

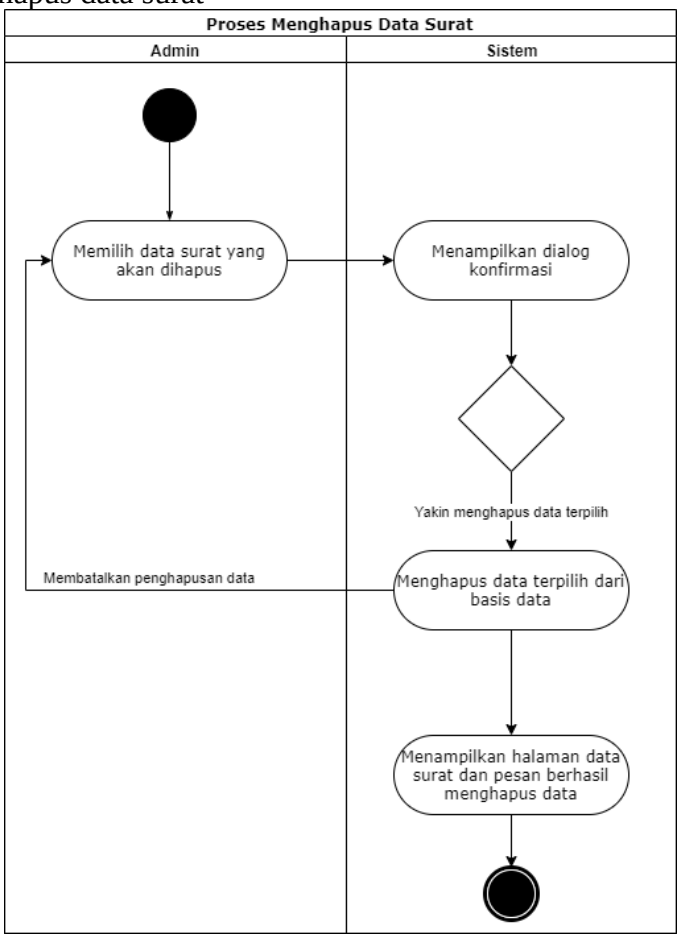


Gambar 4. 27 *Activity* diagram untuk proses pembaruan data surat

Gambar 4.27 mengilustrasikan aktivitas yang terjadi ketika admin melakukan pembaruan data surat. Untuk mengakses halaman ini admin harus terlebih dahulu membuka halaman data layanan surat. Selanjutnya admin harus memilih data surat yang akan diperbarui dan sistem akan menampilkan *form* untuk melakukan pembaruan data surat. Kemudian admin dapat memilih *field* atau atribut apa yang datanya akan diperbarui dan melakukan simpan data. Ketika tombol simpan di-klik sistem akan melakukan validasi apakah semua data telah dimasukkan.

Sehingga jika ada *fields* yang belum terisi atau datanya dihapus sistem akan mengarahkan admin ke *fields* tersebut untuk melengkapi datanya. Sedangkan jika semua data sudah lengkap terisi maka sistem akan memperbarui data yang ada di *database*. Kemudian sistem akan mengarahkan admin ke halaman data daftar surat serta mendapatkan pesan data surat berhasil diperbarui.

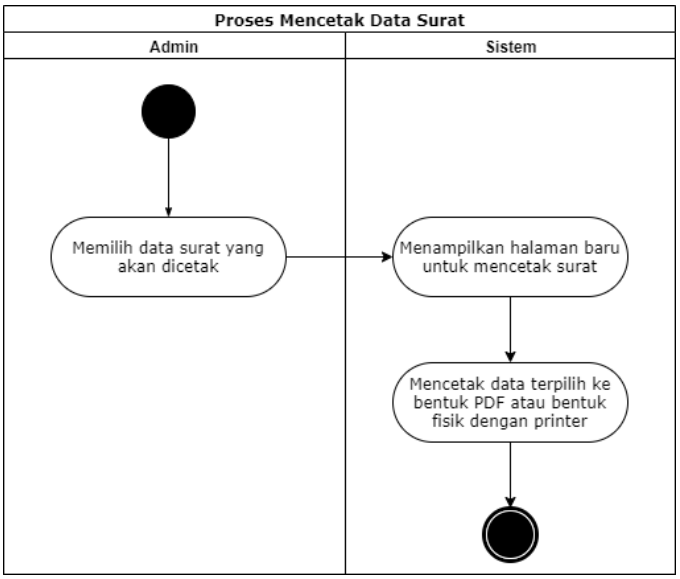
z. Proses menghapus data surat



Gambar 4. 28 *Activity* diagram untuk proses penghapusan data surat

Gambar 4.28 mengilustrasikan aktivitas yang terjadi ketika admin melakukan penghapusan data surat. Untuk mengakses halaman ini admin harus terlebih dahulu membuka halaman data surat. Selanjutnya admin harus memilih data surat yang akan dihapus dan sistem akan menampilkan dialog konfirmasi apakah yakin akan menghapus data surat. Jika admin memilih *OK* maka sistem akan menghapus data terpilih dari *database*. Kemudian sistem akan mengarahkan admin ke halaman data daftar surat serta mendapatkan pesan data surat berhasil dihapus. Sedangkan jika admin memilih *Cancel* maka data surat batal dihapus.

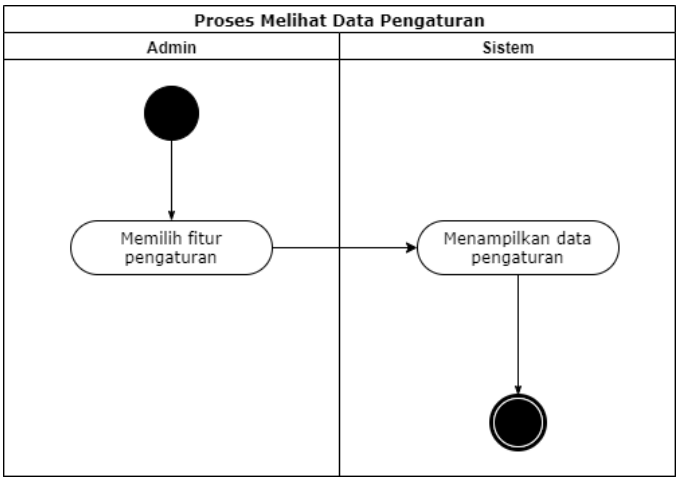
- Proses mencetak data surat



Gambar 4. 29 *Activity* diagram untuk proses pencetakan data surat

Gambar 4.29 mengilustrasikan aktivitas yang terjadi ketika admin memilih fitur untuk mencetak data surat. Untuk mengakses halaman ini admin harus terlebih dahulu membuka halaman data surat. Selanjutnya admin harus memilih data surat yang akan dicetak dan sistem akan mengarahkan admin ke halaman baru untuk mencetak surat dimana admin dapat mengatur *properties* untuk pencetakan surat sesuai dengan kebutuhan.

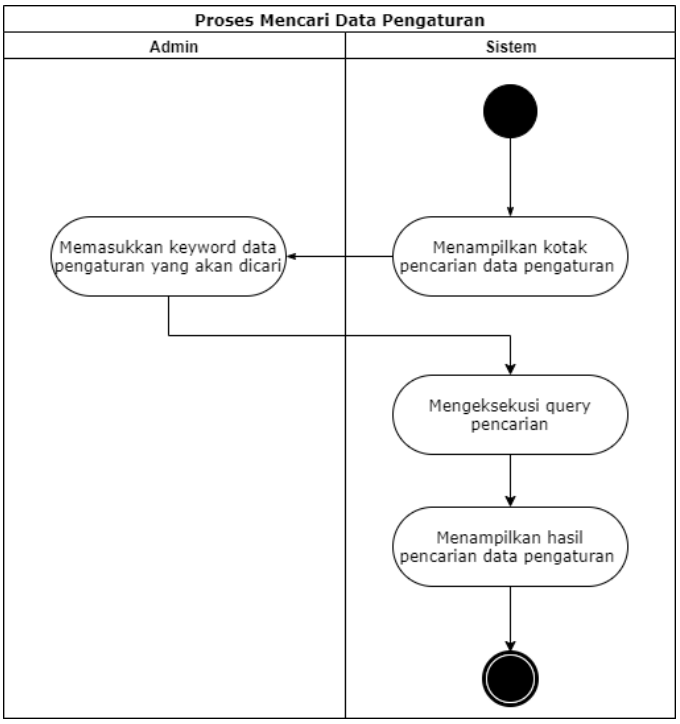
- Proses melihat data pengaturan



Gambar 4. 30 *Activity* diagram untuk proses melihat halaman data pengaturan

Gambar 4.30 mengilustrasikan aktivitas yang terjadi ketika admin memilih fitur untuk melihat data pengaturan. Ketika admin mengakses halaman data pengaturan maka sistem akan menampilkan daftar data pengaturan yang tersimpan di sistem, jika belum terdapat data, maka sistem akan menampilkan keterangan belum ada data.

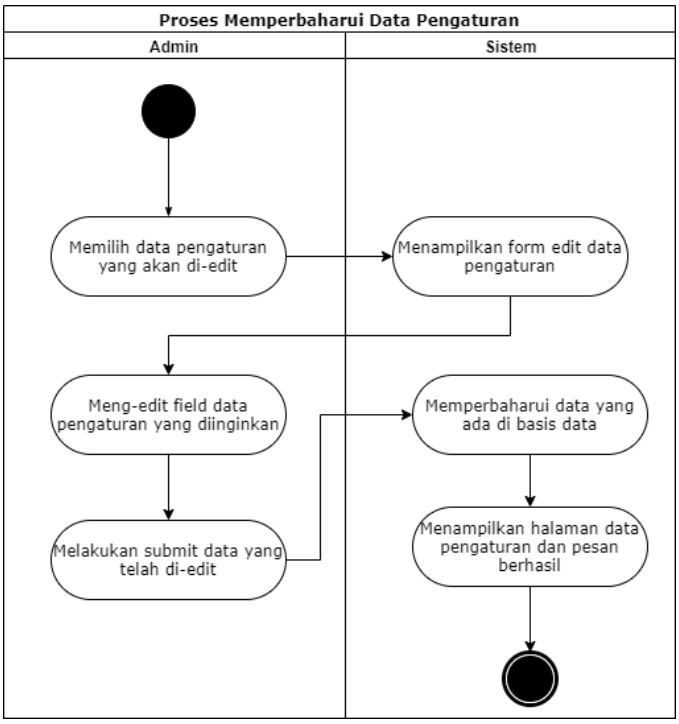
- Proses mencari data pengaturan



Gambar 4. 31 *Activity* diagram untuk proses pencarian data pengaturan

Gambar 4.31 mengilustrasikan aktivitas yang terjadi ketika admin melakukan pencarian data pengaturan. Untuk mengakses halaman ini admin harus terlebih dahulu membuka halaman data pengaturan kemudian sistem akan menampilkan kotak pencarian untuk melakukan pencarian data pengaturan. Selanjutnya admin memasukkan kata kunci dari data pengaturan yang akan dicari. Lalu sistem akan mengeksekusi *query* yang diminta oleh admin dengan melakukan pengecekan pada *database*. Kemudian sistem akan menampilkan hasil pencarian yang sesuai dengan permintaan admin pada tabel daftar pengaturan.

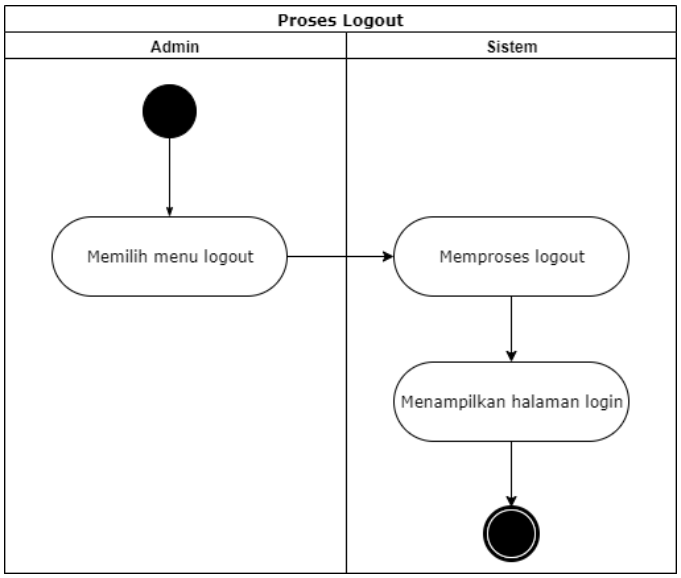
- Proses memperbarui data pengaturan



Gambar 4. 32 *Activity* diagram untuk proses memperbarui data pengaturan

Gambar 4.32 mengilustrasikan aktivitas yang terjadi ketika admin melakukan pembaruan data pengaturan. Untuk mengakses halaman ini admin harus terlebih dahulu membuka halaman data pengaturan. Selanjutnya admin harus memilih data pengaturan yang akan diperbarui dan sistem akan menampilkan *form* untuk melakukan pembaruan data pengaturan. Kemudian admin dapat memilih *field* atau atribut apa yang datanya akan diperbarui dan melakukan simpan data. Ketika tombol simpan di-klik sistem akan melakukan validasi apakah semua data telah dimasukkan. Sehingga jika ada *fields* yang belum terisi atau datanya dihapus sistem akan mengarahkan admin ke *fields* tersebut untuk melengkapi datanya. Sedangkan jika semua data sudah lengkap terisi maka sistem akan memperbarui data yang ada di *database*. Kemudian sistem akan mengarahkan admin ke halaman data daftar pengaturan serta mendapatkan pesan data pengaturan berhasil diperbarui.

- Proses *logout*

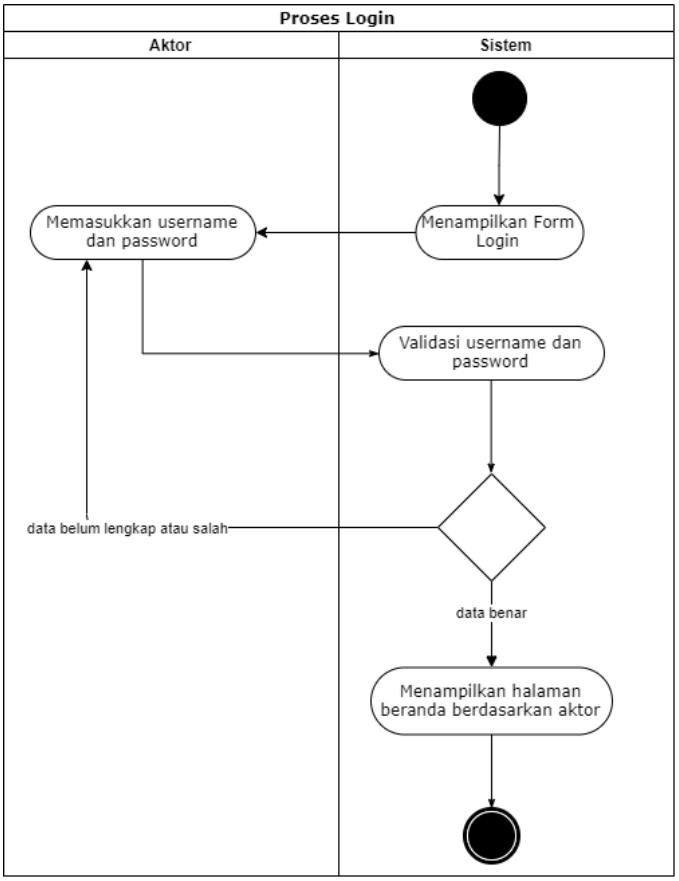


Gambar 4. 33 *Activity* diagram untuk proses *logout*

Gambar 4.33 mengilustrasikan aktivitas untuk keluar dari sistem melalui proses *logout* yang dilakukan oleh *user* admin. Ketika admin memilih menu *logout*, sistem akan memproses *logout* dengan cara mengakhiri *session* dengan perintah *destroy* yang akan menghapus file *session* dari *server*. Ketika proses *logout* berhasil dan *session* berakhir maka admin akan diarahkan ke halaman *login*.

2.*User* Sekretaris :

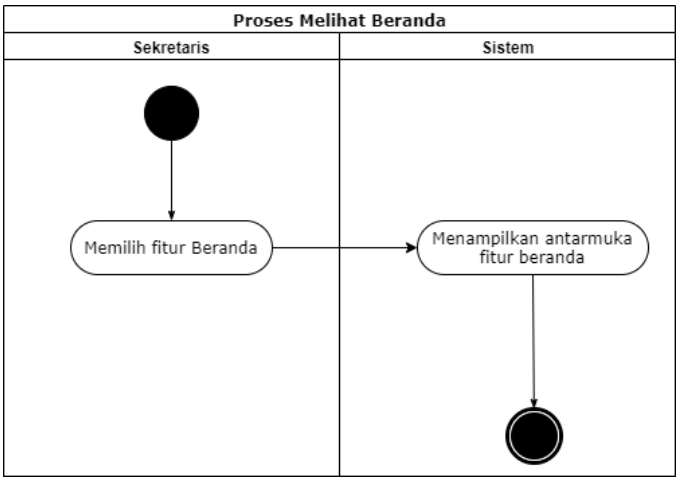
a. Proses *login*



Gambar 4. 34 *Activity* diagram untuk proses *login* sekretaris

Gambar 4.34 mengilustrasikan aktivitas untuk masuk ke dalam sistem melalui proses *login* yang dilakukan oleh *user* sekretaris. Ketika pertama kali diakses, sistem akan menampilkan *form login* yang harus diisi oleh sekretaris, dan selanjutnya sekretaris akan meng-*input*-kan *username* dan *password* sesuai dengan data yang telah terdaftar pada basis data. Berikutnya, ketika sekretaris meng-klik tombol *login* maka sistem akan memeriksa kelengkapan isi *form* yang sudah terisi, apabila terdapat *field* yang belum terisi maka sistem akan menampilkan pesan untuk melengkapi *form* terlebih dahulu. Jika *form login* sudah lengkap terisi, sistem akan melakukan validasi terhadap *username* dan *password* yang telah di-*input*-kan, jika datanya sesuai dengan data yang tersimpan pada basis data maka sekretaris akan diarahkan ke halaman beranda sesuai level *user*-nya. Jika data yang di-*input*-kan tidak sesuai, maka sistem akan menampilkan pesan kesalahan ke *user* dan kembali menampilkan halaman *login*.

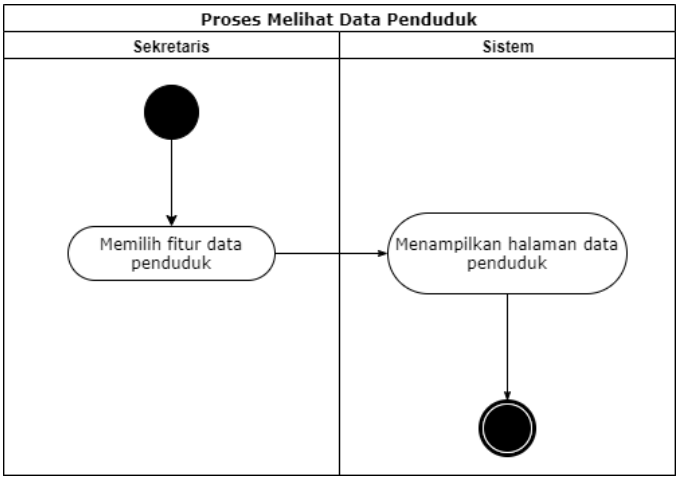
b. Proses melihat beranda



Gambar 4. 35 *Activity* diagram untuk proses melihat beranda

Gambar 4.35 mengilustrasikan aktivitas yang dilakukan untuk menuju ke halaman beranda. Berdasarkan *activity diagram* tersebut, untuk menampilkan halaman beranda, admin hanya perlu untuk memilih menu beranda dan sistem akan menampilkan halaman beranda yang dipilih oleh admin.

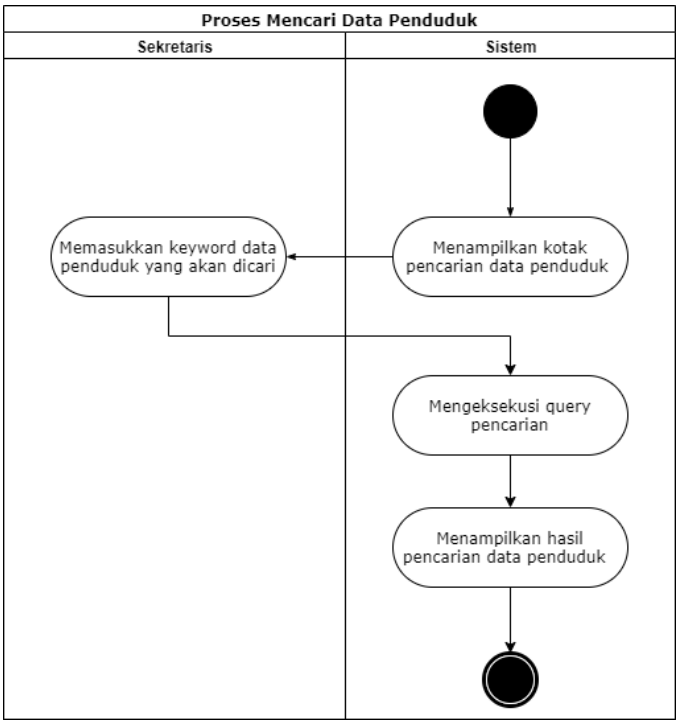
c. Proses melihat data penduduk



Gambar 4. 36 *Activity* diagram untuk proses melihat halaman data penduduk

Gambar 4.36 mengilustrasikan aktivitas yang terjadi ketika admin memilih fitur untuk melihat data penduduk. Ketika admin mengakses halaman data penduduk maka sistem akan menampilkan daftar data penduduk yang tersimpan di sistem, jika belum terdapat data, maka sistem akan menampilkan keterangan belum ada data.

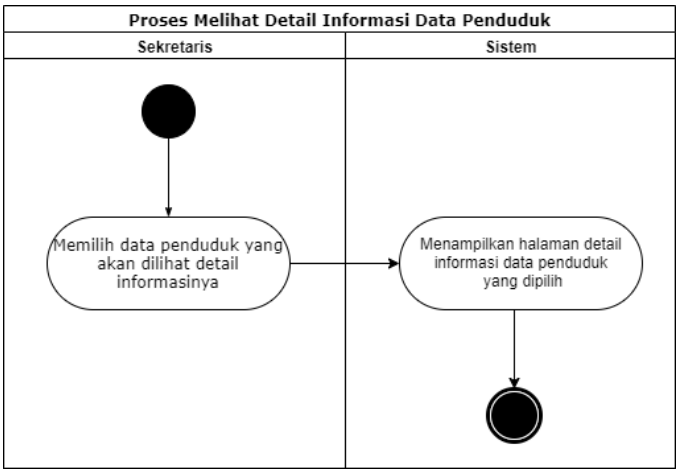
d. Proses mencari data penduduk



Gambar 4. 37 *Activity* diagram untuk proses pencarian data penduduk

Gambar 4.37 mengilustrasikan aktivitas yang terjadi ketika sekretaris melakukan pencarian data penduduk. Untuk mengakses halaman ini sekretaris harus terlebih dahulu membuka halaman data penduduk kemudian sistem akan menampilkan kotak pencarian untuk melakukan pencarian data penduduk. Selanjutnya sekretaris memasukkan kata kunci dari data penduduk yang akan dicari. Lalu sistem akan mengeksekusi *query* yang diminta oleh sekretaris dengan melakukan pengecekan pada *database*. Kemudian sistem akan menampilkan hasil pencarian yang sesuai dengan permintaan sekretaris pada tabel daftar penduduk.

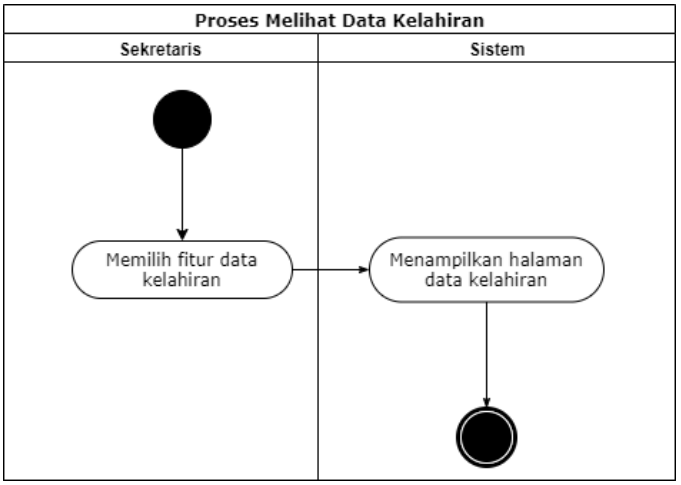
e. Proses melihat detail informasi data penduduk



Gambar 4. 38 *Activity* diagram untuk proses melihat detail informasi data penduduk

Gambar 4.38 mengilustrasikan aktivitas yang terjadi ketika sekretaris memilih fitur untuk melihat informasi data penduduk. Untuk mengakses halaman ini sekretaris harus terlebih dahulu membuka halaman data penduduk. Selanjutnya sekretaris harus memilih data penduduk yang akan dilihat rincian informasinya dan sistem akan menampilkan rincian informasi tentang data penduduk yang dipilih.

