**RANCANG BANGUN SISTEM INFORMASi PENGOLAHAN DATA PENDUDUK DI DESA KALIMANGGIS KECAMATAN**

**MANONJAYA KABUPATEN TASIKMALAYA**

**BERBASIS WEB**

**SKRIPSI**

Diajukan sebagai salah satu syarat untuk memperoleh kelulusan

Jenjang Strata Satu (S1)

Pada program Studi Teknik Informatika

Oleh:

**RISWAN MUHAMMAD RIZKI**

**361743013**



**PROGRAM STUDI TEKNIK INFORMATIKA**

**SEKOLAH TINGGI MANAJEMEN INFORMATIKA & KOMPUTER**

**INDONESIA MANDIRI**

**BANDUNG**

**2021**

# LEMBAR PENGESAHAN

|  |  |
| --- | --- |
| **RANCANG BANGUN SISTEM INFORMASI PENGOLAHAN DATA PENDUDUK DI DESA KALIMANGGIS KECAMATAN**  **MANONJAYA KABUPATEN TASIKMALAYA**  **BERBASIS WEB**  Oleh:  **RISWAN MUHAMMAD RIZKI**  **NIM. 361743013** | |
| Tugas Akhir ini telah diterima dan disahkan untuk  memenuhi persyaratan mencapai gelar  SARJANA TEKNIK INFORMATIKA  Pada  PROGRAM STUDI TEKNIK INFORMATIKA  SEKOLAH TINGGI MANAJEMEN INFORMATIKA &  KOMPUTER INDONESIA MANDIRI | |
| Bandung, Januari 2021  Disahkan oleh: | |
| **Ketua Program Studi,**  **Chalifa Chazar, S.T., M.T.**  NIDN: 0421098704 | **Dosen Pembimbing,**  **Hendra Gunawan, S.T., M.Kom.**  NIDN: 0423037202 |

# LEMBAR PERSETUJUAN REVISI TUGAS AKHIR

**RANCANG BANGUN SISTEM INFORMASI PENGOLAHAN DATA PENDUDUK DI DESA KALIMANGGIS KECAMATAN**

**MANONJAYA KABUPATEN TASIKMALAYA**

**BERBASIS WEB**

Oleh

Riswan Muhammad Rizki

361743013

Telah melakukan sidang skripsi dan telah melakukan revisi sesuai dengan perubahan dan perbaikan yang di mintai pada saat siding tugas akhir.

Bandung, Januari 2021

Menyetujui,

|  |  |  |  |
| --- | --- | --- | --- |
| No | Nama | Keterangan | Tanda Tangan |
| 1 | Hendra Gunawan, S.T., M.Kom. | Pembimbing |  |
| 2 | Chairuddin, Ir., M.T., M.M., Dr. | Penguji 1 |  |
| 3 | Moch. Ali Ramdhani, S.T., M.Kom. | Penguji 2 |  |

Mengetahui

Ketua Program Studi Teknik Informatika

**Chalifa Chazar, S.T., M.T.**

NIDN : 0421098704

**SURAT PERNYATAAN**

**Dengan ini saya menyatakan bahwa :**

1. Naskah Skripsi ini adalah asli dan belum pernah diajukan untuk mendapatkan gelar akademik, baik di Sekolah Tinggi Manajemen Informatika dan Komputer Indonesia Mandiri maupun perguruan tinggi lainnya.
2. Skripsi ini murni merupakan karya penelitian saya sendiri dan tidak menjiplak karya orang lain. Dalam hal ada bantuan atau arahan dari pihak lain maka telah saya sebutkan identitas dan jenis bantuannya di dalam lembar ucapan terima kasih.
3. Seandainya ada karya pihak lain yang ternyata memiliki kemiripan dengan karya saya ini, maka hal ini adalah diluar pengetahuan saya dan terjadi tanpa kesengajaan dari pihak saya.

Pernyataan ini saya buat dengan sesungguhnya dan apabila di kemudian hari terbukti adanya kebohongan dalam pernyataan ini, maka saya bersedia menerima sanksi akademik sesuai norma yang berlaku di Sekolah Tinggi Manajemen Informatika dan Komputer Indonesia Mandiri

Bandung, Januari 2021

Yang membuat pernyataan

Riswan Muhammad Rizki

361743013

**ABSTRAK**

**RANCANG BANGUN SISTEM INFORMASI PENGOLAHAN DATA PENDUDUK DI DESA KALIMANGGIS KECAMATAN**

**MANONJAYA KABUPATEN TASIKMALAYA**

**BERBASIS WEB**

Oleh

Riswan Muhammad Rizki

361743013

Dengan semakin berkembangnya ilmu pengetahuan dan teknologi, semakin bertambah pula kebutuhan manusia mengenai informasi. Diantaranya suatu lembaga pemerintahan membutuhkan aplikasi yang dapat digunakan sebagai media yang dapat mengolah data dengan cepat dan akurat, guna penginputan data penduduk.

Permasalah pun muncul ketika pengguna mencari data penduduk yang di butuhkan secara cepat hal ini disebabkan penyimpanan data masih berupa arsip fisik, sehingga dengan jumlah data yang tersimpan sangat menyulitkan jika mencari data secara cepat. oleh karena itu, perlu ditindak lanjuti dan perlu dicari solusinya. Dalam kesempatan inilah penulis akan merancang sebuah sistem yang asalnya bersifat manual menjadi terkomputerisasi.

Kata Kunci: Sistem Informasi, Lembaga Pemerintahan.

**ABSTRACT**

**DESIGN AND BUILDING INFORMATION SYSTEM FOR POPULATION DATA PROCESSING IN KALIMANGGIS VILLAGE, KECAMATAN**

**MANONJAYA KABUPATEN TASIKMALAYA**

**WEB-BASED**

Oleh

Riswan Muhammad Rizki

361743013

*With the development of science and technology, the human need for information also increases. Among them, a government agency needs an application that can be used as a medium that can process data quickly and accurately, for inputting population data.*

*Problems also arise when users search for population data that is needed quickly. This is because the data storage is still in the form of a physical archive, so the amount of data stored is very difficult to find data quickly. therefore, it is necessary to follow up and find a solution. On this occasion, the author will design a system that is originally manual to be computerized.*

*Keywords: Information System, Government Institution.*

**UCAPAN TERIMA KASIH**

Puji dan syukur kehadirat Illahi Rabbi untuk segala nikmat yang telah diberikan kepada kami, diantaranya nikmat ilmu serta kesehatan yang diberikan sehingga penyusunan Skripsi ini dapat terselesaikan, tidak lupa Shalawat serta Salam semoga tercurahkan kepada Nabi Besar Rasulullah Muhammad Shallallaahu ‘Alaihi Wasallam.

Dalam Penyusunan Skripsi ini, tidak lepas dari berbagai kendala dan hambatan, namun berkat bantuan dan bimbingan dari banyak pihak, akhirnya Skripsi ini dapat diselesaikan. Untuk itu penulis mengucapkan banyak terima kasih berkat dukungan yang diberikan dari berbagai pihak, diantaranya:

1. Bapak Hendra Gunawan, S.T., M.Kom. selaku dosen pembimbing yang telah membantu memberikan ide dan masukan dalam penyusunan Skripsi ini
2. Ibu Chalifa Chazar, S.T., M.T., sebagai Ketua Program Studi Teknik Informatika di STMIK IM Bandung
3. Bapak dan Ibu Dosen Program Studi Teknik Informatika yang telah memberikan ilmu pengetahuan yang bermanfaat bagi penulis
4. Orang tua dan keluarga yang telah memberikan doa dan dukungan serta memfasilitasi kebutuhan untuk menyelesaikan Skripsi ini
5. Teman-teman mahasiswa di STMIK Indonesia Mandiri yang telah memberikan dukungan terbaiknya
6. Dan semua pihak yang tidak dapat penulis sebutkan satu persatu.

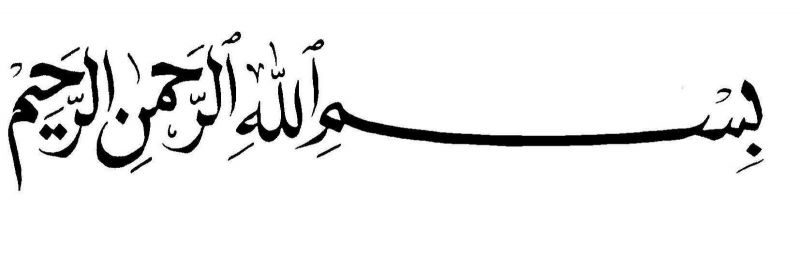
Dengan terselesaikannya Skripsi ini, mudah-mudahan dapat bermanfaat bagi para pembaca umumnya dan juga penulis khususnya. Penulis menyadari bahwa karya tulis ini masih banyak kekurangannya, untuk itu segala kritik dan saran yang sifatnya membangun sangat penulis harapkan dalam rangka menuju hasil yang baik. Semoga Allah Subhanahu Wa Ta’ala senantiasa melimpahkan rahmat dan berkah-Nya kepada kita semua. Amiin

Bandung, Januari 2021

Penulis

Riswan Muhammad Rizki

**KATA PENGANTAR**



Puji dan syukur penulis panjatkan kepada Tuhan Yang Maha Esa atas segala rahmat dan karunia-Nya, sehingga penulis dapat menyelesaikan Skripsi ini dengan judul “**RANCANG BANGUN SISTEM INFORMASI PENGOLAHAN DATA PENDUDUK DI DESA KALIMANGGIS KECAMATAN MANONJAYA KABUPATEN TASIKMALAYA BERBASIS WEB**”.

Skripsi ini berisikan hasil dari seluruh kegiatan yang di lakukan selama penulis melakukan penyusunan Skripsi. Walaupun hasil yang didapat masih terasa kurang tetapi banyak pelajaran yang berharga selama penulis melakukan penelitian.

Dengan segala kerendahan hati penulis menyadari bahwa Skripsi ini masih jauh dari kesempurnaan karena keterbatasan kemampuan penulis. Namun demikian diharapkan agar Skripsi ini dapat memenuhi syarat yang diperlukan.

Semoga Skripsi ini dapat bermanfaat khususnya bagi penulis umumnya bagi pembaca. Penulis menerima kritik dan saran yang membangun untuk lebih mengembangkan keahlian dan kemampuan penulis. Terima kasih.

Bandung, Januari 2020

Penulis

Riswan Muhammad Rizki

**DAFTAR ISI**

LEMBAR JUDUL

LEMBAR PENGESAHAN i

LEMBAR PERSETUJUAN REVISI TUGAS AKHIR ii

SURAT PERNYATAAN iii

ABSTRAK iv

ABSTRACT v

UCAPAN TERIMAKASIH vi

KATA PENGANTAR viii

DAFTAR ISI ix

DAFTAR GAMBAR xiii

DAFTAR TABEL xv

BAB I PENDAHULUAN

* 1. Latar Belakang 1
  2. Identifikasi Masalah 2
  3. Tujuan Penelitian 3
  4. Batasan Masalah 3
  5. Metodologi Penelitian 4
     1. Pengumpulan Data 4
     2. Metode Pengembangan Perangkat lunak 5
  6. Sistematika Penulisan 6

BAB II LANDASAN TEORI

* 1. Teori Dasar 9
     1. Pengertian Sistem Informasi 9
     2. Pengertian Pengolahan Data 9
     3. Pengertian Penduduk 10
     4. Pengertian Desa 11
     5. Pengertian Manajemen Data 12
  2. Gambaran Umum Tempat Penelitian 13
     1. Visi dan Misi 13
     2. Struktur Organisasi 13
  3. Pengertian Pertumbuhan Penduduk 15
     1. Kelahiran (Fertilitas/Natalitas) 16
     2. Kematian (Mortalitas) 17
     3. Migrasi 18
  4. Pengertian Data dan Informasi 19
  5. Pengertian Sistem 21
     1. Karakteristik Sistem 22
     2. Perancangan Sistem 24
     3. Pengertian Sistem Informasi Manajemen (SIM) 24
     4. Analisis dan Perancangan Sistem Informasi 25
     5. Tujuan Perancangan Sistem Informasi 26
     6. Siklus Pengembangan Sistem Informasi 27
  6. Alat Bantu Pembuatan Sistem 27
     1. Entity Relationship Diagram (ERD) 27
     2. Data Flow Diagram (DFD) 29
     3. *Flow* *Chart* 31
  7. Basis Data 33
     1. Pengertian Basis Data 33
     2. *Relasional* *Database* 34
  8. Bahasa Pemrograman 38
     1. Fungsi Bahasa Pemrograman 40
     2. MySQL 40
     3. HTML (*Hypertext Markup Language*) 42
     4. PHP (*Hypertext Preprocessor*) 42
     5. CSS (*Cascading Style Sheet*) 43
     6. Java Script 44
  9. Alat Bantu *Software* 45
     1. Microsoft Visio 45
     2. MySQL 45

BAB III ANALISIS MASALAH DAN PERANCANGAN PROGRAM

* 1. Analisis Sistem 47
     1. Analisis Objek Penelitian 47
     2. Analisis Prosedur Sistem Berjalan 47
     3. Analisis Kebutuhan Fungsional 48
     4. Analisis Kebutuhan Perangkat Lunak 49
     5. Analisis Kebutuhan Perangkat Keras 50
  2. Perancangan Sistem 50
     1. Tujuan Perancangan Sistem 50
  3. Prosedur Yang Akan Di Implementasi 50
     1. *Flowmap* 53
  4. Diagram Konteks 59
     1. DFD Level 0 60
     2. DFD Level 1 61
     3. DFD Level 1 Proses 1 63
     4. DFD Level 1 Proses 2 64
     5. DFD Level 1 Proses 3 65
     6. DFD Level 1 Proses 4 66
     7. DFD Level 1 Proses 5 67
     8. DFD Level 1 Proses 6 66
     9. DFD Level 2 Pengolahan Laporan Untuk Operator 70
     10. DFD Level 2 Pengolahan Laporan Untuk Rw 72
     11. DFD Level 2 Pengolahan Laporan Untuk Rt 73
  5. Perancangan *Entity Relationalship Diagram* (ERD) 73
     1. Perancangan Basis Data 75
     2. Perancangan Antar Muka 79

BAB IV IMPLEMENTASI PROGRAM

* 1. Implementasi Sistem 86
     1. Perangkat Keras dan Perangkat Lunak 86
  2. Deskripsi Fungsi Sistem Yang Akan Dirancang 88
     1. Daftar File *Interface* 88
     2. Implementasi Dokumen File Basis Data 89
  3. Screenshoot / Capture Tampilan Aplikasi 89
     1. Tampilan *Form* Login 90
     2. Tampilan *Form* *Dashboard* 90
     3. Tampilan *Form* Data *User* 91
     4. Tampilan *Form* Data Penduduk 91
     5. Tampilan *Form* Data Rw 92
     6. Tampilan *Form* Data Rt 92
     7. Tampilan *Form* Data Kematian 93
     8. Tampilan *Form* Data Pindahan 93
     9. Tampilan *Form* Laporan Penduduk 94
     10. Tampilan *Form* Laporan Data Kelahiran 94
     11. Tampilan *Form* Laporan Data Kematian 95
     12. Tampilan *Form* Laporan Data Pindahan Masuk 95
     13. Tampilan *Form* Laporan Data Pindahan Keluar 96
     14. Tampilan *Form* Laporan Data Pendidikan 96
  4. Pengujian Sistem 97

BAB V KESIMPULAN DAN SARAN

* 1. Kesimpulan 100
  2. Saran 100

DAFTAR PUSTAKA

Lampiran *Script* Program

# DAFTAR GAMBAR

Gambar 2.1 Hubungan *One-To-One* 35

Gambar 2.2 Hubungan *One-To-Many* 36

Gambar 2.3 Hubungan *Many To Many* 53

Gambar 3.1 Rancangan Flowmap Pengolahan Data Penduduk 53

Gambar 3.2 Rancangan Flowmap Pengolahan Data Kelahiran 54

Gambar 3.3 Rancangan Flowmap Pengolahan Data Kematian 55

Gambar 3.4 Rancangan Flowmap Pengolahan Data Pindah 56

Gambar 3.5 Rancangan Flowmap Pengolahan Data Pindahan 57

Gambar 3.6 Diagram Konteks 58

Gambar 3.7 DFD Level 0 59

Gambar 3.8 DFD Level 1 60

Gambar 3.9 DFD Level 1 Proses 1 61

Gambar 3.10 DFD Level 1 Proses 2 62

Gambar 3.11 DFD Level 1 Proses 3 63

Gambar 3.12 DFD Level 1 Proses 4 64

Gambar 3.13 DFD Level 1 Proses 5 65

Gambar 3.14 DFD Level 1 Proses 6 66

Gambar 3.15 DFD Level 2 Pengolahan Laporan Operator 67

Gambar 3.16 DFD Level 2 Pengolahan Laporan Rw 69

Gambar 3.17 DFD Level 2 Pengolahan Laporan Rt 71

Gambar 3.18 ERD 73

Gambar 3.19 Tampilan *Form* Login 79

Gambar 3.20 Tampilan *Form* *Dashboard* 79

Gambar 3.21 Tampilan *Form* Data Penduduk 80

Gambar 3.22 Tampilan *Form* Data RW 80

Gambar 3.23 Tampilan *Form* Data RT 81

Gambar 3.24 Tampilan *Form* Data Kelahiran 81

Gambar 3.25 Tampilan *Form* Data Kematian 82

Gambar 3.26 Tampilan *Form* Data Pindahan 82

Gambar 3.27 Tampilan Laporan Data Penduduk 83

Gambar 3.28 Tampilan Laporan Data Kelahiran 83

Gambar 3.29 Tampilan Laporan Data Kematian 84

Gambar 3.30 Tampilan Laporan Data Pindah 84

Gambar 4.1 Tampilan *Form* Login 89

Gambar 4.2 Tampilan *Form* *Dashboard* 89

Gambar 4.3 Tampilan *Form* Data *User* 90

Gambar 4.4 Tampilan *Form* Data Penduduk 90

Gambar 4.5 Tampilan *Form* Data RW 83

Gambar 4.6 Tampilan *Form* Data RT 84

Gambar 4.7 Tampilan *Form* Data Kematian 84

Gambar 4.8 Tampilan *Form* Data Pindahan 84

Gambar 4.9 Tampilan Laporan Data Penduduk 86

Gambar 4.10 Tampilan Laporan Data Kelahiran 86

Gambar 4.11 Tampilan Laporan Data Kematian 86

Gambar 4.12 Tampilan Laporan Data Pindahan 86

Gambar 4.13 Tampilan Laporan Data Pindah 86

# DAFTAR TABEL

Tabel 2.1 ERD 27

Tabel 2.2 Simbol Data Flow Diagram 29

Table 2.3 Simbol *Flow* *Chart* 30

Table 2.4 Folder Penting Dalam MySQL 30

Table 3.1 Rancangan Tabel Data Penduduk 75

Tabel 3.2 Rancangan Tabel Data Kematian 76

Tabel 3.3 Rancangan Tabel Data Masuk 77

Table 3.4 Rancangan Tabel Data Pindah 77

Table 3.5 Rancangan Tabel Data Kelahiran 78

Table 4.1 Daftar *File* *Interface* 87

Table 4.2 Daftar *File* Basis Data 88

Table 4.3 Hasil Pengujian Sistem *Login* Admin 87

Table 4.4 Hasil Pengujian Sistem 88

**BAB I**

**PENDAHULUAN**

* 1. **Latar Belakang**

Seiring perkembangan zaman dan pesatnya kemajuan teknologi ini semakin luas penerapannya dalam segi kehidupan agar sesuai dengan kebutuhan dan tidak ketinggalan zaman. Penyesuaian tersebut secara langsung mengubah tatanan dalam sistem makro maupun mikro. Banyak instansi pemerintah atau dinas-dinas yang ada di pemerintah yang mulai menerapkan teknologi tersebut baik dalam proses informasi maupun dalam pembuatan aplikasi terkomputerisasi yang sangat menunjang kinerja yang efektif dalam hal cepatnya arus informasi.

Dengan semakin berkembangnya ilmu pengetahuan dan teknologi, semakin bertambah pula kebutuhan manusia mengenai informasi. Diantaranya suatu lembaga pemerintahan membutuhkan aplikasi yang dapat digunakan sebagai media yang dapat mengolah data dengan cepat dan akurat, guna penginputan data penduduk. Namun informasi yang disajikan masih pendataan catatan dengan manual menggunakan buku tulis. Sehinnga menimbulkan beberapa kendala dalam proses pengerjaanya. Antara lain memerlukan waktu yang lebih lama, tempat dan ruangan yang banyak berkas yang hilang *file* didalam komputer yang hilang dan terkena virus. Hal tersebut menyebabkan infomasi yang disajikan tidak lagi menjadi cepat dan akurat sebagai mana mestinya.

Kependudukan adalah hal ihwal yang berkaitan dengan jumlah, umur, jenis kelamin, agama, kelahiran, perkawinan, kematian, perpindahan, mobilitas dan kualitas serta ketahanannya yang menyangkut politik, ekonomi, sosial dan budaya.

Desa Kalimanggis merupakan suatu wilayah yang ditempati sejumlah penduduk sebagai kesatuan masyarakat yang didalamnya terdapat kesatuan hukum yang memiliki organisasi pemerintahan langsung dibawah camat. Informasi tentang desa merupakan hal yang penting untuk memberikan informasi kepada masyarakat. Bagi masyarakat, biasanya informasi mengenai desa di dapatkan pada saat ada kegiatan desa. Namun walaupun telah mengikuti kegiatan tersebut masih banyak masyarakat yang belum mengetahui secara detail mengenai desanya sendiri.

Saya selaku penulis memiliki analisa kritis terhadap persoalan tersebut yang perlu ditindak lanjuti dan perlu dicari solusinya. Dalam kesempatan inilah penulis akan merancang sebuah sistem yang asalnya bersifat manual menjadi terkomputerisasi yaitu **“Rancang Bangun Sistem Informasi Pengolahan Data Penduduk Di Desa Kalimanggis Kecamatan Manonjaya Kabupaten Tasikmalaya Berbasis Web”**.

* 1. **Identifikasi Masalah**

Sesuai dengan apa yang dipaparkan dalam pendahuluan maka dapat di indentifikasi bahwa dalam aktivitas pengolahan data penduduk terdapat kendala-kendala yang terjadi seperti sulitnya dalam membuat rekapitulasi pengolahan data penduduk dikarenakan data yang dikelola banyak dan digunakan oleh beberapa bagian yang terlibat. Sehingga hal ini pula menyebabkan keterlambatan dalam pembuatan laporan data penduduk.

Permasalah pun muncul ketika pengguna mencari data penduduk yang di butuhkan secara cepat hal ini disebab penyimpanan data masih berupa arsip fisik, sehingga dengan jumlah data yang tersimpan sangat menyulitkan jika mencari data secara cepat.

* 1. **Tujuan Penelitian**

Tujuan yang hendak dicapai oleh penulis dalam merancang dan membangun sistem tersebut adalah:

1. Untuk mempermudah petugas desa dalam mengolah data penduduk.
2. Untuk mempermudah proses pencarian data di Desa Kalimanggis ketika dibutuhkan dan lebih mudah ditemukan karena disimpan didatabase.
3. Untuk mempermudah dan mempercepat dalam mendapatkan informasi tentang Kependudukan.
   1. **Batasan Masalah**

Dari pembahasan diatas dapat diilustrasikan bahwa penulis membatasi permasalahannya yaitu sebagai berikut:

1. Hanya Mencakup Pengolahan Data dan Laporan Penduduk Desa Kalimanggis.
2. Hanya Mencakup Pengolahan Data dan Laporan Kelahiran Penduduk Desa Kalimanggis.
3. Hanya Mencakup Pengolahan Data dan Laporan Kematian Penduduk Desa Kalimanggis.
4. Hanya Mencakup Pengolahan Data dan Laporan Perpindahan Penduduk Desa Kalimanggis.
5. Hanya Mencakup Pengolahan Data dan Laporan Pendatang Penduduk Baru.
   1. **Metodologi Penelitian**

Dalam menyelesaikan Skripsi ini, penulis menggunakan metodologi penelitian sebagai berikut:

* + 1. **Pengumpulan Data**

Data yang diperoleh dari hasil penelitian dikumpulkan sebagai suatu dokumentasi sistem lama. Adapun metodologi pengumpulan data yang digunakan adalah :

1. Observasi

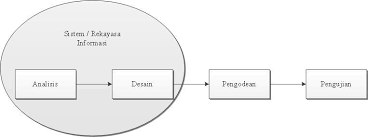
Penulis melakukan pengamatan secara langsung terhadap sistem yang sedang berjalan di lokasi penelitian untuk memperoleh data yang relevan dan akurat.

1. Wawancara

Penulis melakukan wawancara langsung dengan pihak-pihak yang berhubungan dengan materi penelitian.

1. Studi Pustaka
2. Mencari sumber-sumber lain untuk memperkuat dasar teoritis melalui buku-buku, dokumen,serta bahan tulisan yang ada hubunganya dengan masalah yang diteliti.
   * 1. **Metode Pengembangan Perangkat Lunak**

Metode untuk merancang sistem yang digunakan penulis dalam penelitian ini yaitu metode waterfall. Air terjun (waterfall) sering juga disebut model sekuensial linier (sequential linear) atau alur hidup klasik (classic life cycle). Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisis, desain, pengodean, pengujian, dan tahap pendukung (*support*) (Rosa dan Shalahuddin, 29:2013).



Gambar 1.1. Metode Waterfall (Rosa dan Shalahuddin, 29:2013)

1. Perancangan

Perancangan sebuah sistem untuk memperbaiki kekurangan dan kelebihan sistem yang sedang berjalan meliputi :

1. Metode *Waterfall.*
2. Perancangan Basis Data.
3. Perancangan Proses dengan *DFD (Data Flow Diagram*).
4. Perancangan masukan dan keluaaran (*Input and Ouput*).
5. Perancangan struktur menu program aplikasi sistem informasi pengolahan data penduduk.
6. Pengujian

Proses pembuatan program berdasarkan hasil rancangan. Pengujian fokus pada perangkat lunak secara dari segi lojik dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan *(error)* dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.

1. Pendukung (*Support*) atau Pemeliharaan (*Maintenance*)

Tidak menutup kemungkinan sebuah perangkat lunak mengalami perubahan ketika sudah dikirimkan ke user. Perubahan bisa terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru. Tahap pendukung atau pemeliharaan dapat mengulangi proses pengembangan mulai dari analisis spesifikasi untuk perubahan perangkat lunak yang sudah ada, tapi tidak untuk membuat perangkat lunak baru.

* 1. **Sistematika Penulisan**

Untuk memberikan gambaran yang jelas mengenai Skripsi ini, berikut adalah penjelasan mengenai pemaparan setiap bab secara garis besar sebagai berikut:

**BAB I: PENDAHULUAN**

Bab ini menjelaskan tentang beberapa hal seperti, Latar belakang masalah, Identifikasi Masalah, Tujuan Penelitian, Batasan Masalah, Metode penelitian dan sistematika penulisan.

**BAB II: LANDASAN TEORI**

Bab ini menjelaskan beberapa hal mengenai Pengertian dari Penduduk, prosedur kerja, Struktur Organisasi, Pengertian Sistem, Analisis dan Perancangan Sistem Basis Data.

**BAB III:`ANALISIS MASALAH DAN PERANCANGAN PROGRAM**

Bab ini menjelaskan tentang analisa masalah terhadap sistem yang sedang berjalan di Desa Kalimanggis Kecamatan Manonjaya Kabupaten Tasikmalaya, serta dengan solusi yang dapat dilakukan dengan menggunakan komputer dan analisis data yang telah terstruktur. Bab ini juga menjelaskan tentang deskripsi rancangan program aplikasi yang akan dibuat sesuai dengan kebutuhan kantor tersebut, antara lain perancangan basis data (*database*), perancangan kamus data, ERD *(Entity Relationship Diagram)*, DFD (Data *Flow* Diagram), perancangan input dan output sistem di desa tersebut.

**BAB IV:`IMPLEMENTASI DAN UJI COBA**

Bab ini berisikan tentang implementasi atas perancangan sistem baru yang meliputi batasan-batasan terhadap perangkat lunak maupun perangkat keras yang digunakan.

**BAB V: PENUTUP**

Bab ini menjelaskan beberapa hal mengenai Kesimpulan dan Saran setelah selesai mengamati dan meneliti permasalahan yang ada.

