

**SISTEM INFORMASI PELAYANAN DESA MANDALAHAJI
DENGAN MENGAPLIKASIKAN TEKNOLOGI
TRANSPARENT DATA ENCRYPTION
UNTUK KEAMANAN *DATABASE***

PROPOSAL SKRIPSI

**DIAJUKAN SEBAGAI SALAH SATU SYARAT UNTUK MENYUSUN
SKRIPSI PROGRAM STUDI TEKNIK INFORMATIKA**



Disusun Oleh :

NPM	: 17111329
NAMA	: IHSAN NURHAKIM AZIZ
JENJANG STUDI	: STRATA SATU (S1)
PROGRAM STUDI	: TEKNIK INFORMATIKA

**PROGRAM STUDI TEKNIK INFORMATIKA
SEKOLAH TINGGI TEKNOLOGI BANDUNG**

2021

**PENGESAHAN PENGAJUAN
SEMINAR PROPOSAL SKRIPSI**

NPM : 17111329

Nama : Ihsan Nurhakim Aziz

Jenjang Studi : Starata Satu (S1)

Program Studi : Teknik Informatika

Judul Proposal : Sistem Informasi Pelayanan Desa Mandalahaji Dengan
Mengaplikasikan Teknologi *Transparent Data Encryption* Untuk
Keamanan *Database*

Bandung, 16 Maret 2021

Menyetujui

Pembimbing 1

Harya Gusdevi, S.Kom., M.Kom. Tanggal.....

Pembimbing 2

Agus Heryanto, S.T., M.Kom. Tanggal.....

Mengetahui,

Ka. Prodi Teknik Informatika

Ahsani Takwim, S.Kom., M.Kom. Tanggal.....

NIP. 18.95.091

KATA PENGANTAR

Puji syukur kehadiran Allah Swt. atas berkat rahmat dan karunia-Nya, penulis dapat menyelesaikan penusunan proposal skripsi yang berjudul **SISTEM INFORMASI DESA MANDALAHAJI DENGAN MENGAPLIKASIKAN TEKNOLOGI *TRANSPARENT DATA ENCRYPTION* UNTUK KEAMANAN *DATABASE*** sesuai dengan yang direncanakan. Shalawat serta salam semoga selalu terlimpah curahkan kepada jungjunan kita yakni Baginda Nabi Muhammad SAW, kepada keluarganya, para sahabatnya dan mudah-mudahan termasuk kita didalamnya sebagai umatnya yang taat kepada ajarannya.

Penyusunan proposal skripsi ini bertujuan untuk memenuhi Nilai Ujian Akhir Semester dari Mata Kuliah Pra Skripsi Program Studi Teknik Informatika, Sekolah Tinggi Teknologi Bandung. Selama penyusunan proposal ini, penulis telah mendapatkan bimbingan sekaligus dukungan yang begitu besar dari berbagai pihak. Maka pada kesempatan kali ini penulis ingin menyampaikan ucapan terima kasih yang sebesar-besarnya kepada:

1. Bapak Muchammad Naseer, S.Kom., M.T. Selaku Ketua STT Bandung.
2. Bapak Danny Aidil Rismayadi, S.SI., M.Kom. Selaku Wakil Ketua 1.
3. Bapak Ahsani Takwim, S.Kom., M.Kom. Selaku Ketua Program Studi Teknik Informatika.
4. Ibu Harya Gusdevi, S.Kom., M.Kom. Selaku Pembimbing pertama.
5. Bapak Agus Heryanto, S.T., M.Kom. Selaku Pembimbing Kedua.
6. Kedua orang tua penulis tercinta yang selalu memberikan dukungan baik moril dan maetril yang tidak ada habisnya.
7. Metty Puspitawati, S.T. yang selalu memberikan dukungan dan doa.

Penulis Menyadari bahwa dalam penulisan proposal ini masih jauh dari kata sempurna, oleh karena itu kritik dan saran yang bersifat membangun dari pembaca sangat penulis harapkan. Semoga penyusunan proposal ini dapat bermanfaat bagi penulis dan pembaca.

DAFTAR ISI

KATA PENGANTAR.....	i
DAFTAR ISI.....	ii
DAFTAR TABEL.....	v
DAFTAR GAMBAR.....	vi
BAB I.....	1
PENDAHULUAN.....	1
1.1. Latar Belakang.....	1
1.2. Rumusan Masalah.....	3
1.3. Tujuan Penelitian.....	3
1.4. Manfaat Penelitian.....	3
1.4.1. Manfaat Bagi Instansi.....	3
1.4.2. Manfaat Bagi Penulis.....	3
1.5. Ruang Lingkup.....	4
1.6. Sistematika Penulisan.....	4
BAB II.....	5
TINJAUAN PUSTAKA.....	5
2.1. Pengertian E-government.....	5
2.2. Pengertian Sistem Informasi.....	5
2.3. Pengertian Website.....	5
2.4.1. MySQL.....	6
2.5. PHP.....	6
2.6. HTML.....	7
2.7. CSS.....	7
2.8. XML.....	7
2.9. ASP.....	8
2.10. JavaScript.....	8
2.11. Framework.....	8
2.12. Bootstrap.....	8
2.13. Aspek Keamanan.....	9
2.14. Metodologi Keamanan Sistem.....	9
2.15. Enkripsi.....	10
2.16. <i>Transparent Data Encryption</i>	13

2.17. Konsep <i>Transparent Data Encryption</i>	13
2.17.1. <i>Master Encryption Key</i>	13
2.17.2. <i>Wallet</i>	13
2.17.3. <i>Advanced Encryption Standard (AES)</i>	13
2.17.4. Cara Kerja TDE.....	14
2.18. UML (<i>Unified Modelling Language</i>)	15
2.18.1 <i>Use Case Scenario</i>	15
2.18.2 <i>Use Case Diagram</i>	16
2.18.3. ERD (<i>Entity Relationship Diagram</i>)	18
2.18.4 DFD (<i>Data Flow Diagram</i>)	18
2.19. Pengujian Perangkat Lunak	20
2.19.1. <i>Blackbox Testing</i>	20
2.19.2. Pengujian Alpha.....	20
2.19.3 Pengujian Beta.....	20
2.19.4 Technology Acceptance Model (TAM)	20
2.19.5. Skala Pengukuran.....	22
2.20. Normalisasi.....	23
2.21. Metode <i>Waterfall</i>	23
BAB 3.....	33
ANALISIS DAN PERANCANGAN	33
3.1 Metode Penelitian.....	33
3.2. Analisis	33
3.2.1. Analisis Kebutuhan (<i>Requirements Definition</i>)	34
3.3. Perancangan (<i>Design</i>).....	39
3.3.1 Metode Pengembangan Sistem.....	40
3.3.2 Siklus Hidup Pengembangan Sistem Informasi.....	41
3.3.3 Analisis Sistem.....	42
3.4. Gambaran Umum Sistem	42
3.6. Perancangan Sistem.....	43
3.7. Perancangan Basis Data	64
3.8. Desain Antarmuka	69
3.8.1. Halaman Beranda <i>User</i>	69
3.8.2. Halaman <i>About</i>	70
3.8.3. Halaman Pelayanan.....	70

3.8.4. Halaman Berita	71
3.8.5. Halaman Kontak	71
3.8.6. Halaman <i>Login</i>	72
3.8.7 Halaman Registrasi	72
3.8.8. Halaman <i>Dashboard User</i>	73
3.8.9. Halaman <i>Dashboard Admin</i>	73
3.8.10. Halaman <i>Dashboard</i> Kepala Desa	74
3.8.11. Halaman Pengajuan Surat.....	74
3.8.12. Halaman Pengajuan Surat Saya.....	75
3.8.13. Halaman Data Pengajuan Surat.....	75
3.8.14. Halaman Tentang Desa	76
3.8.15. Halaman Manajemen Akun.....	76
3.8.16. Halaman Galeri	77
3.8.17. Halaman Ganti <i>Background</i>	77
3.8.18. Halaman Laporan Pengajuan Surat	78
3.8.19. Halaman Tanda Tangan Virtual	78
3.9. Rencana Pengujian	79
3.9.1. <i>Alpha Testing</i>	79
3.9.2 <i>Beta Testing</i>	85
BAB IV	89
JADWAL KERJA.....	89
DAFTAR PUSTAKA	91
LAMPIRAN.....	93
DOKUMENTASI.....	103

DAFTAR TABEL

Tabel 2.1 Penelitian Terdahulu	24
Tabel 3.1 Kebutuhan Fungsional	35
Tabel 3.2 Kebutuhan Non Fungsional.....	38
Tabel 3.3 Definisi <i>Use Case Diagram</i>	44
Tabel 3.4 Use Case Scenario Mendaftar Sebagai User	49
Tabel 3.5 Use Case Scenario Mengajukan Surat Pengantar.....	51
Tabel 3.6 Use Case Scenario Login Admin	54
Tabel 3.7 Use Case Scenario Penandatanganan Surat kepala Desa.....	55
Tabel 3.8 Use Case Scenario Login Kepala Desa	56
Tabel 3.9 Use Case Scenario Tambah Akun Baru.....	57
Tabel 3.10 Tabel <i>Auth</i>	66
Tabel 3.11 Tabel <i>User</i>	67
Tabel 3.12 Tabel Jenis Surat.....	67
Tabel 3.13 Tabel Surat.....	67
Tabel 3.14 Tabel Profil Desa.....	68
Tabel 3.15 Tabel Galeri.....	68
Tabel 3.16 Tabel Berita	68
Tabel 3.17 Tabel Kontak	69
Tabel 3.18 Tabel Pengajuan Surat	69
Tabel 3.19 Tabel Pengujian Alpha Halaman Beranda Sistem	79
Tabel 3.20 Tabel Pengujian Alpha <i>Login</i>	80
Tabel 3.21 Tabel Pengujian Alpha Halaman <i>Dashboard User</i>	81
Tabel 3.22 Tabel Pengujian Alpha Halaman <i>Dashboard Admin</i>	83
Tabel 3.23 Tabel Pengujian Alpha Halaman <i>Dashboard Kepala Desa</i>	84
Tabel 3.24 Tabel Penelitian Dengan Kuisisioner	86
Tabel 3.25 Tabel Skala.....	87
Tabel 4.1 Jadwal Kerja Penelitian.....	89

DAFTAR GAMBAR

Gambar 2.1 Proses <i>Encryption Data</i>	12
Gambar 2.2 Proses Decryption Data	12
Gambar 2.3 Alur Proses <i>Transparent Data Encryption</i>	14
Gambar 2.4 Use Case Sistem Admin Bank	17
Gambar 3.1 <i>SDLC Waterfall Model</i>	40
Gambar 3.2 Gambaran Umum Sistem.....	42
Gambar 3.3 <i>Use Case Diagram</i> Sistem Informasi Pelayanan Desa	43
Gambar 3.4 <i>Diagram Activity Login Admin</i>	45
Gambar 3.5 <i>Diagram Activity Login Kepala Desa</i>	46
Gambar 3.6 <i>Diagram Activity Login User</i>	46
Gambar 3.7 <i>Diagram Activity Pengajuan Surat</i>	47
Gambar 3.8 <i>Diagram Activity Penerimaan Pengajuan Surat</i>	47
Gambar 3.9 <i>Diagram Activity Tambah Akun</i>	48
Gambar 3.10 <i>Diagram Activity Kelola Akun</i>	48
Gambar 3.11 <i>Sequence Diagram Login Admin</i>	59
Gambar 3.12 <i>Sequence Diagram Login User</i>	60
Gambar 3.13 <i>Sequence Diagram Pendaftaran Akun User</i>	60
Gambar 3.14 <i>Sequence Diagram Login Kepala Desa</i>	61
Gambar 3.15 <i>Sequence Diagram Tampilkan List Akun</i>	61
Gambar 3.16 <i>Sequence Diagram Tambah Akun</i>	62
Gambar 3.17 <i>Sequence Diagram Ubah Akun</i>	62
Gambar 3.18 <i>Sequence Diagram Hapus Akun</i>	63
Gambar 3.19 <i>Sequence Diagram Lihat Pengajuan Surat</i>	63
Gambar 3.20 <i>Sequence Diagram Persetujuan Pengajuan Surat</i>	64
Gambar 3.21 <i>Entity Relationship Diagram</i>	64
Gambar 3.22 Halaman Beranda	70
Gambar 3.23 Halaman <i>About</i>	70
Gambar 3.24 Halaman Pelayanan	71
Gambar 3.25 Halaman Berita.....	71
Gambar 3.26 Halaman Kontak.....	72

Gambar 3.27 Halaman <i>Login</i>	72
Gambar 3.28 Halaman Registrasi	73
Gambar 3.29 Halaman <i>Dashboard User</i>	73
Gambar 3.30 Halaman <i>Dashboard Admin</i>	74
Gambar 3.31 Halaman <i>Dashboard</i> Kepala Desa	74
Gambar 3.32 Halaman Pengajuan Surat	75
Gambar 3.33 Halaman Pengajuan Surat Saya	75
Gambar 3.34 Halaman Data Pengajuan Surat.....	76
Gambar 3.35 Halaman Tentang Desa	76
Gambar 3.36 Halaman Manajemen Akun	77
Gambar 3.37 Halaman Galeri	77
Gambar 3.38 Halaman Ganti <i>Background</i>	78
Gambar 3.39 Halaman Laporan Pengajuan Surat	78
Gambar 3.40 Halaman Tanda Tangan Virtual.....	79

BAB I

PENDAHULUAN

1.1. Latar Belakang

Teknologi informasi adalah teknologi yang menggabungkan antara komunikasi dan komputasi untuk melakukan tugas-tugas informasi sehingga arus informasi dapat berjalan dan terintegrasi dengan baik. Teknologi informasi berkembang di berbagai aspek kehidupan, mulai dari personal hingga instansi. Pada sebuah instansi baik negeri maupun swasta, teknologi informasi sangat dibutuhkan untuk optimalisasi segala proses yang berkaitan dengan pembangunan dan perbaikan sistem, salah satu bentuk optimalisasi tersebut adalah penerapan sistem informasi dengan kriteria sistem informasi adalah fleksibel, efektif dan efisien.

Kantor Desa Mandalahaji merupakan salah satu lembaga pemerintahan yang membutuhkan suatu aplikasi yang menunjang serta bisa memaksimalkan pelayanan pada masa pandemi Covid-19 khususnya dalam melakukan pengajuan surat-surat, diantaranya adalah pembuatan Surat Keterangan Usaha, Surat Keterangan Kematian, Surat Keterangan Nikah, Surat Keterangan Tidak Mampu, Surat Keterangan Kelakuan Baik, Surat Keterangan Kelahiran, Surat Keterangan Pindah dan lain sebagainya. Pada prosesnya apabila warga akan mengajukan surat ataupun izin Kepala Desa, akan melalui proses dan alur yang panjang sehingga akan menjadikan pelayanan berjalan lambat. Untuk itu perlu dibangun sebuah sistem yang dapat membantu permasalahan pelayanan yang terjadi agar proses pelayanan yang lambat dan terhambat dikarenakan banyaknya pembatasan di tengah pandemi Covid-19 dapat teratasi. Syarat utama untuk mengajukan surat yaitu menggunakan *fotocopy* KTP (Kartu Tanda Penduduk) dan KK (Kartu Keluarga) kedua data tersebut digunakan sebagai acuan validasi data yang akan dicek apakah benar yang mengajukan merupakan warga Desa Mandalahaji, jika iya maka akan dilayani. KTP dan KK merupakan data yang tidak boleh sembarang orang gunakan, maka dari itu proses inputan data KTP dan KK digunakan sebuah teknologi yang akan mengamankan kedua data tersebut yaitu menggunakan *Transparent Data Encryption* (TDE). Dengan TDE data akan dienkripsi menggunakan sebuah *master key*, dan *master key* ini akan disimpan dalam sebuah *wallet* dalam komputer. Sehingga walaupun kita mengetahui *master key* nya, namun *key* untuk *wallet* tidak, tetap tidak akan

bisa membuka data yang terenkripsi. Sehingga apabila terjadi pencurian data sekalipun, tetap tidak akan bisa di dekripsi tanpa *key* diatas. Menerapkan enkripsi pada sebuah data adalah salah satu cara yang bisa digunakan untuk melindungi data karena data ini akan dienkripsi dan dirubah formatnya dari *plaintext* ke *Chipertext*. (Wibowo, I., Susanto, B., & Karel, J., 2011).

Transparent Data Encryption (TDE) adalah salah satu jenis enkripsi sebagai kebalikan dari enkripsi manual, membuat proses enkripsi menjadi lebih sederhana dengan meletakkan enkripsi didalam databasenya sendiri. Dimana aplikasi bisa melanjutkan pekerjaannya tanpa membutuhkan *database trigger*, *view* dan aplikasi lain yang digunakan untuk solusi enkripsi database. Data secara otomatis dienkripsi ketika ditulis pada file *database* di disk. TDE melakukan enkripsi dan dekripsi secara *realtime* (waktu nyata) untuk data dan *file log*. Enkripsi menggunakan *master key* dan *master key* ini akan disimpan dalam sebuah *wallet*. TDE memungkinkan sistem mengenkripsi data menggunakan algoritma AES dan 3DES tanpa mengubah sistem dan struktur aplikasinya. (Microsoft, 2019).

Dari sisi inilah bagaimana membangun sebuah sistem pelayanan administrasi yang baik dan memiliki *database* yang terjamin keamanannya, dikarenakan pada proses pengajuan surat-surat administrasi ini memerlukan data berupa No NIK dan No KK maka dari itu perlunya diterapkan keamanan pada sisi *database* dengan menggunakan Teknologi *Transparent Data Encryption* pada sistem yang akan dibangun mengingat banyaknya kejadian kehilangan banyak data, penyalahgunaan data dan pencurian. Perkembangan penggunaan *database* berbarengan pula dengan keharusan pemahaman akan keamanan *database*. (Murray, M. C., 2010) Masih banyak Lembaga atau instansi yang menyepelekan keamanan *database*, sehingga data yang ada bisa diakses oleh orang yang tidak berhak mendapatkannya. Maka dalam penelitian ini diangkat sebuah judul “SISTEM INFORMASI PELAYANAN DI DESA MANDALAHAJI DENGAN MENGAPLIKASIKAN TEKNOLOGI *TRANSPARENT DATA ENCRYPTION* UNTUK KEAMANAN *DATABASE*” dengan di lakukannya penelitian ini diharapkan dapat membantu masyarakat Desa Mandalahaji dalam mendapatkan pelayanan yang optimal sekaligus membantu kantor Desa Mandalahaji dalam memberikan pelayanan ditengah pandemi Covid-19.

1.2. Rumusan Masalah

Setelah mengkaji latar belakang masalah yang telah diuraikan diatas, maka dapat ditarik rumusan masalah yaitu :

1. Bagaimana cara mengajukan pembuatan surat keperluan warga secara efisien dari segi waktu?
2. Bagaimana mengatasi dampak yang dirasakan pada pelayanan desa mandalahaji dimasa pandemi covid-19?
3. Bagaimana agar sistem pelayanan ini dapat mengamankan data yang ada pada sistem pelayanan Desa Mandalahaji?

1.3. Tujuan Penelitian

Sesuai dengan rumusan masalah, adapun tujuan yang ingin dicapai adalah :

1. Dibangunnya sistem informasi ini maka proses pembuatan surat akan lebih mudah mulai dari pengajuan, pemrosesan dan pencetakan.
2. Dengan dibangunnya sistem informasi ini maka pembatasan pelayanan yang terjadi di Kantor Desa Mandalahaji akan membantu mempermudah masyarakat dalam mendapatkan pelayanan dari Kantor Desa Mandalahaji.
3. Sistem informasi yang dibangun menggunakan *Transparent Data Encryption* untuk mengamankan data.

1.4. Manfaat Penelitian

Penulis berharap penelitian ini dapat bermanfaat bagi instansi terkait dan penulis:

1.4.1. Manfaat Bagi Instansi

Memudahkan Desa Mandalahaji dalam memberikan akses pelayanan kepada masyarakat mengenai hal-hal yang berkaitan dengan pelayanan yang ada di Kantor Desa Mandalahaji.

1.4.2. Manfaat Bagi Penulis

Menerapkan ilmu yang telah penulis dapatkan selama menempuh Pendidikan S1 (Strata Satu) di Sekolah Tinggi Teknologi Bandung pada permasalahan dikehidupan nyata khususnya pada sistem informasi.

1.5. Ruang Lingkup

Untuk mendapatkan penelitian seperti yang diharapkan dan penelitian yang terarah, maka ruang lingkup permasalahan dalam penelitian ini adalah sebagai berikut:

1. Sistem Informasi ini dibuat di Kantor Desa Mandalahaji.
2. Sistem informasi menyediakan layanan berupa pengajuan Surat Keterangan Usaha, Surat Keterangan Kematian, Surat Keterangan Nikah, Surat Keterangan Tidak Mampu, Surat Keterangan Kelakuan Baik, Surat Keterangan Kelahiran, Surat Keterangan Pindah dan lain sebagainya, untuk kepentingan warga/masyarakat di Desa Mandalahaji.
3. Keamanan *database* menggunakan teknologi TDE untuk kebutuhan keamanan data.

1.6 Sistematika Penulisan

Dalam penulisan skripsi ini, penulis membagi materi pembahasan dalam 5 bab, dimana dari setiap bab nya dibagi-bagi menjadi beberapa sub bab. Pembagian bab dalam penelitian ini dibuat secara sistematika.

1. BAB 1 Pendahuluan

Pada bab ini menguraikan latar belakang, rumusan masalah, tujuan penelitian, manfaat penelitian, dan batasan masalah dalam penyusunan laporan skripsi.

2. BAB 2 Tinjauan Pustaka

Pada bab ini membahas tentang dasar teori, penelitian terdahulu yang terkait dan dalam bab 2 ini juga dipaparkan integrasi keilmuan penelitian dengan sistem pakar dengan berbagai metode.

3. BAB 3 Metodologi Penelitian

Pada bab ini membahas tentang metodologi penelitian serta Langkah langkah yang dilakukan dalam melakukan penelitian.

4. BAB 4 Hasil dan Pembahasan

Pada bab ini meliputi hasil berupa penjelasan dan menjelaskan mengenai proses analisis yang dilakukan dan membahas hasil yang telah diperoleh.

BAB II

TINJAUAN PUSTAKA

2.1. Pengertian E-government

Electronic government merupakan suatu proses sistem pemerintahan dengan cara memanfaatkan atau menggunakan ICT (*Information, Communication and technology*) sebagai alat untuk memberikan kemudahan proses komunikasi dan transaksi kepada warga masyarakat, organisasi bisnis dan antara Lembaga pemerintah serta stafnya. Sehingga dapat dicapai efisiensi, efektivitas, transparansi dan pertanggung jawaban pemerintah kepada masyarakatnya. Konsep pengembangan *e-Government* menentukan prioritas pengembangan *e-Government* suatu Lembaga pemerintah, menyangkut hubungan *Government to Government (G2G)*, *Government to Business (G2B)* dan *Government to Citizen (G2C)* (Hartono, Dwiarso Utomo,dkk, 2017).

2.2. Pengertian Sistem Informasi

Sistem informasi adalah suatu rangkaian sistem yang dikelompokkan dalam suatu organisasi yang terdiri dari sekumpulan komponen baik yang berbasis computer maupun manual yang dibuat untuk menghimpun dan menyiapkan data-data yang berisikan informasi keluaran untuk pemakai, atau sekumpulan perangkat keras dan perangkat lunak yang dihubungkan untuk menciptakan dan memproses data menjadi informasi yang berguna (Maria, Endang, 2016).

2.3. Pengertian Website

, “*Website* merupakan kumpulan halaman-halaman yang digunakan untuk menampilkan informasi teks, gambar diam atau gerak, animasi, suara, dan atau gabungan dari semuanya, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait, yang masing-masing masing dihubungkan dengan jaringan-jaringan halaman”. (Humaira, 2015).

2.4. Pengertian Database (Basis Data)

Database adalah suatu aplikasi yang menyimpan sekumpulan data. Setiap *database* mempunyai API tertentu untuk membuat, mengakses, mengatur, mencari, menyalin data yang ada di dalamnya (Enterprise, Jubille, 2015).

Basis data terdiri atas 2 kata, yaitu *Basis* dan *Data*. Basis kurang lebih dapat diartikan sebagai markas atau gudang, tempat bersarang/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.