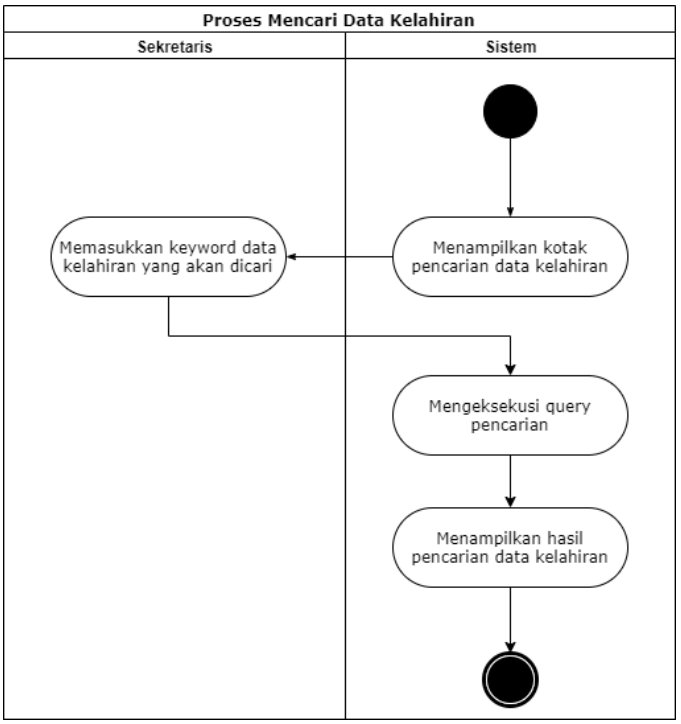
f. Proses melihat data kelahiran



Gambar 4. 39 *Activity* diagram untuk proses melihat data kelahiran

Gambar 4.39 mengilustrasikan aktivitas yang terjadi ketika admin memilih fitur untuk melihat data kelahiran. Ketika admin mengakses halaman data kelahiran maka sistem akan menampilkan daftar data kelahiran yang tersimpan di sistem, jika belum terdapat data, maka sistem akan menampilkan keterangan belum ada data.

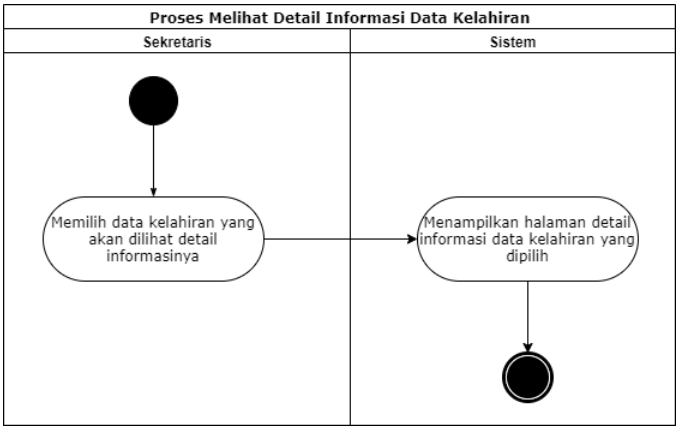
g. Proses mencari data kelahiran



Gambar 4. 40 *Activity* diagram untuk proses pencarian data kelahiran

Gambar 4.40 mengilustrasikan aktivitas yang terjadi ketika sekretaris melakukan pencarian data kelahiran. Untuk mengakses halaman ini sekretaris harus terlebih dahulu membuka halaman data kelahiran kemudian sistem akan menampilkan kotak pencarian untuk melakukan pencarian data kelahiran. Selanjutnya sekretaris memasukkan kata kunci dari data kelahiran yang akan dicari. Lalu sistem akan mengeksekusi *query* yang diminta oleh sekretaris dengan melakukan pengecekan pada *database*. Kemudian sistem akan menampilkan hasil pencarian yang sesuai dengan permintaan sekretaris pada tabel daftar kelahiran.

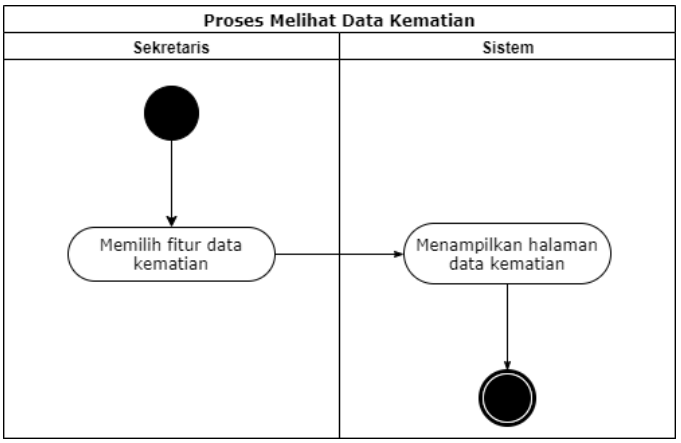
h. Proses melihat detail informasi data kelahiran



Gambar 4. 41 *Activity* diagram untuk proses melihat detail informasi data kelahiran

Gambar 4.41 mengilustrasikan aktivitas yang terjadi ketika sekretaris memilih fitur untuk melihat informasi data kelahiran. Untuk mengakses halaman ini sekretaris harus terlebih dahulu membuka halaman data kelahiran. Selanjutnya sekretaris harus memilih data kelahiran yang akan dilihat rincian informasinya dan sistem akan menampilkan rincian informasi tentang data kelahiran yang dipilih.

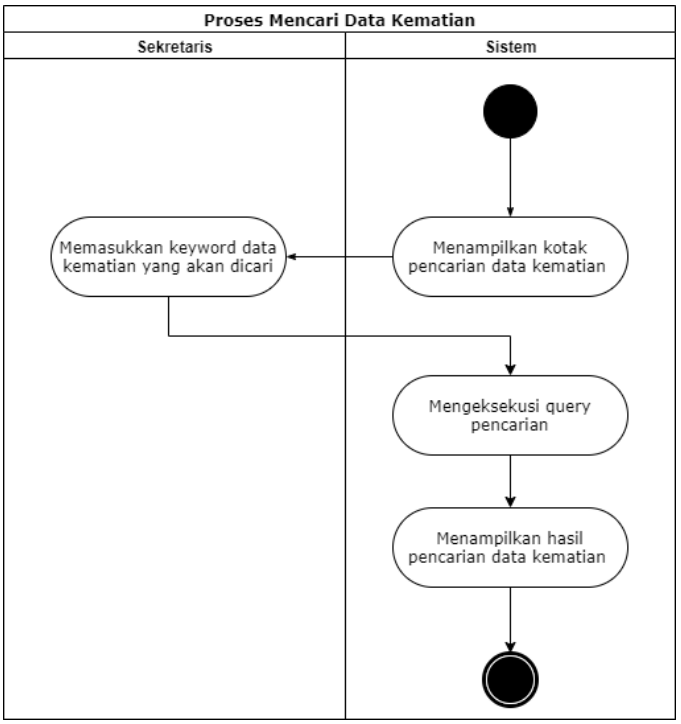
i. Proses melihat data kematian



Gambar 4. 42 *Activity* diagram untuk proses melihat halaman data kematian

Gambar 4.42 mengilustrasikan aktivitas yang terjadi ketika admin memilih fitur untuk melihat data kematian. Ketika admin mengakses halaman data kematian maka sistem akan menampilkan daftar data kematian yang tersimpan di sistem, jika belum terdapat data, maka sistem akan menampilkan keterangan belum ada data.

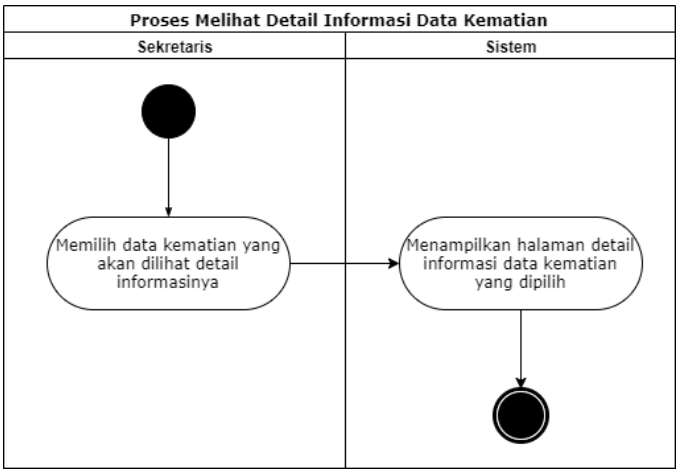
j. Proses mencari data kematian



Gambar 4. 43 *Activity* diagram untuk proses pencarian data kematian

Gambar 4.43 mengilustrasikan aktivitas yang terjadi ketika sekretaris melakukan pencarian data kematian. Untuk mengakses halaman ini sekretaris harus terlebih dahulu membuka halaman data kematian kemudian sistem akan menampilkan kotak pencarian untuk melakukan pencarian data kematian. Selanjutnya sekretaris memasukkan kata kunci dari data kematian yang akan dicari. Lalu sistem akan mengeksekusi *query* yang diminta oleh sekretaris dengan melakukan pengecekan pada *database*. Kemudian sistem akan menampilkan hasil pencarian yang sesuai dengan permintaan sekretaris pada tabel daftar kematian.

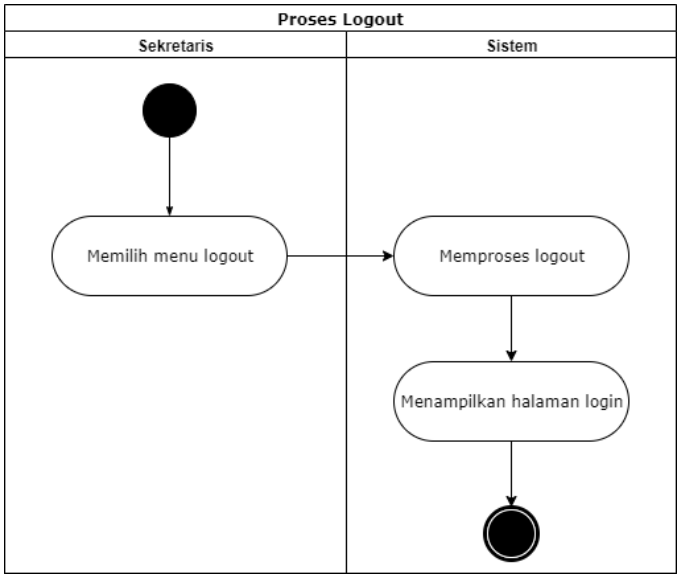
k. Proses melihat detail informasi data kematian



Gambar 4. 44 *Activity* diagram untuk proses melihat detail informasi data kematian

Gambar 4.44 mengilustrasikan aktivitas yang terjadi ketika sekretaris memilih fitur untuk melihat informasi data kematian. Untuk mengakses halaman ini sekretaris harus terlebih dahulu membuka halaman data kematian. Selanjutnya sekretaris harus memilih data kematian yang akan dilihat rincian informasinya dan sistem akan menampilkan rincian informasi tentang data kematian yang dipilih.

l. Proses *logout*



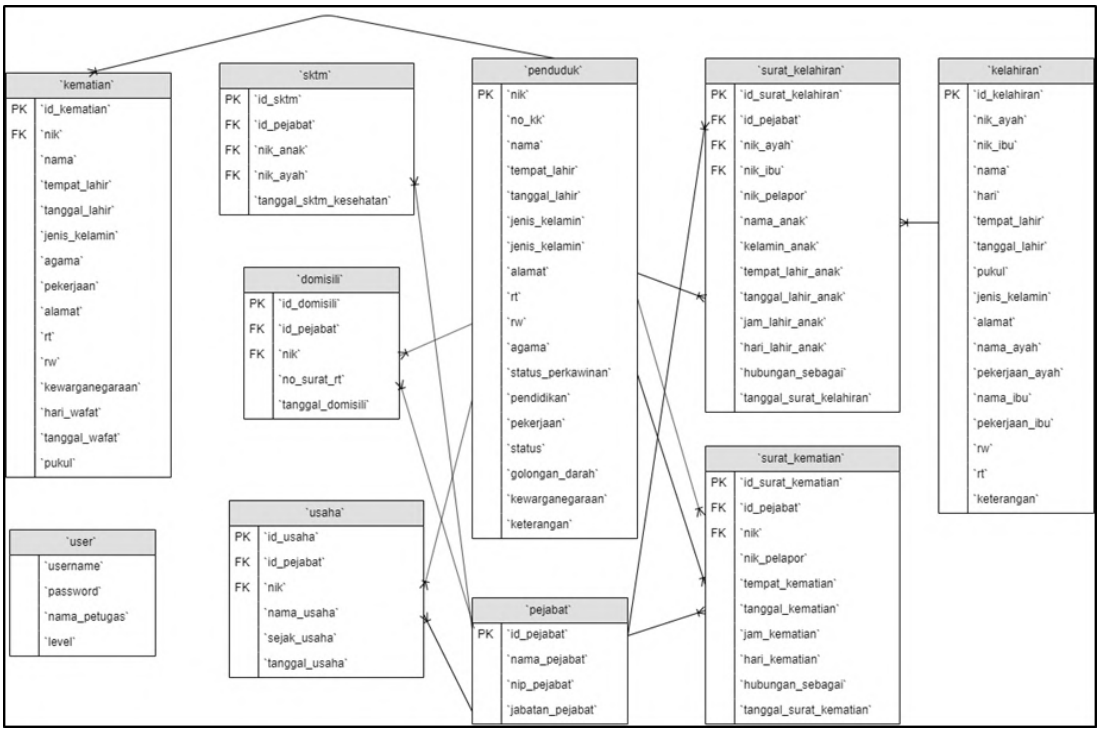
Gambar 4. 45 *Activity* diagram untuk proses *logout*

Gambar 4.45 mengilustrasikan aktivitas untuk keluar dari sistem melalui proses *logout* yang dilakukan oleh *user* sekretaris. Ketika sekretaris memilih menu *logout*, sistem akan memproses *logout* dengan cara mengakhiri *session* dengan perintah *destroy* yang akan menghapus file *session* dari *server*. Ketika proses *logout* berhasil dan *session* berakhir maka sekretaris akan diarahkan ke

halaman *login*.

4.2.3 *Entity Relationship* Diagram (ERD)

Berikut adalah ERD (*Entity Relationship* diagram) dari sistem informasi pengelolaan data penduduk Kantor Kelurahan Karang Baru



Gambar 4. 46 ERD sistem informasi pengelolaan data penduduk Kantor Kelurahan Karang Baru

Dari rancangan ERD di atas, diperoleh 10 entitas yang kemudian selanjutnya menjadi tabel *database* sistem. Adapun tabel-tabel tersebut antara lain sebagai berikut:

a. Tabel domisili

Tabel 4. 1 Struktur tabel domisili

Name field	Type	Null	Default
id_domisili	int (5)	No	None
id_pejabat	int (5)	No	None
nik	varchar (16)	No	None
no_surat_rt	varchar (25)	No	None
tanggal_domisili	date	No	None

b. Tabel kelahiran

Tabel 4. 2 Struktur tabel kelahiran

Name field	Type	Null	Default
id_kelahiran	int (5)	No	None
nama	varchar (35)	No	None
hari	varchar (8)	No	None
tempat_lahir	varchar (20)	No	None
tanggal_lahir	date	No	None
pukul	time	No	None
jenis_kelamin	varchar (10)	No	None
alamat	varchar (35)	No	None
nik_ayah	varchar (16)	No	None
nama_ayah	varchar (20)	No	None
pekerjaan_ayah	varchar (15)	No	None
nik_ibu	varchar (16)	No	None
nama_ibu	varchar (20)	No	None
pekerjaan_ibu	varchar (15)	No	None
rw	varchar (5)	No	None
rt	varchar (5)	No	None
keterangan	varchar (50)	No	None

c. Tabel kematian

Tabel 4. 3 Struktur tabel kematian

Name field	Type	Null	Default
id_kematian	int (5)	No	None
nik	varchar (16)	No	None
nama	varchar (35)	No	None
tempat_lahir	varchar (20)	No	None
tanggal_lahir	varchar (10)	No	None
jenis_kelamin	varchar (10)	No	None
agama	varchar (10)	No	None
pekerjaan	varchar (15)	No	None
alamat	varchar (35)	No	None

rt	varchar (5)	No	None
rw	varchar (5)	No	None
kewarganegaraan	varchar (20)	No	None
hari_wafat	varchar (8)	No	None
tanggal_wafat	date	No	None
pukul	time	No	None

d. Tabel pejabat

Tabel 4. 4 Struktur tabel pejabat

Name field	Type	Null	Default
id_pejabat	int (5)	No	None
nama_pejabat	varchar (35)	No	None
nip_pejabat	varchar (18)	No	None
jabatan_pejabat	varchar (25)	No	None

e. Tabel penduduk

Tabel 4. 5 Struktur tabel penduduk

Name field	Type	Null	Default
nik	varchar (16)	No	None
no_kk	varchar (16)	No	None
nama	varchar (35)	No	None
tempat_lahir	varchar (20)	No	None
tanggal_lahir	date	No	None
jenis_kelamin	varchar (10)	No	None
alamat	varchar (35)	No	None
rt	varchar (5)	No	None
rw	varchar (5)	No	None
agama	varchar (10)	No	None
status_perkawinan	varchar (15)	No	None
pendidikan	varchar (20)	No	None
pekerjaan	varchar (15)	No	None
status	varchar (20)	No	None
golongan_darah	varchar (2)	No	None
kewarganegaraan	varchar (20)	No	None
keterangan	varchar (50)	No	None

f. Tabel sktm

Tabel 4. 6 Struktur tabel sktm

Name field	Type	Null	Default
id_sktm	int (5)	No	None
id_pejabat	int (5)	No	None
nik_anak	varchar (16)	No	None
nik_ayah	varchar (16)	No	None
tanggal_sktm	date	No	None

g. Tabel surat_kelahiran

Tabel 4. 7 Struktur tabel surat_kelahiran

Name field	Type	Null	Default
id_surat_kelahiran	int (5)	No	None
id_pejabat	int (5)	No	None
nik_ayah	varchar (16)	No	None
nik_ibu	varchar (16)	No	None
nik_pelapor	varchar (16)	No	None
nama_anak	varchar (35)	No	None
kelamin_anak	varchar (10)	No	None
tempat_lahir_anak	varchar (20)	No	None
tanggal_lahir_anak	date	No	None
jam_lahir_anak	time	No	None
hari_lahir_anak	varchar (8)	No	None
hubungan_sebagai	varchar (15)	No	None
tanggal_surat_kelahiran	date	No	None

h. Tabel surat_kematian

Tabel 4. 8 Struktur tabel surat_kematian

Name field	Type	Null	Default
id_surat_kematian	int (5)	No	None
id_pejabat	int (5)	No	None
nik	varchar (16)	No	None
nik_pelapor	varchar (16)	No	None
tempat_kematian	varchar (20)	No	None
tanggal_kematian	date	No	None

jam_kematian	time	No	None
hari_kematian	varchar (8)	No	None
hubungan_sebagai	varchar (15)	No	None
tanggal_surat_kematian	date	No	None

i. Tabel usaha

Tabel 4. 9 Struktur tabel usaha

Name field	Type	Null	Default
id_usaha	int (5)	No	None
id_pejabat	int (5)	No	None
nik	varchar (16)	No	None
nama_usaha	varchar (35)	No	None
sejak_usaha	varchar (4)	No	None
tanggal_usaha	date	No	None

j. Tabel *user*

Tabel 4. 10 Struktur tabel *user*

Name field	Type	Null	Default
username	varchar (25)	No	None
password	varchar (32)	No	None
nama_petugas	varchar (35)	No	None
level	enum(‘admin’, ‘sekretaris’)	No	None

4.3 Implementasi Sistem

Pada Implementasi sistem, rancangan yang sudah dibuat diimplementasikan kedalam bentuk program dengan menggunakan bahasa pemrograman. Implementasi sistem pada pembahasan ini mencakup implementasi *database* sistem dan implementasi *interface*. Berikut adalah implementasi sistem dari sistem informasi pengelolaan data kependudukan Kelurahan Karang Baru.

4.3.1 Implementasi *Database* Sistem

a. Implementasi tabel domisili

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
☐	1	id_domisili	int(5)			No	None		AUTO_INCREMENT	Change Drop More
☐	2	id_pejabat	int(5)			No	None			Change Drop More
☐	3	nik	varchar(16)	latin1_swedish_ci		No	None			Change Drop More
☐	4	no_surat_rt	varchar(25)	latin1_swedish_ci		No	None			Change Drop More
☐	5	tanggal_domisili	date			No	None			Change Drop More

Gambar 4. 47 Implementasi tabel domisili

Gambar 4.47 merupakan implementasi dari tabel domisili. Tabel domisili merupakan tabel yang berisi data mengenai Surat Keterangan Domisili. Atribut pada tabel ini antara lain id_domisili, id_pejabat, nik, no_surat_rt dan tanggal domisili. Pada tabel ini atribut id_domisili bertindak sebagai *primary key* dan id_pejabat & nik sebagai *foreign key*. Tabel ini berelasi dengan tabel pejabat dan tabel penduduk.

b. Implementasi tabel kelahiran

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
☐	1	id_kelahiran	int(5)			No	None		AUTO_INCREMENT	Change Drop More
☐	2	nama	varchar(35)	latin1_swedish_ci		No	None			Change Drop More
☐	3	hari	varchar(8)	latin1_swedish_ci		No	None			Change Drop More
☐	4	tempat_lahir	varchar(20)	latin1_swedish_ci		No	None			Change Drop More
☐	5	tanggal_lahir	date			No	None			Change Drop More
☐	6	pukul	time			No	None			Change Drop More
☐	7	jenis_kelamin	varchar(10)	latin1_swedish_ci		No	None			Change Drop More
☐	8	alamat	varchar(35)	latin1_swedish_ci		No	None			Change Drop More
☐	9	nik_ayah	varchar(16)	latin1_swedish_ci		No	None			Change Drop More
☐	10	nama_ayah	varchar(20)	latin1_swedish_ci		No	None			Change Drop More
☐	11	pekerjaan_ayah	varchar(15)	latin1_swedish_ci		No	None			Change Drop More
☐	12	nik_ibu	varchar(16)	latin1_swedish_ci		No	None			Change Drop More
☐	13	nama_ibu	varchar(20)	latin1_swedish_ci		No	None			Change Drop More
☐	14	pekerjaan_ibu	varchar(15)	latin1_swedish_ci		No	None			Change Drop More
☐	15	rw	varchar(5)	latin1_swedish_ci		No	None			Change Drop More
☐	16	rt	varchar(5)	latin1_swedish_ci		No	None			Change Drop More
☐	17	keterangan	varchar(50)	latin1_swedish_ci		No	None			Change Drop More

Gambar 4. 48 Implementasi tabel kelahiran

Gambar 4.48 merupakan implementasi dari tabel kelahiran. Tabel kelahiran merupakan tabel yang berisi data mengenai data kelahiran di Kelurahan Karang Baru. Pada tabel ini terdapat 17 atribut dimana atribut id_domisili bertindak sebagai *primary key*.

c. Implementasi tabel kematian

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id_kematian	int(5)			No	None		AUTO_INCREMENT	Change Drop More
2	nik	varchar(16)	latin1_swedish_ci		No	None			Change Drop More
3	nama	varchar(35)	latin1_swedish_ci		No	None			Change Drop More
4	tempat_lahir	varchar(20)	latin1_swedish_ci		No	None			Change Drop More
5	tanggal_lahir	varchar(10)	latin1_swedish_ci		No	None			Change Drop More
6	jenis_kelamin	varchar(10)	latin1_swedish_ci		No	None			Change Drop More
7	agama	varchar(10)	latin1_swedish_ci		No	None			Change Drop More
8	pekerjaan	varchar(15)	latin1_swedish_ci		No	None			Change Drop More
9	alamat	varchar(35)	latin1_swedish_ci		No	None			Change Drop More
10	rt	varchar(5)	latin1_swedish_ci		No	None			Change Drop More
11	rw	varchar(5)	latin1_swedish_ci		No	None			Change Drop More
12	kewarganegaraan	varchar(20)	latin1_swedish_ci		No	None			Change Drop More
13	hari_wafat	varchar(8)	latin1_swedish_ci		No	None			Change Drop More
14	tanggal_wafat	date			No	None			Change Drop More
15	pukul	time			No	None			Change Drop More

Gambar 4. 49 Implementasi tabel kematian

Gambar 4.49 merupakan implementasi dari tabel kematian. Tabel kelahiran merupakan tabel yang berisi data mengenai data kematian di Kelurahan Karang Baru. Pada tabel ini terdapat 15 atribut dimana atribut nik bertindak sebagai *primary key*. Tabel ini berelasi dengan tabel penduduk.

d. Implementasi tabel pejabat

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id_pejabat	int(5)			No	None			Change Drop More
2	nama_pejabat	varchar(35)	latin1_swedish_ci		No	None			Change Drop More
3	nip_pejabat	varchar(18)	latin1_swedish_ci		No	None			Change Drop More
4	jabatan_pejabat	varchar(25)	latin1_swedish_ci		No	None			Change Drop More