**BAB II**

**LANDASAN TEORI**

* 1. **Teori Dasar**
     1. **Pengertian Sistem Informasi**

Sistem informasi adalah kombinasi antara prosedur kerja, informasi, orang, dan teknologi informasi yang diorganisasikan untuk mencapai satu tujuan dalam sebuah organisasi.

Menurut James A. O’Brien (2015) sistem infomasi adalah kombinasi yang terstruktur antara sumber daya manusia, hardware, software, jaringan komunikasi dan sumber data yang dikumpulkan, mengubah dan menyebarkan informasi. Jadi sistem informasi adalah sebuah sistem yang mengumpulkan, mengolah, dan menganalisa data, serta memisahkan informasi untuk tujuan yang lebih spesifik.

* + 1. **Pengertian Pengolahan Data**

Menurut Ladjamudin (2018:9), Pengolahan data adalah masa atau waktu yang digunakan untuk mendeskripsikan perubahan bentuk data menjadi formasi yang memiliki kegunaan.

Menurut Sutarman (2017:4), Pengolahan data adalah proses perhitungan atau transformasi data *input* menjadi informasi yang mudah dimengerti ataupun sesuai dengan yang diinginkan.

Dari pengertian diatas penulis menyimpulkan Pengolahan Data adalah data yang diolah menjadi bentuk yang lebih berarti yang berupa informasi.

* + 1. **Pengertian Penduduk**

Ada beberapa pengertian tentang penduduk menurut para ahli diantaranya sebagai berikut :

1. Menurut Jonny Purba (2017:5) penduduk adalah orang yang matranya sebagai diri pribadi, anggota keluarga, anggota masyarakat, warga negara, dan himpubab kuantitas yang bertempat tinggal di suatu tempat dalam batas wilayah negara pada waktu tertentu.
2. Menurut Srijanti (2018:9) dan a. Rahman (2018:9) penduduk adalah orang yang mendiami suatu tempat dalam wilayah tertentu dengan tanpa melihat status kewarganegaraan yang dianut oleh orang tersebut.
3. Menurut dr.Kartomo (2019:4) penduduk adalah semua orang yang mendiami suatu wilayah tertentu pada waktu tertentu, terlepas dari warga negara atau bukan warga Negara.
4. Menurut Sri Murtono (2015:6), Hassan Suryono (2015:6), Martiyono (2015:6) penduduk adalah setiap orang yang berdomisili atau bertempat tinggal di dalam wilayah suatu negara dalam waktu yang cukup lama.

Dari beberapa definisi diatas maka dapat disimpulkan mengenai pengertian penduduk adalah orang-orang yang berada didalam suatu wilayah yang terikat oleh aturan-aturan yang berlaku dan saling berinteraksi satu sama lain secara terus menerus/kontinu. Dalam sosiologi, penduduk adalah kumpulan manusia yang menempati wilayah geografi dan ruang tertentu.

* + 1. **Pengertian Desa**

Ada beberapa pengertian tentang penduduk menurut para ahli diantaranya sebagai berikut :

* 1. Bintarto berpendapat bahwa desa merupakan perwujudan atau kesatuan geografi, sosial, ekonomi, politik, serta kultural yang terdapat disuatu daerah dalam hubungan dan pengaruhnya secara timbal balik dengan daerah lain.
  2. Pengertian Desa menurut paul H. Landis adalah daerah dimana hubugan pergaulannya ditandai dengan derajat intensitas yang tinggi dengan jumlah penduduk kurang dari 2500 orang.
  3. Rifhi siddiq mengemukakan bahwa desa adalah suatu wilayah yang mempunyai tingkat kepadatan rendah yang dihuni oleh penduduk dengan interaksi sosial yang bersifat homogen, bermatapencaharian di bidang agraris serta mampu berinteraksi dengan wilayah lain di sekitarnya.
  4. Menurut Undang-Undang Nomor 6 Tahun 2014 Tentang Desa yang mengatur tentang Desa. Mengartikan bahwa desa merupakan kesatuan masyarakat hukum yang memiliki batas wilayah yang berwenang untuk mengatur dan mengurus urusan pemerintahan, kepentingan masyarakat setempat berdasarkan prakarsa masyarakat, hak asal usul, dan/atau hak tradisional yang diakui dan dihormati dalam sistem pemerintahan Negara Kesatuan Republik Indonesia.
     1. **Pengertian Manajemen Data**

Definisi manajemen adalah suatu pencapaian sasaran-sasaran organisasi dengan cara yang efektif dan efisien melalui perencanaan pengorganisasian, kepemimpinan dan pengendalian sumberdaya organisasi.

Definisi data menurut Jogiyanto (2015:31), data merupakan bahan yang berupa huruf, simbol, angka gambar, suara atau penggambaran dari kejadian.

Manajemen data menurut Jogiyanto (2015:32) manajemen data merupakan suatu bentuk kegiatan pengelolaan sumber daya informasi yang meliputi, pertama proses pengumpulan data dan pencatatatn ke dalam dokumen yang berfungsi sebagai masukan (*input*) bagi sistem. Kedua, proses penyimpanan sumber daya informasi ke dalam suatu berkas dokumen. Ketiga, pemeliharaan proses penambahan data baru dan perubahan data yang ada supaya sumber daya informasi tetap mutakhir. Keempat, mengatur (*organize*) proses penyusunan data sedemikian rupa untuk dapat memenuhi kebutuhan informasi para pemakai. Dalam konteks ini manajemen data yang dilakukan dari ketiga proses manajemen.

Manajemen data dalam hal ini dilakukan dengan dukungan teknologi informasi. Sehingga, semua proses yang ada di dalamnya dilakukan secara otomatis. Mulai dari proses pencatatan data dilakukan sebagai input data ke dalam sistem. Kemudian setelah dilakukan pencatatan baru kemudian disimpan ke dalam suatu database. Dari data yang di dalam database dapat dilakukan proses pemeliharaan data mulai dari pembuatan data baru dan peruabahan data yang ada. Sehingga, dari data atau informasi yang ada dalam suatu system akan menghasilkan suatu laporan yang bermanfaat bagi organisasi.

* 1. **Gambaran Umum Tempat Penelitian**
     1. **Visi dan Misi**

1. **Visi**

Mewujudkan masyarakat Desa Kalimanggis yang maju mandiri yang dilandasi iman dan taqwa, mewujudkan perekonomian yang tangguh berbasis Agribisnis dan mengembangkan potensi lokal. Mewujudkan pemerintahan yang disiplin dan berakhlaqul karimah.

1. **Misi**
2. Mewujudkan masyarakat yang beriman, bertaqwa, berqualitas dan mandiri.
3. Mewujudkan perekonomian yang tangguh dan meningkatkan pendapatan.
4. Mengembangkan potensi lokal dan respon terhadap program program pemerintahan.
5. Mengenali Sumber Daya Manusia dan Sumber Daya Alam.
6. Mewujudkan tata pemerintahan yang baik *(Good Governance).*
7. Mewujudkan Intrastuktur wilayah yang lebih merata dengan memperhatikan aspek lingkungan yang asri dan lestari.
   * 1. **Struktur Organisasi**

Kepala Desa : Yana, S.H

Sekretaris Desa : Rahmat Setiadi

Staff Kaur Keuangan : Tri Nuraeni

Kadus Pasirpanjang : Triana Murnosa

Kadus Cisitukaler : Edi Junaedi

Kadus Cisitukidul : Iwan Gunawan

Kudus Gununggede : Iin

Kadus Mekarjaya : Surya M Taufik

Kadus Sukarame : Atep Ikbal

Adapun tugas dan tanggung jawab dari masing-masing jabatan adalah sebagai berikut :

* 1. Kepala Desa

Kepala Desa merupakan pimpinan dari pmerintah desa. Masa jabatan kepala desa adalah 6 (enam) tahun, dan dapat diperpanjang lagi utuk satu kali masa jabatan berikutnya. Kepala Desa tidak bertanggung jawab kepada camat, namun hanya dikoordinasikan saja oleh camat.

* 1. Sekretaris Desa

Sekretaris Desa adalah jabatan sekretaris pada pemerintahan desa. Sekretaris Desa merupakan unsur staf pemerintah desa dipimpin oleh seorang sekretaris desa yang berada dibawah dan bertanggung jawab kepada kepala desa.

* 1. Kepala-Kepala Urusan

Kepala urusan kesejahteraan rakyat sebagaimana dimaksud dalam pasal 2 ayat (4) huruf c mempunyai tugas membantu kepala desa dan sekretaris desa menyusun rencana,pelaksanaan,pengendalian evaluasi dan penyusunan laporan pelaksanaan tugas bidang kesejahteraan rakyat.

* 1. Kepala Urusan Umum

Kepala urusan umum (KAUR) yaitu lebih dominan membantu pekerjaan seorang sekretaris desa (Sekdes), namun ia dituntut untuk menguasai tentang tata kelola administrasi. Bisa dikatakan tugas dan fungsinya adalah sebagai TU atau tata usaha dalam suatu kantor.

* 1. Kepala Urusan Keuangan

1. Melaksanakan urusan keuangan seperti pengurusan administrasi keuangan, administrasi sumber-sumber pendapatan dan pengeluaran.
2. Verfikasi administrasi keuangan, dan administrasi penghasilan kepala desa, perangkat desa,BPD, dan lembaga pemeritahan desa lainya.
3. Kepala-Kepala Dusun

Kepala Dusun adalah orang yang mengetahui sebuah dusun, satu wilayah di bawah desa. Satu desa biasanya terdiri dari beberapa RW dan RT.

1. Kepala Urusan Keuangan
2. Melaksanakan urusan keuangan seperti pengurusan administrasi keuangan, administrasi sumber-sumber pendapatan dan pengeluaran.
3. Verfikasi administrasi keuangan, dan administrasi penghasilan kepala desa, perangkat desa,BPD, dan lembaga pemeritahan desa lainya.
   1. **Pengertian Pertumbuhan Penduduk**

Faktor-faktor yang mempengaruhi pertumbuhan penduduk secara umum ada tiga macam yaitu :

* + 1. **Kelahiran (Fertilitas/Natalitas)**

Kelahiran adalah banyaknya bayi yang dilahirkan oleh seorang wanita selama periode suburnya (sekitar usia 15 – 45 tahun). Kemampuan untuk melahirkan disebut fekunditas. Secara umum, (Fertilitas/Natalitas) meliputi tiga macam kelahiran yaitu :

1. Angka Kelahiran Kasar (Crude Birth Rate/ CBR)

Angka kelahiran kasar adalah angka yang menunjukan banyaknya yang lahir hidup setiap 1000 penduduk dalam waktu 1 tahun. Rumus untuk menghitung angka kelahiran kasar : CBR = L/P x 1.000

Keterangan :

CBR : Angka Kelahiran Kasar (Crude Brith Rate)

L : Jumlah kelahiran selama satu tahun

P : Jumlah penduduk pada pertengahan tahun

Adapun tiga kriteria angka kelahiran kasar, yaitu :

1. Angka kelahiran rendah, apabila jumlah kelahiran < 20.
2. Angka kelahiran sedang, apabila jumlah kelahiran antara 20-30
3. Angka kelahiran tinggi, apabila jumlah kelahiran > 30
4. Angka Kelahiran Khusus (Age Specific Birth Rate/ASBR)

Angka kelahiran khusus adalah angka yang menunjukan banyaknya kelahiran setiap 1.000 penduduk wanita pada kelompok usia tertentu. Rumus untuk menghitung angka kelahiran khusus adalah :

ASBR = Li/Pi x 1.000

Keterangan:

ASBR : Age Specific Birth Rate (Angka Kelahiran Khusus)

Li : Jumlah kelahiran dari wanita pada kelompok usia tertentu

Pi : Jumlah penduduk wanita umur tertentu pada pertengahan tahun

1. Angka Kelahiran Umum (General Fertility Rate/GFR)

Angka Kelahiran Umum adalah angka yang menunjukan banyaknya kelahiran setiap 1.000 wanita yang berumur 15-49 tahun dalam setahun. Rumus untuk menghitung :

GFR = L/(W(15-49)) x 1.000

Keterangan :

GFR : General Fertility Rate (Angka Kelahiran Umum)

L : Jumlah kelahiran selama setahun

W (15-49) : Jumlah penduduk wanita umur 15-49 tahun pada pertengahan tahun

* + 1. **Kematian (Mortalitas)**

Kematian atau mortalitas adalah jumlah orang yang meninggal untuk tiap 1.000 penduduk dalam waktu tahun angka kematian dibedakan menjadi 3 macam yaitu :

* 1. Angka Kematian Kasar (Crude Death Rate/CDR)

Angka kematian kasar adalah angka yang menunjukkan kematian setiap 1.000 penduduk dalam waktu setahun. Rumus menghitung Crude Death Rate atau Angka Kematian Kasar, yaitu :

CMR = M/P x 1.000

Keterangan :

CDR : Crude Death Rate (Angka Kematian Kasar)

M : Jumlah kematian selama setahun

P : Jumlah penduduk pertengahan tahun.

Adapun tiga kriteria kematian kasar, yakni :

1. Angka kematian rendah, apabila kematian < 10.
2. Angka kematian sedang, apabila kematian 10-20.
3. Angka kematian tinggi, apabila kematian > 20.
   1. Angka Kematian Khusus (Age specific Death Rate)

Angka kematian khusus adalah angka yang menunjukkan banyaknya kematian sertiap 1.000 penduduk penggolongan tertentu dalam jangka waktu 1 tahun. Rumus unutk menghitung adalah :

ASDR = Mi/Pi x 1.000

Keterangan :

ASDR : Age specific Death Rate (Angka Kematian Khusus)

Mi : Jumlah kematian pada kelompok penduduk tertentu

Pi : Jumlah penduduk pada kelompok tertentu

* + 1. **Migrasi**

Pengertian migrasi adalah perpindahan penduduk dari satu daerah ke daerah lain baik secara berkelompok maupun individu dan bersifat sementara maupun menetap. Orang yang dikatakan telah melakukan migrasi adalah apabila orang tersebut telah melewati batas administrasi wilayah lain. Berikut ini ada jenis-jenis migrasi antara lain sebagai berikut :

* 1. Migrasi masuk yaitu masuknya penduduk dari wilayah ke wilayah lain dengan tujuan untuk menetap.
  2. Migrasi keluar yaitu keluarnya penduduk dari wilayah menuju wilayah lain dengan tujuan untuk menetap.
  3. **Pengertian Data dan Informasi**

Informasi dapat didefinisikan sebagai hasil dari pengolahan data dalam suatu bentuk yang lebih berguna dan lebih berarti bagi penerimanya yang menggambarkan kejadian-kejadian (*event*) yang nyata (*fact*) yang digunakan untuk pengambilan keputusan.

Data adalah keterangan tertulis mengenai sesuatu fakta (kenyataan) yang masih berdiri sendiri-sendiri, belum mempunyai pengertian sebagai kelompok, belum terkoordinasi satu sama lain, dan belum diolah sesuai keperluan tertentu. Informasi adalah data yang sudah diolah dengan cara tertentu menjadi bentuk yang sesuai dengan keperluan pengguna informasi bersangkutan.

Data merupakan kumpulan segala sesuatu yang dapat diterima dan dapat diolah oleh komputer.

Sebagaimana kita ketahui, ledakan informasi merupakan problem nyata pada era komputerisasi yang berkembang cepat ini. Keperluan akan informasi sudah dikenal benar dan bukan jumlah informasi yang penting tetapi nilainya.

Nilai informasi ditentukan oleh lima karakteristik, yaitu:

1. Ketelitian *(accuracy)*

Ketelitian atau akurasi dapat didefinisikan sebagai perbandingan dari informasi yang benar dengan jumlah seluruh informasi yang dihasilkan pada suatu proses pengolahan data tertentu.

1. Ketepatan Waktu *(timeliness)*

Ketepatan waktu merupakan karakteristik informasi lainnya yang penting. Bukan hanya bernilai baru atau lama, tetapi tepat waktu atau setidaknya saat informasi diperlukan. Kendatipun informasinya akurat tetapi kalau diterimanya atau diketahuinya terlambat tentu saja sudah tidak berguna. Akurasi saja tidaklah informasinya akurat tetapi kalau diterima atau diketahuinya tentu saja sudah tidak berguna. Akuerasi saja tidaklah cukup. Seperti halnya pada faktor ketelitian, ketetapan waktu dari suatu informasi pun ssangat tergantung kepada keperluan akan informasi bersangkutan.

1. Kelengkapan *(complete)*

Informasi yang dihasilkan atau diperlukan harus memiliki kelengkapan yang baik, karena bila informasi yang dihasilkan sebagian-sebagian tentunya akan mempengaruhi dalam pengambilan keputusan atau menentukan tindakan secara keseluruhan, sehingga akan berpengaruh terhadap kemampuannya untuk mengontrol atau memecahkan suatu masalah dengan baik.

1. Keringkasan *(conciseness)*

Karena sering menghadapi masalah kurang lengkapan informasi, maka sering terjadi dalam penyediaan suatu informasi diupayakan secara berkelebihan. Informasinya terlalu rinci sehingga dalam pengambilan keputusan berdasarkan informasi tersebut menjadi bingung, karena itu informasi yang bernilai adalah informasi yang ringkas dan langsung dan langsung mengenai sisitem yang diperlukan (*to the point*).

1. Kesesuaian *(relevancy)*

Informasi hendaknya sesuai (*relevan*) dengan tujuan yang akan dicapai.

* 1. **Pengertian Sistem**

Menurut Moscove (2016), sistem adalah suatu kesatuan dari kesatuan yang terdiri atas bagian-bagian yang saling berinteraksi dan berusaha mencapai satu tujuan atau lebih. Menurut James A. Hall (2016) pengertian sistem adalah suatu grup yang terdiri dari atas dua atau lebih komponen atau subsistem yang menjalankan tujuan umum. Menurut James A. O’Brien (2015), sistem dapat didefinisikan sebagai suatu grup yang tediri dari elemen-elemen yang saling berinteraksi secara keseluruhan membentuk kesatuan.

Dapat disimpulkan bahwa sistem memiliki elemen-elemen sebagai berikut:

1. Terdiri dari dua atau lebih komponen atau subsistem.
2. Setiap subsitem mempunyai tujuan yang spesifik.
   * 1. **Karakteristik Sistem**

Suatu Sistem mempunyai karakteristik atau sifat-sifat tertentu, yaitu mempunyai:

1. Komponen Sistem

Terdiri dari jumlah komponen yang saling berinteraksi dan bekerja sama membentuk satu kesatuan. Komponen-komponen dapat terdiri dari beberapa subsistem atau subbagian, dimana setiap sub sistem tersebut memiliki fungsi khusus dan akan mempengaruhi proses sistem secara keseluruhan.

1. Batasan Sistem

Merupakan daerah yang membatasi antara suatu sistem dengan sistem lainnya atau dengan lingkungan luarnya. Batas sistem ini memungkinkan suatu sistem dipandang sebagai suatu kesatuan batas suatu sistem menunjukan ruang lingkup (*scope*) dari sistem tersebut.

1. Lingkungan Luar Sistem

Adalah apapun diluar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar dapat bersifat menguntungkan dan merugikan. Lingkungan yang menguntungkan harus tetap dijaga dan dipelihara. Sebaliknya lingkungan yang merugikan harus ditahan dan dikendalikan kalau tidak ingin terganggu kelangsungan hidup sistem.

1. Penghubung Sistem

Merupakan media penghubung antar sub sistem, yang memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem lainnya. Keluaran (*output*) dari satu subsistem akan menjadi masukan (*input*) untuk subsistem lainnya melalui penghubung disamping sebagai penghubung untuk mengintegrasi subsistem-subsistem menjadi satu kesatuan.

1. Masukan Sistem

Adalah energi yang dimasukan ke dalam sistem, yang dapat berupa masukan perawatan (*maintenance input*) dan masukan sinyal (*signal input*). Masukan perawatan adalah energi yang dimasukan supaya sistem dapat beroprasi, sedangkan masukan sinyal adalah energi yang diproses untuk mendapatkan keluaran. Sebagai contoh didalah sistem komputer, program adalah maintenance input yang digunakan untuk mengoprasikan komputer dan data adalah signal input untuk diolah menjadi informasi.

1. Keluaran Sistem

Adalah hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan. Keluaran dapat merupakan masukan untuk subsistem yang lain. Misalnya untuk komputer panas yang dihasilkan adalah keluaran yang tidak berguna dan merupakan hasil sisa pembuangan, sedangkan informasi adalah keluaran yang dibutuhkan.

1. Pengolahan Sistem

Suatu sistem dapat mempuyai suatu bagian pengolahan yang dapat merubah masukan menjadi keluaran. Suatu sistem produksi akan mengolah masukan berupa bahan baku dan bahan-bahan lain menjadi keluaran berupa barang jadi. Sistem akuntasi akan merubah data-data transaksi menjadi laporan-laporan keuangan dan laporan-laporan lain yang dibutuhkan oleh manajemen.

1. Sasaran Sistem

Suatu sistem pasti mempunyai tujuan (*goal*) atau sasaran (*objective*). Kalau suatu sistem tidak mempunyai sasaran, maka operasi sistem tidak akan ada gunanya sasaran dari sistem sangat menentukan sekali masukan yang dibutuhkan sistem dan keluaran yang akan dihasilkan sistem. Suatu sistem dikatakan bagus bila mengenai sasaran ataw tujuan.

* + 1. **Perancangan Sistem**

Perancangan sistem adalah tahap awal dimana pendekatan awal untuk menyelesaikan masalah dipilih. Selama perancangan sistem, struktur keseluruhan diputuskan. Arsitektur sistem adalah cara pengorganisasian sistem ke dalam susbsistem.

* + 1. **Pengertian Sistem Informasi Manajemen (SIM)**

Menurut George M.Scott dalam Rohmat Taufiq (2018 : 58) “SIM adalah kumpulan dari interaksi-interaksi sistem-sistem informasi yang menyediakan informasi baik untuk kebutuhan manajerial ataupun kebutuhan operasi.”

Menurut Knnetch C.L dan Jane P.L dalam Rohmat Taufiq (2018 : 58) “SIM adalah suatu kegiatan yang menggabungkan teori-teori ilmu komputer, ilmu manajemen, operasi riset dengan orientasi praktis untuk menggabungkan sistem solusi terhadap masalah-masalah sehari-hari dan mengelola susmber informasi tersebut.

Dari beberapa definisi Sistem Informasi Manajemen diatas bisa disimpulkan bahwa Sistem Informasi Manajemen (SIM) adalah kumpulan dari sub-sub sisem yang saling berhubungan atau terintergrasi antara sistem satu dengan sistem yang lain, yang bertujuan untuk membantu manajemen dalam dalam mendapatkan informasi yang berkualitas.

* + 1. **Analisis dan Perancangan Sistem Informasi**

Dalam membuat suatu sistem yang akan digunakan oleh suatu perusahaan, perlu disusunnya suatu perancangan dari sistem yang akan dibuat. Rancangan tersebut bertujuan untuk memberikan gambaran secara umum mengenai sistem yang akan berjalan.

Menurut Satzinger, Jackson, dan Burd (2017:5), perancangan sistem adalah sekumpulan aktivitas yang menggambarkan secara rinci bagaimana sistem akan berjalan. Hal itu bertujuan untuk menghasilkan produk perangkat lunak yang sesuai dengan kebutuhan User.

Adapun tahapan yang dilakukan di dalam analisis sistem diantaranya:

1. *Identify,* yaitu mengidentifikasi masalah
2. *Understand,* yaitu memahami kerja dari sistem yang ada
3. *Analizy,* yaitu membuat laporan hasil analisis
4. *Report,* yaitu membuat laporan hasil analisis
   * 1. **Tujuan Perancangan Sistem Informasi**

Menurut Satzinger, Jackson, dan Burd (2017:13) Perancangan sistem informasi pada umumnya memiliki dua tujuan yang utama, yakni:

1. Memenuhi kebutuhan kepada pemakai sistem
2. Memberikan gambaran yang jelas dan rancang bangun yang lengkap untuk keperluan pembuatan laporan.

Sedangkan tujuan lainnya dari perancangan sebuah sistem informasi, di antaranya:

1. Agar sistem menghasilkan informasi yang tepat waktu dan relevan untuk pengambilan keputusan manajemen
2. Agar elemen-elemen sistem informasi mempunyai nilai manfaat lebih besar dari biaya
3. Agar keluaran sistem harus mempunyai tingkatan ketelitian yang tinggi dan sistem dapat beroperasi secara efektif.
4. Agar sistem harus mempunyai kapasitas yang memadai untuk menangani periode operasi puncak.
5. Agar sistem harus cukup sederhana sehingga struktur dan operasi dapat dengan mudah dimengerti dan prosedurnya mudah dipahami
6. Agar sistem harus fleksibel untuk menampung perubahan-perubahan kepentingan.
   * 1. **Siklus Pengembangan Sistem Informasi**

Secara umum siklus atau tahapan dalam pengembangan sistem informasi adalah:

1. Siklus pengembangan sistem dimulai dengan adanya kebijakan dan perencanaan sistem informasi yang baru. Hal ini dapat disebabkan adanya kebutuhan informasi, teknologi yang baru atau ketidak sesuaian sistem yang ada dalam memenuhi kebutuhan pemakai sistem informasi.
2. Analisis terhadap kebutuhan-kebutuhan pemakai sistem informasi. Tahap analisis sistem, dimana suatu rencana dikembangkan untuk mengatasi kelemahan-kelemahan sistem yang ada.
3. Perancangan sistem yang baru disesuaikan dengan kebutuhan-kebutuhan yang diperlukan oleh pemakai sistem informasi.
4. Tahap implementasi, merupakan tahap pembuatan dan pengujian sistem yang baru apakah sudah sesuai dengan kebutuhan pemakai sistem informasi.
5. Sistem informasi yang baru dapat beroperasi setelah sistem itu menghasilkan informasi yang sesuai dengan kebutuhan pemakai sistem informasi.
6. Pemeliharaan (*maintenance*) dan modifikasi.
   1. **Alat Bantu Prancangan Sistem**
      1. ***Entity Relationship Diagram* (ERD)**

Sukamto dan Shalahuddin (2019:289), “*Entitiy Relationship Diagram* (ERD) adalah pemodelan awal basis data yang akan dikembangkan berdasarkan teori himpunan dalam bidang matematika untuk pemodelan basis data relasional”.

Sukamto dan Shalahuddin (2019:50), ERD memiliki beberapa aliran notasi seperti notasi Chen (dikembangkan oleh Peter Chen). Barker (dikembangkan oleh Richard Barker, Ian Palmer, Harry Ellis), notasi Crow’s Foot, dan beberapa notasi lain. Namun yang banyak digunakan adalah notasi dari Chen. Berikut adalah simbol-simbol yang digunakan pada ERD dengan notasi Chen:

**Tabel 2.1. *Entity Relationship Diagram* (Sutanta : 2015)**

|  |  |
| --- | --- |
| Lambang | Kegunaan |
|  | Menyatakan himpunan *entity*/entitas. |
|  | Menyatakan atribut, atribut dari entity/entitas. |
|  | Menyatakan himpunan relasi. |
|  | Sebagai penghubung antar himpunan relasi dengan himpunan entity/entitas dan himpunan entity/entitas dengan atributnya. |
|  | Menunjukan *key Atribut.* |

* + 1. ***Data Flow Diagram* (DFD)**

Sukamto dan Shalahuddin (2019:288),  *Data Flow Diagram* atau dalam bahasa Indonesia menjadi Diagram Alir Data adalah refresentasi grafik yang menggambarkan aliran informasi dan transformasi informasi yang diaplikasikan sebagai data yang mengatur dari masukan (*input*) dan keluaran (*output*). DFD tidak sesuai untuk memodelkan sistem yang menggunakan pmrograman berorientasi objek.

