

**RANCANG BANGUN APLIKASI E-VOTING PEMILIHANKE
PALA DESA BERBASIS WEB DI DESA BAROS
KECAMATAN ARJASARI**

SKRIPSI

Karya tulis sebagai syarat memperoleh gelar sarjana Komputer d
ari fakultas informasi Universitas Bale Bandung

Disusun oleh:

**FRADITA AUDINA
NPM.301170013**



PROGRAM STRATA 1

**PROGRAM STUDI TEKNIK INFORMATIKA FAKULTAS
TEKNOLOGI INFORMASI
UNIVERSITAS BALE BANDUNG BANDUNG**

2021

LEMBAR PERSETUJUAN PEMBIMBING

RANCANG BANGUN APLIKASI E-VOTING PEMILIHAN KEPALA DESA BERBASIS WEB DI DESA BAROS KECAMATAN ARJASARI

Disusun oleh:

FRADITA AUDINA

NPM. 301170013

Telah diterima dan disetujui untuk memenuhi persyaratan mencapai gelar
SARJANA KOMPUTER

Pada

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS BALE BANDUNG**

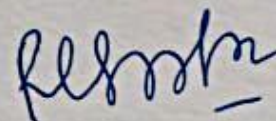
Baleendah, Juli 2021

Pembimbing 1



Yudi Herdiana, S.T., M.T
NIK. 04104808008

Pembimbing 2



Rosmalina, S.T., M.KOM
NIK. 04104808122

LEMBAR PERSETUJUAN PENGUJI

RANCANG BANGUN APLIKASI E-VOTING PEMILIHAN KEPALA DESA BERBASIS WEB DI DESA BAROS KECAMATAN ARJASARI

Disusun oleh:

FRADITA AUDINA

NPM. 301170013

Telah diterima dan disetujui untuk memenuhi persyaratan mencapai
gelar

SARJANA KOMPUTER

Pada

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS BALE BANDUNG**

Baleendah, Juli 2021

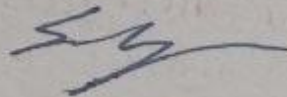
Disetujui Oleh :

Penguji 1



Khilda Nistrina, S.Pd., M.Sc
NIK. 04104808122

Penguji 2



Yaya Suharya, S.KOM., M.T
NIK. 01043170007

LEMBAR PENGESAHAN PROGRAM STUDI
RANCANG BANGUN APLIKASI E-VOTING PEMILIHAN KEPALA DESA
BERBASIS WEB DI DESA BAROS KECAMATAN ARJASARI

Disusun oleh:

FRADITA AUDINA

NPM. 301170013

Telah diterima dan disetujui untuk memenuhi persyaratan mencapai
gelar

SARJANA KOMPUTER

Pada

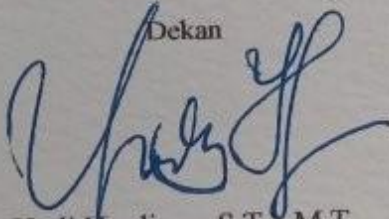
PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS BALE BANDUNG

Baleendah, Juli 2021

Disetujui oleh :

Mengetahui,

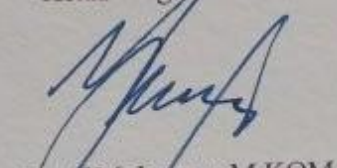
Dekan



Yudi Herdiana, S.T., M.T
NIK. 04104808008

Mengesahkan,

Ketua Program Studi



Yusuf Muharam, M.KOM
NIK. 04104820003

HALAMAN PERNYATAAN

Saya yang bertanda tangan dibawah ini:

Nama : Fradita Audina

NPM : 301170013

Judul Skripsi:

**RANCANG BANGUN APLIKASI E-VOTING PEMILIHAN KEPALA
DESA BERBASIS WEB DI DESA BAROS KECAMATAN ARJASARI
(Studi Kasus Fakultas Teknologi Informasi Universitas Bale Bandung)**

Menyatakan dengan sebenarnya bahwa penulisan skripsi ini berdasarkan hasil penelitian, pemikiran, dan pemaparan asli dari saya sendiri, baik untuk naskah laporan maupun kegiatan *programming* yang tercantum sebagai bagian dari skripsi ini. Jika terdapat karya orang lain, saya mencantumkan sumber yang jelas.

Pernyataan ini saya buat dengan sesungguhnya dan apabila dikemudian hari terdapat penyimpangan dan ketidakbenaran dalam pernyataan ini, maka saya bersedia menerima sanksi akademik berupa pencabutan gelar yang telah diperoleh karena karya tulis ini dan sanksi lain sesuai dengan peraturan yang berlaku di FAKULTAS TEKNOLOGI INFORMASI UNIVERSITAS BALE BANDUNG.

Demikian surat pernyataan ini saya buat dalam keadaan sadar tanpa paksaan dari pihak manapun

Baleendah, Juli 2021

Yang membuat pernyataan

Fradita audina

Npm.301170013

ABSTRAK

Teknologi perlu diterapkan dalam kehidupan demokrasi suatu negara (pilkada) atau pemilihan kepada desa (pilkades). Pelaksanaan pemilihan kepala desa yang dilaksanakan diberbagai daerah membutuhkan pula dukungan sistem e-voting. Sistem e-voting dapat diberdayagunakan sejak pendataan calon pemilih, pendaftaran calon pemilih, pelaksanaan pemungutan suara sampai dengan perhitungan hasil pemungutan suara. Pembuatan aplikasi e-voting pemilihan kepala desa ini di bangun menggunakan HTML, CSS dan Bootstrap sebagai tampilannya. Bahasa pemograman PHP sebagai sistem/ back end dan MYSQL sebagai data base. Hasil dari penelitian ini adalah Aplikasi e-voting Pemilihan Kepala Desa Berbasis website, bisa diakses langsung oleh Admin dan diawasi oleh petugas pelaksanaan e-voting sebagai efisiensi mengelola data yang dibutuhkan oleh admin pemilihan , fitur yang bisa di akses oleh Admin dan di awasi oleh petugas yaitu login untuk admin dan pemilih, menu utama, data calon kepala desa, data pemilih, menu pemilihan, menu hasil e-voting. Permasalahan yang terjadi di desa baros yaitu pemilihan kepala desa masih menggunakan cara semi manual (kartu suara), sehingga banyak kelemahannya yaitu besarnya terjadi kerusakan pada kartu suara dan proses perhitungan yang masih manual memperlambat pengiriman hasil suara dikarenakan pengumuman hasil suara yang akan membutuhkan waktu lama. Metodologi pengembangan sistem yang digunakan yaitu metode air terjun yang dikenal juga dengan istilah metode waterfall. Metode waterfall adalah metode pengembangan perangkat lunak yang memungkinkan pembuatan sistem dilakukan secara terstruktur dan sistematis (berurutan) sesuai dengan situs pengembangan yang ada. Metode ini disebut waterfall atau air terjun karena dalam prosesnya, sistem akan dibuat berurutan setahap demi setahap. Solusi dari permasalahan tersebut maka dalam penelitian ini penulis akan membangun Aplikasi e-voting pemilihan kepala desa berbasis website yang dapat mempermudah Admin dan petugas pemilihan dalam mengelola data calon kepala desa, data pemilih data admin dan hasil pemungutan suara. Kesimpulan dari penelitian ini yaitu penerapan Aplikasi e-voting pemilihan kepala Desa berbasis website bertujuan untuk memudahkan petugas pemilihan dan admin dalam pemilihan calon kepala desa dengan cepat dan baik sehingga dapat berdampak pada kesejahteraan masyarakat desa baros. Aplikasi dibangun se-minimal mungkin untuk memudahkan pengguna dalam mengoperasikan aplikasi e-voting.

Kata Kunci: e-voting, kepala desa, Metode waterfall, mysql, php

ABSTRACT

Technology needs to be applied in the democratic life of a country (pilkada) or elections to villages (pilkades). The implementation of village head elections held in various regions also requires the support of the e-voting system. The e-voting system can be used since the registration of prospective voters, registration of prospective voters, the implementation of voting until the calculation of the results of the vote. The creation of this village head selection e-voting application was built using HTML, CSS and Bootstrap as its appearance. PHP programming language as a system / back end and MYSQL as a data base. The result of this research is the website-based Village Head Election e-voting application, can be accessed directly by admins and supervised by e-voting officers as an efficiency to manage the data needed by election admins, features that can be accessed by admins and supervised by officers. login for admins and voters, main menu, village head candidate data, voter data, election menu, e-voting results menu. The problem that occurs in the village baros is the election of the village head still using semi-manual means (voting cards), so many of the weaknesses are the magnitude of damage to the voting card and the calculation process that is still manual slows the delivery of the results of the vote due to the announcement of the results of the vote. It's going to take a long time. Methodology of system development used is the waterfall method which is also known as the waterfall method is tilan. Waterfall method is a software development method that allows system creation to be done in a structured and systematic (sequential) according to the existing development site. This method is called waterfall or waterfall because in the process, the system will be made sequentially step by step. Solution of the problem then in this study the author will build a website-based village election e-voting application that can facilitate admins and election officers in managing the data of prospective village heads, voter data admin data and voting results. The conclusion of this study is the application of e-voting application for village head election based on the website aims to facilitate election officers and admins in the selection of candidates for village heads quickly and well so that it can have an impact on the welfare of the people of baros village. Applications are built as minimally as possible to make it easier for users to operate e-voting applications.

Keywords: e-voting, village head, waterfall method, mysql, php

KATA PENGANTAR

Puji dan syukur peneliti panjatkan atas kehadiran Allah SWT yang mana berkat limpahan Rahmat dan Karunia-Nya, peneliti dapat menyelesaikan Laporan Skripsi yang berjudul “ RANCANG BANGUN APLIKASI E-VOTING PEMILIHAN KEPALA DESA BERBASIS WEB DI DESA BAROS KECAMATAN ARJASARI ” ini dengan baik dan tepat pada waktunya. terselesaikannya laporan skripsi ini tidak lepas dari bantuan dan bimbingan dari berbagai pihak, baik secara langsung maupun secara tidak langsung. Oleh karena itu, pada kesempatan ini peneliti ingin menyampaikan ucapan terimakasih yang sebesar-besarnya kepada:

1. Allah SWT yang telah memberikan rahmat dan karunia-Nya dalam proses pengerjaan skripsi ini.
2. Yudi Herdiana, S.T, M.T selaku dekan Fakultas Teknologi Informasi Universitas Bale Bandung sekaligus pembimbing 1 saya.
3. Yusuf Muharam, M.Kom selaku ketua Program Studi Teknik Informatika.
4. Rosmalina, S . T ,M.KOM selaku pembimbing ke 2 saya yang sudah mengarahkan saya pada bimbingan skripsi ini.
5. Keluarga dan kerabat yang telah memberikan dukungan serta do’a dalam kelancaran proses pembuatan laporan skripsi ini.
6. Dan juga teman-teman seangkatan yang telah mendukung dalam segala hal.

Bandung, Juli 2021

Fradita Audina

DAFTAR GAMBAR

Gambar 1.1 Metode pengembangan sistem.....	4
Gambar 3.1 kerangka pikir.....	35
Gambar 4.1 Usecase Diagram aplikasi.....	51
Gambar 4.2 Activity diagram admin.....	64
Gambar 4.3 Activity diagram pemilih.....	65
Gambar 4.4 sequence diagram login admin.....	66
Gambar 4.5 sequence diagram data admin.....	66
Gambar 4.6 sequence diagram kepala desa.....	63
Gambar 4.7 sequence diagram data pemilih.....	6
Gambar 4.8 sequence diagram hasil.....	65
Gambar 4.9 sequence diagram laporan.....	65
Gambar 4.10 sequence diagram login pemilih.....	66
Gambar 4.11 class diagram Aplikasi e-voting pemilihan kepala desa.....	67
Gambar 4.12 User Interface Login admin.....	71
Gambar 4.13 User Interface Dashboard admin.....	71
Gambar 4.14 User interface menu daftar kepala desa.....	72
Gambar 4.15 user interface hasil pemilihan.....	72
Gambar 4.16 user interface login pemilih.....	73
Gambar 4.17 user interface home page.....	73
Gambar 4.18 User Interface halaman tentang.....	74
Gambar 4.19 User interface data calon kepala desa.....	74
Gambar 4.20 User interface halaman tata cara pemilihan.....	75
Gambar 5.1 Tampilan Login Admin.....	81
Gambar 5.2 Menu Admin e-voting.....	82

Gambar 5.3 Interface Dasboard Aplikasi e-voting pemilihan.....	83
Gambar 5.4 Menu Administrator.....	83
Gambar 5.5 Menu kepala daerah.....	84
Gambar 5.6 form tambah calon kepala desa.....	85
Gambar 5.7 Menu pilih.....	85
Gambar 5.8 Menu pilih.....	86
Gambar 5.9 tampilan login pilih.....	86
Gambar 5.10 tampilan awal e-voting pemilihan kepala desa.....	87
Gambar 5.11 menu tentang.....	87
Gambar 5.12 tata cara pemilihan.....	88
Gambar 5. 13 Tampilan daftar calon kepala desa.....	89
Gambar 5.14 tampilan tanda terima pemilihan kepala desa.....	90
Gambar 5.15 tampilan pencarian kode tanda pilih.....	90
Gambar 5.16 tampilan e-ktp yang sudah melakukan pemilihan.....	91

DAFTAR TABEL

Table 2.1 Simbol Use Case Diagram.....	30
Tabel 2. 2 Komponen Pada Use case.....	32
Tabel 2. 3 Komponen Class diagram.....	33
Tabel 2. 4 Komponen Activity diagram.....	34
Tabel 2. 5 Komponen Sequence diagram.....	36
Tabel 4.1 Analisis Software.....	44
Tabel 4.2 Tabel Analisis Data.....	47
Tabel 4.3 Deskripsi Aktor pada Usecase Diagram.....	49
Tabel 4.4 Deskripsi Usecase diagram.....	50
Tabel 4.5 Usecase diagram login admin.....	52
Tabel 4.6 Skenario usecase diagram dashboard admin.....	52
Tabel 4.7 Skenario usecase diagram tambah data admin.....	54
Tabel 4.8 Skenario usecase diagram edit data admin.....	55
Tabel 4.9 Skenario usecase diagram hapus data admin.....	56
Tabel 4.10 skenario usecase diagram login pemilih.....	58
Tabel 4.11 skenario usecase diagram home page.....	59
Tabel 4.12 Tabel Administrator.....	68
Tabel 4.13 Tabel kepala desa.....	68
Tabel 4.14 Tabel pemilih.....	69
Tabel 4.15 Tabel hasil.....	70
Tabel 5.1 isi penjelasan tata cara pemilihan.....	88
Tabel 5.2 black box testing.....	93

DAFTAR ISI

Abstrak.....	I
Abstrack.....	Ii
Kata Pengantar.....	Iii
Daftar Gambar.....	Iv
Daftar Tabel.....	V
Daftar Isi.....	vi
BAB I PENDAHULUAN.....	1
1.1. Latar Belakang.....	1
1.2. Rumusan Masalah.....	2
1.3. Batasan Masalah.....	3
1.4. Tujuan.....	3
1.5. Metodologi penelitian.....	4
1.5.1 Metode Pengumpulan Data.....	4
1.5.2 Metode Pengembangan System.....	4
1.5.3 Metode Perancangan.....	7
1.6. Sistematika Penulisan.....	7
BAB II TINJAUAN PUSTAKA.....	9
2.1. Landasan Teori.....	9
2.2. Dasar Teori.....	11
2.2.1. E-Voting.....	11
2.2.2. Pengertian Pemilu.....	13
2.2.3. Pengertian pemilihan kepala desa.....	16
2.2.4. Pengertian Aplikasi.....	18

2.2.5.	Website.....	20
2.2.6.	Pengertian PHP.....	21
2.2.7.	Pengertian MySQL.....	22
2.2.8.	HTML.....	23
2.2.9.	Pengertian CSS.....	24
2.2.10.	Java Script.....	25
2.2.11.	Database.....	25
2.2.12.	Diagram Alir Data (DAD)	26
2.2.13.	ERD (Entity Relationship Diagram).....	27
2.2.14.	XAMPP.....	27
2.2.15.	Bootstrap.....	28
2.2.16.	Web Browser.....	29
2.2.17.	Unified Modeling Language (UML)	30
 BAB III	 METODE PENELITIAN.....	 37
3.1.	Kerangka Pikir.....	37
3.2.	Deskripsi.....	38
3.2.1.	Pengumpulan data.....	39
3.2.2.	Perumusan masalah.....	39
3.2.3.	Analisis kebutuhan.....	40
3.2.4.	Desain.....	42
3.2.5.	Implementasi.....	43
3.2.6.	Verivication.....	43
3.2.7.	Maintenance.....	43
 BAB IV	 ANALISIS DAN PERANCANGAN.....	 44
4.1.	Analisis.....	44
4.1.1	Analisis Masalah.....	44

4.1.2	Analisis Software.....	44
4.1.3	Analisis Pengguna.....	45
4.1.4	User Interface.....	46

4.1.5	Fitur-fitur.....	46
4.1.6.	Analisis Data.....	47
4.2.	Perancangan.....	48
4.2.1	Use case Diagram.....	48
4.2.2	Activity Diagram.....	60
4.2.3	Sequence Diagram.....	67
4.2.4	Class Diagram.....	67
4.2.4.1	Perancangan Tabel.....	67
4.2.5	User Interface.....	70
4.2.5.1	User interface admin.....	70
BAB V	IMPLEMENTASI DAN PENGUJIAN.....	81
5.1	Implementasi.....	81
5.1.1	Implementasi User Interface.....	81
5.1.2.	Implementasi Bahasa Pemograman Php.....	91
5.1.3.	Implementasi XAMPP.....	91
5.1.4	Hasil Implementasi.....	92
5.1.5	Pengujian.....	92
BAB VI	KESIMPULAN DAN SARAN.....	96
6.1.	Kesimpulan.....	96
6.2.	Saran.....	96
DAFTAR PUSTAKA.....		112
LAMPIRAN.....		114
RIWAYAT HIDUP.....		115

BAB 1

PENDAHULUAN

1. Latar belakang

Menurut Badan Penelitian dan Penerapan Teknologi (BPPT), E-Voting merupakan sistem untuk membuat surat suara, memberikan, menghitung, menayangkan perolehan suara, serta menghasilkan dan memelihara jejak audit secara elektronik dan digital. Pemahaman tentang e-voting lebih mengacu pada proses pemanfaatan perangkat elektronik untuk lebih mendukung kelancaran proses dan juga model otomatisasi yang memungkinkan campur tangan minimal dari individu dalam semua prosesnya.

Pembuatan aplikasi E-voting ini di bangun menggunakan HTML, CSS dan Bootstrap sebagai tampilannya. Bahasa pemograman PHP sebagai sistem/ back end dan MYSQL sebagai data base. Hasil dari penelitian ini adalah Aplikasi e-voting pemilihan kepala desa Berbasis *website*, bisa diakses langsung oleh Admin dan Kepala Desa sebagai efisiensi mengelola data keuangan yang dibutuhkan oleh bendahara desa, fitur yang bisa di akses oleh Admin dan Kepala desa .

Permasalahan yang terjadi di Kantor Desa Baros yaitu Proses pemilihan kepala desa, masih menggunakan cara manual, dimana semua masyarakat/penduduk desa tersebut harus dipanggil satu persatu untuk datang ke lokasi pemilihan guna melakukan dan menentukan hak pilihnya masingmasing terhadap calon Kepala Desa yang dikandidatkan, sehingga berbagai kendala ditemukan dalam proses ini, salah satunya adalah saat masyarakat tidak ada ditempat/di desa tersebut, karena mayoritas masyarakat penduduk Desa Baros adalah bertani, jadi kebanyakan masyarakatnya menetap dikebun supaya lebih hemat waktu untuk secara langsung dapat bekerja dikebun setiap harinya, Hal ini dilakukan oleh masyarakat karena jarak tempuh dari desa ke kebun tersebut membutuhkan waktu yang lumayan jauh. Sehingga pada saat akan dilaksanakannya pemilihan kepala desa, semua masyarakat desa yang sedang menetap dikebun dijemput untuk menentukan hak pilihnya.

Metodelogi pengembangan sistem yang digunakan yaitu metode air terjun yang dikenal juga dengan istilah metode *waterfall*. Metode *waterfall* adalah metode pengembangan perangkat lunak yang memungkinkan pembuatan sistem dilakukan secara terstruktur dan sistematis (berurutan) sesuai dengan situs pengembangan yang ada. Metode ini disebut *waterfall* atau air terjun karena dalam prosesnya, sistem akan dibuat berurutan setahap demi setahap

Berdasarkan permasalahan tersebut maka dalam penelitian ini penulis akan membangun Aplikasi E- Voting (*electronic voting*). Dimana e-voting merupakan proses pemungutan suara yang dilakukan secara elektronik (digital) mulai dari proses pendaftaran dan pendataan pemilih, pelaksanaan pemilihan, penghitungan suara, dan pengiriman dan pelaporan hasil suara (purwanti, 2015)

Berdasarkan uraian dari permasalahan diatas penulis ingin mengajukan sebuah judul yakni “ Rancang Bangun Aplikasi E-voting Pemilihan Kepala Desa Di Kantor Desa Baros Berbasis Website”, adanya sistem ini diharapkan bisa membantu dan mengatasi persoalan pada pemilihan kepala desa Kecamatan arjasari.

Kesimpulan dari penelitian ini yaitu Aplikasi E-voting ini dapat mempermudah proses pemilihan kepala desa, karena sebelumnya pemilihan masih dilakukan dengan cara coblos lembar kertas suara, dan sering terjadi kesalahan dalam mencoblos, sehingga banyak kartu yang rusak dan dinyatakan tidak sah. Hal ini membuat proses perhitungan suara menjadi lambat, karena masih menghitung satu persatu lembar kertas suara. Aplikasi ini bertujuan untuk mengurangi kelemahan pada pemilihan kepala desa yang selama ini digunakan di Kecamatan Botupingge. Serta mengurangi penggunaan kertas selama pilkades berlangsung.

1.2 Rumusan Masalah

Dari uraian di atas, rumusan masalah yang akan di bahas dalam penelitian ini yaitu:

1. Bagaimana membangun Aplikasi *e-voting* pemilihan kepala desa Di Kantor Desa Baros yang dapat membantu petugas desa dalam pelaksanaan pemilihan kepala desa secara cepat dan efisien?
2. Bagaimana mengimplementasikan desain sistem *e-voting* pilkades tersebut menjadi sistem berbasis web dengan menggunakan bahasa pemrograman server PHP dan bahasa pemrograman database MySQL.
3. Bagaimana efektifitas penggunaan dan tingkat kelayakan aplikasi *e-voting* pilkades.

1.3 Batasan Masalah

Adapun batasan masalah dalam pelaksanaan penelitian dan pembuatan Aplikasi yaitu:

1. Sistem yang di bangun merupakan sistem *e-voting* pemilihan kepala desa pada desa Baros.
2. Input sistem *e-voting* menggunakan nomor identitas yang tertera pada Kartu Tanda Penduduk atau (KTP).
3. Aplikasi *e-voting* Berbasis Website ini dirancang menggunakan Php dan MySQL.

1.4 Tujuan Penelitian

Tujuan dari penelitian ini yaitu sebagai berikut:

1. Membangun sebuah sistem *e-voting* yang digunakan untuk Pemilihan Kepala Desa (Pilkades) pada desa Baros yang tetep sesuai dengan UU pilkades dan asas pilkades di Indonesia, serta mampu menggantikan sistem voting Pilkades konvensional.

2. Mengimplementasikan sistem *e-voting* dengan menggunakan bahasa pemrograman PHP dan menggunakan database Mysql.
3. Mendapatkan hasil perhitungan suara yang cepat dan akurat.

1.5 Metodologi Penelitian

Dalam pembuatan aplikasi administrasi desa ini melalui beberapa tahapan, salah satunya yaitu tahapan pengumpulan data. Ada tiga teknik pengumpulan data yang digunakan dalam penelitian ini yaitu Wawancara, Observasi dan Studi Pustaka.

1. Observasi

Dalam penelitian ini Penulis melakukan observasi langsung ke Kantor Desa Baros untuk mengamati secara langsung mengenai pengelolaan keuangan desa.

2. Wawancara

Dalam penelitian ini penulis melakukan wawancara kepada Bapak Heru Selaku Sekertaris Desa Baros untuk menanyakan pertanyaan mengenai pengelolaan keuangan desa.

