



**IMPLEMENTASI TEKNOLOGI BLOCKCHAIN PADA
WEBSITE E-VOTE MENGGUNAKAN BAHASA
PEMROGRAMAN PYTHON**

Muhammad Reza Pahlevi

16360039

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI INFORMASI
INSTITUT SAINS DAN TEKNOLOGI NASIONAL
JAKARTA
2020**

SURAT PERNYATAAN ORISINALITAS

Skripsi ini adalah hasil karya saya sendiri dan semua sumber baik yang dikutip maupun dirujuk telah saya nyatakan dengan benar.

Nama : Muhammad Reza Pahlevi

NIM : 16360039

Tanggal : 27 Agustus 2020

Jakarta, 27 Agustus 2020

Yang Menyatakan

Muhammad Reza Pahlevi

SURAT PERNYATAAN NON PLAGIAT

Saya yang bertanda tangan dibawah ini :

Nama : Muhammad Reza Pahlevi

NPM : 16360039

Mahasiswa : Teknik Informatika

Tahun Akademik : 2019/2020

Menyatakan bahwa saya tidak melakukan kegiatan plagiat dalam penulisan Tugas Akhir yang berjudul **“Implementasi Teknologi Blockchain Pada Website E-Vote Menggunakan Bahasa Pemrograman Python”**.

Apabila suatu saat nanti terbukti saya melakukan plagiat, maka saya akan menerima sanksi yang telah ditetapkan.

Demikian Surat Pernyataan ini saya buat dengan sebenar – benarnya.

Jakarta, 24 Agustus 2020

Muhammad Reza Pahlevi

LEMBAR PENGESAHAN

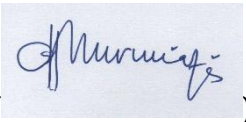
Skripsi ini diajukan oleh :

Nama : Muhammad Reza Pahlevi
NIM : 16360039
Program Studi : Teknik Informatika
Judul Tugas Akhir : Implementasi Teknologi Blockchain Pada Website E-Vote
Menggunakan Bahasa Pemrograman Python

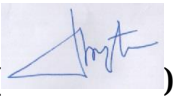
Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai persyaratan yang diperlukan untuk memperoleh gelar **Sarjana Komputer** pada Program Studi **Teknik Informatika**, Fakultas **Sains Dan Teknologi Informasi**, Institut Sains Dan Teknologi Nasional.

DEWAN PENGUJI

Pembimbing : Siti Madinah Ladjamuddin, S.Kom, M.Kom ()

Penguji : Siti Nurmiati, S.Kom, M.Kom ()

Penguji : Andi Suprianto, Ir.M.Kom ()

Penguji : Aryo Nur Utomo, ST.M.Kom ()

Ditetapkan di : Jakarta
Tanggal : 24 Agustus 2020

KATA PENGANTAR

Puji dan syukur kehadiran Tuhan Yang Maha Esa atas berkat rahmat serta kasih-Nya sehingga dapat menyelesaikan skripsi ini yang mengambil judul “*IMPLEMENTASI TEKNOLOGI BLOCKCHAIN PADA WEBSITE E-VOTE MENGGUNAKAN BAHASA PEMROGRAMAN PYTHON*”

Tujuan penulisan skripsi ini untuk memenuhi sebahagian syarat memperoleh gelar Sarjana Komputer (S.Kom) bagi mahasiswa program S-1 di program studi Teknik Informatika Fakultas Sains dan Teknologi Informasi INSTITUT SAINS DAN TEKNOLOGI NASIONAL. menyadari bahwa skripsi ini masih jauh dari kesempurnaan, oleh sebab itu mengharapkan kritik dan saran yang bersifat membangun dari semua pihak demi kesempurnaan skripsi ini.

Terselesaikannya skripsi ini tidak terlepas dari bantuan banyak pihak, sehingga pada kesempatan ini dengan segala kerendahan hati dan penuh rasa hormat pengarang menghaturkan terima kasih yang sebesar-besarnya bagi semua pihak yang telah memberikan bantuan moril maupun materil baik langsung maupun tidak langsung dalam penyusunan skripsi ini hingga selesai, terutama yang saya hormati kepada:

1. Kedua orang tua dan seluruh keluarga yang telah memberi doa, semangat dan perhatian.
2. Ibu Siti Madinah Ladjamuddin, S.Kom, M.Kom selaku dosen pembimbing dalam pembuatan skripsi.
3. Ibu Herly Nurahmi, S.Si, M.Kom selaku kepala program studi Teknik informatika institute sains dan teknologi nasional. Dan
4. Bapak dan Ibu dosen Program Studi Teknik Informatika Institut Sains dan Teknologi Nasional.
5. Teman – teman seperjuangan Teknik Informatika angkatan 2016 yang telah membantu dan memberikan motivasi selama penulisan skripsi ini.

Akhir kata penulis mengucapkan terimakasih kepada semua pihak yang telah membantu dan penulis berharap semoga skripsi ini dapat bermanfaat bagi kita semua dan menjadi bahan masukan dalam dunia pendidikan.

