

**APLIKASI WEB MONITORING PENDATAAN PENDUDUK DESA
HARINGEN BARITO TIMUR KALIMANTAN TENGAH**

TUGAS AKHIR

Disusun untuk Memenuhi Syarat Kelulusan Program Strata I pada
Sekolah Tinggi Manajemen Informatika dan Komputer
(STMIK) Palangkaraya



OLEH

CEN YUPAN WEA
NIM C1555201013
PROGRAM STUDI TEKNIK INFORMATIKA

**SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER
(STMIK) PALANGKARAYA
2020**

**APLIKASI WEB MONITORING PENDATAAN PENDUDUK DESA
HARINGEN BARITO TIMUR KALIMANTAN TENGAH**

TUGAS AKHIR

Disusun untuk Memenuhi Syarat Kelulusan Program Strata I pada
Sekolah Tinggi Manajemen Informatika dan Komputer
(STMIK) Palangkaraya

OLEH

CEN YUPAN WEA
NIM C1555201013
PROGRAM STUDI TEKNIK INFORMATIKA

**SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER
(STMIK) PALANGKARAYA
2020**

ABSTRAK

CEN YUPAN WEA, C1555201013, 2020. *Aplikasi Web Monitoring Pendataan Penduduk Desa Haringen Barito Timur Kalimantan Tengah.* Pembimbing I Maura Widyaningsih, S.Kom, M.C.s. dan Pembimbing II H. Suratno, S.Kom., M.Si.

Di sadari atau tidak penerapan teknologi terutama berbasis internet ini telah berdampak pada keterbukaan informasi. Pesatnya perkembangan teknologi informasi memungkinkan pengembangan layanan yang lebih baik dalam suatu bidang pendataan desa terutama dalam sistem informasi di desa.

Berdasarkan latarbelakang yang telah di uraikan, maka masalah yang dapat di rumuskan adalah bagaimana membuat “ Aplikasi Web Monitoring Pendataan Penduduk Desa Haringen Barito Timur Kalimantan Tengah” Sistem yang dibangun meliputi pendataan penduduk yang ada di Desa Haringen Barito Timur, pindah domisili, blangko pendaftaran berbasis *web responsiv* menggunakan *php, databaseMySQL, HTML dan Bootstrap*.

Metode penulisan yang digunakan dalam penelitian ini meliputi metode pengumpulan data (wawancara dan observasi, studi pustaka, *planning design*, pemrograman dan pengujian) dan metode pengembangan perangkat lunak menggunakan metodologi pengembangan Dalam penyusunan penelitian ini penulis menggunakan metode pengembangan perangkat lunak yaitu *Extreme Programming (XP)*.

Aplikasi yang penulis buat berupa aplikasi monitoring *system* informasi di buat dengan menggunakan *php, databaseMySQL, HTML dan Bootstrap*. Pendataan yang dilakukan hanya pada Desa Haringen di kabupaten Barito Timur, Sistem yang dibangun meliputi pendataan penduduk yang ada di Desa Haringen Barito Timur, pindah domisili, blangko pendaftaran, Aplikasi web yang dibuat adalah aplikasi web yang bersifat *web responsive*, Pada Aplikasi ini dapat menampilkan grafik total warga , rentang umur warga, serta jenis kelamin data warga.

Demikian dari proses implementasi yang didapatkan hasil bahwa Aplikasi ini cukup baik dalam pendataan warga dan informasi warga sehingga dapat di jalankan dan tersimpan.

Kata kunci : *Aplikasi, Web, Monitoring* , Pendataan Penduduk

ABSTRACT

CEN YUPAN WEA, C1555201013, 2020.*Aplikasi Web Monitoring Pendataan Penduduk Desa Haringen Barito Timur Kalimantan Tengah*, Advisor I Maura Widyaningsih,S.Kom,M.C.s. and Advisor II H.Suratno,S.Kom.,M.Si.

It is realized or not the application of technology, especially internet-based, this has an impact on information disclosure. The rapid development of information technology allows the development of better services in a field of village data collection, especially in the information system in the village.

Based on the background that has been described, then the problem that can be formulated is how to make a "Web Application Monitoring Data for Population Data Collection in Haringen Village, East Barito, Central Kalimantan". responsive web uses php, MySQL database, HTML and Bootstrap.

The writing methods used in this study include data collection methods (interviews and observations, library research, planning design, programming and testing) and software development methods using development methodology. In preparing this research the authors used software development methods namely Extreme Programming (XP) .

The application that the author made in the form of an information system monitoring application was made using php, MySQL database, HTML and Bootsrap. registration blank, Web application made is a web application that is responsive web, In this application can display a graph of the total population, age range of citizens, as well as the gender of citizen data.

Thus from the implementation process, the results show that this application is quite good where this decision support system can do the processing of the data needed, so that it can produce a stored data Input and Output data in the form of data history.

Keyword :Application, Web,Monitoring , Population Data Collection

DAFTAR ISI

DAFTAR ISI.....	v
DAFTAR GAMBAR	vii
DAFTAR TABEL.....	ix
BAB I.....	1
PENDAHULUAN	1
A. Latar Belakang Masalah.....	1
B. Rumusan Masalah	2
C. Batasan Masalah.....	2
D. Tujuan dan Manfaat.....	2
E. Metode Penelitian.....	3
BAB II.....	8
KAJIAN TEORI	8
A. Definisi	8
1. Sistem	8
2. Informasi	8
3. Sistem Informasi.....	9
4. Kajian penelitian yang relevan	11
c. Desain Interface	33
BAB III	34
ANALISIS DAN PERANCANGAN SISTEM	34
A. Tinjauan Umum.....	34
B. Analisis.....	35
C. Analisis Kelayakan Sistem.....	39
D. Desain Sistem	41
BAB IV	52
IMPLEMENTASI DAN PEMBAHASAN.....	52
A. Implementasi	52
1. Uji Coba Sistem dan Program.....	52
2. Manual Program	55

BAB V.....	70
PENUTUP.....	70
A. Kesimpulan.....	70
B. Saran.....	71
DAFTAR PUSTAKA	72
LAMPIRAN.....	74

DAFTAR GAMBAR

Gambar 1. Pemodelan Metode <i>Extreme Programming</i>	20
Gambar 2. Xampp	29
Gambar 3. Use Case Diagram	32
Gambar 4. Desain Interface.....	33
Gambar 5. Analisis Sistem Baru	36
Gambar 6. Use Case Admin.....	44
Gambar 7. Activity Diagram Admin.....	46
Gambar 8. Class Diagram	48
Gambar 9. Sequence Diagram Admin Login	38
Gambar 10. Sequence Diagram Admin Kelola Data Warga.....	39
Gambar 11. Sequence Diagram Admin Kelola Data Kartu Keluarga	41
Gambar 12. Sequence Diagram Admin Kelola Data Mutasi	42
Gambar 13. Sequence Diagram Admin Kelola Menu Galeri	43
Gambar 14. Sequence Diagram Admin Kelola Menu User.....	44
Gambar 15. User Interface Admin Login	45
Gambar 16. User Interface Dasbor Admin	46
Gambar 17. User Interface Menu Data Warga.....	47
Gambar 18. User Interface Menu Data Kartu Keluarga	48
Gambar 19. User Interface Menu Data Mutasi	49
Gambar 20. User Interface Menu Galeri.....	50
Gambar 21. User Interface Menu User	51
Gambar 22. Login Aplikasi Pendataan Warga.....	56
Gambar 23. Halaman Beranda Admin	57
Gambar 24. Halaman Data Warga	57
Gambar 25. Halaman Tambah Data Warga	58
Gambar 26. Halaman Lihat Data Warga.....	59
Gambar 27. Refresh Data Warga	60
Gambar 28. Halaman Data Kartu Keluarga	60
Gambar 29. Halaman Tambah Data Kartu Keluarga	61
Gambar 30. Halaman Lihat Data Kartu Keluarga.....	62
Gambar 31. Halaman Refresh Data Kartu Keluarga.....	62
Gambar 32. Halaman Cetak Data Kartu Keluarga.....	63
Gambar 33. Halaman Data Mutasi.....	63
Gambar 34. Halaman Lihat Data Mutasi	64
Gambar 35. Halaman Refresh Data Mutasi	64
Gambar 36. Halaman Cetak Data Mutasi	65
Gambar 37. Halaman Galeri	65

Gambar 38. Halaman Tambah Data Galeri.....	66
Gambar 39. Halaman User.....	67
Gambar 40. Halaman Tambah Data User	67
Gambar 41. Halaman Lihat Data User.....	68
Gambar 42. Tampilan Menginstal Google Chrome.....	68
Gambar 43. Tampilan halaman utama	69

DAFTAR TABEL

Tabel 1 Kajian Penelitian Yang Relevan	11
Tabel 2. Sintak <i>Variable</i> pada PHP.....	14
Tabel 3. Sintak <i>Java Script</i>	16
Tabel 4 . Contoh penggunaan Tipe Data pada <i>Java Script</i>	16
Tabel 5. <i>Operator Java Script</i>	17
Tabel 6. Simbol <i>Use Case Diagram</i>	23
Tabel 7. Simbol Class Diagram	24
Tabel 8. Simbol Class Diagram	25
Tabel 9. Simbol Activity Diagram	26
Tabel 10. Simbol Sequence Diagram.....	27
Tabel 11Deskripsi Aktor	42
Tabel 12Deskripsi <i>Use Case Admin</i>	42
Tabel 13Blackbox Login.....	52
Tabel 14Blackbox Kelola Data Warga	53
Tabel 15Blackbox Kelola Data Kartu Keluarga	54
Tabel 16Blackbox Kelola Data Mutasi	54
Tabel 17Blackbox Kelola Galeri.....	55
Tabel 18Blackbox Kelola User	55

BAB I

PENDAHULUAN

A. Latar Belakang Masalah

Disadari atau tidak penerapan teknologi informasi terutama berbasis internet ini telah berdampak pada keterbukaan informasi, pesatnya perkembangan teknologi informasi memungkinkan pengembangan layanan informasi yang lebih baik dalam suatu bidang pendataan desa terutama dalam pengelolaan sistem informasi di desa.

Penulis menyadari bahwa kebutuhan masyarakat di Desa Haringen Barito Timur Kalimantan Tengah, misalnya pembuatan kartu domisili dan juga data jumlah keluarga yang kurang akurat untuk memperoleh informasi sangat kurang karena sistem di Kantor Desa Haringen masih secara manual atau menggunakan kertas jurnal dalam memberikan data dan informasi.

Untuk mengangkat permasalahan tersebut dalam bentuk teknologi informasi yang akan memudahkan layanan dan penyajian bagi masyarakat Desa Haringen. Untuk memenuhi kebutuhan tersebut maka dibangun sebuah jaringan “Aplikasi *WebMonitoring* Pendataan Penduduk Desa Haringen Barito Timur Kalimantan Tengah”.

Dengan adanya aplikasi ini diharapkan dapat meningkatkan pelayanan perangkat Desa Haringen sehingga dapat memberikan data dan informasi lebih mudah dan efisien kepada masyarakat desa dengan mengembangkan

aplikasi ini kantor desa Haringen bisa lebih efisien dalam data-data penduduk dan informasi sewaktu-waktu jika di perlukan.

B. Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan, maka masalah yang dapat dirumuskan adalah bagaimana membuat “Aplikasi *WebMonitoring* Pendataan Penduduk Desa Haringen Barito Timur Kalimantan Tengah”?

C. Batasan Masalah

Agar permasalahan dalam penelitian ini tidak melebar luar, maka diberikan batasan-batasan untuk pendekatan permasalahan agar lebih terinci dalam pelaksanaannya, yaitu:

1. Pendataan hanya dilakukan di Desa Haringen Barito Timur saja.
2. Sistem yang dibangun meliputi pendataan penduduk yang ada di Desa Haringen Barito Timur, pindah domisili, blangko pendaftaran.
3. Data yang di olah berbasis *web responsive*.
4. Pembuatan *system* informasi di buat dengan menggunakan *php*, *databaseMySQL*, *HTML* dan *Bootsrap*.

D. Tujuan dan Manfaat

1. Tujuan

Tujuan yang ingin dicapai penulis adalah merancang aplikasi *userfriendly* untuk Desa Haringen Barito Timur yang dapat memberikan kemudahan dalam pengolahan data penduduk.

2. Manfaat

Penelitian ini diharapkan dapat bermanfaat bagi kepentingan berbagai pihak, antara lain :

- a. Bagi penulis penelitian ini diharapkan bermanfaat untuk mengembangkan ilmu pengetahuan sesuai dengan bidang yang dipelajari.
- b. Manfaat bagi Desa Haringen Barito Timur adalah untuk memberikan kemudahan dalam pendataan penduduk dan dapat memberikan informasi yang cepat dan tepat.
- c. Bagi STMIK Palangkaraya penelitian ini diharapkan dapat menjadi masukan evaluasi atas kemampuan mahasiswa dalam menerapkan ilmu yang didapat selama masa kuliah. Penelitian ini juga diharapkan dapat digunakan sebagai sarana pengenalan Referensi pada perpustakaan kampus dan dapat menjadi dokumen akademik yang berguna untuk dijadikan acuan bagi *Civitas* akademika di STMIK Palangkaraya.

E. Metode Penelitian

Metode yang digunakan dalam pembuatan aplikasi dan menyusun laporan penelitian :

1. Jenis Penelitian

Penelitian Pengembangan juga diartikan sebagai suatu proses atau langkah-langkah untuk mengembangkan suatu pendataan baru atau menyempurnakan pendataan yang telah ada yang dapat dipertanggung jawabkan.

2. Objek Penelitian

Objek penelitian terdapat pada Desa Haringen Barito Timur yang beralamat di jalan haringen No.45 Kecamatan Dusun Timur Kabupaten Barito Timur Kalimantan Tengah.

3. Metode Pengembangan

Dalam penyusunan penelitian ini penulis menggunakan metode pengembangan perangkat lunak yaitu *Extreme Programming (XP)*.

Menurut Pressman (2015:12), Metode *Extreme Programming (XP)* umumnya mempromosikan disiplin proses manajemen proyek yang mendorong inspeksi dan adaptasi, filosofi kepemimpinan yang mendorong kerja sama tim atau individu, pengorganisasian dan akuntabilitas, praktek rekayasa yang mungkin pengiriman perangkat lunak berkualitas tinggi dengan cepat dan pendekatan bisnis yang sejalan dengan pengembangan kebutuhan *user* dan tujuan perusahaan.

Penjelasan dari tahap-tahap model *Extreme Programming* adalah sebagai berikut:

4.) Metode Pengumpulan Data

Pengumpulan data yang dilakukan selama penelitian adalah sebagai berikut :

a. Pengamatan(*Observation*)

Proses pengumpulan data dengan langsung ke Desa Haringen Barito Timur untuk mengetahui data yang akan diperlukan untuk pembuatan penelitian tugas akhir. data-data yang harus di perlukan seperti nomor induk keluarga, nama, tempat tanggal lahir, status dalam keluarga dan pekerjaan.

b. Wawancara(*Interview*)

Metode yang digunakan penulis untuk mengumpulkan data dengan cara bertanya secara langsung pada pimpinan atau karyawan pada Desa Haringen Barito Timur yang berupa, permasalahan-permasalahan yang terjadi, serta sistem atau cara kerja yang ada sekarang. Sistem yang berjalan sekarang masih manual menggunakan buku jurnal untuk melakukan pendataan penduduk sedangkan untuk jaringan internet sudah menunjang.

c. Studi pustaka

Metode ini dilakukan dengan mempelajari dokumen atau buku mengenai pendataan desa, referensi yang berhubungan dengan sistem yang akan dibangun di Desa Haringen, dengan judul aplikasi web monitoring pendataan penduduk desa haringen barito timur provinsi Kalimantan tengah.

1) Planning

Tahap pertama yaitu bagian awal dari pengerjaan suatu proyek perangkat lunak. Dimulai dengan mempersiapkan segala hal yang diperlukan dalam pelaksanaan proyek.

2) Design (Desain)

Pada tahap ini penulis membuat sebuah aplikasi pendataan dan informasi data desa agar data lebih akurat berdasarkan dari proses wawancara dan *planning* maka penulis menetapkan kebutuhan-kebutuhan apa saja yang digunakan pada sistem untuk membantu menyelesaikan permasalahan pengelolaan sistem berupa *uml, usecase, activity diagram*.

Menurut Jogiyanto (2005:130), di dalam tahap analisis sistem terdapat langkah-langkah dasar yang harus dilakukan oleh analisis sistem sebagai berikut :

3) Pemrograman (*Coding*)

Tahap ketiga sering disebut juga sebagai tahap implementasi perangkat lunak atau *coding*. Dengan kata lain, pada tahap ini dilakukan implementasi hasil rancangan kedalam baris kode program yang dimengerti oleh mesin komputer. Dalam tahap ini yang akan penulis lakukan adalah membuat sistem menggunakan bahasa pemrograman *PHP* dan *FrameworkBootstrap*.

4) Pengujian (*Testing*)

Tahap yang keempat adalah Pengujian atau *testing* Sistem. Dalam tahap ini yang akan penulis lakukan adalah menggabungkan unit program dan mengujinya. Setelah program selesai dibuat maka yang akan dilakukan penulis adalah pengujian atau *testing* dengan menggunakan *black-box testing*. Pada pengujian *black-box* ini dilakukan untuk menguji fungsi-fungsi yang tidak benar atau hilang dan kesalahan *interface*.

B. Sistematika Penulisan

Adapun sistematika penulisan dalam penulisan Tugas Akhir ini terdiri dari beberapa bab dan masing-masing bab membahas dan menguraikan pokok permasalahan yang berbeda. Sebagai gambaran di sini penulis menyertakan garis-garis besarnya yaitu:

BAB I PENDAHULUAN

Berisi latar belakang masalah, rumusan masalah, batasan masalah, tujuan penelitian manfaat penelitian, metodologi, sistematika penulisan laporan, dan jadwal rencana pelaksanaan

BAB II TINJAUAN PUSTAKA

Tinjauan pustaka berisi uraian singkat hasil-hasil penelitian atau analisis terdahulu yang ada hubungannya dengan permasalahan yang akan ditinjau dalam Tugas Akhir.

BAB III LANDASAN TEORI

Membahas mengenai uraian dasar teori yang akan digunakan penulis dalam melakukan perancangan dan pembuatan program

BAB IV ANALIS DAN PERANCANGAN SISTEM

Berisi penjelasan mengenai perangkat lunak yang akan dibuat, serta desain sistem yang akan diterapkan.

BAB V PENUTUP

Pada bab ini penulis akan memberikan kesimpulan terhadap hasil penelitian yang dilakukan dan saran bagi siapa saja yang berminat membuat atau mengembangkan aplikasi serupa.

BAB II

KAJIAN TEORI

A. Definisi

1. Sistem

Istilah sistem paling sering digunakan untuk menunjuk pengertian metode atau cara dan sesuatu himpunan unsur atau komponen yang saling terhubung satu sama lain menjadi satu kesatuan yang utuh.

Menurut Jogiyanto (2015:1) bahwa “sistem adalah suatu jaringan kerja dari prosedur yang saling berhubungan, berkumpul bersama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran yang tertentu”

Menurut, Romney (2015:3), Sistem adalah serangkaian dua atau lebih komponen yang saling terkait dan berinteraksi untuk mencapai tujuan.

Menurut Alexander (2015:2) yang dialih bahasa oleh Mardi adalah: “Suatu *group* dari beberapa elemen, baik berbentuk fisik maupun bukan fisik, yang menunjukkan suatu kumpulan saling berhubungan di antaranya dan berinteraksi bersama menuju satu atau lebih tujuan, sasaran atau akhir dari sistem”.

2. Informasi

Menurut Jogianto (2015:11), Informasi adalah data yang telah diolah menjadi bentuk yang telah berarti bagi yang menerimanya.

Menurut Tantra (2015:1), Informasi dapat dipahami sehingga pemrosesan *input* yang terorganisir, wmemiliki arti, dan berguna bagi orang yang menerimanya.

Sedangkan Menurut Davis yang dialih bahasa oleh Mardi (2015:5) menyatakan bahwa: “Informasi adalah data yang telah diolah ke dalam suatu bentuk yang berguna bagi penerimanya dan nyata atau berupa nilai yang dapat dipahami di dalam keputusan sekarang maupun masa depan.”