Gambar 4. 50 Implementasi tabel pejabat

Gambar 4.50 merupakan implementasi dari tabel pejabat. Tabel pejabat merupakan tabel yang berisi data dari pejabat yang ada di Kantor Kelurahan Karang Baru seperti Lurah, Kepala Seksi dan Sekretaris. Atribut pada tabel ini antara lain adalah id_pejabat, nama_pejabat, nip_pejabat dan jabatan_pejabat. Pada tabel ini atribut atribut id_pejabat bertindak sebagai *primary key*. Tabel ini berelasi dengan tabel domisili, tabel surat kelahiran, tabel surat_kematian, tabel sktm dan tabel usaha.

e. Implementasi tabel penduduk

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	nik	varchar(16)	latin1_swedish_ci		No	None			Change Drop More
2	no_kk	varchar(16)	latin1_swedish_ci		No	None			Change Drop More
3	nama	varchar(35)	latin1_swedish_ci		No	None			Change Drop More
4	tempat_lahir	varchar(20)	latin1_swedish_ci		No	None			Change Drop More
5	tanggal_lahir	date			No	None			Change Drop More
6	jenis_kelamin	varchar(10)	latin1_swedish_ci		No	None			Change Drop More
7	alamat	varchar(35)	latin1_swedish_ci		No	None			Change Drop More
8	rt	varchar(5)	latin1_swedish_ci		No	None			Change Drop More
9	rw	varchar(5)	latin1_swedish_ci		No	None			Change Drop More
10	agama	varchar(10)	latin1_swedish_ci		No	None			Change Drop More
11	status_perkawinan	varchar(15)	latin1_swedish_ci		No	None			Change Drop More
12	pendidikan	varchar(20)	latin1_swedish_ci		No	None			Change Drop More
13	pekerjaan	varchar(15)	latin1_swedish_ci		No	None			Change Drop More
14	status	varchar(20)	latin1_swedish_ci		No	None			Change Drop More
15	golongan_darah	varchar(2)	latin1_swedish_ci		No	None			Change Drop More
16	kewarganegaraan	varchar(20)	latin1_swedish_ci		No	None			Change Drop More
17	keterangan	varchar(50)	latin1_swedish_ci		No	None			Change Drop More

Gambar 4. 51 Implementasi tabel penduduk

Gambar 4.51 merupakan implementasi dari tabel penduduk. Tabel penduduk merupakan tabel yang berisi data dari penduduk Kelurahan Karang Baru yang sudah memiliki NIK. Pada tabel ini terdapat 17 atribut dimana atribut nik bertindak sebagai *primary key*. Tabel ini berelasi dengan tabel domisili, tabel kematian, tabel penduduk, tabel sktm, tabel surat kelahiran, tabel surat_kematian dan tabel usaha.

f. Implementasi tabel sktm

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id_sktm	int(5)			No	None		AUTO_INCREMENT	Change Drop More
2	id_pejabat	int(5)			No	None			Change Drop More
3	nik_anak	varchar(16)	latin1_swedish_ci		No	None			Change Drop More
4	nik_ayah	varchar(16)	latin1_swedish_ci		No	None			Change Drop More
5	tanggal_sktm_kesehatan	date			No	None			Change Drop More

Gambar 4. 52 Implementasi tabel sktm

Gambar 4.52 merupakan implementasi dari tabel sktm. Tabel sktm merupakan tabel yang berisi data dari layanan Surat Keterangan Tidak Mampu. Atribut pada tabel ini antara lain adalah id_sktm, id_pejabat, nik_anak, nik_ayah dan tanggal_sktm. Pada tabel ini atribut id_sktm bertindak sebagai *primary key* dan atribut id_pejabat bertindak sebagai *foreign key*. Tabel ini berelasi dengan tabel pejabat.

g. Implementasi tabel surat_kelahiran

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id_surat_kelahiran	int(5)			No	None		AUTO_INCREMENT	Change Drop More
2	id_pejabat	int(5)			No	None			Change Drop More
3	nik_ayah	varchar(16)	latin1_swedish_ci		No	None			Change Drop More
4	nik_ibu	varchar(16)	latin1_swedish_ci		No	None			Change Drop More
5	nik_pelapor	varchar(16)	latin1_swedish_ci		No	None			Change Drop More
6	nama_anak	varchar(35)	latin1_swedish_ci		No	None			Change Drop More
7	kelamin_anak	varchar(10)	latin1_swedish_ci		No	None			Change Drop More
8	tempat_lahir_anak	varchar(20)	latin1_swedish_ci		No	None			Change Drop More
9	tanggal_lahir_anak	date			No	None			Change Drop More
10	jam_lahir_anak	time			No	None			Change Drop More
11	hari_lahir_anak	varchar(8)	latin1_swedish_ci		No	None			Change Drop More
12	hubungan_sebagai	varchar(15)	latin1_swedish_ci		No	None			Change Drop More
13	tanggal_surat_kelahiran	date			No	None			Change Drop More

Gambar 4. 53 Implementasi tabel surat_kelahiran

Gambar 4.53 merupakan implementasi dari tabel surat_kelahiran. Tabel surat_kelahiran merupakan tabel yang berisi data dari layanan Surat Kelahiran. Pada tabel ini terdapat 13 atribut dimana atribut id_surat_kelahiran bertindak sebagai *primary key* dan atribut id_pejabat, nik_ayah, nik_ibu dan nik_pelapor bertindak sebagai *foreign key*. Tabel ini berelasi dengan tabel pejabat, tabel kelahiran dan tabel penduduk.

h. Implementasi tabel surat_kematian

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id_surat_kematian	int(5)			No	None		AUTO_INCREMENT	Change Drop More
2	id_pejabat	int(5)			No	None			Change Drop More
3	nik	varchar(16)	latin1_swedish_ci		No	None			Change Drop More
4	nik_pelapor	varchar(16)	latin1_swedish_ci		No	None			Change Drop More
5	tempat_kematian	varchar(20)	latin1_swedish_ci		No	None			Change Drop More
6	tanggal_kematian	date			No	None			Change Drop More
7	jam_kematian	time			No	None			Change Drop More
8	hari_kematian	varchar(8)	latin1_swedish_ci		No	None			Change Drop More
9	hubungan_sebagai	varchar(15)	latin1_swedish_ci		No	None			Change Drop More
10	tanggal_surat_kematian	date			No	None			Change Drop More

Gambar 4. 54 Implementasi tabel surat_kematian

Gambar 4.54 merupakan implementasi dari tabel surat_kematian. Tabel surat_kematian merupakan tabel yang berisi data dari layanan Surat Kematian. Pada tabel ini terdapat 11 atribut dimana atribut id_surat_kematian bertindak sebagai *primary key* dan atribut id_pejabat dan nik bertindak sebagai *foreign key*. Tabel ini berelasi dengan tabel pejabat dan tabel penduduk.

i. Implementasi tabel usaha

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id_usaha	int(5)			No	None		AUTO_INCREMENT	Change Drop More
2	id_pejabat	int(5)			No	None			Change Drop More
3	nik	varchar(16)	latin1_swedish_ci		No	None			Change Drop More
4	nama_usaha	varchar(35)	latin1_swedish_ci		No	None			Change Drop More
5	sejak_usaha	varchar(4)	latin1_swedish_ci		No	None			Change Drop More
6	tanggal_usaha	date			No	None			Change Drop More

Gambar 4. 55 Implementasi tabel usaha

Gambar 4.55 merupakan implementasi dari tabel usaha. Tabel usaha merupakan tabel yang berisi data dari layanan Surat Keterangan Usaha. Atribut pada tabel ini antara lain adalah id_usaha, id_pejabat, nik, nama_usaha, sejak_usaha dan tanggal_usaha. Pada tabel ini atribut id_usaha bertindak sebagai *primary key* dan atribut id_pejabat dan nik bertindak sebagai *foreign key*. Tabel ini berelasi dengan tabel pejabat dan tabel penduduk.

j. Implementasi tabel *user*

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	username	varchar(25)	latin1_swedish_ci		No	None			Change Drop More
2	password	varchar(32)	latin1_swedish_ci		No	None			Change Drop More
3	nama_petugas	varchar(35)	latin1_swedish_ci		No	None			Change Drop More
4	level	enum('admin', 'sekretaris')	latin1_swedish_ci		No	None			Change Drop More

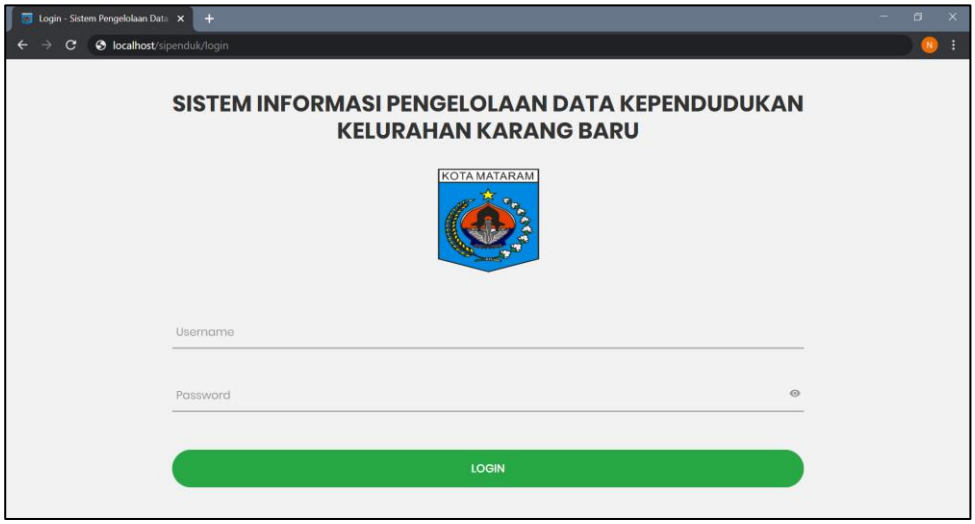
Gambar 4. 56 Implementasi tabel *user*

Gambar 4.56 merupakan implementasi dari tabel *user*. Tabel *user* merupakan tabel yang berisi data dari pengguna sistem informasi pengelolaan data penduduk Kantor Kelurahan Karang Baru. Atribut pada tabel ini antara lain adalah *username*, *password*, nama_petugas dan level. Atribut level berfungsi untuk memberikan tingkatan hak akses bagi pengguna, dimana terdapat dua hak akses yaitu admin dan sekretaris. Pada tabel ini atribut username bertindak sebagai *primary key*.

4.3.2 Implementasi *Interface* Sistem

Berikut adalah implementasi tampilan *(interface*) program dari sistem informasi pengelolaan data penduduk Kantor Kelurahan Karang Baru.

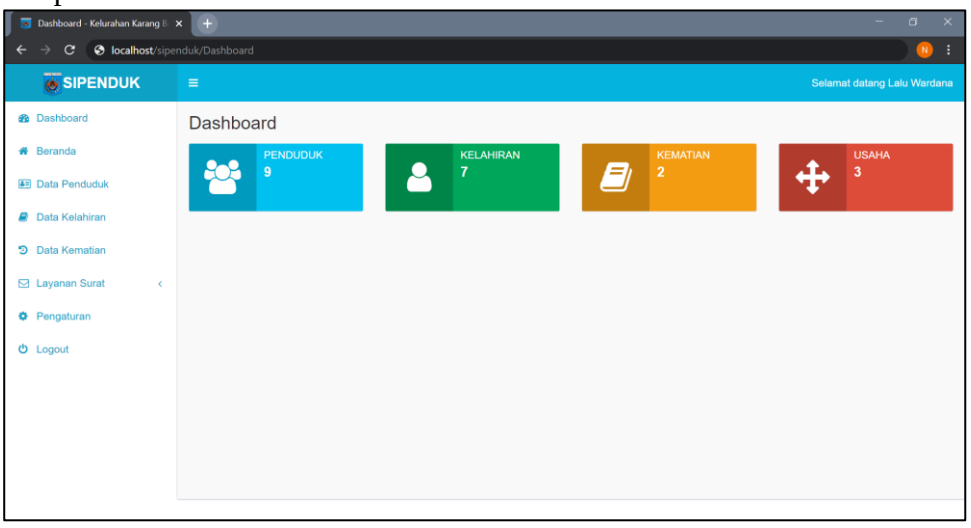
1. Tampilan antarmuka untuk aktor admin :
   a. Tampilan *login*



Gambar 4. 57 Implementasi halaman *login user*

Gambar 4.57 merupakan implementasi halaman *login* dari dari sistem informasi pengelolaan data penduduk Kantor Kelurahan Karang Baru. Halaman ini merupakan tampilan yang akan pertama kali muncul ketika actor admin mengakses program. Untuk masuk ke dalam sistem admin perlu untuk memasukkan username dan password yang sesuai dengan data di *database*.

   b. Tampilan halaman *dashboard*

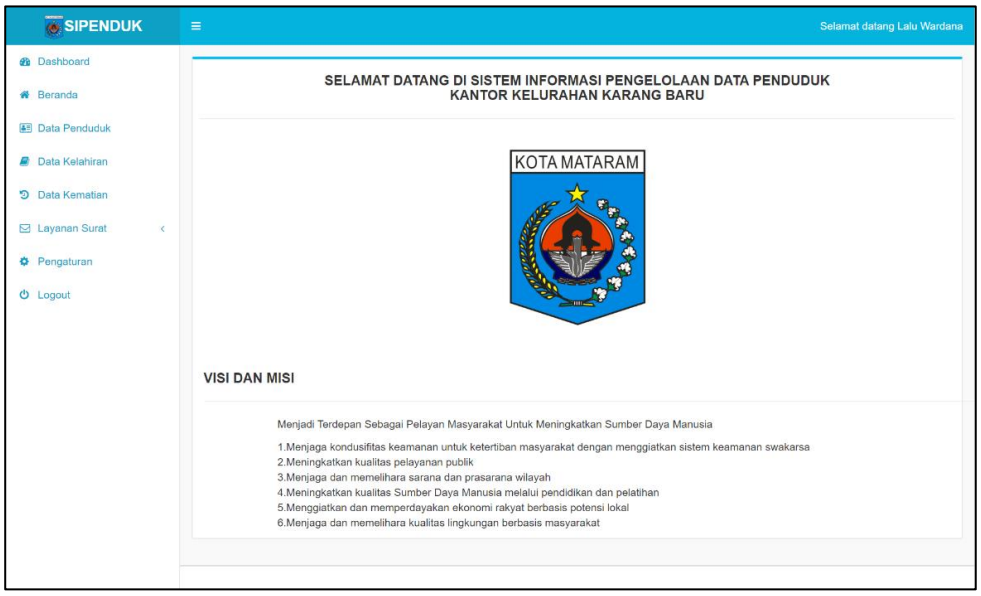


Gambar 4. 58 Implementasi halaman *dashboard*

Gambar 4.58 merupakan implementasi dari halaman *dashboard*.

Halaman *dashboard* berfungsi untuk menampilkan informasi mengenai jumlah data penduduk, jumlah data kelahiran, jumlah data kematian dan jumlah data usaha.

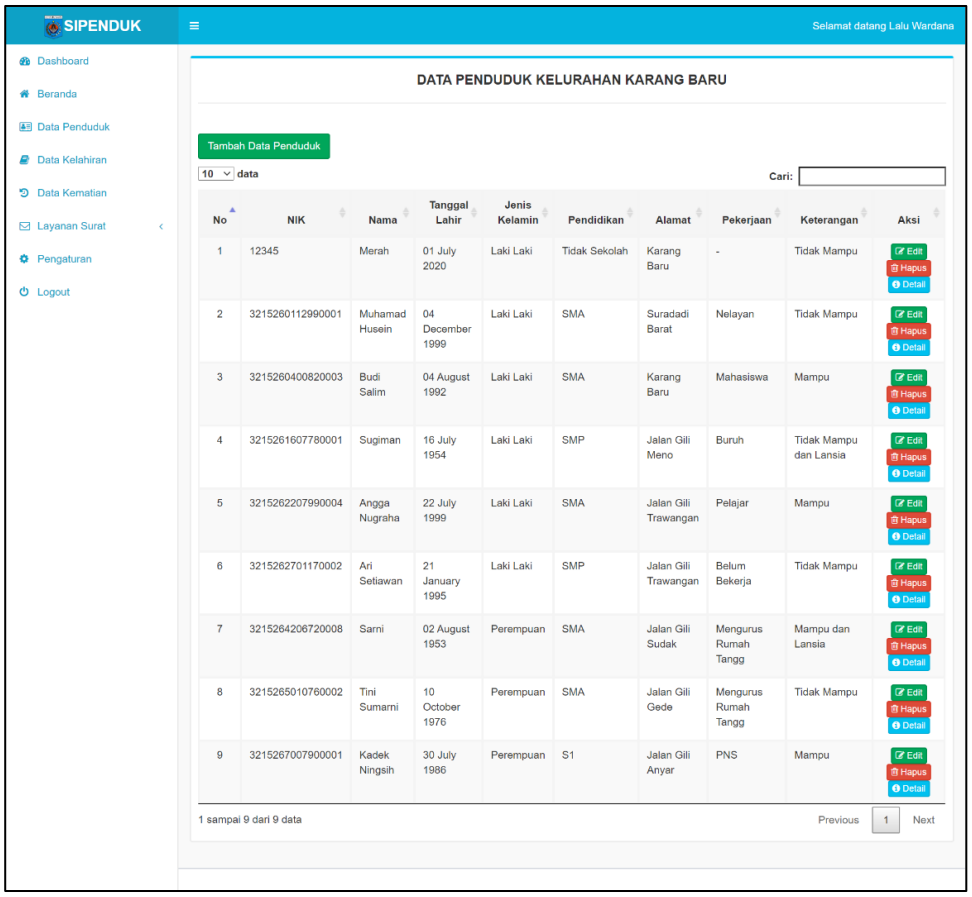
c. Tampilan halaman beranda



Gambar 4. 59 Implementasi halaman beranda.

Gambar 4.59 merupakan implementasi dari halaman beranda. Halaman beranda merupakan halaman yang akan pertama kali ditampilkan setelah admin berhasil melakukan *login* atau masuk ke dalam sistem. Halaman beranda berfungsi untuk menampilkan informasi tentang visi dan misi dari Kelurahan Karang Baru.

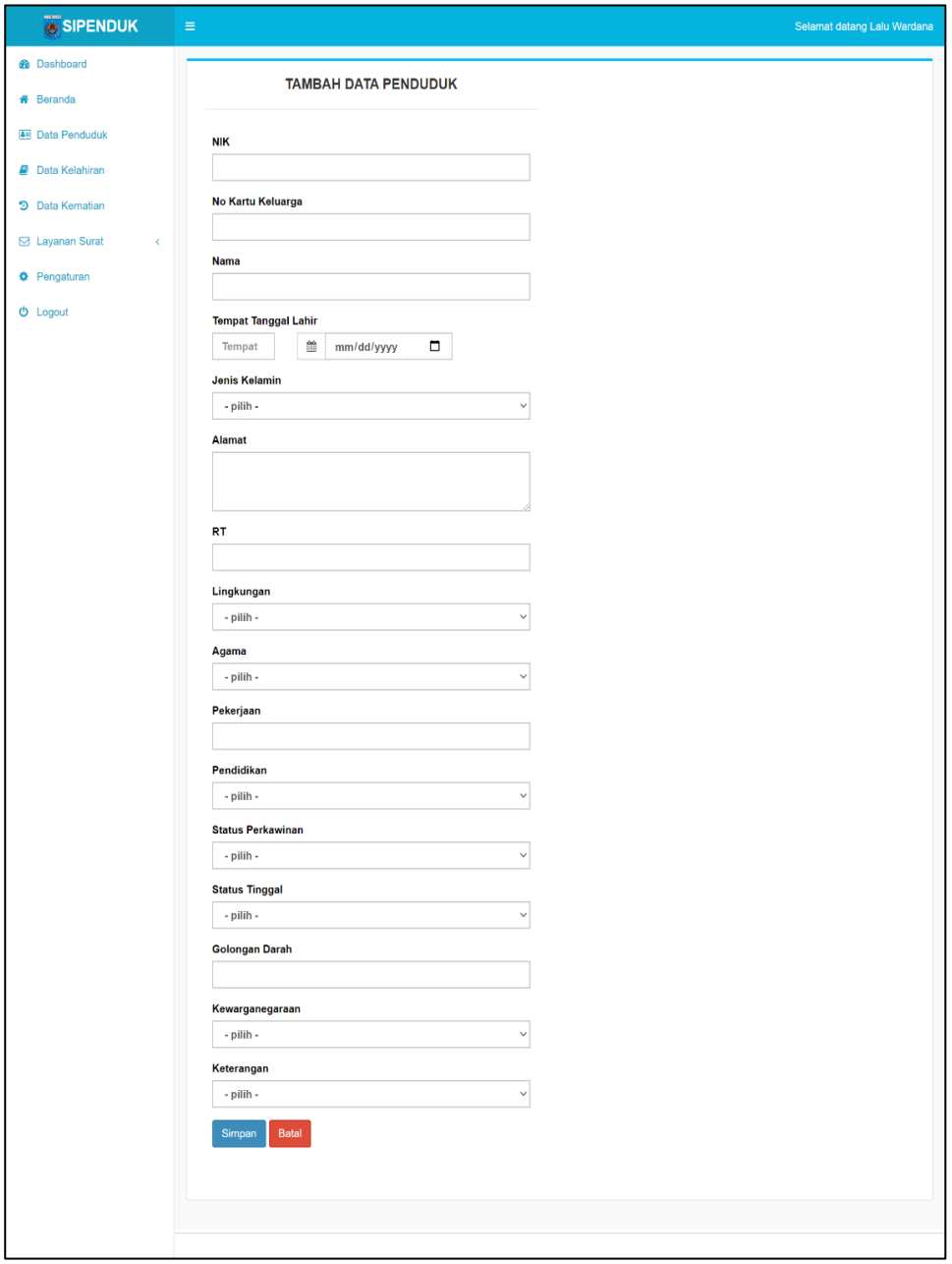
d. Tampilan halaman data penduduk



Gambar 4. 60 Implementasi halaman data penduduk.

Gambar 4.60 merupakan implementasi dari halaman data penduduk. Halaman ini berfungsi untuk menampilkan daftar data penduduk Kelurahan Karang Baru. Pada halaman ini terdapat fitur tambah data penduduk yang akan mengarahkan admin ke halaman tambah data penduduk dan juga terdapat fitur cari data penduduk untuk memudahkan admin dalam menemukan data penduduk tertentu. Pada bagian aksi terdapat beberapa fungsi yaitu tombol untuk mengarahkan admin ke halaman edit data penduduk, tombol untuk melakukan hapus data penduduk dan tombol untuk mengarahkan admin ke halaman lihat rincian data penduduk. Pada bagian bawah halaman juga terdapat tombol untuk menuju ke halaman selanjutnya dari daftar data penduduk.

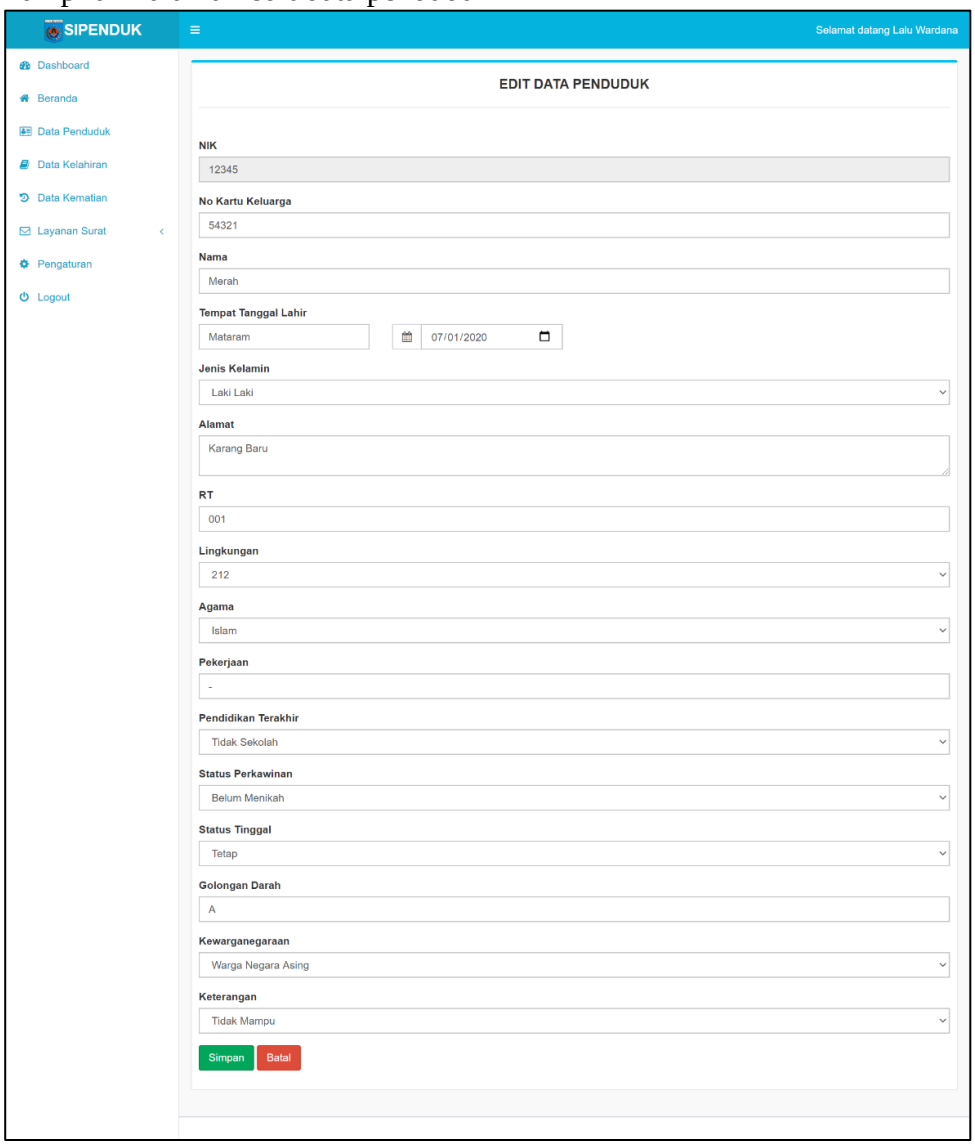
e. Tampilan halaman tambah data penduduk



Gambar 4. 61 Implementasi halaman tambah data penduduk

Gambar 4.61 merupakan tampilan *form* untuk admin melakukan tambah data penduduk, agar data dapat tersimpan di *database*, *form* harus diisi dengan lengkap sesuai dengan data yang diminta. Pada bagian bawah pojok kiri terdapat tombol untuk menyimpan data penduduk dan tombol untuk melakukan pembatalan tambah data penduduk jika tidak jadi melakukan tambah data penduduk. Setelah admin meng-klik tombol simpan maka sistem akan kembali ke halaman daftar data penduduk dan admin dapat melihat bahwa penambahan data yang dilakukan sudah berhasil.