Jakarta, 16 Juli 2020

Penulis,

Muhammad Reza Pahlevi

NIM. 16360039

SURAT PERNYATAAN PERSETUJUAN PUBLIKASI KARYA ILMIAH UNTUK KEPENTINGAN AKADEMIK

Sebagai civitas akademika Institut Sains Dan Teknologi Nasional saya yang bertanda tangan dibawah ini :

Nama : Muhammad Reza Pahlevi
NIM : 16360039
Program Studi : Teknik Informatika
Fakultas : Fakultas Sains Dan Teknologi Informasi
Jenis Karya : Skripsi

Demi pengembangan ilmu pengetahuan, dengan ini menyetujui untuk memberikan kepada Institut Sains Dan Teknologi Nasional **Hak Bebas Royalti Non-Eksklusif (*Non-exclusive Royalti-Free Right*)** atas karya ilmiah saya yang berjudul “Implementasi Teknologi Blockchain Pada Website E-Vote Menggunakan Bahasa Pemrograman Python” beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Non-Eksklusif ini pihak Institut Sains Dan Teknologi Nasional berhak menyimpan, mengalih media/format-kan, mengelola dalam bentuk pangkalan data (*database*) *soft copy* atau *hard copy*, merawat, dan mempublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pemcipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Jakarta

Pada Tanggal : 27 Agustus 2020

Yang Menyatakan

Muhammad Reza Pahlevi

ABSTRAK

Nama : Muhammad Reza Pahlevi

NIM : 16360039

**Judul : Implementasi Teknologi Blockchain Pada Website E-Vote Menggunakan
Bahasa Pemrograman Phyton**

Pemilihan umum adalah hal yang sering menjadi polemik ketika hasil dari *voting* tersebut terasa janggal. Banyak kasus dimana kecurangan dalam sebuah pemilihan terjadi karena banyak kekurangan dalam sistem pemilihan yang diterapkan terutama dalam *electronic voting*. Dalam pelaksanaan pemilihan umum ini masih menggunakan metode konvensional sehingga tidak efektif, efisien, dan fleksibel serta harus menyediakan anggaran untuk pengadaan kertas dan juga tidak ramah lingkungan. Penerapan teknologi Blockchain dalam *website e-voting* merupakan salah satu pencegahan dalam kecurangan dan pembobolan sistem. Dengan menerapkannya akan menghasilkan data yang *real* dan tidak dimanipulasi. Dengan sistem *desentralized distributed database* yang diamankan dengan sistem enkripsi akan menjamin keamanan *database* dengan baik.

Kata Kunci : Blockchain, electronic voting, voter, phyton, website

ABSTRACT

Nama : Muhammad Reza Pahlevi

NIM : 16360039

**Judul : Blockchain Technology Implementation On E-Vote Website Using Python
Programming Language**

General elections are things that often become polemic when the results of the voting seem odd. Many cases where fraud in an election occurs due to many deficiencies in the electoral system that is applied, especially in electronic voting. In the implementation of this general election still using conventional methods so that it is not effective, efficient, and flexible and must provide a budget for paper procurement and also not environmentally friendly. The application of Blockchain technology on e-voting websites is one of the prevention of fraud and system breaches. By implementing it will produce real data and not manipulated. With a decentralized distributed database system that is secured with an encryption system, it will ensure the security of the database properly.

Keywords : Blockchain, electronic voting, voter, python, website

DAFTAR ISI

SURAT PERNYATAAN ORISINALITAS.....	i
SURAT PERNYATAAN NON-PLAGIAT.....	ii
LEMBAR PENGESAHAN.....	iii
KATA PENGANTAR.....	iv
SURAT PERNYATAAN PERSETUJUAN PUBLIKASI KARYA ILMIAH	vi
ABSTRAK	vii
DAFTAR ISI	ix
DAFTAR GAMBAR	xi
DAFTAR TABEL	xiv
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah	2
1.4 Tujuan Penelitian	3
1.5 Manfaat Penelitian	3
BAB II TINJAUAN PUSTAKA	4
2.1 Blockchain	4
2.2 Distributed database and decentralized system.....	5
2.3 Merkle Tree.....	6
2.4 Voting Values On E-Voting.....	7
2.5 Python.....	8
2.6 My Structured Query Language (MySQL).....	9
2.7 Django.....	9
2.8 PyCryptodome.....	10
2.9 Multichain (Blockchain Platform).....	11
2.10 SQLite	11

2.11 UML	12
2.12 SH3-256.....	17
BAB III METODOLOGI PENELITIAN.....	18
3.1 Desain Penelitian.....	19
3.2 Metode Penelitian.....	19
3.3 Alat dan Bahan Penelitian.....	21
3.3.1 Alat Penelitian.....	21
3.3.2 Bahan Penelitian.....	22
3.4 Analisa dan Perancangan Aplikasi.....	22
3.4.1 Flowchart Diagram.....	23
3.4.2 Use Case Diagram.....	24
3.4.3 Activity Diagram.....	25
3.4.4 Django Class Diagram.....	25
3.4.5 Class Diagram SQLite.....	27
3.4.6 Sequence Diagram.....	29
3.4.7 Deployment Diagram.....	30
3.5 Perancangan Antar Muka (User Interface).....	31
3.5.1 Antar Muka Web Utama (Voter).....	32
3.5.2 Antar Muka Web Utama (Admin).....	34
3.5.3 Antar Muka Web Utama (Pengawas).....	37
3.5.4 Antar Muka Web Background Proses.....	39
3.6 Rancangan Database.....	42
BAB IV HASIL DAN PEMBAHASAN.....	51
4.1 Hasil Aplikasi.....	51
4.2 Pengujian.....	77
BAB V PENUTUP.....	81
5.1 Kesimpulan.....	81