Basis data sendiri dapat didefinisikan dalam sejumlah sudut pandang seperti :

- a. Himpunan kelompok data (arsip) yang saling berhubungan yang diorganisasi sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah.
- b. Kumpulan data yang saling berhubungan yang disimpan secara bersama sedemikian rupa dan tanpa pengulangan yang tidak perlu, untuk memenuhi berbagai kebutuhan.
- c. Kumpulan file/table/arsip yang saling berhubungan yang disimpan dalam media penyimpanan elektronik (Fatansyah, 2017).

2.4.1. MySQL

Menurut (Arif, 2019) MySQL adalah sebuah *software* atau perangkat lunak sistem manajemen berbasis data SQL atau juga *multiuser* dan DBMD *Multithread*. Pada dasarnya, MySQL ini sebenarnya adalah turunan yang berasal dari salah satu konsep utama dalam *database* yang memang telah ada sebelumnya yaitu SQL atau *Structured Query Language*.

MySQL adalah salah satu jenis *database server* yang sangat terkenal. Kepopulerannya di sebabkan MySQL menggunakan SQL sebagai bahasa dasar untuk mengakses databasenya. MySQL bersifat free pada berbagai *platform* (kecuali pada Windows, yang bersifat *shareware* atau anda perlu membayar setelah melakukan evaluasi dan memutuskan untuk digunakan untuk keperluan produksi). MySQL dilisensi di bawah *GNU General Public License* (GPL). Dengan adanya keadaan seperti itu maka anda dapat menggunakan *software database* ini dengan bebas tanpa harus takut dengan lisensi yang ada (Sultiano, 2019).

2.5. PHP

PHP merupakan bahasa berbentuk *script* yang di tempatkan dalam *server* dan proses di *server* hasilnya akan dikirimkan ke *client*, tempat pemakai menggunakan

browser, php di kenal sebagai sebuah bahasa *scripting*, yang menyatu dengan *tag-tag* HTML, di eksekusi di *server*, digunakan untuk membuat halaman *web* yang dinamis seperti halnya *active server pages* (ASP) atau *java server pages* (JPS), php juga dapat di lihat sebagai pilihan lain dari ASP.NET/C#VB.NET *Micro system*, dan CGI/perl. Contoh aplikasi lain yang lebih kompleks berupa CMS yang di bangun menggunakan PHP adalah *mambo*, *joomla*, *postnuke*, *xaraya* dan lain-lain (Yuli Kartika,2017).

2.6. HTML

Menurut Fiber Siregar dan Muhammad (2009), HTML adalah kepanjangan dari *Hyper Text Markup Language* yang merupakan bahasa yang digunakan untuk membuat suatu situs *web* atau *home page*, setiap dokumen dalam web ditulis dengan format HTML. Semua format dokumen *hyperlink* yang dapat di klik gambar, dokument, multimedia, *form* yang dapat diisi di dasarkan atas HTML. Setiap menggunakan HTML dalam membuat *web* diperlukan juga *syntax* PHP, dengan menggunakan PHP *user* tidak akan melihat kode-kode PHP yang telah ditulis tersebut di dalam *browser*. Umumnya semua dokumen *web* dibagi menjadi dua. *Section* (bagian), yaitu *section head*, dan *section body*.

2.7. CSS

Cascading Style Sheet atau biasa disingkat CSS, merupakan aturan untuk mengatur beberapa komponen dalam sebuah *web* sehingga menjadi lebih terstruktur dan seragam. Umumnya CSS dipakai untuk memformat tampilan halaman *web* yang dibuat dengan bahasa HTML dan XHTML.

CSS dapat mengendalikan ukuran gambar, warna teks, warna tabel, ukuran *border* dan warna *border*, warna *hyperlink*, warna *mouse over*, spasi antar pragraf, spasi antar teks, margin kiri, kanan, atas, bawah, dan lain-lain. Jadi, dengan CSS kita bisa membuat halaman *website* kita menjadi lebih indah dan nyaman dilihat.

2.8. XML

XML merupakan singkatan dari *Extensible Markup Language*, adalah bahasa *markup* untuk keperluan umum yang disarankan oleh W3C untuk membuat dokumen *markup* keperluan pertukaran data antar sistem yang beraneka ragam. XML merupakan kelanjutan dari HTML (*Hyper Text Markup Language*) yang merupakan bahasa standar untuk melacak internet. XML didesain mampu untuk menyimpan data secara ringkas dan mudah diatur.

2.9. ASP

Active Server Pages atau disingkat ASP adalah salah satu bahasa pemrograman *web* untuk menciptakan halaman *web* yang dinamis. ASP merupakan salah satu bentuk teknologi yang disediakan oleh Microsoft. ASP di proses melalui *web server* dan hasil proses ini menghasilkan HTML yang akan dikirimkan melalui penjelajah *web* (*browser*). ASP bekerja pada *web server* dan merupakan *server side scripting*. Dengan adanya ASP, maka kita bisa mengubah *website* dari HTML yang statis menjadi *website* yang dinamis.

2.10. JavaScript

JavaScript adalah bahasa *script* yang berjalan pada sisi klien atau *client side*. JavaScript merupakan bahasa skrip yang populer di internet dan dapat bekerja disebagian web browser populer seperti Ineternet Explorer, Mozzilla Firefox, Google Chrome dan lain-lain. Kode javascript biasanya ditulis dalam bentuk fungsi yang ditaruh ditag <head> yang dibuka dengan tag <script type="text/javascript">. Kode JavaScript juga bisa diletakkan di *file* tersendiri yang berekstensi .js (singkatan dari JavaScript). Dengan JavaScript kita bisa membuat *website* yang lebih interaktif, sehingga *website* menjadi lebih menarik.

2.11. Framework

Framework adalah kumpulan perintah atau fungsi dasar yang membentuk suatu sistem tertentu agar tersusun dan terstruktur dengan rapi sehingga memudahkan para *programmer* untuk membuat sebuah aplikasi *web*. Untuk saat ini *framework* terbagi dua yaitu *framework* PHP dan *framework* CSS.(Gasan dkk,2018)

2.12. Bootstrap

Bootstrap adalah sebuah *framework* css yang dapat digunakan untuk mempermudah membangun tampilan *web*. Artinya, tampilan *web* yang dibuat oleh *bootstrap* akan menyesuaikan ukuran layar dari *browser* yang kita gunakan baik di *desktop*, *tablet* ataupun *mobile device*. Fitur ini bisa diaktifkan ataupun dinon-aktifkan sesuai dengan keinginan kita sendiri. Sehingga, kita bisa membuat *web* untuk tampilan *desktop* saja dan apabila dirender oleh *mobile browser* maka tampilan dari *web* yang kita buat tidak bisa beradaptasi sesuai layar. *Bootstrap* pertama kali di kembangkan pada pertengahan 2010 di Twitter oleh Mark Otto dan Jacob Thornton. Saat ini *Bootstrap* dikembangkan secara *open source* dengan lisensi MIT. (Fadul, 2018)

2.13. Aspek Keamanan

Membahas keamanan maka tidak akan terlepas dari yang namanya aspek keamanan.

Berikut adalah beberapa aspek keamanan terkait dengan keamanan :

a. Confidentiality

Terkait dengan Tindakan pencegahan akses dari pihak-pihak yang tidak berhak melakukan pengaksesan terhadap informasi, umumnya terkait dengan pemberian informasi kepihak lain

b. Integrity

Terkait dengan jaminan kelengkapan informasi dan menjaga dari kerusakan ancaman lain yang mengakibatkan perubahan informasi dengan kata lain menjamin keaslian sebuah informasi (utuh, akurat, dan belum dimodifikasi oleh pihak yang tidak berhak).

c. Availability

Terkait dengan jaminan pengguna dapat melakukan pengaksesan informasi kapanpun tanpa adanya gangguan. Pengguna dalam hal ini bisa jadi manusia, atau komputer yang tentunya memiliki otorisasi untuk mengakses sistem informasi, umumnya berhubungan dengan ketersediaan informasi ketika dibutuhkan.

d. Authentication

Penerima informasi dapat memastikan keaslian pesan yang didapat tersebut berasal dari orang yang seharusnya.

e. Authority

Informasi melalui sistem jaringan tidak dapat dimodifikasi oleh pihak yang tidak berhak (penyusup).

f. privacy

sifatnya lebih ke data-data yang *private* (pribadi) dari *user*.

2.14. Metodologi Keamanan Sistem

Terkait dengan metodologi keamanan sistem informasi maka keamanan dibagi menjadi 4 (empat) level, yaitu:

a. Keamanan level 0

Keamanan fisik, merupakan keamanan tahap awal dari *computer security*. jika keamanan fisik tidak terjaga dengan baik, maka data-data bahkan *hardware computer* sendiri tidak dapat diamankan.

b. Keamanan level 1

terdiri dari *database*, *data security*, keamanan dari PC itu sendiri, *device* dan aplikasi. Contohnya : jika kita ingin database aman, maka kita harus memperhatikan dahulu apakah aplikasi yang dipakai untuk membuat desain *database* tersebut merupakan aplikasi yang sudah diakui keamanannya. Selain itu kita harus memperhatikan sisi lain yaitu *data security*. *data security* adalah cara untuk mendesain *database*

c. Keamanan level 2

Network security / keamanan jaringan. *Computer* yang terhubung dengan jaringan sangat rawan dalam masalah keamanan, oleh karena itu keamanan level 2 harus dirancang supaya tidak terjadi kebocoran jaringan, akses ilegal yang dapat merusak keamanan data tersebut.

d. Keamanan level 3

Information security / keamanan informasi yang kadang kala tidak begitu diperdulikan oleh administrator, seperti membarikan *password* ke teman, atau menuliskannya dikertas, maka bisa menjadi sesuatu yang fatal jika informasi tersebut diketahui oleh orang yang tidak bertanggung jawab.

e. Keamanan level 4

Merupakan keamanan jaringan secara keseluruhan dari komputer. Jika level 1 – 3 sudah dapat dikerjakan dengan baik maka otomatis keamanan untuk level 4 sudah terpenuhi.

Dilihat dari metodologi keamanan sistem terdiri dari level 0 – 4 yang harus diamankan, namun dalam penelitian ini penulis akan mencoba memfokuskan keamanan pada level 1 (satu), yaitu pengamanan dari sisi sistem *database* atau basisdata dengan cara menerapkan Teknik pengkodean data atau enkripsi.

2.15. Enkripsi

Enkripsi merupakan sebuah cara untuk menjadikan data-data atau informasi tidak dapat terbaca oleh pihak-pihak yang tidak berhak atau tidak memiliki hak akses. Tujuan utama dari enkripsi adalah selain menyembunyikan data atau informasi yang

terkandung didalamnya juga untuk menjaga integritas data/informasi pada saat ditransmisikan.

Data disandikan (*encrypted*) dengan menggunakan sebuah kunci (*key*) dan untuk membuka (*decrypt*) data tersebut digunakan sebuah kunci yang sama dengan kunci untuk mengenkripsi (kasus *key private key cryptography*) atau dengan kunci berbeda (kasus *public key cryptography*)(Sudrajat, 2016).

Type dasar dari enkripsi dibagi menjadi (Sudrajat, 2016):

a. Enkripsi manual

Enkripsi tipe ini sepenuhnya dilakukan oleh *user* dimana *user* harus memilih secara manual objek mana yang akan di enkripsi dan kemudian menjalankan *command* / perintah khusus untuk melakukan enkripsi dan deskripsi *object* tersebut.

b. Enkripsi Semi-Transparent

Enkripsi jenis ini disebut juga enkripsi “*On The Fly*”. Enkripsi ini beroperasi tidak secara permanen, tapi sebelum dan sesudah akses dilakukan pada object-object rahasia atau ketika operasi *read/write*.

c. Enkripsi Transparent

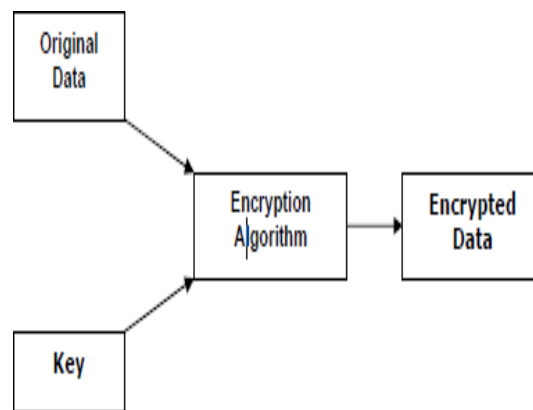
Enkripsi ini bisa dikatakan sebagai kebalikan dari enkripsi manual. Proses enkripsi dan deskripsi dilakukan pada level rendah, secara permanen, ketika semua operasi *read/write*, sehingga data yang dienkripsi selalu disimpan dalam bentuk enkripsi. Dari sisi prinsip-prinsip umum keamanan enkripsi jenis ini adalah tipe yang paling aman dan mudah. *Transparent Data Encryption* membuat proses enkripsi sederhana dengan meletakkan enkripsi di dalam *database* nya sendiri. Dimana aplikasi bisa melanjutkan pekerjaannya tanpa menggunakan database trigger, view, dan aplikasi lain yang digunakan solusi enkripsi database tradisional. *Data* secara otomatis dienkripsi ketika ditulis pada *file database* di *disk*. *Data* secara otomatis di dekripsi untuk semua *database user* setelah dilakukan autentikasi pada *database* dan melewati semua pemeriksaan autentikasi tambahan. Tahapan pemeriksaan ini termasuk juga untuk memastikan apakah *user* mempunyai hak untuk melakukan perintah tertentu (*select, update* dan *delete* pada tabel aplikasi).

Enkripsi *transparent* selain sebagai tindakan pencegahan untuk membantu mengamankan *database* juga dapat digunakan pada kasus apabila terjadi pencurian pada media fisik / *hardware* / media *backup* pada data yang sensitif di level sistem operasi.

Proses Enkripsi :

Cara kerja enkripsi dilakukan dengan menambahkan kode karakter teks sumber dengan teks kunci (*script source code*/algoritma). Membandingkan dengan kunci/*key* yang sudah kita tentukan dengan data sumber dan lakukan perubahan.

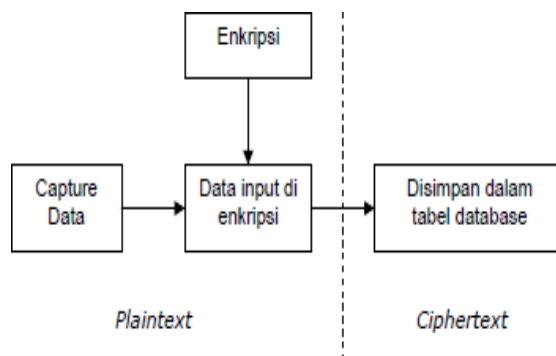
Proses dari enkripsi data dapat dilihat pada gambar 2.1.(Rijayanti Rita, 2012)



Gambar 2.1. Proses *Encryption Data*

Deskripsi adalah kebalikan dari proses enkripsi yaitu proses konversi data yang sudah di enkripsi (*chipertext*) dikembalikan kebentuk aslinya (*Original Plaintext*) sehingga dapat dibaca / dimengerti Kembali.

Proses dari sebuah dekripsi data dapat dilihat pada gambar 2.2.



Gambar 2.2. Proses *Decryption Data*

2.16. *Transparent Data Encryption*

Transparent Data Encryption(TDE) menyediakan kriptografi yang transparan pada *user* yang sah tetapi tidak pada penyusup dari luar maupun dari dalam (Arif Budiman, Noviardi, 2016). Enkripsi ini digunakan untuk kasus apabila terjadi pencurian *hardware* atau *media backup* atau *unauthorized access* pada *data* yang sensitif di level sistem operasi. Salah satu cara untuk mengatasi pencurian media adalah mengenkripsi data yang sensitif di dalam database dan menyimpan *encryption key* nya di lokasi yang terpisah. Karena memang tujuan utamanya dari TDE ini adalah untuk memberikan keamanan data sampai ke level baris didalam sebuah *database*. Tetapi harus dipertimbangkan keseimbangan antara dua konsep yang bertentangan dan kemudahan dimana aplikasi bisa mengakses *encryption keys* dan keamanan yang diperlukan bila terjadi pencurian *key*.

2.17. Konsep *Transparent Data Encryption*

Dalam menciptakan TDE pada sebuah *database*, perlu dipahami beberapa konsep berikut:

2.17.1. *Master Encryption Key*

Master Encryption Key adalah bagian dari arsitektur pengamanan dua tingkat yang melindungi kunci enkripsi. *Master Encryption Key* disimpan pada sebuah lokasi dalam *database* yang disebut *wallet*, sedangkan untuk membuka *wallet* diperlukan sebuah kunci */walletkey*.

2.17.2. *Wallet*

Wallet adalah sebuah lokasi yang disediakan didalam database yang digunakan untuk menyimpan Master key. *Wallet* sendiri mempunyai *Wallet key* yang dibutuhkan untuk membuka *wallet*. Master key tidak akan bisa diambil bila *wallet key* tidak ada. *Wallet* secara default akan tersimpan pada direktori *wallet*. Dan bisa dirubah kelokasi yang diinginkan.

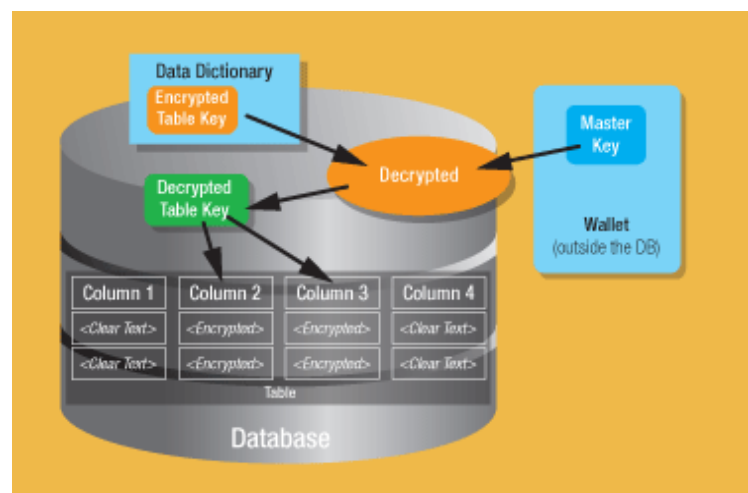
2.17.3. *Advanced Encryption Standard (AES)*

Merupakan algoritma standart yang dipakai untuk melakukan enkripsi. Algoritma ini terdiri dari beberapa jenis diantaranya AES256, AES192, AES128. Digunakan sesuai kebutuhan enkripsi pada *database*. Pengelompokan jenis AES ini adalah berdasarkan panjang kunci yang digunakan. Angka-angka dibelakang kata AES

menggambarkan panjang kunci yang digunakan pada tiap tiap AES, karena algoritma AES ini baik digunakan untuk enkripsi *file*, teks, ataupun *database*.

2.17.4. Cara Kerja TDE

TDE dapat diimplementasikan pada *object table* dan juga bisa diterapkan secara langsung pada *tablespace*. Pada saat sebuah *table* (bisa jadi satu kolom atau semua kolom) telah dienkripsi kemudian seorang *user* menginputkan *data* pada sebuah kolom tersebut, sistem *database* mengambil *master key* dari *wallet*, mendekripsi *encryption key* untuk *table* tersebut dari *data dictionary*, menggunakan *encryption key* pada nilai *input* dan menyimpan data yang dienkripsi pada *database*. Gambar 2.3., menjelaskan bagaimana proses enkripsi pada sebuah tabel.



Gambar 2.3. Alur Proses *Transparent Data Encryption*

Ketika *user* memilih kolom yang terenkripsi, sistem *database* secara transparan mengambil *table key* yang terenkripsi dari *data dictionary*, lalu mengambil *master key* dari *wallet*, dan mendekripsi *table key*. Kemudian database mendekripsi *data* yang terenkripsi pada *disk* dan mengembalikan nilai *clear text* pada *user*.

Dengan *data* yang terenkripsi ini, jika *data* dalam *disk* dicuri, maka *data* yang terenkripsi ini tidak bisa diambil tanpa *master key*, yang berada didalam *wallet* dan bukan dari bagian *data* yang dicuri. Bahkan jika *wallet* nya dicuri, *master key* nya tidak bisa diambil *wallet* tanpa *wallet password*. Oleh karena itu pencurinya tidak bisa mendekripsikan *data*, bahkan jika dia mencuri disknya atau melakukan *copy* pada *file data*.

Sampai sekarang ini satu-satunya *database* yang mendukung *Transparent Data Encryption* adalah *SQL server*, maka pada penelitian ini menggunakan sistem *database* tersebut untuk melakukan implementasi dan analisis.

2.18. UML (*Unified Modelling Language*)

Unified Modeling Language (UML) adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek. UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung. UML muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun, dan dokumentasi dari sistem perangkat lunak. UML hanya berfungsi untuk melakukan pemodelan.

2.18.1 *Use Case Scenario*

Use Case scenario merupakan penjelasan secara tekstual dari sekumpulan skenario interaksi. Setiap skenario mendeskripsikan urutan aksi/langkah yang dilakukan aktor ketika berinteraksi dengan sistem, baik yang berhasil maupun gagal. *UC scenario* dijelaskan secara tekstual dalam beberapa format tergantung kebutuhannya, yaitu singkat (*brief*), informal (*casual*), atau lengkap (*fully dressed*) (Larman, 2005), yang bisa dijelaskan dalam bentuk tabel dengan 1 kolom atau 2 kolom (Cockburn, 2000). Pada format singkat, penjelasan diberikan cukup 1 paragraf yang mengacu hanya pada skenario yang berhasil. Pada format informal, penjelasan diberikan dalam beberapa paragraf yang mencakup semua skenario, baik yang berhasil maupun gagal. Sedangkan, pada format lengkap, penjelasan dibuat secara detil disertai dengan bagian-bagian pendukung yang penting. Format terakhir ini yang banyak digunakan di dalam praktik. Bagian-bagian penting tersebut adalah (Larman, 2005):

- a. **aktor primer** (*primary actor*), yaitu aktor yang menginisiasi layanan sistem untuk mencapai tujuan dari aktor tersebut. Jumlah aktor *primer* dimungkinkan lebih dari 1.
- b. **prakondisi** (*preconditions*), yaitu kondisi spesifik yang harus terpenuhi sebelum sebuah UC bisa diinisiasi atau dieksekusi oleh aktor *primer*. Jumlah prakondisi bisa lebih dari 1 keadaan.

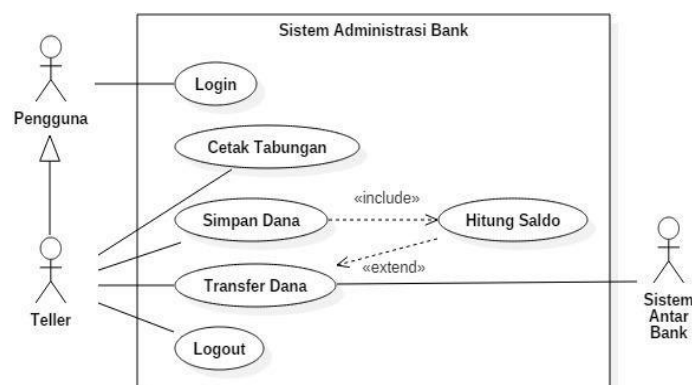
- c. **alur utama** (*main or basic flow*), yaitu jalur interaksi yang mengarahkan pada skenario yang berhasil sehingga tujuan aktor bisa terpenuhi. Jalur ini hanya terdiri dari 1 jalur saja.
- d. **alur alternatif** (*alternative flows*), yaitu jalur alternatif dari interaksi yang terjadi antar aktor dengan sistem yang mencakup pencabangan (pilihan) maupun skenario yang gagal sehingga tujuan aktor tidak terpenuhi. Jalur ini bisa terdiri dari lebih dari 1 jalur kemungkinan.
- e. **kondisi akhir** (*postconditions*), yaitu kondisi spesifik yang harus terjadi ketika UC berhasil dijalankan atau dieksekusi secara lengkap, sebagai representasi dari tujuan yang ingin dicapai oleh aktor *primer*. Jumlah kondisi akhir bisa lebih dari 1 keadaan.

2.18.2 Use Case Diagram

Diagram Use Case adalah gambaran grafis dari interaksi diantara unsur-unsur sistem. Hal ini digunakan dalam analisis sistem untuk mengidentifikasi, menjelaskan dan mengatur persyaratan sistem. *Diagram use case* ini menggambarkan proses apa saja yang dapat dilakukan oleh *actor* (Ginanjar, 2016).