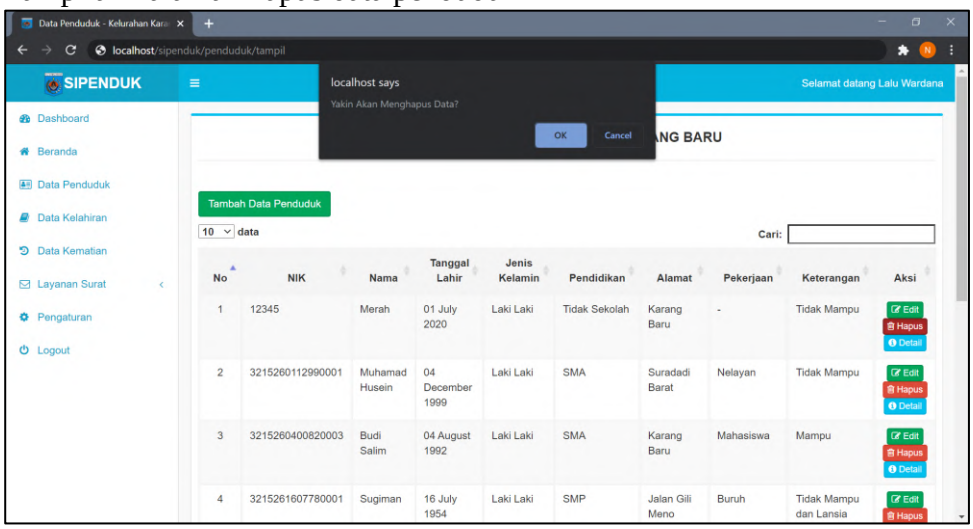
f. Tampilan halaman edit data penduduk



Gambar 4. 62 Implementasi halaman edit data penduduk

Gambar 4.62 merupakan tampilan *form* untuk admin melakukan edit data penduduk, dengan *form* ini admin dapat memperbarui data penduduk yang ada. Pada bagian bawah pojok kiri terdapat tombol untuk menyimpan perubahan data penduduk dan tombol untuk melakukan pembatalan edit data penduduk jika tidak jadi melakukan pembaruan data. Setelah admin meng-klik tombol simpan maka sistem akan kembali ke halaman daftar data penduduk dan admin dapat melihat bahwa pembaruan data sudah berhasil tersimpan.

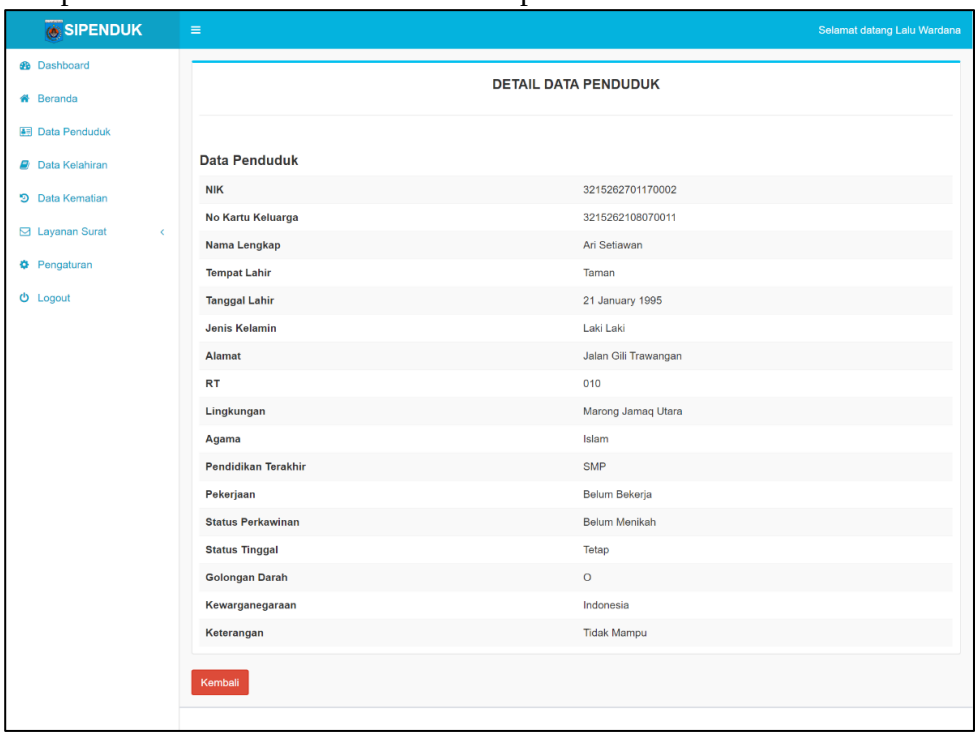
g. Tampilan halaman hapus data penduduk



Gambar 4. 63 Implementasi halaman hapus data penduduk

Gambar 4.63 merupakan tampilan halaman ketika admin akan melakukan hapus data penduduk. Setelah admin mengklik tombol hapus pada salah satu data penduduk sistem akan memunculkan dialog konfirmasi untuk penghapusan data. Jika admin meng-klik *OK* maka data penduduk yang dipilih akan otomatis terhapus dari daftar penduduk. Jika admin tidak jadi melakukan penghapusan data penduduk, admin dapat meng-klik pilihan *cancel* pada kotak dialog.

h. Tampilan halaman informasi detail data penduduk

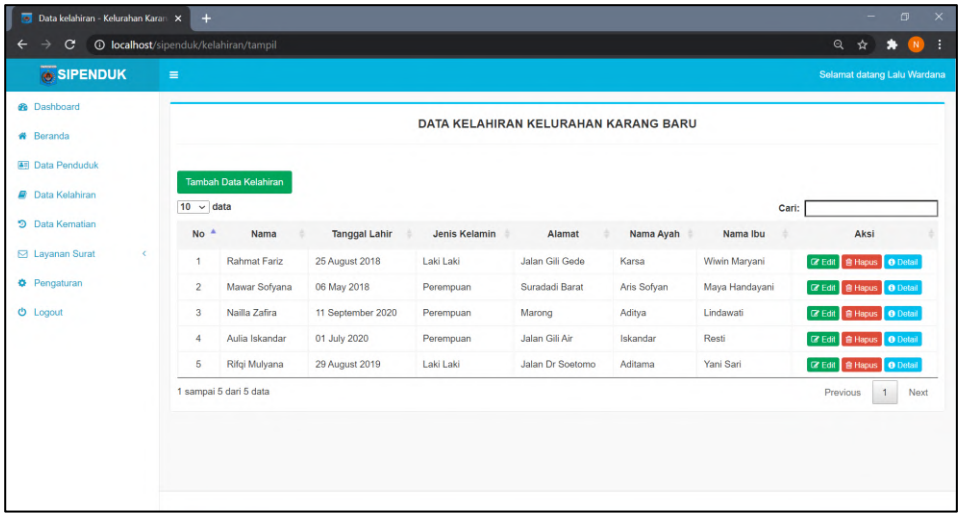


Gambar 4. 64 Implementasi halaman informasi detail data penduduk

Gambar 4.64 merupakan tampilan halaman ketika admin memilih

fitur untuk melihat detail data penduduk. Ketika admin meng-klik tombol detail pada salah satu data penduduk maka akan diarahkan ke halaman ini yang akan menampilkan rincian informasi dari data penduduk yang dipilih. Pada bagian bawah terdapat tombol kembali yang ketika di-klik akan mengarahkan admin kembali ke halaman daftar data penduduk.

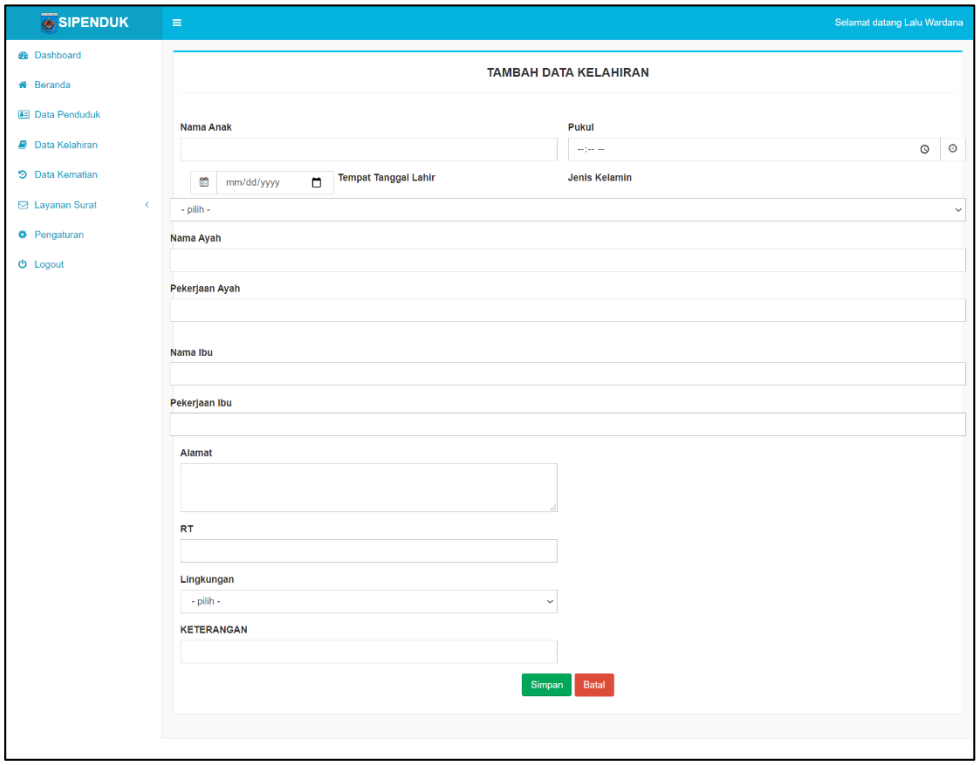
i. Tampilan halaman data kelahiran



Gambar 4. 65 Implementasi halaman data kelahiran

Gambar 4.65 merupakan implementasi dari halaman data kelahiran. Halaman ini berfungsi untuk menampilkan daftar data kelahiran Kelurahan Karang Baru. Pada halaman ini terdapat fitur tambah data kelahiran yang akan mengarahkan admin ke halaman tambah data kelahiran dan juga terdapat fitur cari data kelahiran untuk memudahkan admin dalam menemukan data kelahiran tertentu. Pada bagian aksi terdapat beberapa fungsi yaitu tombol untuk mengarahkan admin ke halaman edit data kelahiran, tombol untuk melakukan hapus data kelahiran dan tombol untuk mengarahkan admin ke halaman lihat rincian data kelahiran. Pada bagian bawah halaman juga terdapat tombol untuk menuju ke halaman selanjutnya dari daftar data kelahiran.

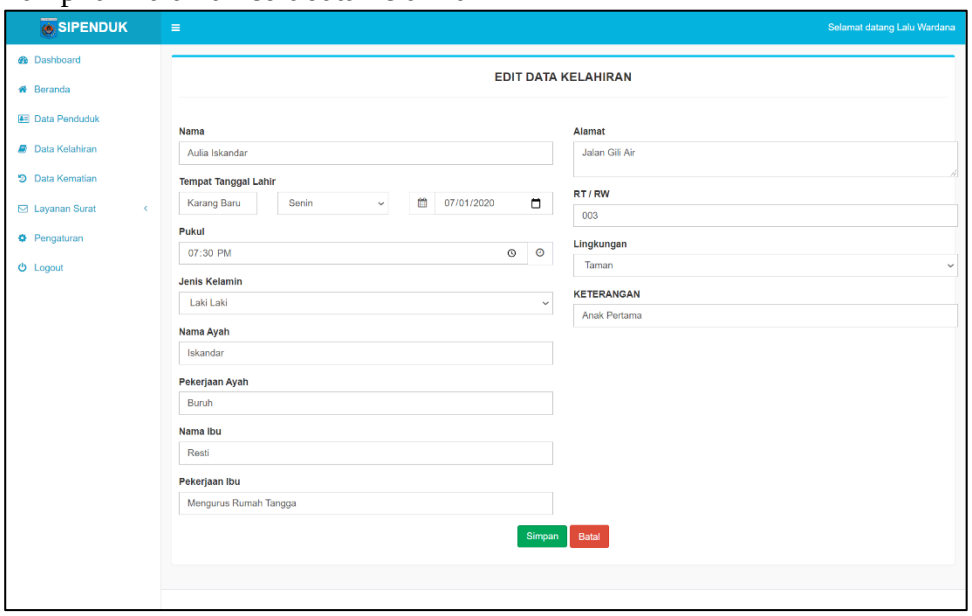
j. Tampilan halaman tambah data kelahiran



Gambar 4. 66 Implementasi halaman tambah data kelahiran

Gambar 4.66 merupakan tampilan *form* untuk admin melakukan tambah data kelahiran, agar data dapat tersimpan di *database*, *form* harus diisi dengan lengkap sesuai dengan data yang diminta. Pada bagian bawah pojok kiri terdapat tombol untuk menyimpan data kelahiran dan tombol untuk melakukan pembatalan tambah data kelahiran jika tidak jadi melakukan tambah data kelahiran. Setelah admin meng-klik tombol simpan maka sistem akan kembali ke halaman daftar data kelahiran dan admin dapat melihat bahwa penambahan data yang dilakukan sudah berhasil.

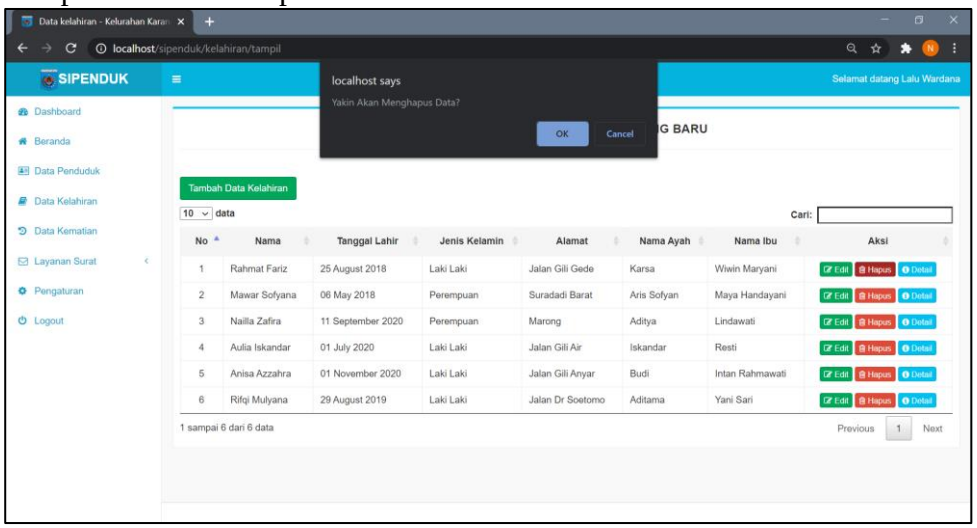
k. Tampilan halaman edit data kelahiran



Gambar 4. 67 Implementasi halaman edit data kelahiran

Gambar 4.67 merupakan tampilan *form* untuk admin melakukan edit data kelahiran, dengan *form* ini admin dapat memperbarui data kelahiran yang ada. Pada bagian bawah pojok kiri terdapat tombol untuk menyimpan perubahan data kelahiran dan tombol untuk melakukan pembatalan edit data kelahiran jika tidak jadi melakukan pembaruan data. Setelah admin meng-klik tombol simpan maka sistem akan kembali ke halaman daftar data kelahiran dan admin dapat melihat bahwa pembaruan data sudah berhasil tersimpan.

l. Tampilan halaman hapus data kelahiran

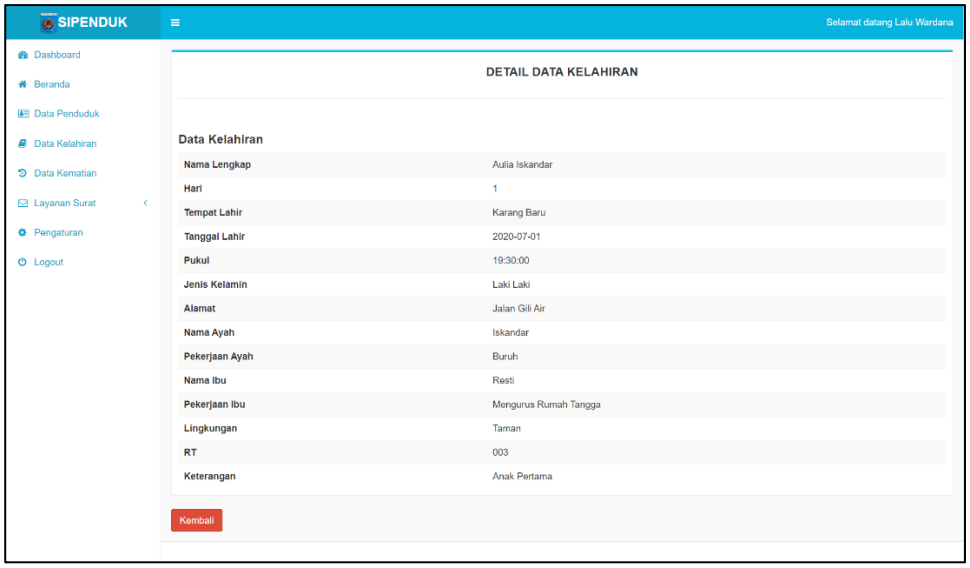


Gambar 4. 68 Implementasi halaman hapus data kelahiran

Gambar 4.68 merupakan tampilan halaman ketika admin akan melakukan hapus data kelahiran. Setelah admin mengklik tombol hapus

pada salah satu data kelahiran sistem akan memunculkan dialog konfirmasi untuk penghapusan data. Jika admin meng-klik *OK* maka data kelahiran yang dipilih akan otomatis terhapus dari daftar kelahiran. Jika admin tidak jadi melakukan penghapusan data kelahiran, admin dapat meng-klik pilihan *cancel* pada kotak dialog.

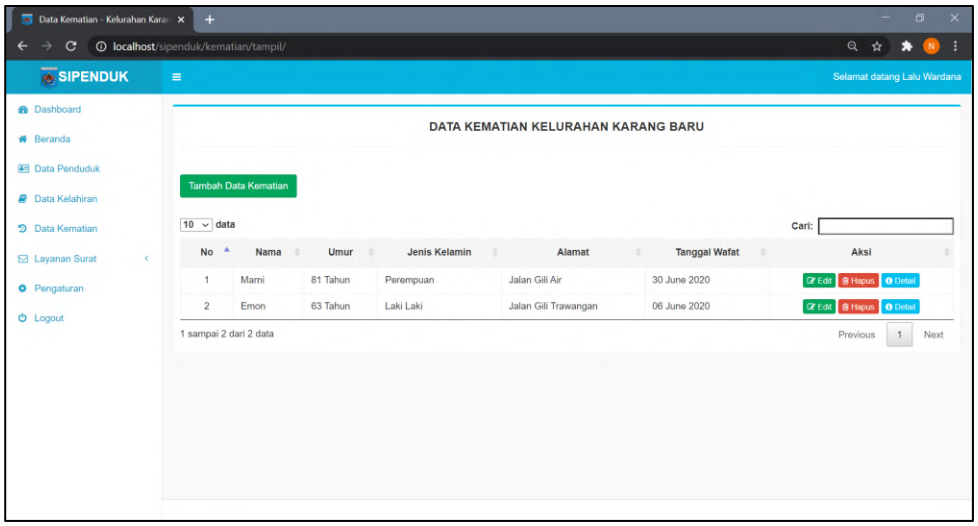
m. Tampilan halaman informasi detail data kelahiran



Gambar 4. 69 Implementasi halaman informasi detail data kelahiran

Gambar 4.69 merupakan tampilan halaman ketika admin memilih fitur untuk melihat detail data kelahiran. Ketika admin meng-klik tombol detail pada salah satu data kelahiran maka akan diarahkan ke halaman ini yang akan menampilkan rincian informasi dari data kelahiran yang dipilih. Pada bagian bawah terdapat tombol kembali yang ketika di-klik akan mengarahkan admin kembali ke halaman daftar data kelahiran.

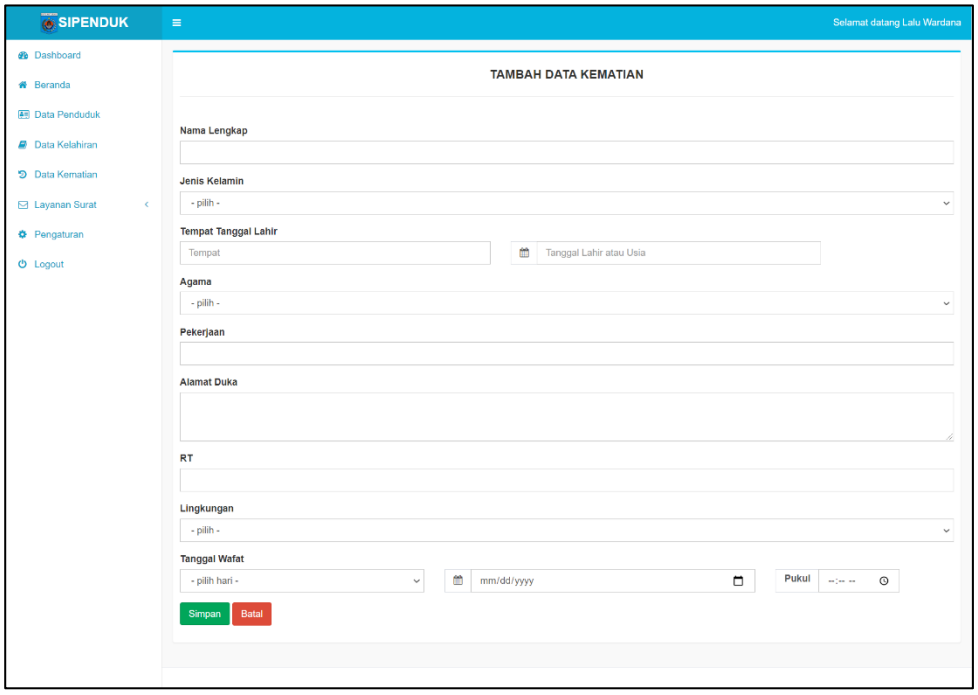
n. Tampilan halaman data kematian



Gambar 4. 70 Implementasi halaman data kematian

Gambar 4.70 merupakan implementasi dari halaman data kematian. Halaman ini berfungsi untuk menampilkan daftar data kematian Kelurahan Karang Baru. Pada halaman ini terdapat fitur tambah data kematian yang akan mengarahkan admin ke halaman tambah data kematian dan juga terdapat fitur cari data kematian untuk memudahkan admin dalam menemukan data kematian tertentu. Pada bagian aksi terdapat beberapa fungsi yaitu tombol untuk mengarahkan admin ke halaman edit data kematian, tombol untuk melakukan hapus data kematian dan tombol untuk mengarahkan admin ke halaman lihat rincian data kematian. Pada bagian bawah halaman juga terdapat tombol untuk menuju ke halaman selanjutnya dari daftar data kematian.