5.2 Saran.....	81
DAFTAR PUSTAKA.....	82

DAFTAR GAMBAR

2.1 Ilustrasi Blockchain	4
2.2 Ilustrasi Sistem.....	6
2.3 Ilustrasi Merkle Tree.....	7
3.1 Kerangka Kerja Penelitian.....	18
3.2 Model Waterfall.....	19
3.3 Flowchart Diagram.....	23
3.4 Use Case Diagram.....	24
3.5 Activity Diagram.....	25
3.6 Django Class Diagram.....	26
3.7 Class Diagram SQLite.....	27
3.8 Sequence Diagram.....	27
3.9 Deployment Diagram.....	29
3.10 Tampilan Menu Utama.....	32
3.11 Tampilan Login.....	32
3.12 Tampilan Memilih Kandidat.....	33
3.13 Tampilan Selesai Memilih.....	33
3.14 Tampilan Hasil Voting Sementara.....	34
3.15 Tampilan Login admin.....	34
3.16 Tampilan Dashboard Admin.....	35
3.17 Tampilan Tambah Data Voter.....	35
3.18 Tampilan Tambah Data Kandidat.....	36
3.19 Halaman Cetak.....	37

3.20	Login Pengawas.....	37
3.21	Dashboard Pengawas.....	38
3.22	Halaman Cetak Pengawas.....	38
3.23	Tampilan Menu Blockchain Background Proses.....	39
3.24	Tampilan Simulasi Block Dibuat.....	40
3.25	Tampilan Setelah Data Enkripsi Dibuat.....	40
3.26	Tampilan detail dari block yang dibuat.....	41
3.27	Tampilan Block yang terantai.....	41
3.28	Tampilan data voter yang dienkripsi.....	42
4.1	Tampilan Menu Utama.....	51
4.2	Tampilan Login.....	52
4.3	Tampilan Pemilihan Kandidat.....	52
4.4	Sudah Memilih.....	53
4.5	Tampilan Tanda Terima.....	53
4.6	Tampilan Hasil Sementara.....	54
4.7	Tampilan Tabel Suara Kandidat.....	54
4.8	Tampilan Ubah Password.....	55
4.9	Login admin.....	56
4.10	Menu Utama Admin.....	56
4.11	Data Kandidat.....	57
4.12	Tambah Kandidat.....	57
4.13	Data Pemilih.....	58
4.14	Tambah Data Pemilih (Voter).....	59
4.15	tampilan cetak kandidat.....	60
4.16	tampilan cetak data pemilih.....	60
4.17	Manu utama pengawas.....	61
4.18	Tampilan data kandidat oleh pengawas.....	62

4.19 Data Voter Oleh Pengawas.....	62
4.20 Ubah password admin dan pengawas.....	63
4.21 Menu Utama Background Proses.....	64
4.22 Pembuatan Block Data.....	65
4.23 Signature Verification.....	66
4.24 Block Detail.....	67
4.25 Command line block.....	67
4.26 Proses Chaining.....	68
4.27 Data Enkripsi Voter.....	69
4.28 Proses Mining Block.....	70
4.29 List Of Block.....	70
4.30 Proses Verifikasi Block.....	71
4.31 Proses Sinkron Block.....	71
4.32 List Of Confirmed Votes.....	72
4.33 Blockchain Data Detail.....	73
4.34 Data Transaksi.....	74
4.35 Data 9 Confirmation.....	75
4.36 Data 1 Confirmation.....	76
4.37 Data Tanda Terima.....	77

DAFTAR TABEL

2.1	Elemen – Elemen Activity Diagram	13
2.2	Elemen – Elemen Use Case Diagram	14
2.3	Elemen – Elemen Class Diagram	16
3.1	Tabel auth_group.....	43
3.2	Tabel auth_group_permission.....	43
3.3	Tabel auth_permission.....	44
3.4	Tabel auth_user.....	44
3.5	Tabel auth_user_group.....	45
3.6	auth_user_user_permission.....	45
3.7	django_admin_log.....	46
3.8	django_content_type.....	46
3.9	django_migration.....	46
3.10	django_session.....	47
3.11	Tabel simulation_block.....	47
3.12	Tabel simulation_vote.....	48
3.13	simulation_votebackup.....	48
3.14	tb_admin.....	49
3.15	tb_pemilih.....	49
3.16	tb_pencalon.....	49
3.17	tb_pilih.....	50
4.1	Hasil Pengujian.....	77
4.2	Pengujian Pada Perangkat Lain.....	80