Sebuah *use case diagram* menyatakan visualisasi interaksi yang terjadi antara pengguna (aktor) dengan sistem. Diagram ini bisa menjadi gambaran yang bagus untuk menjelaskan konteks dari sebuah sistem sehingga terlihat jelas batasan dari sistem. Ada 2 elemen penting yang harus digambarkan, yaitu aktor dan *use case*. Aktor adalah segala sesuatu yang berinteraksi langsung dengan sistem, bisa merupakan orang (yang ditunjukkan dengan perannya dan bukan namanya/personilnya) atau sistem komputer yang lain. Aktor dinotasikan dengan simbol gambar orang-orangan (*stick-man*) dengan nama kata benda di bagian bawah yang menyatakan peran/sistem. Aktor bisa bersifat primer, yaitu yang menginisiasi berjalannya sebuah *use case*, atau sekunder, yaitu yang membantu berjalannya sebuah *use case*. *Use case* dinotasikan dengan simbol elips dengan nama kata kerja aktif di bagian dalam yang menyatakan aktivitas dari perspektif aktor. Setiap aktor dimungkinkan untuk berinteraksi dengan sistem dalam banyak *use case*. Sebaliknya, setiap *use case* bisa dijalankan oleh lebih dari satu aktor. Antar aktor maupun antar *use case* bisa memiliki relasi, masing-masing dengan spesifikasi yang berbeda. Sebuah *use case*, disebut dengan *base use case*, bisa memiliki relasi dengan 1 atau lebih *use case* yang lain, disebut dengan *supplier use case*, dalam bentuk *extend* dan/atau *include*.

Relasi *extend* menyatakan bahwa fungsionalitas dari *base use case* bisa diperluas oleh *supplier use case*, jika dibutuhkan, di dalam aksekusi alur alternatif yang ada pada *use case scenario* dari *base use case*. Sedangkan, relasi *include* menyatakan bahwa fungsionalitas dari *base use case* selalu hanya bisa dipenuhi dengan bantuan dari *supplier use case* di dalam eksekusi alur utama yang ada pada *use case scenario* dari *base use case*. Dalam hal ini, relasi *include* dan *extend* tidak menjelaskan urutan eksekusi apapun antara *base use case* dan supplier UC, baik dalam alur utama maupun alternatif yang dijelaskan dalam *use case scenario* dari *base use case*. Selanjutnya, sebuah aktor, disebut aktor induk, bisa memiliki relasi *inheritance* dengan aktor yang lain, disebut aktor turunan, yang menyatakan bahwa sebuah aktor merupakan turunan dari aktor yang lain. Aktor turunan akan memiliki hak akses terhadap fungsionalitas sistem yang lebih luas dibandingkan dengan aktor induk. Gambar 1 mengilustrasikan sebuah *use case diagram* sederhana yang terdiri dari 2 buah aktor *primer*, yaitu Pengguna dan Teller, serta 1 aktor sekunder, yaitu Sistem Antar Bank. Sistem tersebut memiliki fungsionalitas yang direpresentasikan dalam 5 *use case*, yaitu Login, Cetak Tabungan, Simpan Dana, Transfer Dana, dan Hitung Saldo. Aktor Teller merupakan turunan dari aktor Pengguna, ketika Pengguna dinyatakan *valid* saat melakukan *login*. Siapapun bisa melakukan *login*, itulah yang disebut dengan Pengguna. Teller, saat mentransfer dana, bisa melibatkan proses perhitungan saldo jika transfer yang dilakukan menggunakan rekening nasabah (ditunjukkan dengan relasi *extend*). Di sisi lain, perhitungan saldo harus selalu dilakukan saat Teller melakukan penyimpanan dana di rekening nasabah (ditunjukkan dengan relasi *include*). Untuk mengakhiri akses, Teller melakukan logout.



Gambar 2.4. Sebuah Use Case sederhana Sistem Administrasi Bank(Parsial)

2.18.3. ERD (*Entity Relationship Diagram*)

Pengertian ERD (*Entity Relationship Diagram*) adalah pemodelan data utama yang membantu mengorganisasikan data dalam suatu proyek ke dalam entitas-entitas dan menentukan hubungan antar entitas. ERD digunakan untuk memodelkan struktur data dan hubungan antar data. Dengan ERD model dapat diuji dengan mengabaikan proses yang harus dilakukan. Selain itu dengan ERD kita akan dapat menjawab pertanyaan mengenai data apa yang kita perlukan serta bagaimana data yang satu berhubungan dengan data yang lain. (Ivananda, 2015)

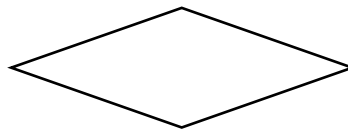
a. Entitas

Entitas adalah suatu objek yang dapat diidentifikasi dalam lingkungan pemakai.



b. Relasi

Relasi menunjukkan adanya hubungan diantara sejumlah entitas yang berbeda.



c. Atribut

Atribut yang mendeskripsikan karakter entitas.



d. Garis

Sebagai penghubung antar relasi dan entitas serta atribut.

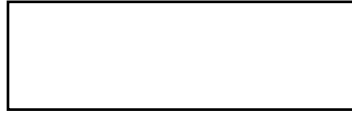


2.18.4 DFD (*Data Flow Diagram*)

Pengertian DFD (*Data Flow Diagram*) adalah suatu diagram yang menggunakan notasi-notasi untuk menggambarkan arus dari data sistem, yang penggunaannya sangat membantu untuk memahami sistem secara logika, terstruktur dan jelas. (Ivananda, 2015)

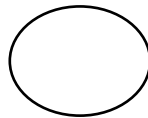
a. Eksternal Entity

Merupakan kesatuan di luar lingkungan sistem yang dapat berupa orang, organisasi atau sistem lain.



b. Proses

Menggambarkan entitas atau proses aliran data. Kegiatan atau kerja yang dilakukan oleh orang, mesin atau komputer dari hasil suatu arus data yang masuk ke dalam proses untuk dihasilkan arus data yang akan keluar dari proses.



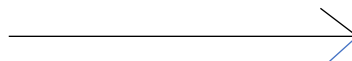
c. Symbol File

Menggambarkan tempat data berupa suatu file, arsip catatan, buku, simbol catatan di simpan.



d. Data Flow

Menjelaskan arah data atau informasi entitas, proses dan store.



2.19. Pengujian Perangkat Lunak

2.19.1. *Blackbox Testing*

Blackbox testing adalah tahap yang digunakan untuk menguji kelancaran program yang telah dibuat. Pengujian ini penting dilakukan agar tidak terjadi kesalahan alur program yang telah dibuat. (A.S., Rosa dan Salahuddin, 2015) ”*Blackbox testing* yaitu menguji perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program. (Agus Suandi, Fata Nidaul Khasanah, Endang R, 2017)

2.19.2. Pengujian Alpha

Pengujian Alpha dibagi ke menjadi dua yaitu pengujian Unit (Unit Testing) dan pengujian Integrasi (Integration Testing) (Rosa dan Salahuddin, 2018).

a. *Unit Testing*

Pengujian ini berfokus pada usaha verifikasi pada unit terkecil pada desain perangkat lunak (komponen atau modul perangkat lunak). Setiap unit diuji agar dapat diketahui apakah input dan output dari unit sudah sesuai dengan yang diinginkan. Karena dalam sebuah perangkat lunak banyak memiliki unit-unit kecil maka untuk menguji unit-unit kecil ini biasanya dibuat program kecil untuk menguji unit-unit perangkat lunak.

b. *Integration Testing*

Pengujian ini adalah suatu teknis yang sistematis untuk mengontrol struktur program seiring dengan menggabungkan fungsi program dengan antarmukanya. Pengujian terintegrasi bertujuan untuk mempergunakan komponen unit program yang sudah diuji dan membangun struktur yang telah didesain sebelumnya.

2.19.3 Pengujian Beta

Beta testing merupakan pengujian yang dilakukan di lingkungan pengguna tanpa kehadiran pihak pengembang aplikasi. Pengujian Beta merupakan pengujian bersifat langsung di lingkungan yang sebenarnya (Agus Suandi, 2017). Pengujian dilakukan dengan mengujicobakan aplikasi kepada pengguna untuk melihat kelayakan aplikasi berdasarkan penilaian pengguna.

2.19.4 Technology Acceptance Model (TAM)

TAM diperkenalkan pertama kali oleh Fred D. Davis pada tahun 1986, model ini merupakan adaptasi dari TRA (Theory of Reasoned Action) yang dibuat khusus

untuk pemodelan penerimaan pengguna terhadap sistem informasi (Flourensia Spty Rahayu, 2017). Menurut Davis yang dikutip oleh (Flourensia Spty Rahayu, 2017), tujuan TAM adalah untuk memberikan dasar untuk penelusuran pengaruh faktor eksternal terhadap kepercayaan sikap, dan tujuan pengguna. TAM menganggap bahwa 2 keyakinan individual, yaitu persepsi manfaat (*perceived usefulness*, disingkat PU) dan persepsi kemudahan pengguna (*perceived easy of use*, disingkat PEOU), adalah pengaruh utama untuk perilaku penerimaan komputer.

Variabel-variabel beserta indikator dari setiap variabel TAM yang digunakan pada penelitian ini adalah (Flourensia Spty Rahayu, 2017):

1. *Perceived Ease of Use* (PEU), Indikatornya terdiri dari:
 - a. X1 = Kemudahan dipelajari
 - b. X2 = Mudah dipahami
 - c. X3 = Mudah sehingga mahir
 - d. X4 = Mudah digunakan
 - e. X5 = Mudah dikendalikan
 - f. X6 = Mudah diingat
2. *Perceived of Fullness* (PU), Indikatornya terdiri dari:
 - a. Y1 = Lebih cepat
 - b. Y2 = Meningkatkan kinerja
 - c. Y3 = Meningkatkan Produktifitas
 - d. Y4 = Meningkatkan efektifitas
 - e. Y5 = Lebih mudah
 - f. Y6 = Bermanfaat
3. *Attitude Toward Using* (ATU), Indikatornya terdiri dari:
 - a. Y7 = Rasa senang
 - b. Y8 = Menikmati
 - c. Y9 = Rasa bosan
 - d. Y10 = Tidak Suka
4. *Behavioral Invention* (BI), Indikatornya terdiri dari:
 - e. Y11 = Menggunakan kapan saja
 - f. Y12 = Menggunakan kondisi apapun
 - g. Y13 = Menggunakan terus
 - h. Y14 = Berharap Menggunakan

5. *Actual Usage* (AU), Indikatornya terdiri dari:

- i. Y15 = Frekuensi penggunaan
- j. Y16 = Durasi penggunaan

2.19.5. Skala Pengukuran

Skala biasanya digunakan untuk mengecek dan menetapkan nilai suatu faktor kualitatif dalam ukuran-ukuran kuantitatif. Skala adalah alat yang disusun dan digunakan oleh peneliti untuk mengubah respons tentang suatu variabel yang bersifat kualitatif menjadi data kuantitatif (Sudaryono, 2019). Adapun macam-macam skala pengukuran, salah satunya skala Likert.

1. Skala *Likert*

Skala *Likert* digunakan untuk mengukur sikap, pendapat dan persepsi seseorang atau sekelompok tentang kejadian atau gejala sosial. Dalam gejala sosial telah ditetapkan secara spesifik oleh peneliti, yang selanjutnya disebut sebagai variabel penelitian. Dengan menggunakan skala *Likert*, maka variabel yang akan diukur dijabarkan menjadi dimensi, dimensi dijabarkan menjadi subvariabel kemudian subvariabel dijabarkan lagi menjadi indikator-indikator yang dapat diukur. Akhirnya indikator-indikator yang telah terukur ini bisa dijadikan titik tolak untuk membuat item instrumen yang berupa pertanyaan atau pernyataan yang perlu dijawab oleh responden. Setiap jawaban dihubungkan dengan bentuk pernyataan atau dukungan sikap yang diungkapkan dengan kata-kata sebagai berikut (Sudaryono, 2019):

- a. Sangat tinggi/ Sangat penting / Sangat benar
- b. Tinggi/ Penting/ Benar
- c. Cukup tinggi/ Cukup penting/ Cukup benar
- d. Rendah/ Kurang penting/ Salah
- e. Rendah sekali/ Tidak penting/ Sangat rendah

Skala yang ada pada skala *likert* dari skala 1-4:

- 1. Sangat Setuju (SS) = 4
- 2. Setuju (S) = 3
- 3. Tidak Setuju (TS) = 2
- 4. Sangat Tidak Setuju (STS) = 1

2.20. Normalisasi

Normalisasi merupakan suatu pendekatan sistematis untuk meminimalkan redundansi data pada suatu *database* agar *database* tersebut dapat bekerja dengan optimal. Jika anda seorang *database administrator* ketika terjadi sesuatu pada *database* seperti penurunan kinerja. (Rudiawan, 2018) Tujuan normalisasi adalah untuk menghilangkan dan mengurangi redundansi data dan tujuan yang kedua adalah memastikan dependensi data (Data berada pada tabel yang tepat). (Rudiawan, 2018) Normalisasi terdiri dari banyak bentuk, dalam ilmu basis data ada setidaknya 9 bentuk normalisasi yang ada yaitu 1NF, 2NF, 3NF, EKNF, BCNF, 4NF, 5NF, DKNF, dan 6NF. Namun dalam prakteknya dalam dunia industri bentuk normalisasi ini yang paling sering digunakan ada sekitar 5 bentuk (Rudiawan, 2018).

2.21. Metode *Waterfall*

Model modified waterfall adalah proses desain berurutan, sering digunakan dalam proses pengembangan perangkat lunak, dalam perkembangannya dia terus mengalir ke bawah (seperti air terjun) melalui tahapan *conception*, inisiasi, analisis, desain, konstruksi, pengujian, produksi / implementasi, dan pemeliharaan. Hasil dari pengembangan model tersebut telah dilakukan oleh Dr. Winston W. Royce pada prosiding yang ditulis olehnya dengan menambahkan validasi dan verifikasi pada tiap-tiap tahapan sehingga jika tahapan tersebut terjadi kesalahan maka tahapan akan kembali ke tahapan sebelumnya dan metode pengembangan akan terlihat seperti siklus yang berulang. Metode ini biasa disebut *Modified Waterfall*.

2.22. Penelitian Terdahulu

Penelitian mengenai perancangan sistem informasi dengan menerapkan *Transparent Data Encryption* (TDE) sudah pernah dilakukan oleh beberapa peneliti. Ringkasan penelitian terdahulu untuk mengetahui sejauh mana penelitian yang sudah ada dapat di lihat pada table 2.1.

Tabel 2.1. Penelitian Terdahulu

No	Judul	Peneliti, Media Publikasi dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelamahan	Perbandingan
1	E-VOTING BERBASIS WEBSITE PADA PEMILIHAN KADES DIRANTAU JAYA (<i>LAKE</i>) DENGAN KEMANAN DATA MENGGUNAKAN ENKRIPSI <i>BASE</i> 64	Harma Oktafia Lingga Wijaya, http://tunasbangsa.ac.id/cjurnal/index.php/jurasik Volume (2) No. 1 Juli 2017	Membuat suatu sistem aplikasi <i>E-Voting</i> pemilihan kades dirantau jaya (<i>lake</i>) berbasis <i>website</i> dengan keamanan <i>database</i> menggunakan enkripsi <i>base</i> 64	<ol style="list-style-type: none"> 1. Aplikasi <i>e-voting</i> yang dibangun dapat menjadi alternatif sebagai sarana untuk mempermudah proses pemilihan kepala desa di desa rantau jaya (<i>lake</i>) dengan mudah efektif dan akurat. 2. Aplikasi <i>e-voting</i> dapat mencegah adanya manipulasi suara dipihak panitia pemilihan <i>real time</i> dapat 	Pengujian kemanan <i>database</i> belum dipastikan dan diuji keamanannya, selain itu program yang dibuat kemungkinan hanya digunakan 1 kali pada pemilihan kades yang dilakukan vote 5 tahun sekali.	Perbandingan dengan penelitian penulis dengan jurnal ini adalah perbedaan antara fungsi di kedua program penelitian penulis lebih menuju ke pembuatan sistem informasi untuk pengelolaan

				dilihat oleh pemilih.		informasi dan pengajuan surat-surat penting warga sedangkan dengan jurnal perbandingan ini lebih kearah <i>e voting</i> untuk pemilihan kades yang hanya akan digunakan 5 tahun sekali
2	PENERAPAN KEAMANAN PENGGUNAAN DATA PADA DATABASE KEPEGAWAIAN	Arif Budiman,Noviardi, journal homepage : http://jurnal.stmik-amik-riau.ac.id SATIN - Sains dan Teknologi Informasi, Vol. 2, No. 2, Desember 2016	Membuat suatu sistem kepegawaian dengan keamanan <i>database</i>	TDE yang diterapkan pada <i>database</i> kepegawaian khususnya pada table pegawai dapat menjadi alternatif keamanan yang kuat. Dengan adanya	Pada jurnal ini diterapkan sistem enkripsi untuk mengamankan data yang mana data tersebut dipilih	Penerapan pada jurnal ini ialah menerapkan keamanan dengan dua

	MENGGUNAKAN TEKNIK <i>TRANSPARENT DATA ENCRYPTION</i> (STUDI KASUS SEKOLAH TINGGI PAKAYUMBUH)		menggunakan TDE.	proses ini secara otomatis administrator dapat mengatur data yang seharusnya tidak dilihat oleh semua <i>user</i> .	berdasarkan 2 kategori yaitu <i>sensitive</i> dan <i>nonsensitive</i> . Jadi saran penulis untuk keamanan lebih baik dilakukan secara keseluruhan jangan hanya jika data itu dirasa tidak <i>sensitive</i> jadi tidak diamankan dengan enkripsi.	kategori yaitu <i>sensitive</i> dan <i>non sensitive</i> , sedangkan penelitian penulis enkripsi digunakan pada seluruh data karena data tersebut merupakan data dari masyarakat yang akan mengajukan surat-surat.
3	APLIKASI DAN DEKRIPSI DATA MENGGUNAKAN	Liana, Sutardi, Nur Fajriah Muchlis 39-48ISSN : 2502-8928 (Online) semanTIK, Vol.4, No.1, Jan-Jun 2018, pp. 39-48	Membuat sistem yang dapat	1. Aplikasi enkripsi dan dekripsi data berhasil dibangun	Adapun saran yang dapat diberikan untuk pengembangan dan	Sistem ini berupa aplikasi berbasis java

	<p><i>TINY</i> <i>ENCRYPTION</i> <i>ALGORITHM</i> <i>(TEA)</i> BERBASIS JAVA</p>		<p>mengamankan data menggunakan TEA. Aplikasi ini dapat melakukan enkripsi dan dekripsi data untuk mengamankan dan membaca data yang telah di enkripsi sebelumnya.</p>	<p>untuk mengamankan berbagai jenis data bertipe txt dan docs.</p> <p>2. Algoritma <i>Tiny Encryption</i> berhasil diimplementasikan dengan melalui beberapa tahap pengujian dalam sistem enkripsi dan dekripsi data dengan menggunakan kunci sepanjang 16 karakter atau lebih. Jika kurang dari 16 karakter algoritma <i>Tiny Encryption</i></p>	<p>perbaikan aplikasi ini untuk selanjutnya adalah diharapkan dapat dilakukan pengembangan pada aplikasi enkripsi dan dekripsi ini, dengan menambahkan jenis file yang dienkripsi dan didekripsi seperti file bertipe pdf, jpeg, xlsx, video, dan audio.</p>	<p>dimana kegunaan aplikasi ini adalah untuk meng enkripsikan data dan dekripsikan data dengan format doc dan txt. Sedangkan sistem yang penulis bangun adalah sebuah sistem pelayanan masyarakat yang keamanannya dicakup keseluruhannya</p>
--	--	--	--	---	--	---

				<p>tidak dapat mengamankan file.</p> <p>3. Ukuran file sangat mempengaruhi lamanya proses enkripsi dekripsi file. Semakin besar ukuran file semakin lama proses yang diperlukan sistem untuk melakukan enkripsi file..</p>		menggunakan TDE
4	IMPLEMENTASI ENKRIPSI DATABASE MENGGUNAKAN <i>TRANSPARENT DATA ENCRYPTION</i> PADA	Antonius Wahyu Hidayat STMIK MDP Palembang Volume 2 nomor 3, oktober 2016	<i>Transparent data encryption</i> menyediakan kriptografi yang transparan pada <i>user</i> yang sah tapi tidak	<p>1. Kecilnya penalti performance ketika melakukan enkripsi dan dekripsi karena pada saat melakukan proses-proses ini, <i>Transparent Data Encryption</i> tidak mengubah tipe data dan panjang data.</p>	<p>1.Kelemahan <i>Transparent Data Encryption</i> adalah TDE <i>Masterkey</i> disimpan dalam keadaan tidak dienkripsi pada SGA (memori dari <i>database</i>.</p> <p>2.Transparent Data</p>	<p>Pada jurnal ini ditekankan kearah enkripsi dan dekripsi pada sistem dengan buka tutup wallet menggunakan</p>

	<p><i>DATABASE</i> <i>ENGINE</i> <i>ORACLE</i></p>		<p>untuk penyusup dari luar. Pada jurnal ini di ceritakan bagaimana sistem dibuat dengan keamanan menggunakan <i>transparent data encryption</i> pada mesin <i>database oracle</i></p>	<p>2.Transparent Data Encryption menawarkan kemudahan dari sisi pengguna, sehingga pengguna bisa langsung melakukan proses enkripsi data tanpa melakukan coding dan kompleksitas manajemen key.</p> <p>3.Dengan menggunakan Transparent Data Encryption, aplikasi bisa melanjutkan pekerjaan tanpa adanya database trigger, view dan perubahan aplikasi lainnya yang diasosiasikan dengan solusi enkripsi database tradisional.</p>	<p><i>Encryption</i> tidak dapat <i>mobile</i> atau sangat sulit untuk dapat diimplementasikan secara sempurna.</p>	<p><i>database engine oracle</i>.</p> <p>Sedangkan pada penelitian penulis hanya menggunakan <i>transparen data encryption</i> untuk keseluruhan data pada <i>database</i> tanpa memilih data <i>sensitive</i> dan <i>non sensitive</i>.</p>
--	--	--	--	---	---	--

5	<i>Transparent Data Encryption – Solution For Security of Database Contents</i>	(Dr. Anwar Pasha Abdul Gafoor Deskmukh, Dr. Rizayuddin Qureshi, 2014), College of Computers And Information Technologi, Universitas of Tabuk, Vol2, No 3 Desember 2014	Penelitian ini bertujuan untuk menghadapi acaman untu basis data yang rentan. TDE akan mengubah dari <i>Plaintext</i> Ke bentuk <i>Chipertext</i> dengan tujuan untuk melakukan enkripsi pada data yang diinginkan.	<i>Transparent Data Encryption</i> memiliki peran yang sangat penting dalam menjaga data saat transit. Microsoft SQL Server 2008 dengan menggunakan teknologi TDE dapat digunakan untuk melindungi data sensitif pada disk drive dan media cadangan dari akses yang tidak sah, membantu mengurangi dampak media yang hilang atau dicuri.	<ol style="list-style-type: none"> 1. TDE tidak menyediakan enkripsi diseluruh saluran komunikasi. 2. Saat mengaktifkan TDE harus mencadangkan sertifikat dan kunci pribadi yang terkait dengan sertifikat. Jika tidak maka data akan terkunci jika lupa master 	Pada jurnal ini disebutkan bahwa TDE akan digunakan pada <i>authentication, Validation</i> dan <i>data protection</i> . Perbandingan dengan sistem yang akan di bangun yaitu ada perlindungan pada database menggunakan TDE yang akan diterapkan
---	---	--	---	--	---	--

					key dan key nya.	pada data yang bersifat sensitif
6	<i>Research on the Application of Data Encryption Technology Based on Network Security Maintenance in Computer Network Security</i>	(Na Zhang, 2021) Shandong Vocational College of Light Industry, ZiBO, China, 255300	Untuk memberikan pengetahuan kepada pengguna jaringan agar bisa menguatkan keamanan jaringan pada komputer. Dan membuat teknologi enkripsi menjadi teknologi keamanan maksimum	Penerapan teknologi enkripsi keamanan data dapat menjamin keamanan informasi jaringan komputer, mempromosikan operasi normal sistem komputer, sehingga dapat lebih memenuhi kebutuhan pekerjaan dan kehidupan pengguna komputer. Oleh karena itu, kita harus memperkuat penelitian dan pengembangan teknologi enkripsi data, mempromosikan inovasi berkelanjutan dari teknologi enkripsi data,	Pada jurnal ini hanya membahas mengenai kemanan yang dapat diandalkan pada saat suatu jaringan sistem diandalkan tanpa menjelaskan dengan rinci keamanan seperti apa yang diterapkan pada kemanan	Pada jurnal ini membahas mengenai keamanan jaringan tanpa menggunakan suatu metode yang jelas berbeda dengan penelitian yang dilakukan, yaitu menggunakan TDE untuk keamanan <i>database</i> .

			pada pemeliharaan jaringan komputer	dan memberikan landasan teori ilmiah untuk pemeliharaan dan keselamatan keamanan jaringan komputer.	tersebut. Saran agar lebih diuraikan kembali keamanan seperti apa dan dimana keamanan tersebut diterapkan.	
--	--	--	-------------------------------------	---	--	--

BAB 3

ANALISIS DAN PERANCANGAN

3.1 Metode Penelitian

Metode penelitian yang dilakukan dalam penelitian ini adalah penelitian kualitatif. Penelitian kualitatif adalah sebuah proses untuk menemukan pengetahuan yang berlandaskan pada filsafat postpositivisme digunakan atau interpretif, digunakan untuk meneliti pada objek yang alamiah, dimana peneliti adalah sebagai instrument kunci, Teknik pengumpulan data dilakukan secara triangulasi, data yang diperoleh cenderung data kualitatif, analisis data bersifat induktif/kualitatif dan hasil penelitian kualitatif bersifat untuk memahami makna, memahami keunikan, mengkonstruksi fenomena, dan menemukan hipotesis. Untuk pendekatan penelitian ini menggunakan metode deskriptif komulatif. Metode deskriptif adalah penelitian yang melukiskan, menggambarkan, atau memaparkan keadaan objek yang diteliti sebagai apa adanya, sesuai dengan situasi dan kondisi Ketika penelitian tersebut dilakukan. penelitian ini juga menggunakan pendekatan kualitatif. pendekatan kualitatif adalah mekanisme kerja penelitian yang berpedoman penilaian subjektif nonstatistik atau nonmatematis, dimana ukuran nilai yang digunakan dalam penelitian ini bukanlah angka-angka skor, melainkan kategorisasi nilai atau kualitasnya.