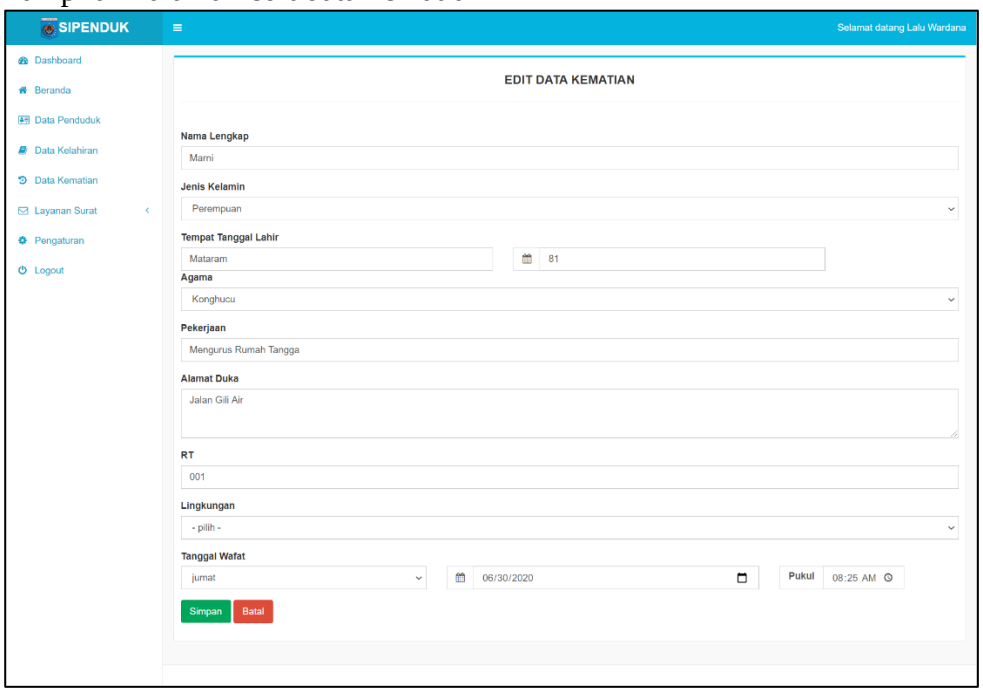
o. Tampilan halaman tambah data kematian



Gambar 4. 71 Implementasi halaman tambah data kematian

Gambar 4.71 merupakan tampilan *form* untuk admin melakukan tambah data kematian, agar data dapat tersimpan di *database*, *form* harus diisi dengan lengkap sesuai dengan data yang diminta. Pada bagian bawah pojok kiri terdapat tombol untuk menyimpan data kematian dan tombol untuk melakukan pembatalan tambah data kematian jika tidak jadi melakukan tambah data kematian. Setelah admin meng-klik tombol simpan maka sistem akan kembali ke halaman daftar data kematian dan admin dapat melihat bahwa penambahan data yang dilakukan sudah berhasil.

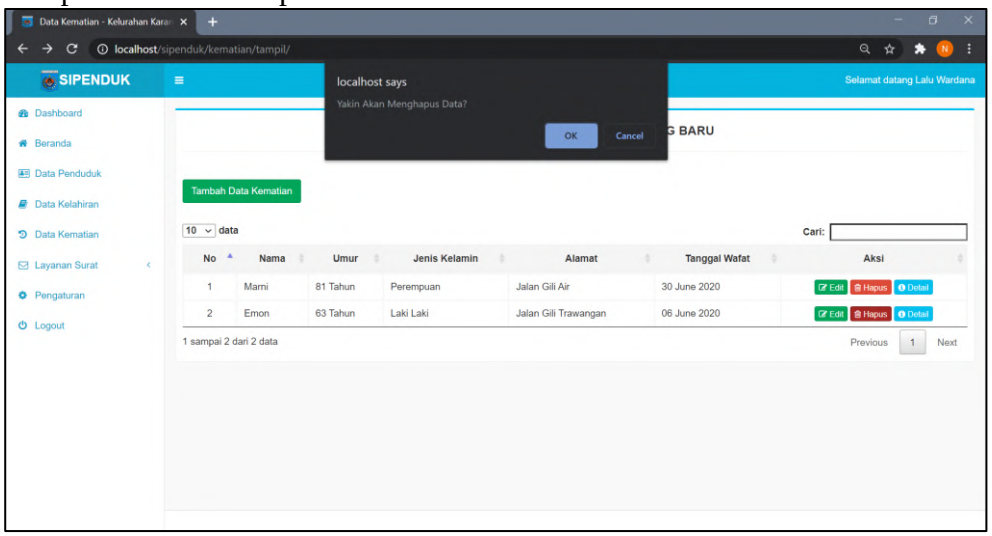
p. Tampilan halaman edit data kematian



Gambar 4. 72 Implementasi halaman edit data kematian

Gambar 4.72 merupakan tampilan *form* untuk admin melakukan edit data kematian, dengan *form* ini admin dapat memperbarui data kematian yang ada. Pada bagian bawah pojok kiri terdapat tombol untuk menyimpan perubahan data kematian dan tombol untuk melakukan pembatalan edit data kematian jika tidak jadi melakukan pembaruan data. Setelah admin meng-klik tombol simpan maka sistem akan kembali ke halaman daftar data kematian dan admin dapat melihat bahwa pembaruan data sudah berhasil tersimpan.

q. Tampilan halaman hapus data kematian

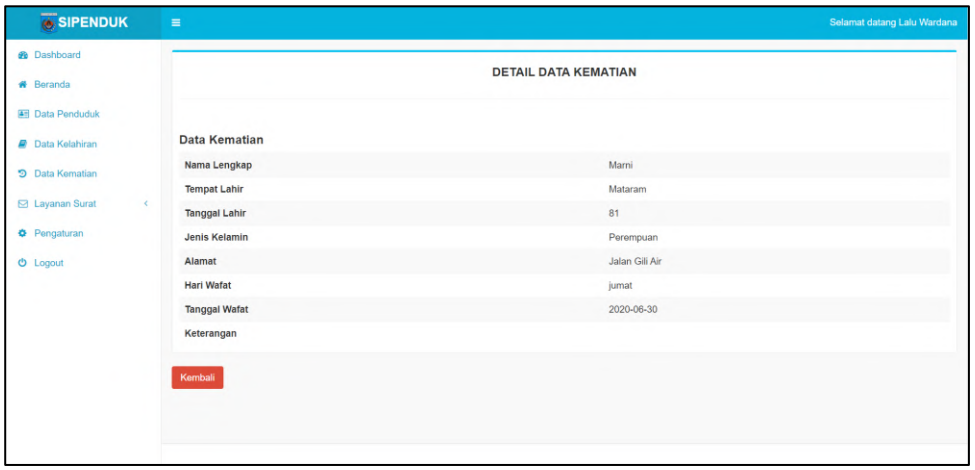


Gambar 4. 73 Implementasi halaman hapus data kematian

Gambar 4.73 merupakan tampilan halaman ketika admin akan

melakukan hapus data kematian. Setelah admin mengklik tombol hapus pada salah satu data kematian sistem akan memunculkan dialog konfirmasi untuk penghapusan data. Jika admin meng-klik *OK* maka data kematian yang dipilih akan otomatis terhapus dari daftar kematian. Jika admin tidak jadi melakukan penghapusan data kematian, admin dapat meng-klik pilihan *cancel* pada kotak dialog.

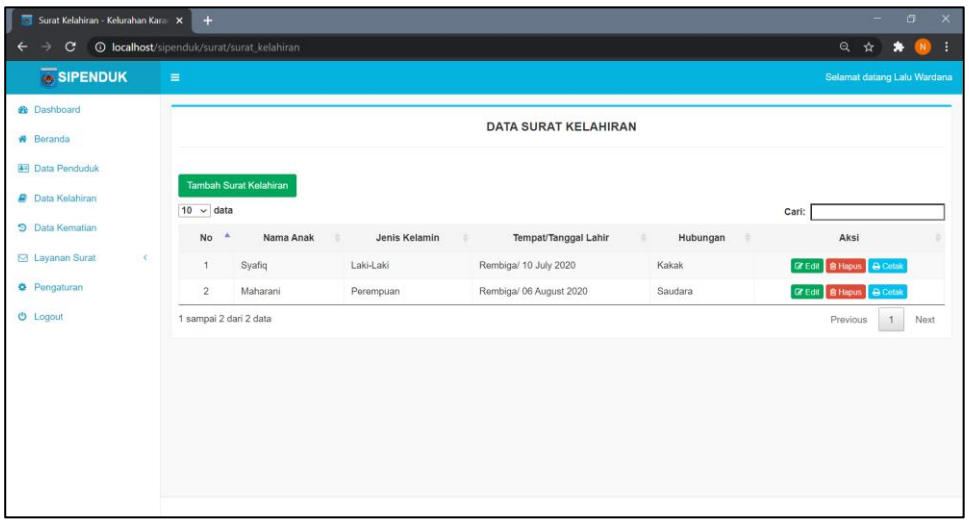
r. Tampilan halaman informasi detail data kematian



Gambar 4. 74 Implementasi halaman informasi detail data kematian

Gambar 4.74 merupakan tampilan halaman ketika admin memilih fitur untuk melihat detail data kematian. Ketika admin meng-klik tombol detail pada salah satu data kematian maka akan diarahkan ke halaman ini yang akan menampilkan rincian informasi dari data kematian yang dipilih. Pada bagian bawah terdapat tombol kembali yang ketika di-klik akan mengarahkan admin kembali ke halaman daftar data kematian.

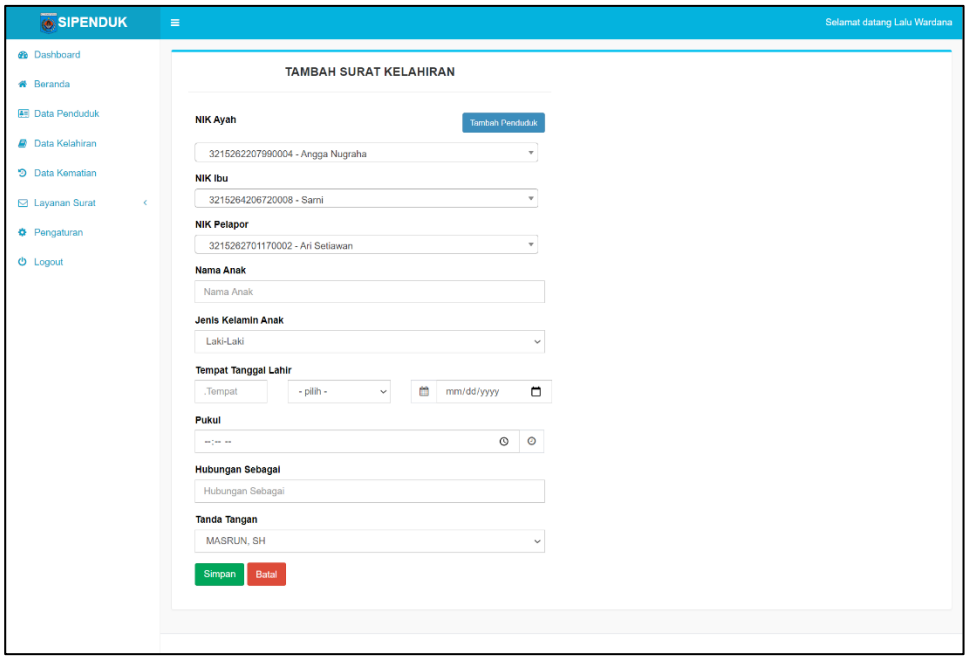
s. Tampilan halaman data layanan surat kelahiran



Gambar 4. 75 Implementasi halaman data layanan surat kelahiran

Gambar 4.75 merupakan implementasi dari halaman data surat kelahiran. Halaman ini berfungsi untuk menampilkan daftar surat kelahiran pada Kelurahan Karang Baru. Pada halaman ini terdapat fitur tambah surat kelahiran yang akan mengarahkan admin ke halaman tambah data surat kelahiran dan juga terdapat fitur cari surat kelahiran untuk memudahkan admin dalam menemukan surat kelahiran tertentu. Pada bagian aksi terdapat beberapa fungsi yaitu tombol untuk mengarahkan admin ke halaman edit data surat kelahiran, tombol untuk melakukan hapus data surat kelahiran dan tombol untuk mengarahkan admin ke halaman untuk mencetak data surat kelahiran. Pada bagian bawah halaman juga terdapat tombol untuk menuju ke halaman selanjutnya dari daftar surat kelahiran.

t. Tampilan halaman tambah data surat kelahiran

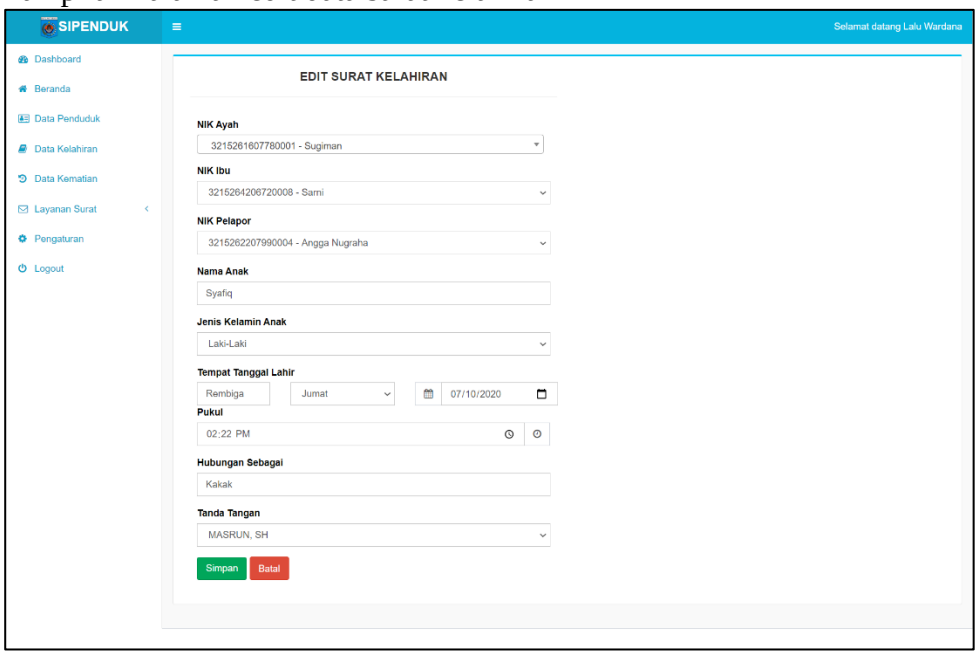


Gambar 4. 76 Implementasi halaman tambah data surat kelahiran

Gambar 4.76 merupakan tampilan *form* untuk admin melakukan tambah data surat kelahiran, agar data dapat tersimpan di *database*, *form* harus diisi dengan lengkap sesuai dengan ketentuan yang diminta. Misalnya ketika memasukkan NIK orang tua, maka harus dipastikan terlebih dahulu bahwa orang tua sudah terdaftar sebagai penduduk pada program, sehingga jika NIK orang tua belum terdaftar maka admin terlebih dahulu manambahkan data orang tua sebagai penduduk dengan tombol tambah data penduduk yang ada di bagian atas. Jika data sudah terisi dengan benar maka admin dapat memilih tombol simpan pada bagian bawah pojok kiri

menyimpan data surat kelahiran. Terdapat juga tombol untuk melakukan pembatalan tambah data surat kelahiran jika tidak jadi melakukan tambah data. Setelah admin meng-klik tombol simpan maka sistem akan kembali ke halaman daftar data surat kelahiran dan admin dapat melihat bahwa penambahan data yang dilakukan sudah berhasil.

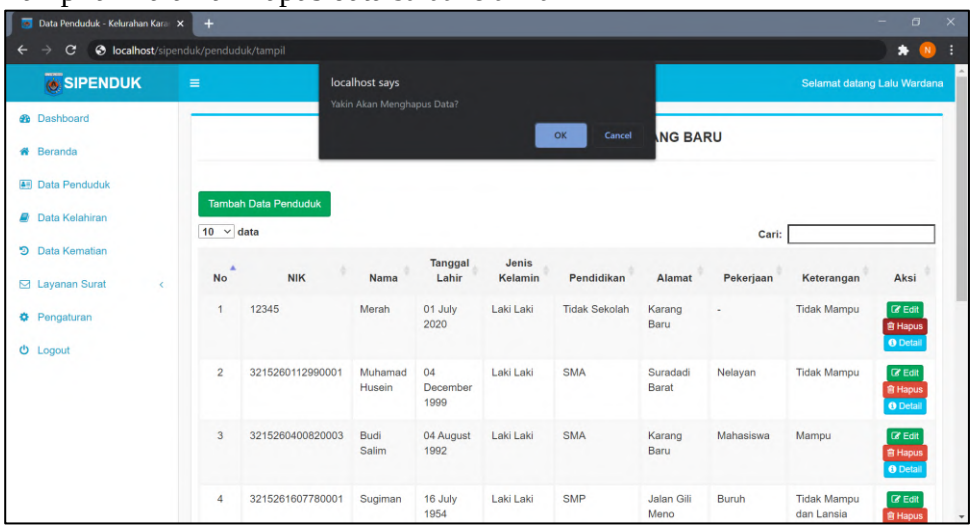
u. Tampilan halaman edit data surat kelahiran



Gambar 4. 77 Implementasi halaman edit data surat kelahiran

Gambar 4.77 merupakan tampilan *form* untuk admin melakukan edit data surat kelahiran, dengan *form* ini admin dapat memperbarui data surat kelahiran yang ada. Pada bagian bawah pojok kiri terdapat tombol untuk menyimpan perubahan data surat kelahiran dan tombol untuk melakukan pembatalan edit data surat kelahiran jika tidak jadi melakukan pembaruan data. Setelah admin meng-klik tombol simpan maka sistem akan kembali ke halaman daftar data surat kelahiran dan admin dapat melihat bahwa pembaruan data sudah berhasil tersimpan.

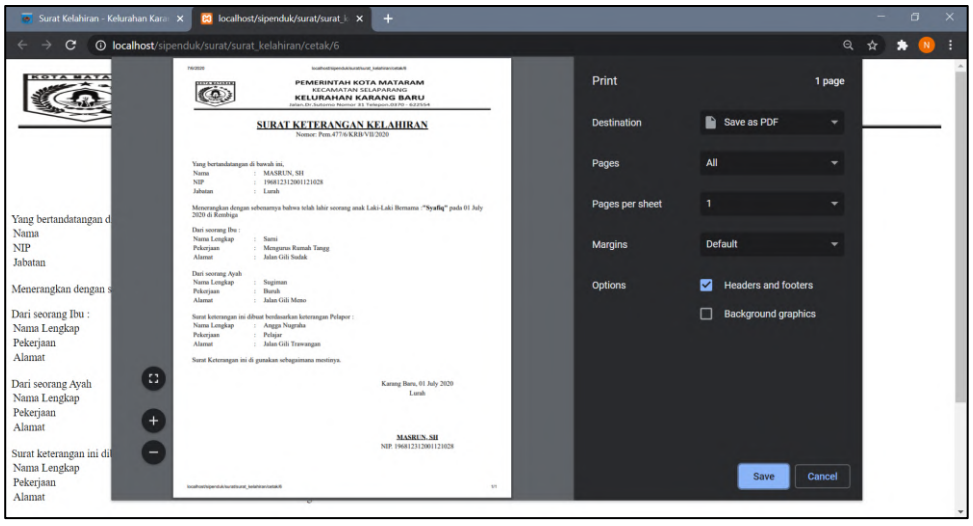
v. Tampilan halaman hapus data surat kelahiran



Gambar 4. 78 Implementasi halaman hapus data surat kelahiran

Gambar 4.78 merupakan tampilan halaman ketika admin akan melakukan hapus data surat kelahiran. Setelah admin mengklik tombol hapus pada salah satu data surat kelahiran maka sistem akan memunculkan dialog konfirmasi untuk penghapusan data. Jika admin meng-klik *OK* maka data surat kelahiran yang dipilih akan otomatis terhapus dari daftar surat kelahiran. Jika admin tidak jadi melakukan penghapusan data surat kelahiran, admin dapat meng-klik pilihan *cancel* pada kotak dialog.

w. Tampilan halaman cetak data surat kelahiran

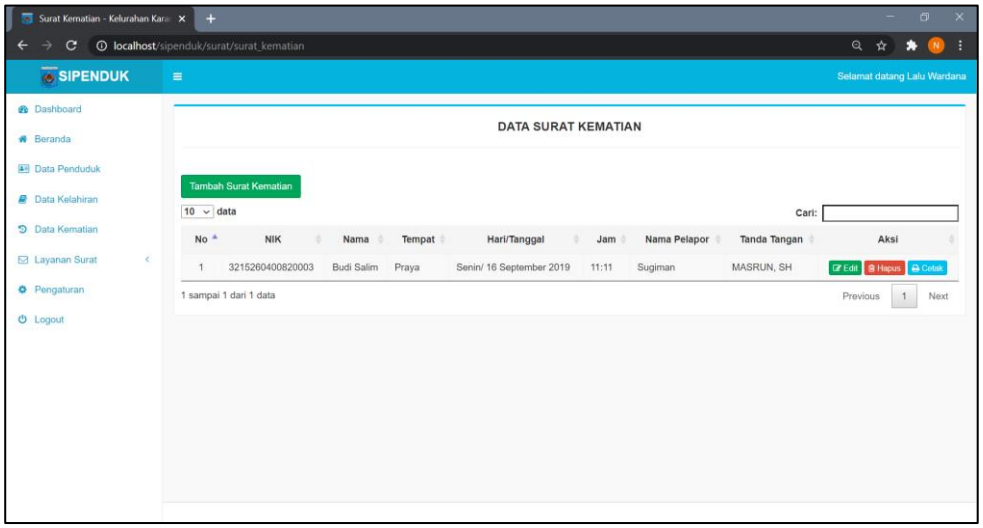


Gambar 4. 79 Implementasi halaman cetak data surat kelahiran

Gambar 4.79 merupakan tampilan halaman ketika admin memilih fitur untuk melakukan cetak data surat kelahiran. Ketika admin meng-klik tombol cetak pada salah satu data surat kelahiran maka akan diarahkan ke halaman baru yang akan menampilkan rincian pengaturan untuk mencetak surat

kelahiran. Hasil pencetakan surat kelahiran dapat diatur sesuai kebutuhan, apakah ingin menyimpan surat sebagai berkas PDF atau mencetak surat ke dalam bentuk fisik. Pada bagian bawah terdapat tombol *cancel* yang ketika di-klik akan membatalkan pencetakan surat.

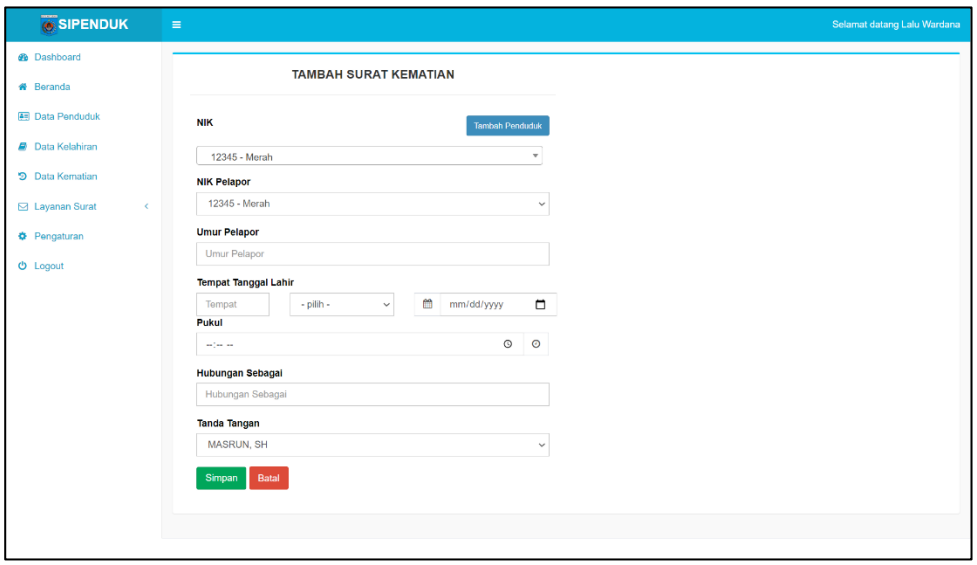
x. Tampilan halaman data layanan surat kematian



Gambar 4. 80 Implementasi halaman data layanan surat kematian

Gambar 4.80 merupakan implementasi dari halaman data surat kematian. Halaman ini berfungsi untuk menampilkan daftar surat kematian pada Kelurahan Karang Baru. Pada halaman ini terdapat fitur tambah surat kematian yang akan mengarahkan admin ke halaman tambah data surat kematian dan juga terdapat fitur cari surat kematian untuk memudahkan admin dalam menemukan surat kematian tertentu. Pada bagian aksi terdapat beberapa fungsi yaitu tombol untuk mengarahkan admin ke halaman edit data surat kematian, tombol untuk melakukan hapus data surat kematian dan tombol untuk mengarahkan admin ke halaman untuk mencetak data surat kematian. Pada bagian bawah halaman juga terdapat tombol untuk menuju ke halaman selanjutnya dari daftar surat kematian.