BAB I

PENDAHULUAN

1.1 Latar Belakang

Electronic voting (e-Voting) pertama kali diperkenalkan oleh David Shaumm pada awal 1980. Sistem yang digunakan adalah dengan *cryptography-key* yang membantu para *voter* untuk tetap tidak terdeteksi. Estonia adalah negara yang pertama kali menggunakan *electronic voting* hanya menggunakan internet dan kartu tanda penduduk elektronik (e-KTP). Negara selanjutnya yang mengimplementasikan *electronic voting* adalah Norwegia. Sistem yang dibuat mirip seperti yang dimiliki Estonia, tetapi terpaksa tidak diteruskan karena banyak pihak yang takut akan keamanan dari sistem itu. Washington D. C. juga mengembangkan *electronic voting* pada tahun 2010. Tetapi banyak sekali masalah keamanan pada saat melakukan pengujian pada sistemnya. Sehingga proyek tersebut tidak pernah diimplementasikan.

Blockchain merupakan teknologi dasar dari sebuah desain arsitektur *cryptocurrency Bitcoin* yang diciptakan oleh Satoshi Nakamoto pada tahun 2008. Ini merupakan bentuk dari *distributed database* yang mana berisi dari transaksi-transaksi yang disimpan dalam sebuah *block* data. Dengan menggunakan teknologi penyimpanan dan keamanan data seperti ini akan membuat sistem pemilihan umum secara *online* atau *e-voting* akan terjaga keamanan dan transparansi data yang ada pada sistem tersebut.

Dengan acuan terkait dari jurnal IMPLEMENTASI BLOCKCHAIN : STUDI KASUS E-VOTING Satria Damai Kurnia Hu, Henry Novianus Palit, Andreas Handojo Program Studi Teknik Informatika, Fakultas Teknologi Industri, Universitas Kristen Petra dan Jurnal APLIKASI VOTING ONLINE DENGAN MENGGUNAKAN TEKNOLOGI BLOCKCHAIN Ahmad Fajar Prasetyo, Program Studi Teknik Informatika, Sekolah Teknik Elektro dan Informatika, Institut Teknologi Bandung.

Oleh karena itu, sistem dan aplikasi yang akan diciptakan ini akan menjawab beberapa kebutuhan pemungutan suara dengan implementasi dari teknologi *blockchain*. Sistem tersebut akan menjawab *transparency*, yaitu data yang disimpan bersifat terbuka untuk publik, sehingga meningkatkan keadilan dan kebenaran. *Anonymity*, yaitu hanya *voter* itu sendiri yang tahu informasi mengenai *vote* dan semua *ballot* yang terkumpul tidak ada hubungannya dengan *voter*. *Dependability*, yaitu setiap *vote* akan dihitung dan tidak dapat diganti, digandakan ataupun dihapus, serta mengeluarkan hasil yang dapat dipercaya. *Eligibility*, yaitu hanya user yang terverifikasi dan memiliki hak suara yang dapat membuat *ballot* dan melakukan *vote*. *Verifiability*, yaitu sistem bersifat terbuka untuk dapat diperiksa kebenarannya dari prosedur sistem hingga hasil yang dikeluarkan. Sehingga dengan begitu fungsi dari aplikasi e-vote ini sebagai media penyelenggara pemilihan umum berbasis website akan dapat tercapai.

1.2 Rumusan Masalah

Berdasarkan dari uraian latar belakang di atas, dirumuskan masalah yang akan dibahas dalam penelitian, yaitu sebagai berikut :

1. Bagaimana menerapkan sistem blockchain pada *website e-vote*
2. Bagaimana membangun *website* dengan bahasa pemrograman *python* dan dengan *framework django* dan *pycryptodome*.

1.3 Batasan Masalah

Agar penelitian lebih terarah, maka diberi batasan masalah sebagai berikut:

1. Aplikasi yang dibangun berbasis website dengan menggunakan bahasa pemrograman *Python* dan database *MySQL*.
2. Database disimpan menggunakan sistem *blockchain*.
3. Metode enkripsi database menggunakan *SHA3-256*.
4. Validasi dilakukan oleh pihak admin atau pengawas.
5. Data voter yang akan dimasukan adalah data yang terverifikasi pihak lain.
6. Sistem tidak memeriksa dan memproses data diluar aplikasi.

1.4 Tujuan Penelitian

Tujuan penelitian ini adalah:

1. Mengimplementasikan sistem *blockchain* dalam *website e-vote* menggunakan bahasa pemrograman *python*.
2. Menerapkan metode enkripsi SHA3-256 pada *database*.