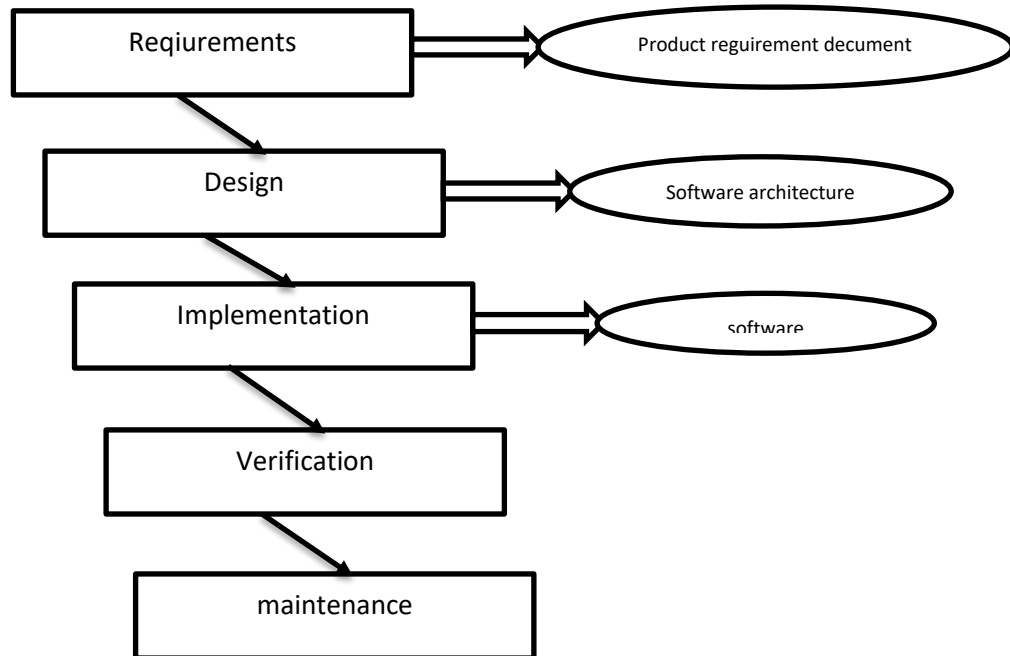
3. Studi pustaka

Studi pustaka dilakukan dengan cara membaca jurnal penelitian terkait dengan objek penelitian sebagai penunjang dan juga mencari beberapa referensi dari *website*, buku, dll.

1.5.1 Metode Pengembangan Sistem

Metode pengembangan sistem yang digunakan pada pembuatan aplikasi *e-voting* pemilihan kepala desa di desa baraos ini adalah metode *Waterfall*. Alasan menggunakan metode ini adalah dikarenakan metode *Waterfall* melakukan pendekatan secara sistematis dan berurutan dalam membangun sebuah sistem. Berikut tahapan metode *Waterfall* (Pressman, 2012)

Tahapan dari metode *waterfall* yaitu sebagai berikut:



Gambar 1.1 Metode pengembangan sistem
(metode *waterfall*)

a. Requirement

Pada tahap ini pengembang harus mengetahui seluruh informasi mengenai kebutuhan *software* seperti kegunaan *software* yang diinginkan oleh pengguna dan batasan *software*. Informasi tersebut biasanya diperoleh dari wawancara, survey, atau pun diskusi. Setelah itu informasi dianalisis sehingga mendapatkan data-data yang lengkap mengenai kebutuhan pengguna akan *software* yang akan dikembangkan.

b. Design

Tahap selanjutnya yaitu Desain. Desain dilakukan sebelum proses *coding* dimulai. Ini bertujuan untuk memberikan gambaran lengkap tentang apa yang harus dikerjakan dan bagaimana tampilan dari sebuah sistem yang

diinginkan. Sehingga membantu menspesifikasikan kebutuhan *hardware* dan sistem, juga mendefinisikan arsitektur sistem yang akan dibuat secara keseluruhan.

c. Implementation

Proses penulisan code ada di tahap ini. Pembuatan *software* akan di pecah menjadi modul-modul kecil nantinya akan digabungkan pada tahap selanjutnya. Dalam tahap ini juga akan dilakukan pemeriksaan lebih dalam terhadap modul yang sudah dibuat, apakah sudah memenuhi fungsi yang sudah dibuat atau belum.

d. Verification

Tahapan verifikasi meliputi pengintegrasian sistem dan juga melakukan testing terhadap aplikasi yang telah dibuat. Sistem akan diverifikasi untuk diuji sejauh mana kelayakannya. Dalam tahapan ini semua modul yang dikerjakan oleh programmer berbeda akan digabungkan kemudian diuji apakah telah sesuai dengan spesifikasi yang ditetapkan atau terdapat kesalahan / eror dalam sistem sebelum kemudian diperbaiki ulang.

e. Maintenance

Maintenance adalah tahapan terakhir dari metode pengembangan *waterfall*. Disini *software* yang sudah jadi akan dijalankan atau di operasikan oleh pengguna nya. Disamping itu dilakukan pula pemeliharaan yang termasuk:

1. Perbaikan kesalahan
2. Perbaikan implementasi unit
3. Peningkatan jasa sistem sesuai kebutuhan baru

1.5.2 Metode Perancangan

Tahap perancangan merupakan pengembangan dari gambaran umum sistem. Dalam tahap perancangan dijelaskan lebih detail tentang isi dari aplikasi yang dibuat yaitu dengan membuat *Flow chart* dan diagram *Unified Modelling Language* (UML) yang meliputi *use case*, *activity diagram* dan *sequence diagram* serta membuat desain *input* dan *output*. Setelah tahap perancangan selesai maka

dilakukan tahap implementasi yaitu menerjemahkan desain ke dalam *source code* berbasis web, yang akan digunakan untuk membuat aplikasi keuangan desa .

1.6 Sistematika Penulisan

1. BAB I PENDAHULUAN

Pada Bagian ini terdiri dari 5 bagian di atas, yaitu latar belakang meliputi penjelasan konseptual tentang apa dan mengapa judul riset tersebut menarik untuk diangkat dan digarap. Rumusan Masalah adalah pertanyaan hasil temuan permasalahan terikat penelitian yang sudah dilakukan. Batasan Masalah digunakan untuk menghindari adanya penyimpangan maupun pelebaran pokok masalah. Tujuan Penelitian dilakukan terkait mengapa riset tersebut dilakukan dan tujuan seperti apa yang ingin dicapai. Metodologi Penelitian adalah metode apa saja yang akan digunakan penulis.

2. BAB II TINJAUAN PUSTAKA

Pada bagian ini terdiri dari landasan teori yaitu penjelasan dalam bentuk literature review atau hasil penelitian yang telah dilakukan dan Dasar Teori yang digunakan untuk menjelaskan tentang definisi konseptual.

3. BAB III METODELOGI PENELITIAN

Pada bagian ini membahas mengenai komponen dari metode Penelitian kemudian mendeskripsikan secara spesifik.

4. BAB IV ANALISIS PERANCANGAN

Pada bagian ini membahas mengenai analisis dimulai dari instrument penelitian, analisis sistem, analisis kebutuhan dan analisis data. Kemudian perancangan aplikasi yang direncanakan untuk membangun aplikasi.

5. **BAB V IMPLEMENTASI DAN PENGUJIAN**

Pada bab ini dilakukan implementasi dan pengujian. Implementasi berisi uraian penggunaan sistem dari hasil pengembangan hasil penelitian yang merupakan jawaban tegas dan lugas atau rumusan masalah. Pengujian berisi tahapan pengujian yang dilakukan sesuai dengan aplikasi.

6. **BAB VI KESIMPULAN DAN SARAN**

Bagian ini membahas tentang kesimpulan yang berisi simpulan hasil penelitian yang merupakan jawaban dari rumusan masalah kemudian saran berisi usulan konkrit serta operasional yang merupakan tindak lanjut sumbangan penelitian.

BAB II

TINJAUAN PUSTAKA

2.1 Landasan Teori

Dalam penelitian ini akan digunakan tiga tinjauan studi yang nantinya mendukung penelitian yang akan dilakukan, dimana tinjauan studi yang di ambil sebagai berikut :

“ PERANCANGAN SISTEM E-VOTING UNTUK PEMILIHAN KETUA OSIS SMP NEGERI 10 PEKANBARU “ (Febrizal Alfarasy Syam, 2013)

SMP Negeri 10 Pekanbaru setiap tahunnya melakukan pemilihan Ketua OSIS. Proses pemilihan Ketua OSIS SMP Negeri 10 Pekanbaru dilakukan dengan cara voting oleh seluruh siswa secara langsung. Namun voting yang dilakukan masih secara konvensional yaitu pemilihan masih menggunakan kertas sebagai sarana untuk memilih menentukan pilihannya dan dalam perhitungan suara hasil pemilihan juga dilakukan secara manual, sehingga akan memerlukan biaya yang besar dan waktu yang cukup lama untuk mengetahui hasilnya. Pemanfaatan sistem e-voting, merupakan solusi yang diberikan penulis untuk menyelesaikan masalah tersebut. Pada tahap analisa dan perancangan penulis menggunakan metode pendekatan Object Oriented Analysis and Design (OOAD). Sedangkan pemodelan sistem menggunakan pendekatan UML (Unified Modelling Language). Hasil dari penelitian ini diharapkan dapat menyelesaikan masalah-masalah tentang proses pemilihan Ketua OSIS SMP Negeri 10 Pekanbaru sehingga dapat dilaksanakan secara efektif, efisien, cepat dan transparan.

“ PERANCANGAN SISTEM E-VOTING UNTUK PEMILIHAN KEPALA DAERAH (PILKADA) “ (Purwati, 2015)

Teknologi Informasi perlu diterapkan dalam kehidupan demokrasi di suatu Negara untuk mendukung pelaksanaan perhitungan suara dalam Pemilihan Umum (Pemilu) atau Pemilihan Kepala Daerah (Pilkada. Pelaksanaan Pemilihan Kepala Daerah yang dilaksanakan di berbagai daerah membutuhkan pula dukungan Sistem e-Voting. Sistem e-Voting dapat diberdayagunakan sejak pendataan calon pemilih, pendaftaran calon pemilih, pelaksanaan pemungutan suara sampai dengan perhitungan hasil pemungutan suara. Dalam tahap pendataan calon pemilih sistem ini dapat terhubung dengan Sistem Kependudukan Nasional untuk menjaring calon pemilih yang memenuhi syarat. Untuk tahapan pemberian suara pemilih cukup mendekatkan e-KTPnya ke perangkat. Perangkat akan mendeteksi apakah e-KTP tersebut telah terdaftar atau belum. Jika terdaftar pemilih disuruh untuk mendekatkan sidik jari. Jika benar maka pintu bilik suara akan terbuka dan pemilih dapat melakukan pilihannya dengan menekan tombol keypad kemudian enter untuk memilih.

“ SMART CONTRACT BLOCKCHAIN PADA E-VOTING “ (Teresa Enades Hari Setia, 2016)

Teknologi yang semakin berkembang di era globalisasi saat ini salah satunya yaitu e-voting. Hal yang sering dilakukan oleh masyarakat Indonesia dalam melakukan e-voting adalah memiliki ketua dalam suatu jabatan contoh paling besar yaitu presiden. Dengan adanya e-voting akan sangat mudah dilaukan. Tetapi selalu ada permasalahan yang terjadi pada hasil pemilihan kandidat yang bisa dimanipulasi oleh siapa saja. Sehingga dari permasalahan yang terjadi dilakukan penelitian terhadap hasil pemilihan untuk mengurangi kecurangan pada pemilihan. Oleh karena itu, penulis melakukan pengujian pengamanan hasil pemilihan, Dengan melakukan pengujian dengan bahasa pemrograman Solidity dan smart contract menghasilkan code unik pada setiap pemilihan baru. Sehingga manipulasi hasil pemilihan tidak akan bisa dilakukan karena setiap pemilih hanya memiliki satu

account dan satu address blockchain. Dari hasil pengujian penulis lakukan dapat disimpulkan bahwa smart contract blockchain bisa digunakan untuk membuktikan hasil pemilihan yang aman dan menciptakan e-voting yang bersifat jujur.

2.1. Dasar Teori

2.2.1. E-Voting

E-voting berasal dari kata electronic voting yang mengacu pada penggunaan teknologi informasi pada pelaksanaan pemungutan suara.

Pilihan teknologi yang digunakan dalam implementasi dari e-Voting sangat bervariasi, seperti penggunaan kartu pintar untuk otentikasi pemilih yang bisa digabung dalam e-KTP, penggunaan internet sebagai sistem pemungutan suara atau pengiriman data, penggunaan layar sentuh sebagai pengganti kartu suara, dan masih banyak variasi teknologi yang bisa digunakan dewasa ini. Dalam perkembangan pemikiran dewasa ini penggunaan perangkat telepon seluler untuk memberikan suara bisa menjadi pilihan karena sudah menggabungkan (konvergensi) perangkat komputer dan jaringan internet dalam satu perangkat tunggal.

Kondisi penerapan dan teknologi e-voting terus berubah seiring perkembangan teknologi informasi yang sangat cepat. Kendala-kendala e-voting yang pernah terjadi di berbagai negara yang pernah dan sedang menerapkannya menjadi penyempurnaan e-voting selanjutnya. Salah satu segi positif dari penerapan e-voting saat ini adalah makin mudahnya perangkat keras yang digunakan dan makin terbukanya perangkat lunak yang digunakan sehingga biaya pelaksanaan e-voting makin murah dari waktu ke waktu dan untuk perangkat lunak makin terbuka untuk diaudit secara bersama. Salah satu konsep penerapan perangkat lunak adalah melalui Indonesia Goes Open Source (IGOS) dengan diperkenalkannya aplikasi e-Demokrasi pada tahun 2007.

Penggunaan e-voting di Indonesia telah dilakukan dalam skala terbatas baik dalam lingkup organisasi, perusahaan maupun pemerintahan di skala paling kecil yaitu dusun atau desa.

Di Kabupaten Jembrana, Bali sejak pertengahan 2009 telah dilakukan puluhan kali pemilihan kepala dusun di desa-desa yang ada di kabupaten tersebut. Penggunaan e-voting di kabupaten Jembrana telah menghemat anggaran lebih dari 60 persen, seperti anggaran untuk kertas suara. E-voting ini juga diawali dengan penggunaan KTP (Kartu Tanda Penduduk) berbasis chip atau kemudian disebut juga e-KTP. Penggunaan e-KTP tersebut membuat pemilih tidak mungkin melakukan pemilihan lebih dari sekali. TPS (tempat pemungutan suara) juga bisa menampung hingga 1000 pemilih, sementara dengan sistem manual sekitar 500-700 pemilih saja per TPS yang layak.

Setelah Mahkamah Konstitusi memutuskan pada Selasa, 30 Maret 2010 bahwa penggunaan e-voting adalah konstitusional sepanjang tidak melanggar asas Pemilu yang luber dan jujur[2] maka e-Voting bisa dilakukan pada skala lebih luas di antaranya Pemilihan umum kepala daerah dan wakil kepala daerah (Pemilukada).

Kabupaten Jembrana, Bali sudah menyatakan kesiapannya untuk menyelenggarakan pemilihan bupati Jembrana pada bulan Oktober 2010 dengan e-voting. Namun berbagai kesiapan masih perlu dilakukan baik dari KPU maupun Bawaslu (Badan Pengawas Pemilihan Umum) dari sisi kesiapan SDM dan pemahaman mengenai e-voting itu sendiri. Juga harus dibuat perubahan Undang-Undang No.32 Tahun 2004 Tentang Pemerintahan Daerah[3] sehingga paling cepat e-voting baru akan dilaksanakan di daerah lain pada tahun 2011.[4] Namun khusus kabupaten Jembrana, Bali diharapkan bisa dilaksanakan menggunakan Peraturan KPU yang bisa diselesaikan sebelum Pemilukada di Jembrana dilaksanakan.

Terkait dengan Pemilu Nasional, CETRO juga pernah mengusulkan Pemilu Elektronik pada tahun 2014 nanti dan dilakukan persiapan sejak saat ini (Agustus 2009 ketika diusulkan) [5]. Keputusan MK tersebut memberi jalan untuk Pemilu Elektronik pada tahun 2014 yang harus diawali dengan selesainya Single Identity Number (SIN) untuk seluruh penduduk Indonesia yang direncanakan selesai pada tahun 2011.

Menurut Shalahuddin, (2009:33) E-voting adalah proses pemungutan suara yang memanfaatkan elektronik. Seiring dengan perkembangan zaman, sudah banyak penelitian pemanfaatan elektronik pada proses pemungutan suara yang menggantikan proses pemungutan suara secara manual, teknologi tersebut disebut e-voting.

Menurut (azhari, 2005) Electronic voting (E- voting) secara umum adalah pengguna teknologi komputer pada pelaksanaan voting. pilihan teknologi yang digunakan dalam implementasi dari E-voting sangat bervariasi, seperti penggunaan smartcard untuk autentikasi pemilih, penggunaan internet sebagai sistem pemungutan suara, penggunaan touch screen sebagai pengganti kartu suara dan masih banyak variasi yang digunakan

Skema E-voting adalah satu set protokol yang menjaga keamanan atau kerahasiaan pemilih dalam melakukan pemilihan serta interaksi dengan panitia pemilihan dan perhitungan suara. E-voting biasanya dibedakan menjadi dua tipe yaitu online dan off line (menggunakan mesin perhitungan suara atau kertas suara). Tujuan dari keamanan sistem e-voting adalah untuk menjamin privasi atau kerahasiaan pemilih dan keakuratan pilihan. Menurut (Canard, 2001). Keamanan sistem ini memiliki beberapa kriteria yaitu:

- 1) *Eligibility*. Hanya pemilih yang terdaftar yang dapat melakukan pemilihan.
- 2) *Unreusability*. Setiap pemilih hanya bisa memberikan satu kali pilihan.
- 3) *Anonymity*. Pilihan pemilih dirahasiakan
- 4) *Accuracy*. Pilihan tidak bisa diubah atau dihapus selama atau setelah pemilihan dan juga tidak bisa ditambahkan setelah pemilihan ditutup.
- 5) *Fairness*. Perhitungan suara sebelum pemilihan ditutup tidak bisa dilakukan.
- 6) *Vote and Go*. Pemilih hanya dapat melakukan pemilihan saja.

Public Verifiability. Setiap orang dapat melakukan pengecekan pada berjalannya proses pemilihan.

2.2.2. Pengertian Pemilu

Berdasarkan UU No. 15 Tahun 2011, Pemilihan Umum atau Pemilu adalah sarana pelaksanaan kedaulatan rakyat yang diselenggarakan secara langsung, umum, bebas, rahasia, jujur, dan adil dalam Negara Kesatuan Republik Indonesia berdasarkan Pancasila dan Undang- Undang Dasar Negara Republik Indonesia Tahun 1945.

Pemilihan Umum (Pemilu) merupakan salah satu media demokrasi yang digunakan untuk mewujudkan partisipasi rakyat. Pemilu dianggap penting dalam proses dinamika kehidupan berbangsa dan bernegara, Pemilihan umum sudah menjadi bagian yang tidak terpisahkan dari suatu negara demokrasi, jika kita melihat hampir seluruh negara demokrasi melaksanakan pemilihan umum.

Dalam negara hukum yang demokratis, kegiatan memilih orang atau sekelompok orang menjadi pemimpin idealnya dilakukan melalui pemilu dengan berasaskan prinsip pemilu yang langsung, umum, bebas, rahasia, jujur dan adil (LUBERDIL). Indonesia menjadikan pemilu sebagai bagian yang sangat penting dalam kegiatan bernegara, peraturan tertinggi mengenai pemilu diatur dalam Undang- Undang Dasar (UUD) 1945 hasil amandemen perubahan III, Bab VIIB tentang Pemilihan Umum, pasal 22.

Pelaksanaan seluruh proses pemilihan umum (Pemilu) di Indonesia melibatkan beberapa pihak yang saling berhubungan satu dengan yang lainnya. Pihak-pihak yang terkait dengan pelaksanaan pemilihan umum sesuai dengan Undang-Undang No 22 Tahun 2007 Tentang Penyelenggara Pemilihan Umum antara lain sebagai berikut:

Komisi Pemilihan Umum (KPU) merupakan lembaga penyelenggara Pemilu yang sifatnya nasional, tetap, dan mandiri.

KPU Provinsi dan KPU Kabupaten/Kota merupakan penyelenggara Pemilu ditingkat Provinsi dan Kabupaten/ Kota.

Panitia Pemilihan Kecamatan (PPK) merupakan panitia yang dibentuk oleh KPU Kabupaten/Kota, bertugas untuk menyelenggarakan Pemilu pada tingkat Kecamatan.

Panitia Pemungutan Suara (PPS) merupakan panitia yang dibentuk oleh KPU Kabupaten/Kota, bertugas untuk menyelenggarakan Pemilu di tingkat Desa/ Kelurahan.

Panitia Pemilihan Luar Negeri (PPLN) merupakan panitia yang dibentuk oleh KPU untuk menyelenggarakan seluruh proses Pemilu di luar negeri.

Kelompok Penyelenggara Pemungutan Suara (KPPS) adalah kelompok yang dibentuk oleh PPS untuk menyelenggarakan pemungutan suara di tempat pemungutan suara.

Kelompok Penyelenggara Pemungutan Suara Luar Negeri (KPPSLN) merupakan kelompok yang dibentuk oleh PPLN untuk menyelenggarakan pemungutan suara di tempat pemungutan suara di luar negeri.

Badan Pengawas Pemilu (Banwaslu) merupakan badan yang bertugas mengawasi penyelenggaraan Pemilu di seluruh Indonesia.

Panitia Pengawas Pemilu (Panwaslu) Provinsi dan Panwaslu Kabupaten/Kota merupakan panitia yang dibentuk oleh Banwaslu dan bertugas untuk mengawasi penyelenggaraan Pemilu di tingkat Provinsi dan Kabupaten/ Kota.

Panwaslu Kecamatan merupakan panitia yang dibentuk oleh Panwaslu Kabupaten/Kota untuk mengawasi penyelenggaraan Pemilu di tingkat Kecamatan.

Pengawas Pemilu Lapangan merupakan petugas yang dibentuk oleh Panwaslu Kecamatan, bertugas untuk mengawasi penyelenggaraan Pemilu di Desa/ Kelurahan.

Pemilih adalah warga negara Indonesia yang telah berusia sekurang-kurangnya 17 tahun atau telah/sudah pernah menikah dan tidak sedang dicabut hak pilihnya.

Peserta Pemilu difungsikan dalam beberapa waktu, diantaranya:

Pada pemilihan anggota DPR, DPRD tingkat 1, dan DPRD tingkat 2 peserta Pemilu adalah partai politik.

Pada Pemiluan anggota DPD, peserta Pemilu adalah perorangan.

Pada pemilihan presiden/wakil presiden, peserta Pemilu adalah wakil partai politik.

2.2.3. Pengertian pemilihan kepala desa

Pilkades merupakan salah satu bentuk pesta demokrasi yang begitu merakyat. Pemilu tingkat desa ini merupakan ajang kompetisi politik yang begitu mengena kalau dapat dimanfaatkan untuk pembelajaran politik bagi masyarakat. Pada moment ini, masyarakat yang akan menentukan siapa pemimpin desanya selama 6 tahun ke depan. Banyak bentuk pesta demokrasi yang telah digelar dalam kehidupan politik kita sekarang. Pilpres, Pilkada Gubernur, Pilkada Bupati dan Pemilu Legislatif. Tak ketinggalan adalah Pilkades. Begitu menarik bagi saya untuk mengkaji lebih dalam tentang budaya pemilihan kepala desa ini.

Dalam pelaksanaannya begitu mendetail keterkaitan antara pihak-pihak yang terkait dalam pelaksanaannya. Sehingga, perlu ketelitian dari tiap calon pemilih dalam menilai calon pemimpin yang akan dipilihnya tersebut. Namun pilkades terasa lebih spesifik dari pada pemilu-pemilu di atasnya. Yaitu adanya kedekatan dan keterkaitan secara langsung antara pemilih dan para calon. Sehingga, suhu politik di lokasi sering kali lebih terasa dari pada saat pemilu pemilu yang lain. Pengenalan atau sosialisasi terhadap calon-calon pemimpin bukan lagi mutlak harus lagi penting. Para bakal calon biasanya sudah banyak dikenal oleh setiap anggota masyarakat yang akan memilih. Namun demikian sosialisasi program atau visi misi sering kali tidak dijadikan sebagai media kampanye atau pendidikan politik yang baik. Kedekatan pribadi, akan sering kali banyak dipakai oleh masyarakat untuk menentukan pilihannya. Di sini unsur nepotisme masih begitu kental membudaya. Demikian juga dengan kolusi,

hubungan baik dalam berbagai posisi juga banyak dijadikan sebagai unsur penentuan hak pilih. Demikian juga dengan unsur Money politik yang sering dijadikan iming-iming dorongan dalam pemilihan. Hal demikian akan menjadikan para calon harus mengeluarkan biaya yang begitu besar. Persaingan antar calon sering kali juga terjadi dengan berlebihan. Kalau demikian ini yang terjadi usaha penghapusan KKN (Korupsi Kolusi dan Nepotisme) akan terasa sulit diwujudkan. Di sini pendidikan politik perlu dikembangkan. Kerelaan berkorban untuk kepentingan desa yang juga merupakan bagian dari bangsa dan negara ini tentu perlu diwujudkan. Tidak semua pengorbanan harus diukur dengan kontribusi uang. Kalau budaya money politik di tingkat desa bisa dikikis, tentu sedikit demi sedikit di tingkat yang lebih atas hingga pemilihan presiden akan dapat diwujudkan proses pemilihan pelaksana pemerintahan yang jujur dan adil.

Pilkades merupakan bagian dari proses kegiatan politik untuk memperkuat partisipasi masyarakat. Sehingga diharapkan akan terjadi perubahan yang signifikan di tingkat pedesaan. Semula kita hanya mengenal pesta demokrasi secara langsung berupa Pilkades ini. Sehingga pelaksanaannya banyak keluar dari etika dan norma politik. Money politic dengan berbagai bentuknya sulit sekali dihindari. Kemudian sejak era reformasi masyarakat dibudayakan dengan pemilihan pimpinan dengan cara pemilihan langsung. Dengan adanya pilkada diharapkan masyarakat dapat terlatih untuk peduli kepada pemimpinnya, serta sadar terhadap apa, siapa, dan bagaimana pemimpin yang akan di pilih nanti.

Hal yang menarik menjelang pilkades saat ini yakni adanya isu putra daerah. Akhir-akhir ini putra daerah dirasa menjadi salah satu syarat pendukung yang perlu dimiliki oleh seorang calon pemimpin desa. Sehingga tidak mengherankan jika seorang calon kepala desa menambahkan keterangan putra daerah pada setiap kampanyenya. Berdasarkan dari fenomena tersebut maka perlu pemikiran secara jernih tentang apa arti dari putra daerah itu sendiri. Apakah yang dimaksud dengan putra daerah adalah penduduk asli dari desa setempat.