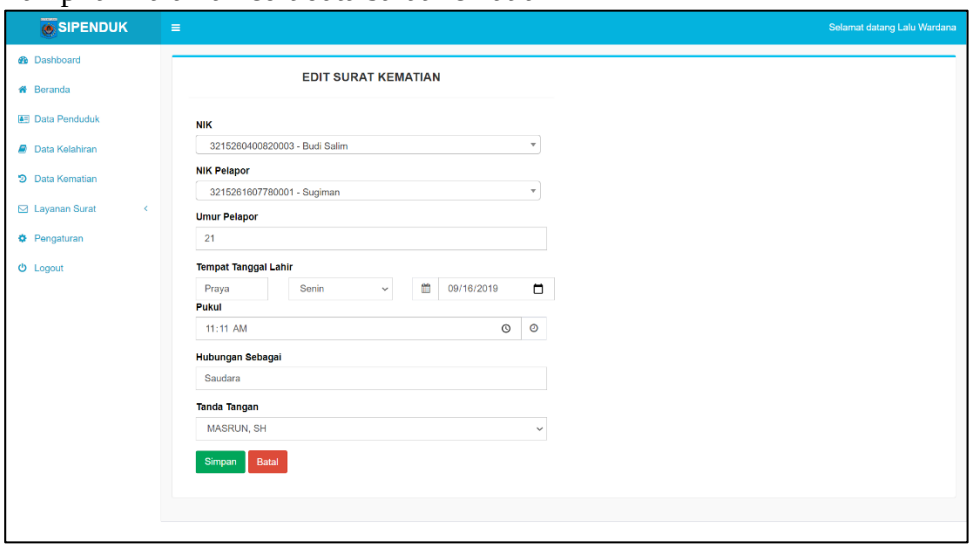
y. Tampilan halaman tambah data surat kematian



Gambar 4. 81 Implementasi halaman tambah data surat kematian

Gambar 4.81 merupakan tampilan *form* untuk admin melakukan tambah data surat kematian, agar data dapat tersimpan di *database*, *form* harus diisi dengan lengkap sesuai dengan ketentuan yang diminta. Misalnya ketika memasukkan NIK penduduk yang bersangkutan, maka harus dipastikan terlebih dahulu bahwa yang bersangkutan sudah terdaftar sebagai penduduk pada program, sehingga jika NIK yang bersangkutan belum terdaftar maka admin terlebih dahulu manambahkan data yang bersangkutan sebagai penduduk dengan tombol tambah data penduduk yang ada di bagian atas. Jika data sudah terisi dengan benar maka admin dapat memilih tombol simpan pada bagian bawah pojok kiri menyimpan data surat kematian. Terdapat juga tombol untuk melakukan pembatalan tambah data surat kematian jika tidak jadi melakukan tambah data. Setelah admin meng-klik tombol simpan maka sistem akan kembali ke halaman daftar data surat kematian dan admin dapat melihat bahwa penambahan data yang dilakukan sudah berhasil.

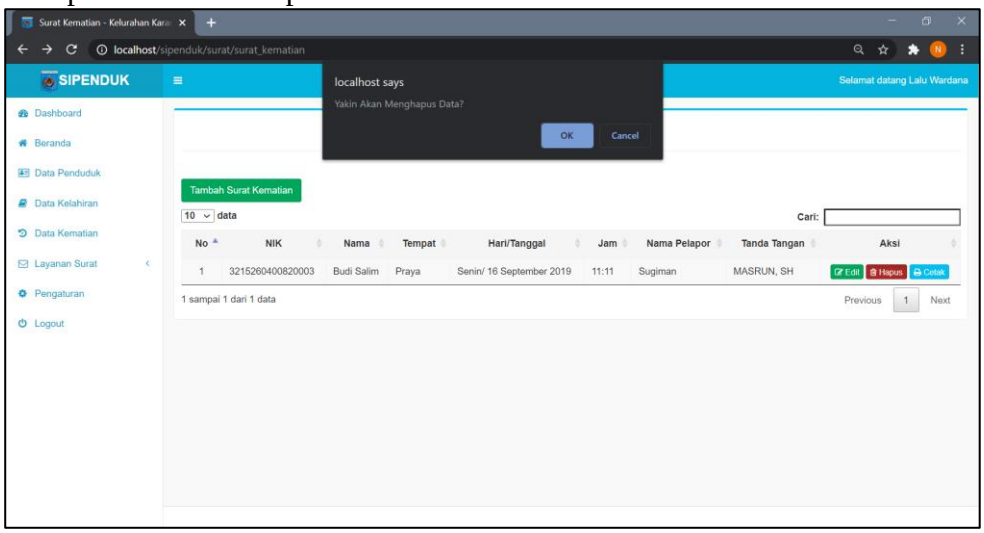
z. Tampilan halaman edit data surat kematian



Gambar 4. 82 Implementasi halaman edit data surat kematian

Gambar 4.82 merupakan tampilan *form* untuk admin melakukan edit data surat kematian, dengan *form* ini admin dapat memperbarui data surat kematian yang ada. Pada bagian bawah pojok kiri terdapat tombol untuk menyimpan perubahan data surat kematian dan tombol untuk melakukan pembatalan edit data surat kematian jika tidak jadi melakukan pembaruan data. Setelah admin meng-klik tombol simpan maka sistem akan kembali ke halaman daftar data surat kematian dan admin dapat melihat bahwa pembaruan data sudah berhasil tersimpan.

- Tampilan halaman hapus data surat kematian

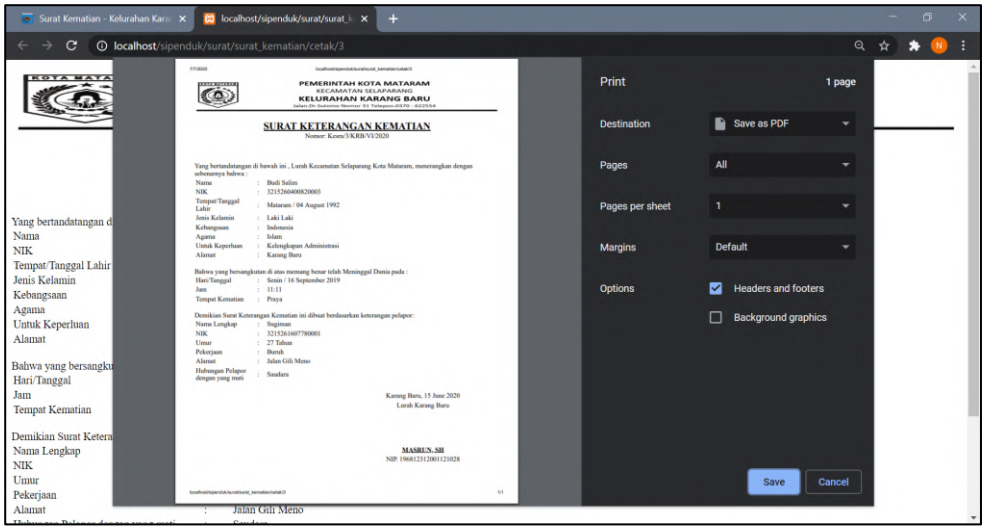


Gambar 4. 83 Implementasi halaman hapus data surat kematian

Gambar 4.83 merupakan tampilan halaman ketika admin akan melakukan hapus data surat kematian. Setelah admin mengklik tombol hapus pada salah satu data surat kematian maka sistem akan memunculkan

dialog konfirmasi untuk penghapusan data. Jika admin meng-klik *OK* maka data surat kematian yang dipilih akan otomatis terhapus dari daftar surat kematian. Jika admin tidak jadi melakukan penghapusan data surat kematian, admin dapat meng-klik pilihan *cancel* pada kotak dialog.

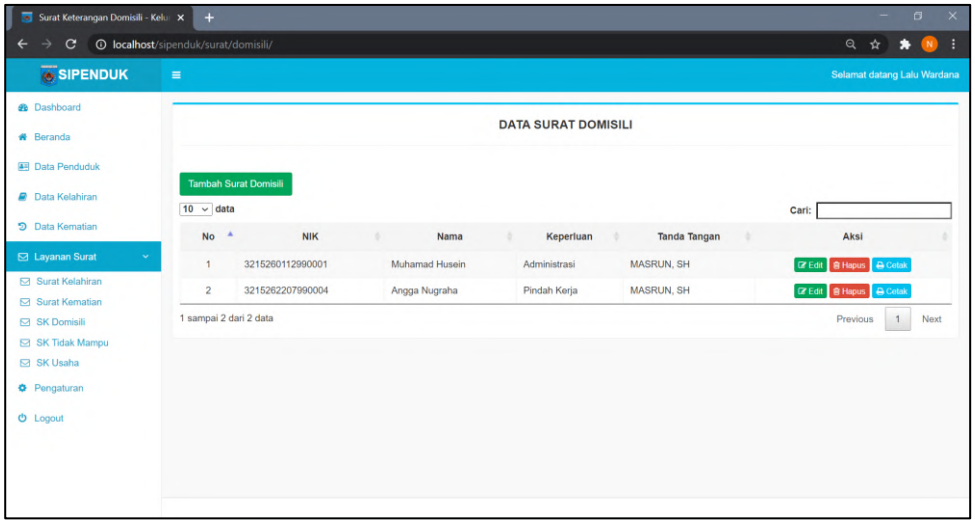
- Tampilan halaman cetak data surat kematian



Gambar 4. 84 Implementasi halaman cetak data surat kematian

Gambar 4.84 merupakan tampilan halaman ketika admin memilih fitur untuk melakukan cetak data surat kematian. Ketika admin meng-klik tombol cetak pada salah satu data surat kematian maka akan diarahkan ke halaman baru yang akan menampilkan rincian pengaturan untuk mencetak surat kematian. Hasil pencetakan surat kematian dapat diatur sesuai kebutuhan, apakah ingin menyimpan surat sebagai berkas PDF atau mencetak surat ke dalam bentuk fisik. Pada bagian bawah terdapat tombol *cancel* yang ketika di-klik akan membatalkan pencetakan surat.

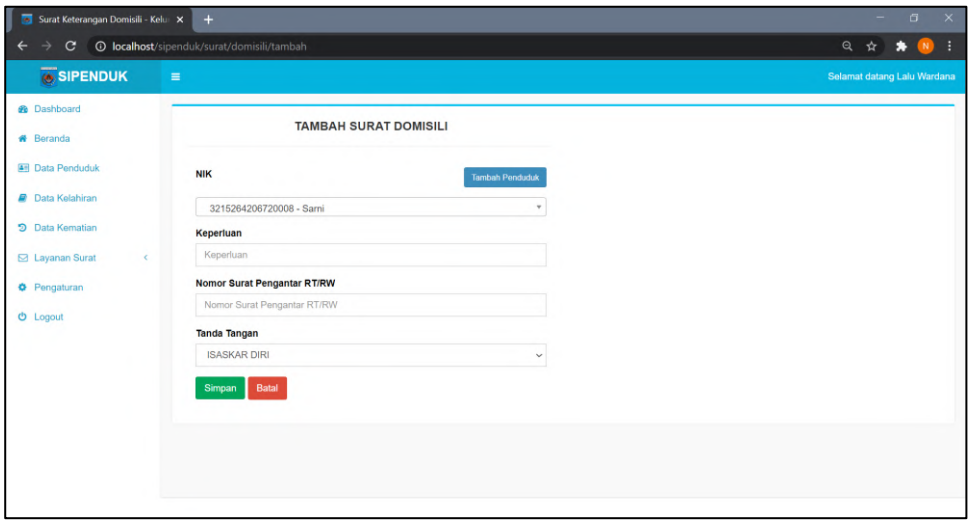
- Tampilan halaman data layanan surat keterangan domisili



Gambar 4. 85 Implementasi halaman data layanan surat keterangan domisili

Gambar 4.85 merupakan implementasi dari halaman data surat keterangan domisili. Halaman ini berfungsi untuk menampilkan daftar surat keterangan domisili pada Kelurahan Karang Baru. Pada halaman ini terdapat fitur tambah surat keterangan domisili yang akan mengarahkan admin ke halaman tambah data surat keterangan domisili dan juga terdapat fitur cari surat keterangan domisili untuk memudahkan admin dalam menemukan surat keterangan domisili tertentu. Pada bagian aksi terdapat beberapa fungsi yaitu tombol untuk mengarahkan admin ke halaman edit data surat keterangan domisili, tombol untuk melakukan hapus data surat keterangan domisili dan tombol untuk mengarahkan admin ke halaman untuk mencetak data surat keterangan domisili. Pada bagian bawah halaman juga terdapat tombol untuk menuju ke halaman selanjutnya dari daftar surat keterangan domisili.

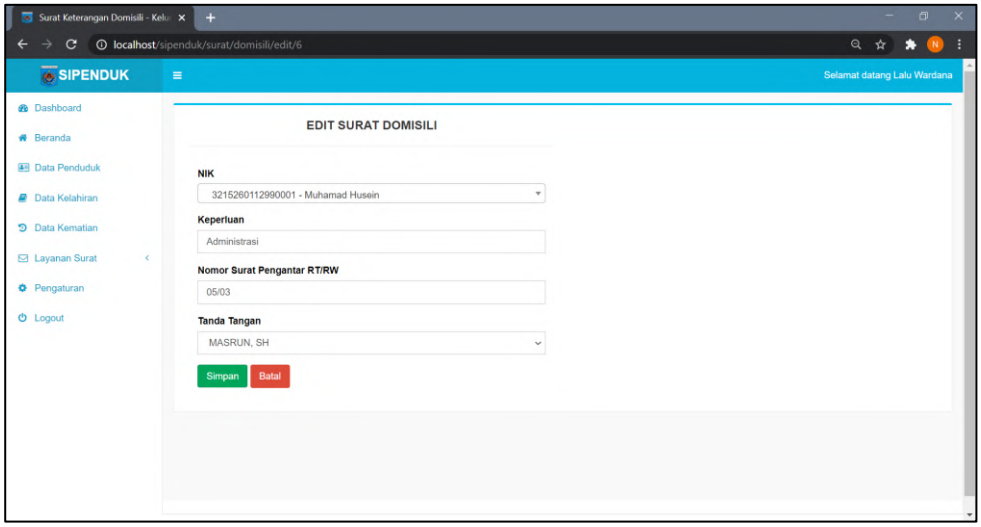
- Tampilan halaman tambah data surat keterangan domisili



Gambar 4. 86 Implementasi halaman tambah data surat keterangan domisili

Gambar 4.86 merupakan tampilan *form* untuk admin melakukan tambah data surat keterangan domisili, agar data dapat tersimpan di *database*, *form* harus diisi dengan lengkap sesuai dengan ketentuan yang diminta. Misalnya ketika memasukkan NIK penduduk yang bersangkutan, maka harus dipastikan terlebih dahulu bahwa yang bersangkutan sudah terdaftar sebagai penduduk pada program, sehingga jika NIK yang bersangkutan belum terdaftar maka admin terlebih dahulu menambahkan data yang bersangkutan sebagai penduduk dengan tombol tambah data penduduk yang ada di bagian atas. Jika data sudah terisi dengan benar maka admin dapat memilih tombol simpan pada bagian bawah pojok kiri menyimpan data surat keterangan domisili. Terdapat juga tombol untuk melakukan pembatalan tambah data surat keterangan domisili jika tidak jadi melakukan tambah data. Setelah admin meng-klik tombol simpan maka sistem akan kembali ke halaman daftar data surat keterangan domisili dan admin dapat melihat bahwa penambahan data yang dilakukan sudah berhasil.

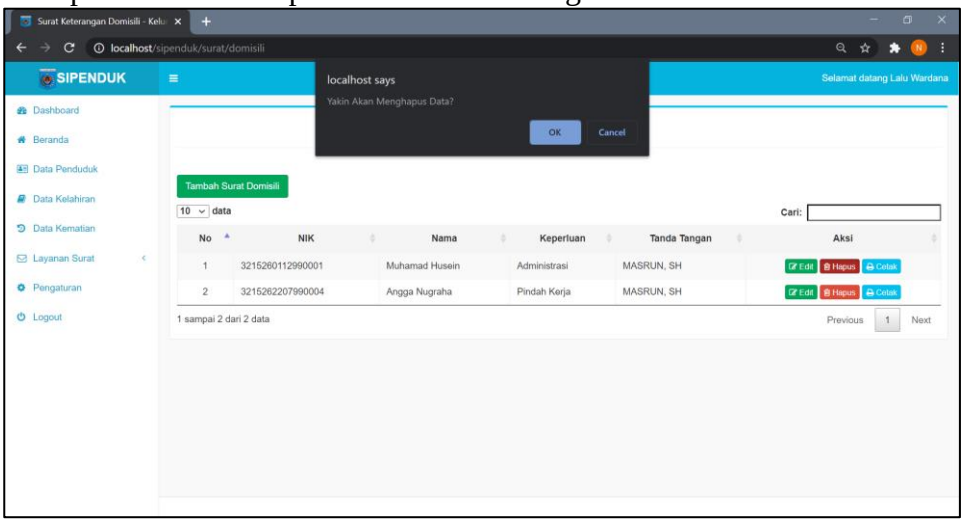
- Tampilan halaman edit data surat keterangan domisili



Gambar 4. 87 Implementasi halaman edit data surat keterangan domisili

Gambar 4.87 merupakan tampilan *form* untuk admin melakukan edit data surat keterangan domisili, dengan *form* ini admin dapat memperbarui data surat keterangan domisili yang ada. Pada bagian bawah pojok kiri terdapat tombol untuk menyimpan perubahan data surat keterangan domisili dan tombol untuk melakukan pembatalan edit data surat keterangan domisili jika tidak jadi melakukan pembaruan data. Setelah admin meng-klik tombol simpan maka sistem akan kembali ke halaman daftar data surat keterangan domisili dan admin dapat melihat bahwa pembaruan data sudah berhasil tersimpan.

- Tampilan halaman hapus data surat keterangan domisili

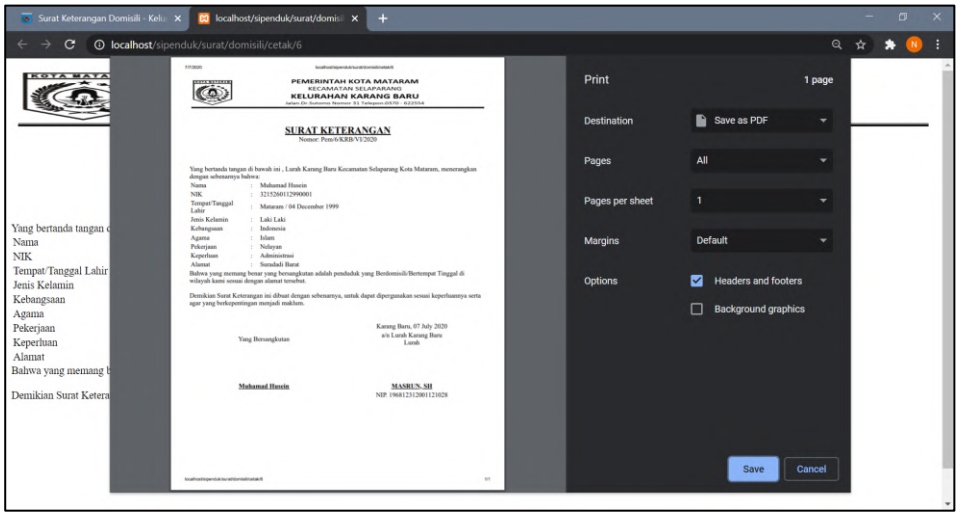


Gambar 4. 88 Implementasi halaman hapus data surat keterangan domisili

Gambar 4.88 merupakan tampilan halaman ketika admin akan melakukan hapus data surat keterangan domisili. Setelah admin mengklik

tombol hapus pada salah satu data surat keterangan domisili maka sistem akan memunculkan dialog konfirmasi untuk penghapusan data. Jika admin meng-klik *OK* maka data surat keterangan domisili yang dipilih akan otomatis terhapus dari daftar surat keterangan domisili. Jika admin tidak jadi melakukan penghapusan data surat keterangan domisili, admin dapat meng-klik pilihan *cancel* pada kotak dialog.

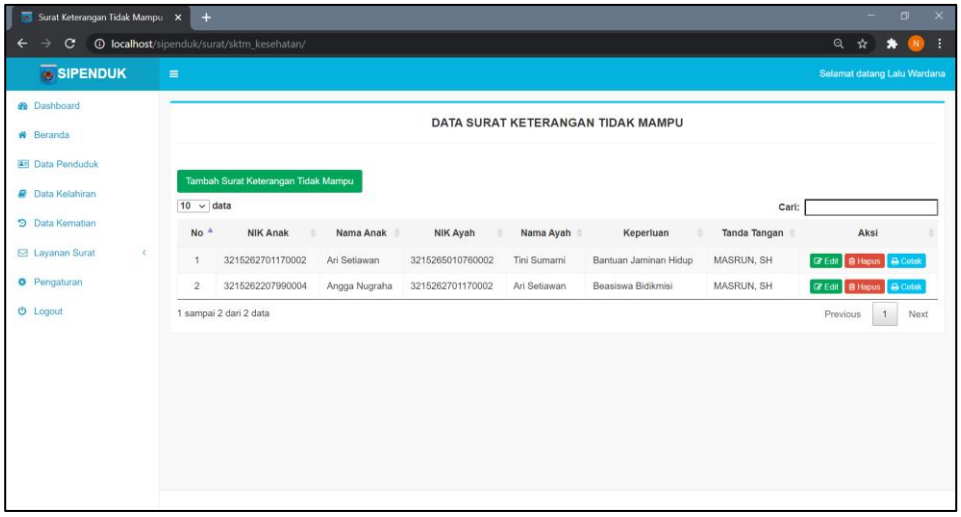
- Tampilan halaman cetak data surat keterangan domisili



Gambar 4. 89 Implementasi halaman cetak data surat keterangan domisili

Gambar 4.89 merupakan tampilan halaman ketika admin memilih fitur untuk melakukan cetak data surat keterangan domisili. Ketika admin meng-klik tombol cetak pada salah satu data surat keterangan domisili maka akan diarahkan ke halaman baru yang akan menampilkan rincian pengaturan untuk mencetak surat keterangan domisili. Hasil pencetakan surat keterangan domisili dapat diatur sesuai kebutuhan, apakah ingin menyimpan surat sebagai berkas PDF atau mencetak surat ke dalam bentuk fisik. Pada bagian bawah terdapat tombol *cancel* yang ketika di-klik akan membatalkan pencetakan surat.

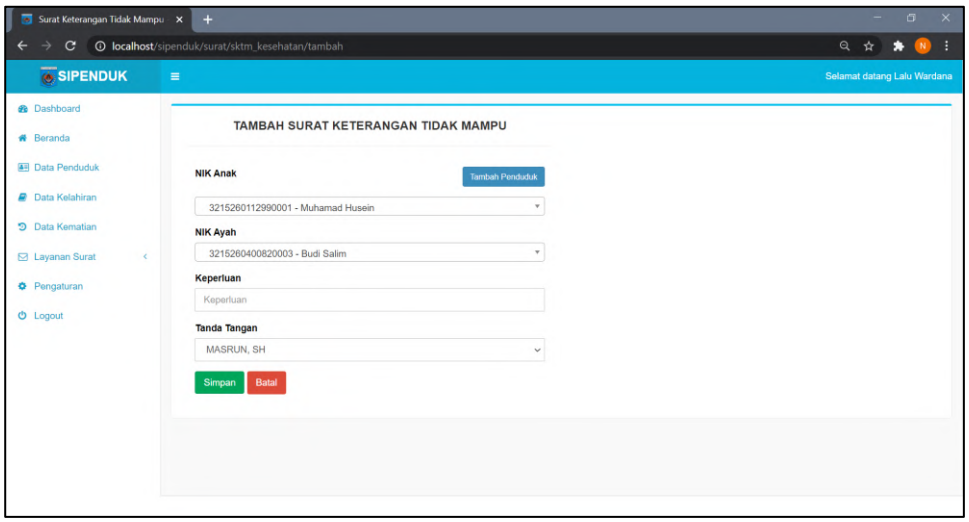
- Tampilan halaman data layanan surat keterangan tidak mampu



Gambar 4. 90 Implementasi halaman data layanan surat keterangan tidak mampu

Gambar 4.90 merupakan implementasi dari halaman data surat keterangan tidak mampu. Halaman ini berfungsi untuk menampilkan daftar surat keterangan tidak mampu pada Kelurahan Karang Baru. Pada halaman ini terdapat fitur tambah surat keterangan tidak mampu yang akan mengarahkan admin ke halaman tambah data surat keterangan tidak mampu dan juga terdapat fitur cari surat keterangan tidak mampu untuk memudahkan admin dalam menemukan surat keterangan tidak mampu tertentu. Pada bagian aksi terdapat beberapa fungsi yaitu tombol untuk mengarahkan admin ke halaman edit data surat keterangan tidak mampu, tombol untuk melakukan hapus data surat keterangan tidak mampu dan tombol untuk mengarahkan admin ke halaman untuk mencetak data surat keterangan tidak mampu. Pada bagian bawah halaman juga terdapat tombol untuk menuju ke halaman selanjutnya dari daftar surat keterangan tidak mampu.