3. Sistem Informasi

Sistem informasi dapat diidentifikasi sebagai suatu sistem didalam sebagai suatu organisasi yang merupakan kombinasi dari orang-orang, fasilitas, teknologi, media prosedur-prosedur dan pengendalian yang ditujukan untuk mendapatkan jalur komunikasi penting, memproses tipe transaksi rutin tertentu, memberi sinyal kepada manajemen dan yang lainnya terhadap kajian-kajian internal dan eksternal yang penting dan menyediakan suatu dasar informasi untuk pengambilan keputusan.

Menurut Sutarman (2015:13) menyatakan Sistem Informasi sebagai berikut: “Sistem Informasi merupakan mengumpulkan memproses, menyimpan, menganalisis, menyebarkan, informasi untuk tujuan tertentu”.

Dengan kata lain, sistem informasi merupakan kesatuan elemen-elemen yang saling berinteraksi secara sistematis dan teratur untuk menciptakan dan membentuk aliran informasi yang akan mendukung pembuatan keputusan dan melakukan kontrol.

Menurut Arief, M.Rudianto (2016:151) didalam bukunya istilah *MySQL* dikembangkan oleh sebuah perusahaan Swedia bernama *MySQLAB* yang pada saat itu bernama *TcXDataKonsult AB* sekitar tahun 1994-1995, namun cikal bakal kodenya sudah ada sejak 1979. Awalnya *TcX* membuat *MySWL* dengan tujuan pengembangan aplikasi *web* untuk klien.*TcX* merupakan perusahaan pengembang *software* dan konsultan *database*. Saat ini *MySQL* sudah diakuisisi oleh *Oracle Corp*. *MySQL* salah satu jenis *database server* yang sangat terkenal dan banyak digunakan untuk membangun aplikasi *web* yang menggunakan *database* sebagai sumber dan pengelolaan datanya. Kelebihan *MySQL* antara lain karena *MySQL* menggunakan *SQL* sebagai bahasa dasar untuk mengakses *database*-nya sehingga mudah untuk digunakan, kinerja *query* cepat, dan mencakupi untuk kebutuhan *database* perusahaan-perusahaan skala menengah-kecil. *MySQL* juga bersifat *open sourcedan free* (anda tidak perlu membayar untuk menggunakannya).

4. Kajian penelitian yang relevan

Untuk menghindari duplikasi dari pihak lain, penulis melakukan penelusuran terhadap penelitian-penelitian terdahulu. Dari hasil penelusuran penelitian terdahulu, diperoleh beberapa masalah yang berkaitan dengan masalah yang akan diteliti, yaitu:

Tabel 1 Kajian Penelitian Yang Relevan

NO	Penulis / Tahun	Topik Penelitian	Metode Pengembangan PL	Hasil	Perbedaan
1	Kusumawati/2018	kependudukan di desa kasiman pelitantan kecamatan Denpasar timur	<i>Waterfall</i>	Sistem Informasi Kependudukan Pada Kecamatan Kadunggora Kabupaten Garut	Sistem Informasi Kependudukan Pada Kecamatan Kadunggora Kabupaten Garut Sedangkan Pendataan Penduduk Desa Haringen Barito Timur Provinsi Kalimantan Tengah
2	Sunardi, Listiyono / 2016	Rancang Bangun Sistem Informasi Kependudukan Berbasis Android	<i>Waterfall</i>	Rancang Bangun Sistem Informasi Kependudukan Berbasis Android	Perbedaan Dalam Penulisan Pertama Adalah Pendataan Untuk Bantuan Desa

3	Noviyanto, Tedy Setiadi, Iis / 2015	Implementasi Sikades (Sistem Informasi Kependudukan Desa) Untuk Kemudahan Layanan Administrasi Desa Berbasis Web Mobile	<i>Waterfall</i>	Model Informasi Implementasi informasi kependudukan layanan administrasi	Perbedaan untuk Melakukan informasi untuk layanan administrasi desa
4	Marini / 2015	Perancangan Sistem Pendataan Penduduk pada Kelurahan Air Itam dengan Object Oriented	<i>Prototype</i>	Sistem Pendataan Penduduk pada Kelurahan Air Itam dengan Object Oriented	Perbedaan Untuk Mempermudah penduduk pada kelurahan air itam dengan object oriented
5	Kholdani/ 2017	Pembangunan Sistem Informasi Pengolahan Data Penduduk Di Kecamatan Banjarbaru Utara	<i>Waterfall</i>	Membangunan Sistem Informasi Pengolahan Data Penduduk Di Kecamatan Banjarbaru Utara	Perbedaan Hanya Pada Membangun sistem informasi pengolahan data penduduk di kecamatan Banjarbaru Utara

1) **HTML**

Solichin (2016:10) mengemukakan bahwa HTML merupakan bahasa pemrograman *web* yang memberitahukan peramban *web* (*web browser*) bagaimana menyusun dan menyajikan konten di halaman web.

2) **Web Mobile**

Menurut Abdurahman (2016:49), *web mobile* adalah lanjutan dari aplikasi *web* tradisional. Saat online menggunakan *smartphone*, tablet, atau komputer, desainnya beradaptasi dengan perangkat yang digunakan tersebut. *Desain yang fleksibel* ini memaksimalkan kemampuan *web browser* yang digunakan di perangkat *mobile* yang digunakan.

Lebih jelasnya, *web mobile* merupakan halaman-halaman yang berisi informasi yang ditampilkan oleh browser seperti *Mozilla Firefox*, *Google Chrome* atau yang lainnya, tapi memiliki tampilan yang dapat menyesuaikan dengan perangkat pengaksesnya.

3) **Bootstrap**

Menurut Alatas (2013:2) *Bootstrap* merupakan *framework* ataupun *tool* untuk membuat aplikasi web ataupun situs web *responsive* secara cepat, mudah dan gratis.

4) **Framework**

Framework adalah kumpulan perintah satu fungsi dasar yang membentuk aturan-aturan tertentu dan saling berinteraksi satu sama lain sehingga dalam pembuatan aplikasi *website*, kita harus mengikuti aturan dari *framework* tersebut (Wardana, 2010:3).

5) PHP (*Hypertext preprocessor*)

Supono dan Putratama (2016:3) mengemukakan bahwa PHP (PHP: *Hypertext Preprocessor*) adalah suatu bahasa pemrograman yang digunakan untuk menerjemahkan baris kode program menjadi kode mesin yang dapat dimengerti oleh komputer yang berbasis *server-side* yang dapat ditambahkan ke dalam HTML. Sedangkan, menurut Solichin (2016:11) mengemukakan bahwa PHP merupakan salah satu bahasa pemrograman berbasis *web* yang ditulis oleh dan untuk pengembang *web*.

Salah satu keunggulan yang dimiliki oleh PHP adalah kemampuannya untuk melakukan koneksi ke berbagai macam *software* sistem manajemen basis data atau *Database Management System* (DBMS), sehingga dapat menciptakan suatu halaman *web* dinamis.

Untuk mengenal sintak dalam PHP, tentu saja juga harus sudah mengenal algoritma dalam pemrograman, karena pada dasarnya semua bahasa pemrograman menggunakan algoritma. Contoh pengenalan *variable* didalam PHP dapat dilihat pada Tabel 1.

Tabel 2. Sintak *Variable* pada PHP

No	Sintak	Keterangan
1	<?	awal sintak php
2	<i>\$nama</i> ="Cen";	<i>variable</i> nama bernilai string Cen
3	<i>\$usia</i> ="22";	<i>variable</i> usia bernilai integer 2
4	<i>echo</i> "Namaku <i>\$nama</i> , dan usiaku <i>\$usia</i> ";	menampilkan nilai dari <i>variable</i> <i>\$nama</i> dan <i>\$usia</i>
5	?>	akhir sintak php

6) *Java Script*

Javascript menurut Efendi (2015) merupakan bahasa pemrograman komputer yang dinamis. Biasanya sering digunakan pada *webbrowser* untuk menciptakan halaman *web* yang menarik, interaktif, serta merapkan berbagai fungsi pada halaman *web*. *Javascript* merupakan salah satu pemrograman *web* yang harus kita pelajari (selain HTML dan CSS).

Java Script merupakan bahasa pemrograman untuk memberikan kemampuan tambahan kedalam bahasa pemrograman HTML (*Hypertext Markup Language*) dengan mengizinkan pengeksekusian perintah-perintah pada sisi *client*, dan bukan sisi *server* dokumen *web*. Pada hakikatnya, bahasa pemrograman *Java Script* berisi Skrip yang pemasangannya terselip disebuah dokumen HTML. Sehingga bahasa *Java Script* ini tidaklah memerlukan sebuah *kompilator* atau penerjemah khusus untuk mengeksekusinya. Hal tersebut juga bergantung pada *navigator* yang terdapat disetiap *browser*.

Untuk menerapkan program *Java Script* dalam *web*, ada beberapa yang harus diketahui oleh seorang perancang *web*, yaitu:

- 1) Seorang perancang harus mengetahui cara menggunakan HTML dan mengedit dokumen HTML.
- 2) Seorang perancang harus menggunakan *browser* yang sudah mendukung pemrograman *Java Script*, misalnya *Internet Explorer* versi 4 atau versi di atasnya, *Netscape* versi 3 atau versidi atasnya. Penggunaan *Java Script* dalam pembuatan *web* yaitu

dengan memasukan program kedalam HTML. Sintak *Script* dari *Java Script* dapat dilihat

Tabel 3. Sintak *Java Script*

No	Sintax	Keterangan
1	<code><script type="text/Java Script"></code>	Awal <i>script</i>
2	<code>var x = 100;</code>	Deklarasi <i>variable x</i> bernilai 100
3	<code>document Writeln ('variabel xbernilai =' + x + '
');</code>	Menampilkan nilai dari <i>variable x</i> yaitu 100
4	<code>var user = 'Sepri;</code>	Deklarasi awal nilai <i>variabel user</i> bernilia <i>string</i> janu
5	<code>document.writeln ('Selamat Datang')</code>	Menampilkan nilai dari <i>variable user</i> yaitu Cen
6	<code>'+user+'
');</code>	Cen
7	<code></script></code>	Akhir <i>script</i>

Variable dalam *Java Script* merupakan tempat untuk menyimpan nilai-nilai atau informasi-informasi pada *Java Script*. Contoh penggunaan *Variable* dapat dilihat pada Tabel 4.

Tabel 4 .Contoh penggunaan Tipe Data pada *Java Script*

No	Tipe data	Contoh
1	String	Var namaku=" Cen";
2	Number	Var nilaiku=100;

Simbol operator merupakan simbol yang digunakan untuk memproses nilai-nilai atau informasi-informasi pada *Java Script*.

Contoh penggunaan Operator dapat dilihat pada Tabel 5.

Tabel 5. *Operator Java Script*

No	Operator	Kegunaan	Jenis
1	=	Untuk mengisi nilai pada sebuah variabel. Misal namaku="Cen"	Operator penugasan
2		Untuk operasi logika atau	Operator logika. Operator logika digunakan untuk melakukan operasi logika yang menghasilkan nilai <i>true</i> (benar) atau <i>false</i> (salah)
3	&&	Untuk operasi perbandingan sama dengan	Operator perbandingan. Operator perbandingan digunakan untuk membandingkan suatu nilai dengan nilai yang lain, hasilnya adalah nilai <i>true</i> (benar) atau (salah).
4	!=	Untuk operasi perbandingan tidak sama dengan	Operator perbandingan
5	<, <=, >, >=	Untuk operasi perbandingan kurang dari, kurang dari sama dengan, lebih dari, lebih dari sama dengan.	Operator perbandingan
6	+, -	Penjumlahan dan pengurangan.	Operator Aritmetika

		Operator + juga bisa digunakan untuk menyambung dua buah string	
7	*, /	Perkalian dan pembagian	Operator Aritmetika

7) Konsep Dasar Pemodelan

Utomo (2010:7), Definisi pemodelan atau model adalah representasi yang disederhanakan dari suatu realitas kompleks, yang tujuannya adalah pemahaman realitas dan mempunyai seluruh fitur yang diperlukan untuk tugas atau pemecahan masalah yang diperlukan.

Untuk mendapatkan suatu definisi yang dapat dipahami dan dimengerti dengan baik dari tentang pemodelan sistem, maka perlu diketahui arti dari kedua kata tersebut, yaitu pemodelan (model) dan sistem.

Model adalah rencana, representasi atau deskripsi yang menjelaskan suatu objek, sistem, konsep yang seringkali berupa penyederhanaan atau idealisasi. Bentuknya dapat berupa model fisik seperti maket, *prototype* dan lain sebagainya, dan dapat juga berupa model citra seperti gambar, grafis, komputerisasi, atau dapat juga berupa rumusan matematis. Sedangkan sistem merupakan suatu kesatuan yang terdiri dari komponen-komponen atau elemen-elemen yang dihubungkan bersama agar memudahkan suatu aliran informasi, materi ataupun energi.

a. *Extreme Programming (XP)*

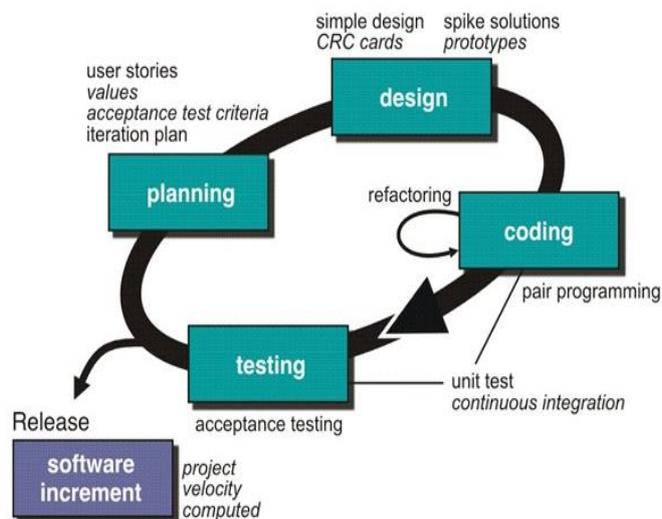
Jadi, pemodelan sistem adalah suatu bentuk penyederhanaan dari sebuah komponen dan elemen yang sangat kompleks sehingga memudahkan pemahaman dan maksud dari informasi yang dibutuhkan.

Adapun konsep pemodelan sistem yang penulis gunakan untuk membangun sistem *web mobile* ini adalah *Extreme Programming* (XP).

Menurut Pressman (2012:71), *Extreme Programming* (XP) adalah metodologi pengembangan perangkat lunak yang ditujukan untuk meningkatkan kualitas perangkat lunak dan tanggap terhadap perubahan kebutuhan pelanggan. Jenis pengembangan perangkat lunak semacam ini dimaksudkan untuk meningkatkan produktivitas dan memperkenalkan pos pemeriksaan dimana persyaratan pelanggan baru dapat diadopsi.

Unsur-unsur lain dari *Extreme Programming* meliputi *paired programming* pada tahapan *coding*, *unit testing* pada semua kode, penghindaran programan fitur kecuali benar-benar diperlukan, struktur manajemen yang datar, kode yang sederhana dan jelas, dan seringnya terjadi komunikasi antara *programmer* dan *user* ketika terjadi perubahan kebutuhan *user* seiring berlalunya waktu berlalu.

Adapun gambar pengembangan sistem perangkat lunak dengan proses *Extreme Programming* (XP) dapat dilihat pada Gambar 1.



Gambar 1. Pemodelan Metode *Extreme Programming*

1) *Planning*

Planning activities seperti kriteria pengguna dan perencanaan pengembangan dimulai dengan team atau individu mengumpulkan segala sesuatu kebutuhan yang diperlukan dalam pengembangan aplikasi. Setelah mengumpulkan kebutuhan, *team* mulai melakukan penyusunan *user stories* untuk menggambarkan *output* yang diperlukan, fitur dan kegunaan.

2) *Design*

Pada tahap ini penulis membuat sebuah desain tampilan pemodelan *system* dan desain *interface* program agar alur dari proses program tersebut jelas berdasarkan *stories* atau gambaran yang telah di dapatkan dari *client* pada saat *planning*.

3) *Coding*

Setelah melakukan *planning* dan *design* pada tahap ini penulis melakukan realisasi perancangan perangkat lunak ke penulisan *coding*

4) *Testing*

Pada tahap akhir ini dimana pihak *client* yaitu pengguna sistem melakukan pengujian terhadap program yang telah selesai, perubahan terjadi dalam program dengan komunikasi antara *programmer* dan *client* jika terdapat kesalahan atau penambahan oleh permintaan *user* tersebut.

Di dalam *testing* dilakukan Pengujian dengan metode *Black-Box* merupakan pengujian perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program. Sebuah perangkat lunak yang diuji menggunakan metode *Black-Box* dikatakan berhasil jika fungsi-fungsi yang ada telah memenuhi spesifikasi kebutuhan yang telah dibuat sebelumnya.

Menurut Shalahuddin dan Rosa (2011), black box testing adalah menguji perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program. Pengujian dimaksudkan untuk mengetahui apakah fungsi-fungsi, masukan, dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan. Pengujian kotak hitam dilakukan dengan membuat kasus uji

yang bersifat mencoba semua fungsi dengan memakai perangkat lunak apakah sesuai

b. *Unified Modeling Language (UML)*

Mulyani (2016:48) mengatakan UML (Unified Modeling Language) adalah sebuah teknik pengembangan sistem yang menggunakan bahasa grafis sebagai alat untuk pendokumentasian dan melakukan spesifikasi pada sistem.

Dengan menggunakan UML lebih mudah dalam membuat model untuk semua jenis aplikasi piranti lunak, aplikasi tersebut dapat berjalan pada piranti keras, sistem operasi dan jaringan apapun, serta ditulis dalam bahasa pemrograman apapun.

Seperti bahasa-bahasa lainnya, UML mendefinisikan notasi dan *syntax*/semantik. Notasi UML merupakan sekumpulan bentuk khusus untuk menggambarkan berbagai diagram piranti lunak. Setiap bentuk memiliki makna tertentu, dan UML *syntax* mendefinisikan bentuk-bentuk UML dapat dikombinasikan.

Dalam UML ada beberapa diagram yang digunakan untuk memodelkan aplikasi piranti lunak, diagram-diagram sebagai berikut:

c. *Use Case Diagram*

Use Case diagram merupakan pemodelan untuk sistem informasi yang akan dibuat. Use Case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang

akan dibuat. Use Case digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi–fungsi itu. Notasi – notasi yang dipakai dalam Use Case ini dapat dilihat pada Tabel6.

Tabel 6. Simbol *Use Case Diagram*

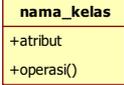
Simbol	Deskripsi
<p><i>Use case</i></p>  <p>nama use case</p>	<p>Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama <i>usecase</i>.</p>
<p>Aktor/<i>Actor</i></p>  <p>nama aktor</p>	<p>Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama aktor.</p>
<p>Asosiasi/<i>Association</i></p> 	<p>Komunikasi antar aktor dan <i>usecase</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.</p>
<p>Ekstensi/<i>Extend</i></p> <p><<extend>></p> 	<p>Relasi <i>usecase</i> tambahan ke sebuah <i>usecase</i> yang ditambahkan dapat berdiri sendiri walaupun tanpa <i>use case</i> tambahan.</p>
<p>Generalisasi/<i>Generalization</i></p> 	<p>Hubungan generalisasi dan spesialisasi (umum – khusus) antara dua buah <i>use case</i>, fungsi yang satu adalah fungsi yang lebih umum dari lainnya.</p>

<p>Menggunakan/<i>Include/Uses</i></p> <p style="text-align: center;"><<include>></p> 	<p>Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> di mana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini.</p>
---	--

a) *Class Diagram*

Class diagram menggambarkan struktur dan deskripsi *class*, *package* dan objek beserta hubungan satu sama lain seperti *containment*, pewarisan, asosiasi, dan lain-lain. Tabel 7 adalah simbol-simbol yang ada pada diagram kelas. Tabel 7.