Sukamto dan Shalahuddin (2019:71), notasi-notasi pada *DFD* (Edward Yourdon dan Tom Demarco) adalah sebagai berikut:

Dalam merancang sistem disini penulis menggunakan pendekatan secara top down. Yaitu memulai perancangan dari bentuk global, diantaranya Diagram Konteks, kemudian Diagram Konteks ini diturunkan ke dalam bentuk yang paling detil lagi. Langkah-langkah secara rinci dari strategi perancangan yang akan digunakan penulis adalah sebagai berikut :

1. Pertama akan dideskripsikan sistem dengan membuat Diagram Konteks, yaitu model yang menggambarkan hubungan sistem dengan hubungan. Untuk menggambarkan Diagram Konteks perlu dideskripsikan data apa saja yang dibutuhkan sistem dan kemana saja data atau informasi tersebut akan diberikan.
2. Setelah berhasil membuat atau menggambarkan Diagram Konteks, Diagram Konteks tersebut akan diturunkan menjadi bentuk yang lebih detil lagi, yaitu DFD Level 0. Untuk menurunkan Diagram Konteks menjadi DFD Level , terlebih dahulu penulis akan menganalisa sistem untuk mendefinisikan proses apa saja yang terdapat dalam sistem tersebut.
3. Bila terdapat proses dalam DFD Level 0 dirasa kurang detil, maka penulis akan menurunkan lagi proses tersebut kedalam DFD Level 1 dari proses tersebut. Dan kalau masih dirasa kurang detil lagi maka penulis akan terus menurunkan proses tersebut sehingga akan didapat proses yang kita harapkan.
4. Dari Diagram Konteks, dapat dilihat informasi apa saja yang mengalir dari dan kedalam sistem. Bentuk informasi yang detil tersebut penulis catat sebagai data dictionary.
5. Untuk setiap proses paling detil dari DFD yang telah dibuat, maka dideskripsikan proses tersebut secara lebih jelas dengan menggunakan spesifikasi proses.
6. Langkah berikutnya adalan pembuatan Entity Relationship Diagran (ERD) dan definisi atribut, yang merupakan rancangan Basis Data dari Sistem.

**Tabel 2.2. Simbol Data Flow Diagram (Wijaya : 2017)**

|  |  |  |
| --- | --- | --- |
| Lambang | Nama Lambang | Kegunaan |
|  | External Entity | Simbol ini digunakan untuk asal atau tujuan data. |
|  | Proses | Simbol ini digunakan untuk proses pengolahan atau transformasi data. |
|  | Data Flow | Simbol ini digunakan untuk aliran data yang sedang berjalan. |
|  | Data Store | Simbol ini digunakan untuk data flow yang sudah disimpan atau diarsipkan. |

* + 1. ***Flow Chart***

Indrajani (2015:36), “*Flow chart* adalah penggambaran secara grafik dari langkah-langkah dan urutan prosedur suatu program.”

Indrajani (2015:38), menjelaskan simbol-simbol dalam *Flow Chart* adalah sebagai berikut :

**Tabel 2.3. Simbol-simbol *Flow Chart* (Jogianto : 2015)**

|  |  |
| --- | --- |
| **SIMBOL** | **KETERANGAN** |
|  | Simbol yang menyatakan bagian dari program (sub program). |
|  | Simbol masukan atau keluaran dari atau ke sebuah pita *magnetic*. |
|  | Simbol *input/output* yang mendefinisikan masukan dan keluaran proses. |
|  | Simbol konektor untuk menyambung proses pada lembar kerja yang sama. |
|  | Simbol konektor untuk menyambung proses pada lembar kerja yang berbeda. |
|  | Simbol masukan atau keluaran dari atau ke sebuah dokumen. |
|  | Simbol untuk memutuskan proses lanjutan dari kondisi tertentu. |
|  | Simbol database atau basis data. |
|  | Simbol yang menyatakan piranti keluaran, seperti layar monitor, *printer,* dll. |
|  | Simbol yang mendefinisikan proses yang dilakukan secara manual. |
|  | Simbol untuk menghubungkan antar proses atau antar simbol. |

* 1. **Basis Data**

Basis data (*Database*) dapat dibayangkan sebagai sebuah lemari arsip. Jika kita memiliki sebuah lemari arsip dan bertugas untuk mengelolanya, maka kemungkinan besar kita akan melakukan hal-hal seperti : memberi sampul pada kumpulan arsip yang akan disimpan, menentukan jenis arsip, memberi penomoran dengan pola tertentu yang nilainya unik pada setiap sampul, lalu menempatkan arsip-arsip tersebut dengan cara tertentu dalam menenpatkan sesuatu yang kita simpan dalam lemari dengan harapan suatu saat nanti, sewaktu kita bermaksud untuk mencari dan mengambil kembali arsip atau yang lain dari lomari masing-masing kita dapat melakukannya dengan mudah dan cepat.

* + 1. **Pengertian Basis Data**

Basis data terdiri dari dua kata, yaitu Basis dan Data. Basis dapat diartikan sebagai markas atau gudang, tempat bersarang atau berkumpul. Sedangkan data adalah representasi fakta dunia nyata yang mewakili suatu objek seperti manusia

(pegawai, siswa, pembeli, pelanggan,) barang, hewan, peristiwa, konsep, keadaan, dan sebagainya, yang direkam dalam bentuk angka, huruf, simbol, teks, gambar, bunyi, atau kombinasinya.

* + 1. ***Relasional* *Database***

Menurut Kristanto (2016 : 22) sebuah sistem manajemen basis data relasional atau dalam bahasa Inggrisnya dikenal sebagai *Relational database management sistem (RDBMS)* adalah sebuah program komputer (atau secara lebih tipikal adalah seperangkat program komputer) yang didesain untuk mengatur/memanajemen sebuah basis data sebagai sekumpulan data yang disimpan secara terstruktur, dan melakukan operasi-operasi atas data atas permintaan penggunanya. Contoh penggunaan *DBMS* ada banyak sekali dan dalam berbagai bidang kerja, misalnya akuntansi, manajemen sumber daya manusia, dan lain sebagainya.

Meskipun pada awalnya *DBMS* hanya dimiliki oleh perusahaan-perusahaan berskala besar yang memiliki perangkat komputer yang sesuai dengan spesifikasi standar yang dibutuhkan (pada saat itu standar yang diminta dapat dikatakan sangat tinggi) untuk mendukung jumlah data yang besar, saat ini implementasinya sudah sangat banyak dan adaptatif dengan kebutuhan spesifikasi data yang rasional sehinggal dapat dimiliki dan diimplementasikan oleh segala kalangan sebagai bagian dari investasi perusahaan.

Peralatan untuk menetapkan/menentukan pendekatan *database* disebut *DBMS*. *DBMS* merupakan *software* dan *hardware* yang khusus didesain untuk melindungi dan me-*manage database.*

Dengan menggunakan *DBMS* maka dapat :

1. Mendefinisikan data dan hubungannya.
2. Mendokumentasikan struktur dan definisi data
3. Menggambarkan, mengorganisasikan dan menyimpan data untuk akses yang selektif/dipilih dan efisien.
4. Hubungan yang sesuai antara *user* dengan sumber daya data.
5. Perlindungan terhadap sumber daya data akan terjamin, dapat diandalkan, konsisten dan benar.
6. Memisahkan masalah *Logical* dan *physical* sehingga merubah implementasi *database* secara fisik tidak menghendaki *user* untuk merubah maksud data (*Logical*).
7. Menentukan pembagian data kepada para *user* untuk mengakses secara concurent pada sumber daya data.

Berikut adalah contoh-contoh *DBMS* :

1. *Database Hierarchy* : Pengaksesan data harus mengikuti aturan *hierarchy* yang sudah didefinisikan terlebih dahulu.
2. *Data Network* : Data membentuk jaringan yang lebih bebas dari model *hierarchy*.
3. *Data Relational* : Data dikelompokkan secara bebas menurut jenisnya lewat proses normalisasi.

Relasional *database* menurut Kristanto (2016) adalah ―hubungan satu *file* data dengan *file* data lain, dimana *file-file* tersebut dapat saling berhubungan melalui suatu atribut yang dijadikan kunci.

Ada beberapa macam relasional *database* yang digunakan dalam kegiatan manajemen *database* adalah :

1. Relasional satu ke satu, adalah hubungan yang terjadi antara satu *file* dengan *file* lain, dihubungkan oleh satu *file* kunci lainnya. Artinya satu *record* pada *entity* A ber-relasi paling banyak satu *record* juga pada *entity* B, begitu juga sebaliknya, satu *record* pada *entity* B, ber-relasi paling banyak satu *record* juga dengan *entity* A.

Contoh : Dalam proses belajar mengajar secara privat misalnya, seorang (satu) *tutor* hanya mengajar satu siswa, begitu juga sebaliknya, satu siswa hanya diajar oleh satu *tutor*.



Gambar 2.1. Hubungan One-to-one (Kristanto : 2016)

1. Relasional satu ke banyak, adalah hubungan yang terjadi antara satu *file* dengan *file* yang lain, yamg dihubungkan dengan *file* kunci dari *file* yang satu ke beberapa *field* kunci yang lain. Pada relasi ini satu *record* pada *entity* A ber-relasi dengan beberapa *record* pada *entity* B, tapi tidak sebaliknya, setiap *record* pada *entity* B ber-relasi paling banyak satu *record* dengan *entity* A. Dalam *diagram* E-R, relasi ini disimbolkan dengan angka **1** untuk menyatakan satu dan huruf **M** atau **N** untuk menyatakan banyak.

Contoh : Dalam sebuah bagian di instansi tertentu memiliki banyak pegawai tapi seorang pegawai tidak dapat bekerja pada banyak bagian.



Gambar 2.2. Hubungan One-to-many (Kristanto : 2016)

1. Relasional banyak ke banyak, adalah hubungan yang terjadi antara satu *file* dengan *file* yang lain, dihubungkan oleh beberapa *field* kunci dari *file* satu kebeberapa *field* kunci *file* yang lain. beberapa *record* pada *entity* A dapat ber relasi dengan beberapa *record* juga pada *entity* B, begitu juga sebaliknya, beberapa *record* pada *entity* B dapat ber-relasi dengan beberapa *record* juga pada *entity* A.. Dalam *diagram* E-R, relasi ini disimbolkan dengan huruf **M** atau **N** untuk menyatakan banyak.

Contoh : Dalam hubungan antara mahasiswa dengan mata kuliah pada perguruan tinggi, seorang mahasiswa dapat mengambil banyak mata kuliah, sebaliknya satu mata kuliah dapat diambil oleh beberapa mahasiswa, sehingga terjadi hubungan banyak ke banyak.



Gambar 2.4. Hubungan Many-to-many (Kristanto : 2016)

* 1. **Bahasa Pemrograman**

Bahasa pemrograman adalah istilah yang menerangkan bahasa yang dapat dimengeri oleh komputer. Dalam dunia komputer dikenal dua jenis bahasa pemrograman, yaitu bahasa pemrograman tingkat tinggi dan bahasa pemrograman tingkat rendah (Kasiman Peranginangin 2016 : 2).

Sebuah bahasa pemrograman atau bahasa komputer di artikan sebagai teknik komunikasi yang terstandarisasi untuk menyatakan instruksi kepada komputer. Bahasa pemrograman adalah kesatuan aturan sintaksis dan semantik yang di gunakan untuk mendefinisikan program.

Sebuah bahasa pemrograman memungkinkan *programmer* untuk menentukan secara tepat apa yang harus dilakukan oleh komputer, cara mengatur transfer data, input-output, dan kegiatan-kegiatan yang harus dilakukan dalam berbagai keadaan. Bahasa pemrograman tingkat rendah merupakan bahasa pemrograman yang sulit dimengerti oleh manusia, sedangkan bahasa pemrograman tingkat tinggi lebih mudah dipahami oleh manusia.

Perkembangan teknologi yang sangat cepat menghasilkan berbagai macam kebutuhan komputasi yang beraneka ragam. Oleh karena itu, muncul berbagai bahasa pemrograman yang berguna untuk memecahkan berbagai macam masalah dengan bantuan komputer. Tiap-tiap bahasa pemrograman pada umumnya diciptakan untuk kondisi yang berbeda-beda. Tidak ada satu bahasa pemrograman pun yang bisa dikatakan unggulsecara mutlak, sebab bahasa-bahasa tersebut diciptakan dengan asumsi-asumsi tertentu dan berjalan di atas sistem operasi atau persyaratan tertentu.

Bahasa pemrograman yang penulis gunakan dalam proses desain sistem informasi ini adalah *PHP* dan *MySQL* sebagai RDBMS (*Relational Database Management Sistem*) dengan mempertimbangkan sebagai berikut :

1. *PHP* merupakan salah satu dari sejumlah bahasa pemrograman yang sangat popular karena dapat menghasilkan berbasis web dan bisa dibuat dinamis, sehingga tampilan program tidak membuat user jenuh.
2. Kemampuan untuk menempatkan fungsi dan prosedur dalam sebuah *file* program sama.
3. Dapat membentuk *file* indek yang lebih cepat dan kompatibel.
4. Keistimewaan adalah bentuk desain yang sangat indah.
5. Program sumber sangat terlindungi.
   * 1. **Fungsi Bahasa Pemrograman**

Fungsi bahasa programan yaitu memerintah komputer untuk mengolah data sesuai dengan alur berpikir yang kita inginkan. Keluaran dari bahasa pmrograman tersebut berupa program/aplikasi. Contohnya adalah program yang digunakan oleh kasir di mal-mal atau swalayan, penggunaan lampu lalu lintas di jalan raya.

Bahasa pemrograman yang kita kenal adalah banyak sekali dibelahan dunia, tentang ilmu komputer dan teknologi dewasa ini. Perkembangannya mengikuti tingginya inovasi yang dilakukan dalam dunia teknologi. Contoh bahasa pmrograman yang kita kenal antara lain adalah untuk membuat aplikasi game, antivirus, web, dan teknologi lainnya.

* + 1. **MySQL**

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (bahasa Inggris: *database management system*) atau DBMS yang *multithread*, *multi-user*, dengan sekitar 6 juta *instalasi* (pemasangan) diseluruh dunia.

Kadir (2018:15), “MySQL adalah nama *database server. Database server* adalah *server* yang berfungsi untuk menangani *database. Database* adalah suatu pengorganisasian data dengan tujuan memudahkan penyimpanan dan pengaksesan data. Dengan menggunakan MySQL, kita bisa menyimpan data dan kemudian data bisa diakses dengan cara yang mudah dan cepat.”

Menurut Enterprise (2019:2), *mysql* yaitu *RDBMS* yang cepat dan mudah digunakan, sudah banyak digunakan untuk berbagai kebutuhan, serta dikembangkan oleh *mysql AB* swedia. MySQL memiliki beberapa keistimewaan, antara lain :

1. Portabilitas. MySQL dapat berjalan stabil pada berbagai sistem operasi seperti Windows, Linux, FreeBSD, Mac Os X Server, Solaris, Amiga, dan masih banyak lagi.
2. Multi-user. MySQL dapat digunakan oleh beberapa pengguna dalam waktu yang bersamaan tanpa mengalami masalah atau konflik.
3. Performance tuning’, MySQL memiliki kecepatan yang menakjubkan dalam menangani query sederhana, dengan kata lain dapat memproses lebih banyak SQL per satuan waktu.
4. Ragam tipe data. MySQL memiliki ragam tipe data yang sangat kaya, seperti signed / unsigned integer, float, double, char, text, date, timestamp, dan lain-lain.
5. Perintah dan Fungsi. MySQL memiliki operator dan fungsi secara penuh yang mendukung perintah Select dan Where dalam perintah (*query*).
6. Keamanan. MySQL memiliki beberapa lapisan keamanan seperti level subnetmask, nama host, dan izin akses *user* dengan sistem perizinan yang mendetail serta sandi terenkripsi.
7. Skalabilitas dan Pembatasan. MySQL mampu menangani basis data dalam skala besar, dengan jumlah rekaman (records) lebih dari 50 juta dan 60 ribu tabel serta 5 milyar baris. Selain itu batas indeks yang dapat ditampung mencapai 32 indeks pada tiap tabelnya.
8. Konektivitas. MySQL dapat melakukan koneksi dengan klien menggunakan protokol TCP/IP, Unix soket (UNIX), atau Named Pipes (NT).
9. Antar Muka. MySQL memiliki antar muka (interface) terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi API (Application Programming Interface).
10. Klien dan Peralatan. MySQL dilengkapi dengan berbagai peralatan (tool) yang dapat digunakan untuk administrasi basis data, dan pada setiap peralatan yang ada disertakan petunjuk online.
11. Struktur tabel. MySQL memiliki struktur tabel yang lebih fleksibel dalam menangani ALTER TABLE, dibandingkan basis data lainnya semacam PostgreSQL ataupun Oracle.
    * 1. **HTML (Hypertext Markup Language)**

HTML (*Hypertext Markup Language*) kode-kode tag yang mengintruksikan *browser* untuk menghasilkan tampilan sesuai dengan yang diinginkan.

Menurut Sibero (2018: 19) “*Hypertext Markup Language* HTML adalah bahasa yang digunakan Pada dokumen *web* sebagai bahasa untuk pertukaran dokumen *web”.* Sedangkan menurut Bekti (2015: 82) “HTML merupakan bahasa pemrograman yang bebas yang tidak dimliki oleh siapa pun, pengembangan dilakukan oleh banyak orang di berbagai negara dan biasanya dikatakan sebagai sebuah bahasa yang dikembangka Bersama-sama secara global”.

Dari pengertian diatas dapat disimpulkan bahwa HTML (*Hypertext Markup Language*) ialah bahasa pemrograman yang digunakan untuk menyusun atau mengatur tampilan konten dari halaman *web* pada saat berpindah ke *web* lain.

* + 1. **PHP (Hypertext Preprocessor) diberbanyak**

PHP (*Hypertext Preprocessor*)merupakan bahasa pemrograman *web* yang digunakan untuk membuat halaman *web* dinamis (*wikipedia*). PHP pertama kali dikembangkan pada tahun 1995 oleh *rasmus lerdorf*, namun sekarang di ambil oleh. oleh The PHP Group.

Pada awalnya PHP adalah singkatan dari *Personal Home Page,* namun dalam perkembanganya, diubah menjadi PHP (*Hypertext Preprocessor*), sebuah kepanjangan rekursif. Fungsi PHPDalam membuat halaman *web****,*** PHP sebenarnya bukanlah Bahasa yang wajib menggunakan. Sebuah *website* sederhana dapat dibuat tanpa menggunakan PHP sama sekali.

Untuk pembuatan *web*, kode PHP biasanya di sisipkan kedalam dokumen HTML. karena hal inilah PHP disebut juga sebagai *Scripting Language* atau bahasa pemograman script.

* + 1. **CSS (Cascading Style Sheet)**

*Cascading Style Sheet (CSS)* merupakan salah satu bahasa pemrograman web untuk mengendalikan beberapa komponen dalam sebuah web sehingga akan lebih terstruktur dan seragam (Kasiman Peranginangin 2016 : 15).

Sama halnya *styles* dalam aplikasi pengolahan kata seperti Microsoft Word yang dapat mengatur beberapa *style*, misalnya *heading*, *subbab*, *bodytext*, *footer*, *images*, dan *style* lainnya untuk dapat digunakan bersama-sama dalam beberapa berkas (*file*). Pada umumnya *CSS* dipakai untuk memformat tampilan halaman web yang dibuat dengan bahasa *HTML* dan *XHTML*.

*CSS* dapat mengendalikan ukuran gambar, warna bagian tubuh pada teks, warna tabel, ukuran *border*, warna *border*, warna *hyperlink*, warna *mouse over*, spasi antar paragraf, spasi antar teks, margin kiri, kanan, atas, bawah, dan parameter lainnya. *CSS* adalah bahasa style sheet yang digunakan untuk mengatur tampilan dokumen. Dengan adanya *CSS* memungkinkan kita untuk menampilkan halaman yang sama dengan format yang berbeda.

* + 1. **Java Script**

*Java Script* adalah bahasa skrip yang populer di internet dan dapat bekerja di sebagian besar penjelajah *web* populer seperti *Internet Explorer*, *Mozilla Firefox*, *Netscape* dan *Opera*. Kode *Java Script* dapat disisipkan dalam halaman *web* menggunakan *tag SCRIPT* (Abdul Kadir 2018:65).

*Java Script* pertama kali dikembangkan oleh Brendan Eich dari *Netscape* dibawah nama *Mocha*, yang nantinya namanya diganti menjadi *LiveScript, dan akhirnya menjadi JavaScript. Navigator* sebelumnya telah mendukung *Java* untuk lebih bisa dimanfaatkan para programmer yang non-*Java*. Maka dikembangkanlah bahasa pemrograman bernama *LiveScript* untuk mengakomodasi hal tersebut.

Bahasa pemrograman inilah yang akhirnya berkembang dan diberi nama *Java Script*, walaupun tidak ada hubungan bahasa antara *Java* dengan Java Script. *Java Script* bisa digunakan untuk banyak tujuan, misalnya untuk membuat efek *rollover* baik di gambar maupun teks, dan yang penting juga adalah untuk membuat *AJAX*. *Java Script* adalah bahasa yang digunakan untuk *AJAX*.

* 1. **Alat Bantu *Software***
     1. **Microsoft Visio**

Microsoft visio adalah sebuah program aplikasi komputer yang sering digunakan untuk membuat diagram, diagram alir (flowchart), brainstorm, dan skema jaringan yang dirilis oleh Microsoft Corporation. Aplikasi ini menggunakan grafik vektor untuk membuat diagram-diagramnya.

Visio aslinya bukanlah buatan Microsoft Corporation, melaikna buatan Visio Corporation, yang diakui oleh Microsoft pada tahun 2000. Versi yang telah menggunakan nama Microsoft Visio adalah Visio 2002, Visio 2003 dan Visio 2007 yang merupakan versi terbaru. Visio Standard dan Profesional menawarkan antar muka pengguna yang sama, tapi seri Profesional menawarkan lebih banyak pilihan template untuk pembuatan diagram yang lebih lanjut dan juga penataan letak (layout). Selain itu, edisi Profesional juga memudahkan pengguna untuk mengoneksikan diagram-diagram buatan mereka terhadap beberapa sumber data dan juga menampilkan informasi secara visual dengan menggunakan grafik.

* + 1. **XAMPP**

Nugroho (2017:1), “XAMPP adalah paket program *web* lengkap yang dapat Anda pakai untuk belajar pemrograman *web*, khususnya PHP dan MySQL, paket ini dapat di *download* secara gratis dan legal.”

Menurut Nugroho (2017:7), dibawah folder utama MYSQL, terdapat beberapa folder penting yang perlu diketahui. Penjelasan fungsinya sebagai berikut :

**Tabel 2.4. Folder-folder penting dalam MYSQL (Nugroho : 2017)**

|  |  |  |
| --- | --- | --- |
| **No.** | **Nama**  **Folder** | **Fungsi** |
| 1. | Apache | Folder utama dari *Apache Webserver*. |
| 2. | Htdocs | Folder utama untuk menyimpan data-data latihan *web*, baik PHP maupun HTML biasa. Pada folder ini, Semua folder dan file program di htdocs bisa diakses dengan mengetikkan alamat <http://localhost/> di *browser*. |
| 3. | Manual | Berisi subfolder yang di dalam terdapat manual program dan *database*, termasuk manual PHP dan MYSQL. |
| 4. | MySQL | Folder utama untuk *database* MYSQL *server*. Di Dalamnya terdapat subfolder data (lengkapnya: C:\MySQL\MySQL\data) untuk merekam semua nama *database*, serta subfolder bin yang berisi *tools klien* dan *server* MYSQL. |
| 5. | PHP | Folder utama untuk program PHP. |

**BAB III**

**ANALISIS DAN PERANCANGAN SISTEM**

* 1. **Analisis Sistem**

Sistem yang dirancang merupakan usulan perancangan sistem untuk memudahkan *user*. Dengan memulai pembangunan suatu program aplikasi, terlebih dahulu dilakukan perancangan pengembangan perangkat lunak berdasarkan pengumpulan data dan kebutuhan dari pengguna yang menggunakan sistem ini. Sehingga dapat memudahkan *user* untuk melakukan proses pengolahan data penduduk di Desa Kalimanggis dengan lebih efektif dan efisien.

* + 1. **Analisis Objek Penelitian**

Sebelum dilakukan penelitian, langkah awal yang harus dilakukan oleh peneliti adalah mengetahui dan menentukan terlebih dahulu metode yang digunakan dalam penelitian.

Objek penelitian yang penulis amati adalah penanganan pengolahan data penduduk, serta pembuatan laporan yang sedang atau telah berlangsung selama ini.

* + 1. **Analisis Prosedur Sistem Berjalan**

Informasi tentang sistem yang sedang berjalan sangat penting sekali dalam proses perancangan sistem informasi. Selama ini proses kegiatan pendataan yang di lakukan masih dengan cara manual yaitu dengan pendataan menggunakan media kertas yang nantinya akan dimasukan ke *Microsoft Excel* sehingga informasinya kurang akurat, sistem yang berjalan selama ini belum bisa mengakomodir semua data-data tersebut dan laporannya belum sesuai dengan apa yang diharapkan.

Dalam setiap pelaksanaan, sistem komputerisasi sangat diperlukan untuk menunjang kegiatan yang akan dilaksanakan sehingga pendataan tersebut dilakukan lebih cepat dan akurat.

Tahap analisis dilakukan setelah tahap perencanaan sistem dan sebelum tahap desain sistem. Tahap ini merupakan tahap yang kritis karena kesalahan dalam tahap ini menyebabkan kesalahan pada tahap selanjutnya.

Berdasarkan hasil analisis kegiatan yang dilakukan penulis untuk Desa Kalimanggis yaitu melakukan pencatatan data penduduk, kematian, kelahiran, kepindahan dan kedatangan penduduk, dan data pendidikan diinput secara komputerisasi, sehingga setiap pekerjaan dapat diselesaikan dengan efektif dan efisien.

* + 1. **Analisis Kebutuhan Fungsional**

Analisis kebutuhan fungsional pada sistem ini adalah :

1. Fungsi *Login* / akses Admin

Login merupakan pintu masuk aplikasi ini. Proses login adalah melakukan validasi dengan membandingkan nama login dan password yang dimasukan dengan data yang ada dalam basis data.

1. Fungsi Entri Data

Pada fungsi ini, sistem dapat melakukan input, edit, dan *delete* pada data komputer.

1. Fungsi Cetak Laporan

Laporan hasil penilaian komputer berdasarkan dari kriteria-kriteria yang telah ditentukan sebelumnya.

1. Fungsi Pencarian

Pada fungsi ini, sistem dapat melakukan pencarian data yang kita inginkan sesuai dengan *keyword* yang dimasukan.

* + 1. **Analisis Kebutuhan Perangkat Lunak**

Perangkat lunak (*Software*) ini dapat berjalan baik, apabila mempunyai

standar minimal dari perangkat keras (*Hardware*). Perangkat lunak yang digunakan minimal memiliki spesifikasi sebagai berikut :

*1.* Sistem operasi *Windows 7 Ultimate 64 Bit.*

*2. Sublime Text* sebagai alat bantu pembuatan aplikasi (*coding*).

*3. MySql* sebagai alat bantu untuk pembuatan *database.*

*4. Google Chrome* sebagai alat untuk pengetesan aplikasi yang telah dibuat.

*5. Xampp* sebagai alat bantu untuk pengoneksian antara Apache dan MYSQL.

* + 1. **Analisis Kebutuhan Perangkat Keras**

Perangkat keras (*Hardware*) ini dapat berjalan baik, apabila mempunyai standar minimal. Spesifikasi standar minimal perangkat keras (*Hardware*) yang digunakan saat implementasi adalah sebagai berikut:

* 1. Processor Intel Dual Core 1,80 GHz
  2. RAM 2 GB
  3. Kapasitas Harddisk yang dipakai 256 MB
  4. **Perancangan Sistem**
     1. **Tujuan Perancangan Sistem**

Tujuan perancangan sistem ini adalah untuk menghasilkan perancangan sistem pengolahan data penduduk yang terkomputerisasi memudahkan mengolah data-data untuk input data penduduk. Dalam tahapan perancangan ini, bertujuan menghasilkan “Sistem informasi Pengolahan Data Penduduk”. Adapun komponen komponen sistem pendukung keputusan ini meliputi :

* 1. Perancangan alur prosedur (Flowmap)
  2. Perancangan *context* *diagram* (DCD) dan data *flow* *diagram* (DFD)
  3. Perancangan basis data (Tabel relasi, Struktur tabel)
  4. Perancangan program (Perancangan Antarmuka)
  5. **Prosedur Yang Akan Di Implementasi**

Berdasarkan analisis masalah, dalam sistem yang sedang dibahas terdapat beberapa macam prosedur, prosedur-prosedur tersebut terbentuk dari beberapa proses dalam sistem yang ada dilokasi penelitian semuanya masih dikerjakan manual. Adapun prosedur-prosedur yang akan dirancang, dikerjakan dan di implementasikan dengan sistem terkomputerisasi adalah sebagai berikut :

1. Pengolahan Data Penduduk dan Laporan Data Penduduk

Pengolahan data tersebut adalah proses penginputan data Penduduk yang biasanya dilakukan secara manual nantinya dilakukan dengan metode komputerisasi. Dan diharapkan dapat meminimalisir kesalahan dan dapat mempermudah proses pencarian. Prosedur untuk melakukan proses pengolahan data penduduk yaitu penduduk mulai mengisi formulir data penduduk yang telah disediakan di bagian RT. Lalu setelah pengisian formulir diberikan ke operator desa untuk diinputkan. Setelah pengisian formulir lengkap, operator desa menginput data penduduk dan disimpan ke dalam *database* dan dibuatkan laporan data penduduk yang nantinya bisa diakses oleh operator, ketua Rt dan Rw.