1.5 Manfaat Penelitian

Manfaat yang diharapkan dalam penelitian ini yaitu :

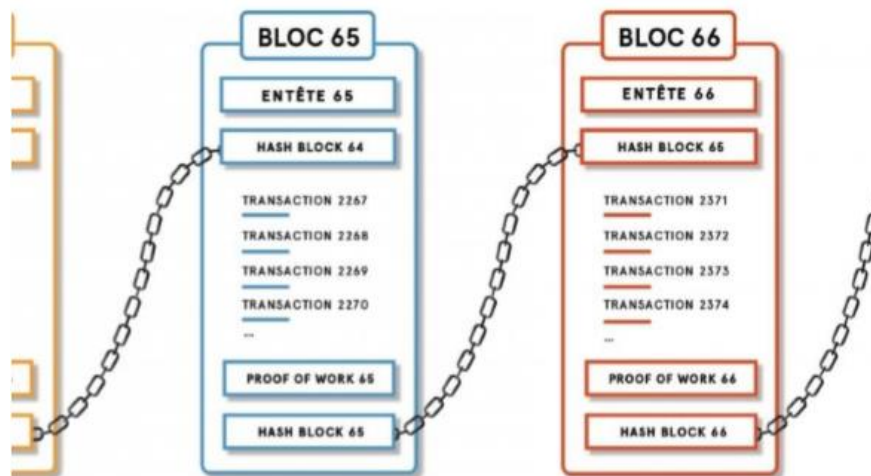
1. Memperkuat sistem keamanan pada *website e-vote*.
2. Mencegah terjadinya kecurangan pada pemungutan suara.
3. Menghemat biaya dan waktu dalam pengadaan *voting*.
4. Menambah ilmu pengetahuan dan pengalaman penulis dibidang sistem keamanan *website* menggunakan *blockchain*.

BAB II

TINJAUAN PUSTAKA

2.1 *Blockchain*

Merupakan sebuah *list of record* yang disebut *blocks*, yang saling terhubung dan diamankan dengan metode *cryptography*. Setiap *block* memiliki *cryptographic-hash* yang berbeda. *Hash* tersebut dibentuk dari isi *block* itu sendiri. Isi *block* tersebut antara lain *timestamp*, beberapa transaksi, dan hash dari *block* sebelumnya sehingga membentuk sebuah rantai (*chain*) dari *blocks*. Ilustrasi struktur *blockchain* dapat dilihat melalui Gambar 2.1.



Gambar 2.1 Ilustrasi Blockchain

Secara desain, *blockchain* resistan terhadap modifikasi data. *Blockchain* merupakan sebuah buku besar terdistribusi (*distributed ledger*) terbuka yang dapat mencatat transaksi antara dua pihak secara efisien dan dengan cara yang dapat diverifikasi dan permanen. Untuk pemanfaatannya sebagai buku besar terdistribusi,

blockchain biasanya dikelola oleh sebuah jaringan peer-to-peer secara kolektif dengan mengikuti protokol tertentu untuk komunikasi antar *node* dan mengkonfirmasi blok-blok baru. Setelah direkam, data dalam blok tidak dapat diubah secara retroaktif tanpa perubahan pada blok-blok berikutnya, yang membutuhkan konsensus mayoritas jaringan.

Konsensus terdesentralisasi dapat dicapai dengan *blockchain*. Hal ini membuat *blockchain* cocok untuk merekam peristiwa, catatan medis, dan aktivitas pengelolaan *record* lainnya, seperti manajemen identitas, pemrosesan transaksi, dokumentasi barang bukti, ketertelusuran makanan (*food traceability*), dan pemungutan suara (*voting*).

Menurut artikel yang dibuat oleh dunia fintech tahun 2018, Pengertian Teknologi *Blockchain* adalah adalah solusi teknologi yang merevolusi cara kerja internet, perbankan dan aplikasi menjadi tanpa server, bersifat terdesentralisasi dan transparansi, serta aman karena setiap peristiwa yang tercatat dalam *Blockchain* tidak dapat diubah.

Blockchain bisa diibaratkan seperti buku kas (*ledger*) suatu bank, namun bentuknya digital dan berskala global. Dalam kasus *electronic voting*, data-data dari *voter* akan dirantai dalam bentuk *block* dari data satu ke data yang lain agar menciptakan struktur data *tree* yang akan memperkuat sistem keamanan data karena data tidak bisa diubah Karena sifat dari sistem *blockchain* ini adalah permanen.

2.2 *Distributed database and decentralized system*

Distributed database merupakan sebuah metode penyimpanan dimana *storage device* tidak terpasang pada sebuah komputer, melainkan beberapa komputer. Beberapa komputer tersebut dapat terkoneksi pada sebuah *storage device* maupun beberapa *storage device* yang berbeda, yang dihubungkan melalui *network*. Metode ini membuat database menjadi transparan terhadap users yang memakainya.

Dalam artikel yang dibuat oleh Iqbal Habiebie tahun 2018, *Database Management System* arti dalam Bahasa Indonesia adalah Sistem manajemen basis data atau kadang disingkat SMBD, adalah suatu sistem atau perangkat lunak yang dirancang untuk mengelola suatu basis data dan menjalankan operasi terhadap data yang diminta banyak pengguna.

Decentralized system merupakan desain sistem yang terdiri dari beberapa komputer yang disebut nodes dalam sebuah jaringan dan memiliki kewenangan untuk mengatur fungsi sendiri untuk mencapai goal dari sistem pusat. Setiap *node* dapat memiliki sistem sendiri dan menjalankan fungsi tertentu untuk memberi kontribusi pada sistem global (main system). Ilustrasi perbedaan sistem *centralized*, dan *decentralized* dapat dilihat pada Gambar 2.2.