Menurut Eep Saefullah Fatah dalam salah satu kolom politiknya, ada 4 jenis dari definisi putra daerah, yakni pertama, putra daerah geologis atau biologis, yaitu seseorang yang dilahirkan dari daerah tersebut. Kategori ini dibagi

menjadi, yakni seseorang yang dilahirkan di desa tersebut yang salah satu atau kedua orang tuanya berasal dari desa tersebut dan mereka yang tidak lahir di desa tersebut tapi memiliki orang tua yang berasal dari desa tersebut. Putra daerah ini terlintas hanya memiliki kepentingan pragmatis dengan daerah asalnya. Mereka menggunakan daerah hanya sebagai basis pemenuhan kepentingan politik dan ekonomi mereka sendiri. Namun sebaliknya daerah itu pun sedikit banyak memperoleh keuntungan politik dan ekonomi dari mereka. Keempat, yakni putra daerah sosiologis, yaitu mereka yang bukan saja memiliki keterkaitan genealogis dengan daerah tersebut tetapi juga hidup, tumbuh, dan besar serta berinteraksi dengan masyarakat daerah tersebut. Mereka menjadi bagian sosiologis dari daerah tersebut.

Dalam pemilihan pemimpin desa yang harus diutamakan ialah tentang kapabilitas dari calon-calon pemimpin tersebut. Suatu desa tidak hanya dapat dipimpin oleh pemimpin yang bermodalkan kefiguritan namun cacat secara intelektual, moral dan sosial. Pemimpin yang dibutuhkan oleh masyarakat sekarang yakni seseorang memiliki akseptabilitas namun ditunjang oleh moral yang baik, memiliki kemampuan yang cukup untuk memimpin dan membimbing masyarakatnya dan juga memiliki kemampuan dalam melaksanakan tugas-tugas administratif dan perpolitikan, serta memiliki wawasan yang luas dan pandangan yang luas terhadap perbaikan masyarakat.

2.2.4. Pengertian Aplikasi

Aplikasi adalah suatu subkelas dari suatu perangkat lunak komputer yang memanfaatkan kemampuan komputer secara langsung untuk melakukan suatu tugas yang diinginkan pengguna web. Aplikasi dapat juga dikatakan sebagai penerjemah perintah-perintah yang dijalankan pengguna komputer untuk diteruskan ke atau diproses oleh perangkat keras. Aplikasi merupakan program yang secara langsung dapat melakukan proses-proses yang digunakan dalam komputer oleh pengguna. Aplikasi merupakan kumpulan dari file-file tertentu

yang berisi kode program yang menghubungkan antara pengguna dan perangkat keras Komputer.

Aplikasi sering juga disebut sebagai perangkat lunak, merupakan program komputer yang isi instruksinya dapat diubah dengan mudah. Aplikasi pada umumnya digunakan untuk mengontrol perangkat keras (yang sering disebut sebagai device driver), melakukan proses perhitungan, dan berinteraksi dengan aplikasi yang lebih mendasar lainnya (seperti sistem operasi, dan bahasa pemrograman). Secara umum aplikasi dapat dibagi menjadi 3 tingkatan yaitu tingkatan program aplikasi (application program misalnya Microsoft Office), tingkatan sistem operasi (operating system misalnya Microsoft Windows), dan tingkatan bahasa pemrograman (misalnya PHP).

Beberapa aplikasi telah digabung menjadi suatu paket aplikasi dan sering disebut sebagai suite aplikasi (application suite). Contohnya adalah Microsoft Office dan OpenOffice.org, yang menggabungkan suatu aplikasi pengolah kata, lembar kerja, serta beberapa aplikasi lainnya. Aplikasi-aplikasi dalam suatu paket biasanya memiliki antarmuka pengguna yang memiliki kesamaan sehingga memudahkan pengguna untuk mempelajari dan menggunakan tiap aplikasi. Sering kali, mereka memiliki kemampuan untuk saling berinteraksi satu sama lain sehingga menguntungkan pengguna. Contohnya, suatu lembar kerja dapat dibenamkan dalam suatu dokumen pengolah kata walaupun dibuat pada aplikasi lembar kerja yang terpisah. Adapun definisi Aplikasi menurut para ahli:

Menurut Rachmad Hakim. S Aplikasi merupakan perangkat lunak yang digunakan untuk tujuan tertentu, seperti mengolah dokumen, mengatur Windows &, permainan (game), dan sebagainya. Beberapa aplikasi yang digabung bersama menjadi suatu paket disebut sebagai suatu paket atau application suite. Aplikasi-aplikasi dalam suatu paket biasanya memiliki antarmuka pengguna yang memiliki kesamaan sehingga memudahkan pengguna untuk mempelajari dan menggunakan tiap aplikasi.

Menurut Ali Zaki & Smitdev Community Aplikasi adalah komponen yang berguna melakukan pengolahan data maupun kegiatan-kegiatan seperti pembuatan

dokumen atau pengolahan data. Aplikasi adalah bagian PC yang berinteraksi langsung dengan user. Aplikasi berjalan di atas sistem operasi, sehingga agar aplikasi bisa diaktifkan, kita perlu melakukan instalasi sistem operasi terlebih dahulu.

Menurut R. Eko. I & Djokopran Aplikasi merupakan proses atau prosedur aliran data dalam infrastruktur teknologi informasi yang dapat dimanfaatkan oleh para pengambil keputusan yang sesuai dengan jenjang dan kebutuhan (relevan).

Menurut Hengky W. Pramana Aplikasi adalah satu unit perangkat lunak yang dibuat untuk melayani kebutuhan akan beberapa aktivitas seperti sistem perniagaan, game, pelayanan masyarakat, periklanan, atau semua proses yang hampir dilakukan manusia.

2.2.5. Website

Website adalah kumpulan halaman web yang saling terhubung dan file-filenya saling terkait. Web terdiri dari page atau halaman, dan kumpulan halaman yang dinamakan homepage. Homepage berada pada posisi teratas, dengan halaman-halaman terkait berada di bawahnya. Biasanya setiap halaman di bawah homepage disebut child page, yang berisi hyperlink ke halaman lain dalam web. Penemu website adalah Sir Timothy John “ Tim ” Berners – Lee. pada tahun 1991. Maksud dari Tim membuat website adalah untuk mempermudah tukar menukar dan memperbarui informasi kepada sesama peneliti di tempatnya bekerja.

Pada tanggal 30 April 1993, CERN (tempat dimana Tim bekerja) menginformasikan bahwa WWW dapat digunakan secara gratis oleh semua orang. Sebuah website bisa berupa hasil kerja dari perorangan atau individu, atau menunjukkan kepemilikan dari sebuah organisasi, perusahaan, dan biasanya website itu menunjukkan beberapa topik khusus dan kepentingan tertentu. Sebuah website dapat berisi hyperlink (pranala) yang menghubungkan ke website lain, jadi, terkadang perbedaan antar website yang dibuat oleh individu perseorangan

dengan website yang dibuat oleh organisasi bisnis bisa saja tidak terlalu terlihat. Website ditulis, atau secara dinamik di konversi menjadi HTML dan diakses melalui sebuah program software yang biasa disebut dengan web browser, yang dikenal juga dengan HTTP Client. Halaman web dapat dilihat atau diakses melalui jaringan komputer dan internet, perangkatnya dapat berupa komputer pribadi, laptop, PDA ataupun telepon selular. Sebuah website dibuat didalam sebuah sistem komputer yang dikenal dengan server web, atau yang disebut HTTP Server, dan pengertian ini dapat menunjuk pada software yang dipakai untuk menjalankan sistem ini, yang kemudian menerima lalu mengirimkan halaman-halaman yang diperlukan untuk merespon permintaan dari pengguna. Apache adalah piranti lunak yang biasa digunakan dalam sebuah webserver, kemudian setelah itu adalah Microsoft Internet Information Services (IIS).

Secara umum, Website atau Web adalah sekumpulan halaman yang terdiri dari beberapa laman dan berisi tentang informasi dalam bentuk digital baik itu tulisan (teks), gambar animasi yang disediakan melalui internet sehingga dapat diakses oleh banyak orang diseluruh dunia yang memiliki koneksi internet.

Maka disimpulkan, website adalah halaman web atau situs yang saling berhubungan oleh perorangan, kelompok atau organisasi. Sebuah situs web biasanya ditempatkan pada sebuah server web yang dapat diakses melalui jaringan seperti Internet atau jaringan area lokal (LAN) melalui alamat internet yang dikenal dengan URL. Kombinasi dari semua situs yang dapat diakses publik di internet juga dikenal dengan World Wide Web atau disingkat WWW. (gregorius, 2012)

2.2.6. Pengertian PHP

Menurut (EMS, 2012) PHP adalah bahasa pelengkap HTML yang memungkinkan dibuatnya aplikasi dinamis yang memungkinkan adanya pengolahan data dan pemrosesan data. Semua syntax yang diberikan akan sepenuhnya dijalankan pada server sedangkan yang dikirimkan ke browser hanya hasilnya saja. Kemudian merupakan bahasa berbentuk script yang ditempatkan

dalam server dan diproses di server. Hasilnya akan dikirimkan ke client, tempat pemakai menggunakan browser.

PHP dikenal sebagai sebuah bahasa scripting, yang menyatu dengan tag-tag HTML, dieksekusi di server, dan digunakan untuk membuat halaman web yang dinamis seperti halnya Active Server Pages (ASP) atau Java Server Pages (JSP). PHP merupakan sebuah software Open Source.

Menurut (kurniawan, 2010) PHP memiliki kelebihan dari bahasa pemrograman lain. Adapun kelebihan bahasa pemrograman PHP dari bahasa pemrograman lain adalah sebagai berikut:

1. Bahasa pemrograman PHP adalah sebuah bahasa script yang tidak melakukan sebuah kompilasi dalam penggunaannya.
2. Web Server yang mendukung PHP dapat ditemukan dimana-mana dari mulai apache, IIS, Lighttpd, hingga Xitami dengan konfigurasi yang relatif mudah. Dalam sisi pengembangan lebih mudah, karena banyaknya milis-milis dan developer yang siap membantu dalam pengembangan. Dalam sisi pemahaman, PHP adalah bahasa scripting yang paling mudah karena memiliki referensi yang banyak.

PHP adalah bahasa open source yang dapat digunakan di berbagai mesin (Linux, Unix, Macintosh, Windows) dan dapat dijalankan secara runtime melalui console serta juga dapat menjalankan perintah-perintah sistem.

Menurut (Nugroho, 2011) program PHP dapat diaktifkan dengan menggunakan paket PHP berbasis Open Source yaitu XAMPP. XAMPP merupakan paket PHP yang dikembangkan oleh komunitas Open Source. Xampp menyediakan program Apache, MySQL, PHP dan phpMyAdmin.

2.2.7. Pengertian MySQL

Menurut (Kurniawan, 2010) MySQL merupakan suatu jenis database server yang sangat terkenal. MySQL termasuk jenis RDBMS (Relational

Database Manajement System). MySQL mendukung bahasa pemrograman PH, bahasa permintaan yang terstruktur, karena pada penggunaannya SQL memiliki beberapa aturan yang telah distandarkan oleh asosiasi yang bernama ANSI. MySQL merupakan RDBMS (Relational Database Management System) server. RDBMS adalah program yang memungkinkan pengguna database untuk membuat, mengelola, dan menggunakan data pada suatu model relational. Dengan demikian, tabel- tabel yang ada pada database memiliki relasi antara satu tabel dengan tabel lainnya. Beberapa keunggulan dari MySQL yaitu:

1. Cepat, handal dan mudah dalam penggunaannya. MySQL lebih epat tiga sampai empat kali dari pada database server komersial yang beredar saat ini, mudah diatur dan tidak memerlukan seseorang yang ahli untuk mengatur administrasi pemasangan MySQL.
2. Didukung oleh berbagai bahasa Database Server MySQL dapat memberikan pesan Error dalam berbagai bahasa seperti Belanda, Portugis, Spanyol, Inggris, Perancis, Jerman, dan Italia.
3. Mampu membuat tabel berukuran sangat besar. Ukuran maksimal dari setiap tabel yang dapat dibuat dengan MySQL adalah 4 GB sampai dengan ukuran file yang dapat ditangani oleh sistem operasi yang dipakai.
4. Lebih murah MySQL bersifat open source dan didistribusikan dengan gratis tanpa biaya untuk UNIX platform, OS/2 dan Windows Platform.

Melekatnya integrasi PHP dengan MySQL. Keterikatan antara PHP dengan MySQL yang sama-sama SoftwareOpen-Source sangat kuat, sehingga koneksi yang terjadi lebih cepat jika dibandingkan dengan menggunakan database server lainnya. Modul MySQL di PHP telah dibuat Built-in sehingga tidak memerlukan konfigurasi tambahan pada File konfigurasi Php.

2.2.8. HTML

Hypertext Markup language (HTML) merupakan suatu bahasa markah yang digunakan untuk membuat halaman web, menampilkan berbagai informasi di dalam sebuah browser web. HTML dibuat dengan menggunakan aplikasi pengolah kata dan disimpan pada format ASCII sehingga menjadi halaman web dengan perintah-perintah HTML. HTML merupakan standar internet yang didefinisikan dan dikendalikan penggunaannya oleh World Wide Web Consortium (W3C). HTML dibuat oleh kolaborasi cillau TIM yang diketuai oleh Berners lee Robert ketika bekerja di CERN pada tahun 1989. Secara umum fungsi HTML adalah untuk mengelola serangkaian data dan informasi sehingga suatu dokumen dapat diakses dan ditampilkan di internet melalui layanan web.

Secara umum fungsi HTML adalah untuk mengelola serangkaian data dan informasi sehingga suatu dokumen dapat diakses dan ditampilkan di internet melalui layanan web.

Fungsi HTML yang lebih spesifik yaitu:

- a) Membuat halaman web
- b) Menampilkan berbagai informasi di dalam sebuah browser internet
- c) Membuat link menuju halaman web lain dengan kode tertentu (hypertext).

2.2.9. Pengertian CSS

CSS adalah bahasa Cascading Style Sheet dan biasanya digunakan untuk mengatur tampilan elemen yang tertulis dalam bahasa markup, seperti HTML. CSS berfungsi untuk memisahkan konten dari tampilan visualnya di situs. CSS dibuat dan dikembangkan oleh W3C (World Wide Web Consortium) pada tahun 1996 untuk alasan yang sederhana. Dulu HTML tidak dilengkapi dengan tags yang berfungsi untuk memformat halaman. Anda hanya perlu menulis markup untuk situs.

HTML dan CSS memiliki keterkaitan yang erat. Karena HTML adalah bahasa markup fondasi situs) dan CSS memperbaiki style (untuk semua aspek) yang terkait dengan tampilan website, maka kedua bahasa pemrograman ini harus berjalan beriringan.

2.2.10. Java Script

Java script adalah bahasa pemrograman tingkat tinggi dan dinamis. Java script populer di internet dan dapat bekerja di sebagian besar penjelajah web populer seperti Google Chrome, Internet Explorer, Netscape, dan Opera. Kode java script dapat disisipkan dalam halaman web menggunakan tag script. Bahasa pemrograman ini merupakan salah satu yang paling utama bagi web developer di antaranya:

1. HTML memungkinkan anda menambahkan konten kedalam web
2. CSS: menentukan layout, style, serta keselarasan website.
3. Java Script: menyempurnakan tampilan dan sistem halaman web.

Cara kerja java script adalah pengunjung situs akan diarahkan melalui file js yang terpisah atau embed langsung ke halaman website. Jadi bisa diartikan sebagai bahasa dari sisi klien. Nanti klien akan mengakses website lalu script tersebut akan terunduh di perangkat klien. Script tersebut juga di proses di perangkat yang di maksud. Jadi pemrosesannya bukan pada web server , tapi pada perangkat klien atau pengunjung website.

2.2.11. Database

Menurut (Fahlevi, 2013) Basis data adalah sekumpulan data yang saling berhubungan secara logis dan terorganisir dengan baik. Kumpulan data tersebut yang saling berhubungan yang disimpan secara bersama sedemikian rupa dan tanpa pengulangan (redundansi) yang tidak perlu, untuk memenuhi berbagai

kebutuhan. Kumpulan file/tabel/arsip yang saling berhubungan yang disimpan dalam media penyimpanan elektronis.

Proses memasukkan dan mengambil data ke dan dari media penyimpanan data memerlukan perangkat lunak yang disebut sistem manajemen basis data (Database Management System/ DBMS). DBMS merupakan sistem perangkat lunak yang memungkinkan user untuk memelihara, mengontrol dan mengakses data secara praktis dan efisien. Dengan kata lain semua akses ke basis data akan ditangani oleh DBMS. Beberapa fungsi yang harus ditangani DBMS yaitu mengolah pendefinisian data, dapat menangani permintaan pemakai untuk mengakses data, memeriksa sekuriti dan integrity data yang didefinisikan oleh DBA (Database Administrator), menangani kegagalan dalam pengaksesan data yang disebabkan oleh kerusakan sistem maupun disk. Dan menangani unjuk kerja semua fungsi secara efisien.

Tujuan utama DBMS adalah untuk memberikan tinjauan abstrak data kepada user (pengguna). Jadi sistem menyembunyikan informasi tentang bagaimana data disimpan, dipelihara, dan tetap dapat diambil (akses) secara efisien. Pertimbangan efisien disini adalah bagaimana merancang struktur data yang kompleks tetapi masih tetap bias digunakan oleh pengguna awam tanpa mengetahui kompleksitas strukturnya.

2.2.12. Diagram Alir Data (DAD)

Menurut (Ladjamuddin, 2006) DAD adalah aliran data yang masuk ke dalam dan keluar dari suatu proses harus sama dengan aliran data yang masuk ke dalam dan keluar dari rincian proses yang pada level atau tingkatan di bawahnya. DFD sering digunakan untuk menggambarkan suatu sistem yang telah ada atau sistem baru yang akan dikembangkan secara logika tanpa mempertimbangkan lingkungan fisik dimana data tersebut mengalir atau dimana data tersebut akan disimpan.

DFD merupakan alat yang digunakan pada metodologi pengembangan sistem terstruktur, yang terdiri dari context diagram dan DFD Levelled

Context Diagram

Berfungsi menggambarkan hubungan antara entitas luar, masukan dan keluaran sistem. Atau dengan kata lain untuk memetakan model lingkungan, yang direpresentasikan dengan lingkaran tunggal yang mewakili keseluruhan sistem.

Diagram Rinci (DFD levelled)

Menggambarkan sistem sebagai jaringan kerja antara fungsi yang berhubungan dengan aliran dan penyimpanan data, model ini hanya memodelkan sistem dari sudut pandang fungsi.

Sebagian besar para perancang sistem informasi dalam menggambarkan informasi dengan memanfaatkan arus data (DAD), Pemilihan metode perancangan sistem ini berdasarkan kepada prinsip:

Mudah dipahami oleh analisis maupun orang awam.

Menggambarkan level sistem tingkat tinggi, beserta batasan-batasan sistem dan informasi arus datanya.

Menggambarkan secara rinci sistem yang ada di setiap komponennya.

2.2.13. ERD (Entity Relationship Diagram)

ERD adalah model konseptual yang mendefinisikan hubungan antar penyimpanan (dalam DFD). ERD digunakan untuk memodelkan struktur data dan hubungan antar data, karena hal ini relatif kompleks. Dengan menggunakan ERD, maka dapat menguji model dengan mengabaikan proses yang harus dilakukan. Model Entity Relationship (ER) adalah model data konseptual (himpunan konsep yang mendefinisikan struktur basis data, transaksi pengambilan dan pembaruan basis data) tingkat tinggi untuk perancangan basis data. Penggambaran model ER secara sistematis dilakukan melalui diagram Entity Relationship.

2.2.14. XAMPP

XAMPP adalah sebuah paket perangkat lunak bebas yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program. Fungsinya adalah sebagai server yang berdiri sendiri, yang terdiri atas program Apache HTTP Server, MySQL database, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl. Dalam prakteknya, XAMPP bisa digunakan untuk menguji kinerja fitur ataupun menampilkan konten yang ada didalam website kepada oranglain tanpa harus terkoneksi dengan internet, cukup akses melalui xampp control panel, atau istilahnya website offline.

2.2.15. Bootstrap

Bootstrap adalah sebuah alat bantu untuk membuat sebuah tampilan halaman website yang dapat mempercepat pekerjaan seorang pengembang website ataupun pendesain halaman website. Sesuai namanya, website yang dibuat dengan alat bantu ini memiliki tampilan halaman yang sama tau mirip dengan tampilan halaman Twitter atau desainer juga mengubah tampilan halaman website sesuai dengan kebutuhan.

Berikut adalah beberapa pengertian bootstrap menurut para ahli:

Menurut Jubilee Enterprise penulis dari buku''Pemograman Bootstrap untuk Pemula'' yang telah mencoba mendesain tampilan website, mengatakan Bootstrap adalah sebuah alat bantu untuk membuat sebuah tampilan halaman website yang dapat mempercepat pekerjaan seorang pengembang website yang dapat mempercepat pekerjaan seorang pengembang website ataupun pendesain halaman website. Sesuai namanya website yang dibuat dengan alat bantu ini memiliki tampilan halaman yang sama/mirip dengan tampilan halaman Twitter atau desainer juga dapat mengubah tampilan halaman website sesuai dengan kebutuhan. (Enterprise, 2015)

Menurut Handoko seorang yang telah melakukan pengolahan sebuah database dengan XML pada jurnalnya mengatakan bahasa markup yang selama ini telah mengalami beberapa perubahan. Dari SGML (Standardized General Markup Language) yang merupakan bahasa markup pertama untuk halaman web, kemudian berubah menjadi HTML (Hyper Tekt Markup Language) yang mendukung multimedia dan beberapa scrpt. (Handoko, 2005).

2.2.16. Web Browser

Web Browser merupakan aplikasi yang bisa menjelajahi, menyajikan, maupun mengambil konten yang ada di berbagai sumber informasi pada jaringan internet atau WWW. Pengertian dari web browser juga sering disebut dengan suatu perangkat lunak dengan fungsi yang dimilikinya sebagai penerima, mengakses, penyaji berbagai informasi dari internet.

Adapun pengertian web browser menurut para ahli yaitu sebagai berikut:

Menurut Lia Kuwayatno, web browser adalah perangkat lunak yang fungsinya untuk membuka website dengan cara menuliskan alamat situs yang dituju pada kotak adres.

Menurut Smitdev Pahala web browser merupakan laman kata dari server yang tugasnya untuk menampilkan halaman website.

Menurut Ali zaki webbrowser adalah sebuah program yang dijalankan untuk mengakses website dengan komputer dan jaringan internet, guna menjadi sebuah sambungan yang bisa di operasikan.

Menurut Rachmad hakim web browser merupakan aplikasi untuk menjelajah di internet yang fungsinya guna menampilkan halaman web dengan dokumen yang disediakan oleh server.

Menurut Yuhevizar web browser adalah metode untuk menampilkan informasi di internet yang berupa gambar, teks, atau suara guna menghubungkan satu dokumen dengan dokumen yang lain.

Menurut Gustav Pahala Frans pengertian web browser adalah aplikasi yang digunakan untuk mengoperasikan internet sebagai media browsing, serta menjelajahi dunia maya.

2.2.17. Unified Modeling Language (UML)


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” (rossa, 2018)


Use Case Diagram

Usecase diagram merupakan pemodelan untuk melakukan (behavior) sistem informasi yang akan dibuat. Use case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat secara kasar yang digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu.

Simbol Simbol yang digunakan pada use case diagram bisa dilihat pada tabel 2.1.



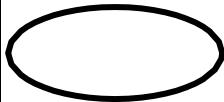

Table 2.1 Simbol Use Case Diagram

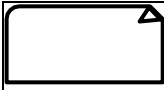
NO	Simbol	Nama	Keterangan
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan use case.