Tabel 7. Simbol Class Diagram

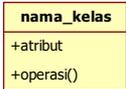
Simbol	Deskripsi
<p>Kelas</p> 	<p>Kelas pada struktur sistem.</p>
<p>Antar Muka/<i>Interface</i></p> 	<p>Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek.</p>
<p>Asosiasi/<i>Association</i></p> 	<p>Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i>.</p>
<p>Generalisasi</p> 	<p>Relasi antar kelas dengan makna generalisasi-spesialisi (umum khusus).</p>

Kebergantungan/ <i>Dependency</i> 	Relasi antar kelas dengan makna kebergantungan antar kelas.
Agregasi/ <i>Aggregation</i> 	Relasi antar kelas dengan makna semua-bagian (<i>whole-part</i>).

b) Class Diagram

Class diagram menggambarkan struktur dan deskripsi *class*, *package* dan objek beserta hubungan satu sama lain seperti *containment*, pewarisan, asosiasi, dan lain-lain. Tabel 8 adalah simbol-simbol yang ada pada diagram kelas Tabel 8.

Tabel 8. Simbol Class Diagram

Simbol	Deskripsi
<p>Kelas</p> 	Kelas pada struktur sistem.
<p>Antar Muka/<i>Interface</i></p>  <p>nama_interface</p>	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek.
<p>Asosiasi/<i>Association</i></p> 	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .

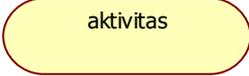
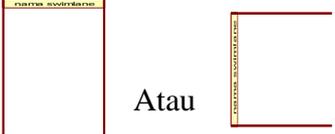
<p>Asosiasi Berarah/<i>Directed Association</i></p> 	<p>Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i>.</p>
<p>Generalisasi</p> 	<p>Relasi antar kelas dengan makna generalisasi-spesialisasi (umum khusus).</p>
<p>Kebergantungan/<i>Dependency</i></p> 	<p>Relasi antar kelas dengan makna kebergantungan antar kelas.</p>
<p>Agregasi/<i>Aggregation</i></p> 	<p>Relasi antar kelas dengan makna semua-bagian (<i>whole-part</i>).</p>

c) *Activity Diagram*

Activity diagrams menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, mengenai masing-masing alir berawal, *decision* yang mungkin terjadi, dan berakhir. *Activity diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi. Simbol *activity* dapat dilihat pada Tabel 9.

Tabel 9. Simbol Activity Diagram

Simbol	Deskripsi
--------	-----------

<p>Status Awal</p> 	Status awal aktivitas sistem, sebuah activity diagram memiliki sebuah status awal.
<p>Aktivitas</p> 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
<p>Percabangan/<i>Decision</i></p> 	Asosiasi percabangan jika ada pilihan aktivitas lebih dari satu.
<p>Status Akhir</p> 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
<p><i>Swimlane</i></p> 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.

d) *Sequence Diagram*

Sequence diagram menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna, *display*, dan sebagainya) berupa *message* yang digambarkan terhadap waktu.

Tabel 9 adalah simbol-simbol yang ada pada *sequence* diagram

Tabel 10. Simbol *Sequence Diagram*

Simbol	Deskripsi
<i>Actor</i>	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang

	<p>akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah orang tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama aktor.</p>
<p>Garis Hidup/<i>Life Line</i></p> 	<p><i>Lifeline</i> mengindikasikan keberadaan sebuah <i>object</i> dalam basis waktu. Notasi untuk <i>Lifeline</i> adalah garis putus-putus <i>vertikal</i> yang ditarik dari sebuah <i>object</i>.</p>
<p><i>Object</i></p> 	<p><i>Object</i> merupakan <i>instance</i> dari sebuah <i>class</i> dan dituliskan tersusun secara <i>horizontal</i>. Digambarkan sebagai sebuah <i>class</i> (kotak) dengan nama <i>object</i> didalamnya yang diawali dengan sebuah titik koma.</p>
<p><i>Activation</i></p> 	<p><i>Activation</i> dinotasikan sebagai sebuah kotak segi empat yang digambar pada sebuah <i>lifeline</i>. mengindikasikan sebuah obyek yang akan melakukan sebuah aksi.</p>
<p><i>Message</i></p> 	<p><i>Message</i>, digambarkan dengan anak panah <i>horizontal</i> antara <i>Activation Message</i> mengindikasikan komunikasi antara <i>object - object</i>.</p>

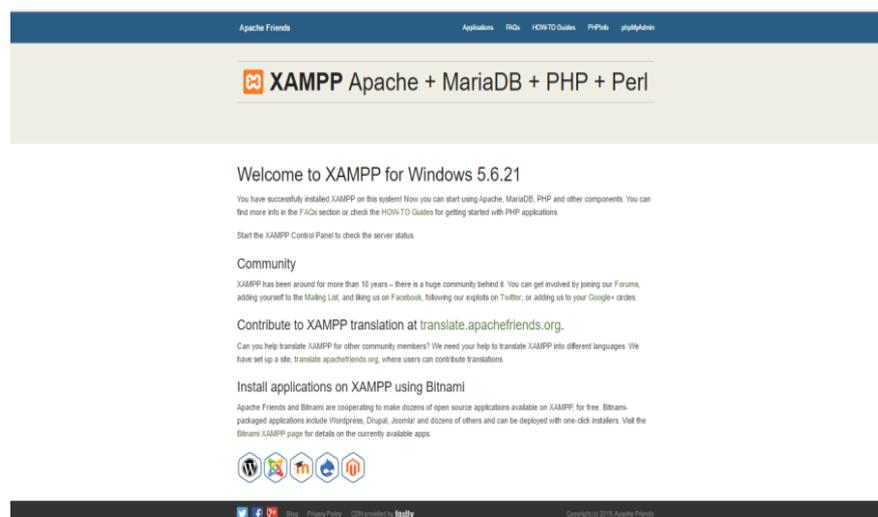
8) Perangkat Lunak Yang Di Gunakan

- a) Xampp

Xampp merupakan aplikasi web server yang banyak dipakai oleh pengembang website, terdiri dari Apache web server, MySQL, PHP, Perl, FTP Server dan phpMyAdmin (Supono dan Putratama, 2016:7).

XAMPP mengkombinasikan beberapa paket perangkat lunak yang berbeda ke dalam satu paket yaitu *Apache (web server)*, *MySQL (database)*, *PHP (server side scripting)* XAMPP juga merupakan sebuah *software web server Apache* yang didalamnya sudah tersedia *database server MySQL* dan *support PHP programming*. XAMPP merupakan *software* yang mudah digunakan, gratis dan mendukung instalasi di *Linux* dan *Windows*.

XAMPP merupakan paket program yang didalamnya terdapat *Apache* sebagai *web server*, *PHP* sebagai bahasa program dan *MySQL* sebagai *database server*. Gambar 2 adalah tampilan dari aplikasi XAMPP.



Gambar 2. Xampp

b) Notepad++

Notepad++ sebenarnya evolusi dari editor Notepad dan juga dapat dijadikan web editor yang berfungsi sebagai pengolah bahasa pemrograman berbasis web seperti PHP, HTML, CSS, JQuery, dan JavaScript.

Menurut Supono dan Putratama (2016:13) menyatakan bahwa Notepad++ merupakan aplikasi teks editor yang gratis serta powerfull yang dapat digunakan oleh seorang pengembang aplikasi (programmer) untuk menuliskan sebuah kode-kode program.

c) *Browser*

Web browser adalah sebuah perangkat lunak yang memungkinkan pengguna untuk menampilkan dan berinteraksi dengan text, gambar, and informasi yang ada pada halaman web pada sebuah di *world wide web*(WWW) atau *local area network* (LAN), menurut (Sufandi, 2007).

Begitu banyak *web browser* yang bisa digunakan, *web browser* yang paling populer yaitu *Mozilla Firefox*, *Internet Explorer*, *Google Chrome*, dan *Opera*. Pada tahap pertama merupakan perbandingan parameter yang dikategorikan menjadi fitur umum, fitur keamanan dan fitur browser. Membandingkan masing-masing *browser* dari segi besarnya pengguna yang sering menggunakan *web browser* tersebut.

9. Perencanaan penelitian

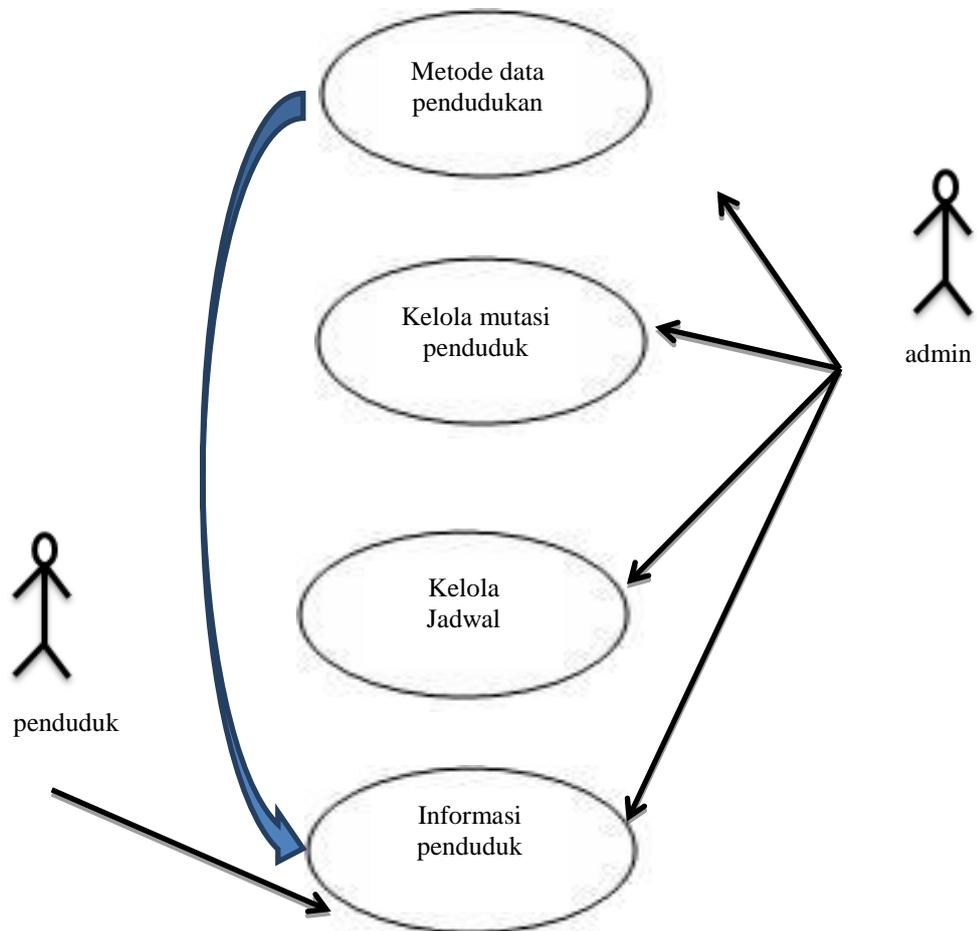
a. Analisis kebutuhan user

Adalah tahap pengumpulan data untuk membangun sistem serta mengidentifikasi kebutuhan yang diinginkan oleh user agar dapat memberikan informasi yang lebih akurat bagi pengguna.

1. Kebutuhan Input

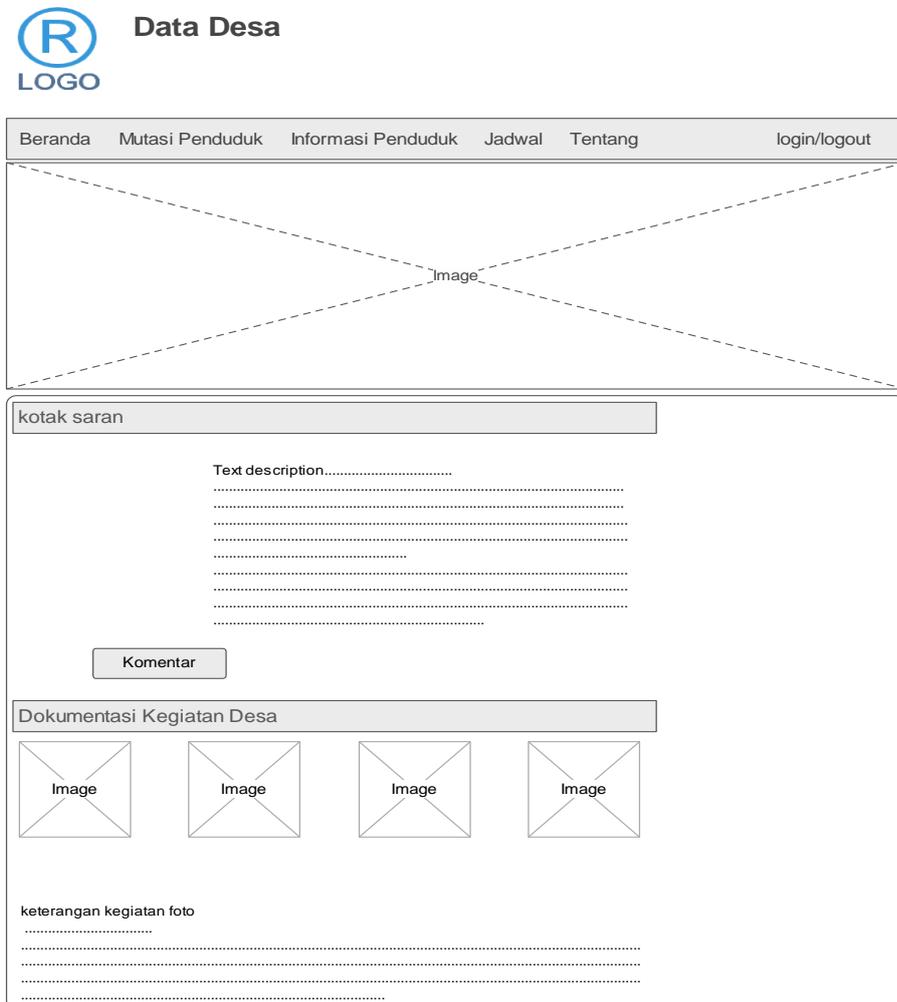
Dalam kebutuhan input data diperlukan pengumpulan data, informasi umum mengenai kependudukan yang diperoleh dari institusi terkait yaitu Data Desa Haringen, Kecamatan Dusun Timur Kabupaten, Barito Timur, Provinsi Kalimantan Tengah dan Dinas Kependudukan Dan Pencatatan

- a. Data penduduk
- b. Data kepala keluarga
- c. Data pekerjaan
- d. Data agama
- e. Data pendidikan
- f. Data pekerjaan
- g. Data status keluarga

b. Desain *System Use Case*

Gambar 3. Use Case Diagram

c. Desain Interface



Gambar 4. Desain Interface

BAB III

ANALISIS DAN PERANCANGAN SISTEM

A. Tinjauan Umum

Istilah sistem paling sering digunakan untuk menunjuk pengertian metode atau cara dan sesuatu himpunan unsur atau komponen yang saling terhubung satu sama lain menjadi satu kesatuan yang utuh.

Informasi adalah data yang telah diolah menjadi bentuk yang telah berarti bagi yang menerimanya. Informasi dapat dipahami sehingga pemrosesan input yang terorganisir, memiliki arti, dan berguna bagi orang yang menerimanya.

Sistem informasi dapat diidentifikasi sebagai suatu sistem didalam sebagai suatu organisasi yang merupakan kombinasi dari orang-orang, fasilitas, teknologi, media prosedur-prosedur dan pengendalian yang ditujukan untuk mendapatkan jalur komunikasi penting, memproses tipe transaksi rutin tertentu, memberi sinyal kepada manajemen dan yang lainnya terhadap kajian-kajian internal dan eksternal yang penting dan menyediakan suatu dasar informasi untuk pengambilan keputusan.

Website atau situs dapat diartikan sebagai kumpulan halaman yang menampilkan informasi data teks, data gambar diam ataupun bergerak, data animasi, suara, video atau gabungan dari semuanya, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling

terkait dimana masing-masing dihubungkan dengan jaringan-jaringan halaman (Hyperlink).

Agar permasalahan dalam penelitian ini tidak melebar luar, maka diberikan batasan-batasan untuk pendekatan permasalahan agar lebih terinci dalam pelaksanaannya, yaitu:

1. Pendataan hanya dilakukan di Desa Haringen Barito Timur saja.
2. Sistem yang dibangun meliputi pendataan penduduk yang ada di Desa Haringen Barito Timur, pindah domisili, blangko pendaftaran.
3. Data yang di olah berbasis *web responsive*.
4. Pembuatan *system* informasi di buat dengan menggunakan *php*, *databaseMySQL*, *HTML* dan *Bootsrap*.

B. Analisis

a. Analisis Kelemahan Sistem

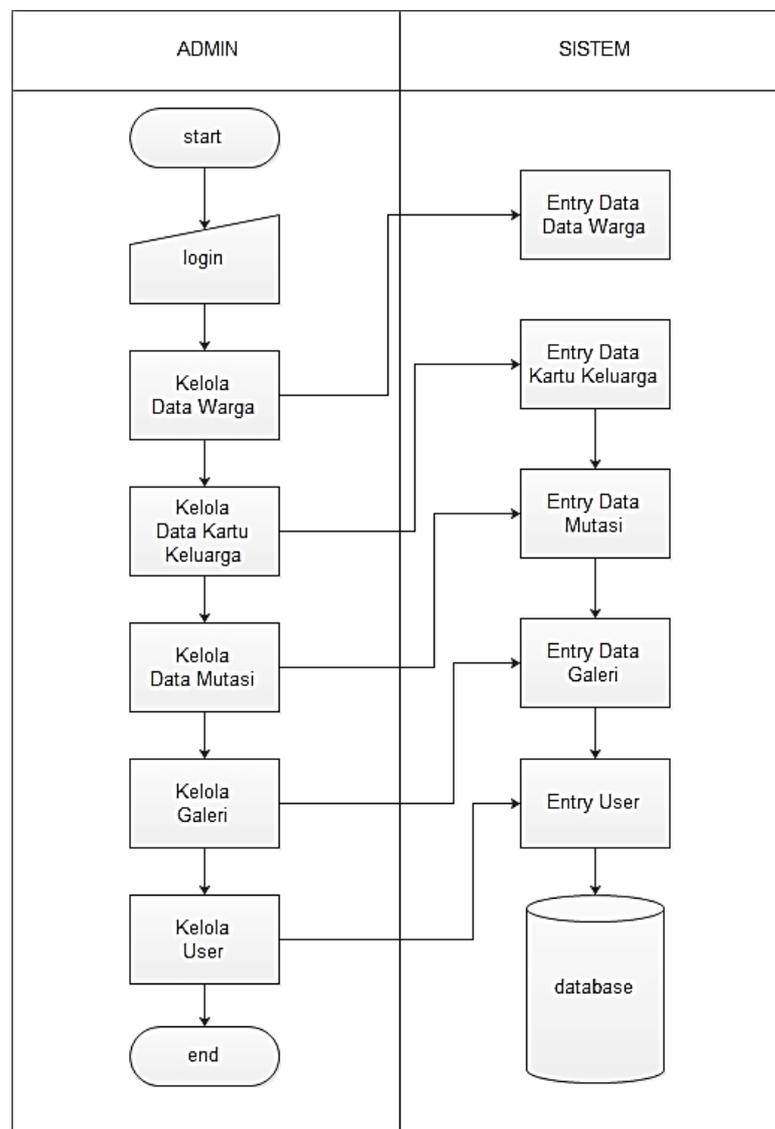
Sistem yang sedang berjalan saat ini dilakukan secara manual, dimana pendataan penduduk Desa Haringen Barito Timur dilakukan secara konvensional yaitu sekretaris desa mencatat data penduduk desa berdasarkan data dari setiap warga, kartu keluarga dan data warga yang mutasi.

b. Analisis Kebutuhan

Aplikasi *Web Monitoring* Pendataan merupakan teknologi yang dapat menampung pendataan dengan jumlah yang banyak. Dengan kelebihan tersebut aplikasi ini dapat diimplementasikan juga sebagai media pendukung dalam kelancaran proses pendataan penduduk. Hal

ini yang menarik peneliti untuk melakukan penelitian mengenai perancangan aplikasi *web monitoring* pendataan penduduk.