Alasan menggunakan metode tersebut karena metode kualitatif bertujuan untuk menggambarkan keadaan atau kondisi sebenarnya yang ada dilapangan terutama dalam kaitannya dengan tema penelitian yang diambil.

3.2. Analisis

Analisis sistem didefinisikan sebagai memahami dan menspesifikasi dengan detail apa yang harus dilakukan oleh sistem. Sedangkan sistem desain diartikan sebagai menjelaskan dengan detail bagaimana bagian-bagian dari sistem informasi diimplementasikan. Sehingga Analisa dan desain sistem informasi (ANSI) bisa didefinisikan sebagai: sebagai organisasional kompleks dimana sistem informasi berbasis computer diimplementasikan. Sehingga bisa dirangkai sebagai berikut:

Tahapan-tahapan penting.

- a. Kesuksesan suatu sistem informasi tergantung pada Analisa dan perancangan yang baik

- b. ANSI telah digunakan secara luas diberbagai industry (teknologi yang telah teruji)
- c. Bagaimana membangun sistem informasi
- d. Bagaimana menganalisa kebutuhan dari sistem informasi
- e. Bagaimana merancang sistem informasi berbasis web
- f. Bagaimana memecahkan permasalahan dalam organisasi melalui sistem informasi

Berdasarkan data yang telah penulis dapatkan dari hasil studi pendahuluan dan wawancara, maka di buat table kebutuhan fungsional dan non fungsional untuk sistem yang dirancang.

3.2.1. Analisis Kebutuhan (*Requirements Definition*)

Yang dilakukan pada tahap ini adalah menganalisis kebutuhan sistem. Analisis kebutuhan dapat berupa pengumpulan data atau informasi, gambaran yang diperlukan untuk pembuatan sistem. Informasi-informasi yang dibutuhkan adalah informasi data surat-surat yang dibutuhkan warga serta data apa saja yang dibutuhkan untuk memproses pembuatan suatu surat pengantar yang dibutuhkan warga. Peneliti akan mengambil data surat khusus apa yang dibutuhkan warga sebagai syarat untuk pembuatan surat kepentingan warga.

Metode yang digunakan untuk analisis kebutuhan adalah sebagai berikut:

a. Metode Pengumpulan Data

Metode ini dilakukan untuk mendapatkan data yang diperlukan dalam memperoleh tujuan penelitian. Pengumpulan data pada penelitian ini dilakukan dengan cara sebagai berikut:

1. Melakukan tanya jawab dengan pihak Kantor Desa Mandalahaji mengenai surat-surat yang sering diajukan warga.
2. Melakukan studi literatur, jurnal, media maupun internet mengenai surat-surat pengantar yang diperlukan warga yang dibuat melalui desa.
3. Macam data yang berfungsi dalam penelitian data primer dan data sekunder. Data primer adalah data yang didapatkan langsung dari hasil tanya jawab. Kemudian data sekunder yang diperoleh dari studi literatur.

b. Metode Analisis Data

Tahap analisis data diawali dengan meneliti informasi dengan keseluruhan yang telah didapat dari fase pengumpulan data, kemudian mengidentifikasi informasi bagaimana tahapan dalam mengamankan data yang ada pada database menggunakan metode *Transparent Data Encryption (TDE)*. Metode TDE pada sistem keamanan database pada Sistem Informasi Desa Mandalahaji.

1. Analisis kebutuhan fungsional

Tabel 3.1. Kebutuhan Fungsional

Kebutuhan Pengguna	Fungsional Requirements & Prioritas			
	Hight Priority	Medium Priority	Low Priority	No Priority
<i>User</i> dapat melakukan pengajuan surat melalui sistem pelayanan surat online	Mengajukan surat melalui sistem untuk selanjutnya dilakukan proses acc	Menerapkan teknologi TDE(<i>Transparent Data Encryption</i>) pada data sensitif	-	-
<i>User</i> dapat melihat profil desa mandalahaji melalui halaman tentang desa	Melihat Profil Desa Mandalahaji			
<i>User</i> dapat melihat berita yang berkaitan dengan desa mandalahaji pada halaman berita.	Melihat Berita			

User dapat menghubungi admin melalui sistem untuk memberikan saran atau kritik melalui kontak			<i>user</i> menghubungi admin untuk penjelesan lebih lanjut saran atau kritik	
Sistem dapat melakukan registrasi untuk <i>user</i>	Sistem dapat melakukan registrasi <i>user</i> agar <i>user</i> dapat melakukan pengajuan surat			
Sistem dapat melakukan <i>login</i> untuk kebutuhan admin desa	Sistem dapat melakukan <i>login</i> admin Desa			
Sistem dapat melakukan setup biodata desa		Operator desa dapat melakukan setup biodata desa		
Sistem dapat melakukan <i>input,update</i> dan menghapus data profil desa		Operator dapat memperbaharui data informasi profil desa dengan <i>Input,Update</i> dan <i>Delete</i>		

Sistem dapat melihat daftar masyarakat yang mengajukan surat-surat.			Operator dapat melihat data masyarakat yang mengisi buku tamu pada sistem	
Sistem dapat memberikan informasi kepada masyarakat desa	Sistem dapat menampilkan informasi yang terdapat di desa mandalahaji			
Sistem dapat melihat daftar buku tamu yang diinput oleh masyarakat desa dan memberikan persetujuan untuk publish terhadap informasi tersebut				Sistem melakukan persetujuan terlebih dahulu sebelum melakukan publish buku tamu ke pihak perseorangan
Sistem dapat memberikan kesempatan kepada masyarakat desa untuk melakukan			Masyarakat desa dapat mengisi buku tamu pada sistem, sebagai tanda pengujung	

input buku tamu yang bertujuan untuk memberikan saran dan masukan			sistem maupun kritik dan saran	
Sistem dapat memberikan akses penuh kepada pengelola <i>website</i> (<i>Admin</i>)	Akses penuh <i>website</i> diberikan kepada <i>Admin</i>			
Sistem dapat memberikan akses kepada kepala desa dalam melakukan login sistem, pemantauan instalasi badan <i>website</i> desa (<i>hosting</i>), serta dapat melihat perkembangan informasi desa	Kepala desa dapat melakukan pemantauan data mulai dari profil hingga laporan pengajuan surat			

2. Kebutuhan Non Fungsional

Tabel 3.2. Kebutuhan Non Fungsional

Quality	Requirement	How
<i>Portability</i>	Sistem informasi ini harus dapat	Sistem informasi ini dibuat kedalam bentuk

	digunakan dimana pun dan kapanpun saat dibutuhkan, Ketika masyarakat hendak mengajukan surat.	aplikasi berbasis website
<i>Security</i>	Sistem ini menggunakan keamanan dengan sistem encryption untuk inputan data sensitif	Menggunakan teknologi <i>Transparent Data Encryption</i> untuk keamanan
<i>Accuracy</i>	Aplikasi harus memiliki data yang akurat dan tepat	Sistem informasi dituntut untuk memiliki akurasi yang baik
<i>Performance</i>	Karena data yang ditampilkan bersifat dinamis maka sistem harus mampu mengirimkan data terupdate dengan cepat	Basis data yang digunakan pada sistem adalah SQL Server.

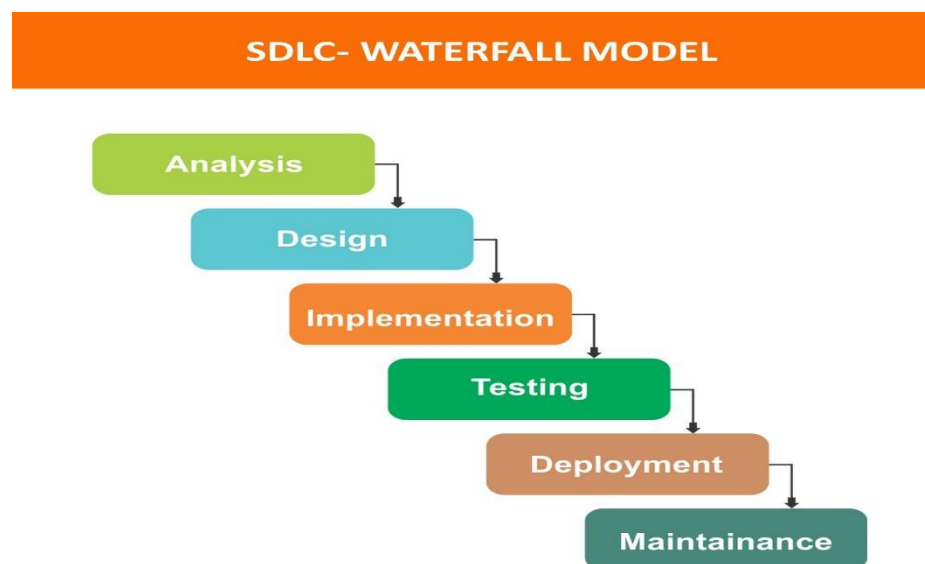
3.3. Perancangan (*Design*)

Perancangan sistem informasi merupakan pengembangan sistem baru dari sistem lama yang ada, dimana masalah-masalah yang terjadi pada sistem lama diharapkan sudah teratasi pada sistem baru . perkembangan teknologi informasi meliputi perkembangan infrastruktur seperti *hardware*, *software*, teknologi penyimpanan data (*storage*) dan teknologi komunikasi dan jaringan.

Setelah melakukan analisis kebutuhan fungsional dan non fungsional, dalam bab perancangan penulis menjabarkan gambaran dari sistem yang akan dibuat.

3.3.1 Metode Pengembangan Sistem

Metode pengembangan sistem yang digunakan dalam penelitian ini adalah menggunakan metode *waterfall*. Metode *waterfall* menerapkan tahapan dasar dari perincian, peningkatan, validasi dan evolusi sebuah sistem berdasarkan pendekatan sistematis mulai dari analisis kebutuhan, perancangan sistem, implementasi sistem, pengujian sistem dan lain sebagainya. Disebut dengan metode *waterfall* karena setiap Langkah yang dikerjakan harus menunggu selesainya tahapan sebelumnya dan berjalan secara bersambutan. Penjelasan dari metode *waterfall* dijelaskan pada gambar 3.1.



Gambar 3.1. *SDLC Waterfall Model*

Adapun penjelasan dari Gambar 3.1 tahapan-tahapan metode *waterfall* menurut Sommerville tersebut sebagai berikut.

1) *Requirement Analysis and Definition*

Requirement Analysis and Definition adalah tahapan penetapan fitur, kendala dan tujuan sistem melalui konsultasi dengan pengguna sistem. Semua hal tersebut akan ditetapkan secara rinci dan berfungsi sebagai spesifikasi sistem.

2) *System and Software Design*

Pada Tahap *System and Software Design* ini akan dibentuk suatu arsitektur sistem berdasarkan persyaratan yang telah ditetapkan. Selain itu juga,

dilakukan identifikasi dan penggambaran terhadap abstraksi dasar sistem perangkat lunak beserta hubungan-hubungannya.

3) *Implementation and Unit Testing*

Dalam tahapan *Implementation and Unit Testing* ini, hasil dari desain perangkat lunak akan direalisasikan sebagai satu set program atau unit program. Setiap unit akan diuji apakah sudah memenuhi spesifikasinya.

4) *Integration and System Testing*

Dalam tahap *Integration and System Testing* ini, setiap unit program akan diintegrasikan satu sama lain dan diuji sebagai satu sistem yang utuh untuk memastikan sistem sudah memenuhi persyaratan yang ada. Setelah itu sistem akan dikirim ke pengguna sistem.

5) *Operation and Maintenance*

Dalam tahap *Operation and Maintenance* ini, sistem diinstal dan mulai digunakan. Selain itu juga memperbaiki *error* yang tidak ditemukan pada tahap pembuatan. Dalam tahap ini juga dilakukan pengembangan sistem seperti penambahan fitur dan fungsi baru.

3.3.2 Siklus Hidup Pengembangan Sistem Informasi

Secara konseptual siklus pengembangan sebuah sistem informasi adalah sbb:

1. **Analisis Sistem:** menganalisis dan mendefinisikan masalah dan kemungkinan solusinya untuk sistem informasi dan proses organisasi.
2. **Perancangan Sistem:** merancang output, input, struktur file, program, prosedur, perangkat keras dan perangkat lunak yang diperlukan untuk mendukung sistem informasi
3. **Pembangunan dan Testing Sistem:** membangun perangkat lunak yang diperlukan untuk mendukung sistem dan melakukan testing secara akurat. Melakukan instalasi dan testing terhadap perangkat keras dan mengoperasikan perangkat lunak
4. **Implementasi Sistem:** beralih dari sistem lama ke sistem baru, melakukan pelatihan dan panduan seperlunya.
5. **Operasi dan Perawatan:** mendukung operasi sistem informasi dan melakukan perubahan atau tambahan fasilitas.

6. Evaluasi Sistem: mengevaluasi sejauh mana sistem telah dibangun dan seberapa bagus sistem telah dioperasikan.

Siklus tersebut berlangsung secara berulang-ulang. Siklus di atas merupakan model klasik dari pengembangan sistem informasi. Model-model baru, seperti prototyping, spiral, 4GT dan kombinasi dikembangkan dari model klasik di atas.

3.3.3 Analisis Sistem

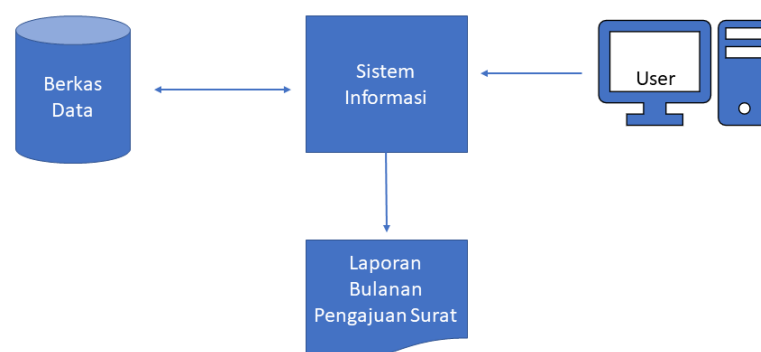
Mengawali dengan analisis sistem:

1. *Problem-solving*: sistem lama tidak berfungsi sesuai dengan kebutuhan. Untuk itu analisis diperlukan untuk memperbaiki sistem sehingga dapat berfungsi sesuai dengan kebutuhan.
2. Kebutuhan baru: adanya kebutuhan baru dalam organisasi atau lingkungan sehingga diperlukan adanya modifikasi atau tambahan sistem informasi untuk mendukung organisasi.
3. Mengimplementasikan ide atau teknologi baru.
4. Meningkatkan performansi sistem secara keseluruhan.

3.4. Gambaran Umum Sistem

Banyak aktifitas manusia yang berhubungan dengan sistem informasi, entah disadari atau tidak, sistem informasi telah banyak membantu manusia.

Perlu diketahui bahwa sistem informasi tidak harus selalu berbentuk kompleks. Gambar 3.1. memperlihatkan sebuah sistem informasi yang bersifat sederhana. Sistem tersebut dapat digunakan sebagai penyedia informasi sekaligus penghubung sistem dengan user yang akan melakukan pengajuan surat. Sistem ini hanya ada 3 actor yang terlibat yaitu user, admin dan kepala desa sebagai aktor kesatu, kedua dan ketiga.



Gambar 3.2. Gambaran Umum Sistem

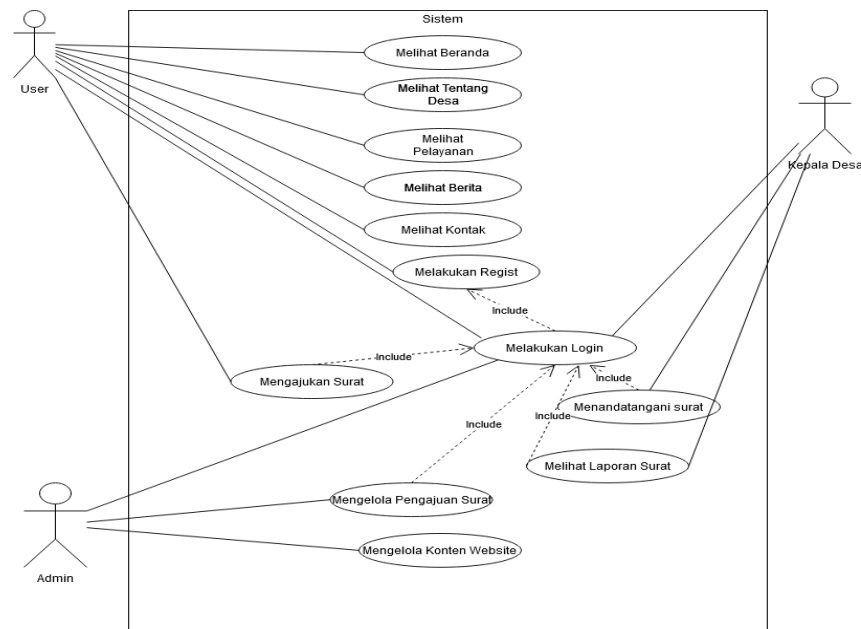
3.6. Perancangan Sistem

Perancangan sistem pendukung sistem informasi pelayanan didesa mandalahaji menggunakan metode *Transparent Data Encryption* untuk keamanan *database* dipresentasikan dalam beberapa diagram yang meliputi:

1) *Use Case Diagram*

Menurut Rosa dan Shalahudin (2014), *use case diagram* merupakan pemodelan untuk *behavior* 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. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Syarat penamaan pada *use case* adalah nama didefinisikan sesimpel mungkin dan dapat dipahami.

Pada *use case diagram* dijabarkan *actor* yang terlibat dan hal-hal yang dapat dilakukan *actor* pada sistem.



Gambar 3.3. *Use Case Diagram* Sistem Informasi Pelayanan Desa

Definisi *use case* pada *use case diagram* Sistem Informasi Desa Mandalahaji ditunjukkan pada table 3.3.

Tabel 3.3. *Definisi Use Case Diagram*

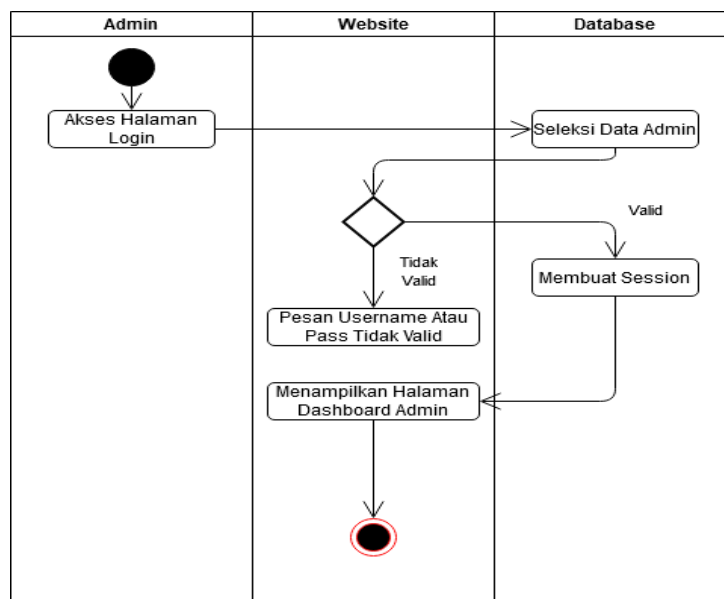
No	Use Case	Deskripsi
1	Melihat Beranda	Pada beranda akan ditampilkan halaman beranda yang berisi konten yang ada pada beranda
2	Melihat Profil Desa	Pada halaman berikut terdapat halaman yang akan menampilkan profil desa
3	Melihat Layanan	Pada halaman layanan terdapat menu dropdown menu tersebut terdapat halaman yang pembuatan surat online, jika user ingin mengajukan pembuatan surat
4	Melihat Berita	Pada halaman berita user dapat melihat berita seputar desa mandalahaji
5	Melihat Kontak	Pada halaman kontak user dapat melihat kontak mengenai desa mandalahaji mulai dari no wa, no telpon, email, dan social media lainnya
6	Melihat Menu	Pada halaman menu ini terdapat menu dropdown login
7	Setup Website Desa	Saat admin telah berhasil masuk ke dashboard admin, admin dapat melakukan perubahan konten pada website desa
8	Mengelola Data Surat	Pada halaman ini admin dapat melihat user yang mengajukan surat, dan pada halaman ini juga admin akan melakukan approvment pengajuan surat yang diajukan untuk selanjutnya diproses hingga percetakan surat

9	Melakukan Penandatanganan Surat	Kepala desa dapat melakukan tanda tangan secara online pada sistem jika kepala desa sedang tidak ada dikantor
---	---------------------------------	---

2) Activity Diagram Activity diagram yang dijabarkan meliputi *activity diagram* yang dilakukan oleh user, admin dan kepala desa

a. Activity Diagram Login Admin

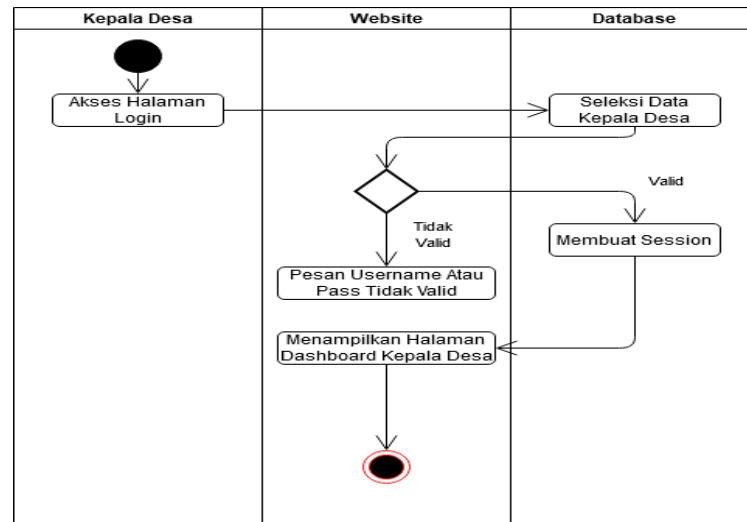
Berikut adalah gambar *activity diagram* Login Admin agar dapat mengelola *input, delete* dan *update* data website, dilakukan oleh admin, dapat dilihat pada gambar 3.4.