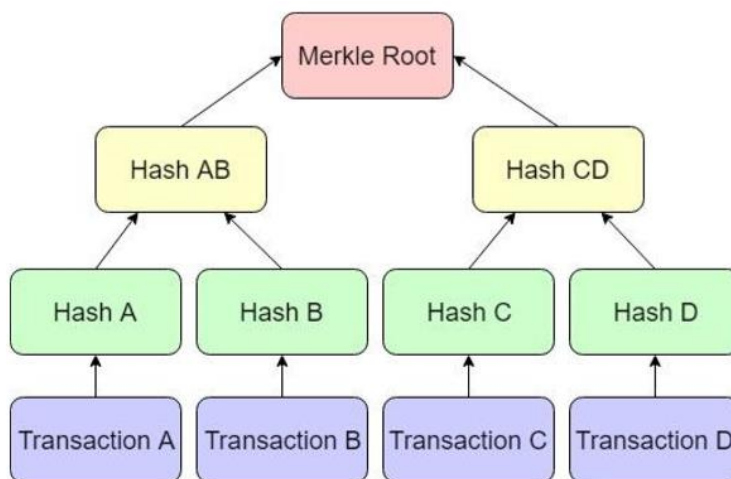


Gambar 2.2 Ilustrasi sistem

2.3 Merkle Tree

Merupakan sebuah struktur *data tree* dimana setiap *leaf node* menyimpan hash dari isi datanya dan node lain (*non-leaf node*) menyimpan hash dari *child nodes* yang dia miliki. Metode ini efektif untuk meningkatkan keamanan dari struktur penyimpanan data yang besar, karena mampu melakukan verifikasi dari setiap node yang ada dalam tree. Ilustrasi struktur *merkle tree* dapat dilihat pada Gambar 2.3.

Dalam jurnal yang dibuat oleh Eka Yusrianto Toisutta pada tahun 2013, Di dalam kriptografi, *Hash Tree* yang juga dikenal sebagai *Merkle Tree* merupakan sebuah struktur data yang mengandung sebuah struktur pohon yang berisikan rangkuman informasi mengenai data yang ukurannya lebih besar. *Hash Tree* merupakan merupakan ekstensi dari *Hash List* yang juga merupakan ekstensi dari fungsi *hash*.



Gambar 2.3 Ilustrasi merkle tree

2.4 Voting Values on e-voting

Electronic Voting yang dibuat haruslah mendukung dengan sifat pemungutan suara secara umum. Berikut sifat dan kebutuhan pemungutan suara elektronik secara umum yang telah dilakukan dalam penelitian sebelumnya.

Dalam penelitian yang dilakukan oleh Ayed tahun 2017, sistem pemungutan suara elektronik harus memiliki nilai *authentication*, *anonymity*, *accuracy*, dan *verifiability*. *Authentication* berarti hanya beberapa orang yang terdaftar yang dapat melakukan *vote*. *Anonymity* berarti identitas *voter* tidak boleh terhubung dengan *ballot* yang dibuat. *Accuracy* berarti *ballot* harus terhitung, dan tidak boleh diganti, digandakan atau dihapus. *Verifiability* berarti sistem seharusnya dapat diverifikasi apakah semua *vote* telah terhitung dengan benar atau tidak.

Dalam penelitian yang dilakukan oleh Lee, James, Ejeta, dan Kim pada tahun 2016, pemungutan suara elektronik memiliki kebutuhan penting seperti *robustness*, *anonymity* dan *transparency*. *Robustness* berarti sistem haruslah tahan terhadap perubahan data. *Anonymity* berarti voter melakukan *vote* dengan rahasia dan tidak diketahui identitasnya. *Transparency* berarti data yang disimpan bersifat transparan dan terbuka untuk *public*.

Dalam penelitian yang dilakukan oleh Liu dan Wang pada tahun 2017, pemungutan suara elektronik yang diusulkan membutuhkan nilai *public verifiability*, *individual verifiability*, *dependability*, *consistency*, *auditability*, *anonymity*, dan *transparency*. *Public verifiability* berarti seluruh pihak terkait dapat memeriksa prosedur dan hasil yang dikeluarkan. *Individual verifiability* berarti voter yang melakukan pemungutan suara dapat memeriksa prosedur yang dilakukan dan *vote* yang dia lakukan apakah sudah masuk dan terhitung atau tidak. *Dependability* berarti sistem yang dibuat memiliki ketahanan terhadap serangan dari luar. *Consistency* berarti setiap suara yang disimpan memiliki data yang sama dan menghasilkan hasil yang sama. *Auditability* berarti seluruh prosedur yang tersimpan dapat diaudit. *Anonymity* berarti hanya voter itu sendiri yang tahu informasi dari *ballot*-nya, dan setiap *ballot* yang telah tersimpan tidak ada berkaitan dengan seseorang. *Transparency* berarti seluruh prosedur bersifat terbuka untuk publik.

2.5 Phyton

Python adalah bahasa pemrograman interpretatif multiguna dengan filosofi perancangan yang berfokus pada tingkat keterbacaan kode. Python diklaim sebagai bahasa yang menggabungkan kapabilitas, kemampuan, dengan sintaksis kode yang sangat jelas, dan dilengkapi dengan fungsionalitas pustaka standar yang besar serta komprehensif. Python juga didukung oleh komunitas yang besar.