2		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor
3		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
4		<i>Extend</i>	Menspesifikasikan bahwa use case target memperluas perilaku dari use case sumber pada suatu titik yang diberikan
5		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
6		<i>Include</i>	Menspesifikasikan bahwa use case sumber secara

			eksplisit.
--	--	--	------------

Tabel 2. 2 Komponen Pada Use case

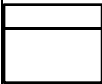
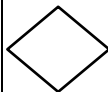

GAMBAR	NAMA	KETERANGAN
	<i>Association</i>	Yang menghubungkan antara objek satu dengan objek lainnya.
	<i>System</i>	Spesifikasi dari paket yang menampilkan sistem secara terbatas.
	<i>Use case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor.
	<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah elemen-elemennya (sinergi).
	<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan


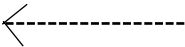
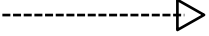

	mencerminkan suatu sumber daya komputasi.
---	---

Class diagram

Merupakan hubungan antar kelas dan penjelasan detail tiap-tiap kelas di dalam model desain dari suatu sistem, juga memperlihatkan aturan-aturan dan tanggung jawab entitas yang menentukan perilaku sistem. Class diagram juga menunjukkan atribut-atribut dan operasi-operasi dari sebuah kelas dan constraint yang berhubungan dengan objek yang dikoneksikan. Class diagram secara khas meliputi kelas (class), relasi associations, generalization dan aggregation, attribute, operasi (operation/method) dan visibility serta tingkat akses objek eksternal kepada suatu operasi atau atribut. Hubungan antar kelas mempunyai keterangan yang disebut dengan multiplicity atau cardinality. Simbol-simbol yang digunakan dalam Class diagram diantaranya adalah sebagai berikut :

Tabel 2. 3 Komponen Class diagram


No.	Gambar	Nama	Keterangan
1		Class	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama
2		Nary Association	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek
3		Generalization	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>)




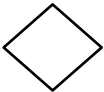
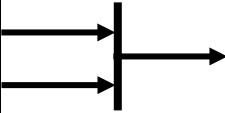
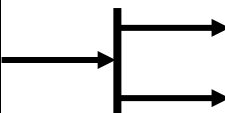
4		Collaboration	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor
5		Dependency	Hubungan perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) yang mempengaruhi elemen yang bergantung padanya
6		Realization	Operasi yang benar-benar dilakukan suatu objek
7		Association	Yang menghubungkan objek satu dengan objek lainnya

Activity diagram

Activity diagram menggambarkan workflow (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Simbol-simbol yang digunakan dalam Activity diagram diantaranya adalah sebagai berikut:

Tabel 2. 4 Komponen Activity diagram

No.	Gambar	Nama	Keterangan
1		<i>Start</i>	Digunakan sebagai awal dari sebuah proses

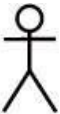
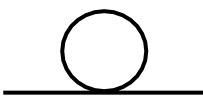
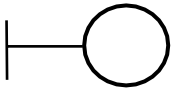


			
2		<i>End</i>	Dgunakan untuk menggambarkan akhir dari sebuah proses
3		<i>Activites</i>	Memperlihatkan bagaimana masing-masing antarmuka saling berinteraksi satu sama lain
4		<i>Decision</i>	Digunakan untuk menggambarkan suatu keputusan
5		<i>Join</i>	Penggabungan pada aliran proses <i>Activity</i> diagram
6		<i>Fork</i>	Percabangan dalam aliran <i>Activity</i> diagram


Sequence diagram

Sequence diagram menggambarkan perilaku setiap objek pada Use case dengan mendeskripsikan waktu hidup objek tersebut serta pesan yang dikirimkan

dan diterima antar objek. Banyaknya Sequence diagram yang harus digambar sebanyak Use case yang memiliki proses sendiri atau yang terpenting adalah semua Use case yang telah didefinisikan interaksi jalannya pesan tersebut sudah dicakup pada Sequence diagram sehingga semakin banyak Use case yang ada, maka Sequence diagram yang harus dibuat akan semakin banyak juga. Adapun simbol-simbol yang digunakan dalam perancangan Sequence diagram ini diantaranya adalah seperti pada tabel berikut ini:

Tabel 2. 5 Komponen Sequence diagram

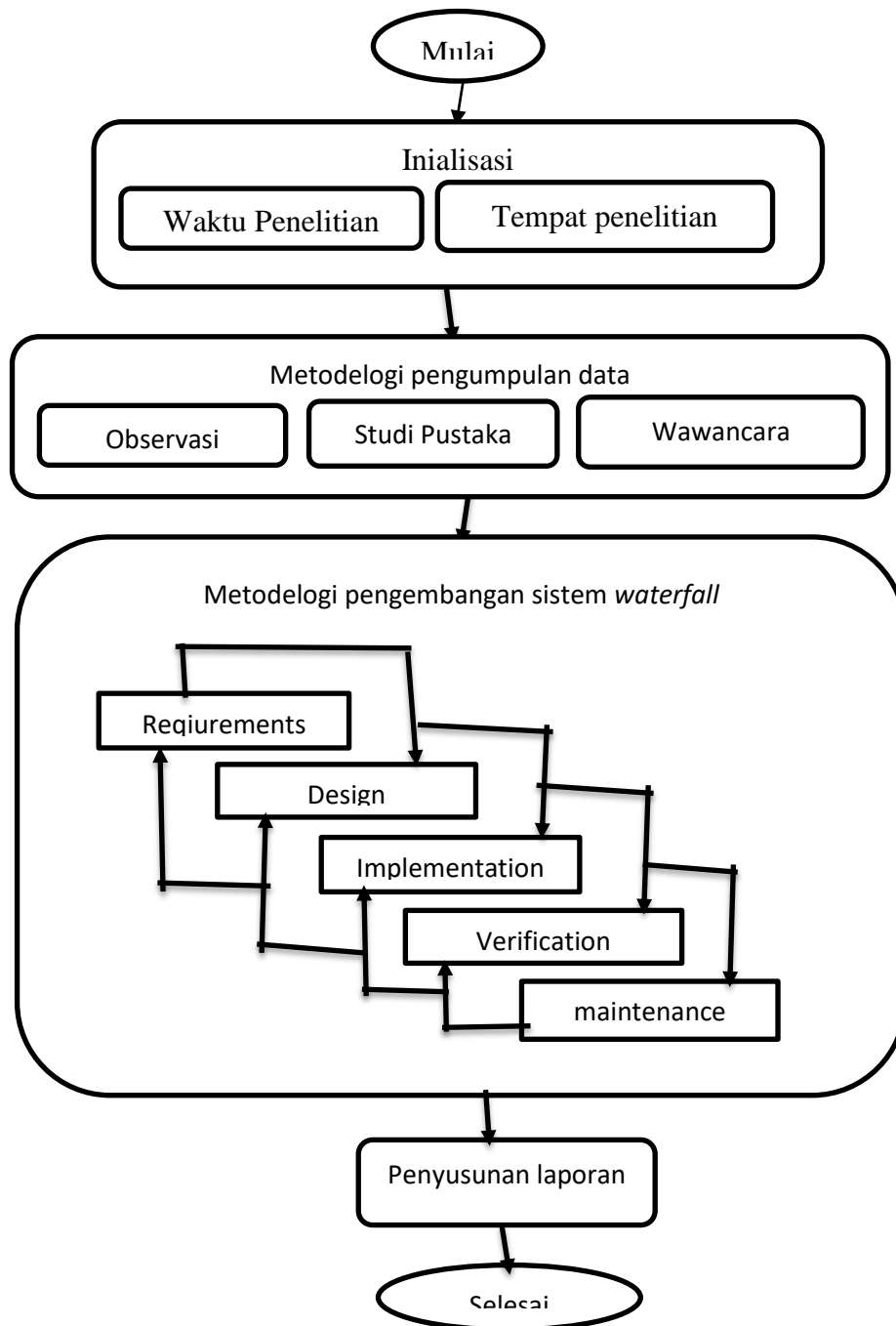
No.	Gambar	Nama	Keterangan
1		<i>Actor</i>	Menggambarkan orang yang berinteraksi dengan sistem
2		<i>Entity Class</i>	Menggambarkan hubungan yang akan dilakukan
3		<i>Boundary Class</i>	Gambaran dari <i>form</i>
4		<i>Control Class</i>	Menggambarkan hubungan antara <i>boundary</i> dan tabel
5		<i>A Focus of control & start of life line</i>	Menggambarkan tempat mulai dan berakhirnya <i>message</i>

6		<i>A Message</i>	Menggambarkan pengiriman pesan
---	---	------------------	-----------------------------------

BAB III

METODE PENELITIAN

3.1 Kerangka Pikir



Gambar 3.1 kerangka pikir

3.2 Deskripsi

Berlatang belakang dari kerangka pikir diatas berisikan tentang penjelasan mengenai langkah langkah atau tahapan yang dilakukan dalam Laporan penelitianini yaitu sebagai berikut:

3.2.1 Inialisasi

Pada tahap ini dilakukan penentu tempat dan waktu penelitian setelah waktu dan tempatr ditentukan selanjunya dilakukan proses pengumpulan data dan identifikasi masalah untuk dijadikan serta judul penelitian.

1. waktu penelitian

Penelitian mengenai pembuatan Aplikasi E-Voting Pemilihan Kepala Desa Berbasis Web Di Desa Baros Kecamatan Arjasari ini dilakukan selama 5 bulan dari mulai maret sampai juli 2021

2. Tempat Penelitian

Tempat penelitian dilaksanakan yaitu di Desa Baros Kecamatan Arjasari.

3.2.2 Pengumpulan data

Pada tahap ini penulis melakukan pengumpulan data dengan 3 tahap sebagai berikut:

1. Observasi

Metode ini melakukan Observasi dengan mendatangi secara langsung ke lokasi penelitian yang dilaksanakan di Kantor Desa Baros Arjasari. Hasil observasi ke kantor desa yang dilakukan, diperoleh informasi bahwa desa baros masih menggunakan cara manual untuk pemilihan kepala desa.

2. Wawancara

Metode ini dilakukan dengan cara melakukan tanya jawab secara langsung dengan kepala desa serta pihak-pihak yang berhubungan dengan judul Skripsi ini, hasil wawancara dapat dilihat pada lampiran.

3. Studi Pustaka

Mengumpulkan, membaca, dan mempelajari sumber-sumber tertulis penelitian terdahulu yang pernah dilakukan sebelumnya, seperti skripsi dan jurnal agar mendapat gambaran secara teoritis. Diantaranya adalah:

1. Oleh febrizal alfarasy syam, yuvi darmayunata, afriansyah afriansyah, (2013) melakukan penelitian berjudul “ Perancangan sistem e-voting untuk pemilihan ketua osis smp negeri 10 pekanbaru “
2. Oleh nani purwanti (2015) melakukan penelitian berjudul “ Perancangan sistem e-voting untuk pemilihan kepala daerah (pilkada) “
3. Oleh teresa enades hari setia, ajib susanto (2016) melakukan penelitian berjudul “ Smart contract blockchain pada e-voting “

3.2.4 Metodologi Pengembangan Sistem

Metodologi penelitian ini digunakan sebagai pedoman peneliti dalam pelaksanaan penelitian ini agar hasil yang dicapai tidak menyimpang dari tujuan yang telah ditentukan sebelumnya. Seperti yang telah diuraikan sebelumnya, penulis menggunakan metode *Waterfall* sebagai metodologi pengembangan sistem dalam penelitian ini, dengan Langkah-langkah sebagai berikut:

1. *Reqiurement*

Berikut adalah analisis kebutuhan yang diperlukan untuk menjawab permasalahan yang ada, yaitu :

Berikut ini adalah analisis kebutuhan yang di perlukan untuk menjawab permasalahan yang terdiri dari 2 yaitu:

1. **Kebutuhan Fungsional**

meliputi sistem yang berfungsi mengelola pemilihan, pengelolaan data, maka dari itu dibuat sistem yang mendukung aktifitas tersebut diantaranya:

- a) Pengelolaan data berisikan data informasi, data calon, data pemilih, data admin dan profil pemilihan tetap.
- b) Berisikan login admin, beranda, login pemilih
- c) Menghasilkan hasil sementara dan hasil keseluruhan.

2. **Kebutuhan non Fungsional**

Kebutuhan non fungsional meliputi Hardware dan Software pendukung untuk menjalankan sistem ini.

1. Hardware

Spesifikasi perangkat keras yang digunakan untuk membangun Sistem Informasi Repositori Skripsi tersebut adalah seperangkat komputer dengan spesifikasi sebagai berikut:

- a. Processor AMD Athlon TM X2 Dual-Core Processor QL-60 (1MB L2Cache, 1.9GHz)
- b. Hardisk berkapasitas 120 GB.
- c. Memory berkapasitas 2,00 GB.

2. Software

Perangkat lunak yang digunakan untuk rancang bangun aplikasi E voting pemilihan kepala desa tersebut adalah sebagai berikut:

- a. Sistem operasi Windows 7 Home Single Language 32-bit
- b. MySQL, XAMPP, Sublime Text
- c. Web browser.

2) **Desain**

Pada tahap ini Perancangan sistem e-voting ini menggunakan Unified Modelling Language (UML). UML Menyediakan beberapa diagram dalam proses perancangan sistem. Dalam sistem yang akan dibangun ini menggunakan beberapa diagram, yaitu: Use case

diagram, Activity diagram, Class diagram dan Sequence diagram. Adapun perancangan dari masing-masing diagram yang dimaksudkan tersebut seperti pada rincian berikut ini:

a) Use case Diagram digunakan untuk menggambarkan keterhubungan actor dan use case yang terlibat dalam penelitian ini adalah:

- Actor:

Panitia dan Pemilih

Use Case Diagram:

Panitia : Dapat menambah informasi tentang pilkades, mengelola data calon kepala desa, mengelola data pemilih, login,

pemilih : Dapat melakukan registrasi, melakukan login, dapat melihat data calon kepala desa, dapat memilih e-voting

b) Activity diagram

Digunakan untuk memodelkan alur kerja dari sistem dan aktivitas dari actor dalam sistem informasi. Adapun diagram alur yang akan dibuat diantaranya:

- Activity diagram registrasi warga
- Activity diagram e-voting
- Activity diagram Login Panitia
- Activity diagram Kelola Data Calon Kades
- Activity diagram Kelola Data Pemilih
- Activity diagram Melihat Hasil Penghitungan Suara

c) Sequence diagram

Diagram digunakan untuk menunjukan rangkaian pesan yang dikirim antara objek dengan objek lain. Berikut Sequence diagram yang akan dibuat:

- Sequence diagram registrasi warga
- Sequence diagram kelola data calon kades
- Sequence diagram kelola data pemilih
- Sequence diagram Melihat hasil e-voting

d) diagram class

- Mengelola data pemilih
- Mengelola dataAdmin
- Mengelola data login
- Mengelola data calon kades

e) Perancangan Sistem Interface

Pada tahap perancangan user interface ini merupakan sebuah tampilan pada Aplikasi yang akan dibangun dengan menggunakan Bootstrap. Maka diperoleh perancangan user interface sebagai berikut:

- Perancangan Menu Login
- Perancangan Menu Utama
- Perancangan Menu Data Pemilih
- Perancangan Menu Registrasi
- Perancangan Menu E-voting
- Perancangan Menu Kandidat Calon kades
- Perancangan Menu Hasil e-voting

3. Implementasi

Sebelum implementasi dilakukan maka diawali dengan melakukan pengujian terlebih dahulu terhadap sistem yang sudah dibuat, Pengujian dilakukan dengan cara mengamati input dan output dari Aplikasi e-voting pemilihan menggunakan teknik Black box yaitu pengujian yang dilakukan hanya mengamati hasil eksekusi melalui data uji dan memeriksa fungsional dari perangkat lunak, Kriteria yang diukur dalam tahapan ini yaitu:

- a. Login, data informasi, data calon, profil calon, serta data user. Menghasilkan database yang mampu menyimpan, mengelola data dan informasi data e-voting.

4. Verivication

Pada tahap ini penulis melakukan pengintegrasian system dan testing terhadap aplikasi yang telah di buat. Sistem akan di verifikasi untuk diuji sejauhmana kelayakannya.

5. Maintenance

Pada tahap ini, software yang sudah jadi akan di jalankan atau di operasikan oleh penggunaanya.

BAB IV

ANALISIS DAN PERANCANGAN

4.1 Analisis

4.1.1 Analisis Masalah

E-voting pemilihan kepala desa yang sedang berjalan di Desa Baros saat ini masih menggunakan cara konvensional, baik dalam pemungutan suara maupun perhitungannya, perhitungan masih dicatat secara manual dipapan hasil voting. Pemungutan yang masih manual tentu saja kurang efisien, banyak kendala dan masalah yang ditemui dalam pemilihan suara, salah satunya kertas pemilihannya yang robek ataupun kurang jelas, kemudian dalam hal perhitungan suara yang masih manual sehingga terjadikesalahan dalam perhitungan suara itu sendiri.

4.1.2 Analisis Software

Pada pembuatan Aplikasi *e-voting* pemilihan kepala desa berbasis web ini, penulis menggunakan beberapa perangkat lunak atau *software* dengan spesifikasi sebagai berikut :

Tabel 4.1 Analisis Software

NO	Software	Fungsi
1	Windows 7 Ultimate 32-bit	Sistem Operasi
2	SublimeText	Teks Editor
3	Xampp (PHP)	Untuk membuat Aplikasi

4	Google Chrome	Web Browser
5	Balsamiq Mockup	Untuk membuat rancangan User Interface
6	Umlet	Untuk membuat UML diagram
7	Microsoft Word 2010	Untuk membuat laporan

4.1.3. Analisis Pengguna

Aplikasi yang dibuat akan membatasi layanan-layanan yang disediakan berdasarkan hak akses yang diberikan kepada pengguna dalam sesuai keperluannya. Secara umum, pengguna admin memiliki hak untuk, *login*, melihat dan mengubah atau menambahkan data calon kepala desa, memilih, mengubah informasi tentang *e-voting* dan *logout*. Sedangkan untuk pemilih memiliki hak untuk *login*, melihat data calon kepala desa, melihat informasi, dapat memilih calon kepala desa yang diinginkan dan *logout*. Berdasarkan hak akses, pengguna yang akan menggunakan aplikasi *e-voting* pemilihan kepala desa berbasis web ini dibagi menjadi 2(dua) pengguna yaitu :

a. Administrator

Administrator adalah pengelola *e-voting* yang diberikan kewenangan untuk melakukan pengelolaan aplikasi. Admin dapat melihat dan mengubah semua data sistem, dan dapat membenahi kesalahan yang terjadi dalam sistem. Semua layanan yang ada pada sistem dapat digunakan oleh admin dengan tujuan agar sistem bisa dimonitor atau dipantau secara penuh oleh admin untuk perbaikan-perbaikan jika terjadi kesalahan pada data ataupun informasi pada *e-voting* tersebut. Adapun hal-hal yang dapat dilakukan seorang Admin dalam sistem adalah sebagai berikut:

1. Login
2. Menambahkan dan mengubah data calon kepala desa
3. Menambahkan dan mengubah data pemilih
4. Menambahkan data Administrator
5. Mengubah informasi
6. Melihat hasil vote

b. Pemilih

Pemilih yang dimaksud adalah masyarakat atau warga desa baros. Adapun hal-hal yang dapat dilakukan seorang pemilih dalam sistem adalah sebagai berikut:

1. Login
2. Melihat informasi
3. Melihat data calon pemilih
4. Melihat tata cara pemilihan
5. Dapat memvote calon kepala desa
6. Dapat kode tanda terima pemilihan

4.1.4. User Interface

Pada dasarnya, fokus dari Interaksi Manusia dan Komputer adalah perancangan dan evaluasi antarmuka pemakai (user interface) pada suatu sistem. User Interface dapat diartikan sebagai bagian sistem komputer yang memungkinkan manusia berinteraksi dengan komputer secara langsung. User interface dari aplikasi *e-voting* pemilihan kepala desa berbasis web ini akan dibuat *user friendly*, yaitu dengan membuat tampilan yang sederhana, sehingga aplikasi dapat dengan mudah dipelajari, mudah dioperasikan dan tetap dapat memberikan informasi mengenai *e-voting* pemilihan kepala desa kepada pemilih.

4.1.5. Fitur-fitur

Fitur-fitur yang akan di buat dalam aplikasi ini akan memudahkan admin atau pengelola *e-voting* dalam melakukan seluruh pengelolaan data dan memudahkan pemilih dalam mendapatkan informasi mengenai *e-voting* Berikut ini yaitu fitur-fitur yang terdapat pada aplikasi tersebut:

1. Pendaftaran ke admin , dapat dilakukan oleh pemilih untuk bisa dan mengakses informasi detail mengenai informasi *e-voting* dan dapat memvote calon kepala desa.
2. Fitur tentang, pemilih dapat melihat informasi tentang pengertian *e-voting*
3. Data cara pemilihan, dapat dilihat informasi tata cara pemilihan oleh pemilih.
4. Data calon kepala desa, yang dapat digunakan pemilih untuk melakukan pemilihan suara pada calon yang ingin di pilih.
5. Fitur kode tanda terima pemilihan, dapat di cek untuk melihat berapa hasil jumlah suara yang telah didapatkan calon kepala desa.
6. logout
7. Pengelolaan data oleh admin, untuk menambahkan, menghapus, atau mengedit data-data calon kepala desa , pemilih dan administrator.
8. Pengelolaan data hasil vote, admin bisa melihat jumlah hasil voting yang sudah dilakukan.

4.1.6. Analisis Data

Berikut ini merupakan analisis data berupa data masukan (*input*), proses, dan keluaran (*Output*) untuk penunjang bagi aplikasi *e-voting* pemilihan kepala desa yang akan dibuat :

Tabel 4.2 Tabel Analisis Data

No	Proses	Masukan	Keluaran
1	Login pemilih	<i>User name, password</i>	Informasi Login
2	Home page	Menu home , calon kepala desa (voting), tata cara pemilihan, kode tanda terima pemilihan	Informasi tentang e-voting , menu page
3	Data calon kepala desa	Nama calon kepala desa, foto calon ,kolom vote , submit	Informasi data calon kepala desa
4	Tata cara pemilihan	Nama pemilih, ID/E KTP pemilih, login, validasi ID/KTP , cara memilih, kode tanda terima pemilihan	Informasi tata cara pemilihan
5	Logout	<i>Username,password</i>	Informasi awal login

4.2. Perancangan

Perancangan Aplikasi pengelolaan *e-voting* pemilihan kepala desa dibuat dengan menggunakan UML (*unified modelling language*) sebagai Bahasa pemodelan. Diagram UML yang digunakan adalah *Usecase Diagram*, *Activity Diagram*, *Sequence Diagram*, dan *Class Diagram*.

4.2.1 Use case Diagram

Use Case Diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Sebuah *use case* merepresentasikan sebuah interaksi antara aktor dengan sistem.

1. Aktor

Seorang/sebuah aktor adalah sebuah entitas manusia atau mesin yang berinteraksi dengan sistem untuk melakukan pekerjaan- pekerjaan tertentu Berikut ini merupakan deskripsi setiap actor pada *usecase diagram* tersebut :

Tabel 4.3 Deskripsi Aktor pada Usecase Diagram

No	aktor	Deskripsi
1	Admin	1. Melakukan Login 2. Melihat tampilan dashboard 3. Menambah, menghapus , mengedit data Administrator 4. Menambah, menghapus, mengedit data calon kepala desa 5. Menambah, menghapus, mengedit data pemilih 6. Melihat hasil vote

2	Pemilih	1. Melakukan pendaftaran 2. Melakukan Login 3. Melihat tampilan home page 4. Melihat informasi tentang <i>e-voting</i> 5. Melihat informasi tata cara pemilihan 6. Melihat informasi data calon kepala desa 7. Melakukan voting calon kepala desa 8. Mendapatkan kodetanda terima pemilihan 9. Dapat melakukan pencarian kode tanda terima pemilihan tersebut
---	---------	---

2. Deskripsi *Usecase Diagram*

Berikut merupakan deskripsi dari *usecase diagram* yang menjelaskan mengenai proses-proses yang dilakukan dalam aplikasi:

Tabel 4.4 Deskripsi *Usecase diagram*

No	Aktor	Deskripsi
1	Login	aktor diharuskan untuk melakukan login terlebih dahulu. Yaitu dengan memasukan <i>Username</i> dan <i>password</i> .
2	Pendaftaran	Pendaftaran dilakukan oleh aktor (pemilih) untuk mendapatkan <i>username</i> dan <i>password</i> yang dapat digunakan untuk login
3	<i>Home page</i>	Setelah login, pemilih akan masuk ke tampilan home page
4	Data calon kepala desa	Pemilih dapat melihat data calon kepala desa dan memvote calon yang dipilih
5	Tata cara pemilihan	Pada menu ini, pemilih bisa memainkan beberapa tata cara pemilihan tersebut sebelum melakukan pemilihan calon, untuk memastikan data yang sudah terdaftar,
6	Kode tanda terima pemilihan	Setelah melakukan pemilihan pemilih akan mendapatkan kode yang bisa dicek untuk melihat hasil jumlah voting pemilihan

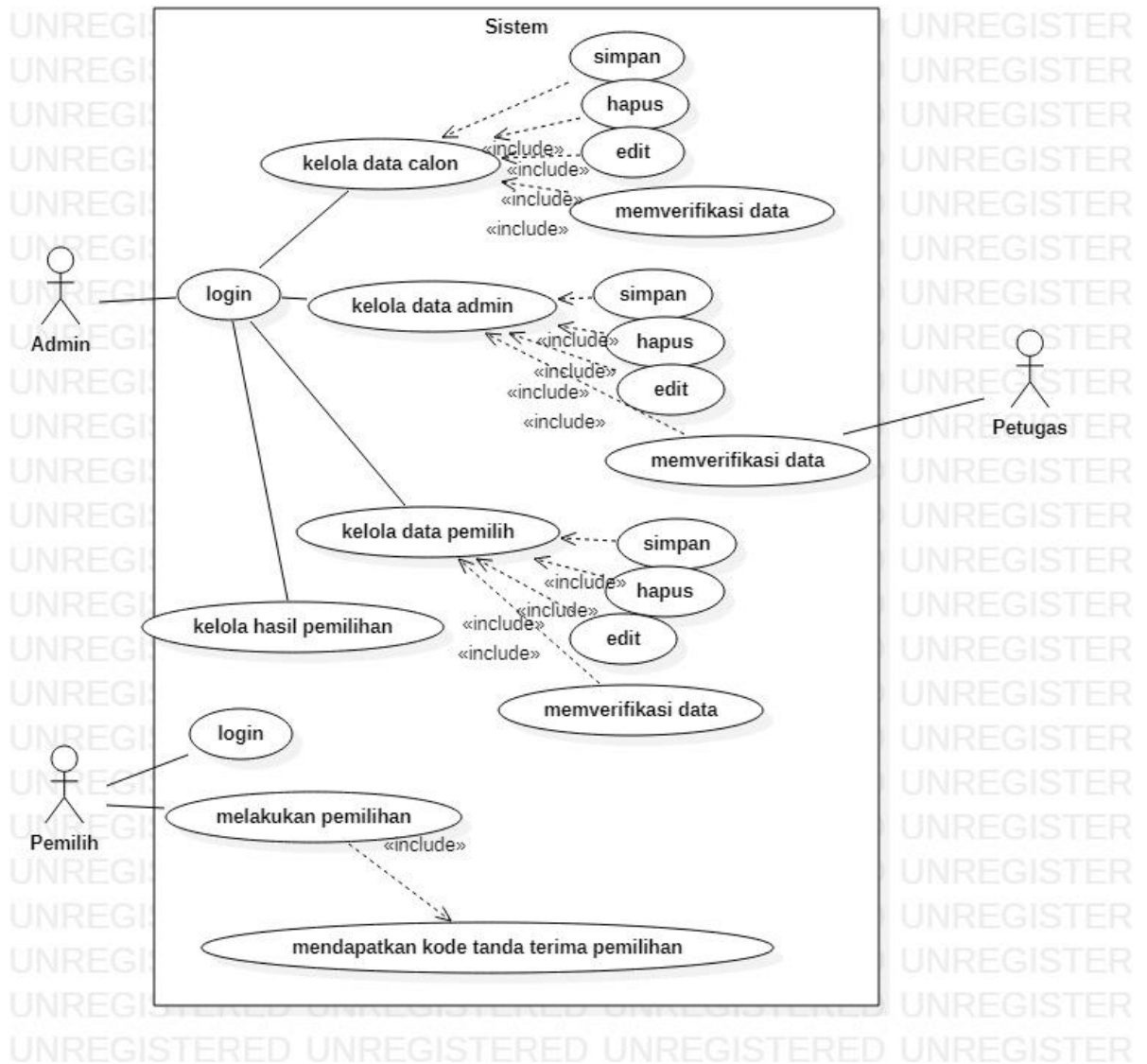
7	<i>Dashboard</i>	Setelah admin melakukan login tampilan <i>dashboard</i> akan muncul dengan menampilkan informasi mengenai <i>e-voting</i> , dan jumlah hasil voting yang sudah dilakukan
8	Admin	Proses pengelolaan data admin, yaitu dapat menambahkan, menghapus dan mengedit data Administrator, data calon kepala desa, data calon pemilih dan informasi tentang <i>e-voting</i> .