Gambar 3.4. Activity Diagram Login Admin

b. Activity Diagram Login Kepala Desa

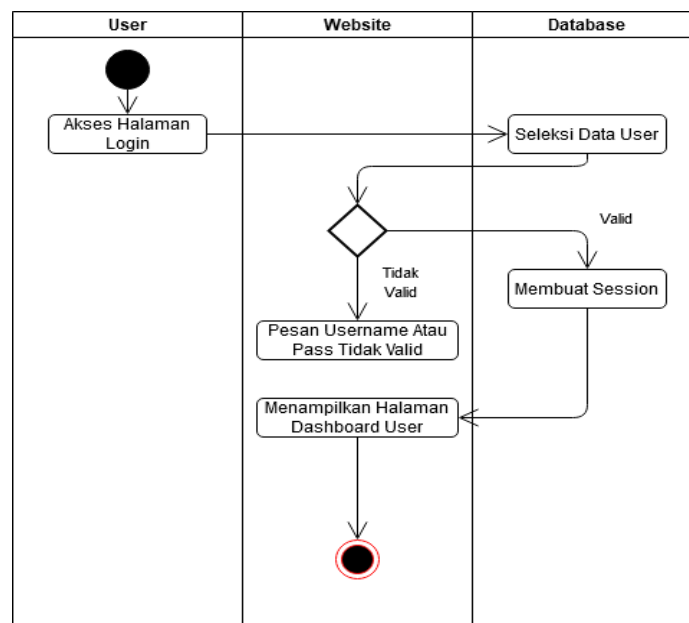
Berikut adalah gambar *activity diagram* Login Kepala Desa agar dapat menandatangani pengajuan surat dan melihat laporan pengajuan surat, dilakukan oleh kepala desa, dapat dilihat pada gambar 3.5.



Gambar 3.5. *Activity Diagram Login Kepala Desa*

c. Activity Diagram Login User

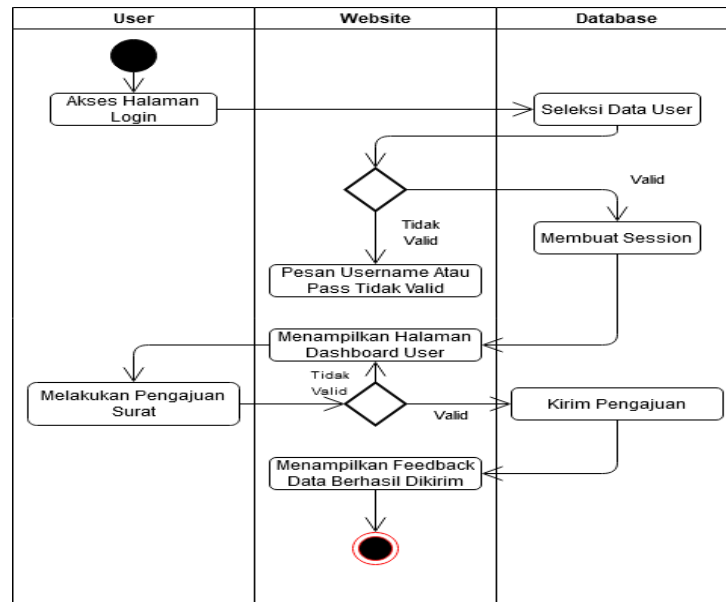
Berikut adalah gambar *activity diagram* Login User agar dapat melakukan pengajuan surat, dilakukan oleh user yang memiliki akses penuh, dapat dilihat pada gambar 3.6.



Gambar 3.6. *Diagram Activity Login User*

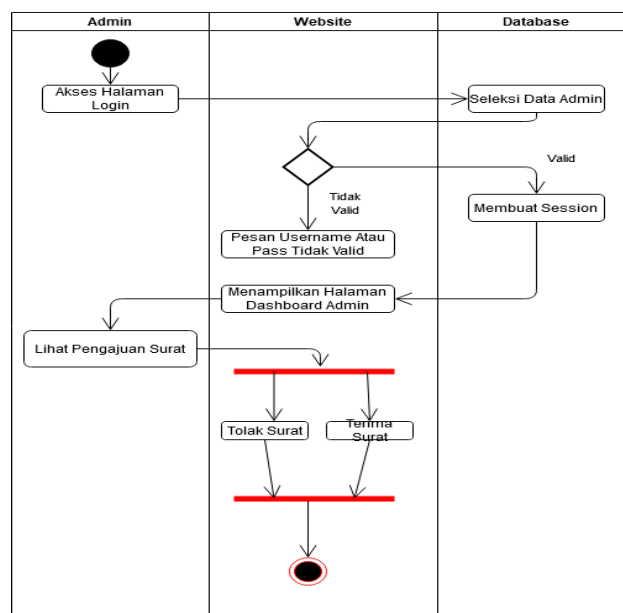
d. Activity Diagram Pengajuan Surat

Berikut adalah gambar *activity diagram* Pengajuan Surat agar user dapat melakukan pengajuan surat, dilakukan oleh user, dapat dilihat pada gambar 3.7.

Gambar 3.7. *Diagram Activity* Pengajuan Surat

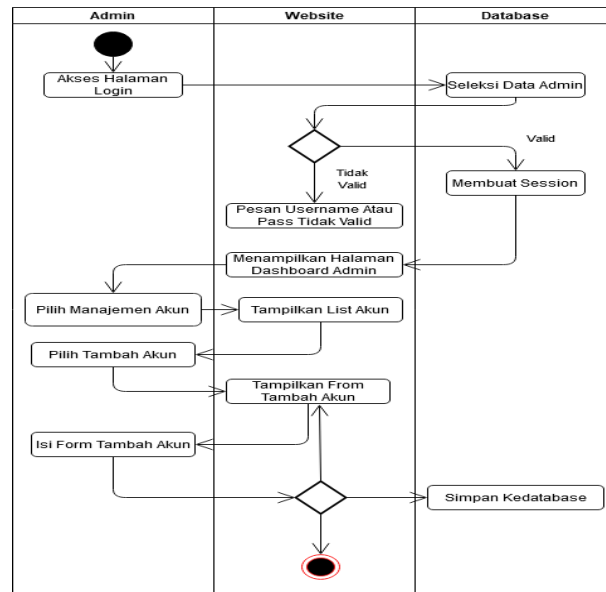
e. ***Diagram Activity* Penerimaan Pengajuan Surat**

Berikut adalah gambar *activity diagram* Penerimaan Pengajuan Surat agar user yang melakukan pengajuan dapat mengetahui pengajuan surat pengantarnya di terima atau tidak, dilakukan oleh admin, dapat dilihat pada gambar 3.8.

Gambar 3.8. *Diagram Activity* Penerimaan Pengajuan Surat

f. ***Diagram Activity* Tambah Akun**

Berikut adalah gambar *activity diagram* Tambah Akun untuk menambah akun user atau operator desa yang ingin mengelola data pada website, dilakukan oleh admin, dapat dilihat pada gambar 3.9.



Gambar 3.9. *Diagram Activity* Tambah Akun

g. *Activity Diagram* Kelola Akun

Berikut adalah gambar *activity diagram* Kelola Akun untuk melakukan ubah dan hapus akun yang ada pada database dan sistem, dilakukan oleh admin dan super admin, dapat dilihat pada gambar 3.10.

Use Case ID Number	01	
Use Case Name	Mendaftar sebagai user sistem informasi desa mandalahaji	
Use Case Description	Use Case ini menggambarkan proses pengguna mendaftar sebagai user di sistem informasi desa mandalahaji	
Primary Actor	User/Pengaju Surat	
Secondary Actor	Admin/Operator Desa	
Pre-Condition	User telah mengakses web aplikasi desa mandalahaji	
Primary Flow of Events	<i>User Action</i>	<i>System Response</i>
	1. User memilih menu login pada aplikasi dihalaman pelayanan lalu pilih pembuatan surat online.	
		2. Aplikasi menampilkan form login.
	3. User memilih menu daftar pada halaman login untuk syarat pengajuan surat	
		4. Aplikasi menampilkan form pendaftaran
	5. User mengisi formulir yang telah disediakan, termasuk data diri seperti Nama dan Alamat Lengkap email dan lainnya.	

	6. User memilih tombol simpan	
	7. Operator menyetujui form request user untuk mendaftar sebagai syarat pengajuan surat	
		8. Aplikasi menyimpan data pendaftar/buku tamu baru
		9. aplikasi mengirimkan username dan pass baru ke email user
Error Flow Events	5. Pengguna bukan merupakan warga Desa Mandalahaji	
		6. Aplikasi tidak akan mendaftarkan diri pengguna dan muncul alert bahwa pendaftar harus merupakan warga desa mandalahaji
Post-Condition	Pengguna dapat melakukan pengajuan surat setelah melakukan login yang di ijinan oleh admin	

Use case scenario untuk proses mengajukan surat pada aplikasi/website desa mandalahaji dapat dilihat pada use case scenario 2 mengajukan surat.

Use Case Scenario 02 mengajukan surat di website desa mandalahaji

Tabel 3.5. Use Case Scenario Mengajukan Surat Pengantar

<i>Use Case ID Number</i>	02	
<i>Use Case Name</i>	Mengajukan surat pada website desa mandalahaji	
<i>Use Case Deskripsi</i>	Use Case ini menggambarkan proses pengguna mengajukan pembuatan surat-surat pengantar pada website desa mandalahaji	
Primary Actor	User/Pengaju Surat	
Secondary Actor	Admin/Operator Desa	
Pre-Condition	User telah mengakses web aplikasi desa mandalahaji lalu masuk ke halaman layanan lalu memilih pelayanan surat online	
Primary Flow of Events	<i>User Action</i>	<i>System Response</i>
	1. Setelah user melakukan login maka user di iijinkan untuk membuka akses halaman pembuatan surat online.	
		2. Aplikasi manampilkan form pembuatan surat online.
	3. User memasukan data diri dan memasukan no NIK	
	4. User mengklik tombol submit	

		5. Sistem memunculkan notifikasi surat berhasil dibuat, tunggu Langkah selanjutnya
	6. User memilih tombol beranda	
		7. Sistem memberika pesan melalui email atau wa untuk proses selanjutnya apakah surat akan berbentuk softfile atau hardfile jika softfile maka surat akan dikirim hari itu juga dan ditanda tangani langsung ke kepada desa
	8. user mendapatkan pesan untuk mengambil surat yang telah di tanda tangani kepala des ajika berupa hardfile	
Error Flow Events	5a. Pengguna salah memasukan NIK dan nama yang berbeda pada data yang di inputkan.	
		5b. Aplikasi tidak akan melakukan proses approve jika NIK yang diinputkan salah dan nama tidak sesuai dengan NIK yang diinputkan
Post-Condition	Pengguna dapat mengunduh surat berupa softfile yang tidak ditandatangani dan harus ditandatangani langsung ke kepala	

	desa dan berupa hardfile yang sudah di tanda tangani kepala desa
--	--

Use Case Scenario 03 Login Admin

Tabel 3.6 Use Case Scenario Login Admin

Use Case ID Number	03	
Use Case Name	Login Admin	
Use Case Deskripsi	Use Case ini menggambarkan proses login dari admin agar dapat mengelola data hapus edit update pengaju surat pengantar dan melakukan set website	
Primary Actor	Admin dan Sistem	
Secondary Actor	-	
Pre-Condition	Admin belum pernah melakukan autentikasi dan pertama kali membuka website	
Primary Flow of Events	<i>User Action</i>	<i>System Response</i>
	1. Admin membuka website Aplikasi Desa Mandalahaji	
		2. Aplikasi menampilkan halaman login dari dropdown menu
	3. Admin memilih halaman login	
		4. Aplikasi menampilkan halaman login
	5. Admin memasukan <i>username</i> dan <i>password</i>	
		6. Aplikasi menampilkan halaman dashboard admin
Error Flow Events	1.a Admin salah memasukan username	
	2.a Admin salah memasukan password	
	3.a Admin tidak terhubung ke jaringan internet	

Post-Condition	Admin dapat melakukan pengelolaan data seperti Create Read Update Delete pada data pengajuan surat pengantar, dan dapat melakukan set website setelah melakukan login pada aplikasi
-----------------------	---

Use Case Scenario Penandatanganan surat 04 Kepala Desa

Tabel 3.7. Use Case Scenario Penandatanganan Surat Kepala Desa

Use Case ID Number	04	
Use Case Name	Penandatanganan Surat oleh Kepala Desa	
Use Case Deskripsi	Use Case ini menggambarkan proses penandatanganan surat oleh kepala desa secara <i>online</i>	
Primary Actor	Kepala Desa	
Secondary Actor	User	
Pre-Condition	Kepala Desa berada di kantor dan tidak diperlukan tanda tangan secara <i>online</i>	
Primary Flow of Events	<i>User Action</i>	<i>System Response</i>
	1. Kepala Desa membuka website Aplikasi Desa Mandalahaji	
		2. Aplikasi menampilkan halaman login dari dropdown menu
	3. Kepala Desa memilih halaman login	
		4. Aplikasi menampilkan halaman login
	5. Kepala Desa memasukan <i>username</i> dan <i>password</i>	
		6. Aplikasi menampilkan halaman dashboard Kepala Desa

	7. Kepala desa memilih halaman pengajuan surat	
		8. Menampilkan data pengajuan surat
	9. Memilih surat yang akan di tandatangani	
		10. Mengirimkan Kembali surat yang telah ditandatangani kepada admin
Error Flow Events	1.a Kepala Desa salah memasukan username	
	2.a Kepala Desa salah memasukan password	
	3.a Kepala Desa tidak terhubung ke jaringan internet	
Post-Condition	Kepala Desa dapat melihat laporan mengenai pengajuan surat yang diajukan user dan dapat melakukan tanda tangan secara <i>online</i> jika sedang tidak berada di kantor	

Use Case Scenario 05 Login Kepala Desa

Tabel 3.8. Use Case Scenario Login Kepala Desa

<i>Use Case ID Number</i>	05	
<i>Use Case Name</i>	Login Kepala Desa	
<i>Use Case Deskripsi</i>	Use Case ini menggambarkan proses login dari kepala desa agar dapat melihat laporan buku tamu/pengunjung website, masyarakat desa mandalahaji yang mengajukan surat pengantar	
Primary Actor	Kepala Desa	
Secondary Actor	-	
Pre-Condition	Kepala Desa belum pernah melakukan autentikasi	
Primary Flow of Events	<i>User Action</i>	<i>System Response</i>

	1. Kepala Desa membuka website Aplikasi Desa Mandalahaji	
		2. Aplikasi menampilkan halaman login dari dropdown menu
	3. Kepala Desa memilih halaman login	
		4. Aplikasi menampilkan halaman login
	5. Kepala Desa memasukan <i>username</i> dan <i>password</i>	
		6. Aplikasi menampilkan halaman dashboard super admin
Error Flow Events	1.a Super Admin salah memasukan username	
	2.a Super Admin salah memasukan password	
	3.a Super Admin tidak terhubung ke jaringan internet	
Post-Condition	Kepala Desa dapat melihat laporan mengenai buku tamu pengunjung website dan masyarakat desa mandalahaji yang mengajukan surat pengantar setelah melakukan login	

Use Case Scenario Tambah Akun Baru

Tabel 3.9. Use Case Scenario Tambah Akun Baru

<i>Use Case ID Number</i>	06
<i>Use Case Name</i>	Mendaftar Akun User Baru
<i>Use Case Deskripsi</i>	Use Case ini menggambarkan proses pendaftaran user yang akan mengajukan pembuatan surat pengantar ke sistem / aplikasi
Primary Actor	User

Secondary Actor	Admin	
Pre-Condition	User yang akan melakukan pengajuan pembuatan surat harus memiliki akun terlebih dahulu	
Primary Flow of Events	<i>User Action</i>	<i>System Response</i>
	1. User membuka website Aplikasi Desa Mandalahaji	
		2. Aplikasi menampilkan halaman login dari dropdown menu
	3. User memilih halaman login	
		4. Aplikasi menampilkan halaman login, daftar dan beranda
	5. User memilih menu daftar	
		6. Aplikasi menampilkan form pendaftaran
	7. User memasukan data ke form pendaftaran	
		8. Sistem memvalidasi untuk pembuatan username dan password yang akan dikirimkan melalui wa atau email
Error Flow Events	1.a Data yang dimasukan user tidak valid	
	2.a Warga yang mendaftar bukan masyarakat desa mandalahaji	
	3.a Pendaftaran tidak disetujui admin karena data yang dimasukan kurang valid	

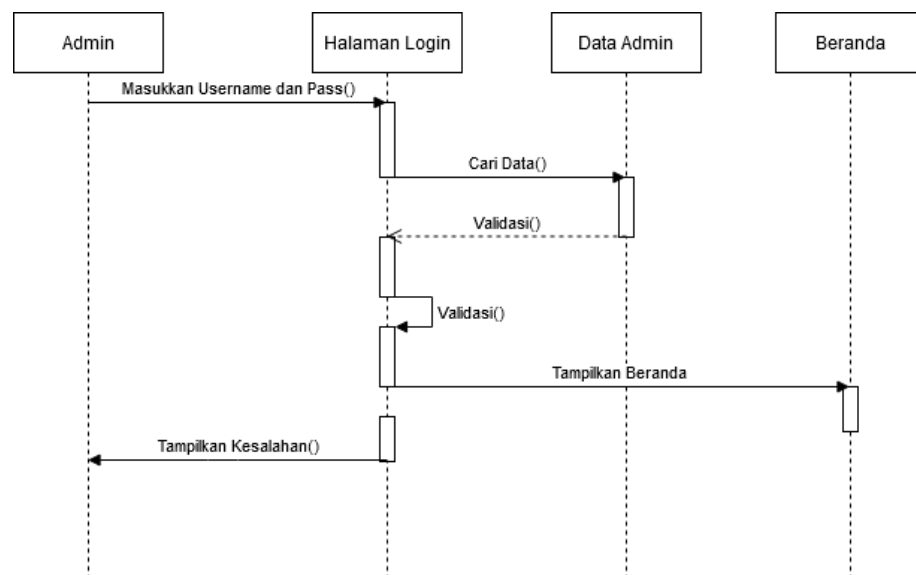
Post-Condition	User dapat melakukan pengajuan surat setelah mengisi buku tamu dengan melakukan login, jika tidak punya akun maka daftar terlebih dahulu lalu admin akan mengirimkan username dan password untuk masuk ke halaman pengisian form pengajuan surat pengantar
-----------------------	--

4) *Sequence Diagram*

Pada bagian ini penulis akan menggambarkan perilaku sistem dengan menggunakan *sequence diagram*. Secara umum *sequence diagram* menunjukkan urutan waktu aliran pesan dari suatu obyek ke obyek yang lainnya.

a. *Sequence Diagram Login Admin*

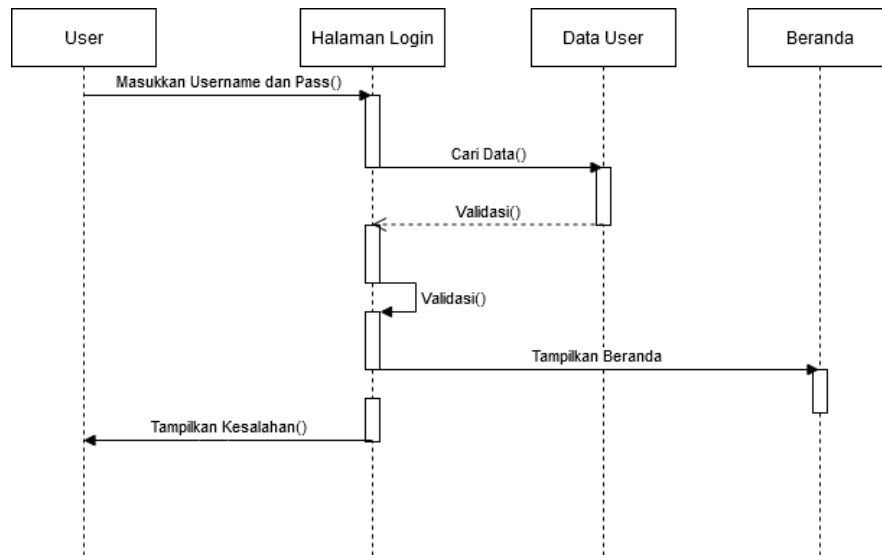
Berikut merupakan *sequence diagram login admin* pada gambar 3.11.



Gambar 3.11. *Sequence Diagram Login Admin*

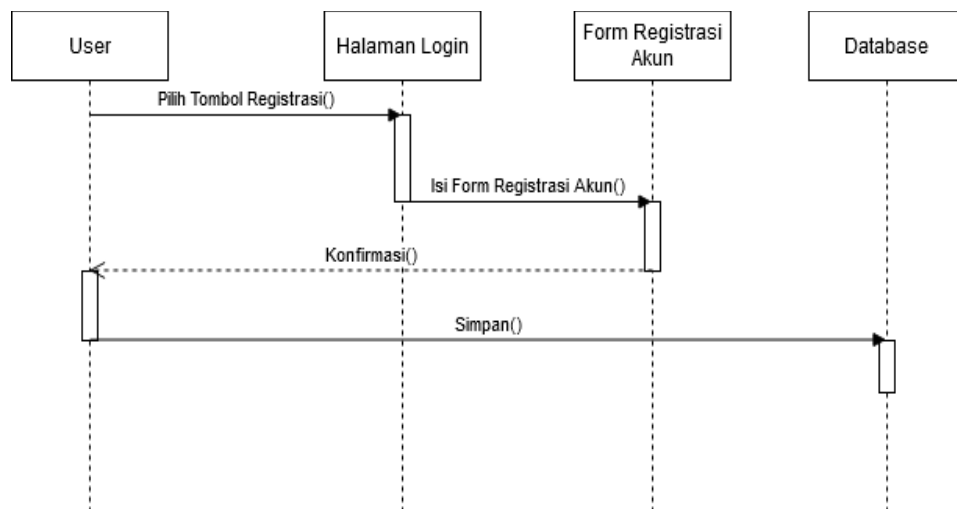
b. *Sequence Diagram Login User*

Berikut merupakan *sequence diagram login user* pada gambar 3.12.

Gambar 3.12. *Sequence Diagram Login User*

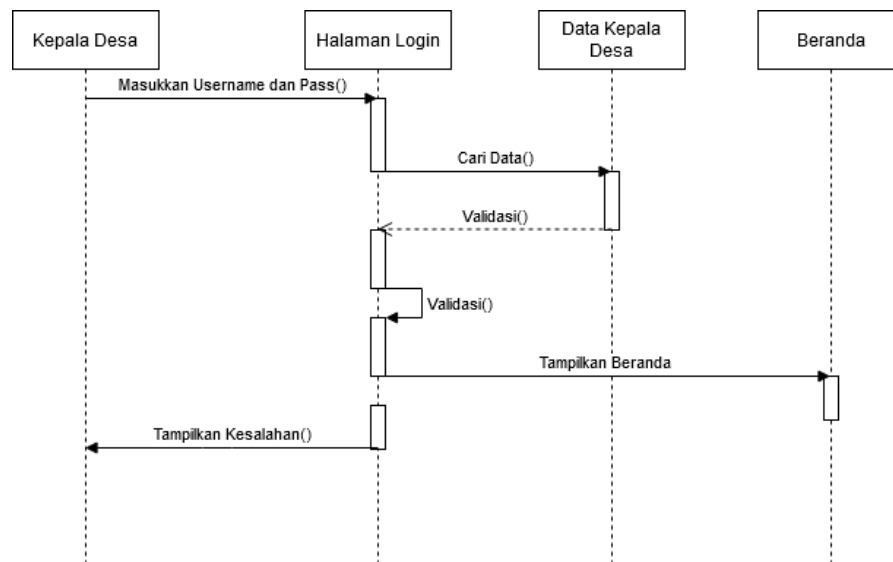
c. *Sequence Diagram Registrasi Akun User*

Berikut merupakan sequence diagram registrasi akun baru yang dilakukan user untuk mengisi buku tamu, untuk melakukan pengajuan surat pada gambar 3.13.