Dengan teknologi ini, setiap penduduk diharapkan tidak perlu lagi untuk mengisi daftar secara manual dengan tulis tangan setiap data yang dibutuhkan karena datanya akan di input oleh admin dan tersimpan didalam database sistem.



Gambar 5. Analisis Sistem Baru

A. Kebutuhan Perangkat Keras

Perangkat keras (hardware) yang digunakan dengan spesifikasi sebagai berikut :

- 1.) 1 Unit Laptop Asus
- 2.) *Processor Intel(R) Core(TM) i5-5200U CPU @ 2.20GHz (4 CPUs), ~2.2GHz*
- 3.) *OS Windows 10 Pro 64-bit*

B. Kebutuhan Perangkat Lunak

Perangkat lunak (software) yang digunakan adalah sebagai berikut :

- 1.) PHP sebagai bahasa pemrograman script server-side yang didesain untuk pengembangan web.
- 2.) Database MySQL sebagai wadah yang menyimpan string (text based) data.
- 3.) HTML untuk membuat sebuah halaman web, yang kemudian dapat diakses untuk menampilkan berbagai informasi di dalam sebuah penjelajah web internet
- 4.) Xampp web server sebagai servernya.
- 5.) Cpanel sebagai salah satu control panel web hosting

C. Kebutuhan Informasi

Proses analisa informasi merupakan analisa yang dimaksudkan untuk mengetahui sejauh mana analisa informasi dari sistem baru website ini. Adapun beberapa informasi mengenai analisa penelitian dan

perancangan serta dalam hal implementasi yang dapat ditemukan di dalam sistem ini adalah :

1. Halaman Login

Sebelum memasuki halaman beranda utama, pengguna harus terlebih dahulu melakukan proses login, dan memasukkan username dan password.

2. Halaman Beranda

Halaman Beranda merupakan halaman awal website yang akan ditampilkan saat setelah berhasil login.

3. Halaman Kelola Data Warga

Halaman Kelola Akun merupakan halaman informasi data warga, dimana informasi data warga di kelola admin.

4. Halaman Kelola Data Kartu Keluarga

Halaman kelola data kartu keluarga merupakan halaman informasi data mengenai identitas penduduk berdasarkan kartu keluarga yang di kelola admin.

5. Halaman Kelola Data Mutasi

Halaman kelola data mutasi merupakan halaman yang berisi informasi tentang data mutasi penduduk.

6. Halaman Kelola Galeri

Halaman kelola galeri merupakan halaman yang berisi tentang data foto yang telah tersimpan dan dikelola oleh admin.

7. Halaman Kelola Akun

Halaman kelola akun merupakan halaman yang berisi tentang hak akses untuk user, yang nantinya akan dikelola oleh admin.

8. Menu Logout

Pada Menu Logout ini digunakan untuk keluar dari halaman website.

D. Kebutuhan Pengguna (User)

Pada tahap ini, ditentukan pengguna yang ada pada sistem website dan aplikasi ini terdiri dari 1 (satu) yaitu Admin.

E. Analisis Kelayakan Sistem

i. Kelayakan Teknologi

Kelayakan teknologi berhubungan dengan penerapan sistem teknologi yang ada. Dari segi hardware, Kantor Desa Haringen sudah memiliki sebuah komputer sehingga hanya dibutuhkan sebuah printer untuk melengkapi kebutuhan hardware sistem yang baru. Dari segi software Kantor Desa Haringen belum menggunakan aplikasi khusus untuk mengolah data penduduk. Sedangkan dari segi brainware, sudah ada beberapa perangkat desa yang bisa mengoperasikan komputer. Jadi sistem baru dikatakan layak untuk dioperasikan dan tidak ada hambatan dari segi teknologi.

ii. Kelayakan Hukum

Evaluasi kelayakan hukum menilai apakah aplikasi sistem ini layak dioperasikan tanpa bertentangan dengan batasan hukum yang berlaku. Hal ini penting karena adakalanya suatu sistem informasi memerlukan

beberapa komponen yang untuk pengadaannya memerlukan pertimbangan hukum terlebih dahulu atau bahkan bertentangan dengan hukum sehingga teknologi tersebut tidak dapat diterapkan atau perlu diganti.

Salah satunya, evaluasi pada lisensi aplikasi yang mendukung sistem informasi yang sedang dijalankan. Salah satunya server database yang berbayar yaitu Microsoft SQL server. Selain itu juga, pengguna aplikasi berlisensi pada sistem operasi yang dipakai yaitu Windows 7.

iii. Kelayakan Operasional

Evaluasi kelayakan operasional menilai apakah aplikasi sistem aplikasi dapat dikerjakan dan berhasil dan apakah sistem sedang atau telah dipakai.

Salah satu evaluasi pada kinerja sistem informasi, adalah pada saat pemakaiannya, mencegah banyak pemakai yang bingung bagaimana cara menggunakan aplikasi tersebut, sehingga dibuatlah *interface*/tampilan aplikasi yang memudahkan pengguna dalam pemakaiannya.

iv. Kelayakan Ekonomi

Aspek paling dominan dari empat aspek kelayakan sistem adalah kelayakan ekonomi. Tidak bisa dipungkiri bahwa motif instansi untuk membangun sebuah sistem adalah demi keuntungan. Dengan demikian aspek untung rugi menjadi pertimbangan utama dalam pengembangan sistem.

F. Desain Sistem

1.) Desain Proses

A .)Pemodelan Sistem *Unified Modelling Language* (UML)

Unified Modelling Language (UML) adalah bahasa pemodelan untuk sistem atau perangkat lunak yang berparadigma berorientasi objek. Abstraksi konsep dasar UML terdiri dari *structural classification, dynamic behavior*, dan model *management* dapat kita pahami *main concepts* sebagai *term* yang akan muncul pada saat membuat diagram dan *view* adalah kategori dari diagram tersebut. UML mendefinisikan diagram-diagram sebagai *Use case diagram, Class diagram, Statechart diagram, Activity diagram, Sequence diagram, Collaboration diagram, Component diagram, dan Deployment diagram* (Sutejo, 2016).

2 .)UseCase Diagram

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

1. Deskripsi Aktor

Berikut adalah deskripsi dari aktor yang berinteraksi langsung dengan sistem :

Tabel 11 Deskripsi Aktor

No.	Aktor	Deskripsi
1.	Admin	Orang yang dapat mengakses <i>website</i> , melakukan <i>login</i> , melakukan aksi seperti kelola data warga, data kartu keluarga, data mutasi, galeri, <i>user</i> dan <i>login</i> .

2. Deskripsi Use Case Admin

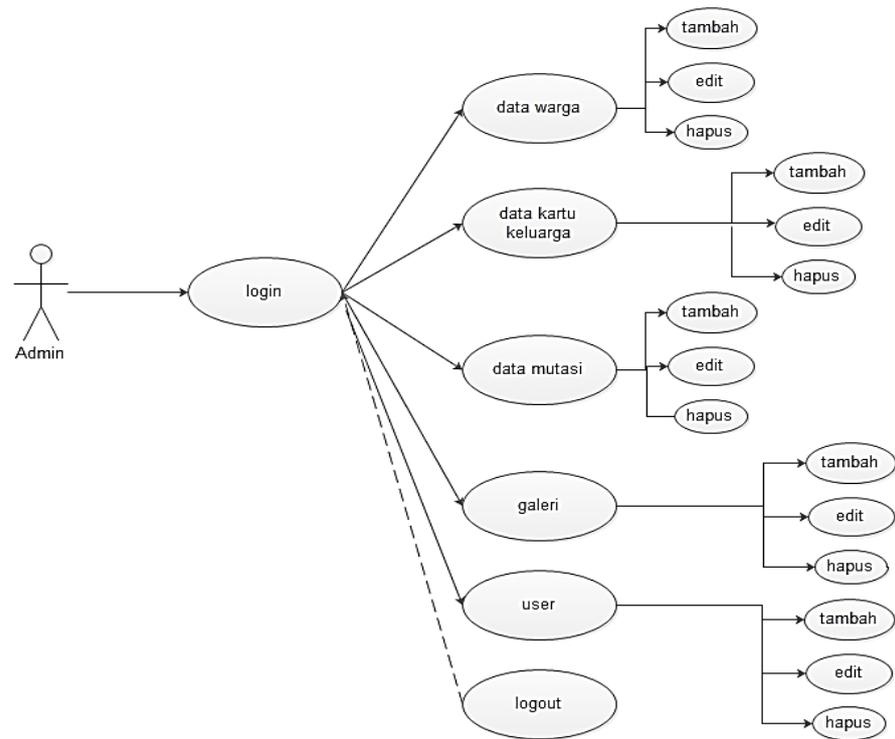
Berikut ini adalah deksripsi pendefinisian *use case* sesuai peran dari aktor admin pada sistem ini.

Tabel 12 Deskripsi Use Case Admin

No.	Use Case	Deskripsi
1.	Login	Untuk dapat masuk ke dalam sistem, admin harus melakukan <i>login</i> terlebih dahulu
2.	Pilih Menu Data Warga	<i>Pre-Condition</i> : Admin ingin memilih menu Data Warga. <i>Proses</i> : Menampilkan menu data warga <i>Post-Condition</i> : Admin telah memilih menu data warga
3.	Pilih Menu Data Kartu Keluarga	<i>Pre-Condition</i> : Admin ingin memilih menu Kartu Keluarga. <i>Proses</i> : Menampilkan menu Kartu Keluarga <i>Post-Condition</i> : Admin telah memilih menu Kartu Keluarga

4.	Pilih Menu Data Mutasi	<i>Pre-Condition :</i> Admin ingin memilih menu Data Mutasi. Proses : Menampilkan menu Data Mutasi <i>Post-Condition :</i> Admin telah memilih menu Data Mutasi
4.	Pilih Menu Galeri	<i>Pre-Condition :</i> Admin ingin memilih menu Galeri. Proses : Menampilkan menu Galeri <i>Post-Condition :</i> Admin telah memilih menu Galeri
5	Pilih Menu User	<i>Pre-Condition :</i> Admin ingin memilih menu User. Proses : Menampilkan menu User <i>Post-Condition :</i> Admin telah memilih menu User

Use case diagram aplikasi sistem ini akan mempresentasikan interaksi yang dilakukan antara aktor Admin dengan sistem yang dapat dilihat pada Gambar 6.



Gambar 6. Use Case Admin

Berdasarkan Gambar 6 maka dapat dijelaskan sebagai berikut :

1. Aktor

Aktor yang digambarkan adalah Admin yang merupakan pengguna sistem yang memiliki hak akses terhadap semua menu yang tersedia di sistem ini.

2. Kegiatan – kegiatan aktor Admin adalah sebagai berikut :

- a. Memilih menu *Login*, yaitu kegiatan yang dilakukan aktor Admin jika ingin menggunakan semua fitur yang ada pada sistem.
- b. Memilih menu Data Warga, yaitu kegiatan yang dilakukan aktor admin jika ingin memilih menu akun.

- c. Memilih menu Data Kartu Keluarga, yaitu kegiatan yang dilakukan aktor admin jika ingin memilih menu Data Kartu Keluarga.
- d. Memilih menu Data Mutasi, yaitu kegiatan yang dilakukan aktor admin jika ingin memilih menu Data Mutasi.
- e. Memilih menu Galeri, yaitu kegiatan yang dilakukan aktor admin jika ingin memilih menu Galeri.
- f. Memilih menu User, yaitu kegiatan yang dilakukan aktor admin jika ingin memilih menu User.
- g. Memilih menu *Logout*, yaitu kegiatan yang dilakukan aktor Admin jika ingin keluar dari sistem.

c. Activity Diagram

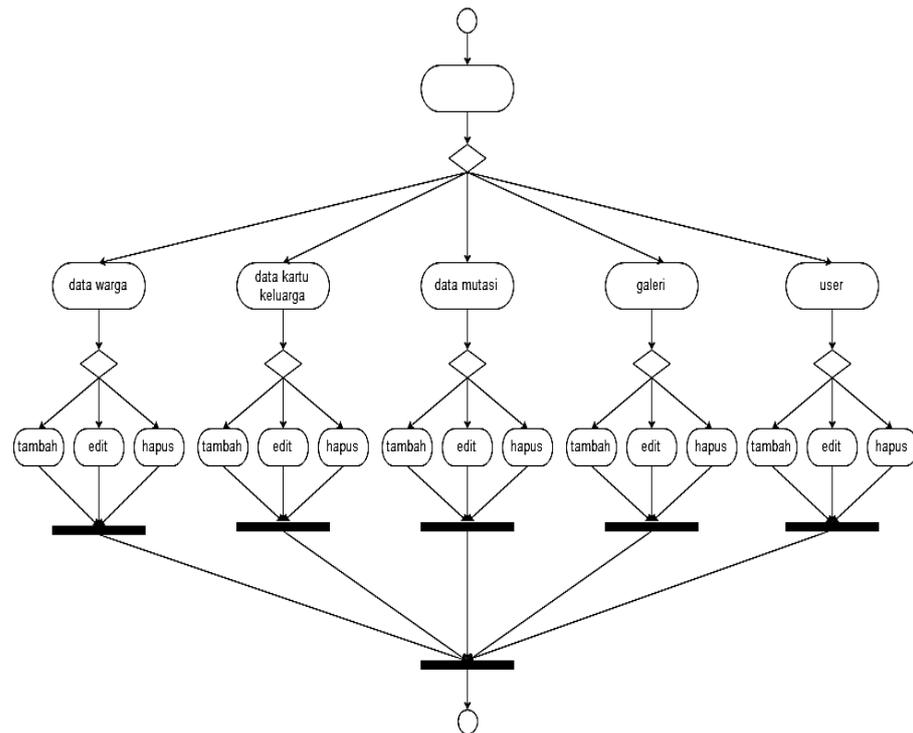
Activity diagram atau diagram aktivitas menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. *Activity diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi. *Activity diagram* merupakan *state diagram* khusus, dimana sebagian besar *state* adalah *action* dan sebagian besar transisi di *trigger* oleh selesainya *state* sebelumnya (*internal processing*). Oleh karena itu, *activity diagram* tidak menggambarkan *behaviour* internal sebuah sistem secara eksak, tetapi lebih menggambarkan proses – proses dan jalur – jalur

aktivitas dari level atas secara umum. Menggambarkan proses bisnis dan urutan aktivitas – aktivitas dalam sebuah proses.

Berikut adalah gambaran perilaku sistem untuk setiap aktivitas pengguna :

a. Activity Diagram Admin

Activity diagram Admin akan menggambarkan aktivitas – aktivitas yang dapat dikerjakan oleh Admin. Penggambarannya dapat dilihat pada Gambar 3.3.



Gambar 7. Activity Diagram Admin

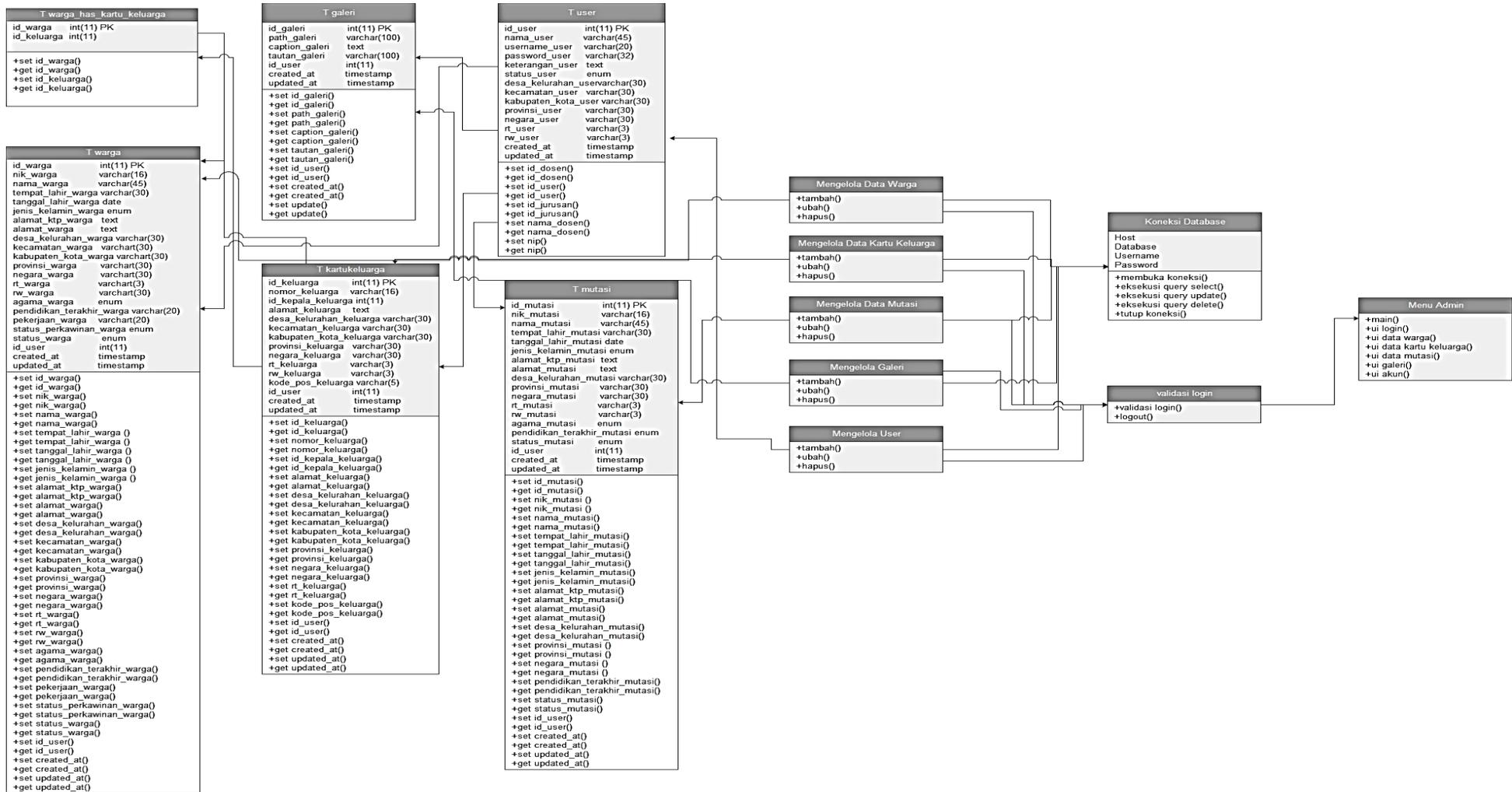
Gambar 7 menjelaskan aktivitas – aktivitas yang dilakukan oleh Admin yang dimulai dengan sebuah *start* dimana Admin dapat memilih menu, diantaranya mengelola Data Warga,

Data Kartu Keluarga, Data Mutasi, Galeri, User dan *Logout* untuk tidak melanjutkan aktivitas dari sistem.

d. **Class Diagram**

Class diagram menggambarkan struktur sistem dari segi pendefinisian kelas – kelas yang akan dibuat untuk membangun sistem. Struktur sistem dapat melakukan fungsi – fungsi dengan kebutuhan sistem dalam program perangkat lunak dan sesuai dengan perancangan kelas diagram yang telah dibuat.

Class adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. Class Diagram menggambarkan keadaan (atribut/property) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metode/fungsi). Class diagram menggambarkan struktur dan deskripsi class, package dan objek beserta hubungan satu sama lain. Pemodelan dalam pembuatan Class Diagram yaitu menentukan kelas, atribut dan hubungan antar kelas, pada tahap ini belum menentukan method (fungsi) karena akan dilakukan pada tahap desain. Salah satu cara penentuan kelas adalah mencari kandidat kelas yang berasal dari use case.