9	Laporan	Menampilkan hasil jumlah voting yang sudah dilakukan
---	---------	--

3. Usecase aplikasi

Berikut ini merupakan usecase aplikasi *e-voting* pemilihan kepala desa di desabaros:

a. Usecase Diagram aplikasi



Gambar 4.1 Usecase Diagram aplikasi

Berikut adalah skenario Usecase diagram admin, yang pertama adalah skenario dari usecase diagram login untuk admin :

Tabel 4.5 Usecase diagram login admin

Nama	Login admin	
Level	1	
Aktor	Admin	
Tujuan	Masuk kedalam aplikasi untuk mengelola semua data dalam aplikasi	
<i>Precondition</i>	Memasukan <i>username</i> dan <i>password</i>	
<i>Postcondition</i>	Berhasil memasuki <i>Dashboard</i>	
Langkah	User	Sistem
	1.Memasukan <i>username</i> dan <i>password</i>	
		2.Melakukan Validasi Username dan password
		3. Menampilkan Halaman Dashboard

Kemudian setelah login admin akan masuk ke tampil

an dashboard yang menampilkan menu lainnya, lalu admin dapat melihat data-data dihalaman admin, calon kepala desa, pemilih, informasi *e-voting* dan laporan hasil voting masuk. Berikut ini skenario usecase diagram admin untuk melihat data:

Tabel 4.6 Skenario usecase diagram dashboard admin

Nama	Dashboard	
Level	1	
Aktor	Admin	
Tujuan	Masuk ke dashboard untuk mengelola data-data yang terdapat pada tampilan dashboard	
<i>Precondition</i>	Masuk ke menu admin, calon kepala desa, pemilih, informasi <i>e-voting</i> dan laporan hasil voting masuk	
<i>Postcondition</i>	Melihat admin, calon kepala desa, pemilih, informasi <i>e-voting</i> dan laporan hasil voting masuk	
	User	Sistem
	1. Masuk ke menu	

Langkah	Admin	
		2.Menampilkan halaman admin
	3. Masuk ke menu calon kepala desa	
		4.Menampilkan halaman kepala desa
	5,Masuk ke menu Pemilih	
		6.Menampilkan halaman data pemilih
	7.Masuk ke menu Hasil	
		8.Menampilkan Halaman jumlah hasil voting yang masuk

Selain admin dapat melihat informasi mengenai admin, admin, calon kepaladesa, pemilih, informasi *e-voting* dan laporan hasil voting masuk . Admin juga dapat menambahkan data. Berikut usecase aplikasi admin untuk menambah data :Tabel 4.7 Sekenario usecase diagram tambah data admin

Nama	Dashboard	
Level	1	
Aktor	Admin	
Tujuan	Masuk ke dashboard untuk menambah data- data yang terdapat pada tampilan dashboard	
<i>Precondition</i>	Masuk ke menu admin, admin, calon kepala desa, pemilih. Lalu pilih tombol tambah	
<i>Postcondition</i>	Menambah data admin, calon kepala desa dan pemilih.	
Langkah	User	Sistem
	1.Masuk ke menu admin, tambah data admin baru	

		2.Menampilkan halaman admin, data admin ditambahkan
	3.Masuk ke menu kepala desa, tambah data calon kepa desa yang baru	
		4.Menampilkan halaman kepala desa , data kepala desa Ditambahkan

	5.Masuk ke menu pilih, tambahkan data pilih baru	
		6.Menampilkan halaman pilih, data pilih baru Ditambahkan

selanjutnya admin dapat mengedit informasi mengenai admin, calon kepala desa, pemilih, informasi *e-voting*. Berikut usecase aplikasi admin untuk mengedit data :

Tabel 4.8 Sekenario usecase diagram edit data admin

Nama	Dashboard	
Level	1	
Aktor	Admin	
Tujuan	Masuk ke dashboard untuk mengedit data-data yang terdapat pada tampilan dashboard	
<i>Precondition</i>	Masuk ke menu admin, calon kepala desa, pemilih, informasi <i>e-voting</i> dan laporan hasil voting masuk Lalu pilih tombol edit	
<i>Postcondition</i>	Mengedit data admin, calon kepala desa, pemilih, informasi <i>e-voting</i> .	
	User	Sistem

	1.Masuk ke menu admin, edit data admin	
		2.Menampilkan halaman admin, data admin akan

Langkah		berubah
	3.Masuk ke menu kepala desa edit data kepala desa	

		4.Menampilkan halaman kepala desa, data kepala desa akan Berubah
	5.Masuk ke menu pilih, edit data pilih	
		6.Menampilkan halamanpilih , data pilih akan berubah

Kemudian admin dapat menghapus informasi mengenai admin, kepala desa, pilih . Berikut usecase aplikasi admin untuk menghapus data

Tabel 4.9 Skenario usecase diagram hapus data admin

Nama	Dashboard
------	-----------

Level	1
Aktor	Admin

Tujuan	Masuk ke dashboard untuk menghapus data- data yang terdapat pada tampilan dashboard	
<i>Precondition</i>	Masuk ke menu admin, kepala desa, pemilih. Lalu pilih tombol hapus	
<i>Postcondition</i>	Menghapus data admin, kepala desa, pemilih.	
	User	Sistem
	1.Masuk ke menu admin, pilih hapus data admin	
		2.Menampilkan halaman admin, data admin akan dihapus
	3.Masuk ke menu kepala desa, pilih hapus data kepala desa	

Langkah		4.Menampilkan halaman kepala desa, data kepala desa akan dihapus
	5.Masuk ke menu pemilih, hapus data p pemilih	
		6.Menampilkan halaman pemilih, data pemilih akan Dihapus

Berikut adalah skenario dari usecase diagram pemilih, yang pertama adalah usecase diagram login pemilih :

Tabel 4.10 skenario usecase diagram login pemilih

Nama	Login pemilih
Level	2
Aktor	Pemilih
Tujuan	Masuk kedalam aplikasi untuk melihat informasi home page
<i>Precondition</i>	Memasukan <i>username</i> dan <i>password</i>
<i>Postcondition</i>	Berhasil memasuki <i>home</i>

Langkah	User	Sistem
---------	------	--------

	1.Memasukan <i>username</i> dan <i>password</i>	
		2.Melakukan Validasi Username dan password
		3.Menampilkan Halaman <i>home page</i>

Jika pemilih atau masyarakat belum mempunyai akun untuk login, maka pemilih harus konfirmasi ke admin, setelah konfirmasi ke admin, admin akan memberikan password yang sudah disiapkan, pemilih dapat login ke aplikasi untuk melihat dan melakukan voting. Berikut adalah skenario usecase diagram setelah login untuk melihat informasi tentang *e-voting*:

Tabel 4.11 skenario usecase diagram
home page

Nama	Home	
Level	2	
Aktor	Pemilih	
Tujuan	Masuk ke menu home untk menilhat informasi tentang e-voting, tatacara pemilihan dan data calon kepla desa.	
<i>Precondition</i>	Pilih menu home	
<i>Postcondition</i>	Melihat halaman home page dan membaca informasi pemilihan	
Langkah	User	Sistem
	1.masuk ke menu Home	
		2.Menampilkan halaman menu home
	3.pilih tombol menu kepala desa	

		<p>4.Masuk ke halaman</p> <p>Data calon kepala desa, dan bisa langsung memvote calon kepla desa yang akan di pilih</p>
	5.pilih tombol menu tata cara pemilihan	
		6.Menampilkan

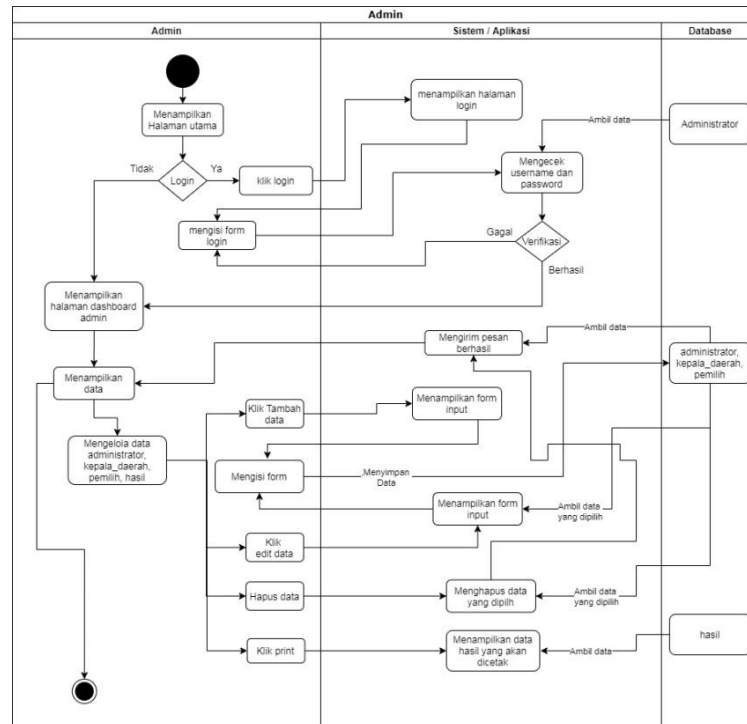
		Informasi tata cara pemilihan
	7.konfirmasi Hasil vote yang sudah di pilih	
		8.menampilkan halaman kode tanda terima pemilihan

4.2.2 Activity Diagram

Activity Diagram menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alur berawal,

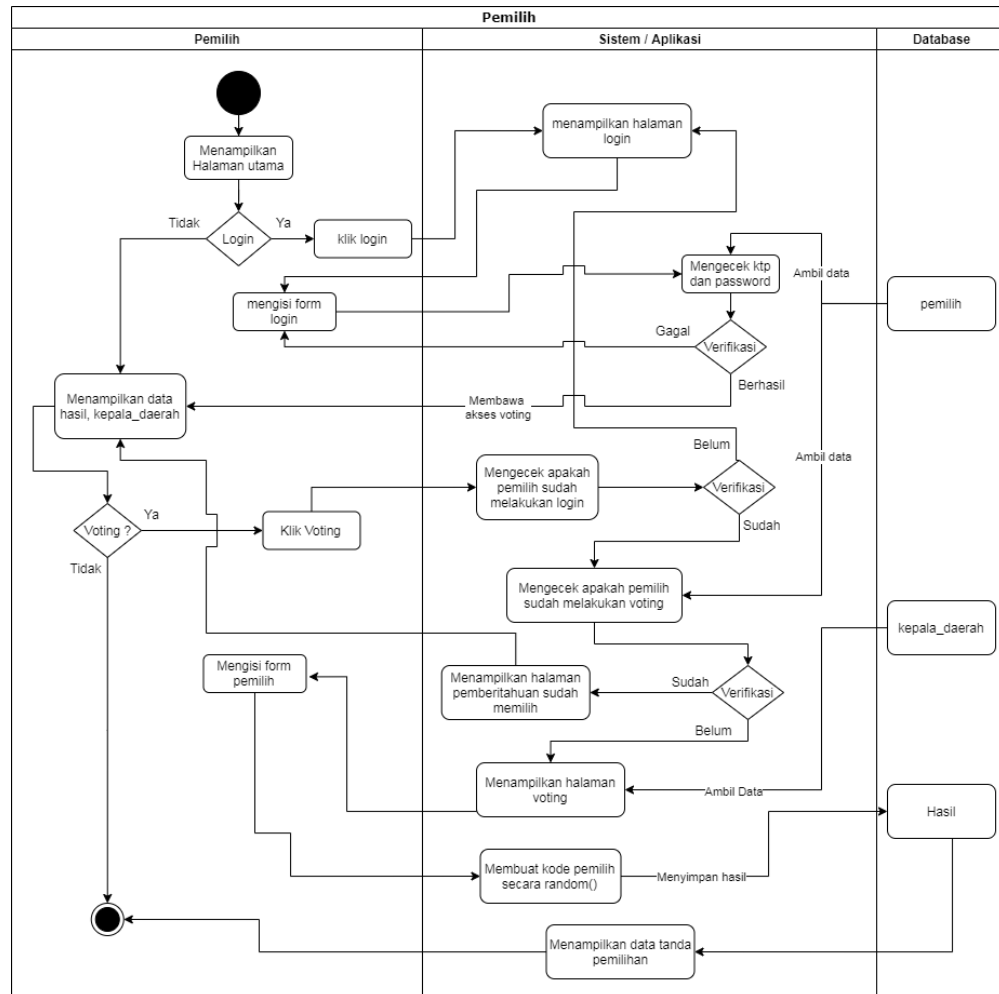
decision yang mungkin terjadi, dan bagaimana mereka berakhir. Berikutini merupakan *Activity Diagram* dari aplikasi e-voting pemilihan kepala desa berbasis web di desa baroskecamatan arjasari:

1. Activity diagram Admin



Gambar 4.2 *Activity diagram* admin

2. Activity diagram pemilih

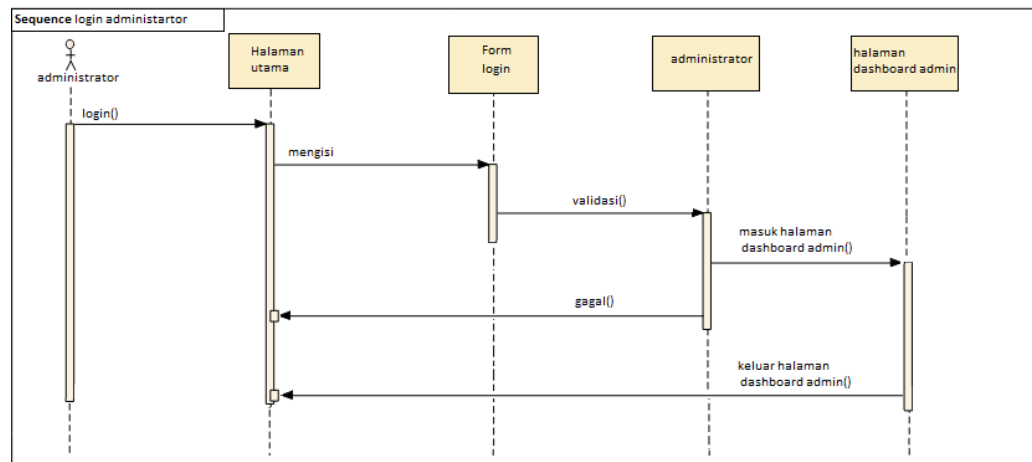


Gambar 4.3 Activity diagram pemilih

4.2.3 Sequence Diagram

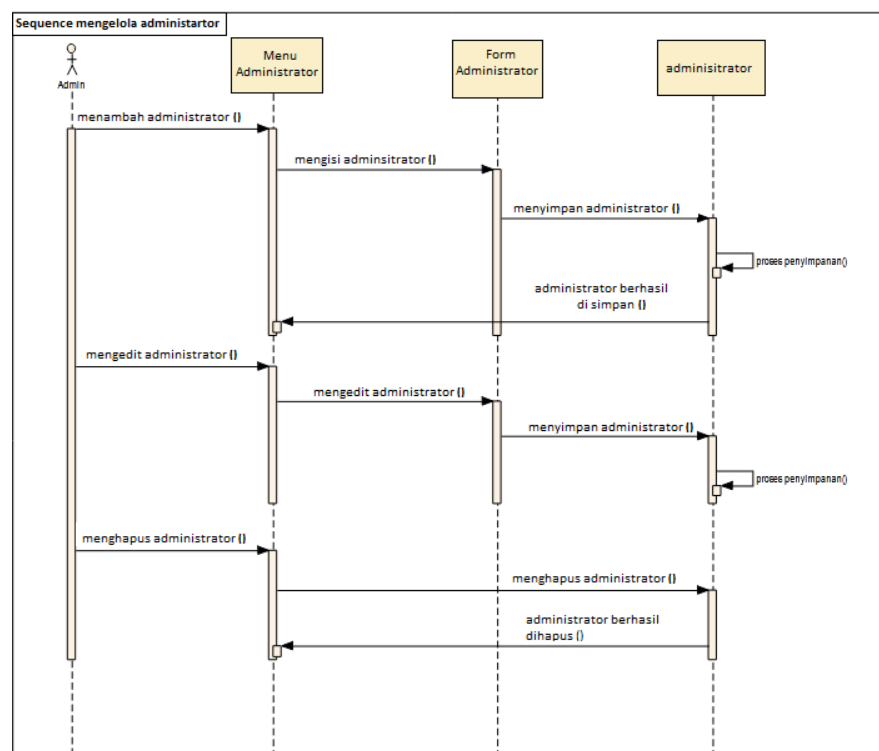
Berikut ini adalah sequence diagram aplikasi e-voting pemilihan kepala desa di desa baros:

1. Sequence diagram login admin



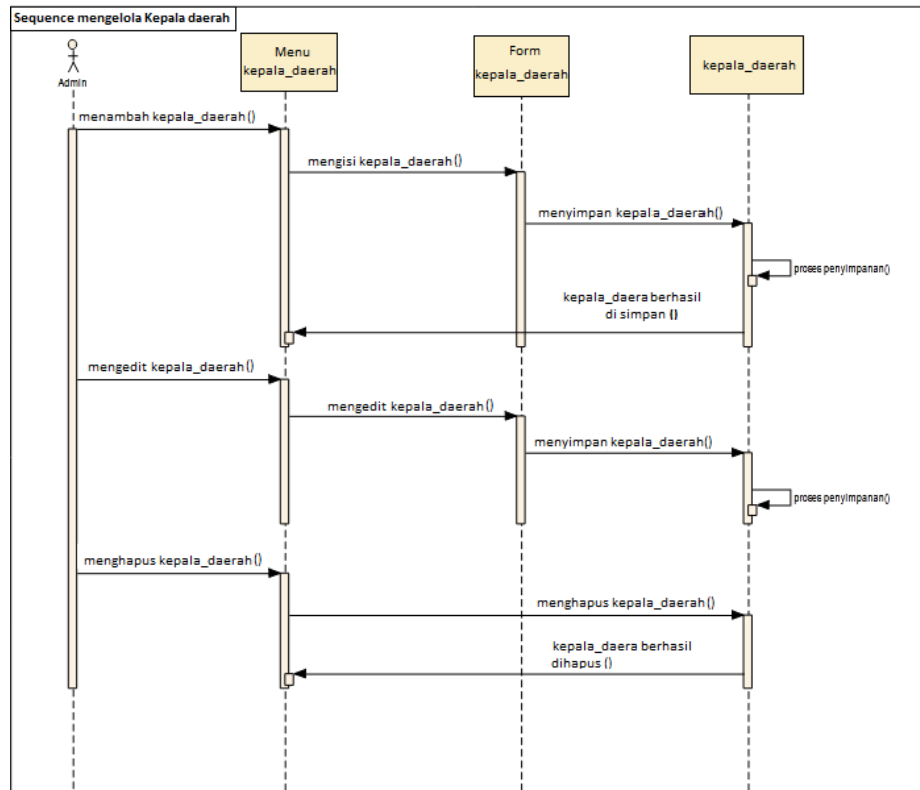
Gambar 4.4 sequence diagram login admin

2. Sequence diagram data admin



Gambar 4.5 sequence diagram data admin

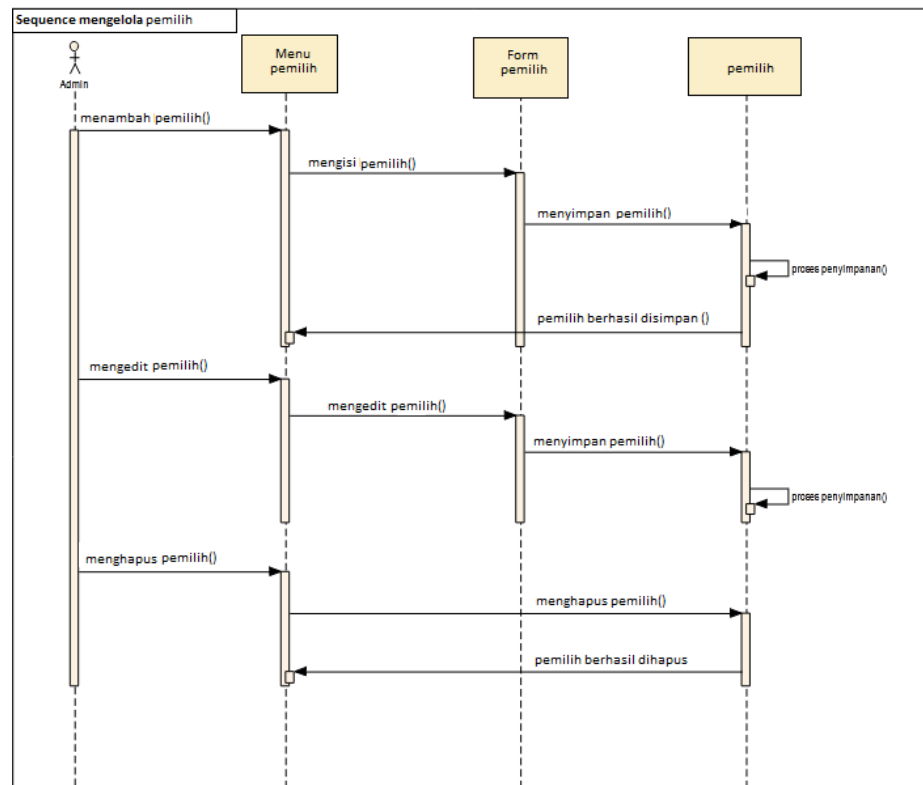
3. Sequence diagram kepala desa



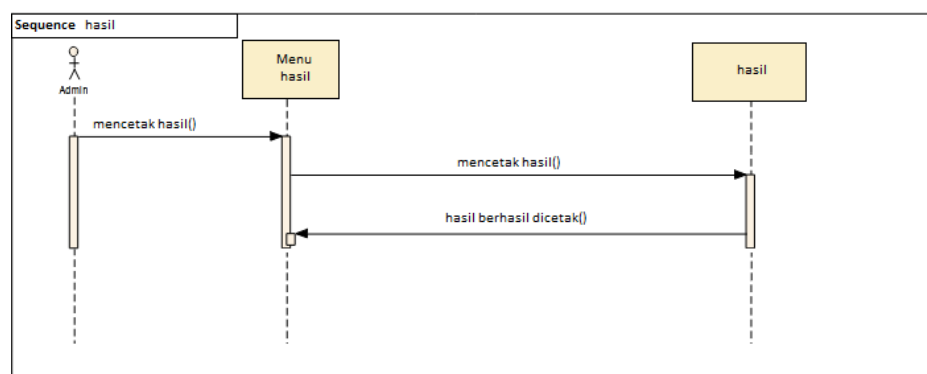
Gambar 4.6 sequence diagram kepala desa

4. Sequence diagram data pemilih

Gambar 4.7 sequence diagram data pemilih

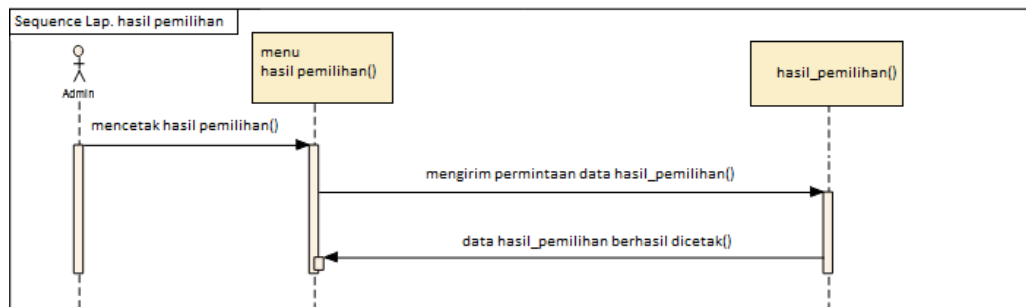


5. Sequence diagram hail



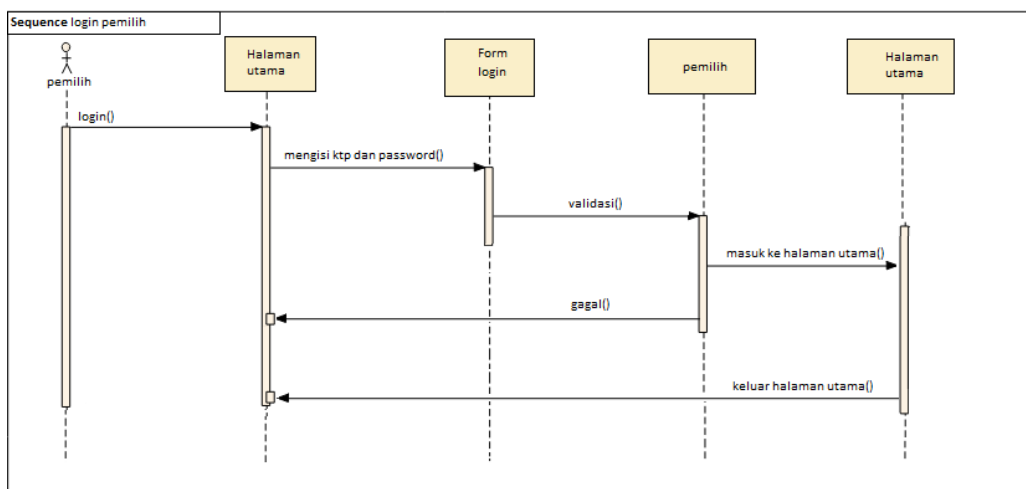
Gambar 4.8 sequence diagram hasil

6. Sequence diagram laporan



Gambar 4.9 sequence diagram laporan

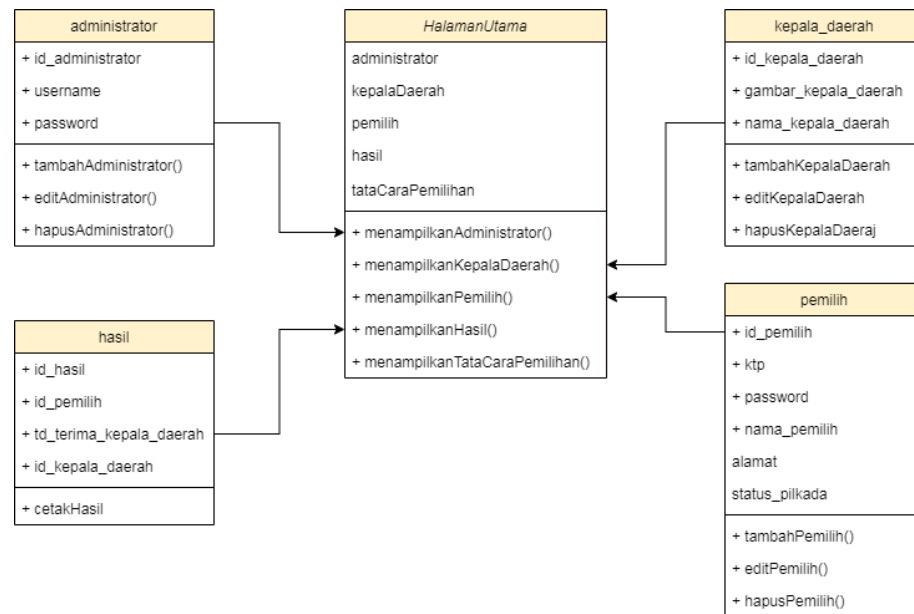
7. Sequence diagram login pemilih



Gambar 4.10 sequence diagram login
pemilih

4.2.4 Class Diagram

Diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Berikut merupakan class diagram Aplikasi-e-voting pemilihan kepala desa berbasis web didesa baros kecamatan Arjasari:



Gambar 4.11 class diagram Aplikasi e-voting pemilihan kepala desa

4.2.3.1. Perancangan Tabel

Berikut ini dijelaskan tabel- tabel yang digunakan dalam perancangantabel dengan deskripsi sebagai berikut :

1. Tabel admin

Nama Tabel : Admin

Keterangan : Menampung data Admin

Tabel 4.12 Tabel Administrator

NO	Nama field	Tipe Data	Keterangan
1	id_administrtor	Int (12)	Nomor identitas untuk pengurutan otomatis
2	Username	Varchar (255)	Nama administrator
3	Password	Varchar (255)	Password administrator

2. Tabel kepala desa

Nama Tabel : kepala desa

Keterangan :data calon

kepala desa

Tabel 4.13 Tabel kepala desa

NO	Nama field	Tipe Data	Keterangan
1	id_kepala_desa	Int (12)	Nomor identitas untuk pengurutan otomatis
2	Nama_kepala_des	Varchar (255)	Nama kamar

	A		
3	Gambar_kepala_d esa		Gambar calon kepala desa

3. Tabel pemilih

Nama Tabel : pemilih

Keterangan : daftar pemilih

Tabel 4.14 Tabel pemilih

NO	Nama field	Tipe Data	Keterangan
1	id_pemilih	Int (12)	Nomor identitas untuk pengurutan otomatis
2	Ktp	Int(12)	Nomor identitas User
3	Password	Varchar(255)	Password pemilih
4	Nama_pemilih	Int(12)	Nama pemilih
5	Alamat	Int(12)	Alamat pemilih
6	Status_pilkades	Tinyint(1)	Status pemilihan

4. Tabel hasil

Nama Tabel : id_hasil

Keterangan : melihat hasil voting masuk

Tabel 4.15 Tabel hasil

Nama field	Tipe Data	Keterangan
Id_hasil	Int (11)	Nomor identitas untuk pengurutan otomatis
Id_pemilih	Int(12)	Nomor identitas user
Id_terima_kepala desa	Int(12)	Nomor kode terima pemilihan kepala desa
Id_kepala_desa	Int(12)	Nomor identitas kepala desa

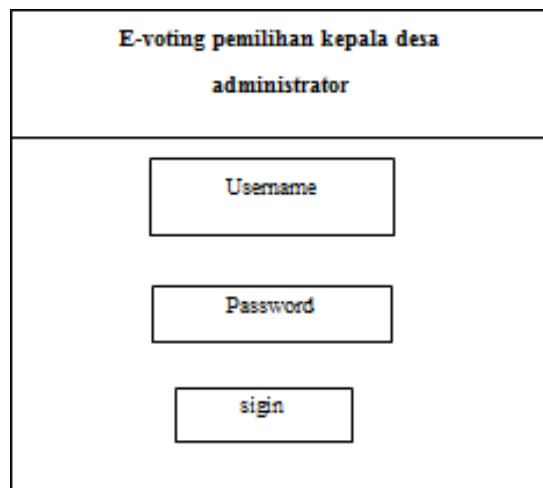
4.2.5 User Interface

User interface merupakan tampilan yang dibuat agar memudahkan komunikasi antara pengguna dengan sistem aplikasi. Perancangan *user interface* sendiri dibuat sesederhana mungkin untuk memudahkan pengguna dalam mengakses aplikasi sehingga dapat mengurangi tingkat kebingungan pengguna dalam menggunakan aplikasi.

4.2.5.1 User interface admin

➤ User Interface Login admin

Halaman login admin memuat form untuk mengisi user name dan password agar admin dapat masuk ke dashboard untuk mengelola data informasi e voting.



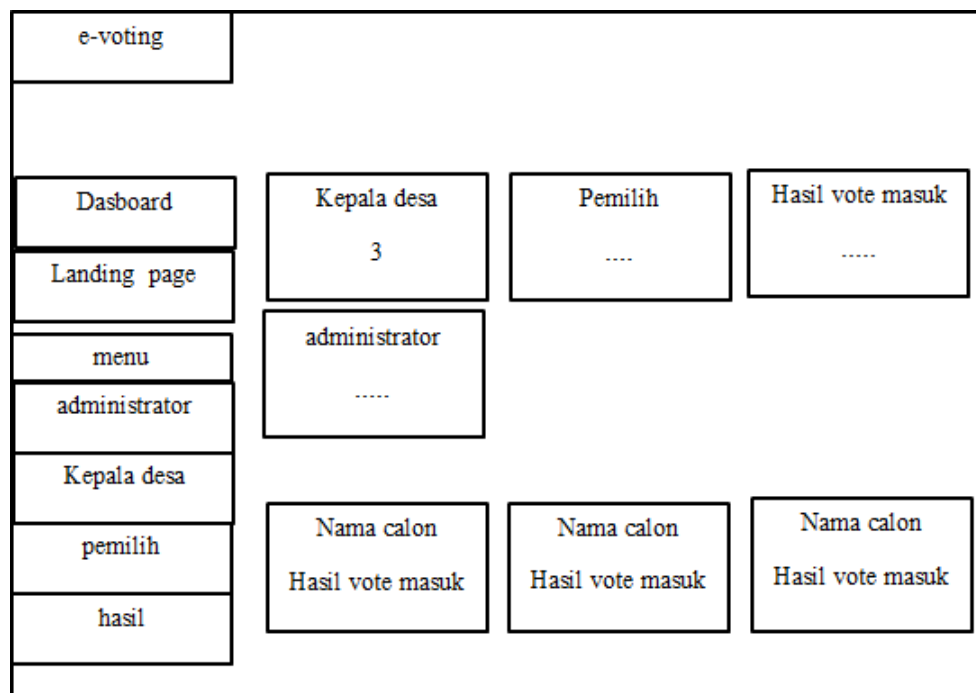
The image shows a login form for an administrator. It is titled "E-voting pemilihan kepala desa administrator". The form contains three input fields: "Username", "Password", and a "sign" button.

E-voting pemilihan kepala desa administrator
<input type="text" value="Username"/>
<input type="password" value="Password"/>
<input type="button" value="sign"/>

Gambar 4.12 User Interface Login admin

➤ User Interface Dashboard admin

Setelah admin berhasil melakukan login, admin akan masuk ke halaman dashboard yang memuat informasi mengenai data administrator, data kepala desa, data pemilih, data hasil vote masuk, dan beberapa nama calon kepala desa.



Gambar 4.13 User Interface Dashboard admin.

➤ User Interface data kepala desa admin

Setelah di menu dashboard admin dapat memilih menu kepala desa, dihalaman menu kepala desa memuat informasi nama kepala desa, gambar. Admin juga dapat mencari, menambah, mengedit dan menghapus data kepala desa.

Daftar kepala desa				
Tambah data			Search for....	
no	gambar	Nama kepala desa	action	
1			edit	hapus

Gambar 4.14 User interface menu
daftar kepala desa

➤ User Interface data pemilih admin

Setelah di menu dashboard admin selain menu kepala desa ada jugamenu pemilih, dihalaman menu pemilih memuat informasi ktp,nama pemilih, alamat,dan status. Admin juga dapat mencari, menambah, mengedit dan menghapus data pemilih.

Daftar pemilih						
Tambah data				Search for....		
no	ktp	Nama pemilih	alamat	status	action	
1					edit	hapus

Gambar 4.15 user interface data pemilih
adminUser Interface hasil pemiihan

User interface hasil pemilihan memuat informasi mengenai hasil pemilihan, yaitu id pemilih,tanda terima kepala desa. Dan admin juga dapat melakukan cetak laporan hasil pemilihan .

Daftar tabel hasil pemilihan			
cetak		Search for....	
no	Id pemilih	Tanda terims kepala desa	Id kepala desa
1			

Gambar 4.16 user interface hasil pemilihan

4.2.5.2 User Interface pemilih

➤ User Interface login pemilih

Halaman login pemilih memuat form untuk mengisi user name dan password agar admin dapat masuk ke home page untuk melihat informasi dan melakukan pemilihan.

The diagram shows a login form for 'E-voting pemilihan kepala desa'. The form is enclosed in a rectangular border. At the top, the title 'E-voting pemilihan kepala desa' is centered, with 'pemilih' centered below it. In the center of the form, there are three input fields stacked vertically: 'Nik KTP', 'Password', and 'signin'.

E-voting pemilihan kepala desa pemilih
Nik KTP
Password
signin

Gambar 4.17 user interface login pemilih

➤ User Interface home page

User interface home page memuat informasi mengenai informasi tentang *e-voting* ,data calon kepala desa, tata cara pemilihan .Pemilih dapat langsung memulai untuk memvote calon kepala desa yang akan dipilih.

E-voting		Home	tentang	Calon kepala desa	Tata cara pemilihan	logout
----------	--	------	---------	-------------------	---------------------	--------

E voting pemilihan kepala desa

Mulai voting sekarang

Gambar 4.18 user interface home page

➤ User Interface halaman tentang

User Interface halaman tentang ini menginformasikan pengertian *e-voting*

E-voting		Home	tentang	Calon kepala desa	Tata cara pemilihan	logout
----------	--	------	---------	-------------------	---------------------	--------

Pengertian e voting

Gambar 4.19 User Interface halaman tentang *e-voting*

➤ User Interface halaman data calon kepala desa

User interface halaman data calon kepala desa memuat data gambar dan nama calon kepala desa, pemilih juga bisa langsung vote salah satu calon kepala desadan klik menu submit untuk mengirim hasil vote.

E-voting	Home	tentang	Calon kepala desa	Tata cara pemilihan	logout									
<table border="1"> <tr> <td>Gambar dan nama calon Kepala desa</td> <td>Gambar dan nama calon Kepala desa</td> <td>Gambar dan nama calon Kepala desa</td> </tr> <tr> <td><input type="radio"/></td> <td><input type="radio"/></td> <td><input type="radio"/></td> </tr> <tr> <td colspan="3"> <input type="button" value="submit"/> </td> </tr> </table>						Gambar dan nama calon Kepala desa	Gambar dan nama calon Kepala desa	Gambar dan nama calon Kepala desa	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="button" value="submit"/>		
Gambar dan nama calon Kepala desa	Gambar dan nama calon Kepala desa	Gambar dan nama calon Kepala desa												
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>												
<input type="button" value="submit"/>														

Gambar 4.20 User interface data calon kepala desa

➤ User Interface tatacara pemilihan

User interface halaman tata cara pemilihan ini berisi beberapa tatacara pemilihan yang harus dipastikan. Yang berisikan nama pemilih, ID/E KTP pemilih, login / masuk, validasi ID/E KTP pemilih, cara memilih, kode tanda terima pemilihan

E-voting		Home	tentang	Calon kepala desa	Tata cara pemilihan	logout						
		Tata cara pemilihan										
<table border="1"> <tr> <td>Nama pemilih</td> </tr> <tr> <td>ID / E KTP pemilih</td> </tr> <tr> <td>Login / masuk</td> </tr> <tr> <td>Validasi ID / E KTP Pemilih</td> </tr> <tr> <td>Cara memilih</td> </tr> <tr> <td>Kode tanda terima pemilihan</td> </tr> </table>							Nama pemilih	ID / E KTP pemilih	Login / masuk	Validasi ID / E KTP Pemilih	Cara memilih	Kode tanda terima pemilihan
Nama pemilih												
ID / E KTP pemilih												
Login / masuk												
Validasi ID / E KTP Pemilih												
Cara memilih												
Kode tanda terima pemilihan												

Gambar 4.21 User interface halaman tata cara pemilihan.

BAB V

IMPLEMENTASI DAN PENGUJIAN

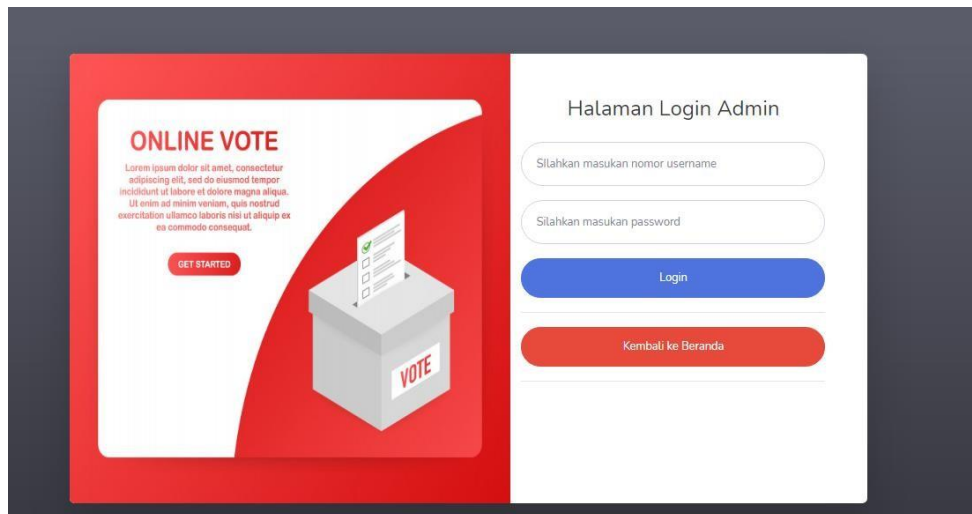
5.1 Implementasi

Implementasi adalah tahap penerapan sistem berdasarkan hasil analisis dan perancangan yang dilakukan pada bab IV. Pada bab V ini merupakan implementasi hasil rancangan menjadi Aplikasi E-Voting Pemilihan Kepala Desa Berbasis Web Di Desa Baros menggunakan PHP, MySQL, dan XAMPP untuk memudahkan panitia dalam melaksanakan pemilihan kepala desa.

5.1.1 Implementasi User Interface

Berikut adalah beberapa gambar hasil implementasi user interface:

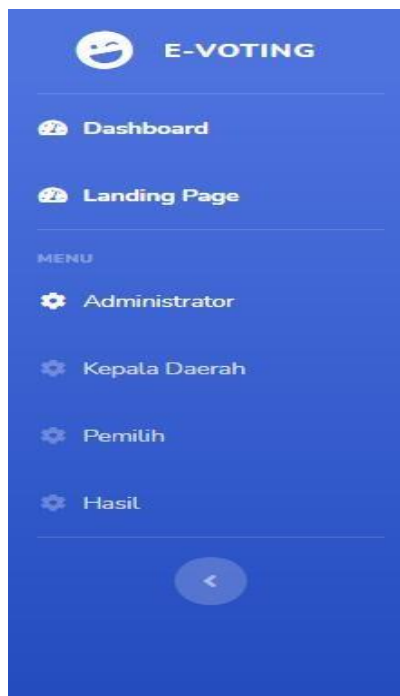
1. Tampilan E_voting untuk Admin
 - a. Tampilan login Admin



Gambar 5.1 Tampilan Login Admin

Gambar 5.1 diatas merupakan tampilan menu login pada *aplikasi e-voting* pemilihan kepala desa. Terdapat dua kolom untuk mengisi user name, password dan terdapat menu sig-in untuk masuk kedalam aplikasi.

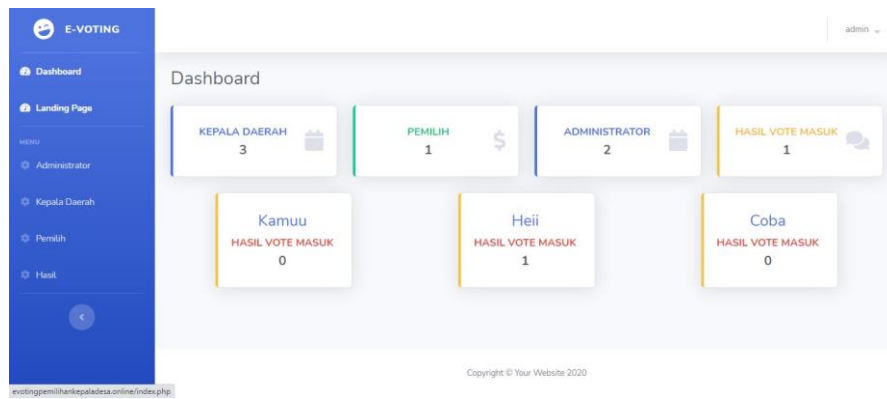
b. Tampilan Interface menu admin Aplikasi e-voting
pemilihan kepala desa



Gambar 5.2 Menu Admin *e-voting*

Pada Gambar 5.2 diatas merupakan menu admin pada *e-voting*, yang terdapat beberapa sub menu yang terdiri dari: menu dashboard, menu Administrator, menu kepala desa, menu pemilih, dan menu hasil.

Berikut ini beberapa tampilan from dari sub menu Administrator, menu kepala desa, menu pemilih, dan menu hasil yaitu sebagai berikut

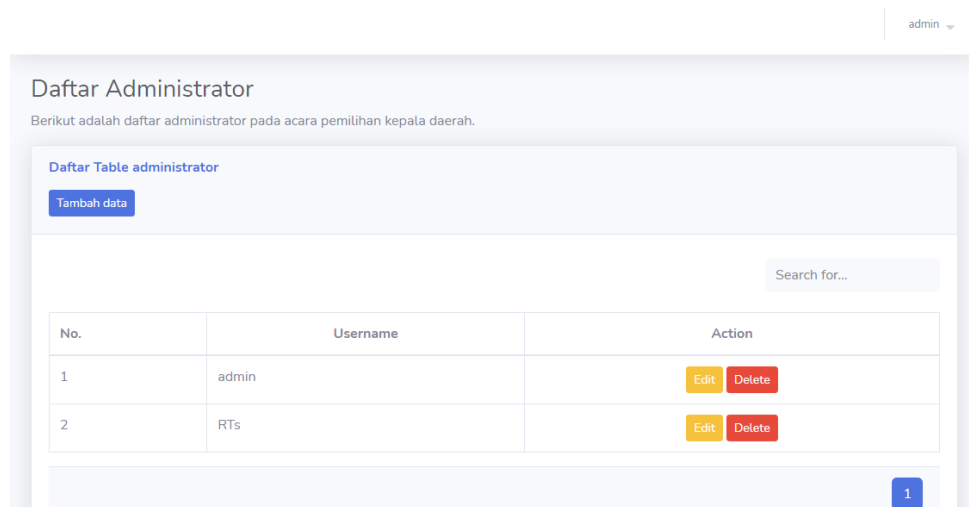


5.2 Menu Dashboard

Gambar 5.3 Interface Dashboard
Aplikasi *e-voting* pemilihan

Pada gambar diatas merupakan gambar interface dashboard aplikasi e_voting pemilihan kepala desa, yang memuat beberapa menu seperti menu kepala daerah, pemilih, administrasi, hasil vote masu, dan 3 contoh menu nama calon kepala desa.

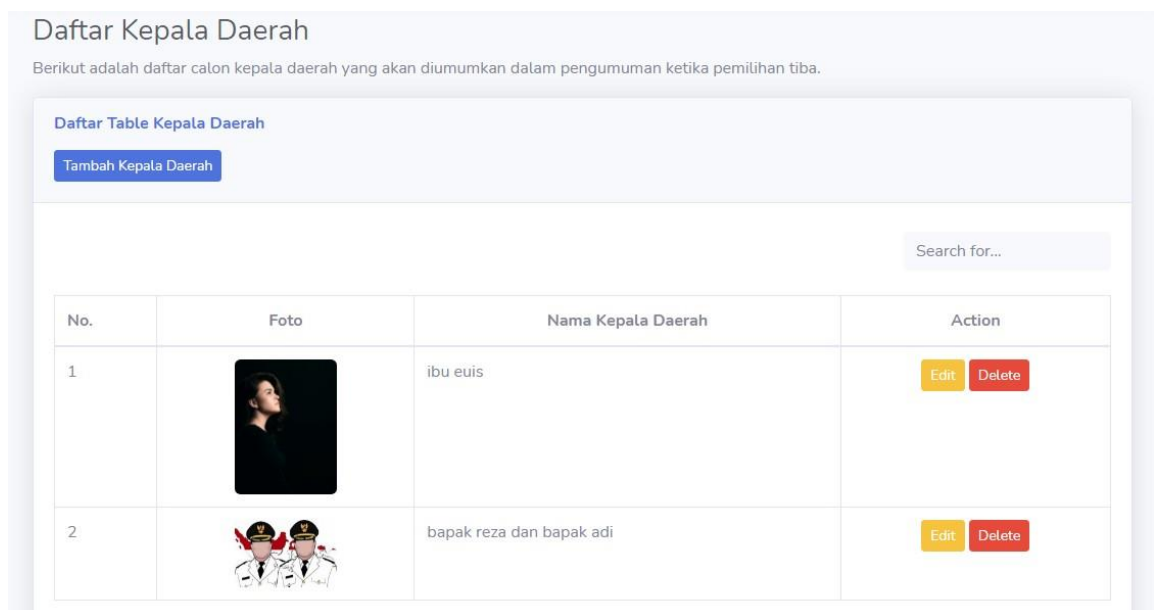
5.3 Menu Administrator



Gambar 5.4 Menu Administrator

Pada gambar 5.5 ini merupakan menu daftar administrator yang bertugas mengurus administrasi. admin juga bisa menambahkan form admin *e-voting* pemilihan kepaladesa.