1. Pengolahan Data Kelahiran dan Laporan Data Kelahiran

Pengolahan data tersebut adalah proses penginputan data Kelahiran yang biasanya dilakukan secara manual nantinya dilakukan dengan metode komputerisasi. Dan diharapkan dapat meminimalisir kesalahan dan dapat mempermudah proses pencarian. Prosedur untuk melakukan proses pengolahan data kelahiran yaitu penduduk mulai mengisi formulir data kelahiran yang telah disediakan di bagian RT. Lalu setelah pengisian formulir diberikan ke operator desa untuk diinputkan. Setelah pengisian formulir lengkap, operator desa menginput data kelahiran dan disimpan ke dalam *database* dan dibuatkan laporan data kelahiran yang nantinya bisa diakses oleh operator, ketua Rt, ketua Rw, dan penduduk yang nantinya bisa menjadi surat penghantar pembuatan akta kelahiran.

1. Pengolahan Data Kematian dan Laporan Data Kematian

Pengolahan data tersebut adalah proses penginputan data Kematian yang biasanya dilakukan secara manual nantinya dilakukan dengan metode komputerisasi. Dan diharapkan dapat meminimalisir kesalahan dan dapat mempermudah proses pencarian. Prosedur untuk melakukan proses pengolahan data kematian yaitu penduduk mulai mengisi formulir data kematian yang telah disediakan di bagian RT. Lalu setelah pengisian formulir diberikan ke operator desa untuk diinputkan. Setelah pengisian formulir lengkap, operator desa menginput data kematian dan disimpan ke dalam *database* dan dibuatkan laporan data kematian yang nantinya bisa diakses oleh operator, ketua Rt, ketua Rw, dan penduduk yang nantinya bisa menjadi surat penghantar pembuatan surat kematian.

1. Pengolahan Data Pindah dan Laporan Data Pindah

Pengolahan data tersebut adalah proses penginputan data Pindah yang biasanya dilakukan secara manual nantinya dilakukan dengan metode komputerisasi. Dan diharapkan dapat meminimalisir kesalahan dan dapat mempermudah proses pencarian. Prosedur untuk melakukan proses pengolahan data pindah yaitu penduduk mulai mengisi formulir data pindah yang telah disediakan di bagian RT. Lalu setelah pengisian formulir diberikan ke operator desa untuk diinputkan. Setelah pengisian formulir lengkap, operator desa menginput data pindah dan disimpan ke dalam *database* dan dibuatkan laporan data pindah yang nantinya bisa diakses oleh operator, ketua Rt, ketua Rw, dan pindah yang nantinya bisa menjadi surat penghantar pembuatan surat pindah.

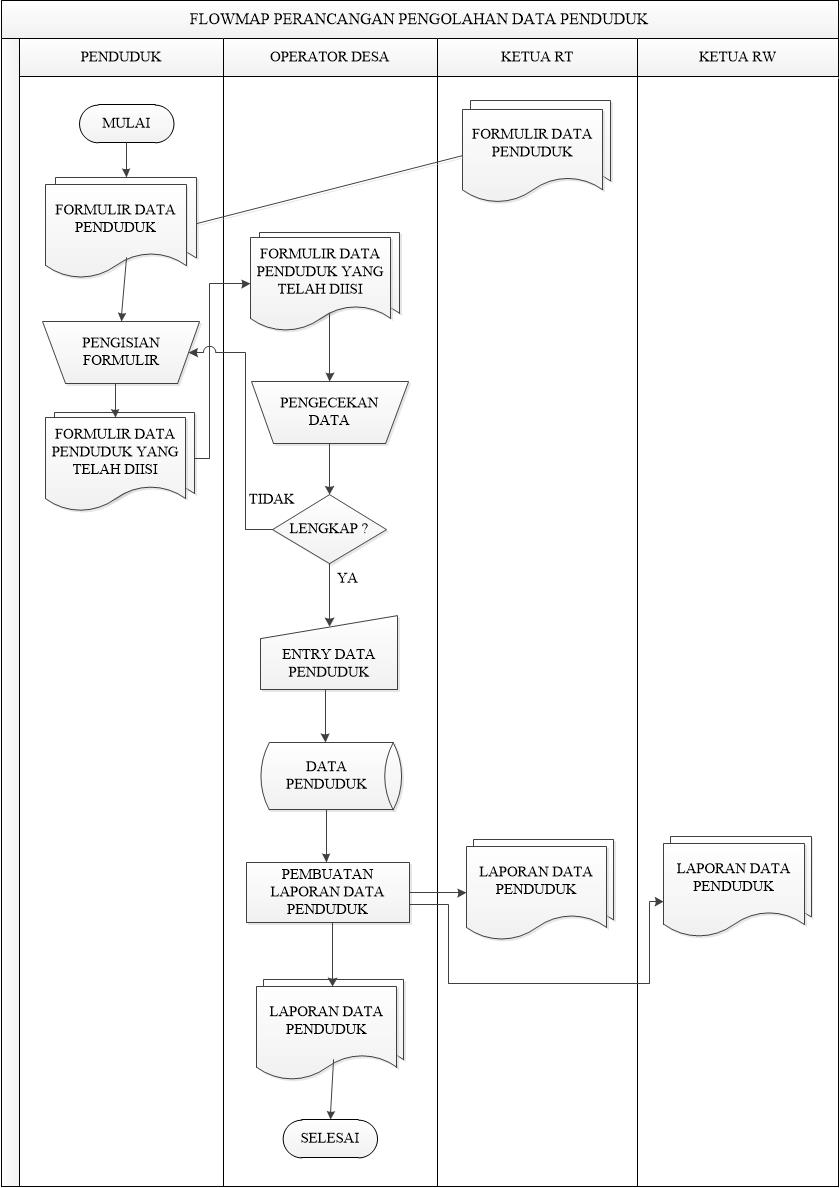
1. Pengolahan Data Masuk dan Laporan Data Masuk (Pindahan)

Pengolahan data tersebut adalah proses penginputan data Masuk yang biasanya dilakukan secara manual nantinya dilakukan dengan metode komputerisasi. Dan diharapkan dapat meminimalisir kesalahan dan dapat mempermudah proses pencarian. Prosedur untuk melakukan proses pengolahan data masuk (pindahan) yaitu penduduk mulai mengisi formulir data masuk (pindahan) yang telah disediakan di bagian RT. Lalu setelah pengisian formulir diberikan ke operator desa untuk diinputkan. Setelah pengisian formulir lengkap, operator desa menginput data masuk (pindahan) dan disimpan ke dalam *database* dan dibuatkan laporan data masuk (pindahan) yang nantinya bisa diakses oleh operator, ketua Rt, ketua Rw, dan penduduk yang nantinya bisa menjadi surat keterangan pindahan.

* + 1. ***Flowmap***

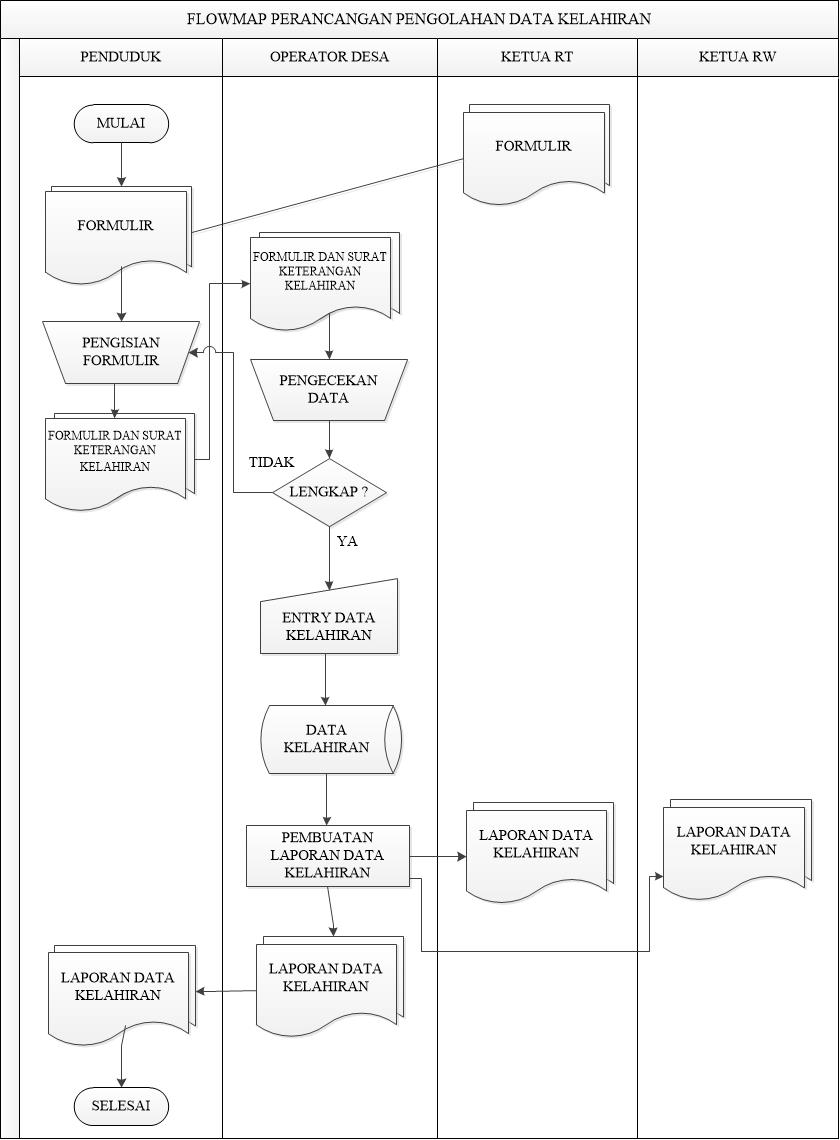
*Flow map* merupakan suatu diagram untuk meggambarkan aliran data atau informasi antar bagian-bagian yang terkait dalam sistem. Informasi yang disajikan dengan penggambaran *flow map* ini lebih menekankan pada urutan aktivitas disetiap entitas yang berada dalam sistem desa pasirpanjang. Entitas yang terdapat pada sistem lama ini terdiri dari penduduk, petugas desa dan kecamatan yang masing-masing entitas tersebut memiliki peranan yang sangat penting di dalam sistem tersebut.

* + - 1. ***Flowmap* Sistem Informasi Pengolahan Data Penduduk**



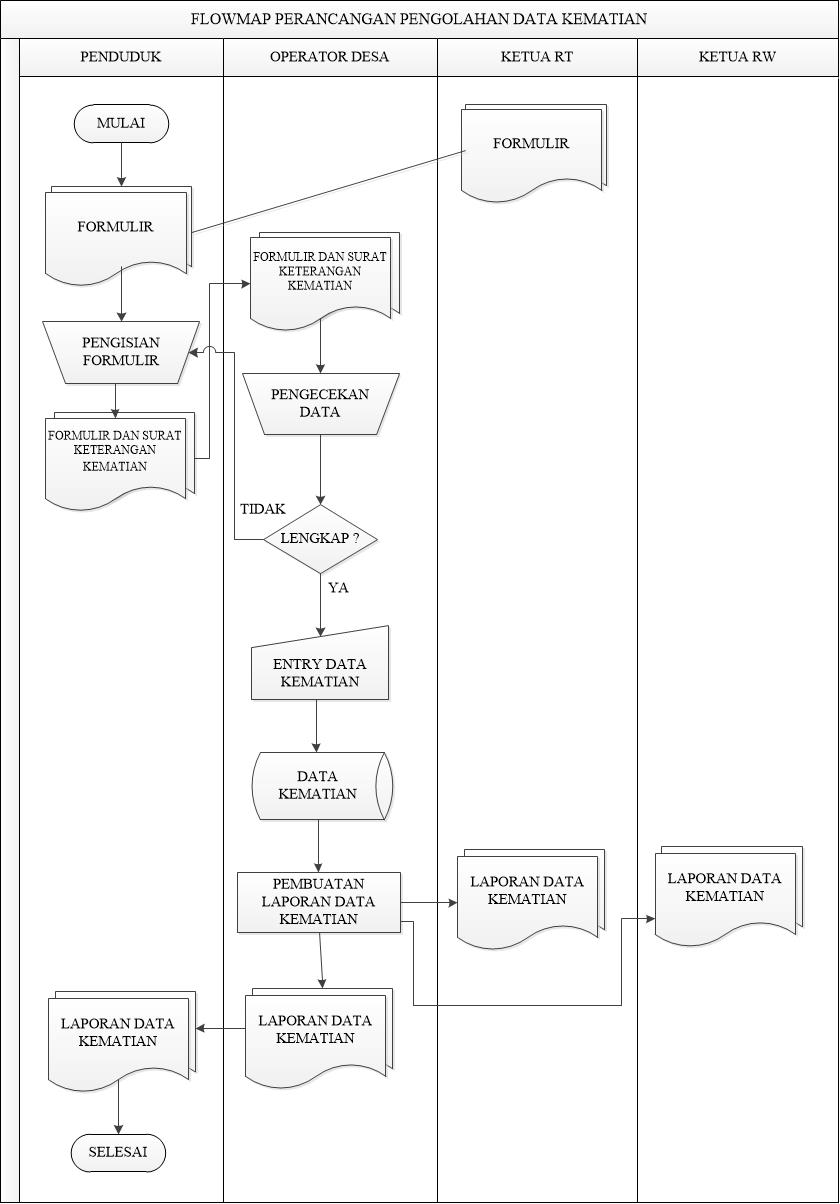
Gambar 3.1. Flowmap Data Penduduk

* + - 1. ***Flowmap* Sistem Informasi Pengolahan Data Kelahiran**

****

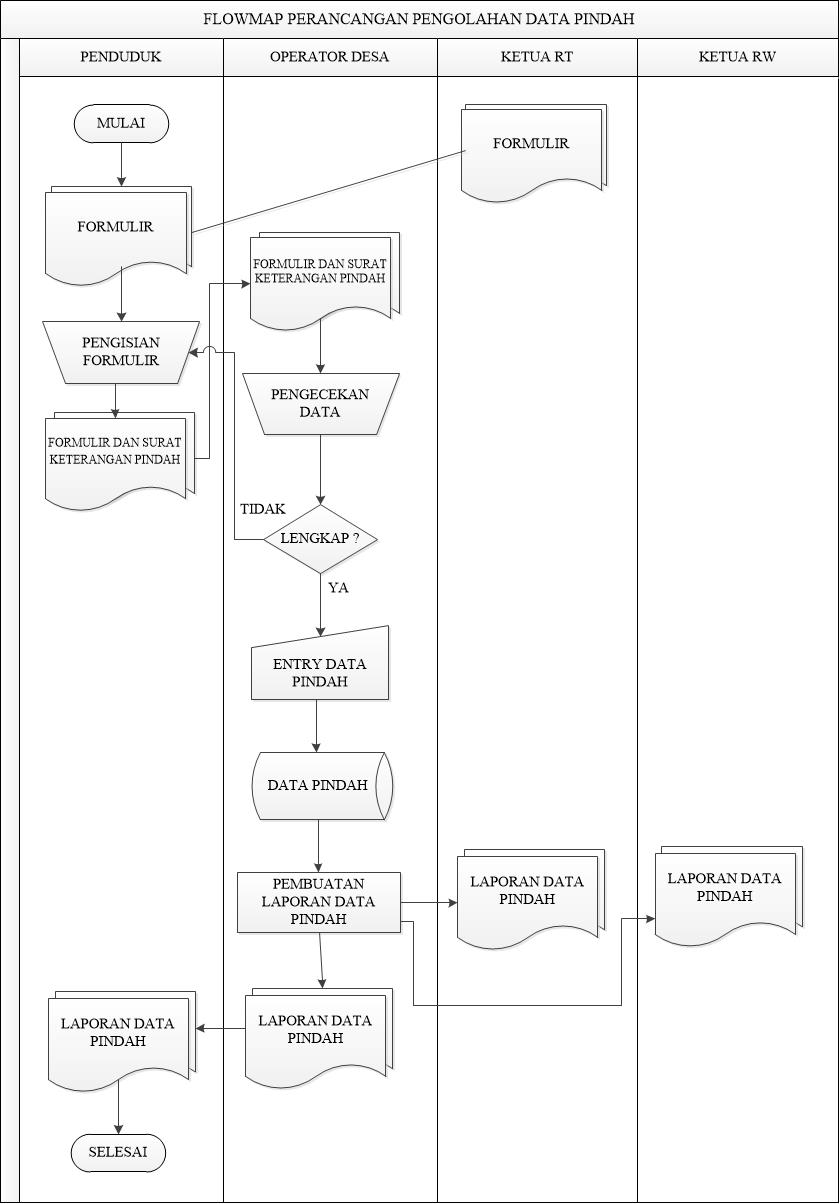
Gambar 3.2. Flowmap Data Kelahiran

* + - 1. ***/Flowmap* Sistem Informasi Pengolahan Data Kematian**



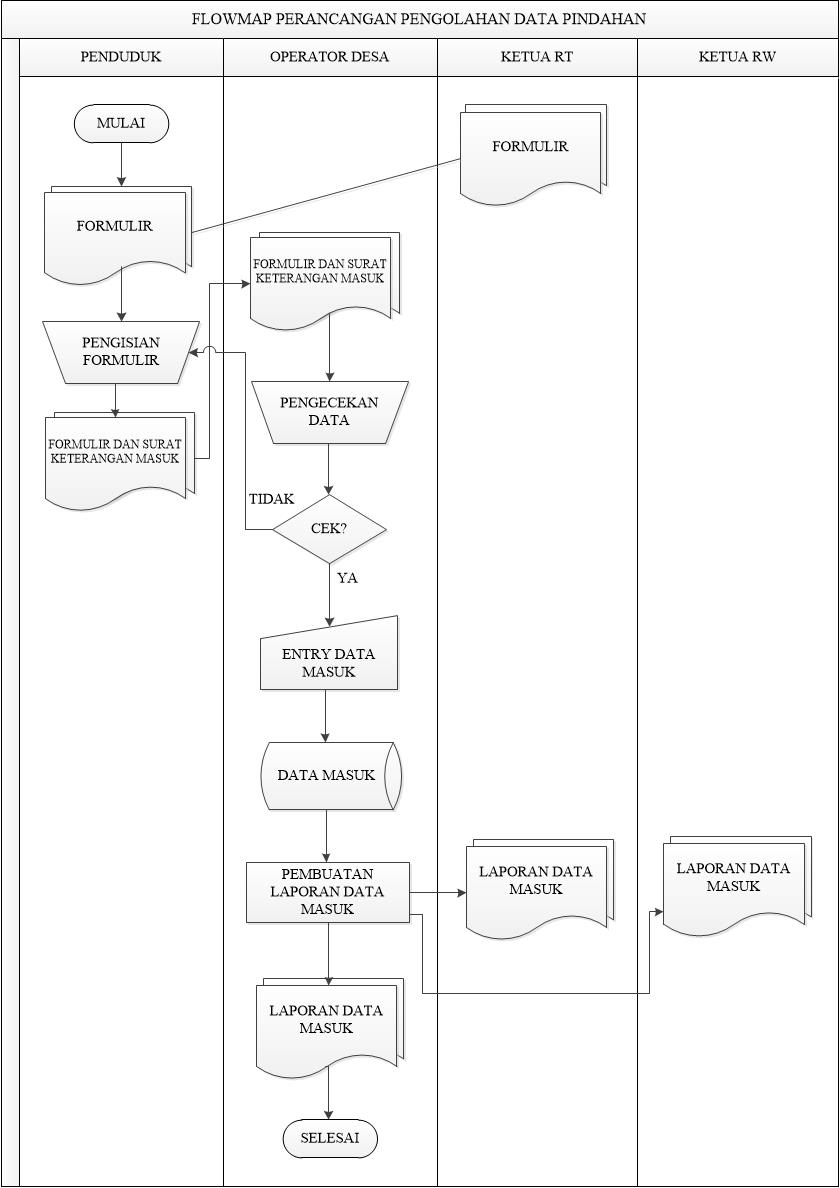
Gambar 3.3. Flowmap Data Kematian

* + - 1. ***Flowmap* Sistem Informasi Pengolahan Data Pindah**



Gambar 3.4. Flowmap Data Pindah

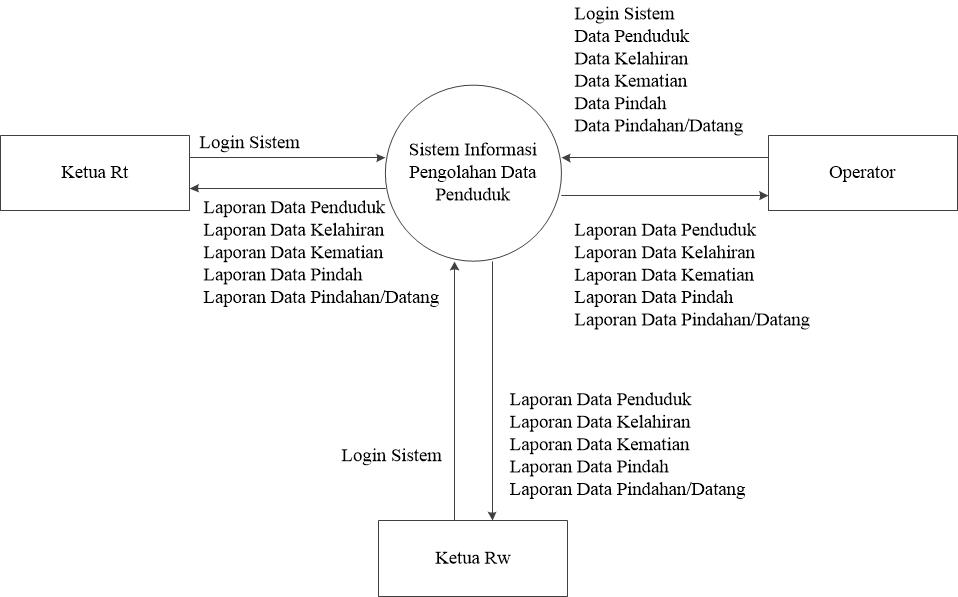
* + - 1. ***Flowmap* Sistem Informasi Pengolahan Data Pindahan**



Gambar 3.5. Flowmap Pindahan

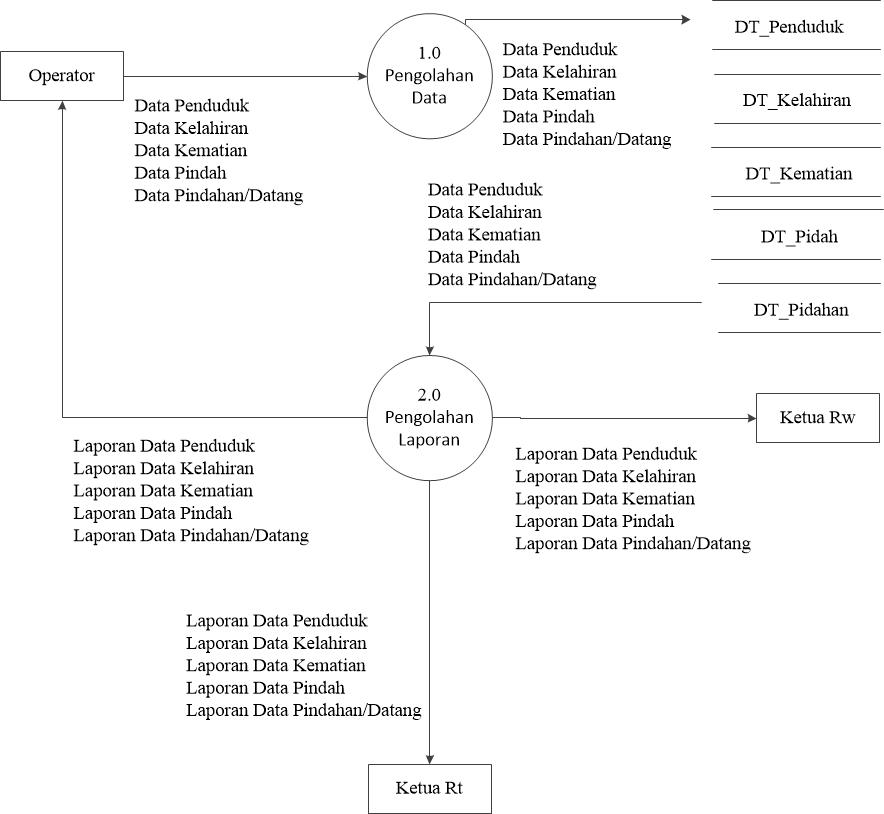
* 1. **Diagram Konteks**

Untuk membuat diagram konteks, penulis terlebih dahulu akan menganalisa sistem informasi dibuat yang akan menghasilkan informasi apa saja serta membutuhkan data apa saja, dan selanjutnya menentukan sumber data apa saja yang dibutuhkan sistem dan tujuan informasi yang dihasilkan sistem. Maka dari analisa pada bab sebelumnya, menurut penulis diperoleh diagram konteks sebagai berikut :

****

Gambar 3.6 Diagram Konteks

* + 1. **DFD Level 0**



Gambar 3.7 DFD Level 0

**Keterangan Proses :**

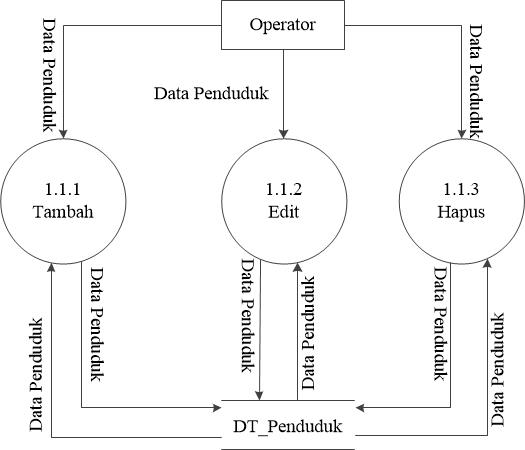
1. Pengolahan data merupakan pengolahan yang terdiri data penduduk, data kelahiran, data kematian, data pindah, data kelahiran, data kematian, data pindah dan data pindahan/datang.
2. Pembuatan laporan merupakan proses pembuatan untuk menghasilkan laporan data penduduk, laporan data kelahiran, laporan data kematian, laporan data pidah dan laporan data pindahan/data.
   * 1. **DFD Level 1**



Gambar 3.8 DFD Level 1

**Keterangan Proses :**

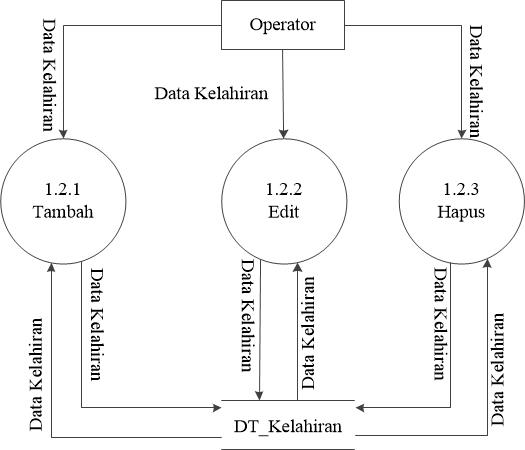
1. Operator menambahkan data penduduk ke dalam proses pengolahan data penduduk. Lalu operator dapat melihat data penduduk dari proses data penduduk dan disimpan ke dalam *storage* DT\_Penduduk.
2. Operator menambahkan data kelahiran ke dalam proses pengolahan data kelahiran. Lalu operator dapat melihat data kelahiran dari proses data kelahiran dan disimpan ke dalam *storage* DT\_Kelahiran.
3. Operator menambahkan data kematian ke dalam proses pengolahan data kematian. Lalu operator dapat melihat data kematian dari proses data kematian dan disimpan ke dalam *storage* DT\_Kematian.
4. Operator menambahkan data pindah ke dalam proses pengolahan data pindah. Lalu operator dapat melihat data pindah dari proses data pindah dan disimpan ke dalam *storage* DT\_Pindah.
5. Operator menambahkan data pindahan/datang ke dalam proses pengolahan data pindahan/datang. Lalu operator dapat melihat data pindahan/datang dari proses data pindahan/datang dan disimpan ke dalam *storage* DT\_Pindahan.
   * 1. **DFD Level 1 Proses 1**



Gambar 3.9 DFD Level 1 Proses 1

**Keterangan Proses :**

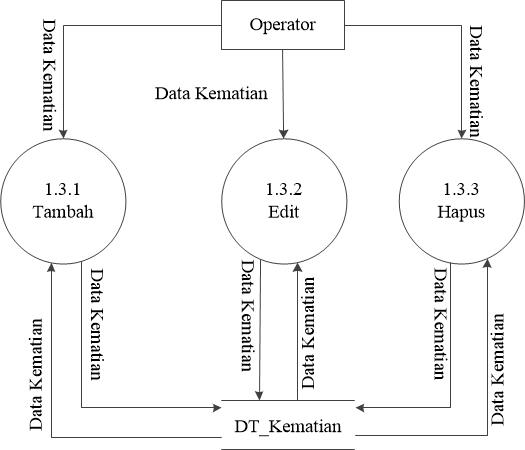
1. Proses 1.1.1 merupakan proses tambah data penduduk yang kemudian disimpan pada tabel data penduduk.
2. Proses 1.1.2 merupakan proses edit data penduduk yang kemudian disimpan di data penduduk.
3. Proses 1.1.3 merupakan proses hapus data penduduk pada tabel data penduduk.
   * 1. **DFD Level 1 Proses 2**

****

Gambar 3.10 DFD Level 1 Proses 4

**Keterangan Proses :**

1. Proses 1.2.1 merupakan proses tambah data kelahiran yang kemudian disimpan pada tabel data kelahiran.
2. Proses 1.2.2 merupakan proses edit data kelahiran yang kemudian disimpan di data kelahiran.
3. Proses 1.2.3 merupakan proses hapus data kelahiran pada tabel data kelahiran.
   * 1. **DFD Level 1 Proses 3**

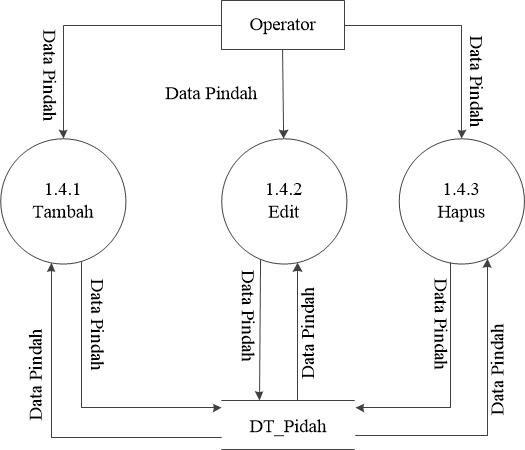
****

Gambar 3.11 DFD Level 1 Proses 4

**Keterangan Proses :**

1. Proses 1.3.1 merupakan proses tambah data kematian yang kemudian disimpan pada tabel data kematian.
2. Proses 1.3.2 merupakan proses edit data kematian yang kemudian disimpan di data kematian.
3. Proses 1.3.3 merupakan proses hapus data kematian pada tabel data kematian.