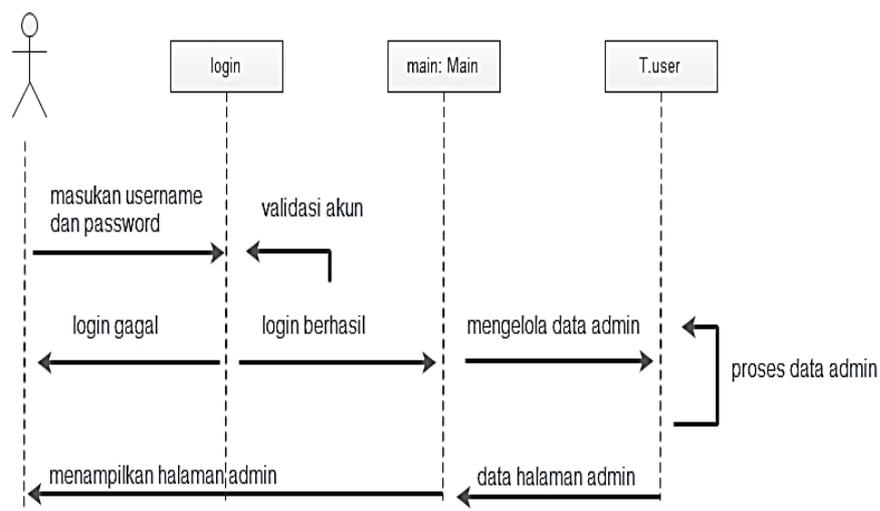
Gambar 8. Class Diagram

e. Sequence Diagram

Sequence diagram menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna, display, dan sebagainya) berupa message yang digambarkan terhadap waktu. Pada sequence diagram menjabarkannya menurut apa yang dikerjakan oleh actor, yaitu admin.

1. Sequence Diagram Admin Login

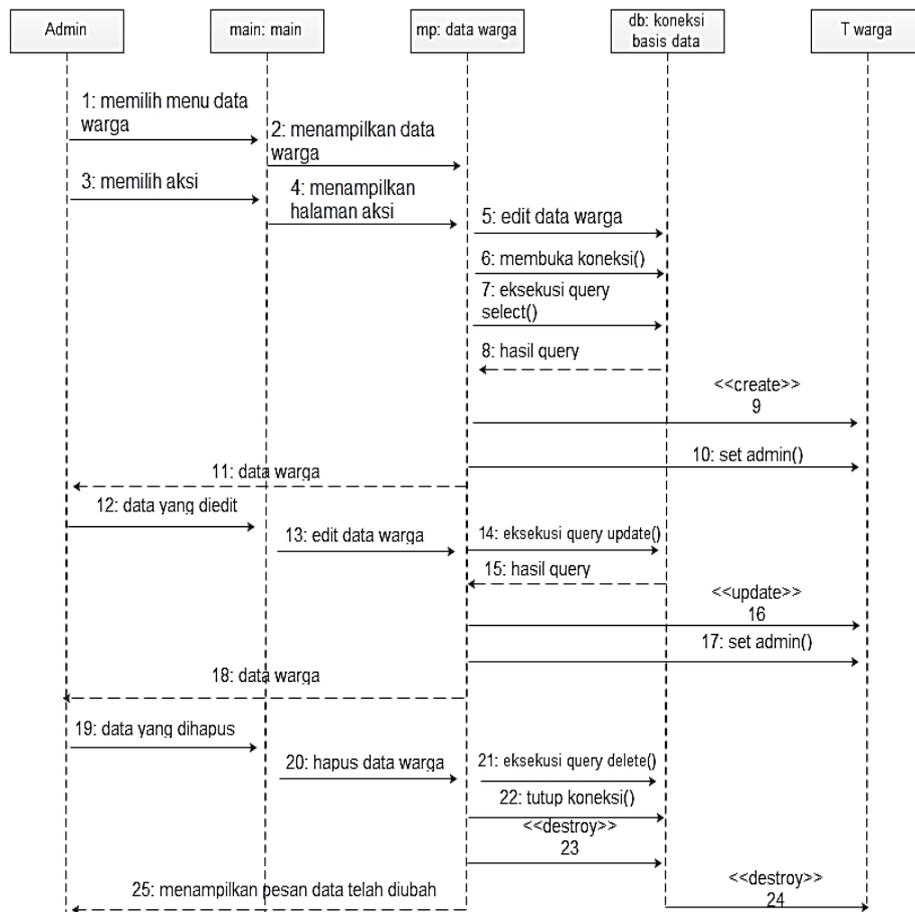
Pada sequence diagram, menggambarkan proses dimana admin melakukan login dengan memasukan username dan password. Jika berhasil, maka sistem akan melakukan proses untuk menampilkan halaman admin. Dimana proses tersebut terhubung ke tabel user.



Gambar 9. Sequence Diagram Admin Login

2. Sequence Diagram Admin Kelola Data Warga

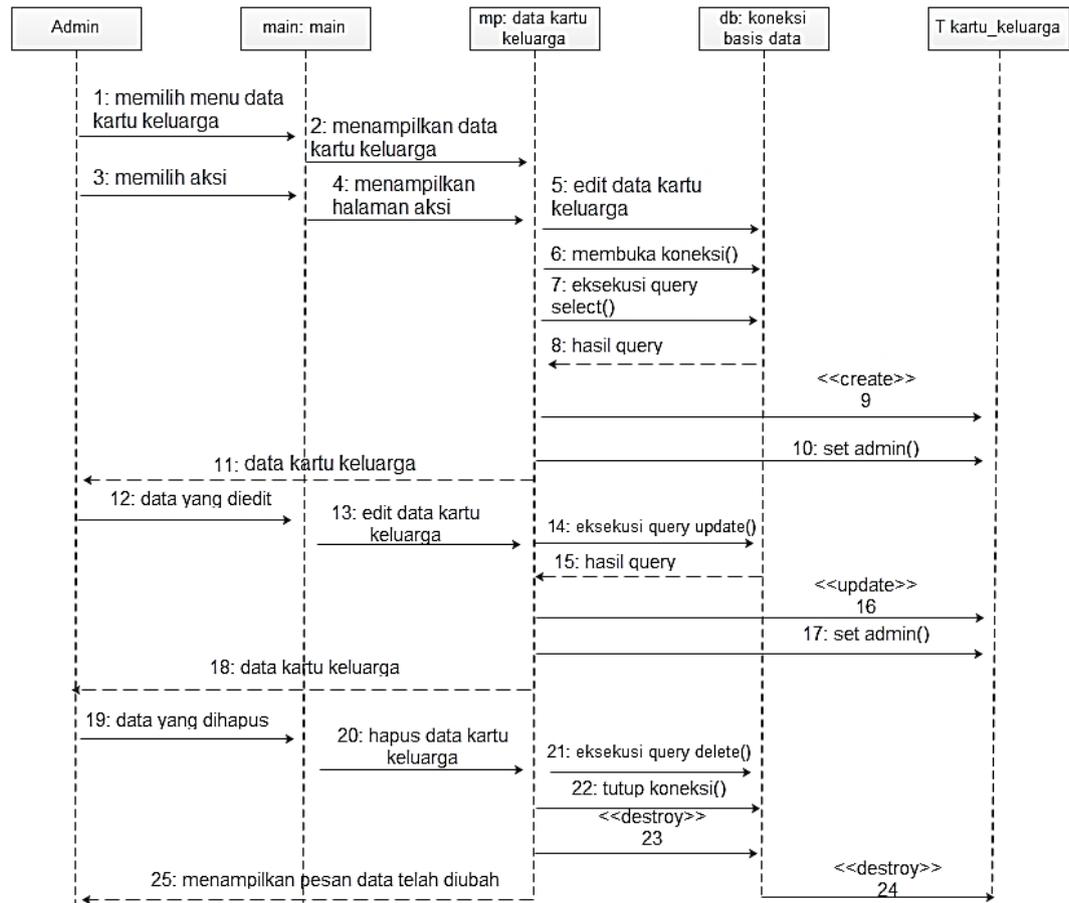
Pada sequence diagram tersebut, menggambarkan proses dimana admin melakukan kelola data warga. Jika proses kelola data warga berhasil, maka sistem akan melakukan proses untuk menampilkan halaman admin. Dimana proses tersebut terhubung ke tabel warga.



Gambar 10. Sequence Diagram Admin Kelola Data Warga

3. Sequence Diagram Admin Kelola Data Kartu Keluarga

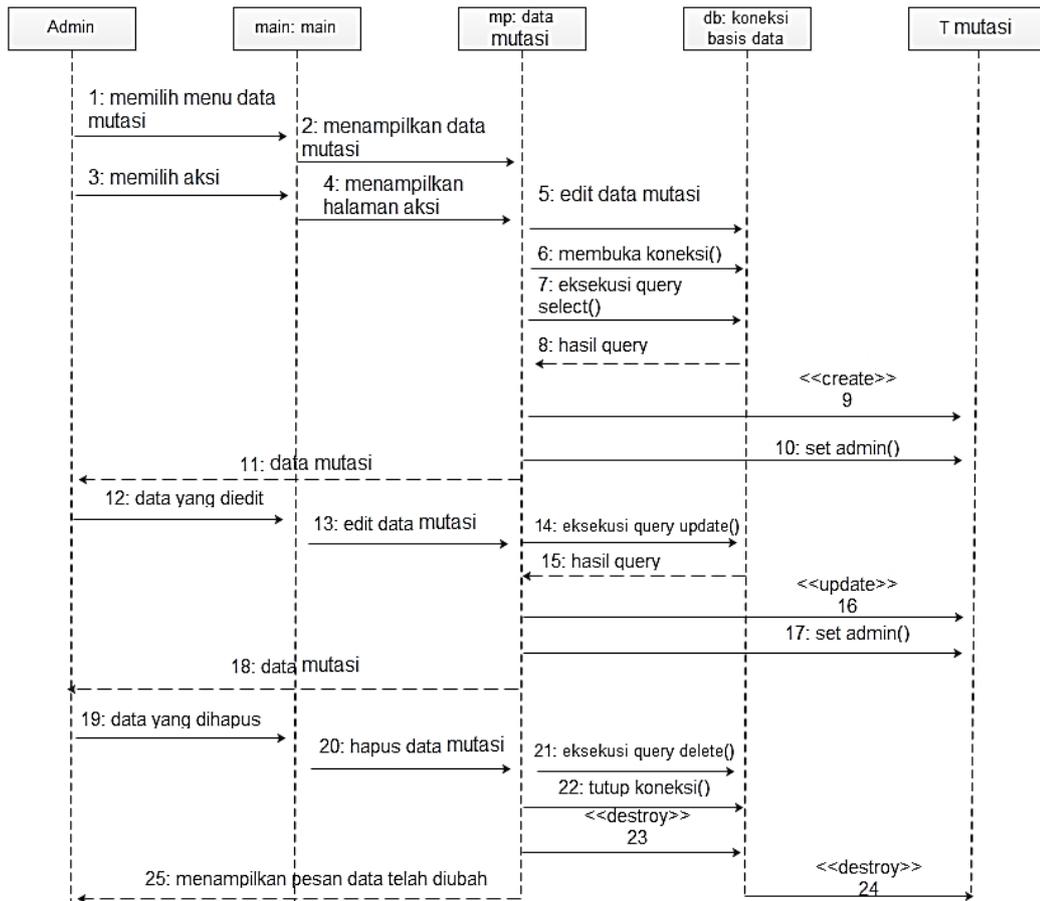
Pada sequence diagram tersebut, menggambarkan proses dimana admin melakukan kelola data kartu keluarga. Jika proses kelola data kartu keluarga berhasil, maka sistem akan melakukan proses untuk menampilkan halaman admin. Dimana proses tersebut terhubung ke tabel `kartu_keluarga`.



Gambar 11. Sequence Diagram Admin Kelola Data Kartu Keluarga

4. Sequence Diagram Admin Kelola Data Mutasi

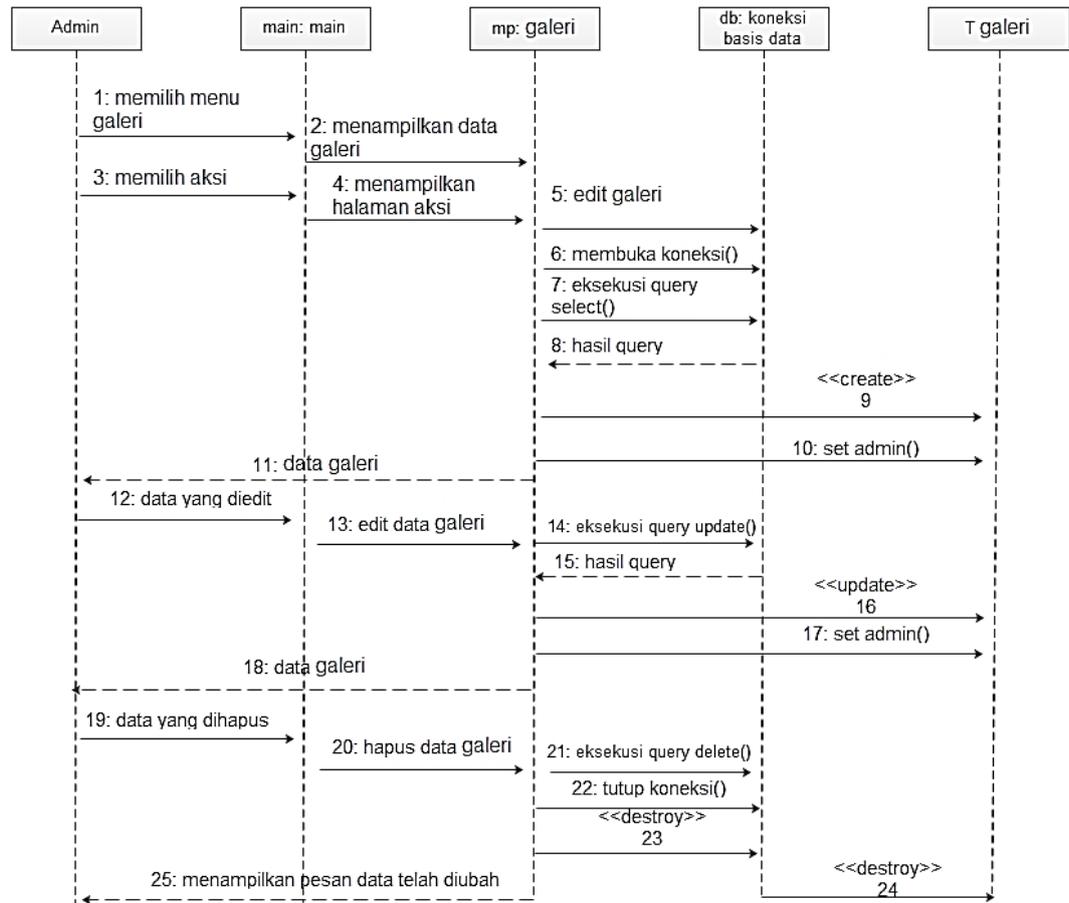
Pada sequence diagram tersebut, menggambarkan proses dimana admin melakukan kelola data mutasi. Jika proses kelola data mutasi berhasil, maka sistem akan melakukan proses untuk menampilkan halaman admin. Dimana proses tersebut terhubung ke tabel mutasi.



Gambar 12. Sequence Diagram Admin Kelola Data Mutasi

5. Sequence Diagram Admin Kelola Menu Galeri

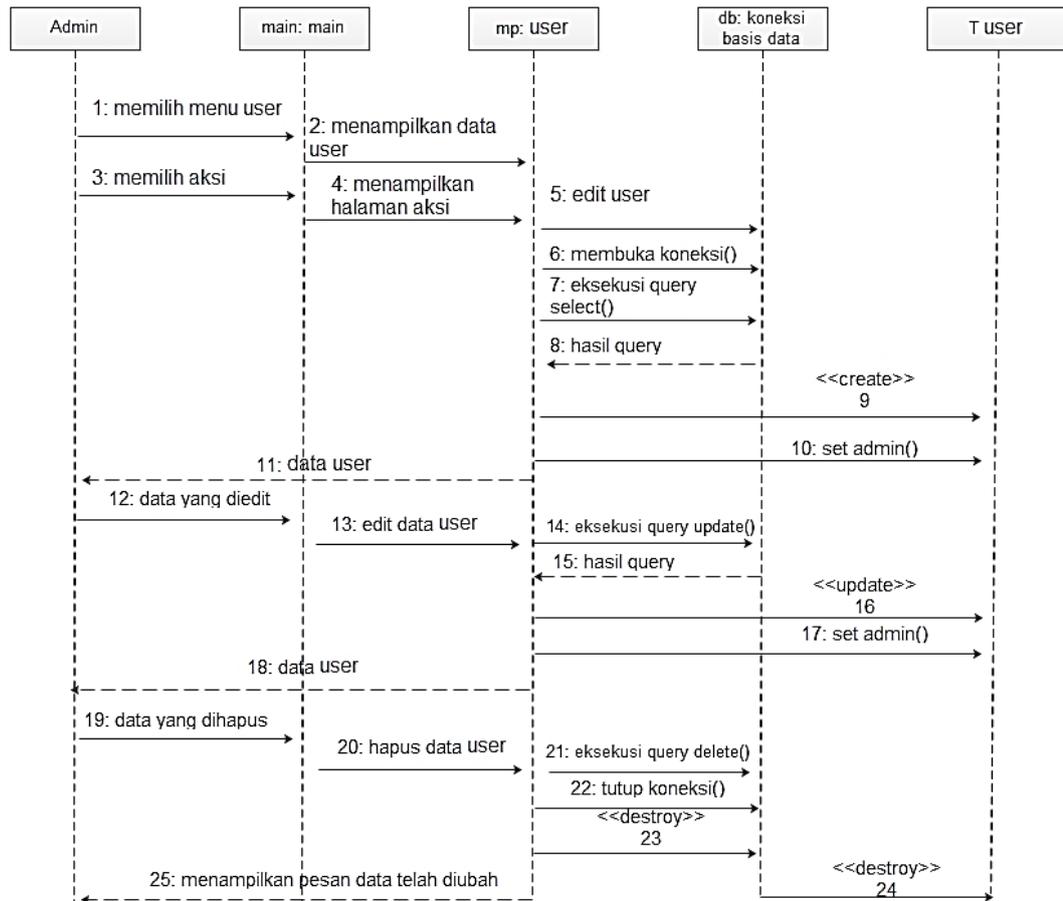
Pada sequence diagram tersebut, menggambarkan proses dimana admin melakukan kelola menu galeri. Jika proses kelola menu galeri berhasil, maka sistem akan melakukan proses untuk menampilkan halaman admin. Dimana proses tersebut terhubung ke tabel galeri.



Gambar 13.Sequence Diagram Admin Kelola Menu Galeri

6. Sequence Diagram Admin Kelola Menu User

Pada sequence diagram tersebut, menggambarkan proses dimana admin melakukan kelola menu user. Jika proses kelola menu user berhasil, maka sistem akan melakukan proses untuk menampilkan halaman admin. Dimana proses tersebut terhubung ke tabel user.



Gambar 14.Sequence Diagram Admin Kelola Menu User

f. Presentation Modeling

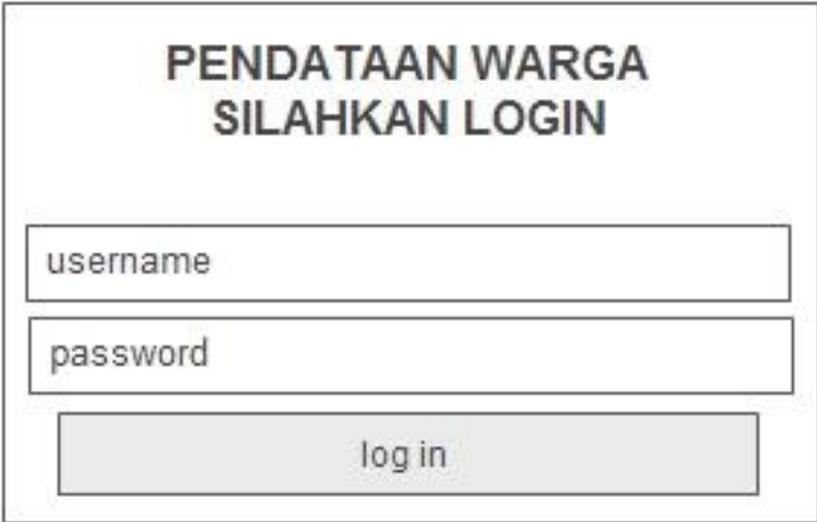
Tahapan ini mempersentasikan struktur dan sifat dari user interface hasil pemodelan proses (process Modelling). Pada presentation model ini akan dilakukan representasi struktur *web* ke dalam sebuah *userinterface* dan mendesain halaman *web* sebagai berikut.