Gambar 3.13. *Sequence Diagram Pendaftaran Akun User*

d. *Sequence Diagram Login Kepala Desa*

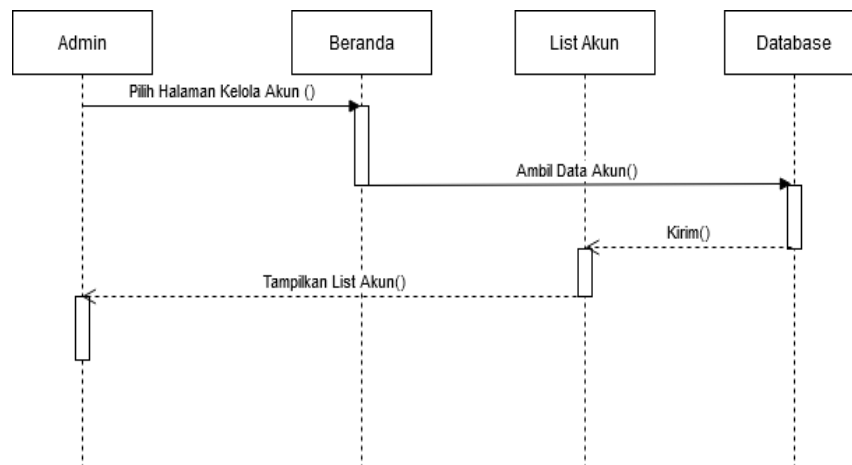
Berikut merupakan *sequence diagram login* kepala desa pada gambar 3.14.



Gambar 3.14. *Sequence Diagram Login Kepala Desa*

e. ***Sequence Diagram Tampilkan List Akun***

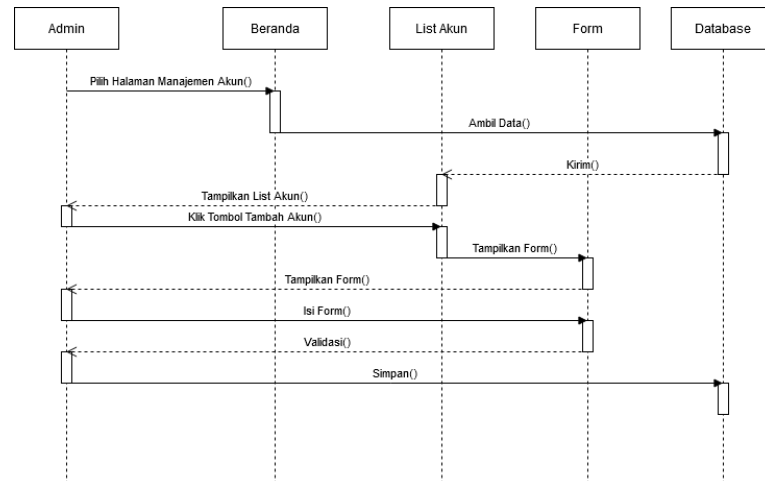
Berikut merupakan *sequence diagram* untuk menampilkan *list akun* yang telah terdaftar yang dilakukan oleh admin pada gambar 3.15.



Gambar 3.15. *Sequence Diagram Tampilkan List Akun*

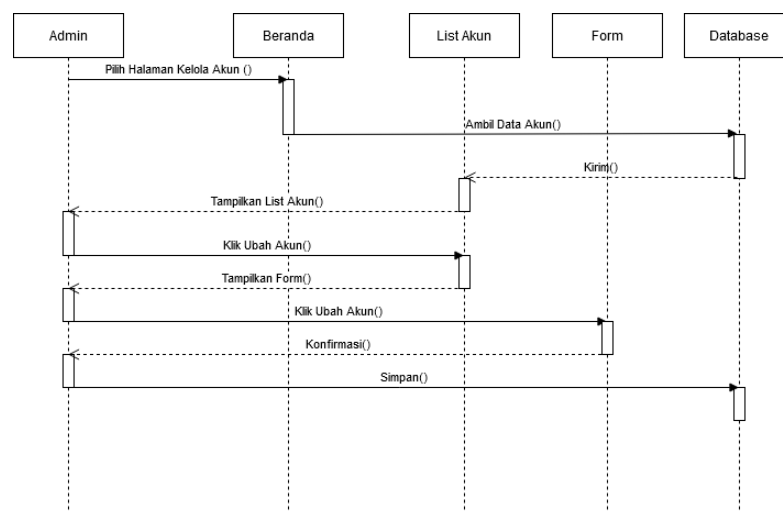
f. ***Sequence Diagram Tambah Akun***

Berikut merupakan *sequence diagram* untuk menambah akun yang akan didaftarkan dan dilakukan oleh admin pada gambar 3.16.

Gambar 3.16. *Sequence Diagram* Tambah Akun

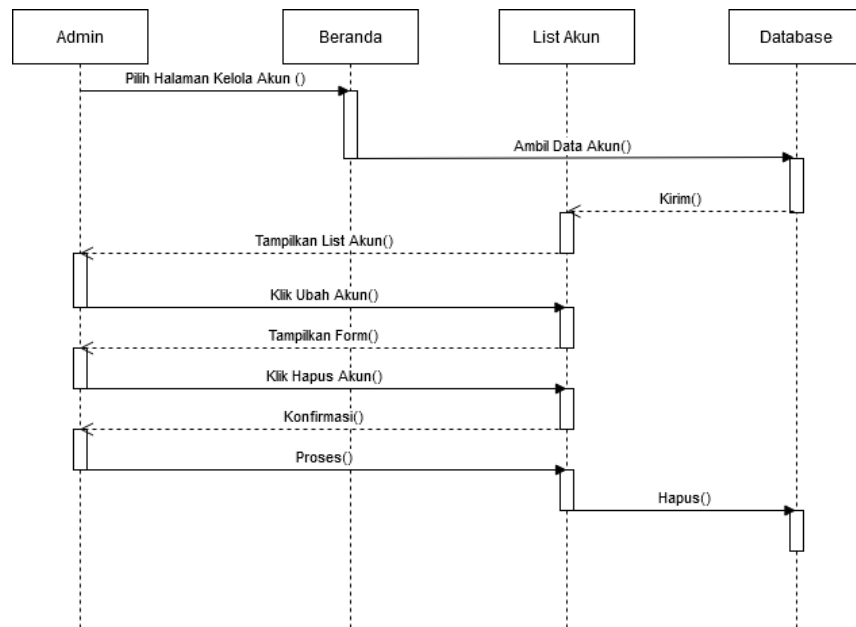
g. *Sequence Diagram* Ubah Akun

Berikut merupakan *sequence diagram* untuk ubah akun yang akan didaftarkan dan dilakukan oleh admin pada gambar 3.17.

Gambar 3.17. *Sequence Diagram* Ubah Akun

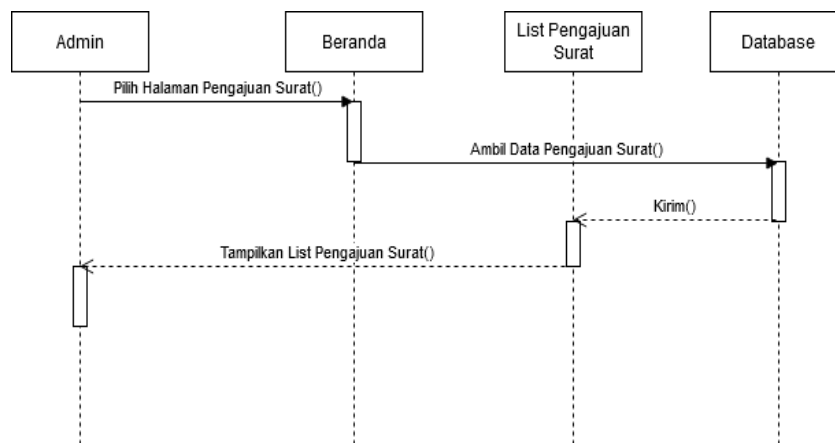
h. *Sequence Diagram* Hapus Akun

Berikut merupakan *sequence diagram* untuk hapus akun yang telah terdaftar dan dilakukan oleh admin pada gambar 3.18.

Gambar 3.18. *Sequence Diagram* Hapus Akun

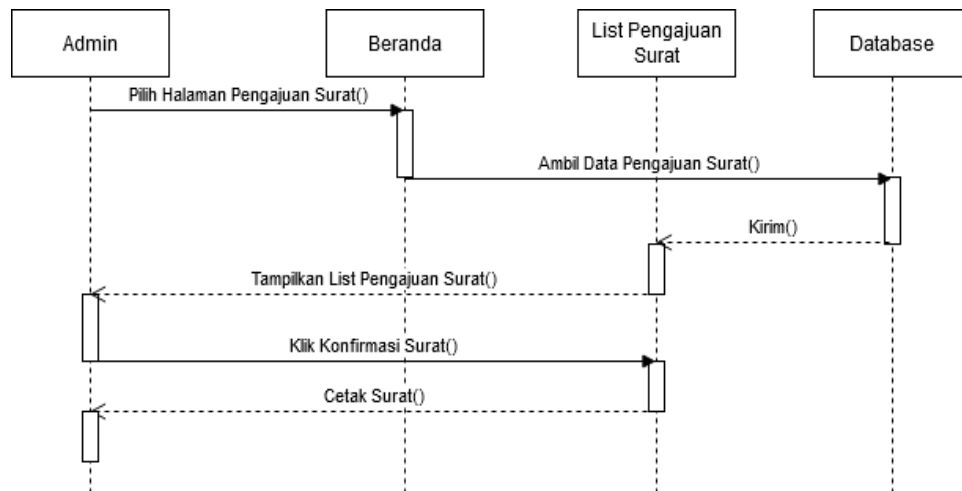
i. ***Sequence Diagram* Lihat Pengajuan Surat**

Berikut merupakan *sequence diagram* untuk lihat pengajuan surat yang telah diajukan oleh user dan lihat pengajuan ini dilakukan oleh admin pada gambar 3.19.

Gambar 3.19. *Sequence Diagram* Lihat Pengajuan Surat

j. ***Sequence Diagram* Persetujuan Pengajuan Surat**

Berikut merupakan *sequence diagram* untuk persetujuan pengajuan surat yang telah diajukan oleh *user* dan persetujuan pengajuan surat ini dilakukan oleh admin pada gambar 3.20.

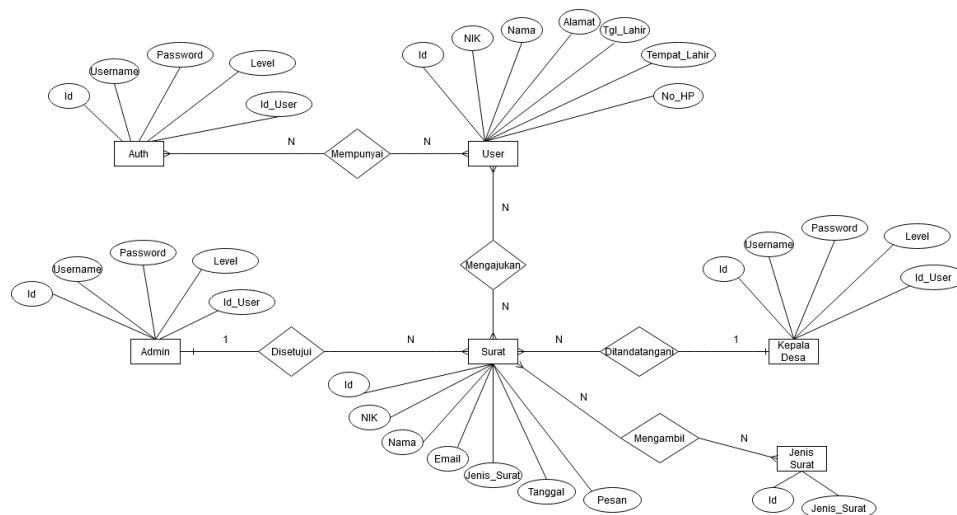


Gambar 3.20. *Sequence Diagram* Persetujuan Pengajuan Surat

3.7. Perancangan Basis Data

3.7.1 *Entity Relationship Diagram (ERD)*

ER Diagram menggambarkan *structure database* yang akan dirancang dan relasi antar tabel. Berikut ER Pada diagram



Gambar 3.21. *Entity Relationship Diagram*

User memiliki atribut:

- id (integer) sebagai Primary Key
- NIK (integer)
- Nama (varchar)
- Alamat (text)
- Tgl_Lahir (date)

- f) Tempat_Lahir (varchar)
- g) No_HP (varchar)

Admin memiliki atribut:

- a) Id (integer) sebagai Primary Key
- b) Username (varchar)
- c) Password (varchar)
- d) Level (varchar)
- e) Id_user (foreignt Key)

Kepala Desa memiliki atribut:

- a) Id (integer) sebagai Primary Key
- b) Username (varchar)
- c) Password (varchar)
- d) Level (varchar)
- e) Id_user (foreignt Key)

Surat memiliki atribut:

- a) Id (integer) sebagai primary key
- b) NIK (varchar)
- c) Nama (varchar)
- d) Email (varchar)
- e) Jenis_surat (varchar)
- f) Tanggal (date)
- g) Pesan (varchar)

Jenis_surat memiliki atribut:

- a) Id (integer) sebagai primary key
- b) Jenis_surat (varchar)

Menentukan kardinalitas relasi dan *foreignt key*:

- a) Auth mempunyai User

Tabel utama: Auth

Tabel kedua: User

Relasi: Many to Many (N-N)

Atribut penghubung: id_user (Foreign Key id_user)

- b) User mengajukan surat

Tabel utama: User

Tabel kedua: Surat

Relasi: Many to Many (N-N)

Atribut penghubung: Jenis_Surat (Foreign Key jenis_surat)

c) Admin Menyetujui Surat

Tabel utama: Admin

Tabel kedua: Surat

Relasi: One to Many (1-N)

Atribut Penghubung: id_user, jenis_surat (Foreign Key Id_user di user dan jenis_surat di Jenis_Surat)

d) Kepala Desa Menandatangani Surat

Tabel utama: Kepala desa

Tabel kedua: Surat

Relasi: One to Many (1-N)

Atribut Penghubung: id_user, jenis_surat (Foreign Key Id_user di user dan jenis_surat di Jenis_Surat)

e) Surat mengambil Jenis_Surat

Tabel utama: Surat

Tabel Kedua: Jenis_Surat

Relasi: Many to Many (N-N)

Atribut Penghubung: jenis_surat (Foreign Key jenis_surat)

Struktur Data

Nama Tabel : Tb_Auth

Primay Key : Id

Foreignt Key : NIP

Tabel 3.10. Tabel *Auth*

No	Nama Field	Tipe Data	Panjang
1	Id	Int	10
2	Username	Varchar	30
3	Password	Varchar	10
4	Level	Varchar	20
5	Id_User	Int	11

Nama Tabel : Tb_User

Primay Key : Id

Foreignt Key : NIK

Tabel 3.11. Tabel *User*

No	Nama Field	Tipe Data	Panjang
1	Id	Int	10
2	NIK	Varchar	30
3	Nama	Varchar	50
4	Alamat	Text	
5	Tgl_Lahir	Date	
6	Tempat_Lahir	Varchar	30
7	No_Hp	Varchar	20

Nama Tabel : Tb_Jenis_Surat

Primay Key : Id

Foreignt Key : NIP

Tabel 3.12. Tabel Jenis Surat

No	Nama Field	Tipe Data	Panjang
1	Id	Int	10
2	Jenis_Surat	Varchar	200

Nama Tabel : Tb_Surat

Primay Key : Id

Foreignt Key : Username

Tabel 3.13. Tabel Surat

No	Nama Field	Tipe Data	Panjang
1	Id	Int	10
2	NIK	Varchar	30
3	Nama	Varchar	15
4	Email	Varchar	50
5	Wa	Varchar	100
6	Jenis Surat	Varchar	20
7	Tanggal	Varchar	12
8	Pesan	Varchar	20

Nama Tabel : Tb_ProfilDesa

Primay Key : Id

Foreignt Key :-

Tabel 3.14. Tabel Profil Desa

No	Nama Field	Tipe Data	Panjang
1	Id	Int	10
2	Judul	Varchar	30
3	Sub_Judul	Varchar	10
4	Isi	Varchar	50
5	Foto	Varchar	100
6	Ukuran	Int	10
7	Tipe	Varchar	12

Nama Tabel : Tb_Galeri

Primay Key : Id

Foreignt Key :-

Tabel 3.15. Tabel Galeri

No	Nama Field	Tipe Data	Panjang
1	Id	Int	10
2	Foto	Varchar	30
3	Ukuran	Int	10
4	Tipe	Varchar	10
5	Ket	Varchar	100

Nama Tabel : Tb_Berita

Primay Key : Id

Foreignt Key :-

Tabel 3.16. Tabel Berita

No	Nama Field	Tipe Data	Panjang
1	Id	Int	10
2	Judul	Varchar	30
3	Sub_Judul	Int	10
4	Isi	Varchar	10
5	Foto	Varchar	100
6	Ukuran	Int	20
7	Tipe	Varchar	10
8	Tgl	Date	

Nama Tabel : Tb_Kontak

Primay Key : Id

Foreignt Key : No_Wa

Tabel 3.17. Tabel Kontak

No	Nama Field	Tipe Data	Panjang
1	Id	Int	10
2	No_Wa	Varchar	15
3	Alamat	Varchar	100
4	Email	Varchar	40
5	Fb	Varchar	40
6	Ig	Varchar	40
7	Twitter	Varchar	40

Nama Tabel : Tb_PengajuanSurat

Primay Key : Id

Foreignt Key : Kode_Surat

Tabel 3.18. Tabel Pengajuan Surat

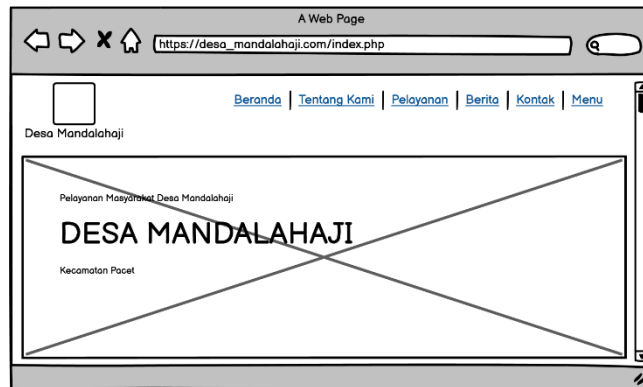
No	Nama Field	Tipe Data	Panjang
1	Id	Int	10
2	Kode_Surat	Varchar	15
3	Jenis_Surat	Varchar	100
4	NIK	Varchar	40
5	No_KK	Varchar	40
6	No_Wa	Varchar	40
7	Email	Varchar	40
8	Pesan	Varchar	200

3.8. Desain Antarmuka

Desain antarmuka pengguna merupakan tampilan yang akan berinteraksi langsung dengan pengguna. Sebelum melakukan *layouting* langsung pada aplikasi, peneliti membuat desain berupa *wireframes* dan desain akan diperlihatkan kepada calon pengguna untuk menyesuaikan kebutuhan pengguna.

3.8.1. Halaman Beranda *User*

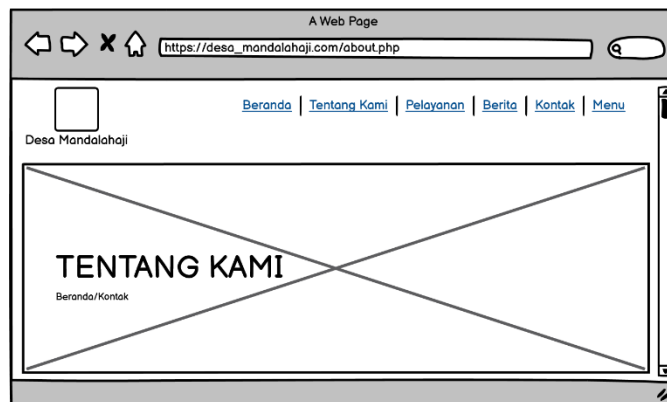
Halaman beranda adalah halaman yang pertama kali di akses ketika pengguna membutuhkan informasi dan akan melakukan pengajuan surat. Berikut tampilan Beranda pada gambar 3.22.



Gambar 3.22 Halaman Beranda

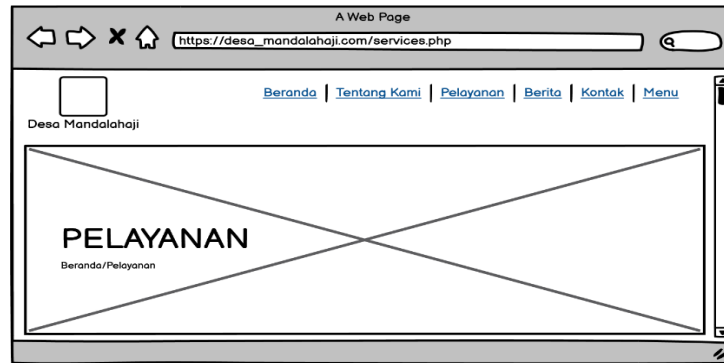
3.8.2. Halaman *About*

Halaman *about* adalah halaman yang menampilkan profil dari Desa Mandalahaji yang dapat diakses langsung oleh pengguna setelah halaman beranda. Berikut tampilan halaman *about* pada gambar 3.23.

Gambar 3.23 Halaman *About*

3.8.3. Halaman Pelayanan

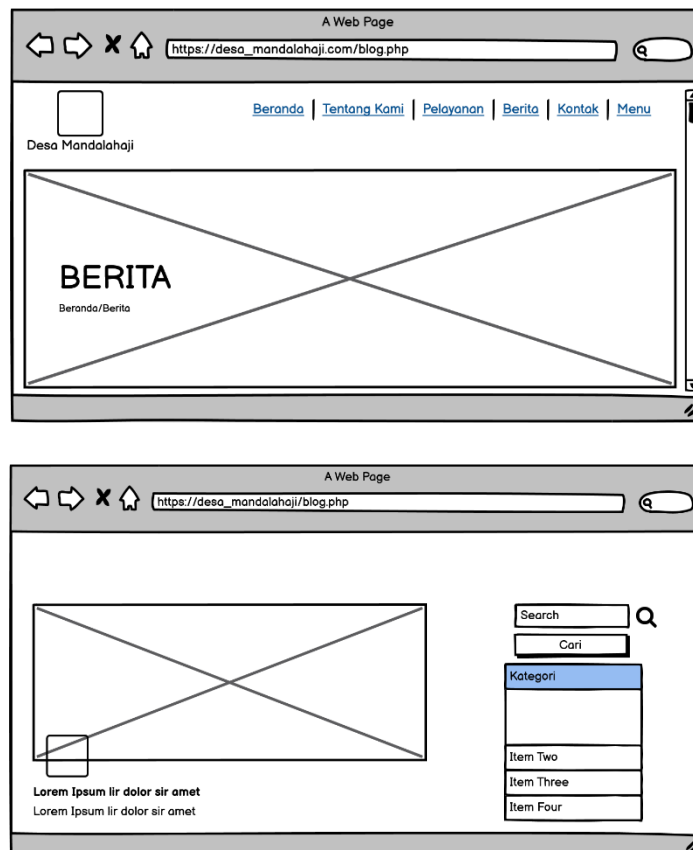
Halaman pelayanan adalah halaman yang menampilkan jenis-jenis pelayanan yang ada pada sistem informasi Desa Mandalahaji. Berikut tampilan halaman layanan pada gambar 3.24.



Gambar 3.24 Halaman Pelayanan

3.8.4. Halaman Berita

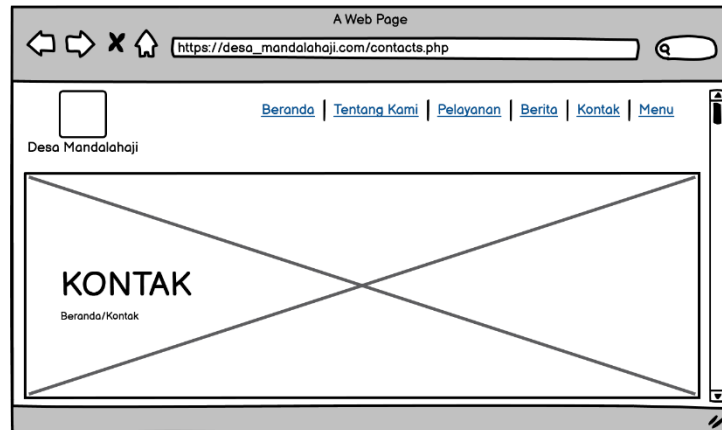
Halaman berita adalah halaman yang menampilkan berita yang ada pada sistem informasi Desa Mandalahaji. Berikut tampilan halaman berita pada gambar 3.25.