5.4 Menu kepala daerah



Gambar 5.5 Menu kepala daerah

Pada gambar 5.5 ini yaitu menu daftar calon kepala desa , admin bisa mengubah maupun memasukan data calon kepla desa pada kolom tambah kepala desa, seperti pada gambar berikut :

Gambar 5.6 form tambah calon kepala
desa

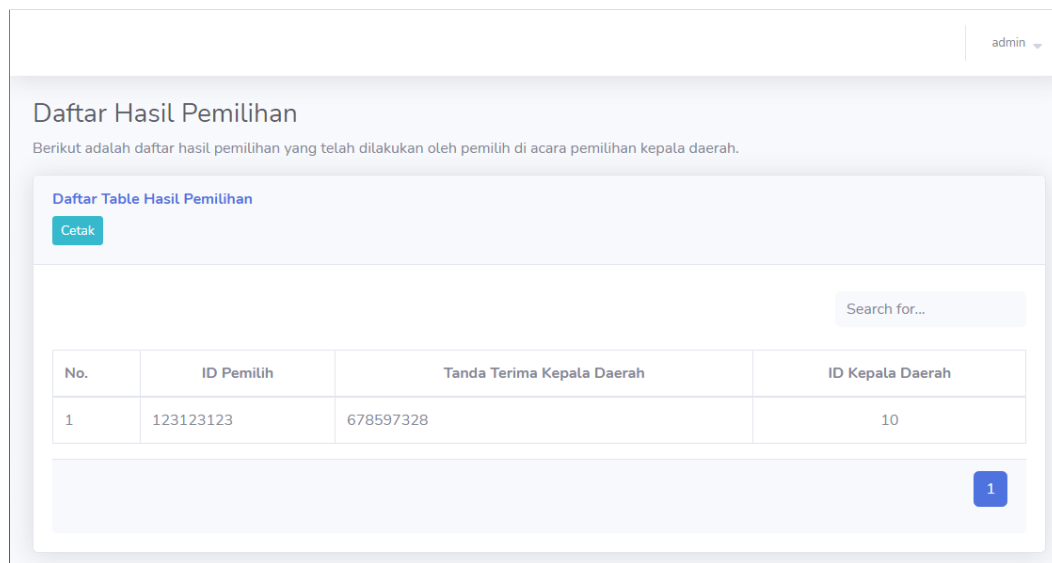
5.5 Menu Daftar Pemilih

No.	KTP	Nama Pemilih	Alamat Pemilih	Status Pemilih	Action
1	123123123	agus	asasa	Sudah Memilih	Edit Delete

Gambar 5.7 Menu pemilih

Pada gambar 5.7 ini admin bisa mengisi data pemilih sesuai dengan KTP.

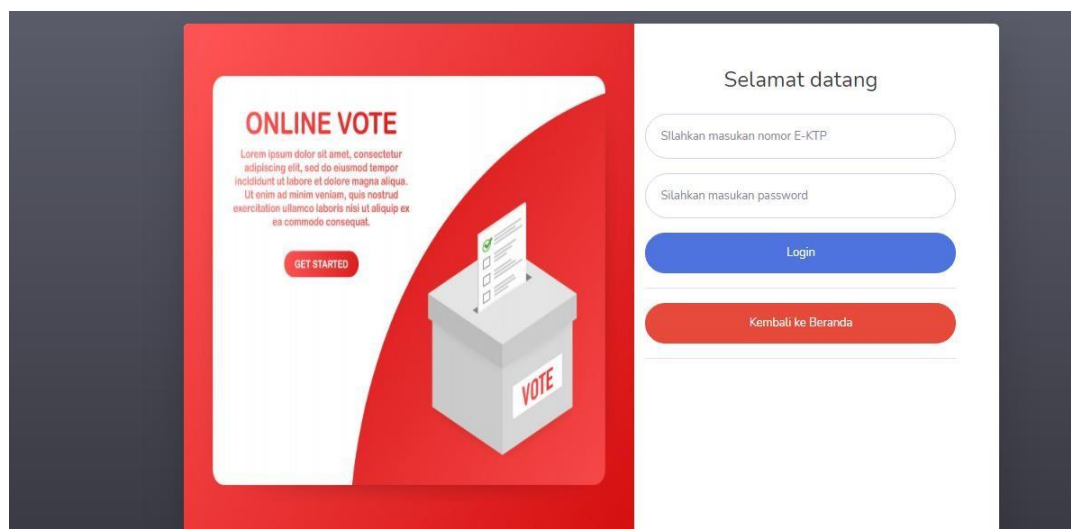
5.6 Menu Hasil pemilihan



Gambar 5.8 Menu pemilih

Pada gambar 5.8 ini yaitu jumlah hasil voting untuk calon kepala desa pada pemilihan yang sudah di vote oleh pemilih (Warga)

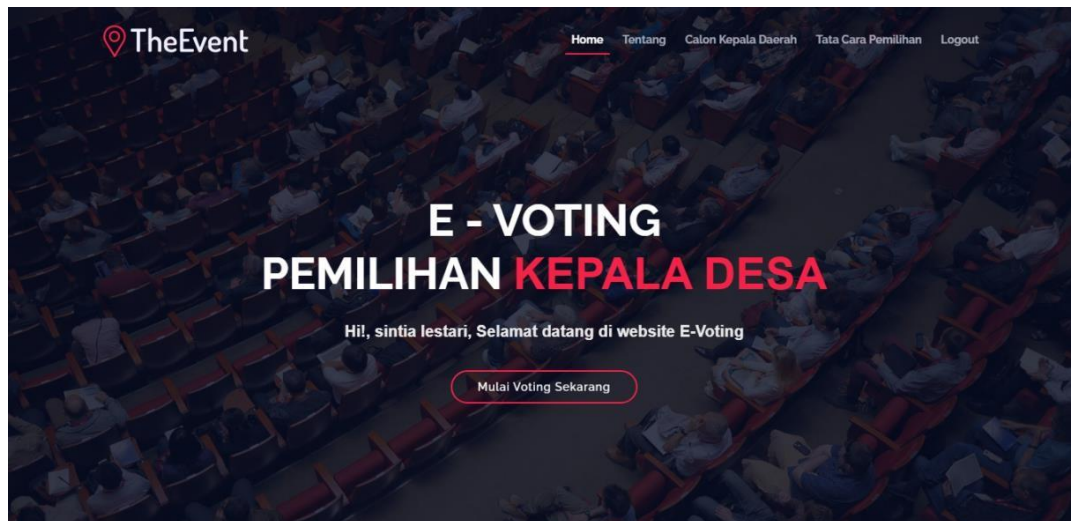
1. Tampilan E-voting Untuk pemilih
 - a. Tampilan login pemilih



Gambar 5.9 tampilan login pemilih

Pada gambar ini yaitu tampilan login pemilih e-voting, pemilih yang sudah terdaftar di data admin akan diberikan username yaitu no NIK KTP dan password.

b. Tampilan awal e-voting pemilihan kepala desa



Gambar 5.10 tampilan awal e-voting
pemilihan kepala desa

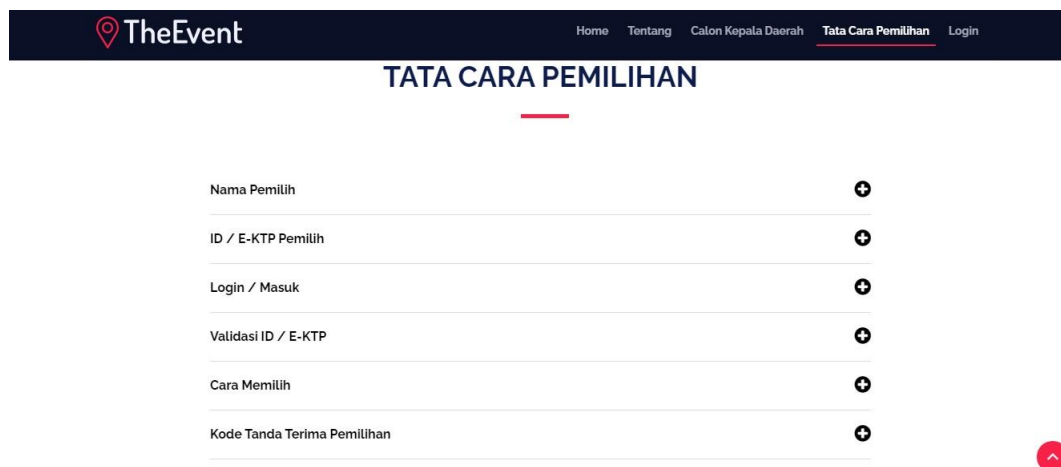
Pada gambar ini setelah pemilih melakukan login terdapat beberapa menu pada home page pemilih yaitu menu tentang, menu daftar calon kepala desa, menu tata cara pemilihan dan logout.

1) Menu tentang *e-voting*



Gambar 5.11 menu tentang Pada gambar ini menjelaskan tentang pengertian *e-voting*

Menu Tata Cara Pemilihan



Gambar 5.12 tata cara pemilihan

Pada gambar ini menjelaskan tata cara pemilihan, yang berisikan penjelasan pada tabel dibawah ini :

Tabel 5.1 isi penjelasan tata cara
pemilihan

Nama pemilih	Pastikan nama anda terdaftar pada daftar pemilih.
ID / E-KTP pemilih	Pastikan menerima kode sandi Pilkada, yang dibagikan tertutup oleh KPPS.
Login / masuk	Masukan nomor E-KTP dan password kedalam menu login yang terdapat pada navigasi bar
validasi ID / E-KTP	Pastikan data diri sesuai dengan pada E-KTP anda, apabila ada kesalahan hubungi KPPS.
Cara memilih	Pilihlah dengan mengklik tombol lingkaran yang terdapat dibawah calon Kepala Daerah, kemudian klik submit
Kode tanda terima pemilihan	Setelah melakukan submit, anda akan diarahkan pada halaman kode tanda pemilih, Harap simpan kode tersebut untuk mengecek hasil pemilihan. *catatan : Apabila terjadi kesusahan dalam pemilihan hubungi KPPS setempat.

2) Menu Daftar Calon Kepala Desa

TheEvent Home Form Pemilihan Logout

SINTIA LESTARI
 Nomor E-KTP : 320450517
 Alamat : desa baros kecamatan Arjasari

IBU EUIS
 Jika anda ingin memilih ibu euis mohon centang kolom vote dibawah
☐ Vote

BAPAK REZA DAN BAPAK ADI
 Jika anda ingin memilih bapak reza dan bapak adi mohon centang kolom vote dibawah
☐ Vote

BAPAK ARI
 Jika anda ingin memilih bapak ari mohon centang kolom vote dibawah
☒ Vote

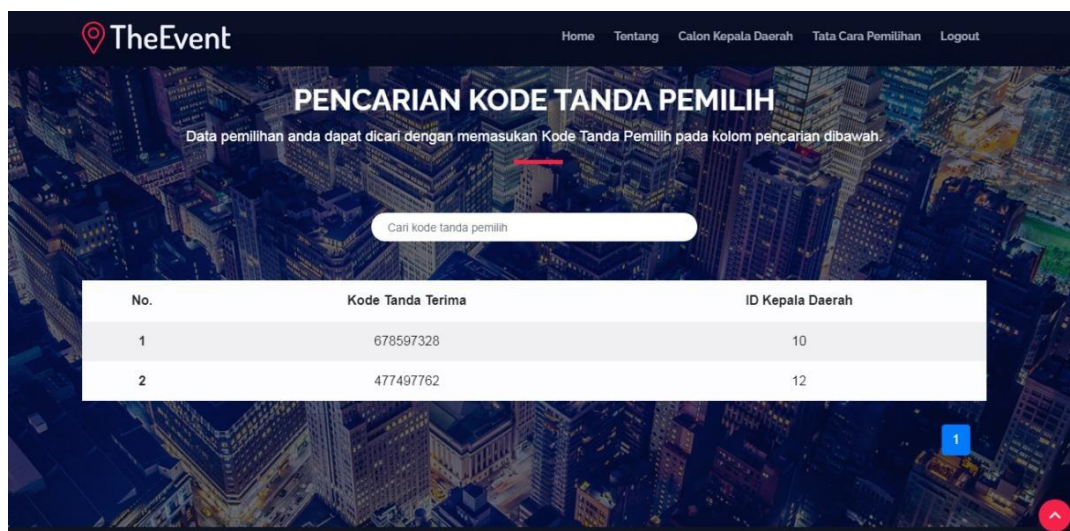
Submit

Gambar 5. 13 Tampilan daftar calon kepala desa

Pada gambar ini pemilih bisa melihat daftar calon kepala desa yang sudah terdaftar dan juga bisa langsung memilih calon kepala desa yang sudah terdaftar. Jika sudah memilih kemudian klik menu submit untuk mengirimkan hasil vote kepanitia petugas pemilihan tersebut. Setelah melakukan submit, setelah akan diarahkan pada halaman kode tanda pemilih, untuk pemilih Harap simpan kode tersebut untuk mengecek hasil pemilihan yang pemilih vote pada gambar berikut :



Gambar 5.14 tampilan tanda terima pemilihan kepala desa



Gambar 5.15 tampilan pencarian kode tanda pemilih

Pada pemilihan tersebut pemilih hanya bisa melakukan pemilihan 1 kali, maka akan menampilkan tampilan page seperti dibawah ini jika pemilih melakukan pemilihan lebih dari 1 kali.



Gambar 5.16 tampilan e-ktp yang sudah melakukan pemilihan

5.1.2. Implementasi Bahasa Pemograman Php

Pada Penelitian ini, peneliti menggunakan bahasa pemograman Php. Karena php mendukung berbagai macam database. Php mudah dipelajari dan berjalan secara efisien di server. Php mendukung banyak jumlah protocol besar seperti POP3,IMAP, dan LADP. PHP 4 menambahkan dukungan untuk bahasa pemograman Java dan arsitektur objek yang terdistribusi (COM dan CORBA).

5.1.3 Implementasi XAMPP

Pada penelitian ini, peneliti menggunakan software xampp. Penggunaan dari xampp sangat dibutuhkan untuk dapat mengembangkan software ataupun tampilan website dengan lebih mudah, cepat, dan terstruktur. Terdapat tigakomponen penyusun utama dari tool ini, yaitu HTDoc, Control Panel, dan PhpMy Admin.penggunaan aplikasi web server ini sebagai tool bantuan untuk mulai belajar tahapan pengembangan perangkat lunak sesuai kebutuhan atau proyek bisnis.

5.1.4 Hasil Implementasi

Hasil implementasi adalah untuk menerapkan dan mewujudkan rencana yang telah disusun menjadi bentuk nyata. Adapun keseluruhan implementasi meliputi: pengkodean terhadap basis data, antar muka, coding aplikasi, perancangan sistem dan lain-lain. Hasil dari tahapan tersebut adalah Aplikasi *e-voting* pemilihan kepala desa yang dapat diakses melalui website.

5.1.5 Pengujian

Pada penelitian ini, penulis menggunakan *Black Box Testing* untuk menguji aplikasi. *Black Box Testing* merupakan pengujian yang dapat dilakukan dengan melakukan pengamatan, pada hasil eksekusi melalui beberapa data uji dan memeriksa fungsional yang terdapat pada perangkat lunak. Jadi dapat kita dianalogikan seperti halnya kita melihat kedalam kotak hitam, sehingga kita hanya bisa melihat tampilan luarnya saja tanpa kita tau apa yang ada di dalam kotak hitam tersebut.

Sehingga sama seperti halnya dengan *Black Box Testing* yang hanya dapat mengevaluasi dari tampilan luarnya dan fungsionalitasnya. Tanpa harus mengetahui apa sesungguhnya yang terjadi dalam proses detailnya. yakni pada aplikasi yang seharusnya dilakukan. Menggunakan deskripsi eksternal dalam perangkat lunak, termasuk spesifikasi persyaratan, dan desain untuk digunakan dalam uji kasus, pengujian ini dapat menjadikan suatu fungsional atau non fungsional meski pun pengujian tersebut sering fungsional.

Pengujian *Black Box* akan dilakukan dengan cara yang relative bertentangan dengan kebutuhan yang ada dan memasukan sistem dapat menangani semua masukan yang tidak sesuai. Oleh karena itu, pengguna hanya bisa memasukan data yang benar ke dalam sistem. Pengujian ini berusaha menemukan kesalahan misalnya, seperti:

Fungsi-fungsi yang tidak benar atau hilang terdapat pada perangkat lunak.

- 1) Kesalahan dalam *interface* perangkat lunak
- 2) Kesalahan dalam struktur data atau akses database eksternal dalam perangkat lunak.
- 3) Permasalahan dalam kerja perangkat lunak.
- 4) Inisialisasi dan kesalahan terminasi perangkat lunak.

Berikut ini merupakan table Black Box Testing yang bisa digunakan dalam melakukan pengujian perangkat lunak, yaitu sebagai berikut:

Tabel 5.2 black box testing

Skenario Pengujian	Kasus Pengujian	Hasil Yang Diharapkan	Hasil Pengujian	Kesimpulan

Tambah data calon kepala desa, data admin dan data pemilih	Masukan data calon, data admin dan data pemilih klik tombol tambah	Data calon kepala desa, data admin dan data pemilih di tambah	Sesuai	Normal
Edit data calon kepala desa, data admin dan data pemilih	Klik data yang ingin di edit klik tombol edit	Data calon kepala desa, data admin dan data pemilih berhasil di edit	Sesuai	Normal
Delete data calon kepala desa, data	Klik data yang	Data calon	Sesuai	Normal

admin dan data pemilih	ingin di hapus kilik tombol hapus	kepala desa,data admin dan data pemilih berhasil di hapus		
---------------------------	--------------------------------------	--	--	--

Berikut ini merupakan teknik yang sering digunakan *Black Box Testing* desain meliputi:

1. Descision Table

Decision Table merupakan cara yang tepat untuk memodelkan logika yang cukup rumit, seperti diagram alur, *if-then-else* dan *switch* laporan kasus. Dalam kondisi ini mengaitkan dengan tindakan untuk melakukan, akan tetapi banyak kasus melakukannya dengan cara yang lebih elegan.

2. All-Pairs Testing

All-Pairs Testing atau disebut pairwise testing merupakan metode pengujian perangkat lunak kombinatorial yang digunakan untuk setiap pasangan parameter yang masuk kedalam sistem atau algoritma yang ada pada perangkat lunak.

3. State Transition Table

State Transition Table merupakan teori automata dan logika skuensial, pada table yang menunjukan *state* dalam pengujian. Pada dasarnya sebuah table *state* merupakan table kebenaran yang digunakan untuk beberapa *input* dan *output* termasuk dengan *state* berikutnya dengan kondisi yang sebenarnya terjadi.

4. Equivalence Partitioning

Equivalence Partitioning merupakan teknik yang membagi data masukan dari beberapa unit perangkat lunak menjadi beberapa partisi data dari mana *test case* dapat diturunkan. Pada prinsipnya, uji kasus ini dirancang untuk menutupi setiap partisi minimal

BAB VI

KESIMPULAN DAN SARAN

6.1. Kesimpulan

Berdasarkan pengamatan dari setiap tahap-tahap yang dilakukan oleh penulis pada perancangan aplikasi e-voting pemilihan kepala desa di desa baros maka penulis mengambil kesimpulan sebagai berikut :

1. Aplikasi e-voting akan membantu kegiatan pemilihan kepala desa pada tahap pemungutan suara dan penghitungan suara.
2. Dengan proses autentikasi sebelum pemilih melakukan pemungutan suara (voting) dapat menjaga identitas pemilih karena data daftar pemilih tetap tersimpan dalam database.
3. Pemungutan suara secara elektronik dapat menjamin keaslian suara karena suara yang diberikan pemilih diproses oleh sistem dan tersimpan secara terpusat.
4. Penghitungan suara secara elektronik dengan aplikasi e-voting ini menjadi lebih akurat serta hasil perhitungan suara menjadi lebih cepat diperoleh dibandingkan harus melakukan perhitungan secara manual yang memakan waktu lama dan membutuhkan ketelitian ekstra. Sehingga untuk pembuatan laporan hasil perolehan suara pun lebih mudah dan cepat.

6.2. Saran

Berdasarkan kesimpulan diatas maka penulisan menyarankan beberapa hal terkait “Ranca bangun Aplikasi E-voting pemilihan kepala desa di desa baros, sebagai berikut :

1. Aplikasi e-voting ini hanya membantu kegiatan pemungutan suara dan penghitungan suara saja pada pemilihan kepala desa . Untuk lebih mempermudah tahapan-tahapan pada kegiatan pemilihan tersebut, aplikasi

ini dapat dikembangkan dengan menambahkan fungsi pendaftaran pemilih .

2. Untuk tahap autentikasi pemilih pada aplikasi e-voting ini masih menggunakan proses login dengan input No.KTP dan No.Verifikasi. Seiring berkembangnya teknologi proses login dapat digantikan dengan proses Tap e- KTP. Sehingga proses autentikasi pemilih lebih mudah dan keamanan data pemilih pun lebih terjamin.

DAFTAR PUSTAKA

- azhari. (2005). Electronic voting (E- voting).
- Canard. (2001). *KRITERIA SISTEM E-VOTING*.
- EMS, t. (2012). PENGERTIAN PHP.
- Enterprise, J. (2015). PEMROGRAMAN BOOTSTRAP UNTUK PEMULA.
- Fahlevi. (2013). *PENGERTIAN DATA BASE*.
- Febrizal Alfarasy Syam, Y. D. (2013). PERANCANGAN SISTEM E-VOTING UNTUK PEMILIHAN KETUA OSIS SMP NEGERI 10 PEKANBARU.
- gregorius. (2012). PENGERTIAN WWW (WORLD WIDE WEB).
- Handoko. (2005). PERUBAHAN SGML MENJADI HTML.
- Kurniawan. (2010). *PENEGRTIAN MYSQL*.
- Ladjamuddin. (2006). *PENGERTIAN DAD (DIAGRAM ALIR DATA)*.
- Nugroho. (2011). PENGERTIAN PHP MENURUT PARA AHLI.
- Pressman. (2012). *Metode pengembangan sistem yang digunakan pada pembuatan aplikasi e-voting pemilihan kepala desa adalah metode Waterfall*.
- purwanti. (2015). *Pengrtian E-voting*.
- Purwati, N. (2015). PERANCANGAN SISTEM E-VOTING UNTUK PEMILIHAN KEPALA DAERAH (PILKADA) .
- rossa, s. (2018). *PENGERTIAN UNIFIED MODELING LANGUAGE (UML)*.
- Teresa Enades Hari Setia, A. S. (2016). SMART CONTRACT BLOCKCHAIN PADA E-VOTING.