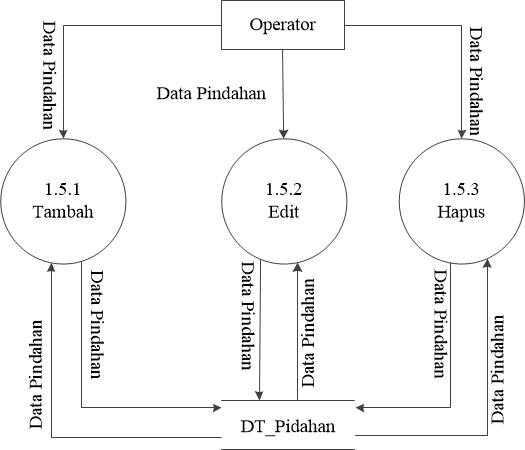
**3.4.6 DFD Level 1 Proses 4**



Gambar 3.12 DFD Level 1 Proses 4

**Keterangan Proses :**

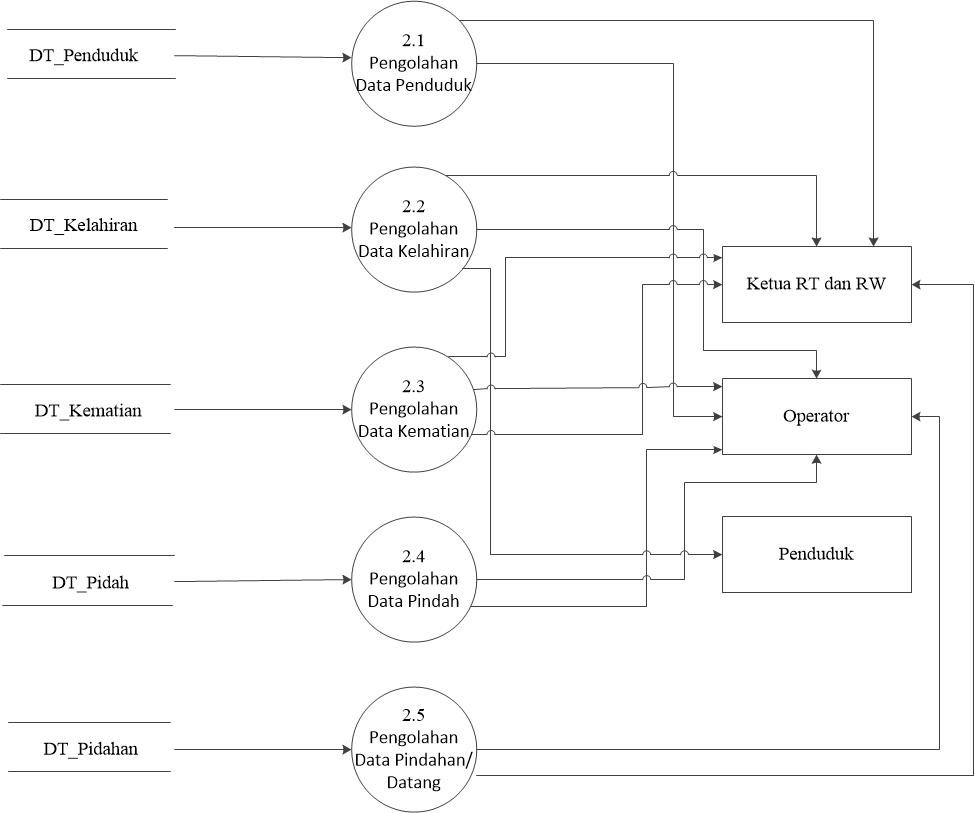
* + - 1. Proses 1.4.1 merupakan proses tambah data pindah yang kemudian disimpan pada tabel data pindah.
      2. Proses 1.4.2 merupakan proses edit data pindah yang kemudian disimpan di data pindah.
      3. Proses 1.4.3 merupakan proses hapus data pindah pada tabel data pindah.
    1. **DFD Level 1 Proses 5**



Gambar 3.13 DFD Level 1 Proses 5

**Keterangan Proses :**

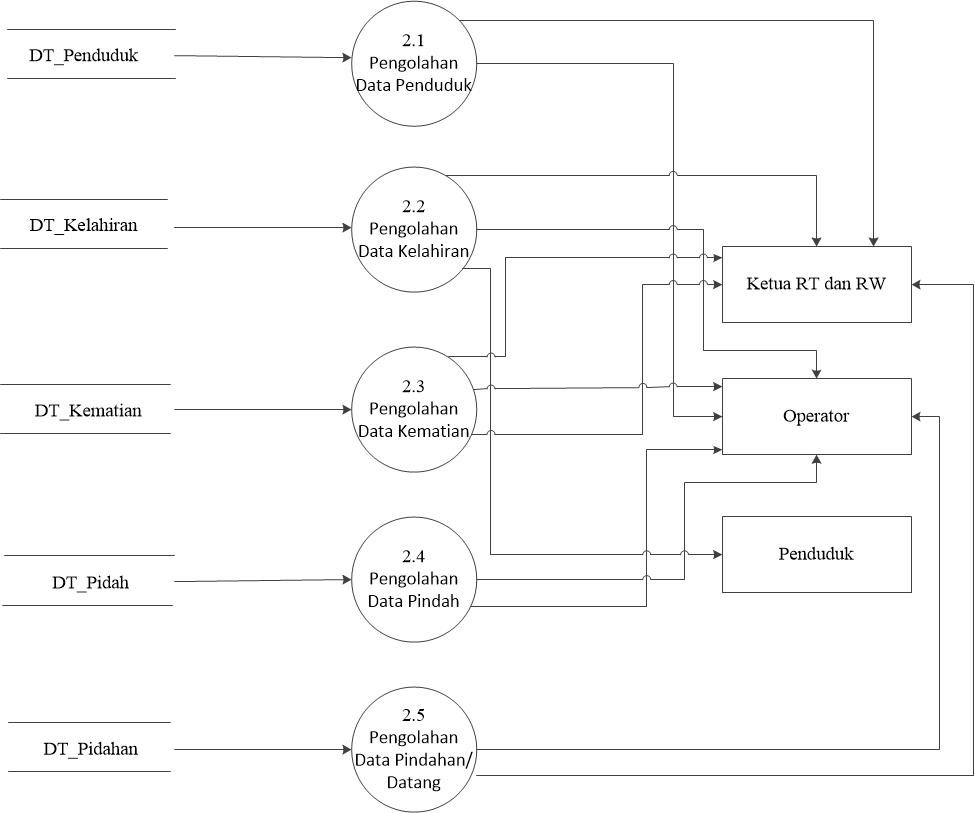
1. Proses 1.5.1 merupakan pross tambah data pindahan yang kemudian disimpan pada tabel data pindahan.
2. Proses 1.5.2 merupakan proses edit data pindahan yang kemudian disimpan di data pindahan.
3. Proses 1.5.3 merupakan proses hapus data pindahan pada tabel data pindahan.
   * 1. **DFD Level 2 Pengolahan Laporan Untuk Operator**



Gambar 3.14 DFD Level 2 Pengolahan Laporan Untuk Operator

**Keterangan Proses :**

1. Simpan data penduduk mengalirkan data penduduk kemudian 2.1 pengolahan pembuatan data penduduk diproses dan menghasilkan laporan data penduduk kemudian diberikan ke ketua RW dan Operator.
2. Simpan data kelahiran mengalirkan 2 data kelahiran kemudian 2.3 pengolahan pembuatan data kelahiran diproses dan menghasilkan laporan data kelahiran kemudian diberikan ke ketua RW, lalu 2.4 pengolahan pembuatan surat pengantar kelahiran diproses dan menghasilkan surat pengantar kelahiran kemudian diberikan kependuduk.
3. Simpanan data kematian mengalirkan data kematian kemudian 2.5 pengolahan pembuatan data kematian diproses dan menghasilkan laporan data kematian kemudian diberikan ke ketua RW.
4. Simpanan data pindah mengalirkan data pindah kemudian 2.6 pengolahan pembuatan data pindah diproses dan menghasilkan laporan data pindah kemudian diberikan ke ketua RW.
5. Simpanan data pindahan mengalirkan data pindahan kemudian 2.7 pengolahan pembuatan data pindahan diproses dan menghasilkan laporan data pindahan diproses dan menghasilkan laporan data pindahan kemudian diberikan ke ketua RW.
   * 1. **DFD Level 2 Pengolahan Laporan Untuk RW**



Gambar 3.15 DFD Level 2 Pengolahan Laporan Untuk RW

**Keterangan Proses :**

Simpan data penduduk mengalirkan data penduduk kemudian 2.1 pengolahan pembuatan laporan data penduduk diproses dan menghasilkan laporan data penduduk kemudian diberikan ke ketua RW dan Operator.

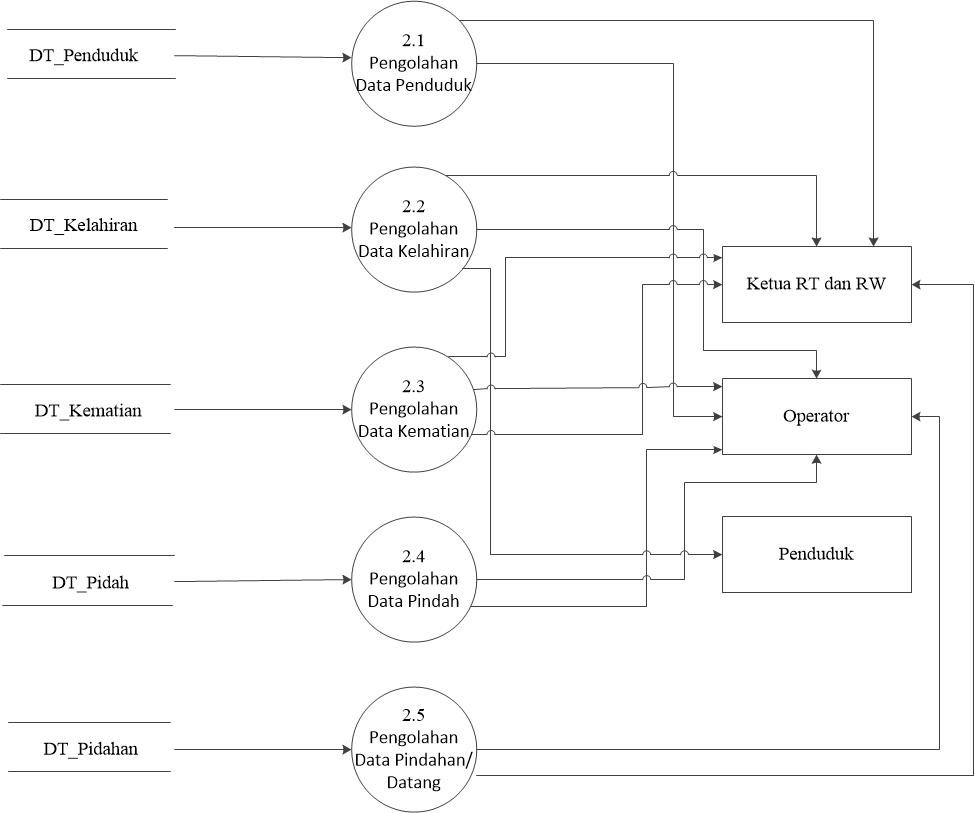
Simpan data kelahiran mengalirkan 2 data kelahiran kemudian 2.3 pengolahan pembuatan laporan data kelahiran diproses dan menghasilkan laporan data kelahiran kemudian diberikan ke ketua RW, lalu 2.4 pengolahan pembuatan surat pengantar kelahiran diproses dan menghasilkan surat pengantar kelahiran kemudian diberikan kependuduk.

Simpanan data kematian mengalirkan data kematian kemudian 2.5 pengolahan pembuatan laporan data kematian diproses dan menghasilkan laporan data kematian kemudian diberikan ke ketua RW.

Simpanan data pindah mengalirkan data pindah kemudian 2.6 pengolahan pembuatan laporan data pindah diproses dan menghasilkan laporan data pindah kemudian diberikan ke ketua RW.

Simpanan data pindahan mengalirkan data pindahan kemudian 2.7 pengolahan pembuatan laporan data pindahan diproses dan menghasilkan laporan data pindahan diproses dan menghasilkan laporan data pindahan kemudian diberikan ke ketua RW.

* + 1. **DFD Level 2 Pengolahan Laporan Untuk RT**



Gambar 3.16 DFD Level 2 Pengolahan Laporan Untuk RT

**Keterangan Proses :**

1. Simpan data penduduk mengalirkan data penduduk kemudian 2.1 pengolahan pembuatan laporan data penduduk diproses dan menghasilkan laporan data penduduk kemudian diberikan ke ketua RW dan Operator.
2. Simpan data kelahiran mengalirkan 2 data kelahiran kemudian 2.3 pengolahan pembuatan laporan data kelahiran diproses dan menghasilkan laporan data kelahiran kemudian diberikan ke ketua RW, lalu 2.4 pengolahan pembuatan surat pengantar kelahiran diproses dan menghasilkan surat pengantar kelahiran kemudian diberikan ke penduduk.
3. Simpanan data kematian mengalirkan data kematian kemudian 2.5 pengolahan pembuatan laporan data kematian diproses dan menghasilkan laporan data kematian kemudian diberikan ke ketua RW.
4. Simpanan data pindah mengalirkan data pindah kemudian 2.6 pengolahan pembuatan laporan data pindah diproses dan menghasilkan laporan data pindah kemudian diberikan ke ketua RW.
5. Simpanan data pindahan mengalirkan data pindahan kemudian 2.7 pengolahan pembuatan laporan data pindahan diproses dan menghasilkan laporan data pindahan diproses dan menghasilkan laporan data pindahan kemudian diberikan ke ketua RW.
   1. **Perancangan *Entity Relationalship Diagram* (ERD)**

Diagram hubungan entitas atau yang lebih dikenal dengan sebutan *Entity Relationship Diagram (ERD)* merupakan suatu modal jaringan yang menggambarkan rancangan atau susunan *data store* dari sistem pada level pemisah yang tinggi. Jadi didalamnya terdapat informasi apa saja yang terkandung di dalam *data store* dan juga hubungan apa yang ada diantara *data store.*

Rancangan dari diagram hubungan entitas dari Sistem Informasi yang diusulkan dan akan diimplementasikan adalah sebagai berikut:



Gambar 3.17 *Entity* *Relationship Diagram* (ERD)

**Kamus Data:**

|  |  |  |
| --- | --- | --- |
| Penduduk | : | (#nik, no\_kk, nama, tempat\_lahir, tanggal\_lahir, jenis\_kelamin, gol\_darah, alamat, rt, rw, kelurahan, kecamatan, kota, provinsi, agama, status\_perkawinan, pekerjaan, kewarganegaraan, pendidikan, nama\_ayah, nama\_ibu, id\_user, status, posisi). |
| Kematian | : | (#id\_kematian, no\_surat, nik, tgl\_kematian, tempat\_kematian, sebab\_kematian, yang\_menentukan). |
| Masuk | : | (#id\_masuk, no\_surat, nik, alasan\_masuk, alamat\_tujuan, jenis\_masuk). |
| Pindah | : | (#id\_pindahan, no\_surat, nik, alasan\_pindah, alamat\_pindah, jenis\_pindah). |
| Kelahiran | : | (#id\_kelahiran, no\_surat, nik, tgl\_lahir, kelahiran\_ke). |

* + 1. **Perancangan Basis Data**

Rancangan struktur table dalam basis data merupakan spesifikasi dari *file-file* yang digunakan didalam sistem, diusulkan dan dijabarkan dalam bentuk tabel. Dalam tabel ini terdapat nama tabel dan kunci *indeks* yang digunakan serta kolom (*field*) berikut baris (*record*).

Setiap *field* dari suatu tabel basis data memiliki tipe dan panjang karakter / angka serta panjang decimal yang dapat ditentukan sesuai dengan kebutuhan. Panjang decimal merupakan banyaknya angka pecahan yang hanya ada dalam tipe *field* *numeric* / angka, tipe-tipe lain yang umum dipakai dalam tabel basis data diantaranya:

Rancangan file - file atau database merupakan spesifikasi dari file – file yang digunakan didalam sistem yang disolusikan dan dijabarkan dalam bentuk tabel, yang di dalamnya terdapat file, organisasi dan index key yang digunakan serta record – record yang menunjang isi suatu file atau database. Rancangan dari tabel yang diusulkan penulis adalah sebagai berikut :

* + - 1. Nama Tabel : Mst\_penduduk

Primary Key : #nik

**Tabel 3.1 Rancangan Tabel Data Penduduk**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Nama Field** | **Type** | **Size** | **Keterangan** |
| 1. | Nik | Varchar | 16 | Nomor Induk Kependudukan |
| 2. | No\_kk | Varchar | 16 | Nomor Kartu Keluarga |
| 3. | Nama | Varchar | 30 | Nama Lengkap |
| 4. | Tempat\_lahir | Varchar | 25 | Tempat Lahir Penduduk |
| 5. | Tanggal\_lahir | Date | - | Tanggal Lahir |
| 6. | Jenis\_kelamin | Varchar | 9 | Jenis Kelamin |
| 7. | Gol\_darah | Varchar | 2 | Golongan Darah Penduduk |
| 8. | Alamat | Text | - | Alamat Lengkap sesuai KTP |
| 9. | Rt | Int | 5 | RT Penduduk Sesuai KTP |
| 10. | Rw | Int | 5 | RW Penduduk Sesuai KTP |
| 11. | Kelurahan | Char | 20 | Nama Kelurahan |
| 12. | Kecamatan | Char | 20 | Nama Kecamatan |
| 13. | Kota | Char | 20 | Nama Kabupaten/Kota |
| 14. | Provinsi | Char | 20 | Nama Provinsi |
| 15. | Agama | Varchar | 20 | Nama Agama |
| 16. | Status\_perkawinan | Varchar | 20 | Status Perkawinan Penduduk |
| 17. | Pekerjaan | Varchar | 50 | Pekerjaan Penduduk |
| 18. | Kewarganegaraan | Varchar | 50 | Kebangsaan Penduduk |
| 19. | Pendidikan | Varchar | 50 | Pendidikan Terakhir |
| 20. | Nama\_ayah | Varchar | 30 | Nama Ayah Kandung |
| 21. | Nama\_ibu | Varchar | 30 | Nama Ibu Kandung |
| 22. | Id\_user | Int | 10 | Kode User |
| 23. | Status | Varchar | 10 | Status di Lingkungan |
| 24. | Posisi | Varchar | 50 | Posisi di Keluarga |

* + - 1. Nama Tabel : Mst\_kematian

Primary Key : #id\_kematian

**Tabel 3.2 Rancangan Tabel Data Kematian**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Nama Field** | **Type** | **Size** | **Keterangan** |
| 1. | Id\_kematian | Int | 10 | Kode Kematian |
| 2. | No\_surat | Varchar | 50 | No Surat Kematian |
| 3. | Nik | Varchar | 16 | Nomor Induk Kependudukan |
| 4. | Tgl\_kematian | Date | - | Tanggal Kematian |
| 5. | Tempat\_kematian | Varchar | 50 | Tempat Kematian |
| 6. | Sebab\_kematian | Varchar | 50 | Penyebab Kematian |
| 7. | Yang\_menentukan | Varchar | 50 | Yang memponis Kematian |

* + - 1. Nama Tabel : Mst\_masuk

Primary Key : #Id\_masuk

**Tabel 3.3 Rancangan Tabel Data Masuk**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Nama Field** | **Type** | **Size** | **Keterangan** |
| 1. | Id\_masuk | Int | 10 | Kode Masuk |
| 2. | No\_surat | Varchar | 50 | No Surat Pindah |
| 3. | Nik | Varchar | 16 | Nomor Induk Kependudukan |
| 4. | Alasan\_masuk | Varchar | 50 | Alasan Pindah |
| 5. | Alamat\_tujuan | Varchar | 50 | Alamat Tujuan Pindah |
| 6. | Jenis\_masuk | Varchar | 50 | Jenis Pindahan |

* + - 1. Nama Tabel : Mst\_pindah

Primary Key : #id\_pindahan

**Tabel 3.4 Rancangan Tabel Data Pindah**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Nama Field** | **Type** | **Size** | **Keterangan** |
| 1. | Id\_pindahan | Int | 10 | Kode Pindahan |
| 2. | No\_surat | Varchar | 20 | Nomor Surat |
| 3. | Nik | Varchar | 16 | Nomor Induk Kependudukan |
| 4. | Alasan\_pindah | Varchar | 50 | Alasan Pindah |
| 5. | Alamat\_tujuan | Varchar | 50 | Alatat Tujuan |
| 6. | Jenis\_pindahan | Varchar | 50 | Jenis Pindah |

6. Nama Tabel : Mst\_kelahiran

Primary Key : #Id

**Tabel 3.5 Rancangan Tabel Data Kelahiran**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Nama Field** | **Type** | **Size** | **Keterangan** |
| 1. | Id\_kelahiran | Int | 10 | Kode Kelahiran |
| 2. | No\_surat | Varchar | 20 | Nomor Surat |
| 3. | Nik | Varchar | 50 | Nomor Induk Kependudukan |
| 4. | Tgl\_lahir | Date | - | Tanggal Lahir |
| 5. | Kelahiran\_ke | Int | 2 | Kelahiran Nomor Anak |

* + 1. **Perancangan Antarmuka**

Rancangan antarmuka yang penulis buat dalam implementasi program adalah Form Utama, yang terdiri dari menu antara lain :

1. Data Master, yang terdiri dari Form Data Penduduk,   
   Form Data Keluarga, Form Data Kematian, Form Data Pindah, Form Data Pindahan.
2. Laporan, yang terdiri dari laporan data keluarga, laporan data penduduk, laporan data kelahiran, laporan data kematian, laporan data pindah, laporan data pindahan.
3. Users, yang terdiri dari beberapa jenis hak akses antara tiap-tiap RT, RW dan operator untuk mengakses aplikasi.
   * + 1. **Rancangan Masukan**

Rancangan antar muka untuk masukan data dalam program yang dibuat oleh penulis terdiri dari :