1. *User Interface Berbasis Website*

a. Admin

1. User Interface Login Admin

Halaman *login* merupakan halaman yang digunakan oleh *user* untuk dapat *login* ke aplikasi dengan memasukkan *username* dan *password*.

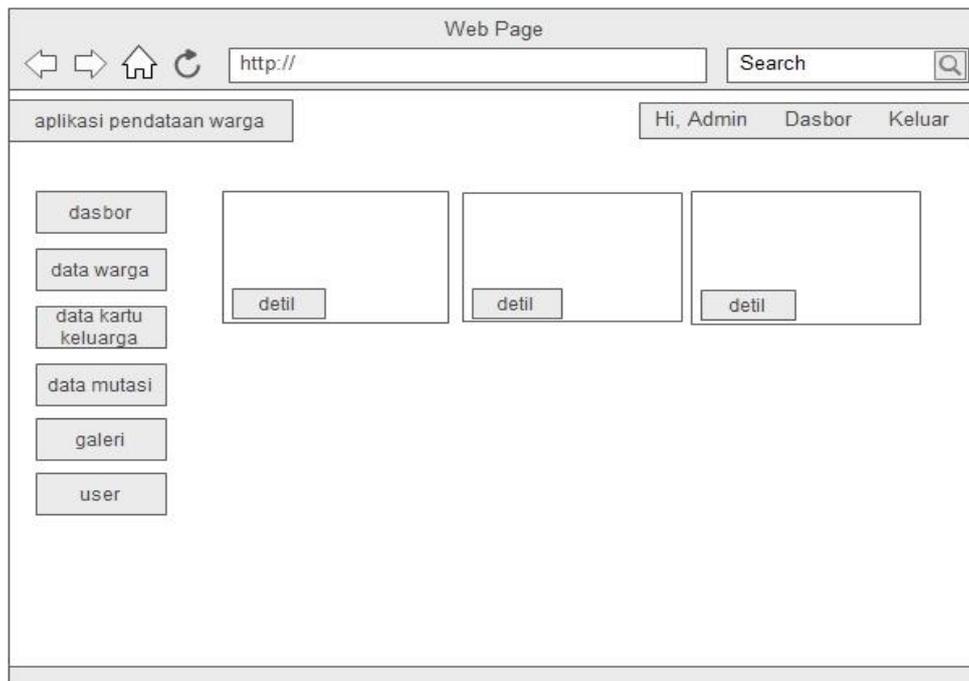


The image shows a login form titled "PENDATAAN WARGA SILAHKAN LOGIN". It contains three input fields: a text field for "username", a text field for "password", and a button labeled "log in".

Gambar 15. User Interface Admin Login

2. User Interface Dasbor Admin

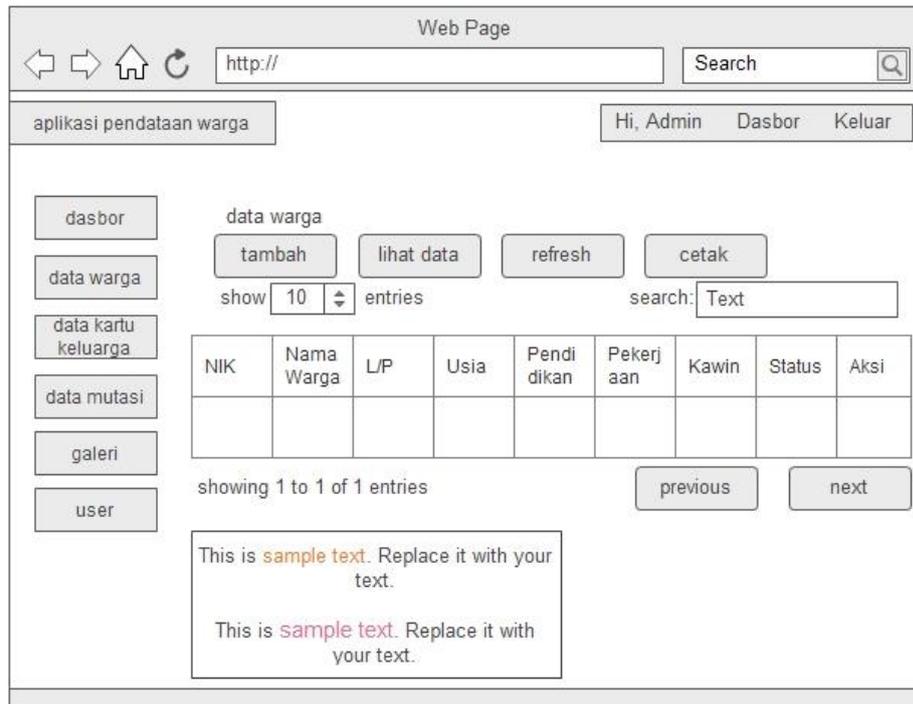
Halaman Dasbor Admin merupakan halaman beranda yang tampil saat admin pertama kali masuk ke dalam sistem setelah berhasil melakukan login.



Gambar 16. User Interface Dasbor Admin

3. User Interface Menu Data Warga

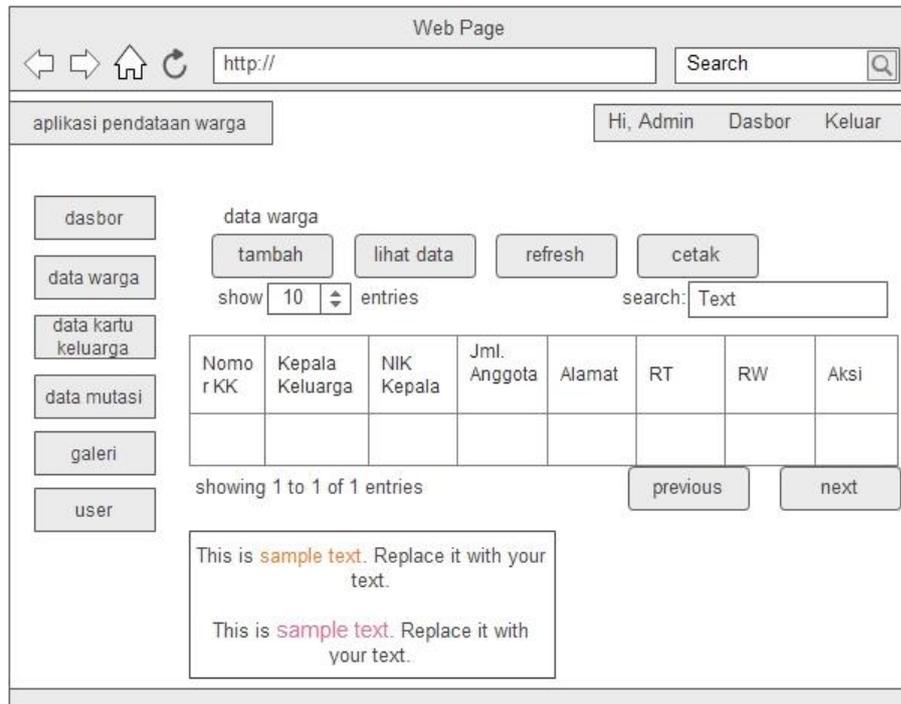
Halaman menu data warga merupakan halaman yang digunakan oleh admin untuk menambah informasi data warga yang akan digunakan untuk melengkapi informasi data warga pada aplikasi pendataan warga.



Gambar 17. User Interface Menu Data Warga

4. User Interface Menu Data Kartu Keluarga

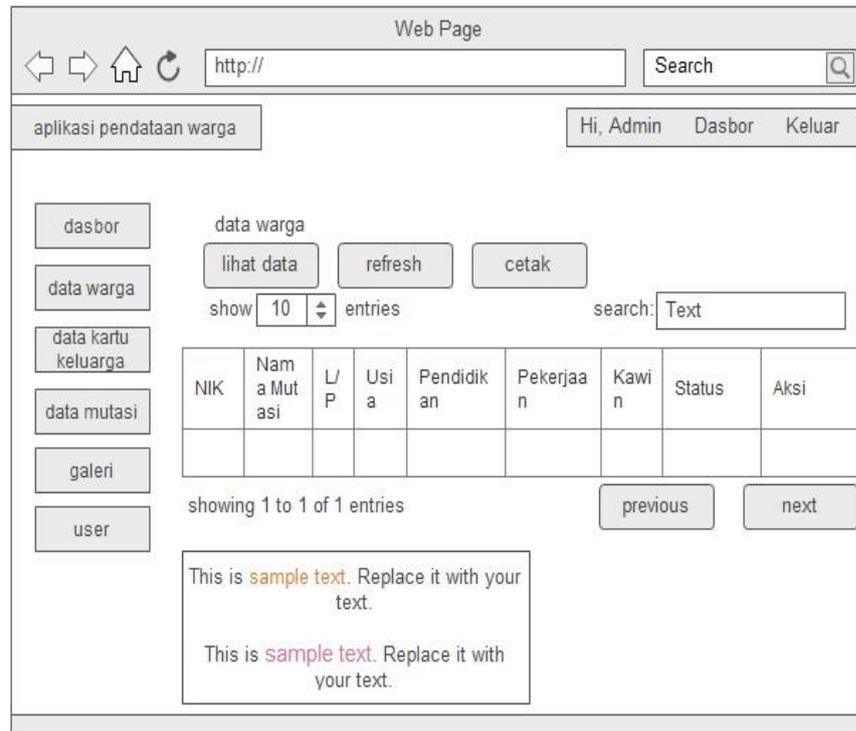
Halaman menu data kartu keluarga merupakan halaman yang digunakan oleh admin untuk menambah informasi data kartu keluarga dengan memasukkan informasi nomor kk, kepala keluarga, nik kepala, jumlah anggota, alamat, rt dan rw yang nantinya akan digunakan untuk melengkapi informasi aplikasi pendataan warga.



Gambar 18. User Interface Menu Data Kartu Keluarga

5. User Interface Menu Data Mutasi

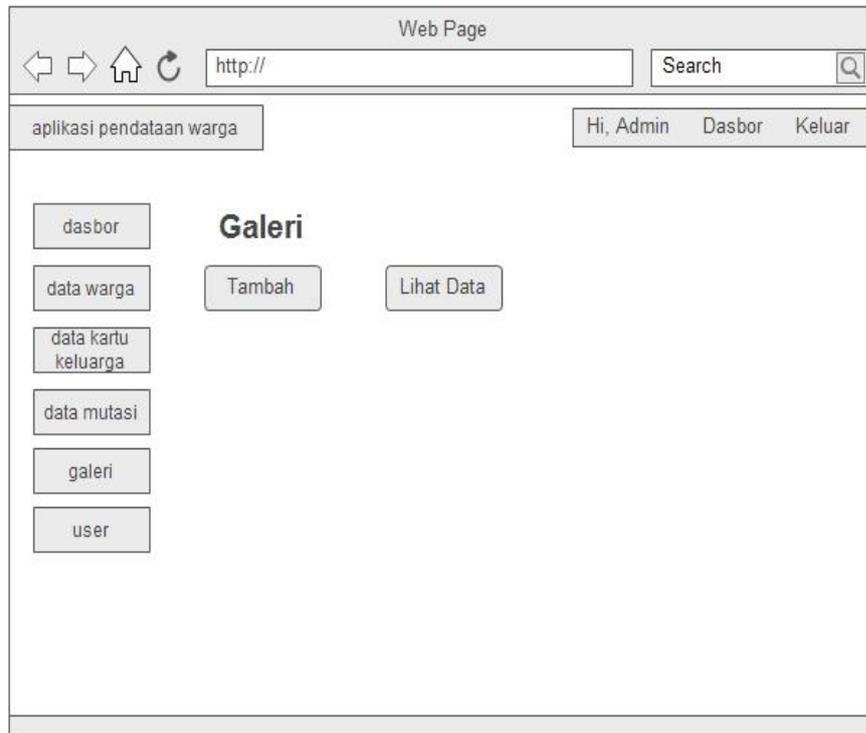
Halaman menu data mutasi merupakan halaman yang digunakan oleh admin untuk menambah informasi data data mutasi dengan memasukan informasi nik, nama mutasi, jenis kelamin, usia, pendidikan, pekerjaan, status perkawinan, dan status yang nantinya akan digunakan untuk melengkapi informasi aplikasi pendataan warga.



Gambar 19. User Interface Menu Data Mutasi

6. User Interface Menu Galeri

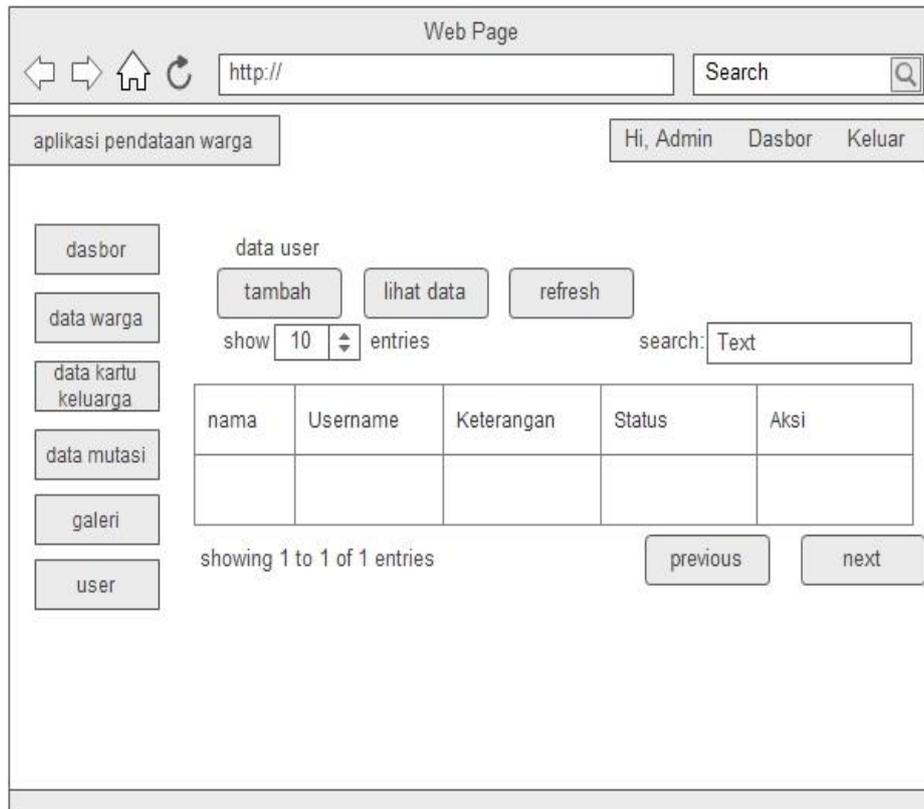
Halaman menu galeri merupakan halaman yang digunakan oleh admin untuk menambah informasi data galeri dengan memasukan gambar yang nantinya akan digunakan untuk melengkapi informasi aplikasi pendataan warga.



Gambar 20. User Interface Menu Galeri

7. User Interface Menu User

Halaman menu user merupakan halaman yang digunakan oleh admin untuk menambah informasi data user dengan memasukan nama, username, keterangan, dan status yang nantinya akan digunakan untuk melengkapi informasi aplikasi pendataan warga.



Gambar 21. User Interface Menu User

BAB IV

IMPLEMENTASI DAN PEMBAHASAN

A. Implementasi

1. Uji Coba Sistem dan Program

Pada tahap ini, penulis akan melakukan pengujian terhadap semua sistem informasi, apakah sudah sesuai dengan fungsi dan kegunaannya dalam pengujian, pada pengujian ini menggunakan *black box*.

B. Pengujian Admin

1. Login

Tabel 13 Blackbox Login

No.	Aksi	Hasil yang Diharapkan	Hasil Keluaran	Keterangan
1.	Mengisi data Username dan Password	Masuk ke halaman beranda admin	Menampilkan beranda admin	√
2.	Salah mengisi data Username dan Password	Menampilkan pesan "Login Gagal Username atau Password Salah"	Menampilkan pesan "Login Gagal Username atau Password Salah"	√

2. Kelola Data Warga

Tabel 14 *Blackbox* Kelola Data Warga

No.	Aksi	Hasil yang Diharapkan	Hasil Keluaran	Keterangan
1.	Klik Tambah	Masukan NIK, Nama Warga, Tempat Lahir, Tanggal Lahir, Jenis Kelamin, Alamat KTP, Alamat, Desa/kelurahan, Kecamatan, Kabupaten/Kota, Provinsi, Negara, RT, RW, Agama, Pendidikan Terakhir, Pekerjaan, Status Perkawinan, Status Tinggal.	Data Warga berhasil ditambahkan	√
2.	Lihat Data	Tampilan Data Warga seperti NIK, Nama Warga, L/P, Usia, Pendidikan, Pekerjaan, Kawin, Status.	Data Warga berhasil di lihat.	√
3.	Refresh	Yakin memperbaharui data warga?	Data berhasil diperbaharui.	√
4.	Cetak	Yakin ingin melakukan cetak data warga?	Data berhasil dicetak.	√

3. Kelola Data Kartu Keluarga

Tabel 15 *Blackbox* Kelola Data Kartu Keluarga

No.	Aksi	Hasil yang Diharapkan	Hasil Keluaran	Keterangan
1.	Klik Tambah	Masukan Nomor Kartu Keluarga, ID Kepala Keluarga, Alamat, Desa/kelurahan, Kecamatan, Kabupaten/kota, Provinsi, Negara, RT, RW, Kode Pos.	Data kartu keluarga berhasil ditambahkan	√
2.	Lihat Data	Tampilan data kartu keluarga seperti nomor KK, Kepala Keluarga, NIK Kepala, Jml. Anggota, Alamat, RT, RW.	Data kartu keluarga berhasil di lihat.	√
3.	Refresh	Yakin memperbaharui data kartu keluarga?	Data kartu keluarga berhasil diperbaharui.	√
4.	Cetak	Yakin ingin melakukan cetak data kartu keluarga?	Data Berhasil dicetak.	√

4. Kelola Data Mutasi

Tabel 16 *Blackbox* Kelola Data Mutasi

No.	Aksi	Hasil yang Diharapkan	Hasil Keluaran	Keterangan
1.	Lihat Data	Tampilan data mutasi seperti, NIK, Nama Mutasi, L/P, Usia, Pendidikan, Pekerjaan, Kawin, Status.	Data mutasi berhasil di lihat.	√
3.	Refresh	Yakin memperbaharui data mutasi?	Data mutasi berhasil diperbaharui.	√
4.	Cetak	Yakin ingin melakukan cetak data mutasi?	Data mutasi berhasil dicetak.	√

5. Kelola Galeri

Tabel 17 *Blackbox* Kelola Galeri

No.	Aksi	Hasil yang Diharapkan	Hasil Keluaran	Keterangan
1.	Klik Tambah	Tambahkan data galeri seperti, Path, Caption dan Tautan.	Data galeri berhasil di lihat.	√
2.	Lihat Data	Menampilkan data galeri seperti, Path, Caption dan Tautan.	Data galeri berhasil di lihat.	√

6. Kelola User

Tabel 18 *Blackbox* Kelola User

No.	Aksi	Hasil yang Diharapkan	Hasil Keluaran	Keterangan
1.	Klik Tambah	Tambahkan data user seperti, Nama User, Username, Password, Keterangan, Status, Desa/kelurahan, Kecamatan, Kabupaten/kota, Provinsi, Negara, RT dan RW.	Data user berhasil di tambahkan.	√
2.	Lihat Data	Menampilkan data user seperti, Nama, Username, Keterangan, Status.	Data user berhasil di lihat.	√
3.	Refresh	Yakin memperbaharui data user?	Data user berhasil diperbaharui.	√

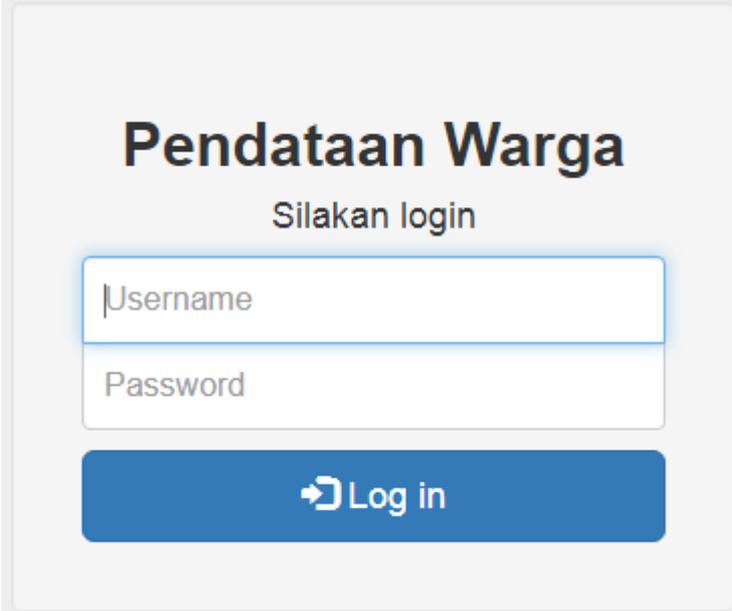
2. Manual Program

Dari desain yang telah dibuat, baik itu dari desain basis data ataupun desain tampilan, memasukan kode perintah serta merancang laporan. Maka

diperlukan implementasi yang dapat menunjukkan aplikasi yang dibuat dapat berjalan dengan baik sesuai dengan yang diharapkan.