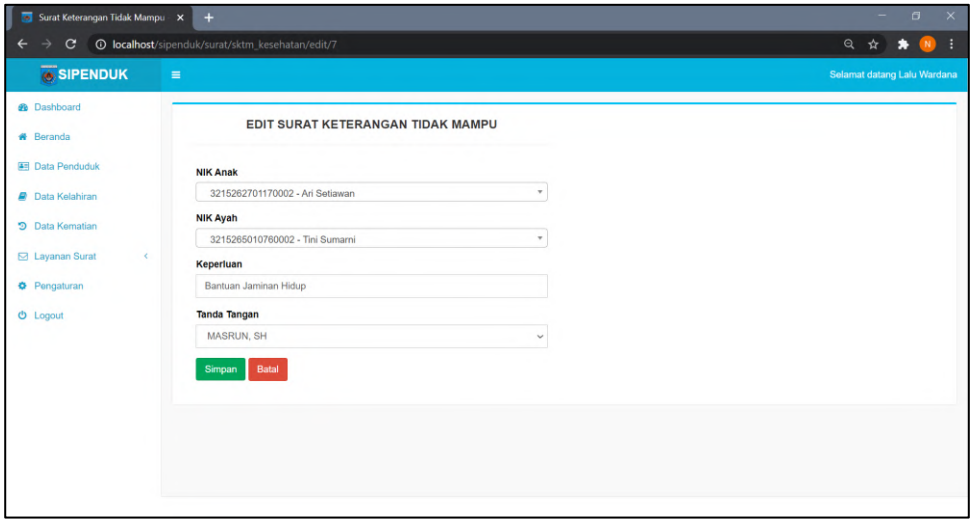
- Tampilan halaman tambah data surat keterangan tidak mampu



Gambar 4. 91 Implementasi halaman tambah data surat keterangan tidak mampu

Gambar 4.91 merupakan tampilan *form* untuk admin melakukan tambah data surat keterangan tidak mampu, agar data dapat tersimpan di *database*, *form* harus diisi dengan lengkap sesuai dengan ketentuan yang diminta. Misalnya ketika memasukkan NIK penduduk yang bersangkutan, maka harus dipastikan terlebih dahulu bahwa yang bersangkutan sudah terdaftar sebagai penduduk pada program, sehingga jika NIK yang bersangkutan belum terdaftar maka admin terlebih dahulu menambahkan data yang bersangkutan sebagai penduduk dengan tombol tambah data penduduk yang ada di bagian atas. Jika data sudah terisi dengan benar maka admin dapat memilih tombol simpan pada bagian bawah pojok kiri menyimpan data surat keterangan tidak mampu. Terdapat juga tombol untuk melakukan pembatalan tambah data surat keterangan tidak mampu jika tidak jadi melakukan tambah data. Setelah admin meng-klik tombol simpan maka sistem akan kembali ke halaman daftar data surat keterangan tidak mampu dan admin dapat melihat bahwa penambahan data yang dilakukan sudah berhasil.

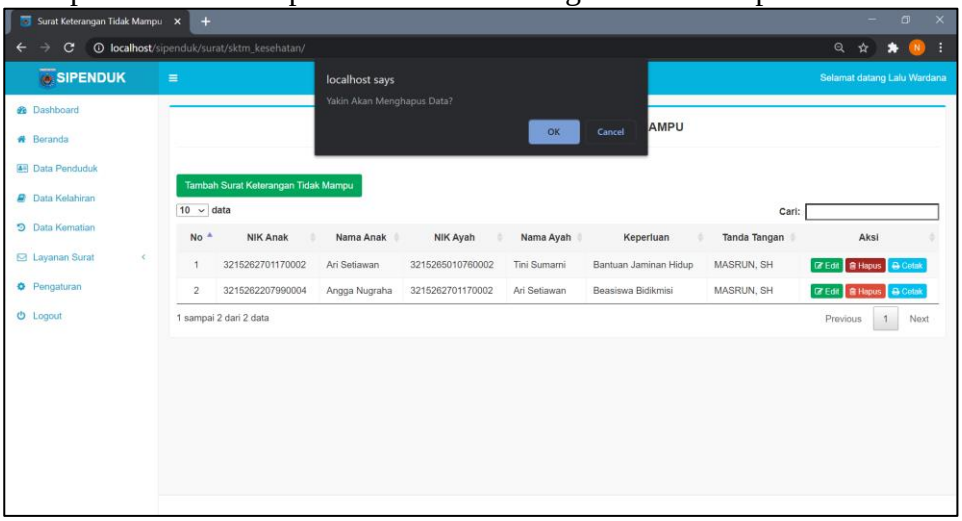
- Tampilan halaman edit data surat keterangan tidak mampu



Gambar 4. 92 Implementasi halaman edit data surat keterangan tidak mampu

Gambar 4.92 merupakan tampilan *form* untuk admin melakukan edit data surat keterangan tidak mampu, dengan *form* ini admin dapat memperbarui data surat keterangan tidak mampu yang ada. Pada bagian bawah pojok kiri terdapat tombol untuk menyimpan perubahan data surat keterangan tidak mampu dan tombol untuk melakukan pembatalan edit data surat keterangan tidak mampu jika tidak jadi melakukan pembaruan data. Setelah admin meng-klik tombol simpan maka sistem akan kembali ke halaman daftar data surat keterangan tidak mampu dan admin dapat melihat bahwa pembaruan data sudah berhasil tersimpan.

- Tampilan halaman hapus data surat keterangan tidak mampu

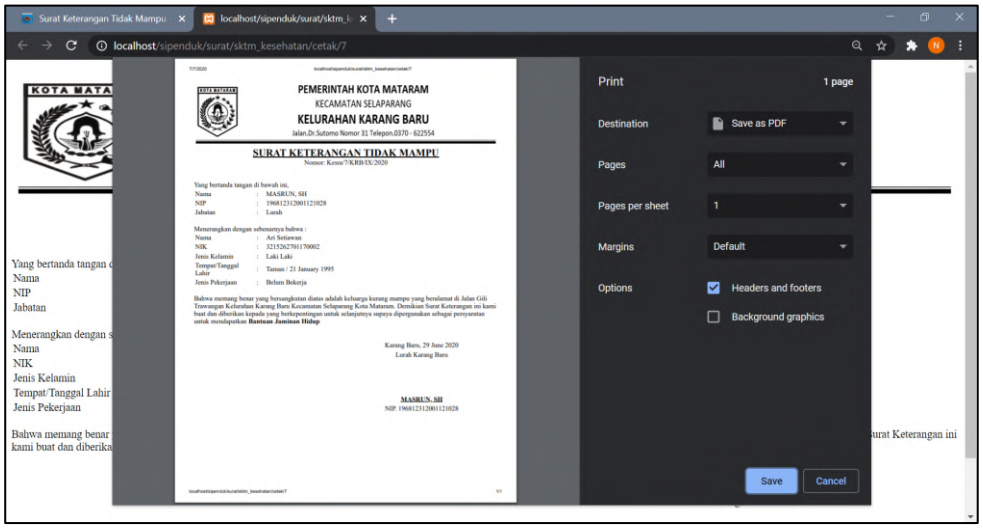


Gambar 4. 93 Implementasi halaman hapus data surat keterangan tidak mampu

Gambar 4.93 merupakan tampilan halaman ketika admin akan melakukan hapus data surat keterangan tidak mampu. Setelah admin

mengklik tombol hapus pada salah satu data surat keterangan tidak mampu maka sistem akan memunculkan dialog konfirmasi untuk penghapusan data. Jika admin meng-klik *OK* maka data surat keterangan tidak mampu yang dipilih akan otomatis terhapus dari daftar surat keterangan tidak mampu. Jika admin tidak jadi melakukan penghapusan data surat keterangan tidak mampu, admin dapat meng-klik pilihan *cancel* pada kotak dialog.

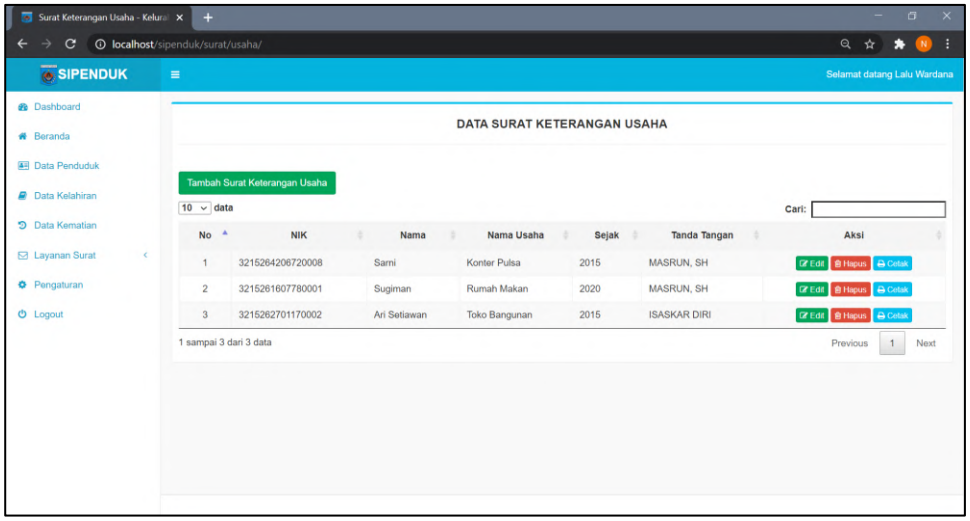
- Tampilan halaman cetak data surat keterangan tidak mampu



Gambar 4. 94 Implementasi halaman cetak data surat keterangan tidak mampu

Gambar 4.94 merupakan tampilan halaman ketika admin memilih fitur untuk melakukan cetak data surat keterangan tidak mampu. Ketika admin meng-klik tombol cetak pada salah satu data surat keterangan tidak mampu maka akan diarahkan ke halaman baru yang akan menampilkan rincian pengaturan untuk mencetak surat keterangan tidak mampu. Hasil pencetakan surat keterangan tidak mampu dapat diatur sesuai kebutuhan, apakah ingin menyimpan surat sebagai berkas PDF atau mencetak surat ke dalam bentuk fisik. Pada bagian bawah terdapat tombol *cancel* yang ketika di-klik akan membatalkan pencetakan surat.

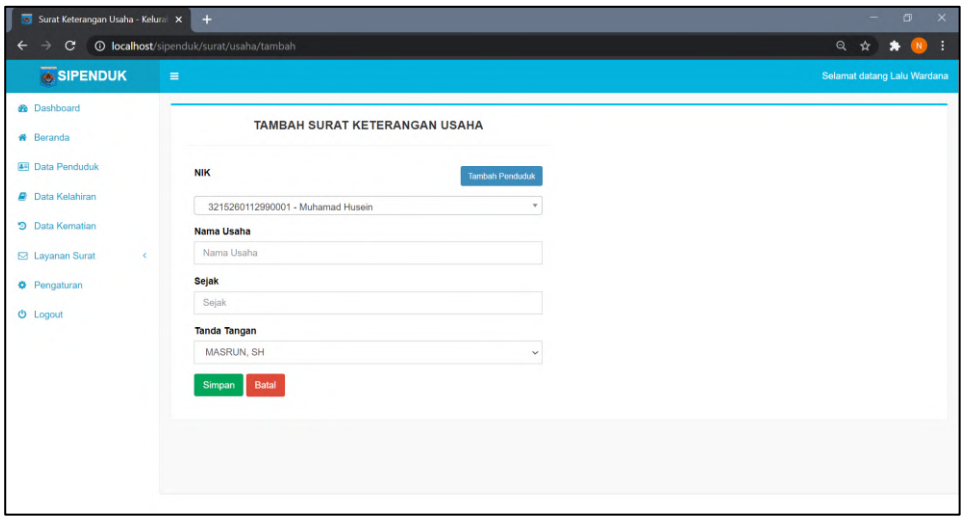
- Tampilan halaman data layanan surat keterangan usaha



Gambar 4. 95 Implementasi halaman data layanan surat keterangan usaha

Gambar 4.95 merupakan implementasi dari halaman data surat keterangan usaha. Halaman ini berfungsi untuk menampilkan daftar surat keterangan usaha pada Kelurahan Karang Baru. Pada halaman ini terdapat fitur tambah surat keterangan usaha yang akan mengarahkan admin ke halaman tambah data surat keterangan usaha dan juga terdapat fitur cari surat keterangan usaha untuk memudahkan admin dalam menemukan surat keterangan usaha tertentu. Pada bagian aksi terdapat beberapa fungsi yaitu tombol untuk mengarahkan admin ke halaman edit data surat keterangan usaha, tombol untuk melakukan hapus data surat keterangan usaha dan tombol untuk mengarahkan admin ke halaman untuk mencetak data surat keterangan usaha. Pada bagian bawah halaman juga terdapat tombol untuk menuju ke halaman selanjutnya dari daftar surat keterangan usaha.

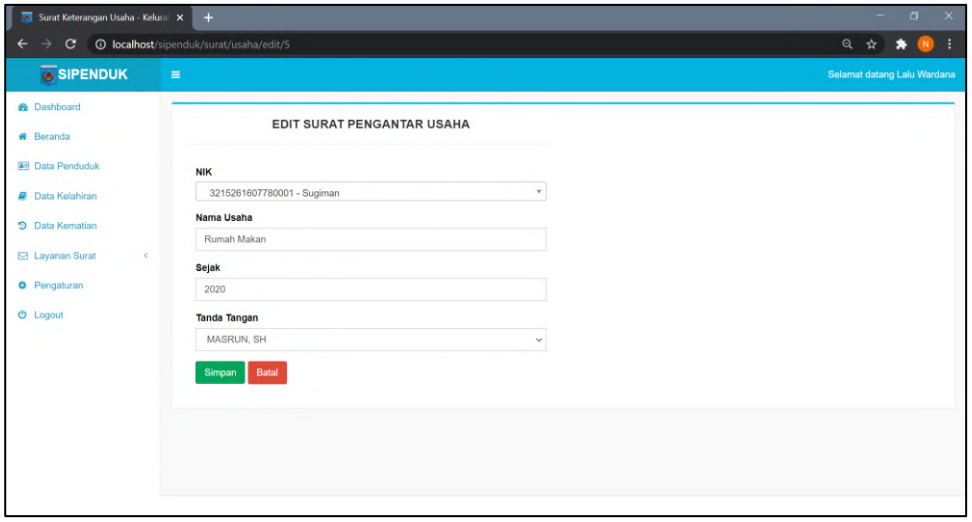
- Tampilan halaman tambah data surat keterangan usaha



Gambar 4. 96 Implementasi halaman tambah data surat keterangan usaha

Gambar 4.96 merupakan tampilan *form* untuk admin melakukan tambah data surat keterangan usaha, agar data dapat tersimpan di *database*, *form* harus diisi dengan lengkap sesuai dengan ketentuan yang diminta. Misalnya ketika memasukkan NIK penduduk yang bersangkutan, maka harus dipastikan terlebih dahulu bahwa yang bersangkutan sudah terdaftar sebagai penduduk pada program, sehingga jika NIK yang bersangkutan belum terdaftar maka admin terlebih dahulu menambahkan data yang bersangkutan sebagai penduduk dengan tombol tambah data penduduk yang ada di bagian atas. Jika data sudah terisi dengan benar maka admin dapat memilih tombol simpan pada bagian bawah pojok kiri menyimpan data surat keterangan usaha. Terdapat juga tombol untuk melakukan pembatalan tambah data surat keterangan usaha jika tidak jadi melakukan tambah data. Setelah admin meng-klik tombol simpan maka sistem akan kembali ke halaman daftar data surat keterangan usaha dan admin dapat melihat bahwa penambahan data yang dilakukan sudah berhasil.

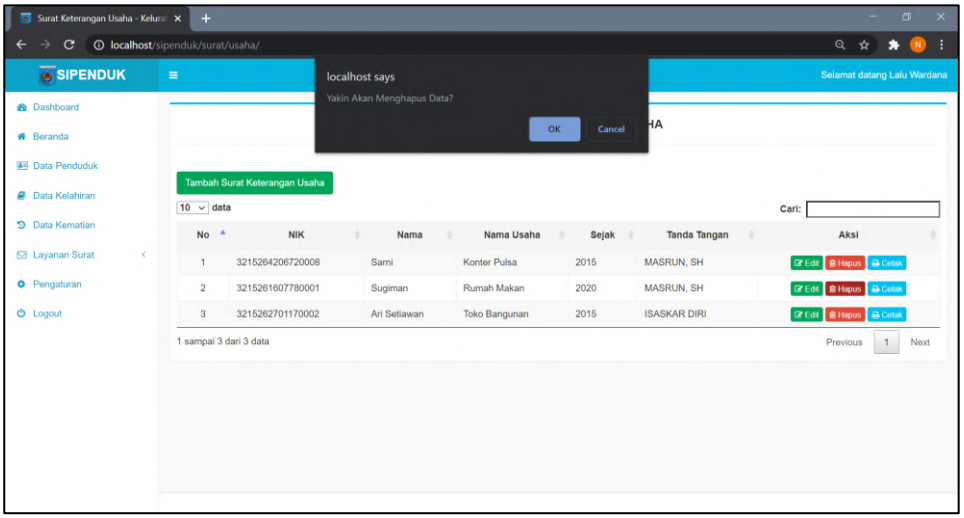
- Tampilan halaman edit data surat keterangan usaha



Gambar 4. 97 Implementasi halaman edit data surat keterangan usaha

Gambar 4.97 merupakan tampilan *form* untuk admin melakukan edit data surat usaha, dengan *form* ini admin dapat memperbarui data surat usaha yang ada. Pada bagian bawah pojok kiri terdapat tombol untuk menyimpan perubahan data surat usaha dan tombol untuk melakukan pembatalan edit data surat usaha jika tidak jadi melakukan pembaruan data. Setelah admin meng-klik tombol simpan maka sistem akan kembali ke halaman daftar data surat usaha dan admin dapat melihat bahwa pembaruan data sudah berhasil tersimpan.

- Tampilan halaman hapus data surat keterangan usaha

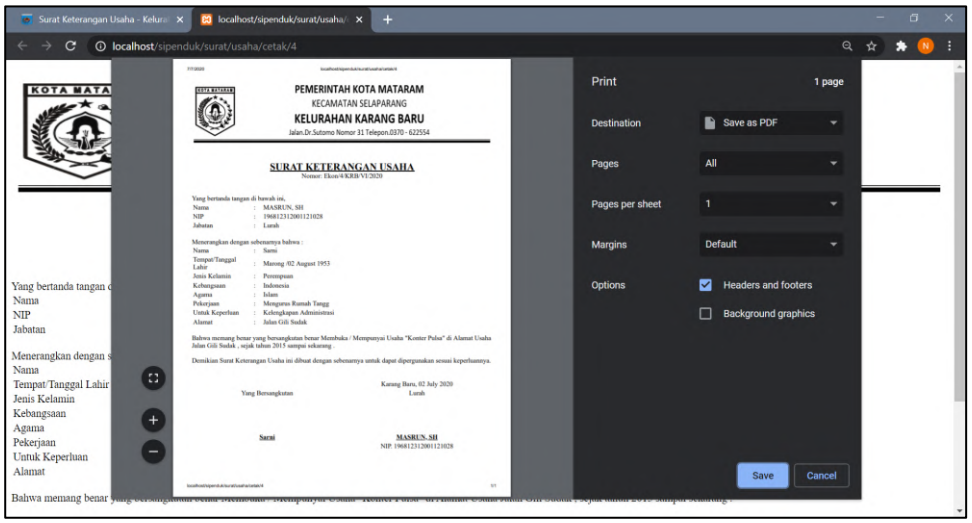


Gambar 4. 98 Implementasi halaman hapus data surat keterangan usaha

Gambar 4.98 merupakan tampilan halaman ketika admin akan melakukan hapus data surat keterangan usaha. Setelah admin mengklik tombol hapus pada salah satu data surat keterangan usaha maka sistem akan

memunculkan dialog konfirmasi untuk penghapusan data. Jika admin meng-klik *OK* maka data surat keterangan usaha yang dipilih akan otomatis terhapus dari daftar surat keterangan usaha. Jika admin tidak jadi melakukan penghapusan data surat keterangan usaha, admin dapat meng-klik pilihan *cancel* pada kotak dialog.

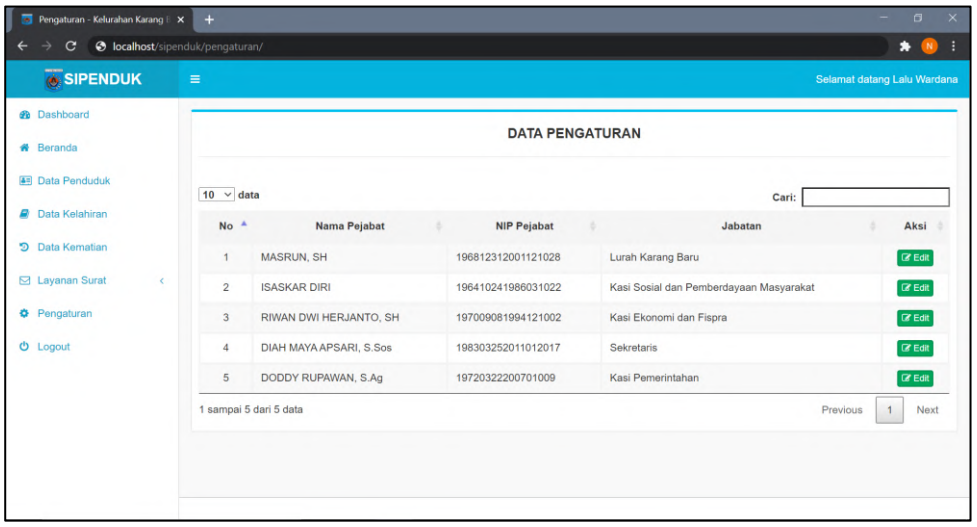
- Tampilan halaman cetak data surat keterangan usaha



Gambar 4. 99 Implementasi halaman cetak data surat keterangan usaha

Gambar 4.99 merupakan tampilan halaman ketika admin memilih fitur untuk melakukan cetak data surat keterangan usaha. Ketika admin meng-klik tombol cetak pada salah satu data surat keterangan usaha maka akan diarahkan ke halaman baru yang akan menampilkan rincian pengaturan untuk mencetak surat keterangan usaha. Hasil pencetakan surat keterangan usaha dapat diatur sesuai kebutuhan, apakah ingin menyimpan surat sebagai berkas PDF atau mencetak surat ke dalam bentuk fisik. Pada bagian bawah terdapat tombol *cancel* yang ketika di-klik akan membatalkan pencetakan surat.

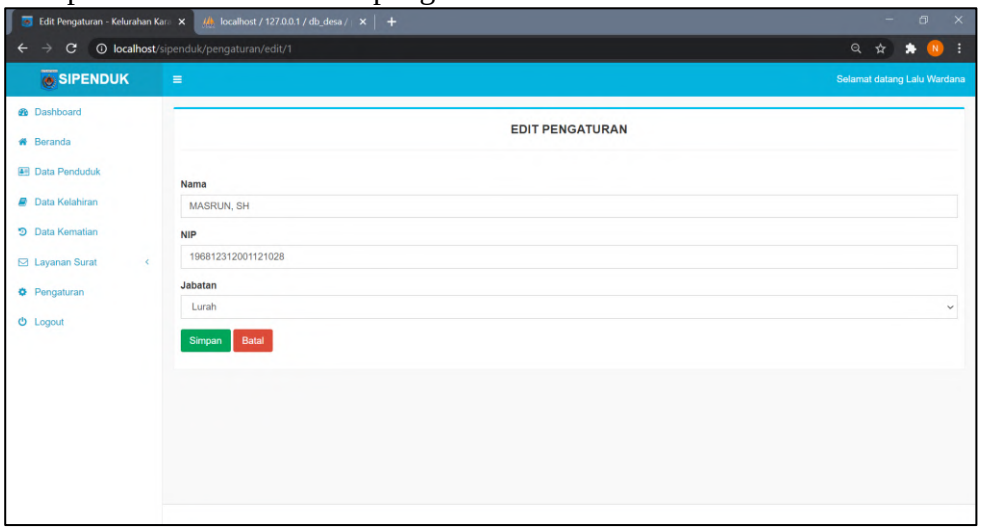
- Tampilan halaman data pengaturan



Gambar 4. 100 Implementasi halaman data pengaturan

Gambar 4.100 merupakan implementasi dari halaman data pengaturan. Halaman ini berfungsi untuk menampilkan data pengaturan dari sistem, dimana pengaturan ini berisi data dari pejabat Kantor Lurah Karang Baru yang berhak untuk menandatangani surat – surat yang keluar dari instansi. Pada halaman ini terdapat fitur cari data pengaturan untuk memudahkan admin dalam menemukan data pengaturan tertentu. Pada bagian aksi hanya terdapat satu fungsi yaitu tombol untuk mengarahkan admin ke halaman edit data pengaturan. Pada bagian bawah halaman juga terdapat tombol untuk menuju ke halaman selanjutnya dari daftar data pengaturan.