LAMPIRAN

Codding Function Dashboard

```
<?php
```

```
require_once '../config/get_connection.php';
```

```
function GetKepalaDaerah()
```

```
{  
    $query = "SELECT * FROM kepala_daerah";  
    $exe = mysqli_query(connect(), $query);  
    while ($data = mysqli_fetch_array($exe)) {  
        $datas[] = array(  
            'id_kepala_daerah' => $data['id_kepala_daerah'],  
            'nama_kepala_daerah' => $data['nama_kepala_daerah'],  
            'gambar_kepala_daerah' => $data['gambar_kepala_daerah'],  
        );  
    }  
    return $datas;  
}
```

```
function GetPemilih()
```

```
{  
    $query = "SELECT * FROM pemilih";  
    $exe = mysqli_query(connect(), $query);  
    while ($data = mysqli_fetch_array($exe)) {  
        $datas[] = array(  
            'id_pemilih' => $data['id_pemilih'],  
            'nama_pemilih' => $data['nama_pemilih'],  
            'gambar_pemilih' => $data['gambar_pemilih'],  
        );  
    }  
    return $datas;  
}
```

```
function GetHasil()
```

```

{
    $datas = [];
    $query = "SELECT * FROM hasil";
    $exe = mysqli_query(connect(), $query);
    while ($data = mysqli_fetch_array($exe)) {
        $datas[] = array(
            'id_hasil' => $data['id_hasil'],
            'id_pemilih' => $data['id_pemilih'],
            'tdterima_kepala_daerah' => $data['tdterima_kepala_daerah'],
            'id_kepala_daerah' => $data['id_kepala_daerah'],
        );
    }
    return $datas;
}

```

```

function GetAdministrator()
{
    $query = "SELECT * FROM administrator";
    $exe = mysqli_query(connect(), $query);
    while ($data = mysqli_fetch_array($exe)) {
        $datas[] = array(
            'id_administration' => $data['id_administration'],
            'username' => $data['username'],
            'password' => $data['password'],
        );
    }
    return $datas;
}

```

Coding Login Admin

```
<?php
```

```
require_once('../templates/header.html');
```

```
?>
```

```
<body class="bg-gradient-dark">
```

```

<div class="container mt-lg-4">

<!-- Outer Row -->
<div class="row justify-content-center">

<div class="col-xl-10 col-lg-12 col-md-9">

<div class="card o-hidden border-0 shadow-lg my-5">
<div class="card-body p-0">
<!-- Nested Row within Card Body -->
<div class="row">
<div class="col-lg-6 d-none d-lg-block">

</div>
<div class="col-lg-6">
<div class="p-5">
<div class="text-center">
<h1 class="h4 text-gray-900 mb-4">Selamat datang</h1>
</div>
<form action="function.php" method="POST" class="user">
<div class="form-group">
<input type="text" name="username" class="form-control form-control-user"
id="exampleInputEmail" aria-describedby="emailHelp" placeholder="Silahkan masukan nomor
username">
</div>
<div class="form-group">
<input type="password" name="password" class="form-control form-control-user"
id="exampleInputPassword" placeholder="Silahkan masukan password">
</div>
<button type="submit" name="admin_login" class="btn btn-primary btn-user btn-
block">
Login
</button>
<hr>
<a href="../../users/index.php" class="btn btn-danger btn-user btn-block">Kembali ke
Beranda</a>
<hr>
</form>

```

```

        <div class="text-center">
            <p>Apakah anda ingin login sebagai user ? Silahkan klik <a class="small"
href="index.php"><u>Disini.</u></a>
        </div>
    </div>
</div>
</div>
</div>
</div>
</div>

</div>

</div>
</body>

```

Coding Administrator

```

<?php
    session_start();
    require_once '../config/get_connection.php';
    function getAll()
    {
        $exe = pagination()["exe"];
        while ($row = mysqli_fetch_assoc($exe)) {
            $data[] = [
                'id_administrator' =>
$row['id_administrator'],
                'username' => $row['username'],
                'password' => $row['password'],
            ];
        }

        return $data;
    }
    function pagination()

```

```

{
    $min_data = 10;
    $page = isset($_GET['page']) ? $_GET['page'] : 1;
    $start = ($page > 1) ? ($page * $min_data) - $min_data : 0;
    $result_page = mysqli_query(connect(), "SELECT * FROM
administrator");

    $total = mysqli_num_rows($result_page);
    $total_page = ceil($total / $min_data);

    $query = "SELECT * FROM administrator ORDER BY
id_administrator ASC LIMIT $start, $min_data";
    $exe = mysqli_query(connect(), $query);

    $output = [
        'exe' => $exe,
        'total_page' => $total_page,
        'page' => $page
    ];
    return $output;
}

function GetById($id)
{
    $query = "SELECT * FROM `administrator` WHERE
`id_administrator` = '$id'";

    $exe = mysqli_query(connect(), $query);
    while ($data = mysqli_fetch_array($exe)) {
        $datas[] = array(
            'id_administrator' =>
$data['id_administrator'],
            'username' => $data['username'],
            'password' => $data['password'],
        );
    }
    return $datas;
}

function GetBySearch($search)

```

```

{
    $query = "SELECT * FROM `administrator` WHERE
username LIKE '%$search%' OR password LIKE '%$search%' ";
    $exe = mysqli_query(connect(), $query);
    while ($data = mysqli_fetch_array($exe)) {
        $datas[] = array(
            'id_administrator' =>
$data['id_administrator'],

            'username' => $data['username'],
            'password' => $data['password'],

        );
    }
    return $datas;
}

function insert()
{
    $username = $_POST['username'];
    $password = $_POST['password'];

    $query = "INSERT INTO `administrator`
(`id_administrator`,`username`,`password`)
VALUES (NULL,'$username','$password)";
    $exe = mysqli_query(connect(), $query);
    if ($exe) {
        // kalau berhasil
        $_SESSION['success'] = " Data added! ";
        header("Location: index.php");
    } else {
        $_SESSION['failed'] = " Data failed to add ";
        header("Location: index.php");
    }
}

function Update($id)
{
    $username = $_POST['username'];
    $password = $_POST['password'];

```

```

        $query = "UPDATE `administrator` SET `username` =
'$username', `password` = '$password' WHERE `id_administrator` = '$id'";
        $exe = mysqli_query(connect(), $query);
        if ($exe) {
            // kalau berhasil
            $_SESSION['success'] = " Data updated! ";
            header("Location: index.php");
        } else {
            $_SESSION['failed'] = " Data failed to updated ";
            header("Location: index.php");
        }
    }
    function Delete($id)
    {
        $query = "DELETE FROM `administrator` WHERE
`id_administrator` = '$id'";
        $exe = mysqli_query(connect(), $query);
        if ($exe) {
            // kalau berhasil
            $_SESSION['success'] = " Data deleted! ";
            header("Location: index.php");
        } else {
            $_SESSION['failed'] = " Data failed to delete ";
            header("Location: index.php");
        }
    }
    if (isset($_POST['insert'])) {
        insert();
    } else if (isset($_POST['update'])) {
        update($_POST['id_administrator']);
    } else if (isset($_POST['delete'])) {
        delete($_POST['id_administrator']);
    } else if (isset($_POST['search'])) {
        GetBySearch($_POST['search']);
    }
    ?>

```

Coding User

```

<?php
session_start();
require_once '../config/get_connection.php';

function getAll()
{
    $exe = pagination()["exe"];
    while ($row = mysqli_fetch_assoc($exe)) {
        $data[] = [
            'id_hasil' => $row['id_hasil'],
            'id_pemilih' => $row['id_pemilih'],
            'tdterima_kepala_daerah' => $row['tdterima_kepala_daerah'],
            'id_kepala_daerah' => $row['id_kepala_daerah'],
            'status' => $row['status'],
            'ktp' => $row['ktp'],
            'nama_kepala_daerah' => $row['nama_kepala_daerah'],
        ];
    }

    return $data;
}

function pagination()
{
    $min_data = 10;
    $page = isset($_GET['page']) ? $_GET['page'] : 1;
    $start = ($page > 1) ? ($page * $min_data) - $min_data : 0;
    $result_page = mysqli_query(connect(), "SELECT * FROM hasil");
    $total = mysqli_num_rows($result_page);
    $total_page = ceil($total / $min_data);

    $query = "SELECT * FROM hasil INNER JOIN pemilih ON hasil.id_pemilih =
pemilih.id_pemilih INNER JOIN kepala_daerah ON hasil.id_kepala_daerah =
kepala_daerah.id_kepala_daerah ORDER BY id_hasil ASC LIMIT $start, $min_data";
    $exe = mysqli_query(connect(), $query);

    $output = [

```



```

'exe' => $exe,
'total_page' => $total_page,
'page' => $page
];
return $output;
}

function insert()
{

$id_pemilih = $_POST['id_pemilih'];
// mt_srand($id_pemilih);
// $tdterima_kepala_daerah = mt_rand(1000000, 9000000);
$ktp = $_POST['ktp'];
$sub_ktp = substr($ktp, 0, 10);
mt_srand($sub_ktp);
$tdterima_kepala_daerah = mt_rand(100000000, 999999999);
$id_kepala_daerah = $_POST['id_kepala_daerah'];

$query = "INSERT INTO `hasil`
(`id_hasil`,`id_pemilih`,`tdterima_kepala_daerah`,`id_kepala_daerah`) VALUES
(NULL,'$id_pemilih','$tdterima_kepala_daerah','$id_kepala_daerah')";

$status_pilkada = "Sudah Memilih";
$query2 = "UPDATE `pemilih` SET `status_pilkada` = '$status_pilkada' WHERE `id_pemilih` =
'$id_pemilih'";

$exe = mysqli_query(connect(), $query);
$exe2 = mysqli_query(connect(), $query2);
if ($exe && $exe2) {
    // kalau berhasil

    $_SESSION['success'] = " Data added! ";
    header("Location: code.php");
} else {
    $_SESSION['failed'] = " Data failed to add ";
    header("Location: index.php");
}
}

```

```
}
```

```
function GetBySearch($search)
```

```
{
```

```
    $query = "SELECT * FROM hasil INNER JOIN pemilihan ON hasil.id_pemilih =  
pemilih.id_pemilih WHERE tdterima_kepala_daerah LIKE '%$search%' OR id_kepala_daerah  
LIKE '%$search%'";
```

```
    $exe = mysqli_query(connect(), $query);
```

```
    while ($data = mysqli_fetch_array($exe)) {
```

```
        $datas[] = array(
```

```
            'id_hasil' => $data['id_hasil'],
```

```
            'id_pemilih' => $data['id_pemilih'],
```

```
            'tdterima_kepala_daerah' => $data['tdterima_kepala_daerah'],
```

```
            'ktp' => $data['ktp'],
```

```
            'id_kepala_daerah' => $data['id_kepala_daerah'],
```

```
            'nama_kepala_daerah' => $data['nama_kepala_daerah'],
```

```
        );
```

```
    }
```

```
    $query2 = "SELECT * FROM kepala_daerah WHERE nama_kepala_daerah LIKE '%$search%'  
";
```

```
    $exe2 = mysqli_query(connect(), $query2);
```

```
    while ($data = mysqli_fetch_array($exe2)) {
```

```
        $datas[] = array(
```

```
            'id_kepala_daerah' => $data['id_kepala_daerah'],
```

```
            'nama_kepala_daerah' => $data['nama_kepala_daerah'],
```

```
        );
```

```
    }
```

```
    return $datas;
```

```
}
```

```
function GetHasil()
```

```
{
```

```
    $query = "SELECT * FROM `hasil` ";
```

```

$exe = mysqli_query(connect(), $query);
while ($data = mysqli_fetch_array($exe)) {
    $datas[] = array(
        'id_pemilih' => $data['id_pemilih'],
        'status' => $data['status'],
        'tdterima_kepala_daerah' => $data['tdterima_kepala_daerah'],

    );
}
return $datas;
}

```

```

function GetPemilih()
{
    $query = "SELECT * FROM `pemilih` ";
    $exe = mysqli_query(connect(), $query);
    while ($data = mysqli_fetch_array($exe)) {
        $datas[] = array(
            'id_pemilih' => $data['id_pemilih'],
            'status_pilkada' => $data['status_pilkada'],
        );
    }
    return $datas;
}

```

```

function GetKepalaDaerah()
{
    $query = "SELECT * FROM kepala_daerah";
    $exe = mysqli_query(connect(), $query);
    while ($data = mysqli_fetch_array($exe)) {
        $datas[] = array(
            'id_kepala_daerah' => $data['id_kepala_daerah'],
            'nama_kepala_daerah' => $data['nama_kepala_daerah'],
            'gambar_kepala_daerah' => $data['gambar_kepala_daerah'],
        );
    }
    return $datas;
}

```

```

if (isset($_POST['insert'])) {
    insert();
} else if (isset($_POST['update'])) {
    update($_POST['id_pemilih']);
} else if (isset($_POST['delete'])) {
    delete($_POST['id_pemilih']);
} else if (isset($_POST['search'])) {
    GetBySearch($_POST['search']);
} else if (isset($_POST['login'])) {
    GetLogin($_POST['login']);
}

```

Coding Pemilih

```

<?php

session_start();
require_once '../config/get_connection.php';
function getAll()
{
    $exe = pagination()["exe"];
    while ($row = mysqli_fetch_assoc($exe)) {
        $data[] = [
            'id_pemilih' => $row['id_pemilih'],
            'ktp' => $row['ktp'],
            'password' => $row['password'],
            'nama_pemilih' => $row['nama_pemilih'],
            'alamat' => $row['alamat'],
            'status_pilkada' => $row['status_pilkada'],
        ];
    }

    return $data;
}

function pagination()
{
    $min_data = 10;
    $page = isset($_GET['page']) ? $_GET['page'] : 1;
    $start = ($page > 1) ? ($page * $min_data) - $min_data : 0;

```

```

        $result_page = mysqli_query(connect(), "SELECT * FROM
pemilih");

        $total = mysqli_num_rows($result_page);
        $total_page = ceil($total / $min_data);

        $query = "SELECT * FROM pemilihan ORDER BY id_pemilih
ASC LIMIT $start, $min_data";
        $exe = mysqli_query(connect(), $query);

        $output = [
            'exe' => $exe,
            'total_page' => $total_page,
            'page' => $page
        ];
        return $output;
    }

    function GetById($id)
    {
        $query = "SELECT * FROM `pemilih` WHERE `id_pemilih`
= '$id'";

        $exe = mysqli_query(connect(), $query);
        while ($data = mysqli_fetch_array($exe)) {
            $datas[] = array(
                'id_pemilih' => $data['id_pemilih'],
                'ktp' => $data['ktp'],
                'password' => $data['password'],
                'nama_pemilih' =>
$data['nama_pemilih'],
                'alamat' => $data['alamat'],
                'status_pilkada' =>
$data['status_pilkada'],

            );
        }
        return $datas;
    }

    function GetBySearch($search)

```

```

    {
        $query = "SELECT * FROM `pemilih` WHERE ktp LIKE
'%$search%' OR password LIKE '%$search%' OR nama_pemilih LIKE '%$search%' OR alamat
LIKE '%$search%' OR status_pilkada LIKE '%$search%' ";
        $exe = mysqli_query(connect(), $query);
        while ($data = mysqli_fetch_array($exe)) {
            $datas[] = array(
                'id_pemilih' => $data['id_pemilih'],
                'ktp' => $data['ktp'],
                'password' => $data['password'],
                'nama_pemilih' =>
$data['nama_pemilih'],
                'alamat' => $data['alamat'],
                'status_pilkada' =>
$data['status_pilkada'],
            );
        }
        return $datas;
    }

function insert()
{
    $ktp = $_POST['ktp'];
    $password = $_POST['password'];
    $nama_pemilih = $_POST['nama_pemilih'];
    $alamat = $_POST['alamat'];
    $status_pilkada = $_POST['status_pilkada'];

    $query = "INSERT INTO `pemilih`
(`id_pemilih`,`ktp`,`password`,`nama_pemilih`,`alamat`,`status_pilkada`)
VALUES
(NULL,$ktp,$password,$nama_pemilih,$alamat,$status_pilkada)";
    $exe = mysqli_query(connect(), $query);
    if ($exe) {
        // kalau berhasil
        $_SESSION['success'] = " Data added! ";
        header("Location: index.php");
    } else {

```

```

        $_SESSION['failed'] = " Data failed to add ";
        header("Location: index.php");
    }
}

function Update($id)
{
    $ktp = $_POST['ktp'];
    $password = $_POST['password'];
    $nama_pemilih = $_POST['nama_pemilih'];
    $alamat = $_POST['alamat'];
    $status_pilkada = $_POST['status_pilkada'];

    $query = "UPDATE `pemilih` SET `ktp` = '$ktp', `password` =
'$password', `nama_pemilih` = '$nama_pemilih', `alamat` = '$alamat', `status_pilkada` =
'$status_pilkada' WHERE `id_pemilih` = '$id'";

    $exe = mysqli_query(connect(), $query);
    if ($exe) {
        // kalau berhasil
        $_SESSION['success'] = " Data updated! ";
        header("Location: index.php");
    } else {
        $_SESSION['failed'] = " Data failed to updated ";
        header("Location: index.php");
    }
}

function Delete($id)
{
    $query = "DELETE FROM `pemilih` WHERE `id_pemilih` =
'$id'";

    $exe = mysqli_query(connect(), $query);
    if ($exe) {
        // kalau berhasil
        $_SESSION['success'] = " Data deleted! ";
        header("Location: index.php");
    } else {
        $_SESSION['failed'] = " Data failed to delete ";
        header("Location: index.php");
    }
}

```

```

if (isset($_POST['insert'])) {
    insert();
} else if (isset($_POST['update'])) {
    update($_POST['id_pemilih']);
} else if (isset($_POST['delete'])) {
    delete($_POST['id_pemilih']);
} else if (isset($_POST['search'])) {
    GetBySearch($_POST['search']);
}
?>

```

Coding Kepala Daerah

```

<?php
    session_start();
    require_once '../config/get_connection.php';
    function getAll()
    {
        $exe = pagination()["exe"];
        while ($row = mysqli_fetch_assoc($exe)) {
            $data[] = [
                'id_kepala_daerah' =>
                    $row['id_kepala_daerah'],
                'nama_kepala_daerah' =>
                    $row['nama_kepala_daerah'],
                'gambar_kepala_daerah' =>
                    $row['gambar_kepala_daerah'],
            ];
        }

        return $data;
    }
}

```



```

function pagination()
{
    $min_data = 10;
    $page = isset($_GET['page']) ? $_GET['page'] : 1;
    $start = ($page > 1) ? ($page * $min_data) - $min_data : 0;
    $result_page = mysqli_query(connect(), "SELECT * FROM
kepala_daerah");

    $total = mysqli_num_rows($result_page);
    $total_page = ceil($total / $min_data);

    $query = "SELECT * FROM kepala_daerah ORDER BY
id_kepala_daerah ASC LIMIT $start, $min_data";
    $exe = mysqli_query(connect(), $query);

    $output = [
        'exe' => $exe,
        'total_page' => $total_page,
        'page' => $page
    ];
    return $output;
}

function GetById($id)
{
    $query = "SELECT * FROM `kepala_daerah` WHERE
`id_kepala_daerah` = '$id'";
    $exe = mysqli_query(connect(), $query);
    while ($data = mysqli_fetch_array($exe)) {
        $datas[] = array(
            'id_kepala_daerah' =>
$data['id_kepala_daerah'],
            'nama_kepala_daerah' =>
$data['nama_kepala_daerah'],
            'gambar_kepala_daerah' =>
$data['gambar_kepala_daerah'],
        );
    }
    return $datas;
}

```

```

    }

    function GetBySearch($search)
    {
        $query = "SELECT * FROM `kepala_daerah` WHERE
nama_kepala_daerah LIKE '%$search%' OR gambar_kepala_daerah LIKE '%$search%' ";
        $exe = mysqli_query(connect(), $query);
        while ($data = mysqli_fetch_array($exe)) {
            $datas[] = array(
                'id_kepala_daerah' =>
$data['id_kepala_daerah'],
                'nama_kepala_daerah' =>
$data['nama_kepala_daerah'],
                'gambar_kepala_daerah' =>
$data['gambar_kepala_daerah'],

            );
        }
        return $datas;
    }

    function insert()
    {
        $nama_kepala_daerah = $_POST['nama_kepala_daerah'];
        $nama_file = $_FILES['gambar_kepala_daerah']['name'];
        $source = $_FILES['gambar_kepala_daerah']['tmp_name'];
        $folder = '../assets/img/kepala_daerah/';

        move_uploaded_file($source, $folder . $nama_file);
        $query = "INSERT INTO `kepala_daerah`
(`id_kepala_daerah`,`nama_kepala_daerah`,`gambar_kepala_daerah`)
VALUES (NULL,$nama_kepala_daerah,$nama_file)";
        $exe = mysqli_query(connect(), $query);
        if ($exe) {
            // kalau berhasil
            $_SESSION['success'] = " Data added! ";
            header("Location: index.php");
        } else {
            $_SESSION['failed'] = " Data failed to add ";

```

```

        header("Location: index.php");
    }
}
function Update($id)
{
    $nama_kepala_daerah = $_POST['nama_kepala_daerah'];
    $nama_file = $_FILES['gambar_kepala_daerah']['name'];
    $source = $_FILES['gambar_kepala_daerah']['tmp_name'];
    $folder = '../assets/img/kepala_daerah/';

    if ($nama_file != "") {
        move_uploaded_file($source, $folder . $nama_file);
        $query = "UPDATE `kepala_daerah` SET
`nama_kepala_daerah` = '$nama_kepala_daerah', `gambar_kepala_daerah` = '$nama_file' WHERE
`id_kepala_daerah` = '$id'";

        $exe = mysqli_query(connect(), $query);
        if ($exe) {
            // kalau berhasil
            $_SESSION['success'] = " Data added! ";
            header("Location: index.php");
        } else {
            $_SESSION['failed'] = " Data failed to
add ";

            header("Location: index.php");
        }
    } else {
        $query = "UPDATE `kepala_daerah` SET
`nama_kepala_daerah` = '$nama_kepala_daerah' WHERE `id_kepala_daerah` = '$id'";
        $exe = mysqli_query(connect(), $query);
        if ($exe) {
            // kalau berhasil
            $_SESSION['success'] = " Data updated!
";

            header("Location: index.php");
        } else {
            $_SESSION['failed'] = " Data failed to
updated ";

            header("Location: index.php");
        }
    }
}

```

```

    }
}
function Delete($id)
{
    $query = "DELETE FROM `kepala_daerah` WHERE
`id_kepala_daerah` = '$id'";
    $exe = mysqli_query(connect(), $query);
    if ($exe) {
        // kalau berhasil
        $_SESSION['success'] = " Data deleted! ";
        header("Location: index.php");
    } else {
        $_SESSION['failed'] = " Data failed to delete ";
        header("Location: index.php");
    }
}
if (isset($_POST['insert'])) {
    insert();
} else if (isset($_POST['update'])) {
    update($_POST['id_kepala_daerah']);
} else if (isset($_POST['delete'])) {
    delete($_POST['id_kepala_daerah']);
} else if (isset($_POST['search'])) {
    GetBySearch($_POST['search']);
}
?>

```

Coding Hasil

```

<?php
session_start();
require_once '../config/get_connection.php';
function getAll()
{
    $exe = pagination()["exe"];
    while ($row = mysqli_fetch_assoc($exe)) {
        $data[] = [
            'id_hasil' => $row['id_hasil'],
            'id_pemilih' => $row['id_pemilih'],

```

```

                                'tdterima_kepala_daerah' =>
$row['tdterima_kepala_daerah'],
                                'id_kepala_daerah' =>
$row['id_kepala_daerah'],
                                'status' => $row['status'],
                                'ktp' => $row['ktp'],
                                'nama_kepala_daerah' =>
$row['nama_kepala_daerah'],
                                ];
                                }

                                return $data;
                                }

                                function pagination()
                                {
                                        $min_data = 10;
                                        $page = isset($_GET['page']) ? $_GET['page'] : 1;
                                        $start = ($page > 1) ? ($page * $min_data) - $min_data : 0;
                                        $result_page = mysqli_query(connect(), "SELECT * FROM
hasil");

                                        $total = mysqli_num_rows($result_page);
                                        $total_page = ceil($total / $min_data);

                                        $query = "SELECT * FROM hasil INNER JOIN pemilihan ON
hasil.id_pemilih = pemilihan.id_pemilih INNER JOIN kepala_daerah ON hasil.id_kepala_daerah =
kepala_daerah.id_kepala_daerah ORDER BY id_hasil ASC LIMIT $start, $min_data";
                                        $exe = mysqli_query(connect(), $query);

                                        $output = [
                                                'exe' => $exe,
                                                'total_page' => $total_page,
                                                'page' => $page
                                        ];
                                        return $output;
                                }

                                function GetById($id)
                                {

```

```

$query = "SELECT * FROM `hasil` WHERE `id_hasil` =
'$id'";

$exe = mysqli_query(connect(), $query);
while ($data = mysqli_fetch_array($exe)) {
    $datas[] = array(
        'id_hasil' => $data['id_hasil'],
        'id_pemilih' => $data['id_pemilih'],
        'tdterima_kepala_daerah' =>
$data['tdterima_kepala_daerah'],
        'id_kepala_daerah' =>
$data['id_kepala_daerah'],
        'status' => $data['status'],
    );
}
return $datas;
}

function GetBySearch($search)
{
    $query = "SELECT * FROM hasil INNER JOIN pemilih ON
hasil.id_pemilih = pemilih.id_pemilih WHERE tdterima_kepala_daerah LIKE '%$search%' OR
id_kepala_daerah LIKE '%$search%' OR ktp LIKE '%$search%' ";
    $exe = mysqli_query(connect(), $query);
    while ($data = mysqli_fetch_array($exe)) {
        $datas[] = array(
            'id_hasil' => $data['id_hasil'],
            'id_pemilih' => $data['id_pemilih'],
            'tdterima_kepala_daerah' =>
$data['tdterima_kepala_daerah'],
            'ktp' => $data['ktp'],
            'id_kepala_daerah' =>
$data['id_kepala_daerah'],
        );
    }
    return $datas;
}

```

```

function Update($id)
{
    $tdterima_kepala_daerah = $_POST['tdterima_kepala_daerah'];
    $id_kepala_daerah = $_POST['id_kepala_daerah'];

    $query = "UPDATE `hasil` SET `tdterima_kepala_daerah` =
'$tdterima_kepala_daerah',`id_kepala_daerah` = '$id_kepala_daerah' WHERE `id_hasil` = '$id'";
    $exe = mysqli_query(connect(), $query);
    if ($exe) {
        // kalau berhasil
        $_SESSION['success'] = " Data updated! ";
        header("Location: index.php");
    } else {
        $_SESSION['failed'] = " Data failed to updated ";
        header("Location: index.php");
    }
}

function Delete($id)
{
    $query = "DELETE FROM `hasil` WHERE `id_hasil` = '$id'";
    $exe = mysqli_query(connect(), $query);
    if ($exe) {
        // kalau berhasil
        $_SESSION['success'] = " Data deleted! ";
        header("Location: index.php");
    } else {
        $_SESSION['failed'] = " Data failed to delete ";
        header("Location: index.php");
    }
}

function GetPemilih()
{
    $query = "SELECT * FROM pemilih";
    $exe = mysqli_query(connect(), $query);
    while ($data = mysqli_fetch_array($exe)) {
        $datas[] = array(
            'id_pemilih' => $data['id_pemilih'],

```

```

                                'status_pilkada' =>
$data['status_pilkada'],
                                );
                                }
                                return $datas;
                                }

function GetKepalaDaerah()
{
    $query = "SELECT * FROM kepala_daerah";
    $exe = mysqli_query(connect(), $query);
    while ($data = mysqli_fetch_array($exe)) {
        $datas[] = array(
                                'id_kepala_daerah' =>
                                $data['id_kepala_daerah'],
                                'nama_kepala_daerah' =>
                                $data['nama_kepala_daerah'],
                                'gambar_kepala_daerah' =>
                                $data['gambar_kepala_daerah'],
                                );
        }
    return $datas;
}

if (isset($_POST['insert'])) {
    insert();
} else if (isset($_POST['update'])) {
    update($_POST['id_hasil']);
} else if (isset($_POST['delete'])) {
    delete($_POST['id_hasil']);
} else if (isset($_POST['search'])) {
    GetBySearch($_POST['search']);
}
?>

```


RIWAYAT HIDUP



Fradita Audina, lahir di Garut pada tanggal 05 Agustus 1997. Anak ke dua dari tiga bersaudara, yang lahir dari pasangan Toto Suyono dan Noneng Suryamah. Mulai mengenyam Pendidikan Dasar di SDN Bayongbong 1 (2005-2010), melanjutkan pendidikan tingkat menengah pertama di SMPN 1 Bayongbong – Garut (2011-2013), serta melanjutkan pendidikan menengah atas di SMAN 19 Garut (2014-2016). Untuk mendapatkan gelar sarjana, Fradita melanjutkan ke jenjang S1 di Universitas Bale Bandung Fakultas Teknologi Informasi jurusan Teknik Informatika.

“The wisest person may swallow the most painful life, hurt by his surroundings. But he chose to make it a valuable lesson. ”