1. Implementasi *User Interface Login* Aplikasi Pendataan Warga

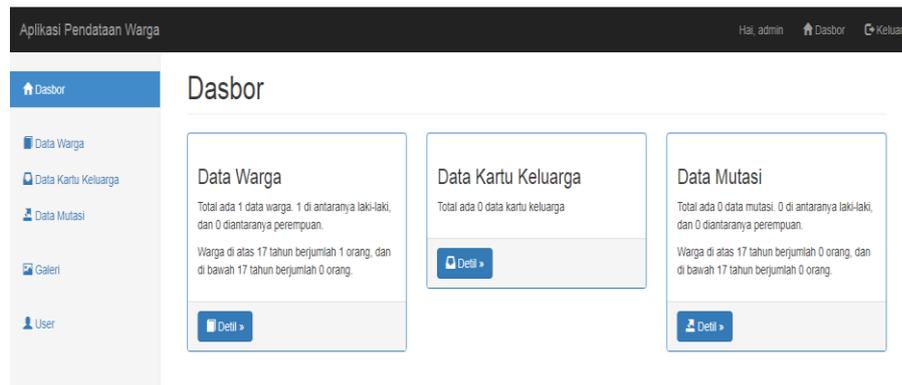
Pada implementasi ini, user dapat masuk ke dalam sistem dengan cara login memasukkan username dan password, lalu klik tombol Log In.



Gambar 22.Login Aplikasi Pendataan Warga

2. Implementasi *User Interface Halaman Beranda Admin*

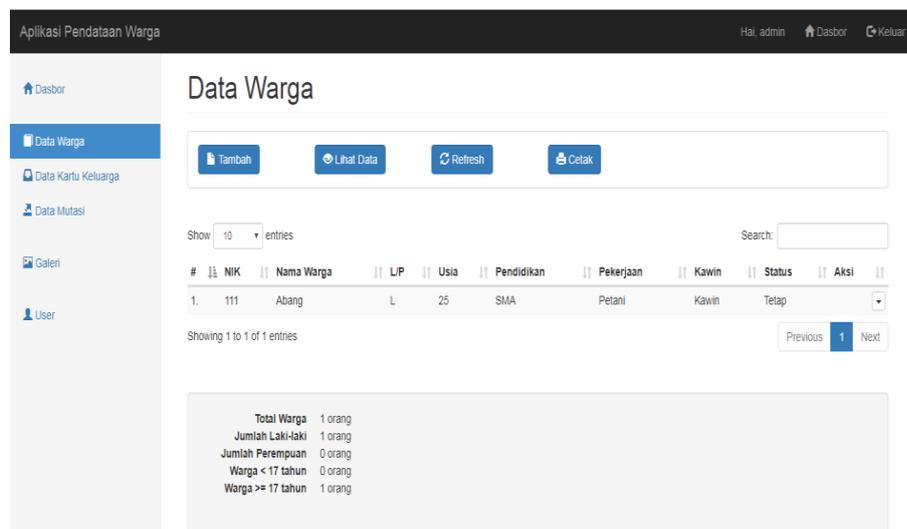
Pada implementasi ini, admin berhasil melakukan login dan masuk ke halaman beranda, serta dapat melihat data warga, data kartu keluarga dan data mutasi yang terlihat pada dasbor admin.



Gambar 23. Halaman Beranda Admin

3. Implementasi *User Interface* Halaman Data Warga

Pada implementasi ini, admin berhasil melakukan login dan masuk ke halaman beranda, serta dapat melihat data warga, data kartu keluarga dan data mutasi yang terlihat pada dasbor admin.



Gambar 24. Halaman Data Warga

4. Implementasi *User Interface* Halaman Tambah Data Warga

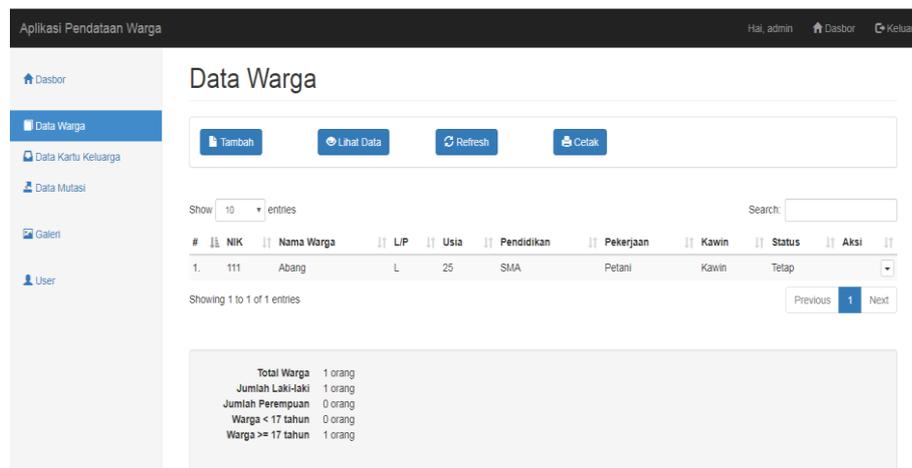
Pada implementasi ini, admin berhasil melakukan login dan masuk ke halaman data warga, dan melakukan aksi tambah data warga dengan memasukkan data seperti NIK, Nama Warga, Tempat Lahir, Tanggal Lahir, Jenis Kelamin, Alamat KTP, Alamat, Desa/kelurahan, Kecamatan, Kabupaten/Kota, Provinsi, Negara, RT, RW, Agama, Pendidikan Terakhir, Pekerjaan, Status Perkawinan, Status Tinggal pada tambah data warga.

The screenshot shows a web application interface for managing citizen data. The page title is 'Data Warga'. On the left, there is a sidebar menu with the following items: 'Desbor', 'Data Warga' (highlighted), 'Data Kartu Keluarga', 'Data Mutasi', 'Galeri', and 'User'. The main content area has a title 'Data Warga' and four buttons: 'Tambah', 'Ujral Data', 'Refresh', and 'Cetak'. Below the buttons, there are two sections: 'A. Data Pribadi' and 'B. Data Alamat'. Section A includes fields for NIK, Name Warga, Tempat Lahir, Tanggal Lahir (with a date picker showing 2019-10-23), and Jenis Kelamin (with a dropdown menu). Section B includes fields for Alamat KTP, Alamat, Desa/Kelurahan (with a dropdown menu showing 'Kibondalem'), and Kecamatan (with a dropdown menu showing 'Pamalang').

Gambar 25. Halaman Tambah Data Warga

5. Implementasi *User Interface* Halaman Lihat Data Warga

Pada implementasi ini, admin berhasil melakukan login dan masuk ke halaman data warga, serta dapat melihat data warga, seperti NIK, Nama Warga, L/P, Usia, Pendidikan, Pekerjaan, Kawin dan Status yang terlihat pada data warga admin.



The screenshot displays the 'Data Warga' interface. At the top, there are navigation buttons: 'Tambah', 'Lihat Data', 'Refresh', and 'Cetak'. Below these, a table lists the resident data. The table has columns for '#', 'NIK', 'Nama Warga', 'L/P', 'Usia', 'Pendidikan', 'Pekerjaan', 'Kawin', 'Status', and 'Aksi'. The first row contains the data for 'Abang' with NIK '111', age '25', education 'SMA', occupation 'Petani', marital status 'Kawin', and status 'Tetap'. Below the table, a summary box provides the following statistics:

Total Warga	1 orang
Jumlah Laki-laki	1 orang
Jumlah Perempuan	0 orang
Warga < 17 tahun	0 orang
Warga >= 17 tahun	1 orang

Gambar 26. Halaman Lihat Data Warga

6. Implementasi *User Interface* Halaman Refresh Data Warga

Pada implementasi ini, admin berhasil melakukan login dan masuk ke halaman data warga, serta dapat melakukan refresh atau pembaharuan halaman pada data warga admin.

The screenshot shows the 'Data Warga' page. At the top, there are navigation links for 'Tambah', 'Lihat Data', 'Refresh', and 'Cetak'. Below this, there is a search bar and a table with the following data:

#	NIK	Nama Warga	LIP	Usia	Pendidikan	Pekerjaan	Kawin	Status	Aksi
1.	111	Abang	L	25	SMA	Petani	Kawin	Tetap	

Below the table, there is a summary box with the following statistics:

- Total Warga: 1 orang
- Jumlah Laki-laki: 1 orang
- Jumlah Perempuan: 0 orang
- Warga < 17 tahun: 0 orang
- Warga >= 17 tahun: 1 orang

Gambar 27.Refresh Data Warga

7. Implementasi *User Interface* Halaman Data Kartu Keluarga

Pada implementasi ini, admin berhasil melakukan login dan masuk ke halaman data kartu keluarga.

The screenshot shows the 'Data Kartu Keluarga' page. At the top, there are navigation links for 'Tambah', 'Lihat Data', 'Refresh', and 'Cetak'. Below this, there is a search bar and a table with the following data:

#	Nomor KK	Kepala Keluarga	NIK Kepala	Jml. Anggota	Alamat	RT	RW	Aksi
No data available in table								

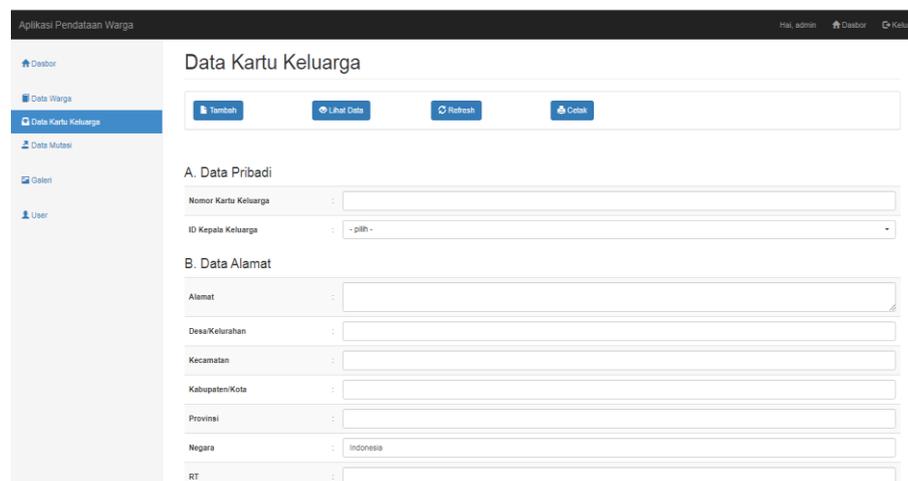
Below the table, there is a summary box with the following statistics:

- Total Kartu Keluarga: 0 keluarga

Gambar 28.Halaman Data Kartu Keluarga

8. Implementasi *User Interface* Halaman Tambah Data Kartu Keluarga

Pada implementasi ini, admin berhasil melakukan login dan masuk ke halaman data kartu keluarga, dan melakukan aksi tambah data kartu keluarga dengan memasukan data seperti Nomor Kartu Keluarga, ID Kepala Keluarga, Alamat, Desa/kelurahan, Kecamatan, Kabupaten/kota, Provinsi, Negara, RT, RW, Kode Pos pada tambah data kartu keluarga

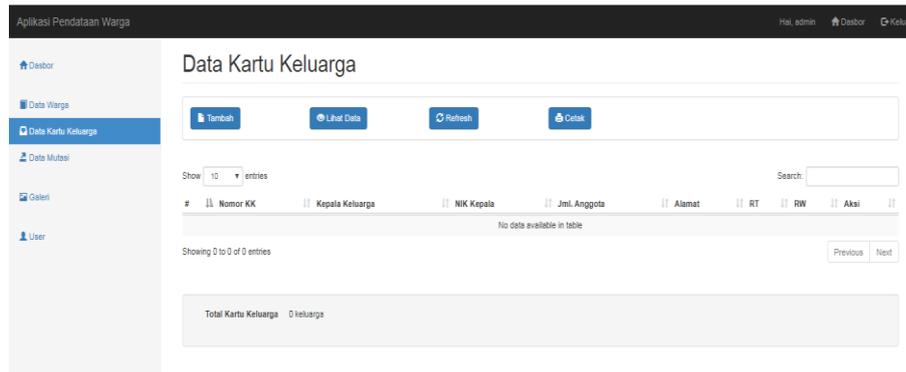


The screenshot shows a web application interface for adding family data. The title is 'Data Kartu Keluarga'. There are four buttons at the top: 'Tambah', 'Lihat Data', 'Refresh', and 'Cetak'. The form is divided into two sections: 'A. Data Pribadi' and 'B. Data Alamat'. Section A includes 'Nomor Kartu Keluarga' (text input), 'ID Kepala Keluarga' (dropdown menu with '- pilih -'), and 'RT' (text input). Section B includes 'Alamat' (text input), 'Desa/Kelurahan' (text input), 'Kecamatan' (text input), 'Kabupaten/Kota' (text input), 'Provinsi' (text input), 'Negara' (text input with 'Indonesia' selected), and 'RT' (text input).

Gambar 29. Halaman Tambah Data Kartu Keluarga

9. Implementasi *User Interface* Halaman Lihat Data Kartu Keluarga

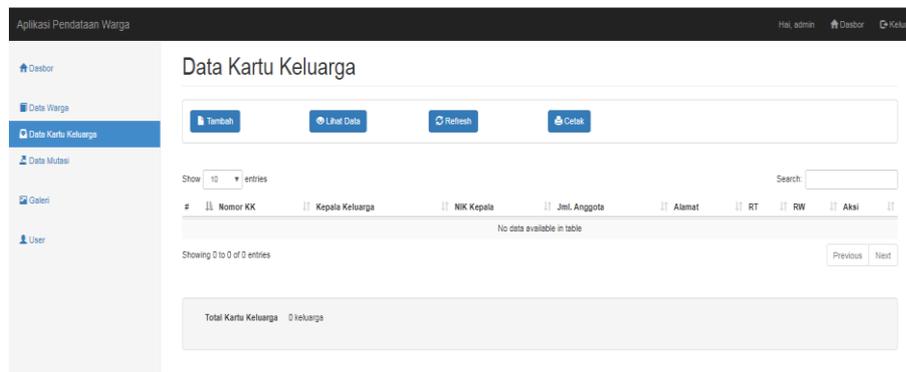
Pada implementasi ini, admin berhasil melakukan login dan masuk ke halaman data kartu keluarga, dan melakukan aksi lihat data kartu keluarga seperti Nomor Kartu Keluarga, Kepala Keluarga, Jumlah Anggota, Alamat, RT, dan RW pada lihat data kartu keluarga.



Gambar 30. Halaman Lihat Data Kartu Keluarga

10. Implementasi *User Interface* Halaman Refresh Data Kartu Keluarga

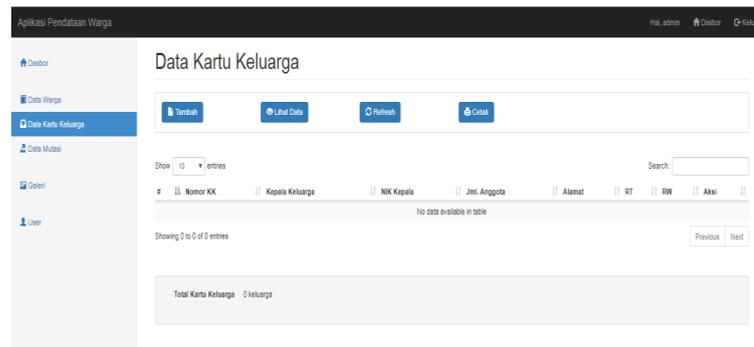
Pada implementasi ini, admin berhasil melakukan login dan masuk ke halaman data kartu keluarga, serta dapat melakukan refresh atau pembaharuan halaman pada data kartu keluarga.



Gambar 31. Halaman Refresh Data Kartu Keluarga

11. Implementasi *User Interface* Halaman Cetak Data Kartu Keluarga

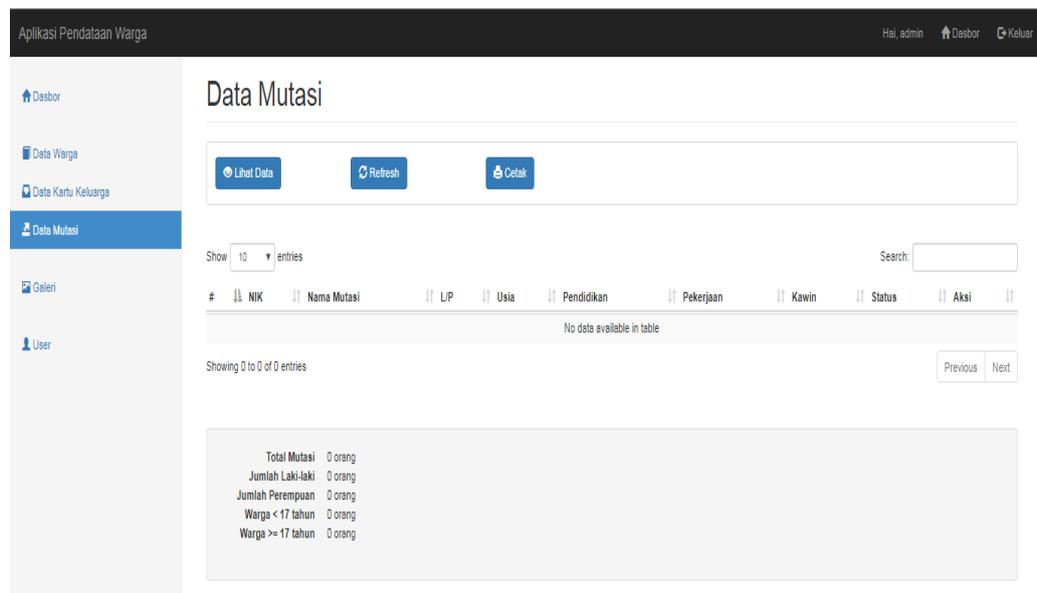
Pada implementasi ini, admin berhasil melakukan login dan masuk ke halaman data kartu keluarga, serta dapat melakukan cetak pada data kartu keluarga.