- Tampilan halaman edit data pengaturan

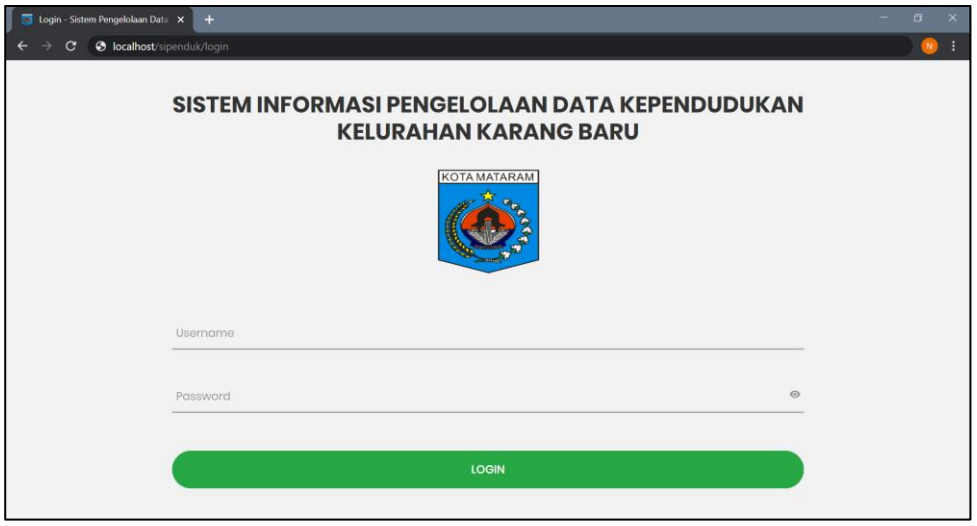


Gambar 4. 101 Implementasi halaman edit data pengaturan

Gambar 4.101 merupakan tampilan *form* untuk admin melakukan edit data pengaturan, dengan *form* ini data pengaturan yang dapat diperbarui oleh

admin adalah data nama, data NIP dan data jabatan. Pada bagian bawah pojok kiri terdapat tombol untuk menyimpan perubahan data pengaturan dan tombol untuk melakukan pembatalan edit data pengaturan jika tidak jadi melakukan pembaruan data. Setelah admin meng-klik tombol simpan maka sistem akan kembali ke halaman daftar data pengaturan dan admin dapat melihat bahwa pembaruan data sudah berhasil tersimpan.

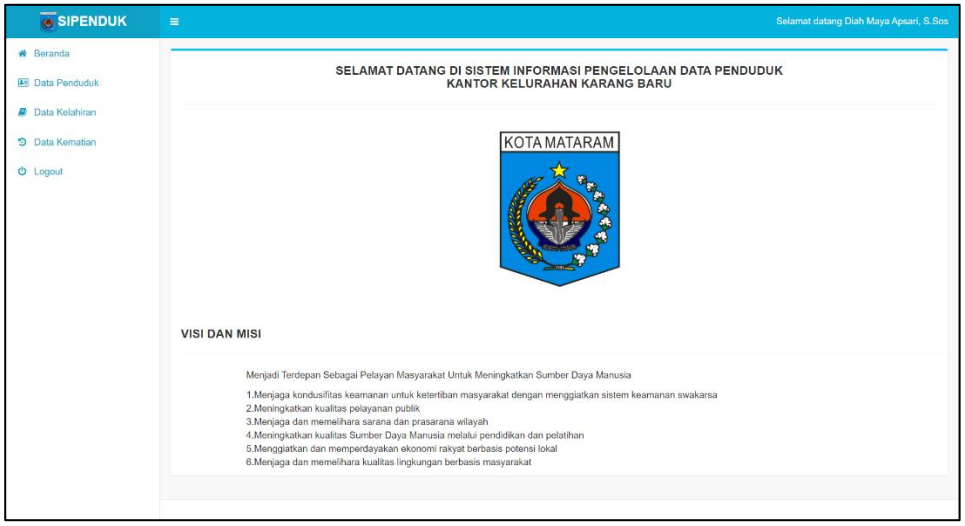
2. Tampilan antarmuka untuk aktor sekretaris :
   a. Tampilan *login*



Gambar 4. 102 Implementasi halaman *login user*

Gambar 4.102 merupakan implementasi halaman *login* dari dari sistem informasi pengelolaan data penduduk Kantor Kelurahan Karang Baru. Halaman ini merupakan tampilan yang akan pertama kali muncul ketika aktor sekretaris mengakses program. Untuk masuk ke dalam sistem sekretaris perlu untuk memasukkan *username* dan *password* yang sesuai dengan data di *database*.

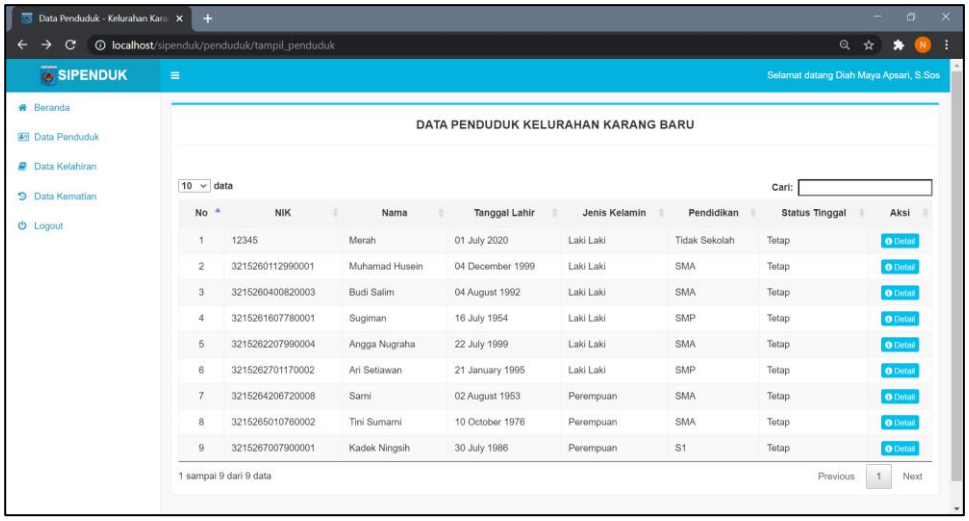
b. Tampilan halaman beranda



Gambar 4. 103 Implementasi halaman beranda.

Gambar 4.103 merupakan implementasi dari halaman beranda. Halaman beranda merupakan halaman yang akan pertama kali ditampilkan setelah sekretaris berhasil melakukan *login* atau masuk ke dalam sistem. Halaman beranda berfungsi untuk menampilkan informasi tentang visi dan misi dari Kelurahan Karang Baru.

c. Tampilan halaman data penduduk

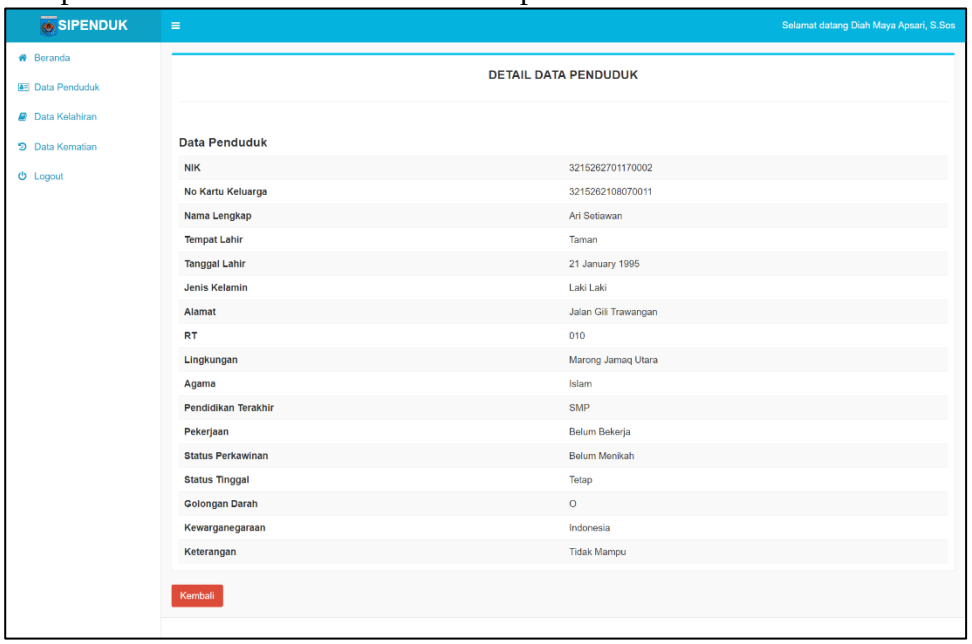


Gambar 4. 104 Implementasi halaman data penduduk.

Gambar 4.104 merupakan implementasi dari halaman data penduduk. Halaman ini berfungsi untuk menampilkan daftar data penduduk Kelurahan Karang Baru. Pada halaman ini fitur yang dapat diakses oleh aktor sekretaris adalah cari data penduduk yang berfungsi untuk memudahkan sekretaris dalam menemukan data penduduk tertentu dan pada bagian aksi terdapat

fitur untuk melihat detail informasi data penduduk yang akan mengarahkan sekretaris ke halaman rincian data penduduk. Pada bagian bawah halaman juga terdapat tombol untuk menuju ke halaman selanjutnya dari daftar data penduduk.

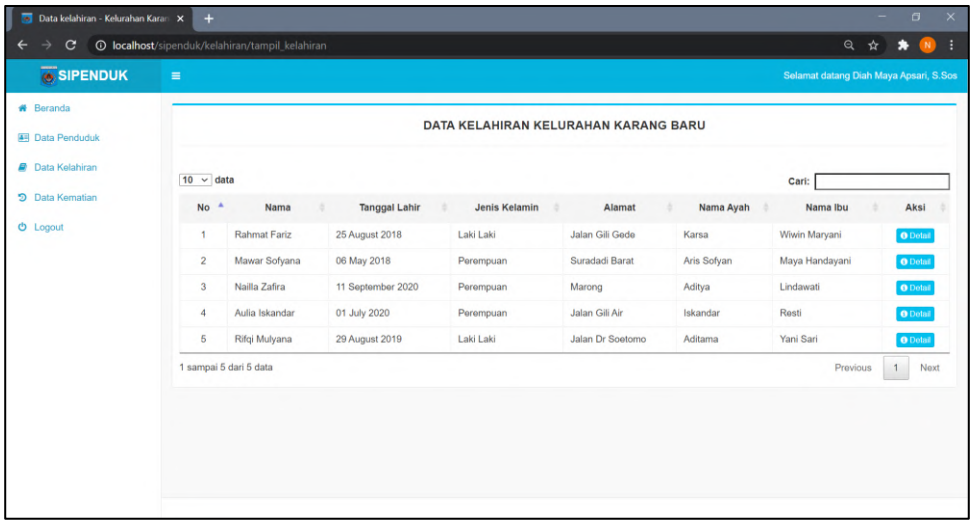
d. Tampilan halaman informasi detail data penduduk



Gambar 4. 105 Implementasi halaman informasi detail data penduduk

Gambar 4.105 merupakan tampilan halaman ketika sekretaris memilih fitur untuk melihat detail data penduduk. Ketika sekretaris meng-klik tombol detail pada salah satu data penduduk maka akan diarahkan ke halaman ini yang akan menampilkan rincian informasi dari data penduduk yang dipilih. Pada bagian bawah terdapat tombol kembali yang ketika di-klik akan mengarahkan sekretaris kembali ke halaman daftar data penduduk.

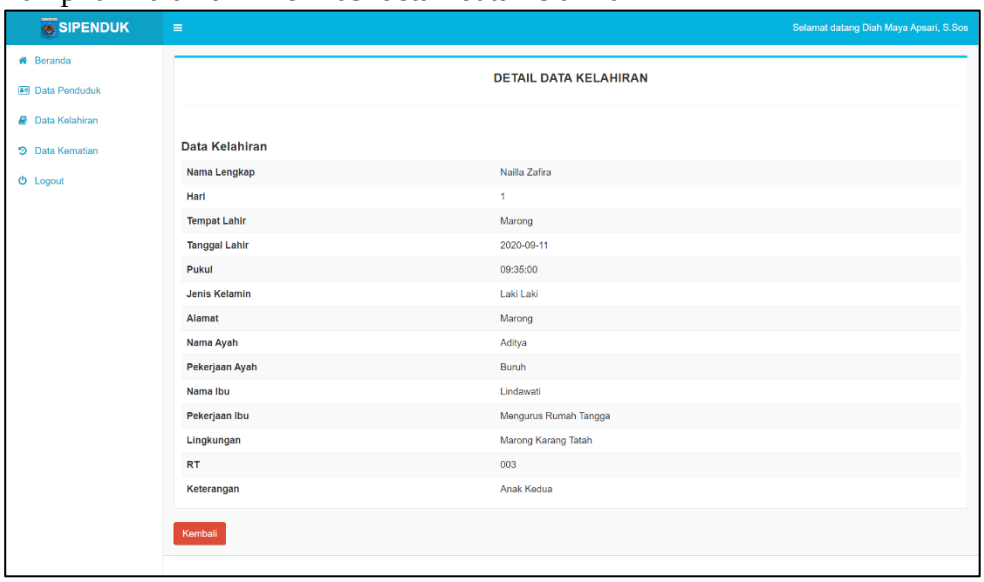
e. Tampilan halaman data kelahiran



Gambar 4. 106 Implementasi halaman data kelahiran.

Gambar 4.106 merupakan implementasi dari halaman data kelahiran. Halaman ini berfungsi untuk menampilkan daftar data kelahiran Kelurahan Karang Baru. Pada halaman ini fitur yang dapat diakses oleh aktor sekretaris adalah cari data kelahiran yang berfungsi untuk memudahkan sekretaris dalam menemukan data kelahiran tertentu dan pada bagian aksi terdapat fitur untuk melihat detail informasi data kelahiran yang akan mengarahkan sekretaris ke halaman rincian data kelahiran. Pada bagian bawah halaman juga terdapat tombol untuk menuju ke halaman selanjutnya dari daftar data kelahiran.

f. Tampilan halaman informasi detail data kelahiran

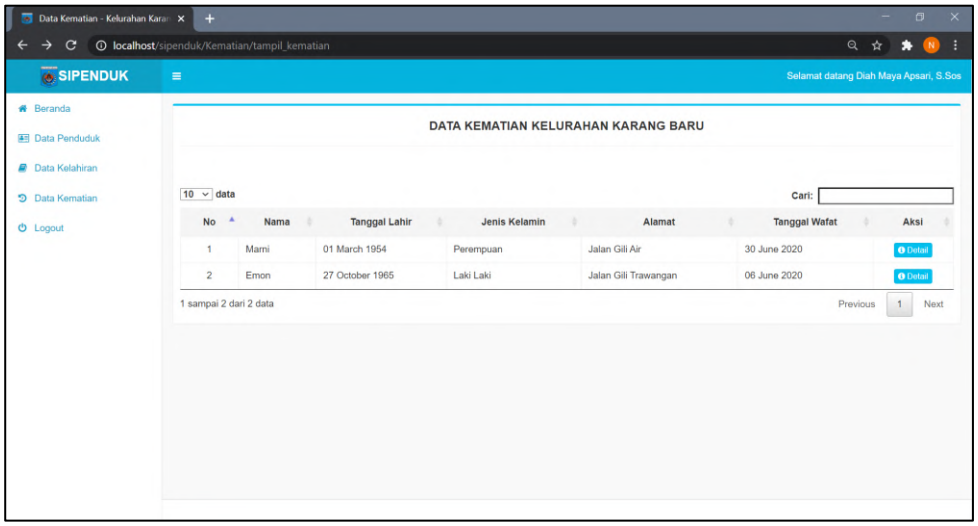


Gambar 4. 107 Implementasi halaman informasi detail data kelahiran

Gambar 4.107 merupakan tampilan halaman ketika sekretaris

memilih fitur untuk melihat detail data kelahiran. Ketika sekretaris meng-klik tombol detail pada salah satu data kelahiran maka akan diarahkan ke halaman ini yang akan menampilkan rincian informasi dari data kelahiran yang dipilih. Pada bagian bawah terdapat tombol kembali yang ketika di-klik akan mengarahkan sekretaris kembali ke halaman daftar data kelahiran.

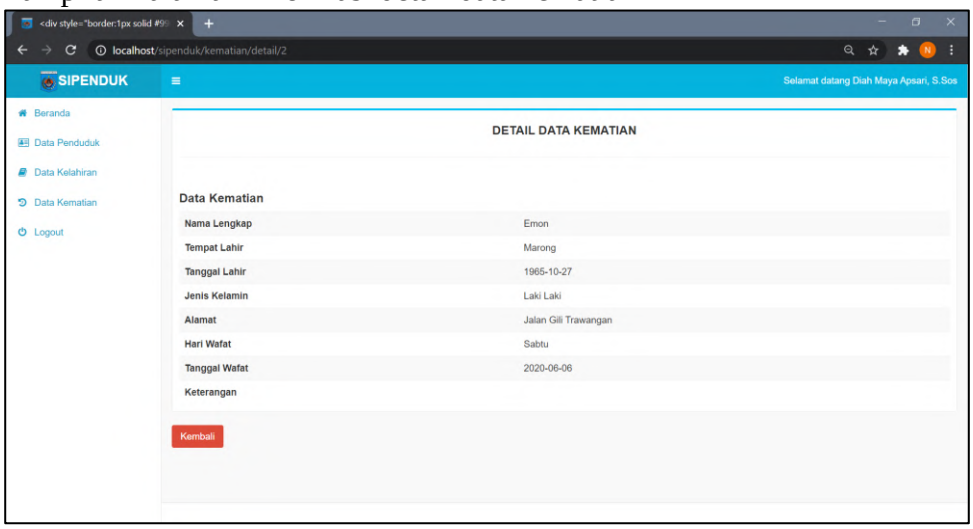
g. Tampilan halaman data kematian



Gambar 4. 108 Implementasi halaman data kematian.

Gambar 4.108 merupakan implementasi dari halaman data kematian. Halaman ini berfungsi untuk menampilkan daftar data kematian Kelurahan Karang Baru. Pada halaman ini fitur yang dapat diakses oleh aktor sekretaris adalah cari data kematian yang berfungsi untuk memudahkan sekretaris dalam menemukan data kematian tertentu dan pada bagian aksi terdapat fitur untuk melihat detail informasi data kematian yang akan mengarahkan sekretaris ke halaman rincian data kematian. Pada bagian bawah halaman juga terdapat tombol untuk menuju ke halaman selanjutnya dari daftar data kematian.

h. Tampilan halaman informasi detail data kematian



Gambar 4. 109 Implementasi halaman informasi detail data kematian

Gambar 4.109 merupakan tampilan halaman ketika sekretaris memilih fitur untuk melihat detail data kematian. Ketika sekretaris meng-klik tombol detail pada salah satu data kematian maka akan diarahkan ke halaman ini yang akan menampilkan rincian informasi dari data kematian yang dipilih. Pada bagian bawah terdapat tombol kembali yang ketika di-klik akan mengarahkan sekretaris kembali ke halaman daftar data kematian.

4.4 Pengujian Sistem

Penulis melakukan pengujian dengan memberikan kuesioner kepada 10 orang responden yang merupakan pegawai Kantor Kelurahan Karang Baru. Hal ini dilakukan untuk mengetahui apakah sistem sudah sesuai dengan kebutuhan sekaligus untuk bahan evaluasi sistem. Pilihan jawaban dari kuesioner terdiri dari 5 kategori yaitu Sangat Setuju (SS), Setuju (S), Biasa Saja (BS), Tidak Setuju (TS) dan Sangat Tidak Setuju (STS). Berikut merupakan hasil dari pengujian yang dilakukan:

Tabel 4.11 Persentase hasil pengujian sistem

No	Pernyataan	Jawaban				
		SS	S	BS	TS	STS
1	Tampilan sistem sudah sangat baik dan mudah digunakan.	4	6	0	0	0
2	Sistem mempermudah pekerjaan pegawai.	8	2	0	0	0
3	Sistem sudah bekerja sesuai kegunaan dan fungsinya.	5	5	0	0	0
4	Sistem sudah dapat menghasilkan keluaran berupa informasi kependudukan dan surat.	5	5	0	0	0

5	Sistem sudah sesuai dengan kebutuhan instansi.	7	3	0	0	0
Total		29	21	**0**	**0**	**0**
Persentase		58%	42%			

Berdasarkan hasil pengujian kuesioner terhadap 10 responden mengenai kepuasan penggunaan Sistem Informasi Pengelolaan Data Kependudukan Kelurahan Karang Baru yang telah dilakukan, maka didapatkan persentase hasil dari pernyataan responden yaitu Sangat Setuju sebesar 58% atau sebanyak 29 poin dan responden yang memilih Setuju sebesar 42% atau sebanyak 21 poin. Sehingga dapat disimpulkan bahwa sistem ini berperan dalam memudahkan dan membantu pegawai Kantor Kelurahan Karang Baru dalam menyelesaikan pekerjaannya.

BAB V
PENUTUP

5.1 Kesimpulan

Berdasarkan hasil Praktik Kerja Lapangan di Kantor Lurah Karang Baru yaitu berupa Sistem Informasi Pengelolaan Data Kependudukan Kantor Kelurahan Karang Baru, dapat diambil beberapa kesimpulan sebagai berikut:

1. Perancangan Sistem Informasi Pengelolaan Data Kependudukan Kantor Kelurahan Karang Baru berbasis website dengan PHP MYSQL dan *Framework* CodeIgniter ini telah dapat diselesaikan dengan didasarkan pada hasil analisis terhadap kebutuhan serta lingkungan kerja di Kantor Lurah Karang Baru.
2. Pembuatan sistem informasi administrasi kependudukan ini dapat membantu dalam pengolahan data kependudukan seperti data penduduk, data kelahiran, data kematian dan data usaha. Dengan adanya sistem informasi ini juga dapat membantu melakukan pengolahan data kependudukan secara efektif dan efisien karena telah menggunakan sistem basis data.
3. Penggunaan sistem informasi ini dapat meminimalisasi terjadinya kesalahan dan kehilangan data karena semua data telah tersimpan di sistem basis data.
4. Pemanfaatan sistem ini dapat memperbaiki kualitas layanan publik di lingkungan Kantor Kelurahan Karang Baru karena dapat menghasilkan informasi dengan lebih cepat.
5. Berdasarkan hasil kuesioner pengujian sistem dengan responden pegawai Kantor Kelurahan Karang Baru, 60% responden setuju bahwa tampilan sistem sudah sangat baik dan mudah digunakan, 80% responden menyatakan sangat setuju bahwa sistem mempermudah pekerjaan pegawai, 50% responden sangat setuju bahwa sistem sudah bekerja sesuai kegunaan dan fungsinya, 50% responden menyatakan sangat setuju pada pernyataan sistem sudah dapat menghasilkan keluaran berupa informasi kependudukan dan surat, serta 70% responden menyatakan sangat setuju bahwa Sistem sudah sesuai dengan kebutuhan instansi. Sehingga dapat disimpulkan bahwa sistem yang dibuat mempunyai kemampuan yang baik jika diterapkan, karena membantu pegawai dalam melakukan pekerjaannya dengan adanya pengolahan data yang menjadi lebih otomatis.

5.2 Saran

Adapun saran yang dapat diberikan penulis agar Sistem Informasi Pengelolaan Data Kependudukan Kantor Kelurahan Karang Baru ini menjadi lebih baik di masa yang akan datang yakni sebgai berikut:

1. Sistem Informasi ini diharapkan kedepannya dapat memasukan data kependudukan secara banyak dengan cara meng-*import* langsung data kependudukan dari excel.
2. Untuk pengembangan selanjutnya diharapkan cakupan sistem menjadi lebih kompleks dan dilengkapi dengan data demografi penduduk, peta potensi SDM dan potensi sarana dan prasarana.

DAFTAR PUSTAKA

[1] A. B. B. Ladjamudin, *Analisis dan Desain Sistem Informasi*. Yogyakarta: Graha Ilmu, 2005.

[2] A. Kadir, *Pengenalan Sistem Informasi*. Yogyakarta: CV.Andi Offset, 2014.

[3] A. Kristanto, *Perancangan SI dan Aplikasinya*. Yogyakarta: Penerbit Gava Media, 2003.

[4] A. Simangunsong, "Sistem Informasi Pengarsipan Dokumen Berbasis Web," Jurnal Mantik Penusa, vol.2, no.1, E-ISSN 2580-9741, pp. 11-19, 2018.

[5] B. Raharjo, I. Heryanto, dan E.Rosdiana K. *Modul Pemograman Web HTML, PHP & MySQL*. Bandung : Modula, 2010.

[6] B. Sidik, *Pemrograman Web dengan PHP*. Bandung : Informatika, 2012.

[7] F. Marisa, *Web Programming untuk Membangun Portal*. Yogyakarta: CV Budi Utama, 2017.

[8] F. T. Yuniko dan F. K. Putra, "Penerapan Teknologi Informasi *Web Programming* untuk Meningkatkan Pelayanan Publik Dalam Bidang Kebijakan Administrasi Kekematianan," JOISIE (*Journal Of Information Systems And Informatics Engineering*), 1(1), 13, 2019.

[9] M. R. Arief, *Pemrograman Basis Data Menggunakan Transact- SQL dengan Microsoft SQL Server 2000.* Yogyakarta: Andi, 2006.

[10] M. R. Arief, *Pemrograman Web Dinamis Menggunakan PHP & MySQL*. Yogyakarta: Andi, 2011.

[11] Marimin, H. Tanjung, dan H. Prabowo, *Sistem Informasi Manajemen Sumber Daya Manusia*. Jakarta: PT Gramedia Widiasarana Indonesia, 2006.

[12] R. A. Sukamto dan M. Shalahuddin, *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek.* Bandung: Informatika, 2014.

[13] Sutarman, *Membangun Aplikasi Web dengan PHP dan MySQL*. Yogyakarta : Penerbit Graha Ilmu, 2003.

[14] Wahana Komputer, *Membuat Aplikasi Database dengan Java dan MySQL*. Yogyakarta : Andi, 2008.