Menurut jurnal yang dibuat Trie Maya Kadarina dan Muhammad Hafizd Ibnu Hajar tahun 2019, Bahasa ini merupakan bahasa pemrograman yang tidak

menggunakan *compiler*. Dengan sifat *open-source* yang dimilikinya, bahasa ini dapat digunakan untuk melakukan berbagai hal dari mulai mengembangkan *web*, mengembangkan *video game*, membangun GUI *desktop*, maupun mengembangkan perangkat lunak.

2.6 *My Structure Query Language (MySQL)*

MySQL adalah sebuah perangkat lunak system manajemen basis data SQL (DBMS) yang multithread, dan multi-user. MySQL adalah implementasi dari system manajemen basisdata relasional (RDBMS).

MySQL sebenarnya merupakan turunan salah satu konsep utama dalam database sejak lama, yaitu SQL (*Structured Query Language*). SQL adalah sebuah konsep pengoperasian database, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis. Keandalan suatu sistem database (DBMS) dapat diketahui dari cara kerja optimizer-nya dalam melakukan proses perintah-perintah SQL, yang dibuat oleh user maupun program-program aplikasinya.

Menurut jurnal Mara Destiningrum dan Qadhli Jafar Adrian tahun 2017, Suatu sistem basis *data relation* atau *Relational Database managemnt System*(RDBMS) yang mampu bekerja secara cepat dan mudah digunakan MySQL juga merupakan program pengakses *database* yang bersifat jaringan, sehingga sapat digunakan untuk aplikasi multi user (banyak pengguna). MySQL didistribusikan gratis dibawah lisensi GPL (*General Public License*). Dimana setiap program bebas menggunakan MySQL namun tidak bisa dijadikan produk turunan yang dijadikan closed source atau komersial.

2.7 Django

Django adalah sebuah *framework full-stack* untuk membuat aplikasi web dengan bahasa pemrograman Python. *Framework* akan membantu kita membuat web lebih cepat, dibandingkan menulis kode dari nol.

Full-stack artinya, django meliputi sisi *front-end* dan juga *back-end*. *Front-end* adalah sisi depan yang akan dilihat oleh pengguna, sedangkan *back-end* adalah sisi belakang yang berhubungan dengan database dan logika bisnis.

Dalam artikel yang dibuat oleh Ahmad Muhardian tahun 2016, Django adalah sebuah ***framework full-stack*** untuk membuat aplikasi web dengan bahasa pemrograman Python. Framework akan membantu kita membuat web lebih cepat, dibandingkan menulis kode dari nol.

2.8 PyCryptodome

PyCryptodome adalah sebuah package dari fitur bahasa pemrograman Python yang berguna untuk membuat data enkripsi dalam sebuah tulisan program Python.

Bentuk dari fitur ini lebih seperti *add-in* pada sebuah launch program bahasa pemrograman Python agar para pengguna tidak perlu repot membuat enkripsi tambahan. Dalam penelitian ini, fitur dari *framework* yang digunakan berupa enkripsi keamanan SHA3-256. Penjelasan SHA3-256 ditampilkan pada lampiran 1.

Dalam artikel yang dibuat di halaman *website* PyCryptodome tahun 2019, PyCryptodome adalah paket yang berisi bahasa python dari cryptography yang sangat tua, paket ini terus dikembangkan hingga menjadi kriptografi modern sampai saat ini.

2.9 *Multichain (Blockchain Platform)*

Multichain merupakan sebuah tools untuk membantu pengguna dalam membangun dan menjalankan aplikasi *blockchain*. *Tools* ini dapat membantu dalam mengelola satu atau lebih *blockchain* yang dijalankan pada sebuah komputer. *Multichain* yang telah berjalan pada sebuah komputer akan menjadi sebuah *node*.

Multichain juga menyediakan fungsi yang dapat dipanggil oleh pengguna untuk mengelola *blockchain* dari aplikasi lain ataupun dari luar node. Fungsi yang disediakan adalah JSON-RPC API call.

Dalam jurnal yang dibuat oleh Dimas Yoan Rizaldi dan Ibnu Febry Kurniawan tahun 2019, MultiChain merupakan sebuah platform untuk pembuatan dan penyebaran *blockchain* pribadi, baik di dalam atau di antara organisasi. MultiChain memecahkan masalah terkait mining, privasi, dan keterbukaan melalui pengelolaan izin pengguna yang terintegrasi.

2.10 *SQLite*

SQLite merupakan sebuah sistem manajemen basisdata relasional yang bersifat ACID-compliant dan memiliki ukuran pustaka kode yang relatif kecil, ditulis dalam bahasa C. SQLite merupakan proyek yang bersifat public domain yang dikerjakan oleh D. Richard Hipp.

Tidak seperti pada paradigma client-server umumnya, Inti SQLite bukanlah sebuah sistem yang mandiri yang berkomunikasi dengan sebuah program, melainkan sebagai bagian integral dari sebuah program secara keseluruhan. Sehingga protokol komunikasi utama yang digunakan adalah melalui pemanggilan API secara langsung melalui bahasa pemrograman.