Gambar 32. Halaman Cetak Data Kartu Keluarga

12. Implementasi *User Interface* Halaman Data Mutasi

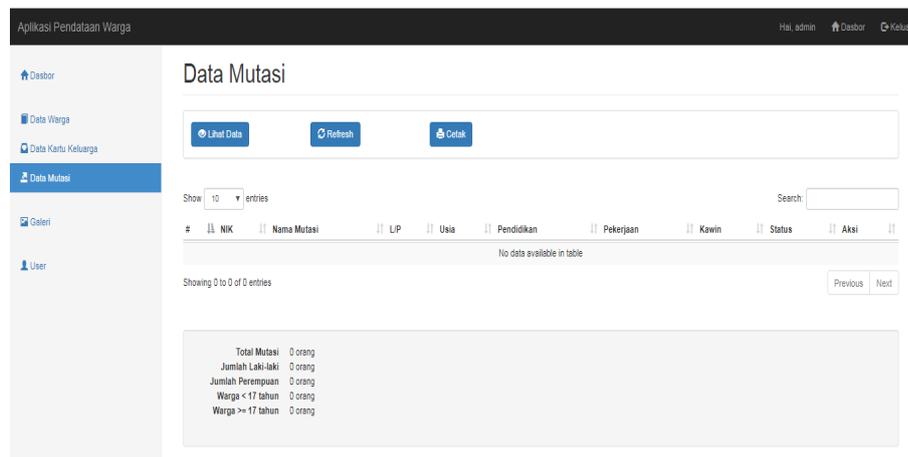
Pada implementasi ini, admin berhasil melakukan login dan masuk ke halaman data mutasi dan dapat mengelola data seperti lihat data, refresh dan cetak data mutasi.



Gambar 33. Halaman Data Mutasi

13. Implementasi Halaman Lihat Data Mutasi *User Interface*

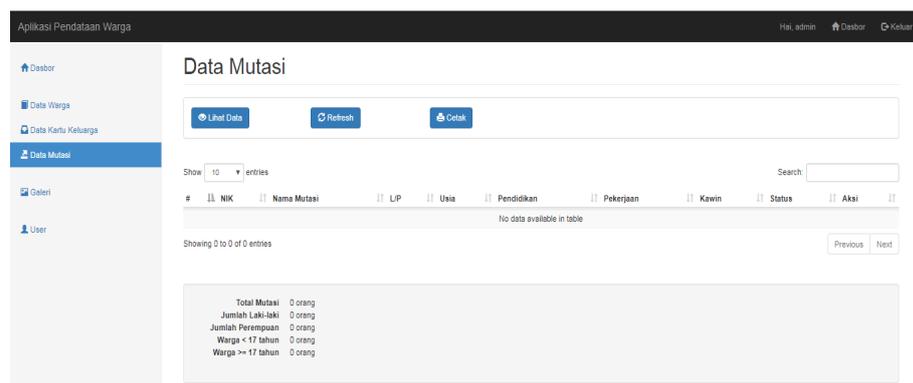
Pada implementasi ini, admin berhasil melakukan login dan masuk ke halaman data mutasi dan dapat mengelola data seperti lihat data, refresh dan cetak data mutasi.



Gambar 34. Halaman Lihat Data Mutasi

14. Implementasi *User Interface* Halaman Refresh Data Mutasi

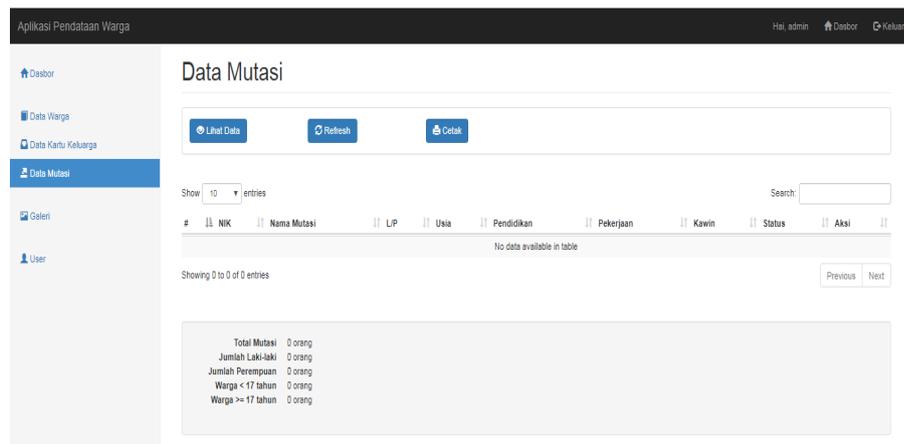
Pada implementasi ini, admin berhasil melakukan login dan masuk ke halaman data mutasi serta dapat melakukan refresh atau pembaharuan halaman pada data mutasi.



Gambar 35. Halaman Refresh Data Mutasi

15. Implementasi *User Interface* Halaman Cetak Data Mutasi

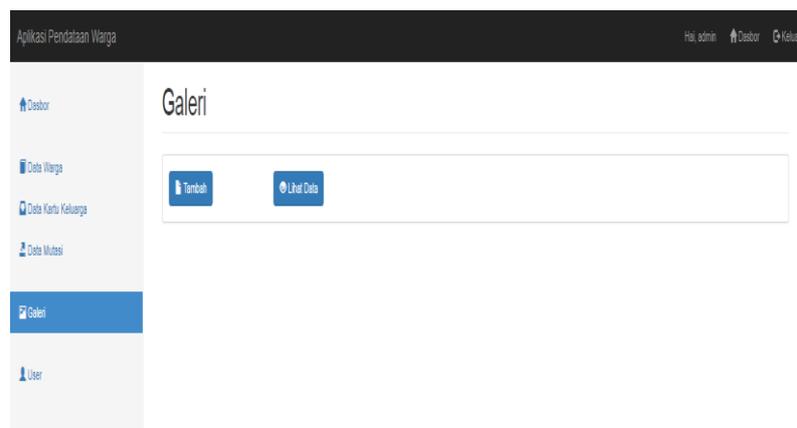
Pada implementasi ini, admin berhasil melakukan login dan masuk ke halaman cetak data mutasi serta dapat melakukan cetak pada data mutasi.



Gambar 36. Halaman Cetak Data Mutasi

16. Implementasi *User Interface* Halaman Galeri

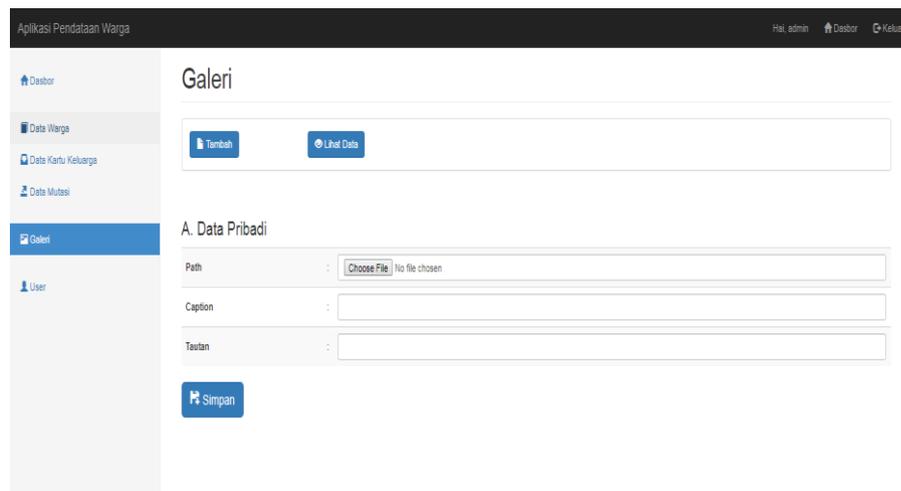
Pada implementasi ini, admin berhasil melakukan login dan masuk ke halaman galeri dan dapat mengelola data seperti tambah dan lihat data galeri.



Gambar 37. Halaman Galeri

17. Implementasi *User Interface* Halaman Tambah Data Galeri

Pada implementasi ini, admin berhasil melakukan login dan masuk ke halaman tambah data galeri dengan memasukkan data berupa Path, Caption dan Tautan.

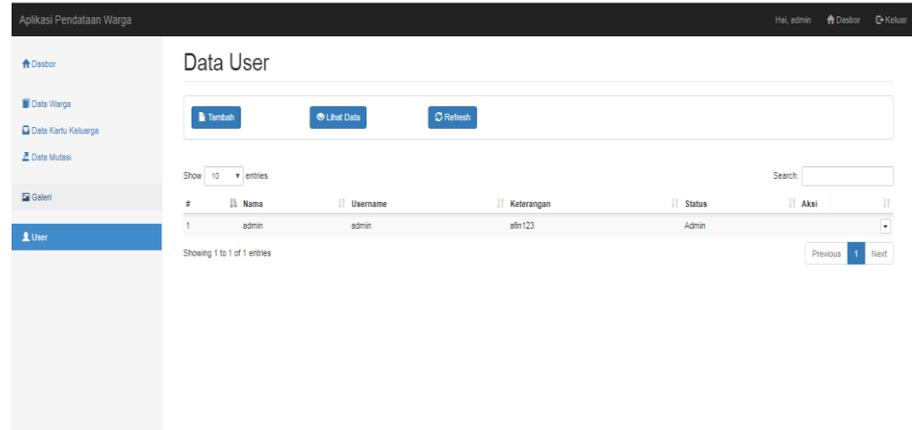


The screenshot displays the 'Galeri' page within the 'Apikasi Pendaftaran Warga' application. The top navigation bar includes the application name, user information ('Hai, admin'), and navigation icons for 'Dashboard' and 'Logout'. The left sidebar lists menu items: 'Dashboard', 'Data Warga', 'Data Kartu Keluarga', 'Data Mutesi', 'Galeri' (highlighted), and 'User'. The main content area features a 'Galeri' header, a 'Tambah' button, and a 'Lihat Data' button. Below this is a section titled 'A. Data Pribadi' containing three input fields: 'Path' (with a 'Choose File' button and 'No file chosen' text), 'Caption', and 'Tautan'. A 'Simpan' button is positioned at the bottom of the form.

Gambar 38. Halaman Tambah Data Galeri

18. Implementasi *User Interface* Halaman User

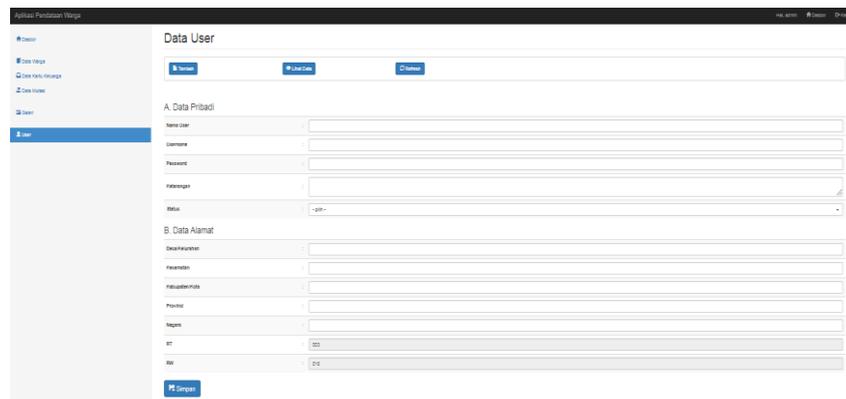
Pada implementasi ini, admin berhasil melakukan login dan masuk ke halaman user dan dapat melakukan kelola data seperti tambah, lihat data dan refresh data user.



Gambar 39. Halaman User

19. Implementasi *User Interface* Halaman Tambah Data User

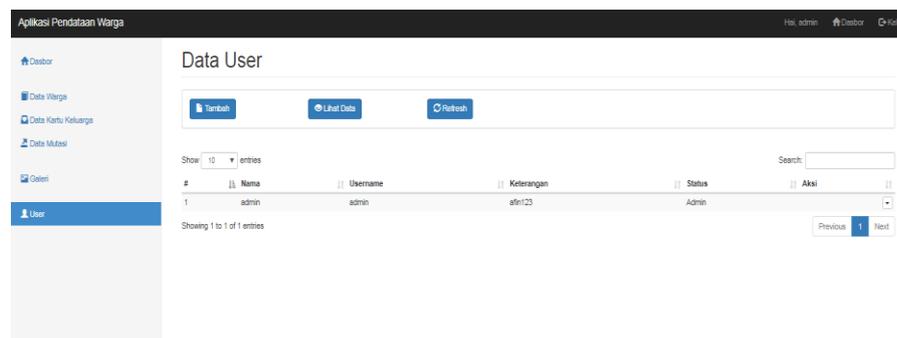
Pada implementasi ini, admin berhasil melakukan login dan masuk ke halaman user dan dapat melakukan kelola tambah data user dengan memasukan informasi seperti nama user, username, password, keterangan, status, desa/kelurahan, kecamatan, kabupaten/kota, provinsi, negara, rt dan rw.



Gambar 40. Halaman Tambah Data User

20. Implementasi *User Interface* Halaman Lihat Data User

Pada implementasi ini, admin berhasil melakukan login dan masuk ke halaman user dan dapat melihat informasi data user seperti nama user, username, keterangan, dan status.

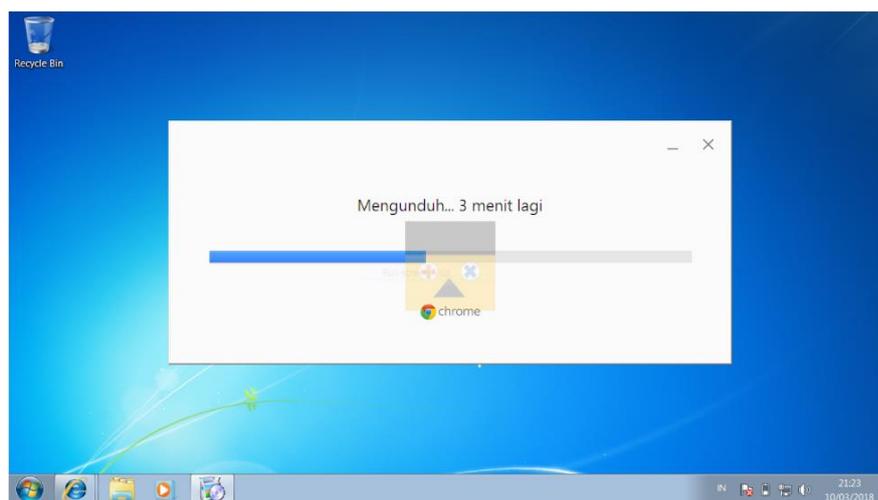


Gambar 41. Halaman Lihat Data User

1. Manual *Instalasi*

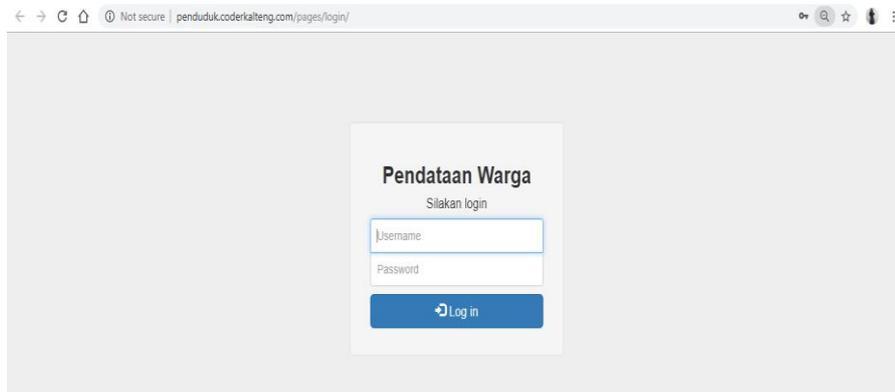
a. *Instalasi* Aplikasi Browser (*Chrome, Firefox, Internet Explorer*)

1. *Instal* chrome dengan cara klik jalankan atau simpan pada aplikasi yang sudah di *running*.



Gambar 42. Tampilan Menginstal Google Chrome

2. Setelah terinstal, buka browser dan masuk url berikut <https://desaharingen.000webhostapp.com/pages/login> dan aplikasi pun dapat digunakan.



Gambar 43. Tampilan halaman utama

2. Pemeliharaan Sistem

Dalam melakukan pemeliharaan pada sistem ini ada beberapa hal yang akan penulis lakukan, diantaranya adalah sebagai berikut:

- a. Melakukan perbaikan pada sistem saat sistem tersebut sudah digunakan dan terjadi *error* atau *bug* pada struktur kode program penyusun sistem.
- b. Memastikan bahwa sistem berjalan dengan baik, dengan cara mengecek satu persatu fungsi-fungsi tombol pada sistem apakah sudah sesuai dengan yang diharapkan atau belum.
- c. Sistem operasi disini memegang peran yang sangat penting karena berpengaruh pada kinerja dari sistem yang dibuat. Agar sistem dapat berjalan dengan baik tentunya didukung pula dengan sistem operasi yang baik, dalam hal ini sistem operasi yang penulis gunakan untuk membangun “*Aplikasi Web Monitoring Pendataan Penduduk Desa Haringen Barito Timur Kalimantan Tengah*”

BAB V

PENUTUP

A. Kesimpulan

Berdasarkan uraian hasil uji coba sistem dari program dan pembuatan program yang telah dijabarkan pada bab sebelumnya, dapat disimpulkan sebagai berikut :

1. Aplikasi yang penulis buat berupa aplikasi *monitoringsystem* informasi di buat dengan menggunakan *php*, *databaseMySQL*, *HTML* dan *Bootsrap*.
2. Aplikasi Dapat berjalan di online dengan menggunakan hosting pada domain <https://desaharingen.000webhostapp.com/>
3. Pendataan yang dilakukan hanya pada Desa Haringen di kabupaten Barito Timur.
4. Sistem yang dibangun meliputi pendataan penduduk yang ada di Desa Haringen Barito Timur, pindah domisili, blangko pendaftaran.
5. Aplikasi web yang dibuat adalah aplikasi web yang bersifat *web responsive*.
6. Pada Aplikasi ini dapat menampilkan grafik total warga , rentang umur warga, serta jenis kelamin data warga.

Demikian dari proses implementasi yang didapatkan hasil bahwa Aplikasi ini cukup baik dimana Sistem pendukung keputusan ini dapat melakukan Pengolahan terhadap data – data yang diperlukan, sehingga dapat menghasilkan *Input* data yang tersimpan dan *Output* data berupa data *history*.

B. Saran

Saran yang diberikan saat aplikasi ini diselesaikan, agar kedepannya dapat dilanjutkan dimana ada beberapa hal yang diperlukan menyangkut pendataan penduduk desa haringen. Dengan adanya aplikasi yang sudah di buat bisa membantu kantor desa haringen dalam pendataan dan memberikan informasi yang akurat karena bersifat responsif

DAFTAR PUSTAKA

Jurnal Ilmiah Seminar Nasional Komputer dan Informatika 2017, *Sistem Informasi Kependudukan Pada Kecamatan Kadungora Kabupaten Garut*, Kusumawati Ani, Hendro Pudjiantoro, Nursantika Dian, Cimahi.

Jurnal Ilmiah PROSISKO, Volume 3 No.1, *Rancang Bangun Sistem Extreme Programming Sebagai Metodologi Pengembangan Sistem*, Fatoni Ahmad & Dwi Dhany, Serang.

Jurnal Ilmiah Ilmu Komputer, Volume 3 No.1, *Perancangan dan pembuatan aplikasi alih media dokumen rekam medis berbasis web di RSUD dr. R. Soedarsono Pasuruan*, Jinan Nachrul & Permatasari Vita, Jember.

Jurnal Ilmiah Teknologi Informasi, Volume 21 No.21, *Rancang Bangun Sistem Informasi Kependudukan Berbasis Android*, Sunardi, Hersatoto Listiyono, Semarang.

Jurnal Ilmiah Teknik Informatika, Volume 8, No.1, *Implementasi Sikades (Sistem Informasi Kependudukan Desa) Untuk Kemudahan Layanan Administrasi Desa Berbasis Web Mobile*, Noviyanto Fiftin, Setiadi Tedy, Wahyuningsih Iis, , Jogjakarta.

Jurnal Edukasi dan Penelitian Informatika, Volume 1, No.2, *Perancangan Sistem Pendataan Penduduk pada Kelurahan Air Itam dengan Object Oriented*, Marini, Bangka Belitung.

Romney, Steinbert, 2015, *Accounting Information System Edisi 13*, Salemba Empat Jakarta.

Sutarman. 2015. *“Pengantar Teknologi Informasi”*.

<http://emeraldsalamatunnisa.blogspot.co.id/2015/04/sistem-informasi.html>.

Diakses 13 Mei 2017.

Tantra, Rudy, 2015, *Manajemen Proyek Sistem informasi*, Andi.Yogyakarta

Arief, M.Rudianto. 2016. *Pemrograman Web Dinamis Menggunakan Php dan Mysql*,Andi.Yogyakarta.

LAMPIRAN