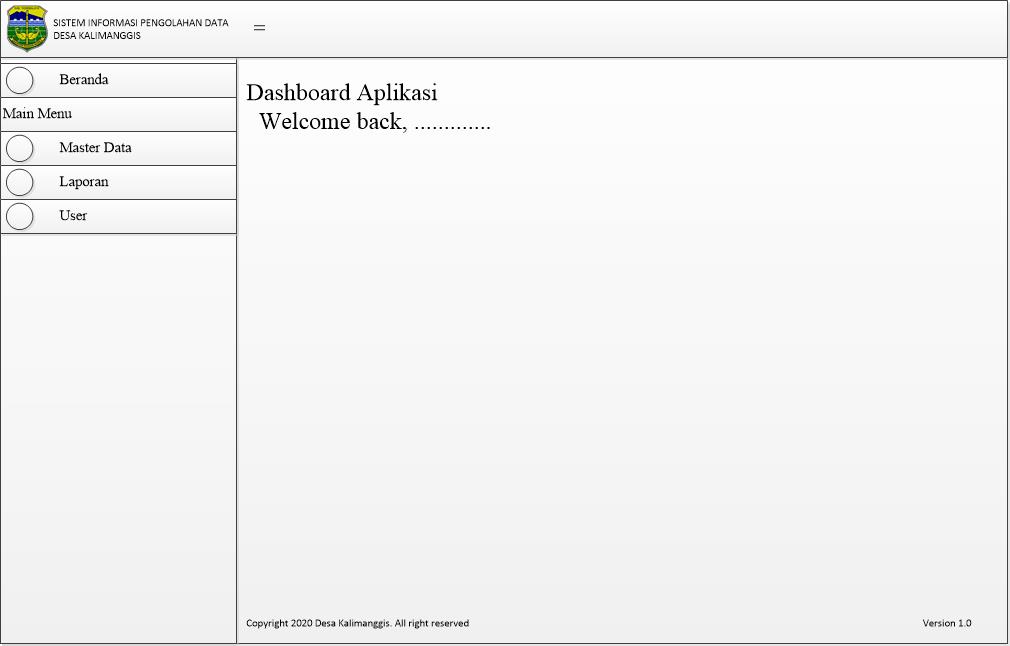
* 1. Rancangan Masukan

1. Rancangan Form Login



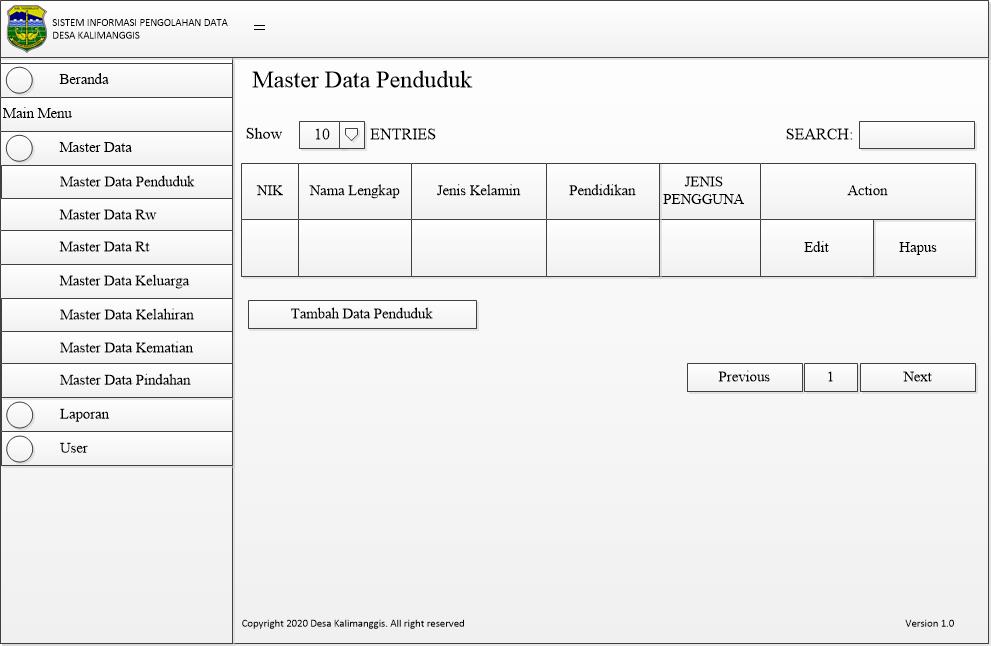
Gambar 3.18 Rancangan Form Login

1. Rancangan Form *Dashboard*



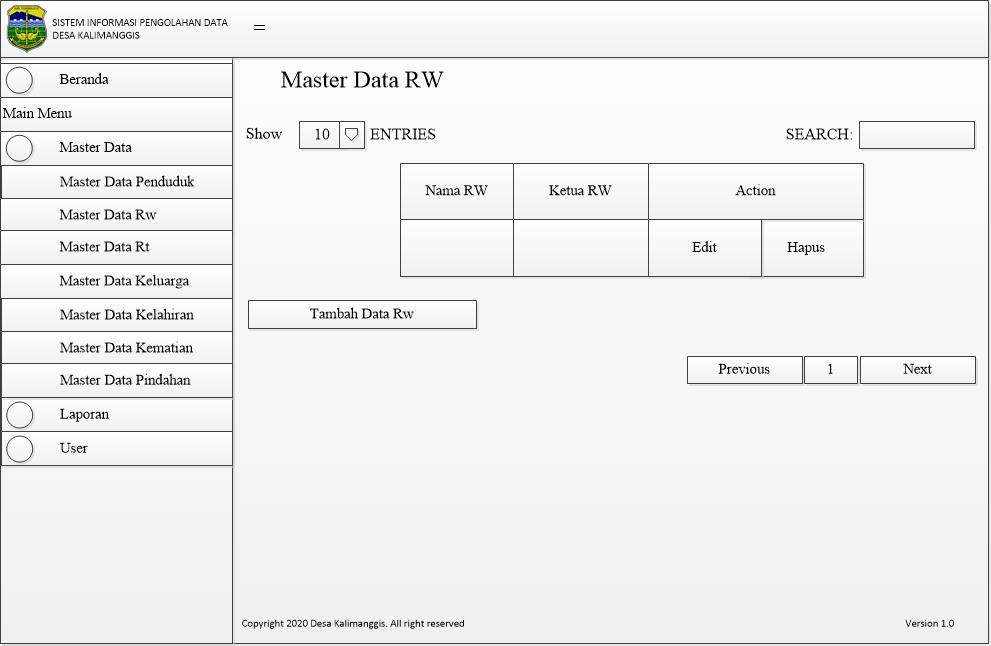
Gambar 3.19 Rancangan Form *Dashboard*

1. Rancangan FormData Penduduk

****

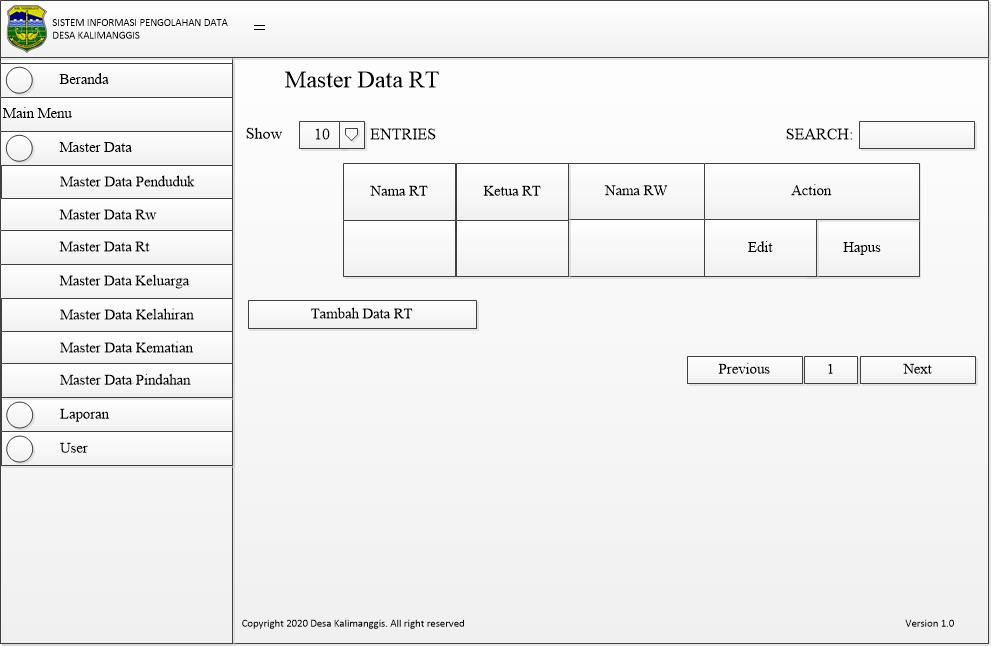
Gambar 3.20 Rancangan FormData Penduduk

1. Rancangan Form Data RW

****

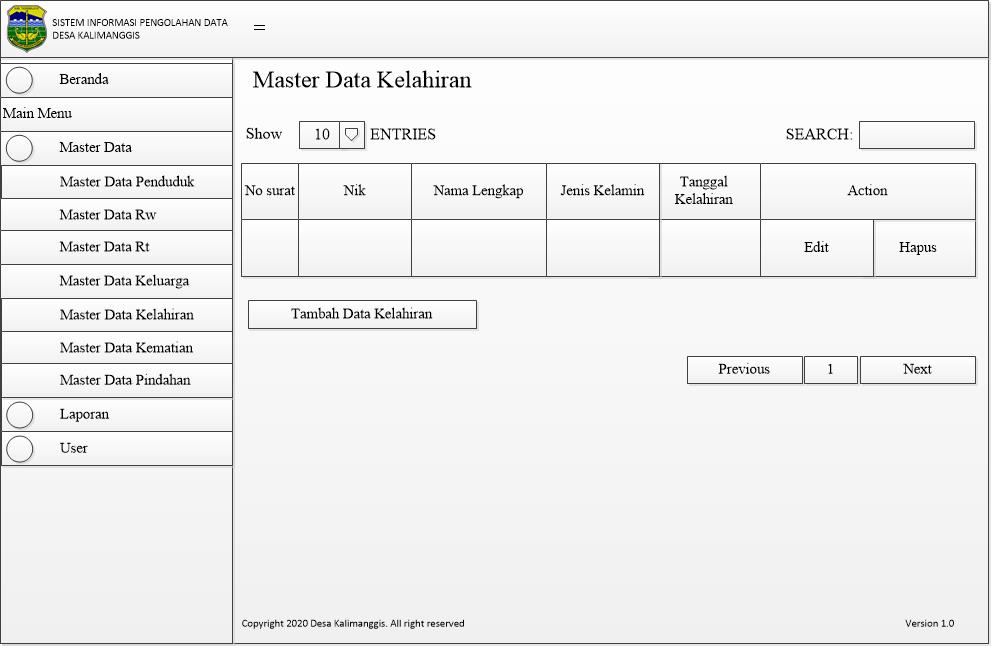
Gambar 3.21 Rancangan FormData RW

1. Rancangan Form Data RT

****

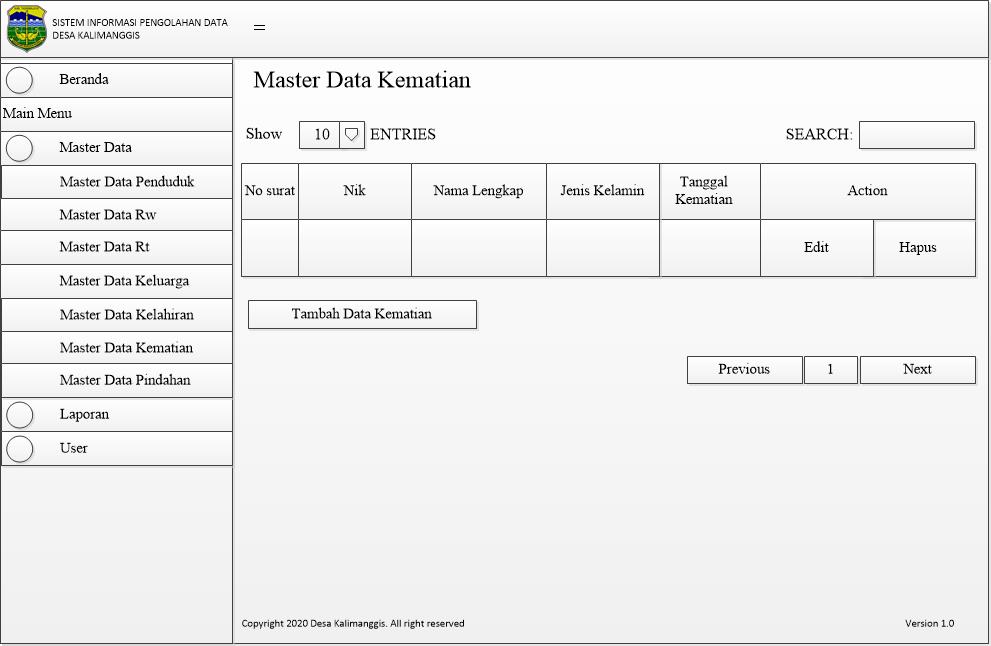
Gambar 3.22 Rancangan FormData RT

1. Pancangan Form Data Kelahiran

****

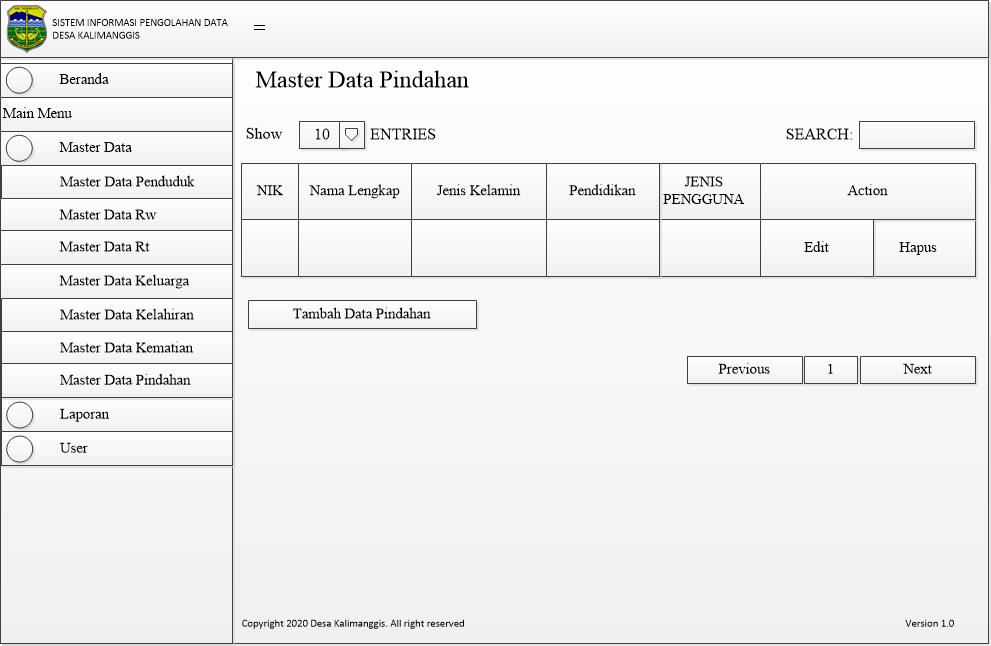
Gambar 3.23 Rancangan Form Data Kelahiran

1. Rancangan Form Data Kematian

****

Gambar 3.24 Rancangan Form Data Kematian

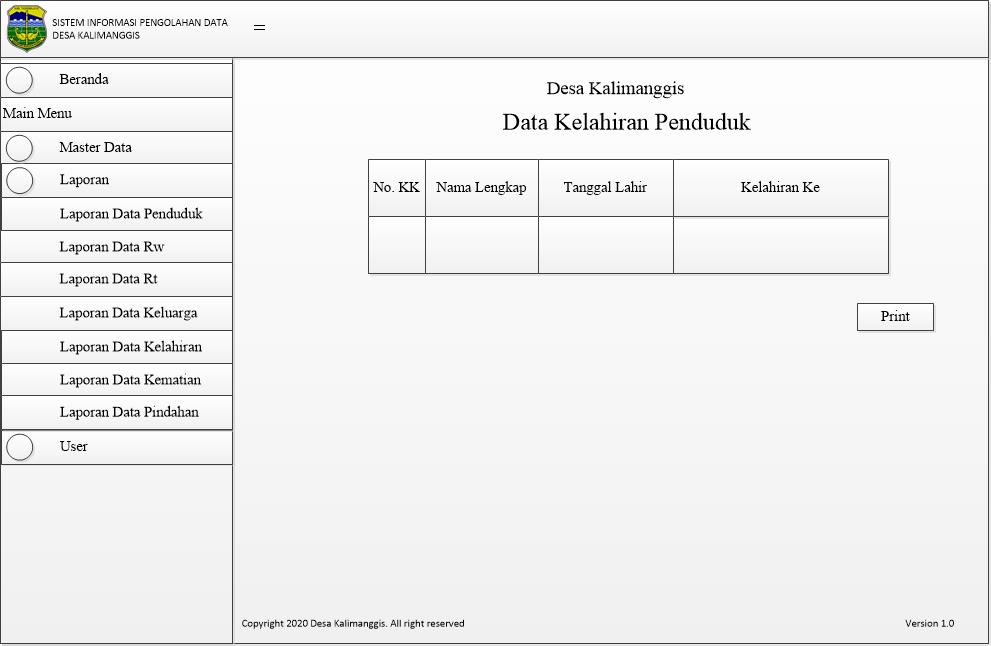
1. Rancangan Form Data Pindahan

****

Gambar 3.25 Rancangan Form Data Pindahan

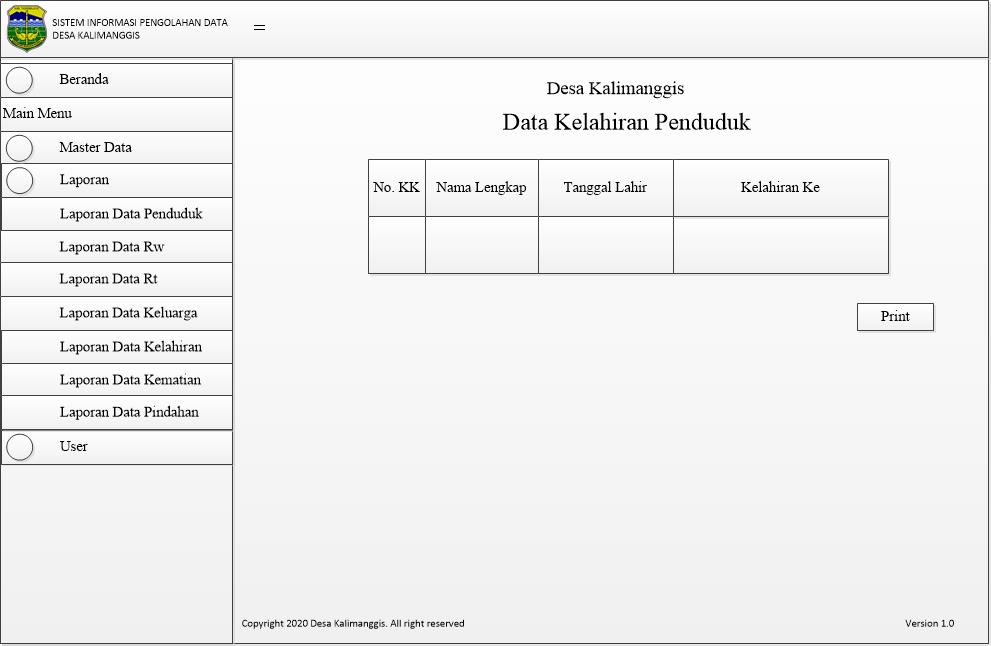
* 1. Rancangan Keluaran

1. Rancangan Tampilan Laporan Data Penduduk

****

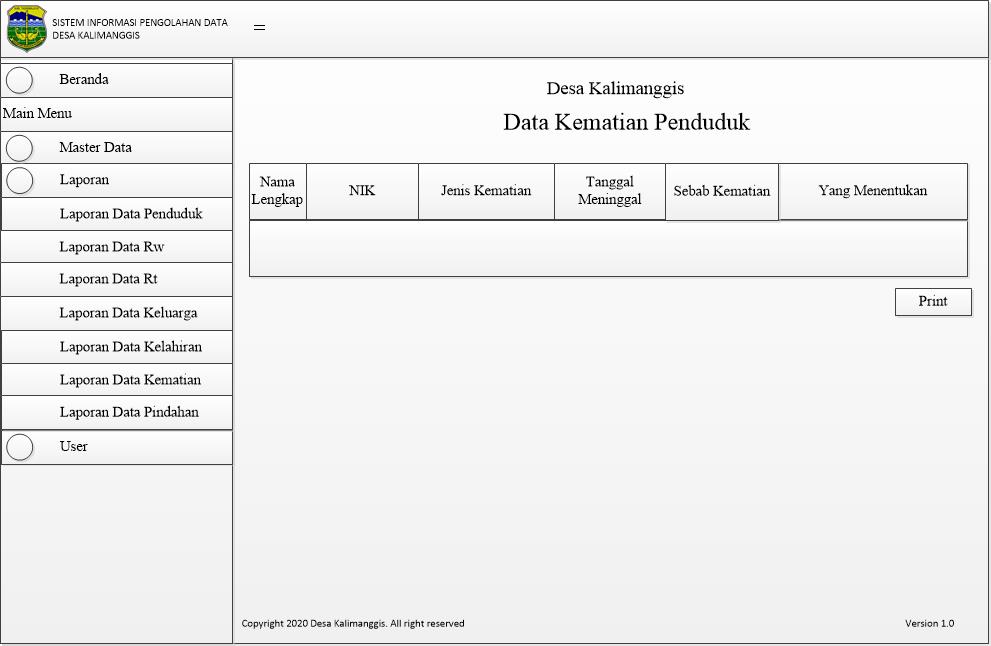
Gambar 3.26 Rancangan Tampilan Laporan Data Penduduk

1. Rancangan Tampilan Laporan Data Kelahiran



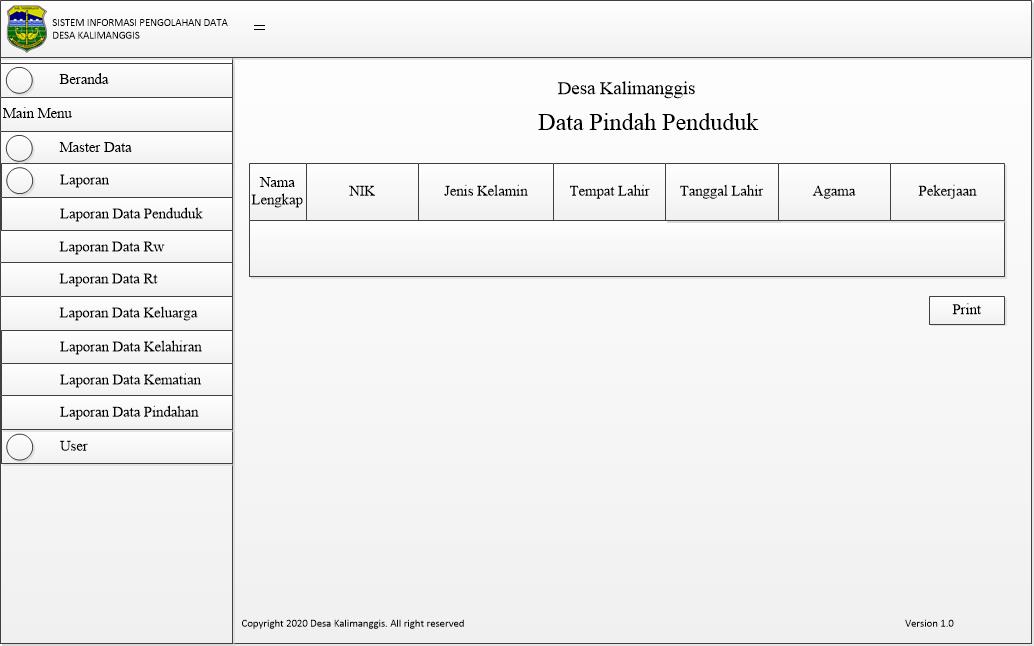
Gambar 3.27 Rancangan Tampilan Laporan Data Kelahiran

4. Rancangan Tampilan Laporan Data Kematian

****

Gambar 3.28 Rancangan Tampilan Laporan Data Kematian

5. Rancangan Tampilan Laporan Data Pindah

****

Gambar 3.29 Rancangan Tampilan Laporan Data Pindah

**BAB IV**

**IMPLEMENTASI DAN UJI COBA**

* 1. **Implementasi Sistem**

Implementasi sistem merupakan proses akhir dari penerapan sistem yang dirancang, dimana tahap ini merupakan tahap meletakkan sistem agar siap untuk dioprasikan dan dipandang sebagai hasil usaha mewujudkan sistem yang telah dirancang.

Pada bab ini akan diuraikan tentang proses pengujian dari sistem hasil analsis dan perancangan yang telah dibuat yaitu berupa pembuatan perangkat lunak dan perangkat keras. Pengujian ini meliputi percobaan pada setiap spesifikasi minimal yang dapat digunakan dan pengujian terhadap sistem yang sudah dibuat berdasarkan rancangannya yang diharapkan dapat berjalan sesuai dengan yang sudah direncanakan.

Pengujian ini dilakukan dengan tujuan untuk mengetahui cara kerja dari sitem yang telah dibuat tersebut, proses masukan dan keluaran agar sesuai dengan target yang diharapkan.

* + 1. **Perangkat Keras (*Hardware*) dan Perangkat Lunak (*Software*)**

1. Perangkat keras yang dapat digunakan pada spesifikasi minimal :
   1. Processor Intel Dual Core 1,80 GHz
   2. RAM 2 GB
   3. Kapasitas Harddisk yang dipakai 256 MB
   4. *Printer*
2. Perangkat keras yang digunakan Developer :

Laptop yang digunakan sebagai alat pembuatan Skripsi dan pembuatan Sistem Informasi Pengolahan Data Penduduk, dimana spesifikasi laptop yang digunakan penulis yaitu :

1. Processor : Intel(R) Core(TM) i3-2348M 2.3GHz
2. Harddisk : 500 GB
3. RAM : 4GB
4. Perangkat lunak yang digunakan :
5. Microsoft Windows 10 sebagai system operasi yang digunakan penulis.
6. Proses pembuatan aplikasi (coding) menggunakan bahasa PHP dengan bantuan aplikasi Notepad++.
7. Pembuatan Database menggunakan MYSQL.
8. XAMPP sebagai alat bantu untuk pengoneksian antara Apache dan MYSQL.
9. Menggunakan Microsoft Office 2016 sebagai alat bantu pengolahan kata dalam pembuatan laporan Tugas Akhir.
10. Menggunakan Microsoft Office Visio 2016 sebagai alat bantu dalam mendesain rancangan-rancangan.
11. Menggunakan Mozilla Firefox dan Google Chrome untuk pengetesan aplikasi yang telah dibuat.
    1. **Deskripsi Fungsi Sistem Yang Akan Dirancang**
       1. **Daftar File *Interface***

File-file *interface* yang sudah dibuat :

**Tabel 4.1 Daftar File *Interface* yang dibuat**

|  |  |  |
| --- | --- | --- |
| **No.** | **Nama Form** | **Keterangan** |
| 1. | Login | Halaman Login *User* |
| 2. | Dashboard | Halaman Menu Utama Bagian *User* |
| 3. | Master Data Penduduk | Halaman Data Penduduk |
| 4. | Master Data RW | Halaman Data RW |
| 5. | Master Data RT | Halaman Data RT |
| 6. | Master Data Kelahiran | Halaman Data Kelahiran |
| 7. | Master Data Kematian | Halaman Data Kematian |
| 8. | Master Data Pindahan | Halaman Data Pindahan |
| 9. | Laporan Data Penduduk | Halaman Laporan Data Penduduk |
| 10. | Laporan Data Kelahiran | Halaman Laporan Data Kelahiran |
| 11. | Laporan Data Kematian | Halaman Laporan Data Kematian |
| 12. | Laporan Data Pindahan | Halaman Laporan Data Pindahan |

* + 1. **Implementasi Dokumen File Basis Data**

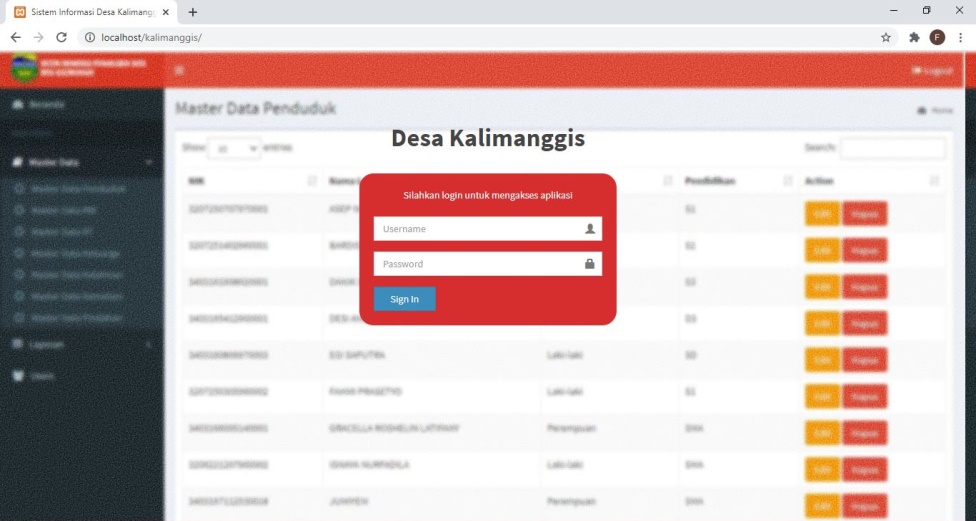
Dalam perancangan tabel-tabel basis data, penulis menggunakan MySQL, adapun file-file basis data yang terbentuk sebagai berikut :

**Tabel 4.2 Daftar File Basis Data**

|  |  |  |
| --- | --- | --- |
| **No.** | **Nama Tabel** | **Keterangan** |
| 1. | Mst\_kelahiran | Tabel Data Kelahiran |
| 2. | Mst\_kematian | Tabel Data Kematian |
| 3. | Mst\_penduduk | Tabel Data Penduduk |
| 4. | Mst\_pindahan | Tabel Data Pindahan |
| 5. | Mst\_rt | Tabel Data RT |
| 6. | Mst\_rw | Tabel Data RW |
| 7. | Provinces | Tabel Data Provinsi |
| 8. | *Users* | Tabel Data *User* |

* 1. ***Screenshoot* / Capture Tampilan Aplikasi**
     1. **Tampilan *Form* *Login***

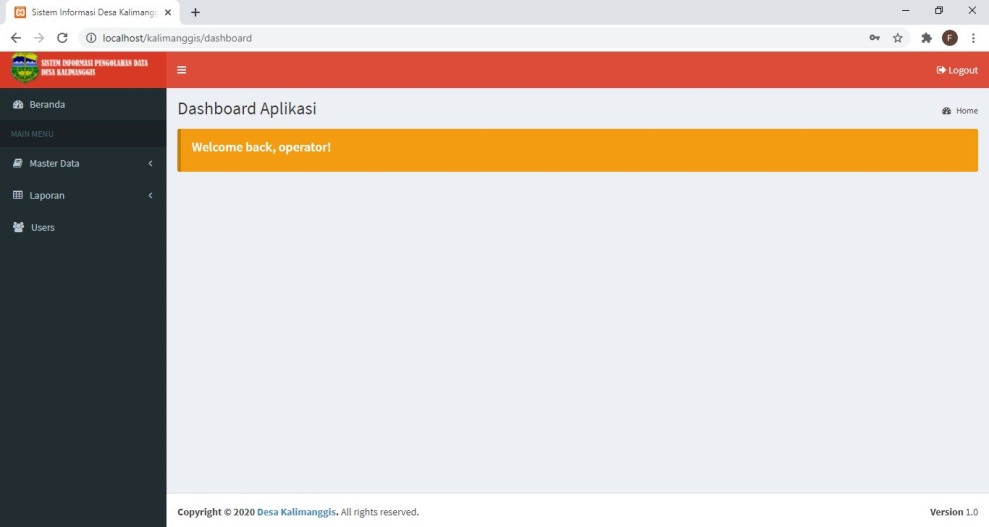
Tampilan ini digunakan untuk Proses autentikasi *login* *user*. Dapat dilihat pada gambar 4.1.