Gambar 3.25 Halaman Berita

3.8.5. Halaman Kontak

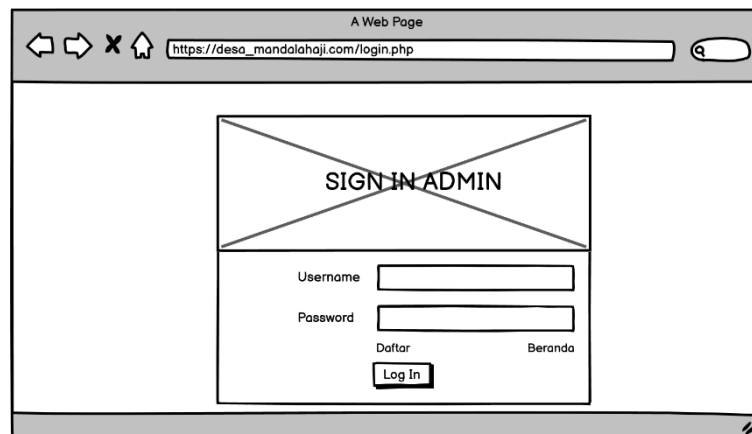
Halaman kontak adalah halaman yang menampilkan kontak yang diperlukan untuk pelayanan di Desa Mandalahaji. Berikut tampilan halaman kontak pada gambar 3.26.



Gambar 3.26 Halaman Kontak

3.8.6. Halaman *Login*

Halaman *login* adalah halaman yang digunakan untuk masuk ke halaman dengan akses tertentu seperti akses untuk pengguna, admin dan kepala desa pada sistem informasi desa mandalahaji. Berikut tampilan halaman *login* pada gambar 3.27.

Gambar 3.27 Halaman *Login*

3.8.7 Halaman Registrasi

Halaman registrasi adalah halaman yang digunakan untuk mendaftarkan diri sebagai user pada sistem informasi Desa Mandalahaji, setelah dilakukan registrasi pengguna dapat melakukan *login* dan dapat melakukan pengajuan surat online setelah masuk ke halaman dashboard *user*. Berikut tampilan halaman registrasi pada gambar 3.28.

A Web Page

https://desa_mandalahaji.com/registrasi.php

Registrasi

Nama

Alamat

Daftar Beranda

Daftar

Gambar 3.28 Halaman Registrasi

3.8.8. Halaman *Dashboard User*

Halaman ini menampilkan tampilan dari *dashboard user* dan pada halaman ini juga pengguna dapat melakukan pengajuan surat pada sistem. Berikut tampilan halaman *dashboard admin* pada gambar 3.29.

A Web Page

https://desa_mandalahaji/index_user.php

[Beranda](#) | [Kontak](#) | [Menu](#)

Lorem ipsum sir dolor sir amet
Lorem ipsum sir dolor sir amet

Data Pengguna

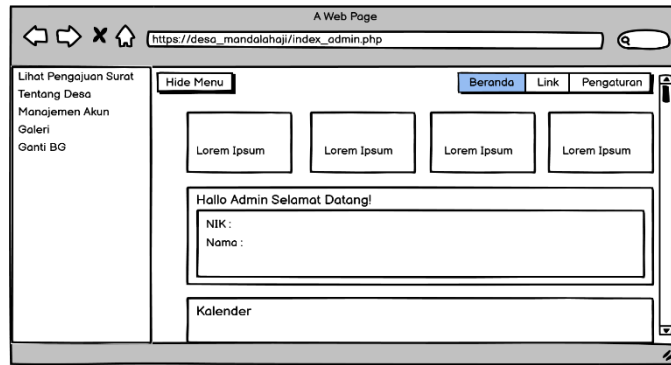
Lorem Ipsum Sir Dolor Sir Amet
Lorem Ipsum Sir Dolor Sir Amet

[a link](#) [a link](#)

Gambar 3.29 Halaman *Dashboard User*

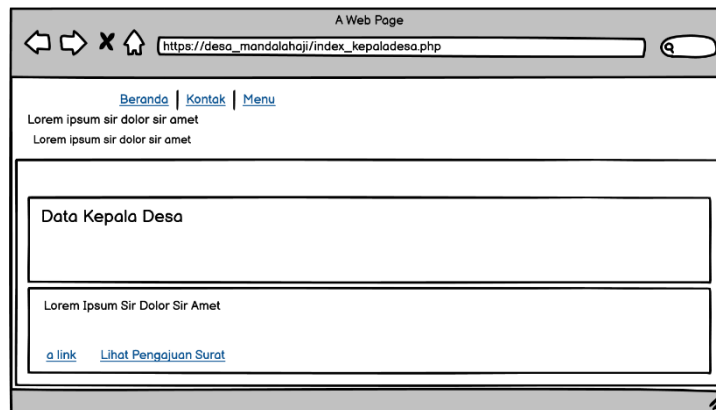
3.8.9. Halaman *Dashboard Admin*

Halaman ini menampilkan tampilan dari *dashboard admin*, pada halaman ini admin dapat melakukan pengelolaan konten pada sistem informasi dan melakukan *approval* terhadap pengajuan surat pengguna. Berikut tampilan halaman *dashboard admin* pada gambar 3.30.

Gambar 3.30 Halaman *Dashboard Admin*

3.8.10. Halaman *Dashboard Kepala Desa*

Halaman ini menampilkan *dashboard* dari kepala desa, pada halaman ini kepala desa dapat melihat laporan pengajuan dari masyarakat dan melakukan tanda tangan secara virtual jika sedang tidak berada di kantor Desa Mandalahaji. Berikut tampilan halaman *dashboard* kepala desa pada gambar 3.31.

Gambar 3.31 Halaman *Dashboard Kepala Desa*

3.8.11. Halaman Pengajuan Surat

Halaman pengajuan surat adalah halaman yang menampilkan *form* untuk melakukan pengajuan surat pada sistem. Berikut tampilan halaman pengajuan surat pada gambar 3.32.

Gambar 3.32 Halaman Pengajuan Surat

3.8.12. Halaman Pengajuan Surat Saya

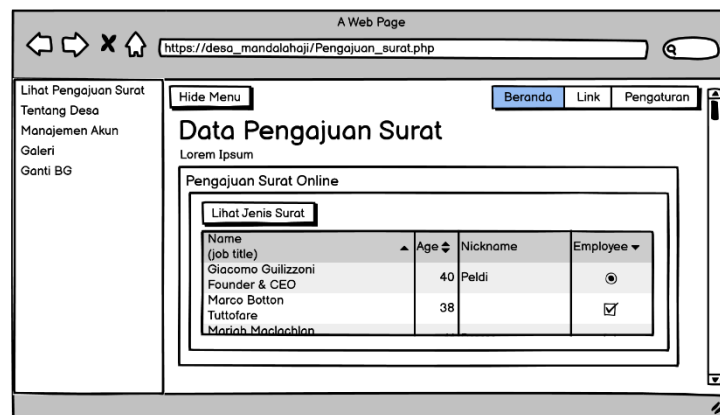
Halaman ini menampilkan halaman pengajuan surat yang diajukan oleh pengguna yang telah melakukan login pada sistem. Berikut tampilan halaman pengajuan surat saya pada gambar 3.33.

Name (job title)	Age	Nickname	Employee
Giacomo Guilizzoni Founder & CEO	40	Peldi	<input type="radio"/>
Marco Botton Tuttofare	38		<input checked="" type="checkbox"/>
Mariah MacLachlan	41	Patata	<input type="checkbox"/>

Gambar 3.33 Halaman Pengajuan Surat Saya

3.8.13. Halaman Data Pengajuan Surat

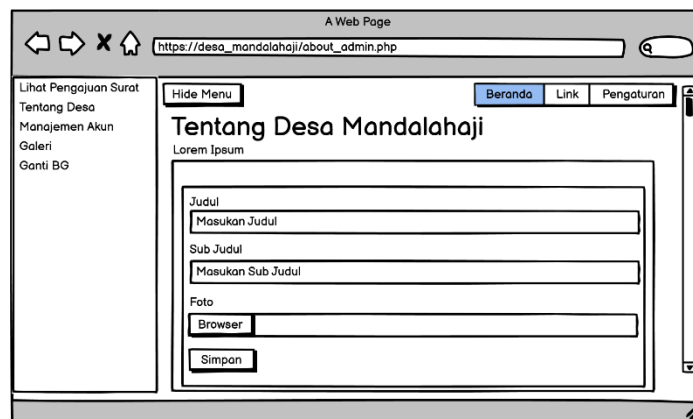
Halaman ini menampilkan data pengajuan surat pada *dashboard admin* halaman ini dapat digunakan untuk menyetujui pengajuan atau membatalkan pengajuan surat yang dilakukan oleh masyarakat. Berikut tampilan halaman data pengajuan surat pada gambar 3.34.



Gambar 3.34 Halaman Data Pengajuan Surat

3.8.14. Halaman Tentang Desa

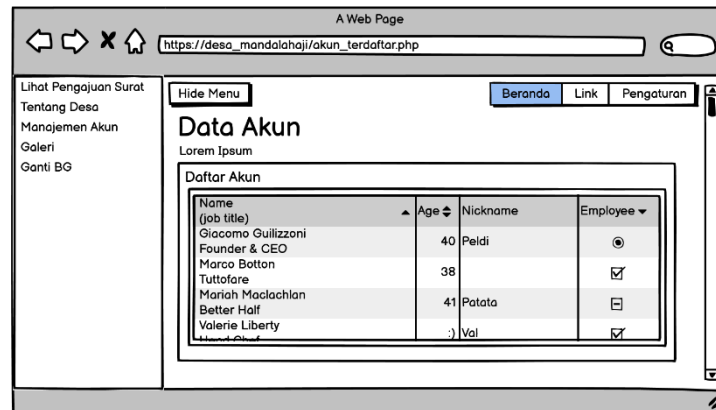
Halaman ini menampilkan halaman tentang desa pada *dashboard admin* dan digunakan untuk membuat, mengedit dan menghapus konten untuk profile Desa Mandalahaji. Berikut tampilan dari halaman tentang desa pada gambar 3.35.



Gambar 3.35. Halaman Tentang Desa

3.8.15. Halaman Manajemen Akun

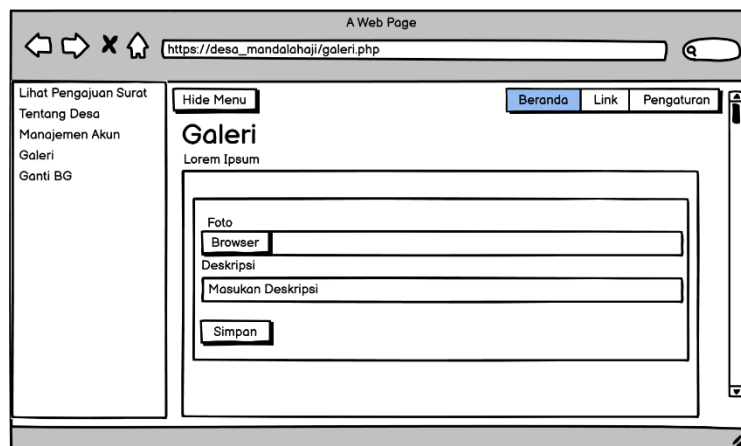
Halaman ini menampilkan akun yang terdaftar pada sistem, pada halaman ini juga admin dapat melakukan penambahan akun, edit dan hapus akun. Berikut tampilan dari halaman manajemen akun pada gambar 3.36.



Gambar 3.36 Halaman Manajemen Akun

3.8.16. Halaman Galeri

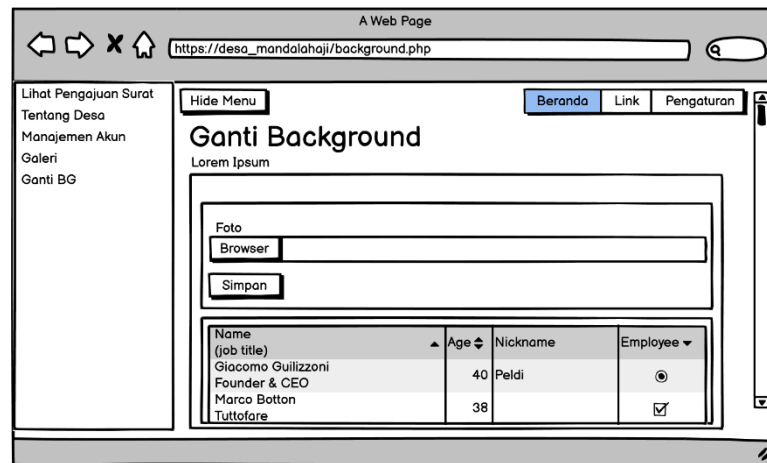
Halaman ini menampilkan konten yang berupa foto yang akan ditampilkan pada halaman beranda sistem informasi, pada halaman ini dapat melakukan penambahan dan hapus foto. Berikut tampilan halaman galeri pada gambar 3.37.



Gambar 3.37 Halaman Galeri

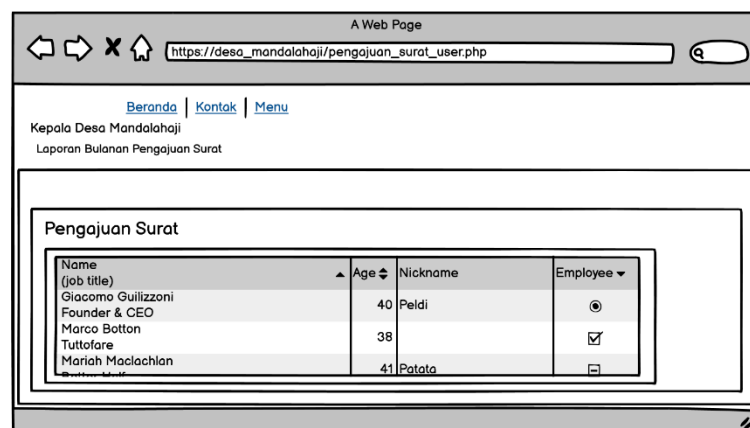
3.8.17. Halaman Ganti *Background*

Halaman ini menampilkan halaman ganti *background* pada halaman *dashboard admin*, pada halaman ini admin dapat mengganti *background* pada halaman beranda sistem. Berikut tampilan halaman ganti *background* pada gambar 3.38.

Gambar 3.38 Halaman *Ganti Background*

3.8.18. Halaman Laporan Pengajuan Surat

Halaman ini menampilkan laporan berupa pengajuan surat dari masyarakat dan data ini akan ditampilkan dalam tenggat waktu satu bulan laporan, dan halaman ini diakses oleh kepala desa. Berikut tampilan halaman laporan pengajuan surat pada gambar 3.39.



Gambar 3.39 Halaman Laporan Pengajuan Surat

3.8.19. Halaman Tanda Tangan Virtual

Halaman ini menampilkan halaman yang menyediakan tanda tangan virtual yang akan di akses oleh kepala desa mandalahaji jika sedang tidak berada di kantor Desa Mandalahaji untuk melakukan tanda tangan pada surat administrasi yang dibutuhkan. Berikut tampilan halaman tanda tangan virtual pada gambar 3.40

Gambar 3.40 Halaman Tanda Tangan Virtual

3.9. Rencana Pengujian

Pengujian merupakan tahap pengujian program yang telah dibangun, tujuannya adalah untuk menguji kelayakan program dan mengecek kembali apabila ada kesalahan maka program akan dibetulkan dan jika sudah berjalan dengan baik. Pada pengembangan *waterfall*, tahap pengujian alpha dilakukan pada tahapan *Construction* sedangkan tahapan pengujian Beta dilakukan pada tahapan *Deployment*.

3.9.1. Alpha Testing

Pengujian yang akan digunakan dalam penelitian ini adalah pengujian *Black Box Testing*, menurut pengujian aplikasi dengan *Black Box Testing* bertujuan melihat program tersebut sama dengan tugas program tersebut tanpa mengetahui kode program yang dipakai. Dalam *Black Box Testing* akan diuji fungsionalitas sistem apakah sudah sesuai dengan fungsinya.

Untuk melakukan pengujian terhadap aplikasi/*website*, penulis akan menguji setiap fungsi antara lain pendaftaran akun *user*, login user, admin, kepala desa dan pengajuan surat, persetujuan surat, tanda tangan online dan pencetakan. Berikut merupakan beberapa tabel untuk rencana pengujian *Alpha*:

Tabel 3.19 Pengujian *Alpha* Halaman Beranda Sistem

Skenario Uji	Hasil yang diharapkan	Hasil pengujian	
		Berhasil	Tidak berhasil

Jika pengguna memilih halaman menu beranda	Halaman akan tetap di halaman beranda		
Jika pengguna memilih halaman tentang desa	Halaman akan menampilkan Profil Desa Mandalahaji		
Jika pengguna memilih halaman pelayanan	Halaman akan menampilkan pelayanan yang ada di desa mandalahaji		
Jika pengguna memilih halaman Berita	Halaman akan menampilkan halaman berita		
Jika pengguna memilih halaman kontak	Halaman akan menampilkan halaman kontak		
Jika pengguna memilih halaman menu	Maka menu bar menu akan menampilkan dropdown berupa login		

Tabel 3.20 Pengujian *Alpha Login*

Skenario Uji	Hasil yang diharapkan	Hasil pengujian	
		Berhasil	Tidak berhasil
Ketika pengguna memasukkan	Muncul pesan popup password harus di isi!		

<i>username</i> lalu klik tombol <i>login</i>			
Ketika pengguna memasukkan <i>password</i> lalu klik tombol <i>login</i>	Muncul pesan pop up <i>username</i> harus di isi!		
Jika pengguna memasukkan <i>username</i> dan <i>password</i> yang salah dan klik tombol <i>login</i>	Muncul pesan pop up <i>username</i> dan <i>password</i> yang anda masukan salah!		
Jika pengguna memasukkan <i>username</i> dan <i>password</i> dengan benar lalu klik tombol login	Halaman akan berpindah sesuai hak akses. Jika pengguna akan ke <i>dashboard</i> pengguna jika admin akan ke halaman <i>dashboard</i> admin		
Jika pengguna mengklik tombol daftar	Halaman akan menampilkan form registrasi pendaftaran		
Jika pengguna mengklik tombol beranda	Halaman akan kembali ke beranda sistem		

Tabel 3.21 Tabel Pengujian *Alpha* Halaman *Dashboard User*

		Hasil pengujian
--	--	-----------------

Skenario Uji	Hasil yang diharapkan	Berhasil	Tidak berhasil
Jika pengguna melakukan login dengan benar	Sistem akan menampilkan halaman dashboard user		
Jika pengguna telah melakukan registrasi lalu melakukan login	Sistem akan menampilkan data dari pengguna seperti NIK, Nama, Alamat dan lain sebagainya.		
Jika pengguna mengklik tombol pengajuan surat	Sistem akan menampilkan <i>form</i> pengajuan surat		
Jika pengguna mengirimkan pengajuan surat	Maka sistem akan mengirimkan pengajuan tersebut kepada admin, lalu muncul pesan <i>pop up</i> “pengajuan telah dikirim”		
Jika pengguna memilih tombol pengajuan surat saya	Sistem akan menampilkan halaman pengajuan yang dilakukan pengguna dengan pesan proses, sudah selesai, silahkan ambil.		

Jika pengguna memilih menu pengaturan dan <i>logout</i>	Sistem akan menampilkan <i>pop up</i> “apakah anda yakin untuk keluar?” jika iya maka sistem akan menampilkan halaman <i>login</i>		
Jika pengguna melakukan pengajuan surat dan mengisi <i>form</i> jika salah satu <i>form</i> kosong lalu klik simpan	Sistem akan menampilkan peringatan pada <i>text input danger</i> berwarna merah sesuai dengan text inputan yang tidak di isi oleh pengguna		

Tabel 3.22 Tabel Pengujian *Alpha* Halaman *Dashboard Admin*

Skenario Uji	Hasil yang diharapkan	Hasil pengujian	
		Berhasil	Tidak berhasil
Jika admin melakukan login	Sistem menampilkan halaman dashboard admin		
Jika admin memilih menu lihat pengajuan surat	Sistem akan menampilkan halaman pengajuan surat dari <i>user</i> , pada halaman ini admin dapat meambah, mengedit dan menghapus pengajuan surat. Termasuk		

	melakukan <i>approval</i> surat		
Jika admin memilih halaman Tentang Desa	Sistem akan menampilkan halaman profile desa, halaman ini dapat digunakan untuk mengelola konten profil desa		
Admin memilih halaman manajemen akun	Sistem akan menampilkan halaman akun yang terdaftar pada sistem. Pada sistem ini admin dapat mengelola akun yang terdaftar pada sistem		
Admin memilih halaman galeri	Sistem menampilkan halaman galeri		
Admin memilih halaman ganti <i>background</i>	Sistem menampilkan halaman ganti <i>background</i>		

Tabel 3.23 Tabel Pengujian Alpha Halaman *Dashboard* Kepala Desa

Skenario Uji	Hasil yang diharapkan	Hasil pengujian	
		Berhasil	Tidak berhasil

Jika kepala desa melakukan login dengan benar	Sistem akan menampilkan halaman <i>dashboard</i> kepala desa		
Jika kepala desa memilih lihat laporan pengajuan surat	Sistem akan menampilkan data pengajuan surat berdasarkan bulan.		
Jika kepala desa mengklik tombol pengajuan surat	Sistem akan menampilkan <i>form</i> pengajuan surat		
Jika kepala desa memilih halaman tanda tangan surat	Maka sistem akan menampilkan surat yang harus ditandatangani		
Jika memilih halaman log out	Sistem akan menampilkan pesan pop up “Apakah anda yakin?” jika iya maka sistem akan mengembalikannya ke halaman login.		

3.9.2 Beta *Testing*

Pengujian beta dilakukan di lingkungan pengguna tanpa kehadiran pihak pembangun aplikasi. Pengujian beta merupakan pengujian yang bersifat langsung di lingkungan yang sebenarnya. Kuesioner Beta merupakan media yang digunakan pengguna aplikasi untuk memberikan penilaian terhadap aplikasi yang dibangun. Berdasarkan kuesioner tersebut akan dilakukan penghitungan agar dapat diambil kesimpulan terhadap penilaian terhadap aplikasi yang dibangun. Kuesioner ini terdiri

dari pertanyaan dengan menggunakan skala likert dari skala 1 sampai 4. Tabel 3.19 adalah skor penilaian yang diberikan dengan menggunakan skala likert untuk setiap pertanyaan kuisisioner.