****

**Gambar 4.1 Tampilan *Form* *Login***

* + 1. **Tampilan *Form* *Dashboard***

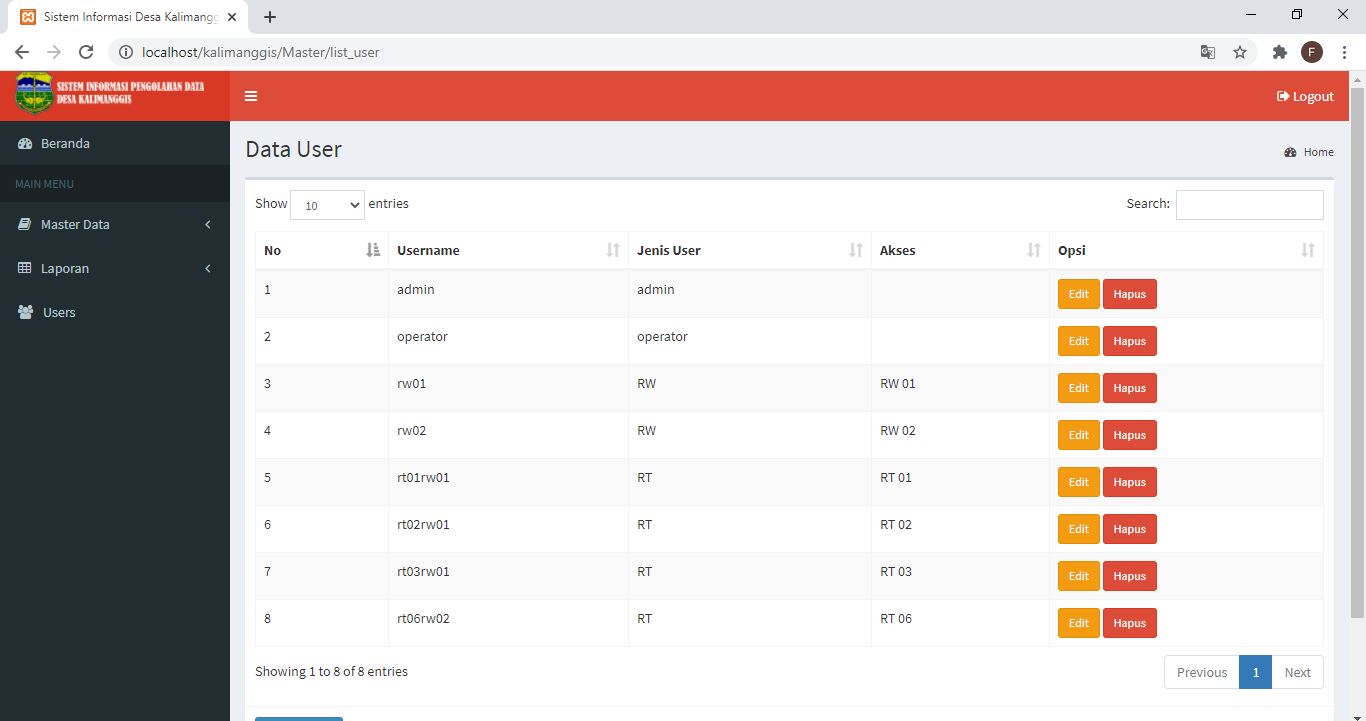
Berikut tampilan menu utama. Dapat dilihat pada gambar 4.2.

****

**Gambar 4.2 Tampilan *Form* *Dashboard***

* + 1. **Tampilan *Form* Data *User***

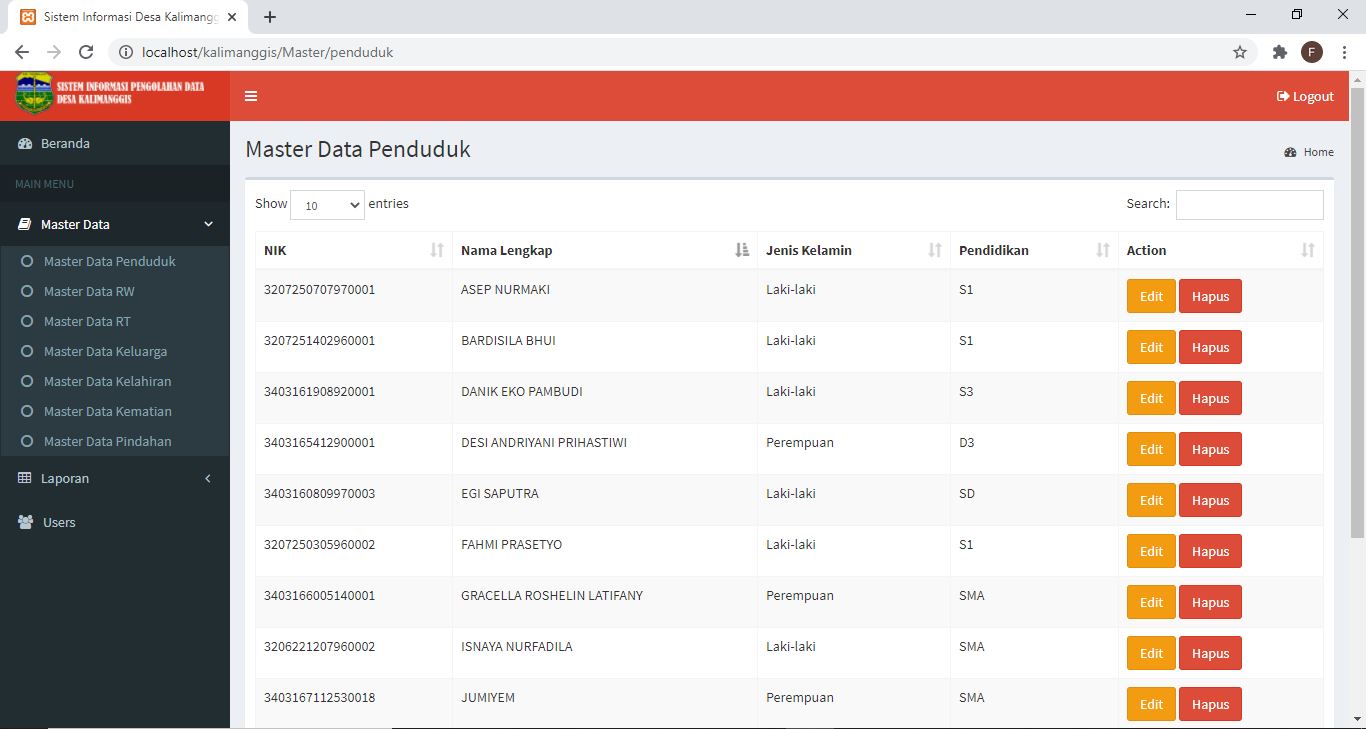
Berikut tampilan data *user*. Dapat dilihat pada gambar 4.3.

****

**Gambar 4.3 Tampilan *Form* Data *User***

* + 1. **Tampilan *Form* Data Penduduk**

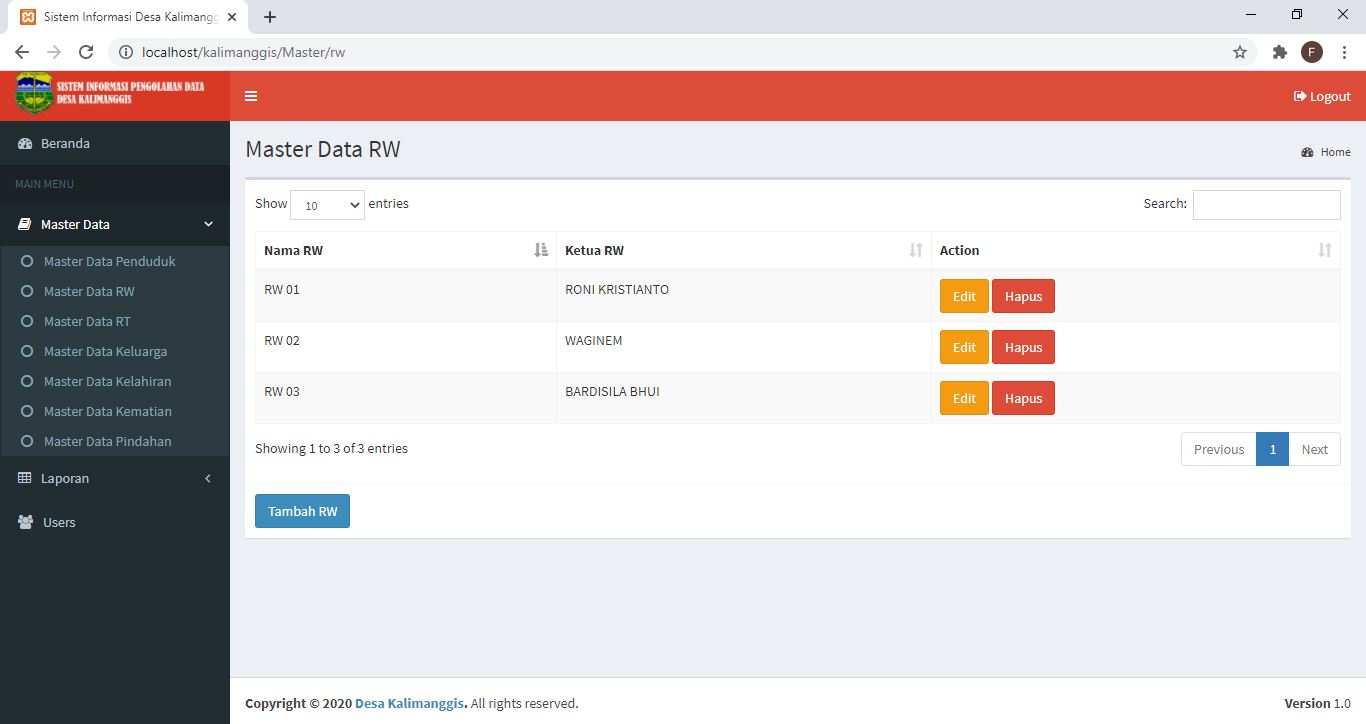
Berikut tampilan *form* data penduduk. Dapat dilihat pada gambar 4.4.

****

**Gambar 4.4 Tampilan *Form* Data Penduduk**

* + 1. **Tampilan *Form* Data RW**

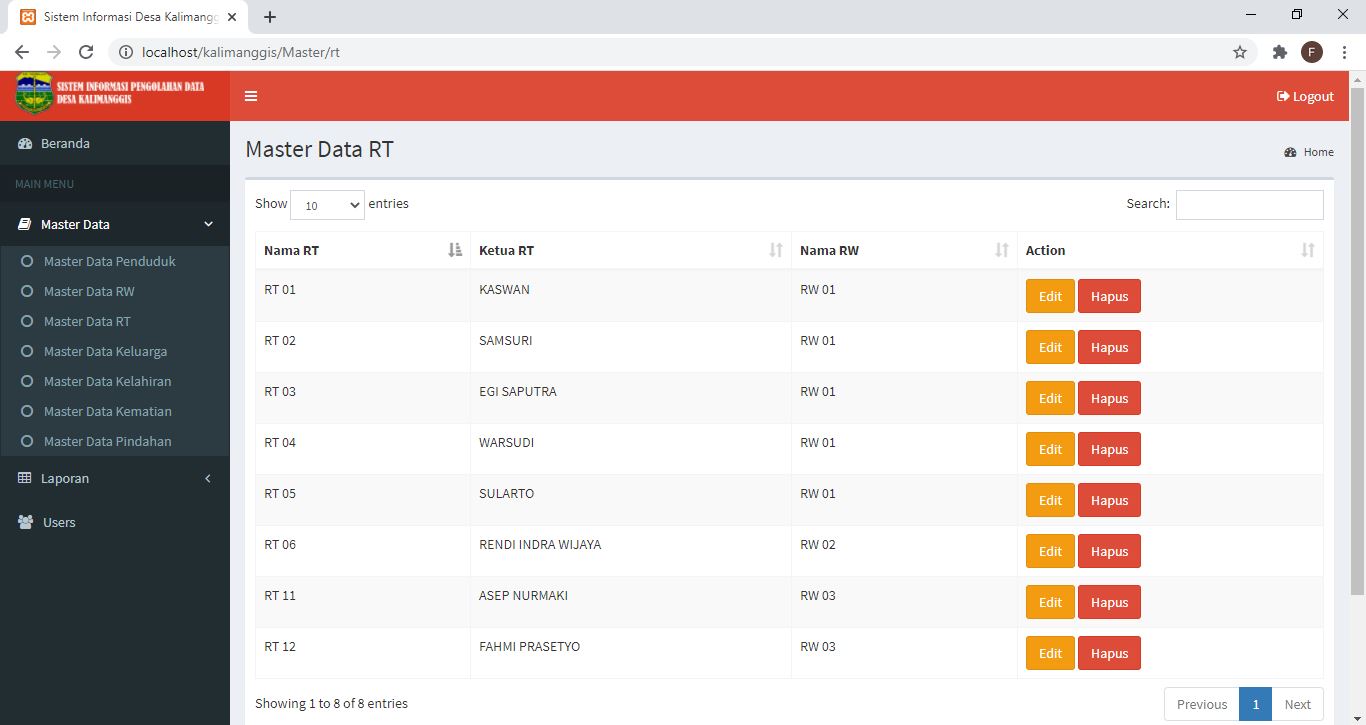
Berikut tampilan *form* data RW. Dapat dilihat pada gambar 4.5.

****

**Gambar 4.5 Tampilan *Form* Data RW**

* + 1. **Tampilan *Form* Data RT**

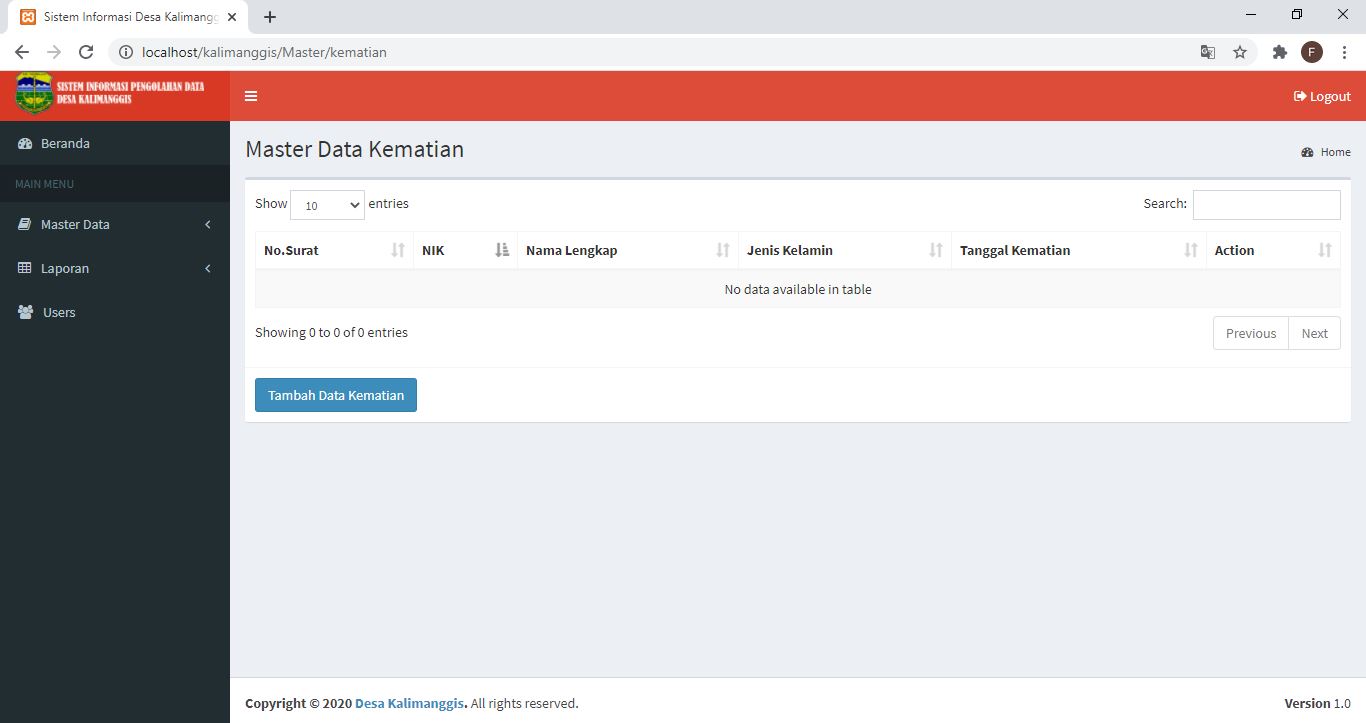
Berikut tampilan *form* data RT. Dapat dilihat pada gambar 4.6.

****

**Gambar 4.6 Tampilan *Form* Data RT**

* + 1. **Tampilan *Form* Data Kematian**

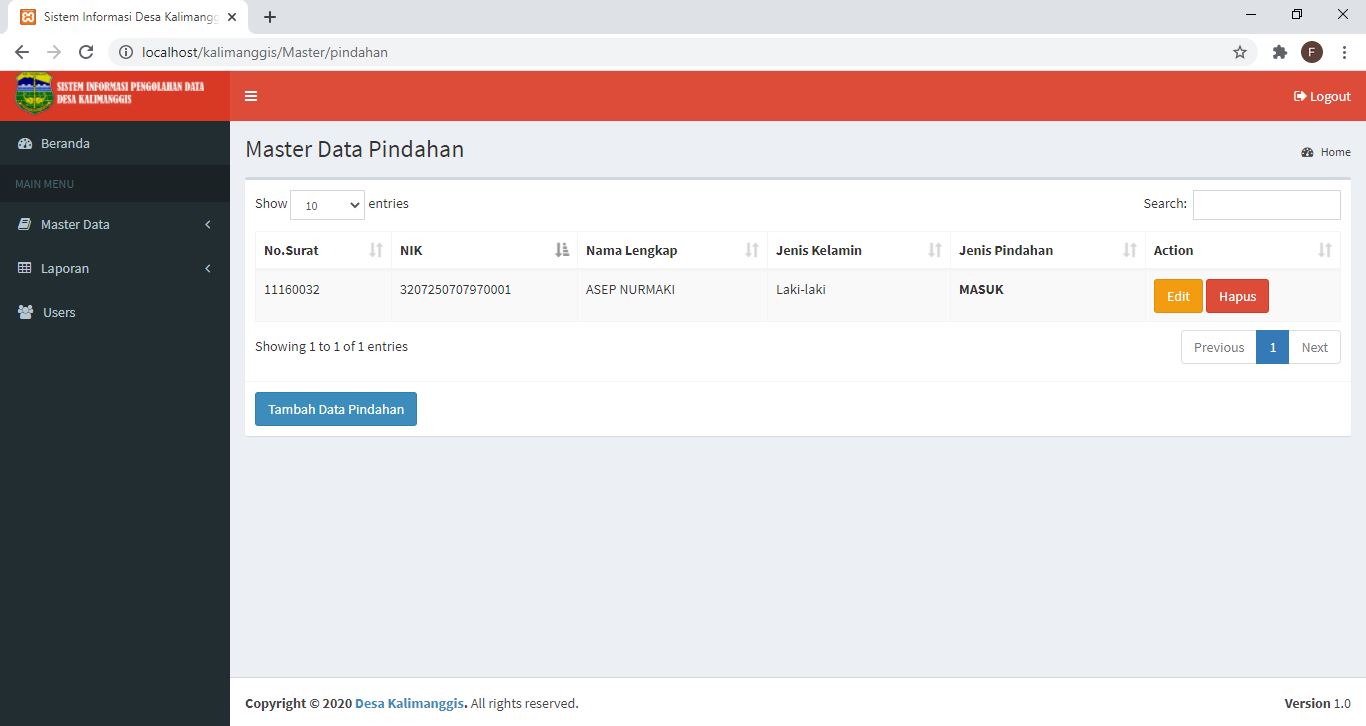
Berikut tampilan *form* data kematian. Dapat dilihat pada gambar 4.7.

****

**Gambar 4.7 Tampilan *Form* Data Kematian**

* + 1. **Tampilan *Form* Data Pindahan**

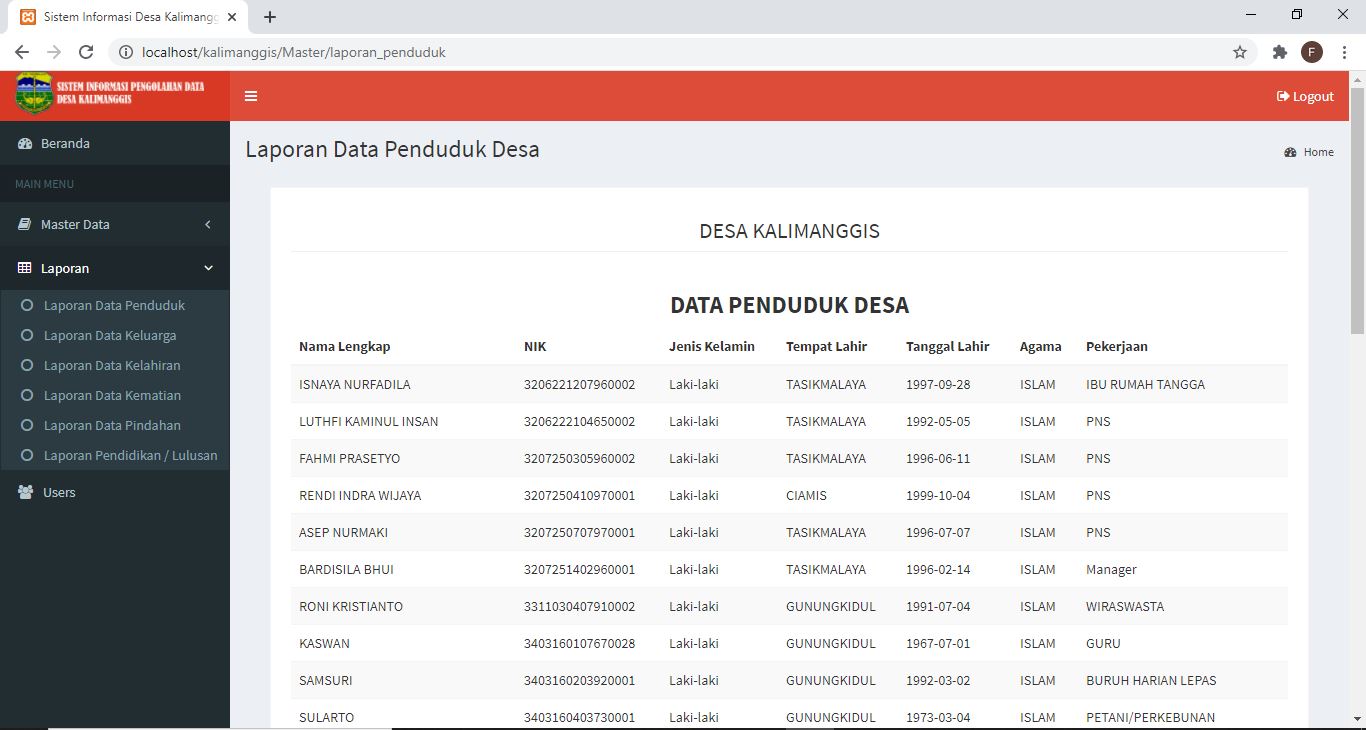
Berikut tampilan *form* data pindahan. Dapat dilihat pada gambar 4.8.

****

**Gambar 4.8 Tampilan *Form* Data Pindahan**

* + 1. **Tampilan *Form* Laporan Penduduk**

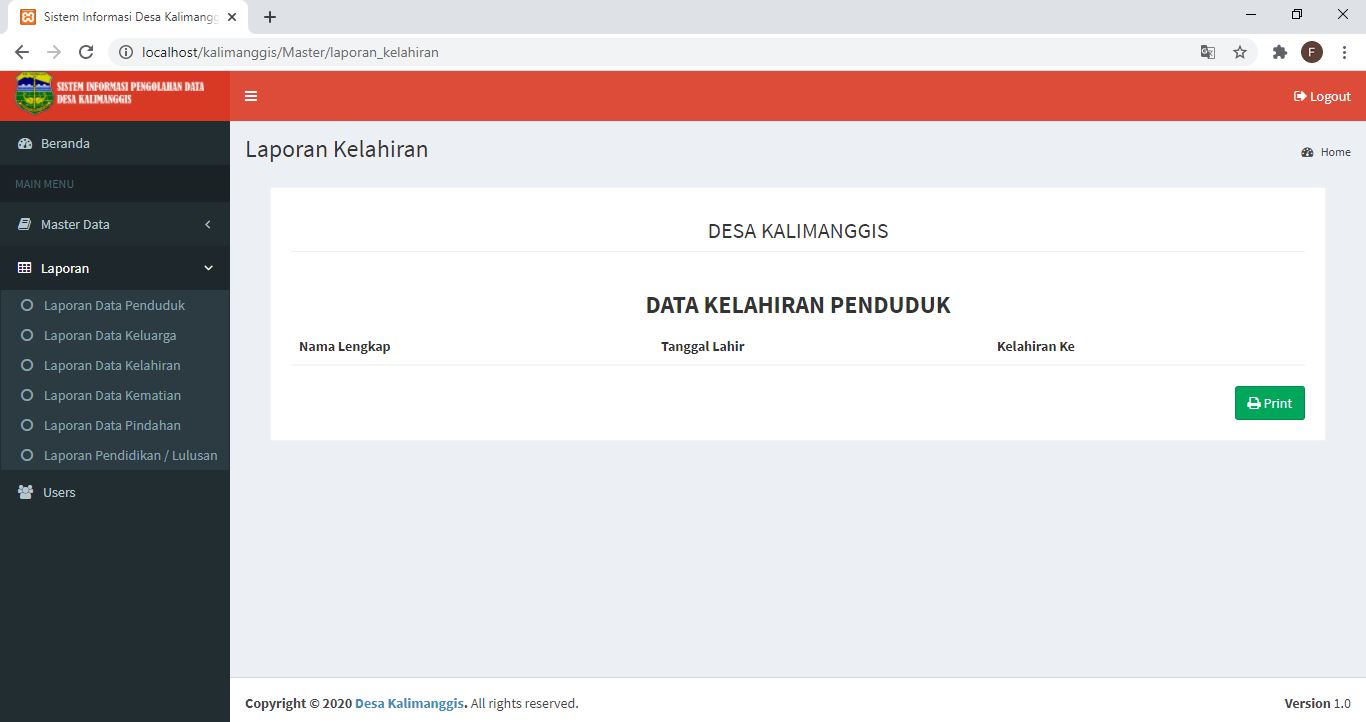
Berikut tampilan laporan data penduduk. Dapat dilihat pada gambar 4.9.

****

**Gambar 4.9 Tampilan *Form* Laporan Penduduk**

* + 1. **Tampilan *Form* Laporan Data Kelahiran**

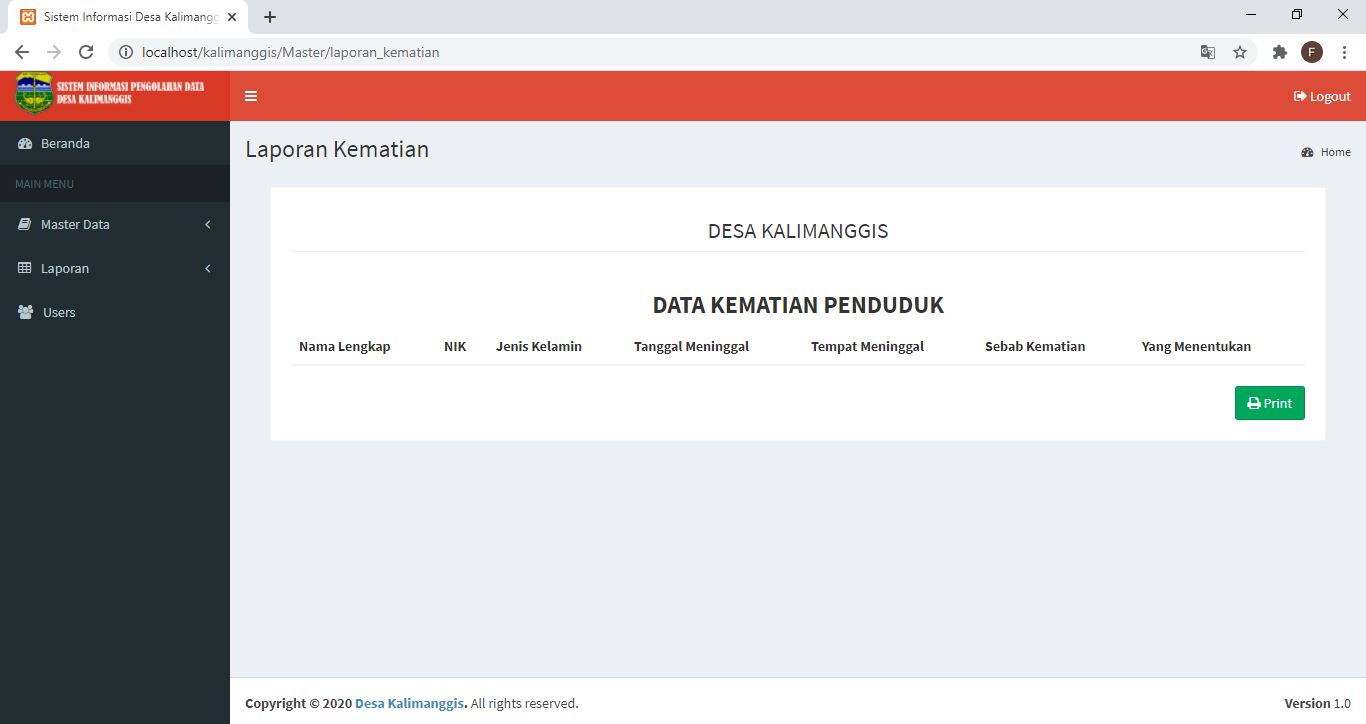
Berikut tampilan laporan data kelahiran. Dapat dilihat pada gambar 4.10.

****

**Gambar 4.10 Tampilan *Form* Laporan Data Kelahiran**

* + 1. **Tampilan *Form* Laporan Data Kematian**

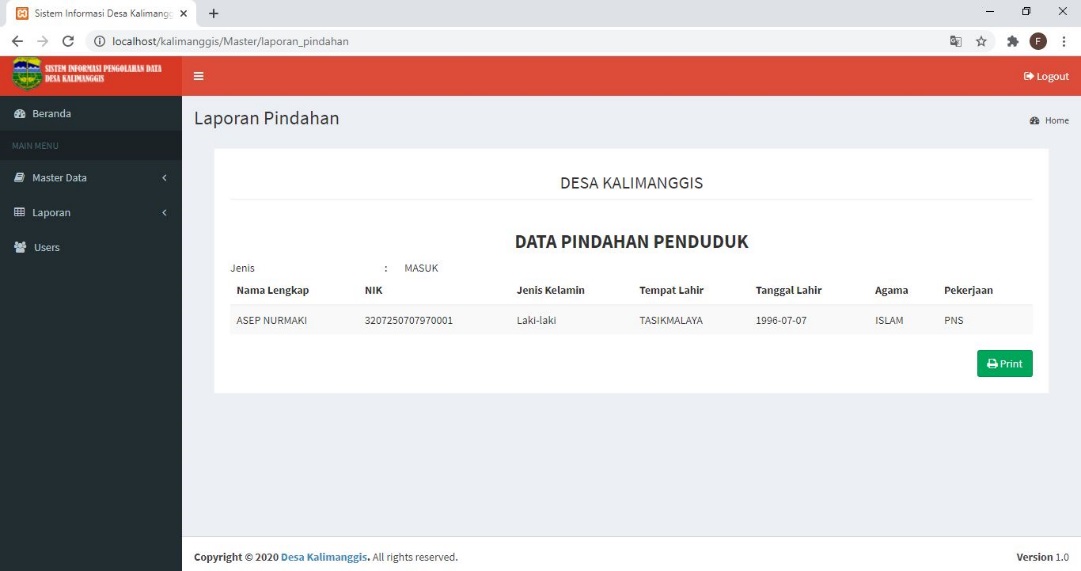
Berikut tampilan laporan data kematian. Dapat dilihat pada gambar 4.11.

****

**Gambar 4.11 Tampilan *Form* Laporan Data Kematian**

* + 1. **Tampilan *Form* Laporan Data Pindahan**

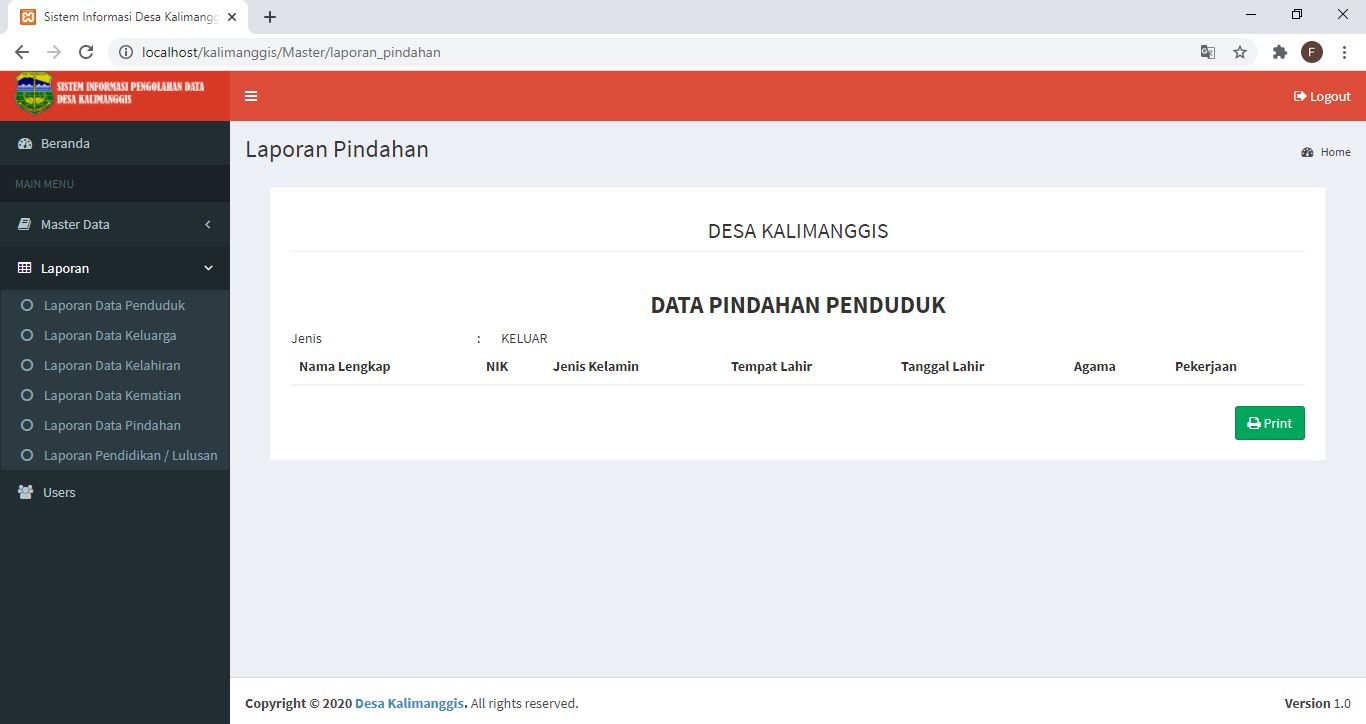
Berikut tampilan laporan data pindahan masuk. Dapat dilihat pada gambar 4.12.

****

**Gambar 4.12 Tampilan *Form* Laporan Data Pindahan Masuk**

* + 1. **Tampilan *Form* Laporan Data Pindah**

Berikut tampilan laporan data pindahan keluar. Dapat dilihat pada gambar 4.13.



**Gambar 4.13 Tampilan *Form* Data Pindahan Keluar**

* 1. **Pengujian Sistem**

Hasil pengujian dilakukan dengan menggunakan metode Black Box Testing. Metode black box memungkinkan perekayasa perangkat lunak mendapatkan serangkaian kondisi input yang sepenuhnya menggunakan semua persyaratan fungsional untuk suatu program. Pengujian ini berusaha menemukan kesalahan dalam kategori fungsi-fungsi yang tidak benar atau hilang, kesalahan interface, kesalahan dalam struktur data atau akses basis data eksternal, kesalahan kinerja, dan inisialisasi dan kesalahan terminal. Dengan menggunakan metode pengujian black box, perekayasa sistem dapat menemukan kesalahan dalam kategori sebagai berikut:

1. Fungsi tidak benar atau hilang

2. Kesalahan antar muka

3. Kesalahan pada struktur data (pengaksesan database)

4. Kesalahan inisialisasi dan akhir program

5. Kesalahan kinerja

Berikut pengujian yang dilakukan terhadap sistem dengan metode blackbox testing :

**Tabel 4.3** Hasil Pengujian Sistem *Login* Admin

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Deskripsi** | **Hal Yang Diharapkan** | **Hasil Pengujian** | **Keterangan** |
| 1 | *Input user* dan  *password* benar | Masuk Ke Sistem Dan Dapat Mengakses Menu Utama | Masuk Menu Utama | Sesuai |
| 2 | *Input user* dan  *password* salah | Tidak Dapat Masuk Ke Sistem | *Login* Gagal | Sesuai |

**Tabel 4.4** Hasil Pengujian Sistem

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Pengujian** | **Data Masukan** | **Yang diharapkan** | **Hasil Pengujian** | **Keterangan** |
| 1 | Master Data Penduduk | Memasukkan Kriteria Lengkap Sesuai Dengan Ketentuan | Data Disimpan Ke *Database* (Data Tidah Boleh Kosong) | Data Berhasil Disimpan Ke Database | Sesuai |
| 2 | Master Data RW | Memasukkan Kriteria Lengkap Sesuai Dengan Ketentuan | Data Disimpan Ke *Database* (Data Tidah Boleh Kosong) | Data Berhasil Disimpan Ke Database | Sesuai |
| 3 | Master Data RT | Memasukkan Kriteria Lengkap Sesuai Dengan Ketentuan | Data Disimpan Ke *Database* (Data Tidah Boleh Kosong) | Data Berhasil Disimpan Ke Database | Sesuai |
| 4 | Master Data Kelahiran | Memasukkan Kriteria Lengkap Sesuai Dengan Ketentuan | Data Disimpan Ke *Database* (Data Tidah Boleh Kosong) | Data Berhasil Disimpan Ke Database | Sesuai |
| 5 | Master Data Kematian | Memasukkan Kriteria Lengkap Sesuai Dengan Ketentuan | Data Disimpan Ke *Database* (Data Tidah Boleh Kosong) | Data Berhasil Disimpan Ke Database | Sesuai |
| 6 | Mater Data Pindahan | Memasukkan Kriteria Lengkap Sesuai Dengan Ketentuan | Data Disimpan Ke *Database* (Data Tidah Boleh Kosong) | Data Berhasil Disimpan Ke Database | Sesuai |
| 7 | Mencetak data Kriteria | Klik Cetak | Data Kriteria Di *Database* Dicetak Ke Kertas | Data Kriteria Di *Database* Berhasil Dicetak Ke Kertas | Sesuai |
| 8 | Mencari data Kriteria | Masukan Nama Kriteria Dikolom Pencarian | Data Kriteria Tampil | Data Kriteria Tampil | Sesuai |
| 9 | Lihat data Kriteria | Masuk Ke Menu Data Kriteria | Data Kriteria Tampil | Data Kriteria Tampil | Sesuai |

**Tabel 4.5** Hasil Pengujian Sistem Data Laporan

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Pengujian** | **Data Masukan** | **Yang diharapkan** | **Hasil Pengujian** | **Keterangan** |
| 1 | Lihat Perhitungan | Masuk Ke Menu Perhitungan | Data Perhitungan Tampil | Data Perhitungan Tampil | Sesuai |
| 2 | Mencetak Hasil Perhitungan | Klik Cetak | Data Perhitungan Di *Database* Dicetak Ke Kertas | Data Perhitungan Di *Database* Berhasil Dicetak Ke Kertas | Sesuai |

**Tabel 4.6** Hasil Pengujian Sistem Data *Password*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Pengujian** | **Data Masukan** | **Yang diharapkan** | **Hasil Pengujian** | **Keterangan** |
| 1 | Ubah Data Password | Mengupdate Password Sesuai Dengan Ketentuan | Data Diupdate Ke *Database* | Data Berhasil Disimpan Ke Database | Sesuai |
| 2 | Ubah Data Password | Mengganti Password | Data Berhasil Diupdate Ke *Database* | Data Berhasil Disimpan Ke Database | Sesuai |

**BAB V**

**PENUTUP**

* 1. **Kesimpulan**

Dari hasil analisis yang dilakukan oleh penulis terhadap sistem pengolahan data penduduk di Desa Kalimanggis, maka penulis dapat menyimpulkan beberapa hal sebagai berikut.

1. Pengolahan data penduduk di Desa Kalimanggis masih menggunakan sistem manual, sehingga memerlukan waktu yang lama dan terkadang menimbulkan kesalahan. Dengan sistem yang dirancang penulis didasarkan pada analisis dan perancangan sistem yang sedang berjalan supaya membuat kerja menjadi lebih efektif dan efisien.
2. Mempermudah pegawai Desa Kalimanggis dalam mengolah data penduduk dan membuat laporan, sehingga dapat mengurangi resiko kesalahan dalam kinerja.
   1. **Saran**

Agar sistem ini bisa bekerja dengan lebih baik sesuai dengan yang diharapkan pembaca Skripsi ini, maka penulis sarankan beberapa hal sebagai berikut.

1. Disarankan memberi pelatihan yang cukup agar operator atau petugas Desa Kalimanggis sebelum menggunakan aplikasi supaya dapat melakukan pemasukan data dengan benar.
2. Disarankan agar bisa membuat aplikasi pada *Smartphoe* Android supaya lebih mudah mengakses laporan lewat *handphone*.

**DAFTAR PUSTAKA**

Hendrianto, D. E. (2014). Pembuatan Sistem Informasi Perpustakaan Berbasis Website Pada Sekolah Menegah Pertama Negeri 1 Donorojo Kabupaten Pacitan. *Indonesia Journal on Networking and Security*, *3*(4), 57–64.

Irviani, R., & Oktaviana, R. (2017). Aplikasi Perpustakaan Pada SMA N1 Kelumbayan Barat Menggunakan Visual Basic. *Jurnal TAM ( Technology Acceptance Model )*, *8*(1), 63–69.

Jurnal, H. (2018). Jurnal Manajemen Dan Teknik Informatika. *Jumantaka*, *02*(1), 1. http://jurnal.stmik-dci.ac.id/index.php/jumantaka/article/view/364

Kecamatan, P., & Kota, C. (2019). Jurnal Manajemen Dan Teknik. *Jumantaka*, *03*(01), 81–90.

Nugraha, A. R., & Pramukasari, G. (2017). Sistem Informasi Akademik Sekolah Berbasis Web Di Sekolah Menengah Pertama Negeri 11 Tasikmalaya. *Jurnal Manajemen Informatika*, *4*(2), 1–10. https://www.cambridge.org/core/product/identifier/CBO9781107415324A009/type/book\_part

Pratama, Y. A., & Junianto, E. (2016). Sistem Pakar Diagnosa Penyakit Ginjal Dan Saluran Kemih Dengan Metode Breadth First Search. *Jurnal Informatika*, *2*(1). https://doi.org/10.31311/ji.v2i1.69

Siregar, S. R. S., & Sundari, P. (2016). Rancangan Sistem Informasi Pengelolaan Data Kependudukan Desa ( Studi Kasus di Kantor Desa Sangiang Kecamatan Sepatan Timur ). *Sisfotek Global*, *6*(1), 76–82.

Yulianto, A. A. (2009). Analisis perancangan sistem. *Analisis Perancangan Sistem*, 1–225.

# Lampiran *Script* Program

<!-- Content Header (Page header) -->

<section class="content-header">

<h1>

Dashboard Aplikasi

</h1>

<ol class="breadcrumb">

<li><a href="#"><i class="fa fa-dashboard"></i> Home</a></li>

</ol>

</section>

<!-- Main content -->

<section class="content">

<div class="row">

<div class="col-md-12">

<div class="callout callout-warning">

<h4>Welcome back, <?php echo $this->session-

>userdata('log\_username'); ?>!</h4>

</div>

</div>

</div>

</section>

<!-- /.content -->

<!-- Content Header (Page header) -->

<section class="content-header">

<h1>

<?php echo $title; ?>

</h1>

<ol class="breadcrumb">

<li><a href="#"><i class="fa fa-dashboard"></i> Home</a></li>

</ol>

</section>

<!-- Main content -->

<section class="content">

<div class="row">

<div class="col-xs-12">

<div class="box">

<!-- /.box-header -->

<div class="box-body">

<table id="mytable" class="table table-bordered table-striped" style="width:100%">

<thead>

<tr>

<th>NIK</th>

<th>Nama Lengkap</th>

<th>Jenis Kelamin</th>

<th>Pendidikan</th>

<th>Action</th>

</tr>

</thead>

<tbody>

<?php

foreach($data\_penduduk as $key => $val)

{

echo "

<tr>

<td>".$val['nik']."</td>

<td>".strtoupper($val['nama'])."</td>

<td>".$val['jenis\_kelamin']."</td>

<td>".$val['pendidikan']."</td>

<td>

<a class='btn btn-warning' href='".base\_url()."Master/edit\_penduduk/".$val['nik']."' >Edit</a>

<a class='btn btn-danger' href='".base\_url()."Master/hapus\_penduduk/".$val['nik']."'>Hapus</a>

</td>

</tr>

";

}

?>

</tbody>

</table>

</div>

<div class="box-footer">

<a href="<?php echo base\_url(); ?>Master/add\_penduduk" class="btn btn-primary">Tambah Penduduk</a>

</div>

</div>

<!-- /.box -->

</div>

<!-- /.col -->

</div>

<!-- /.row -->

</section>

<!-- /.content -->

<!-- Content Header (Page header) -->

<section class="content-header">

<h1>

<?php echo $title; ?>

</h1>

<ol class="breadcrumb">

<li><a href="#"><i class="fa fa-dashboard"></i> Home</a></li>

</ol>

</section>

<!-- Main content -->

<section class="content">

<div class="row">

<div class="col-xs-12">

<div class="box">

<!-- /.box-header -->

<div class="box-body">

<table id="mytable" class="table table-bordered table-striped" style="width:100%">

<thead>

<tr>

<th>No.KK</th>

<th>Kepala Keluarga</th>

<th>RT</th>

<th>RW</th>

<th>Action</th>

</tr>

</thead>

<tbody>

<?php

foreach($data\_keluarga as $key => $val)

{

/\*GET RT\*/

$getRT = $this->db->where('id\_rt',$val['rt'])->get('mst\_rt')->row\_array();

$getRW = $this->db->where('id\_rw',$val['rw'])->get('mst\_rw')->row\_array();

/\*GET KEPALA KELUARGA\*/

$getKepala = $this->db->where('posisi','KEPALAKELUARGA')->where('no\_kk',$val['no\_kk'])->get('mst\_penduduk')->row\_array();

echo "

<tr>

<td>".$val['no\_kk']."</td>

<td>".$getKepala['nama']."</td>

<td>".$getRT['nama\_rt']."</td>

<td>".$getRW['nama\_rw']."</td>

<td>

<a class='btn btn-success' href='".base\_url()."Master/edit\_keluarga/".$val['no\_kk']."' >Detail</a>

<a class='btn btn-danger' href='".base\_url()."Master/print\_keluarga/".$val['no\_kk']."' target='\_blank'>Print Kartu Keluarga</a>

</td>

</tr>

";

}

?>

</tbody>

</table>

</div>

</div>

<!-- /.box -->

</div>

<!-- /.col -->

</div>

<!-- /.row -->

</section>

<!-- /.content -->

<!-- Content Header (Page header) -->

<section class="content-header">

<h1>

<?php echo $title; ?>

</h1>

<ol class="breadcrumb">

<li><a href="#"><i class="fa fa-dashboard"></i> Home</a></li>

</ol>

</section>

<!-- Main content -->

<section class="content">

<div class="row">

<div class="col-xs-12">

<div class="box">

<!-- /.box-header -->

<div class="box-body">

<table id="mytable" class="table table-bordered table-striped" style="width:100%">

<thead>

<tr>

<th>No.Surat</th>

<th>Nama Lengkap</th>

<th>Tanggal Lahir</th>

<th>Kelahiran Ke-</th>

<th>Action</th>

</tr>

</thead>

<tbody>

<?php

foreach($data\_kelahiran as $key => $val)

{

echo "

<tr>

<td>".$val['no\_surat']."</td>

<td>".$val['nik']."</td>

<td>".$val['tgl\_lahir']."</td>

<td>".$val['kelahiran\_ke']."</td>

<td>

<a class='btn btn-warning' href='".base\_url()."Master/edit\_kelahiran/".$val['id\_kelahiran']."' >Edit</a>

<a class='btn btn-danger' href='".base\_url()."Master/hapus\_kelahiran/".$val['id\_kelahiran']."'>Hapus</a>

<a class='btn btn-success' href='".base\_url()."Master/cetak\_kelahiran/".$val['id\_kelahiran']."'>Cetak Surat Pengantar</a>

</td>

</tr>

";

}

?>

</tbody>

</table>

</div>

<div class="box-footer">

<a href="<?php echo base\_url(); ?>Master/add\_kelahiran" class="btn btn-primary">Tambah Kelahiran</a>

</div>

</div>

<!-- /.box -->

</div>

<!-- /.col -->

</div>

<!-- /.row -->

</section>

<!-- /.content -->

<!-- Content Header (Page header) -->

<section class="content-header">

<h1>

<?php echo $title; ?>

</h1>

<ol class="breadcrumb">

<li><a href="#"><i class="fa fa-dashboard"></i> Home</a></li>

</ol>

</section>

<!-- Main content -->

<section class="content">

<div class="row">

<div class="col-xs-12">

<div class="box">

<!-- /.box-header -->

<div class="box-body">

<table id="mytable" class="table table-bordered table-striped" style="width:100%">

<thead>

<tr>

<th>No.Surat</th>

<th>NIK</th>

<th>Nama Lengkap</th>

<th>Jenis Kelamin</th>

<th>Tanggal Kematian</th>

<th>Action</th>

</tr>

</thead>

<tbody>

<?php

foreach($data\_kematian as $key => $val)

{

/\*GET PENDUDUK\*/

$getPenduduk = $this->db->where('nik',$val['nik'])->get('mst\_penduduk')->row\_array();

echo "

<tr>

<td>".$val['no\_surat']."</td>

<td>".$val['nik']."</td>

<td>".strtoupper($getPenduduk['nama'])."</td>

<td>".$getPenduduk['jenis\_kelamin']."</td>

<td>".$val['tgl\_kematian']."</td>

<td>

<a class='btn btn-warning' href='".base\_url()."Master/edit\_kematian/".$val['id\_kematian']."' >Edit</a>

<a class='btn btn-danger' href='".base\_url()."Master/hapus\_kematian/".$val['id\_kematian']."'>Hapus</a>

</td>

</tr>

";

}

?>

</tbody>

</table>

</div>

<div class="box-footer">

<a href="<?php echo base\_url(); ?>Master/add\_kematian" class="btn btn-primary">Tambah Data Kematian</a>

</div>

</div>

<!-- /.box -->

</div>

<!-- /.col -->

</div>

<!-- /.row -->

</section>

<!-- /.content -->

<!-- Content Header (Page header) -->

<section class="content-header">

<h1>

<?php echo $title; ?>

</h1>

<ol class="breadcrumb">

<li><a href="#"><i class="fa fa-dashboard"></i> Home</a></li>

</ol>

</section>

<!-- Main content -->

<section class="content">

<div class="row">

<div class="col-xs-12">

<div class="box">

<!-- /.box-header -->

<div class="box-body">

<table id="mytable" class="table table-bordered table-striped" style="width:100%">

<thead>

<tr>

<th>No.Surat</th>

<th>NIK</th>

<th>Nama Lengkap</th>

<th>Jenis Kelamin</th>

<th>Jenis Pindahan</th>

<th>Action</th>

</tr>

</thead>

<tbody>

<?php

foreach($data\_pindahan as $key => $val)

{

/\*GET PENDUDUK\*/

$getPenduduk = $this->db->where('nik',$val['nik'])->get('mst\_penduduk')->row\_array();

echo "

<tr>

<td>".$val['no\_surat']."</td>

<td>".$val['nik']."</td>

<td>".strtoupper($getPenduduk['nama'])."</td>

<td>".$getPenduduk['jenis\_kelamin']."</td>

<td><b>".strtoupper($val['jenis\_pindahan'])."</b></td>

<td>

<a class='btn btn-warning' href='".base\_url()."Master/edit\_pindahan/".$val['id\_pindahan']."' >Edit</a>

<a class='btn btn-danger' href='".base\_url()."Master/hapus\_pindahan/".$val['id\_pindahan']."'>Hapus</a>

</td>

</tr>

";

}

?>

</tbody>

</table>

</div>

<div class="box-footer">

<a href="<?php echo base\_url(); ?>Master/add\_pindahan" class="btn btn-primary">Tambah Data Pindahan</a>

</div>

</div>

<!-- /.box -->

</div>

<!-- /.col -->

</div>

<!-- /.row -->

</section>

<!-- /.content -->

<!-- Content Header (Page header) -->

<section class="content-header">

<h1>

<?php echo $title; ?>

</h1>

<ol class="breadcrumb">

<li><a href="#"><i class="fa fa-dashboard"></i> Home</a></li>

</ol>

</section>

<!-- Main content -->

<section class="content">

<div class="row">

<div class="col-xs-12">

<div class="box">

<!-- /.box-header -->

<div class="box-body">

<table id="mytable" class="table table-bordered table-striped" style="width:100%">

<thead>

<tr>

<th>No</th>

<th>Username</th>

<th>Jenis User</th>

<th>Akses</th>

<th>Opsi</th>

</tr>

</thead>

<tbody>

<?php

$no = 1;

foreach($list\_user as $key => $val){

?>

<tr>

<td><?php echo $no; ?></td>

<td><?php echo $val['username']; ?></td>

<td><?php echo $val['jenis\_user']; ?></td>

<?php if($val['jenis\_user'] == 'RT' || $val['jenis\_user'] == 'RW'){ $sqlAuth = $this->db->where('id\_'.strtolower($val['jenis\_user']).'',$val['auth\_id'])->get('mst\_'.strtolower($val['jenis\_user']).'')->row\_array(); } ?>

<td><?php if($val['jenis\_user'] == 'RT' || $val['jenis\_user'] == 'RW'){echo $sqlAuth['nama\_'.strtolower($val['jenis\_user']).''];} ?></td>

<td>

<a href="<?php echo base\_url(); ?>master/edit\_user/<?php echo $val['id\_user']; ?>" class="btn btn-warning btn-sm">Edit</a>

<a href="<?php echo base\_url(); ?>master/hapus\_user/<?php echo $val['id\_user']; ?>" class="btn btn-danger btn-sm" onclick="return confirm('Apakah anda yakin data ini akan dihapus?')">Hapus</a>

</td>

</tr>

<?php $no++; } ?>

</tbody>

</table>

</div>

<!-- /.box-body -->

<div class="box-footer">

<a href="<?php echo base\_url(); ?>Master/add\_user" class="btn btn-primary btn-sm">Tambah User</a>

</div>

</div>

<!-- /.box -->

</div>

<!-- /.col -->

</div>

<!-- /.row -->

</section>

<!-- /.content -->