Tabel 3.24 Tabel Penelitian Dengan Kuisiorer

No	Pernyataan	Alternatif Jawaban				
		STS	TS	N	S	SS
Perceived Ease of Use (PEU)						
1.	Sistem informasi Pelayanan Desa Mandalahaji mudah dipelajari					
2.	Saya mudah memahami menjalankan sistem informasi pelayanan Desa Mandalahaji					
3.	Saya menjadi mahir menggunakan sistem Informasi pelayanan Desa Mandalahaji					
4.	Sistem informasi pelayanan Desa Mandalahaji mudah digunakan					
5.	Fitur-fitur pada sistem informasi pelayanan Desa Mandalahaji mudah di ingat					
Perceived Usefulness (PU)						
1.	Pengajuan pelayanan surat saya menjadi lebih cepat dengan sistem informasi Desa Mandalahaji					
2.	Sistem informasi pelayanan Desa Mandalahaji memberikan saya dalam mendapatkan pelayanan					
3.	Saya menjadi lebih produktif ketika memberikan pelayanan kepada masyarakat dengan menggunakan sistem informasi pelayanan Desa Mandalahaji					
4.	Sistem informasi pelayanan Desa Mandalahaji mampu meningkatkan efektifitas pelayanan kepada masyarakat					
5.	Proses pengajuan pencetakan surat menjadi lebih mudah dengan sistem informasi pelayanan Desa Mandalahaji					
6.	Sistem informasi ini berguna untuk memberikan pelayanan kepada masyarakat khusus nya Desa Mandalahaji					
Attitude Toward Using (ATU)						

1.	Saya merasa senang ketika menggunakan sistem informasi pelayanan desa mandalahaji ketika membutuhkan pelayanan					
2.	Sistem informasi pelayanan desa mandalahaji memudahkan saya ketika mendapatkan pelayanan dari Desa Mandalahaji					
3.	Fitur dan tampilan pada aplikasi sistem informasi Desa Mandalahaji menarik					
4.	Sistem informasi pelayanan Desa Mandalahaji menyulitkan saya dalam bekerja					
Behavioral Invention (BI)						
1.	Saya dapat menggunakan sistem informasi Pelayanan Desa Mandalahaji kapan saja					
2.	Saya dapat menggunakan sistem informasi pelayanan Desa Mandalahaji dalam kondisi apapun					
3.	Saya akan terus menggunakan sistem informasi pelayanan Desa Mandalahaji					
4.	Saya berharap dapat menggunakan sistem informasi pelayanan Desa Mandalahaji ketika tidak berada di wilayah Desa Mandalahaji					
Actual Usage (AU)						
1.	Saya selalu menggunakan sistem informasi pelayanan Desa Mandalahaji dalam bekerja					
2.	Saya menghabiskan waktu cukup lama dalam menggunakan sistem sistem informasi pelayanan Desa Mandalahaji					

Tabel 3.25 Tabel Skala

Tingkat Kepuasan	Skala
Sangat Setuju	5
Setuju	4
Netral	3
Tidak Setuju	2
Sangat Tidak Setuju	1

Berdasarkan data hasil kuisioner , didapat persentase masing-masing jawaban dengan menggunakan rumus :

$$Y = \frac{\varepsilon(N.R)}{Skor\ ideal} \times 100\%$$

Keterangan:

Y = Nilai persentase yang dicari

X = Jumlah nilai kategori jawaban dikalikan dengan frekuensi $\varepsilon = N.R$

N = Nilai dari setiap jawaban

R = Frekuensi

Skor ideal = Nilai tertinggi dikalikan dengan jumlah sampel (4 X 5 =20)

BAB IV

JADWAL KERJA

Tabel 4.1. Jadwal Kerja Penelitian

No	Tahapan	Target Output	Bulan															
			Desember				Januari				Februari				Maret			
			1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1	Melakukan pengumpulan data dengan studi literatur, observasi dan wawancara																	
2	Menyusun proposal penelitian																	
3	Melaksanakan seminar proposal																	

4	Melakukan analisis kebutuhan pengguna																	
5	Melakukan perancangan sistem yang akan dibuat																	
6	Menulis kode program untuk sistem																	
7	Merancangan sisem berbasis web																	
8	Melakukan pengujian sistem																	
9	Melaksanakan seminar terbuka																	
10	Melaksanakan siding akhir/ skripsi																	
11	Implementasi sistem																	

DAFTAR PUSTAKA

A.S., Rosa dan Salahuddin, 2015. Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek Bandung: Informatika Bandung. In: *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek Bandung: Informatika Bandung*. Bandung: s.n., p. 275.

Antonius Wahyu Hidayat, 2016. IMPLEMENTASI ENKRIPSI DATABASE MENGGUNAKAN TRANSPARENT DATA ENCRYPTION PADA DATABASE ENGINE ORACLE. *STMIK MDP Palembang*, Volume 2, p. 3.

Arif Budiman Noviardi, 2016. Penerapan Keamanan Penggunaan Data Pada Database Kepegawaian Menggunakan Teknik Transparent Data Encryption (Studi Kasus Sekolah Tinggi Pakayumbuh). *SATIN*, Volume 2, p. 2.

Arif, A. Y., 2019. *Pengertian MySQL, Kelebihan dan Kekurangan*. s.l.:s.n.

Dr. Anwar Pasha Abdul Gafoor Deskmukh, Dr. Rizayuddin Qureshi, 2014. Transparent Data Encryption - Solution for Security of Database Content. *College of Computers And Information Technology, Universitas of Tabuk*, Volume 2, No 3, p. 126.

Enterprise, J., 2015. *Pengenalan Pemrograman Komputer*. Jakarta: Elex Media Komputindo.

Enterprise, Jubille, 2015. *Pengenalan Pemrograman Komputer*. Jakarta: Elex Media Komputindo.

Fatansyah, 2017. *Buku Teks Komputer Basis Data*. s.l.:s.n.

Ginanjari, 2016. Analisis dan Perancangan Sistem Informasi It-Helpdesk (Studi Kasus :pt.Lontar Papyrus Pulp & Paper Industry. *Tekno Inf*, Volume 2, pp. 296-305.

Harma Oktafia Lingga Wijaya, 2017. E-Voting Berbasis Website Pada Pemilihan Kades Dirantau Jaya (LAKE) Dengan Keamanan Data Menggunakan Ekripsi Base 64. *jurasiq*, Volume 2, p. 35.

Hartono, Dwiarto Utomo,dkk, 2017. Elektronik Governmetn Pemberdayaan Pemerintahan Potensi Desa Berbasis WEB. Volume 6, p. 1.

Humaira, B., 2015. *Dream Weaver CS6, CSS dan JQuery*. Yogyakarta: ANDI OFFSET.

Ivananda, 2015. *Diagram UML*. Jakarta: Andi Mano.

LARMAN, C., 2005. *Applying UML and Patterns*. 3rd ed. NJ: Prentice Hall.

Liana, S. N. F. M., 2018. APLIKASI DAN DEKRIPSI DATA MENGGUNAKAN TINY ENCRYPTION ALGORITHM (TEA) BERBASIS JAVA. *semanTIK*, Volume 4, p. 1.

Maria, Endang, 2016. *Skripsi Sistem Informasi Sekolah Berbasis WEB*. s.l.:s.n.

Muhtar, M., Putro, 2017. Penentu Transparansi Pemerintah Daerah : Sebuah Studi Pada Pemerintah Daerah di Indonesia. *Jurnal Akuntansi dan Auditing Indonesia*, 21(2), pp. 141-146.

Murray, M. C., 2010. Database Security: What Students Need to Know. *Journal Information of Technology Education*, Volume 9, pp. 61-77.

Ningrum, d. F. C., 2019. Pengujian Black Box pada Aplikasi Sistem Seleksi Sales Terbaik Menggunakan Teknik Equivalence Partitions. *Jurnal Informatika Universitas Pamulang*, Volume 4, pp. 125-130.

Rijayanti Rita, 2012, 2012. Memaksimalkan Keamanan Sistem Dengan Konsep Encryption dan De Militeries Zone.

Rudiawan, 2018. *normalisasi-database*. [Online]
[Accessed 24 12 2020].

Ryan Muhammad Bahrudin, Mohammad Ridwan, Hardjito S Darmojo, 2019. Penerapan Helpdesk Ticketing system Dalam Penanganan Keluhan Penggunaan Sistem Informasi Berbasis Web. *JUTIS*, Volume 7, p. 1.

Sani, R. A., 2017. Konstruksi Timbangan Digital Menggunakan Load Cell Berbasis Arduino Uno Dengan Tampilan LCD (Liquid Crystal Digital). *Jurnal Hasil Penelitian Bidang Fisika*, pp. 16-19.

Sofyani & Prayudi, 2018. Implementation Of Performance Based Budgets In Regional Government With "A" Performace Accountability. *Jurnal Ilmiah Akuntansi dan Bisnis*, 13(01), p. 06.

Sudrajat, A. Q., 2016. Implementasi Database Menggunakan Transparent Data Encryption pada Database Engine Oracle.

Sugiyono, 2017. *Penelitian Kualitatif*, AlfaBeta. Bandung: s.n.

Sultiani, A., 2019. *Jurnal Peranan E-government Dalam Rangka Mewujudkan Potensi dalam Desa Karang Sari*.


Sutrisna, E., 2018. Implementasi Knowledge Management System Berbasis Website Dengan Model Spiral Pada PT. Trans Retail Indonesia. *Jurnal Informatika Universitas Pamulang*, pp. 64-67.

Wibowo, I., Susanto, B., & Karel, J., 2011. Penerapan Algoritma Kriptografi Asimetris rsa Untuk Keamanan Data di Oracle. *Jurnal Informatika*, Volume 1, pp. 32-68.

Yuli Kartika, 2017. Aplikasi E government Pada Sinar Harapan di Kabupaten Tenggamas.

LAMPIRAN

1. Surat Permohonan Izin Penelitian



**SEKOLAH TINGGI
TEKNOLOGI BANDUNG**
Jl. Soekarno Hatta No. 378 Bandung - 40235
Telp. (022) 522 4000 | website: www.sttbandung.ac.id | email: info@sttbandung.ac.id

Nomor : 3352/WKI/STTBANDUNG/LPPIB/XII/2020
Lamp : -
Hal : Permohonan Izin Penelitian


Kepada
Kantor Kepala Desa Mandalahaji
Jl. Raya cigagak Rt 54/17 Desa Mandalahaji Kecamatan Pacet
Kabupaten Bandung 40385

Dengan Hormat,
Dalam rangka pelaksanaan kegiatan Tri Dharma Perguruan Tinggi yang terdiri dari kegiatan Pendidikan, Penelitian, dan Pengabdian kepada masyarakat bagi mahasiswa kami dari Program Studi Teknik Informatika - Sekolah Tinggi Teknologi Bandung (TIF-STTB), maka melalui surat ini perkenankanlah kami menyampaikan permohonan izin penelitian bagi mahasiswa kami yang akan melaksanakan Tugas Mata Kuliah Tugas Akhir, yang disebutkan dibawah ini:

Nama : Ihsan Nurhakim Aziz
NPM : 17111329
Program Studi : Teknik Informatika

Besar harapan kami semoga Bapak/Ibu berkenan mempertimbangkan dan mengabulkan permohonan kami.
Atas perhatian Bapak/Ibu kami ucapkan terima kasih.


Bandung, 07 Desember 2020
Wakil Ketua I Bidang Akademik



STTB
SEKOLAH TINGGI
TEKNOLOGI BANDUNG
Daany Achi Rismayadi, S.SI., M.Kom

Your Partner To Global Competition

2. Surat Balasan Permohonan Izin Penelitian

		PEMERINTAH KABUPATEN BANDUNG KECAMATAN PACET DESA MANDALAHAJI Jalan Cigagak No. 17 Telp. 081224999082 Pacet Kode Pos 40385
Nomor	: 141.1/ /Ds/I/2021	Pacet, 12 Januari 2021
Sifat	: Penting	Kepada :
Lampiran	: -	Yth, Sekolah Tinggi Teknologi
Perihal	: Surat Balasan Permohonan Izin Penelitian	Bandung (STTB)
		Di
		Tempat

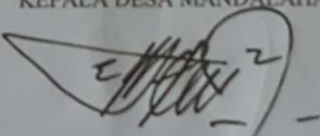
Berdasarkan Surat Saudara nomor : 3352/WKI/STTBANDUNG/LPPIB/XII/2020 Tanggal 7 Desember 2020 perihal Permohonan Izin Pelaksanaan Kegiatan Tri Dharma Perguruan Tinggi bagi mahasiswa dibawah ini.

Nama : IHSAN NURHAKIM AZIZ
NPM : 17111329
Program Studi : Teknik Informatika

Untuk melaksanakan Pendidikan, Penelitian dan Pengabdian kepada masyarakat mulai tanggal 26 Desember 2020 s.d 4 Januari 2021, berdasarkan surat izin yang bersangkutan dari pihak Perguruan Tinggi.

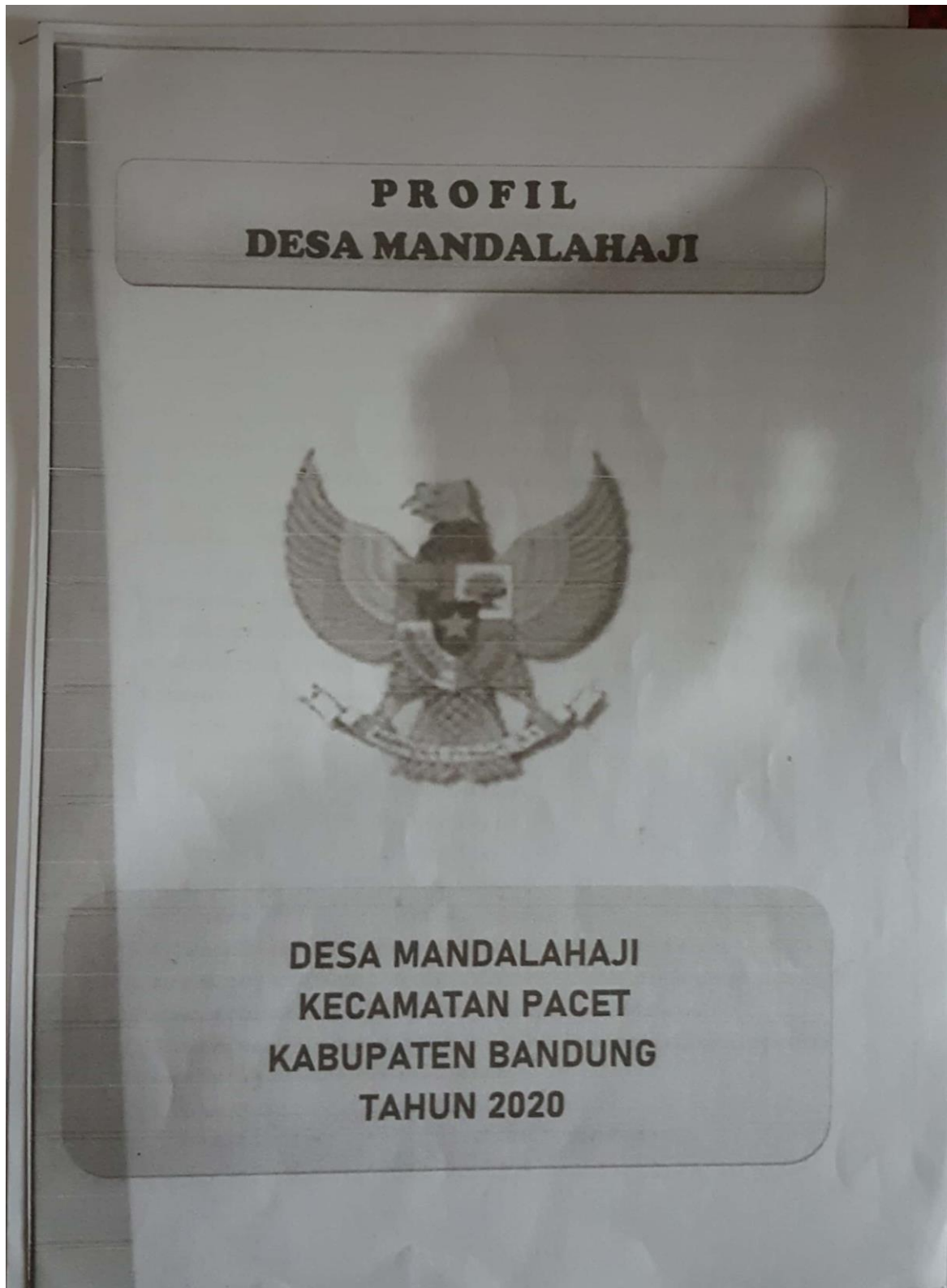
Demikian surat ini kami sampaikan untuk di ketahui dan dijadikan bahan selanjutnya.

KEPALA DESA MANDALAHAJI



ENDANG JUMARA

3. Buku Profil Desa Mandalahaji



4. Daftar Wawancara

DAFTAR PERTANYAAN WAWANCARA

Daftar pertanyaan wawancara ini berfungsi untuk menjawab rumusan masalah pada penelitian yang berjudul "**Sistem Informasi Desa Mandalahaji Menggunakan Teknologi Transparent Data Encryption Sebagai Keamanan Database**". Berikut daftar pertanyaan wawancara untuk menjawab rumusan masalah bagaimana implementasi sistem informasi desa mandalahaji serta siklus pembuatan surat di masa pandemic covid-19.

Laporan Hasil Penelitian

(Hasil Interview)

Tanggal : 12 Januari 2021
Waktu : 10:00 – 12:00 WIB
Narasumber : Bapak Iwan Abdurahman
Jabatan : Sekertaris Desa

Daftar pertanyaan:

1. Apakah sistem yang saat ini berjalan mampu membantu kinerja dari dari karyawan desa mandalahaji dan warga masyarakat yang ada di wilayah kerja mandalahaji?
2. Bagaimana proses pelayanan yang ada di sistem informasi yang diterapkan saat ini?
3. Berapa jumlah karyawan di kantor desa mandalahaji?
4. Apakah tugas dan fungsi berbagai bidang pelayanan sudah berjalan dengan baik?
5. Dokumen apa saja yang digunakan masyarakat untuk mengajukan surat-surat penting?
6. Apa saja catatan yang dijadikan acuan sebagai proses pengesahan dan peng acc an suatu surat yang diajukan masyarakat desa mandalahaji?
7. Berapa lama masa berlaku surat yang diajukan masyarakat seperti SKU dan lain-lain?

8. Apakah terdapat perbedaan pelayanan di Desa Mandalahaji dalam masa pandemic dan sebelum masa pandemic?
9. Berapa kali kira-kira masyarakat mengajukan pembuatan surat dalam kurun waktu 1 bulan?
10. Surat apa yang sering kali diajukan masyarakat?
11. Berapa luas cakupan wilayah Desa Mandalahaji ?
12. Berapa wilayah yang dinaungi Desa Mandalahaji ?
13. Ada berapa bidang pelayanan masyarakat di desa mandalahaji ?
14. Bagaimana pelayanan kepada masyarakat dimasa pandemic?
15. Bagaimana alur proses pembuatan surat pada sistem yang sedang berjalan ?

Jawaban :

1. Di waktu sebelumnya memang pernah di terapkan satu sistem informasi di desa mandalahaji yang dibuat oleh diskominfo dan di distribusikan untuk setiap desa, jadi setiap desa yang ada di wilayah kabupaten bandung pasti memiliki sistem informasi sebagai bentuk pelayanan digital sarana informasi bagi masyarakat. Namun hal ini menjadi kendala tersendiri, dimana untuk akses / membuka web itu sendiri terkadang masih sangat sulit pertama karena jaringan yang tidak merata dan yang kedua selalu adanya kendala pada sistem informasi yang dibuat untuk desa mandalahaji tersebut. Kadang website hilang dan bahkan tidak bisa diakses sama sekali, kami orang awam mengenai sistem yang dipakai tidak tau menahu mengenai persoalan atau gangguan yang ada disistem, jadi kami berhenti menggunakan sistem informasi desa tersebut lalu beralih ke sistem manual, yang dimaksud sistem manual disini ialah untuk penyampaian informasi kepada masyarakat seperti adanya bantuan tunai, keterbukaan dana dan lain-lain yakni dengan cara dilakukannya pemanggilan tiap-tiap RW yang ada di masing- masing dusun yang berjumlah 4 Dusun 21 RW(Rukun Tetangga) dan 72 RT (Rukun Tetangga), setelah itu informasi disampaikan kepada rt dan

rw lalu informasi tersebut disampaikan Kembali ke masyarakat di masing2 kampung oleh ketua rt dan rw masing-masing.

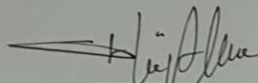
2. Karena sesuai dengan permasalahan diatas sistem yang ada saat ini belum bisa digunakan sepenuhnya dikarenakan masih terdapat kendala dalam penggunaan sistem informasi tersebut.
3. Untuk jumlah total keseluruhan staff sekertaris desa dan kepala desa total ada 15 orang dengan berbagai bidang.
4. Sudah sesuai dengan SOP pelayanan dan Tupoksi masing-masing dari aparatur desa.
5. Untuk dokumen yang digunakan masyarakat sebagai syarat pengajuan surat yaitu menggunakan Kartu Keluarga(KK) dan photocopy Kartu Tanda Penduduk (KTP). Karena dalam pengajuan ini validitas data sangat penting, karena data sudah tercatat dalam sistem untuk ditinjau Kembali apakah benar data si pengaju surat ini benar-benar valid datanya atau tidak.
6. Yang pertama kami tidak semena-mena memberikan surat-surat dengan tanpa adanya pertimbangan, pertimbangan kita adalah di data yang tertera pada no NIK yang ada pada KTP dan KK apakah sipengaju surat tersebut benar-benar valid datanya dan tercatat di data yang ada pada sistem lalu jika data tersebut benar maka proses pertimbangan acc akan dilakukan, dan yang kedua dilakukan wawancara Kembali untuk apa surat itu dibuat, tujuan kenapa surat itu dibuat?. Dan untuk contoh lain seperti pembuatan SKU(Surat Kewirausahaan) ditinjau Kembali apakah benar si pengaju surat SKU tersebut memiliki usaha, jika ada dimana tempat dan bidang usahanya dan lain-lain. Hal-hal seperti itulah yang menjadi pertimbangan kami dalam proses pembuatan dan acc surat-surat.
7. Untuk masa berlaku surat tersebut ada beberapa perbedaan dari masing-masing surat ada yang dua minggu, satu bulan bahkan ada yang berlaku hanya 3 hari setelah pencetakan.
8. Tentu ada, mulai dari pelayanan, perbedaan kinerja juga ada. Karena kami taat kepada pemerintah yang mana dalam prosedur

kerja pandemic saat ini adanya WFH(Work From Home) adanya Shift untuk pergantian karyawan bahkan adanya pembatasan pelayanan terhadap masyarakat. Mungkin seperti itulah perubahan-perubahan yang dirasa tepat untuk mengurangi kekhawatiran penularan covid-19 ini.

9. Hal ini mungkin akan ditanyakan langsung ke front office didepan karena daftar tamu untuk pengajuan surat-surat ini ada di front office, setelah didapatkan data dari front office mengenai jumlah pengajuan surat dalam kurun 1 bulan kurang lebih ada 30-40 pengajuan dalam satu bulan dengan berbagai pengajuan surat.
10. Tentunya beragam dan sesuai kebutuhan masyarakat, mungkin untuk bulan-bulan ini lebih sering kearah SKU,SKTM, pengantar SKCK, Domisili, Keterangan Keliru Nama di KTP dan lain sebagainya.
11. Luas Cakupan Wilayah Desa Mandalahaji 537733 Ha.
12. Ada 4 Dusun Dengan jumlah 21 RW dan 72 RT dengan cakupan wilayah luas sekitar 537,733 Ha Yang dinaungi desa mandalahaji.
13. Bidang Pekayanan pemerintagan social keagamaan ekonomi kesehtana Pendidikan pertahanan.
14. Pelayanan Desa Tidak menentu hari libur tidak terkait ruang dan waktu (kapanpun masyrakat membutuhkan desa mandalahaji siap melayani meskipun dimasa pandemic covid-19 dengan memperhatikan protocol Kesehatan.
15. Untuk alur proses itu sendiri masyarakat dalam melakukan pengajuan surat pengantar harus dibarengi dengan ijin dari RT dan RW setempat setelah itu masyarakat datang ke kantor desa mandalahaji mengisi buku tamu yang disediakan di *front office* lalu menunggu proses antrian setelah itu masyarakat memberikan persyaratan berupa Fotocopy KK dan KTP untuk proses pembuatan dan pendataan sebagai proses pertimbangan pembuatan surat pengantar, setelah surat berhasil diajukan dan di cetak proses selanjutnya ialah di tanda tangani oleh kepala desa mandalahaji. Jika

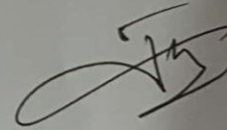
kepala desa mandalahaji tidak dinas luar maka surat dapat di tanda tangani hari itu juga, jika tidak maka akan dilakukan penandatanganan besok hari atau jika memang surat pengantar yang diajukan amat penting maka di wakili oleh sekertaris desa.

Peneliti



Ihsan Nurhakim Aziz

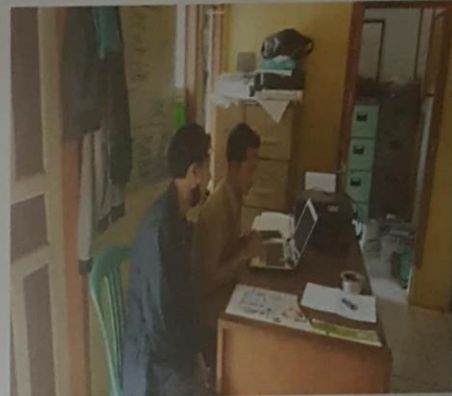
Narasumber



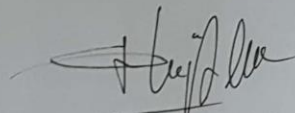
Iwan Abdurahman

Lampiran





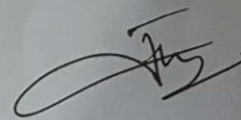
Penulis


Ihsan Nurnakim Aziz

Mengetahui,
Kepala Desa
Mandalahaji


H. Endang Jumara

Narasumber


Iwan Abdurahman

DOKUMENTASI