Mekanisme seperti ini tentunya membawa keuntungan karena dapat mereduksi *overhead*, *latency times*, dan secara keseluruhan lebih sederhana. Seluruh elemen basisdata (definisi data, tabel, indeks, dan data) disimpan sebagai sebuah file.

Kesederhanaan dari sisi disain tersebut bisa diraih dengan cara mengunci keseluruhan file basis data pada saat sebuah transaksi dimulai.

Dalam jurnal yang dibuat oleh ANGGA SETIYADI, TATI HARIHAYATI 2016, SQLite adalah suatu *library* yang menerapkan mesin *database self-contained*, serverless, zero-configuration, dan transactional. Berbeda dengan *relationalDBMS* pada umumnya, SQLite memiliki kode pustaka relatif kecil, mudah digunakan tanpa membutuhkan penginstalasian dan konfigurasi.

2.11 Unified Modelling Language (UML)

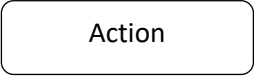
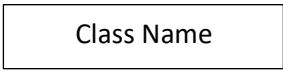
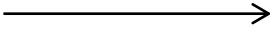
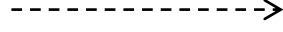
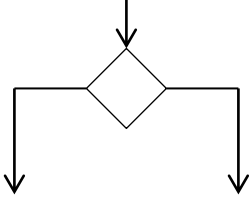
UML (Unified Modelling Language) merupakan sebuah bahasa yang berdasarkan grafik / gambar untuk memvisualisasi, menspesifikasikan, membangun, dan pendokumentasian dari sebuah perangkat lunak yang berbasis object oriented. Dengan kata lain UML adalah sebuah bahasa standard untuk pengembangan sebuah software yang dapat menyampaikan bagaimana membuat dan membentuk model-model. UML juga bukan sebuah bahasa pemrograman visual saja, namun juga dapat secara langsung dihubungkan ke berbagai bahasa pemrograman, seperti JAVA, C++, Visual Basic atau bahkan dihubungkan secara langsung ke dalam sebuah object - oriented database (Kawano et al., 1983).

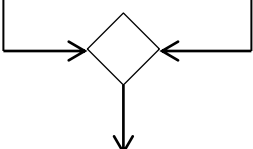
Beberapa macam diagram UML yaitu:

1. Activity Diagram

Diagram ini menggambarkan serangkaian aliran aktivitas, yang digunakan untuk menggambarkan aktivitas yang membentuk operasi, sehingga dapat digunakan untuk aktivitas lain, seperti *Use Case* atau interaksi. Sesuai dengan namanya, diagram ini membahas kegiatan yang dilakukan dalam sistem dari yang pertama hingga yang terakhir. Diagram ini dijelaskan dalam bentuk grafik. Fungsi diagram ini adalah sebagai deskripsi kegiatan proses sistem. Berikut adalah elemen-elemen pada *Activity Diagram*.

Tabel 2.1 Elemen – Elemen Activity Diagram

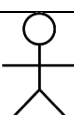
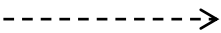
Nama	Simbol	Deskripsi
Action		Untuk menggambarkan perilaku yang sederhana dan bersifat <i>non decomposable</i>
Activity		Untuk mewakili kumpulan aksi (<i>action</i>)
Object Node		Untuk mewakili <i>object</i> yang terhubung dengan kumpulan <i>object flow</i>
Control Flow		Menunjukkan rangkaian pelaksanaan
Object Flow		Menunjukkan aliran sebuah objek dari sebuah aktifitas atau aksi ke aktifitas atau aksi lainnya
Initial Node		Menandakan awal dari kumpulan aksi
Final Activity Node		Untuk menghentikan seluruh <i>control flow</i> atau <i>object flow</i> pada sebuah aktifitas
Final Flow Node		Untuk menghentikan <i>control flow</i> atau <i>object flow</i> aktifitas tertentu
Decision Node		Untuk mewakili suatu kondisi pengujian yang bertujuan untuk memastikan <i>control flow</i> atau <i>object flow</i> menuju ke satu arah

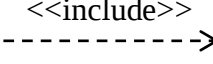
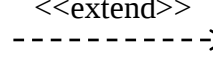
Merge Node		Untuk menyatukan kembali <i>decision path</i> yang dibuat dengan menggunakan <i>decision node</i>
------------	---	---

2. Use case diagram

Diagram ini menggambarkan teknik yang digunakan dalam pengembangan perangkat lunak atau sistem informasi untuk persyaratan fungsional sistem. Use Case menjelaskan interaksi yang terjadi antara 'aktor' - penggagas langkah-langkah sederhana seiri segan segan segan segan segan segan segan yu. Berikut adalah elemen-elemen dalam diagram Use Case.

Tabel 2.2 Elemen – Elemen Use Case Diagram

Nama	Simbol	Deskripsi
Actor		Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i>
Dependency		Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri
Generalization		Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data

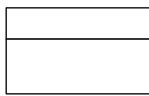
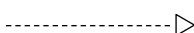
		dari objek yang ada di atasnya objek induk (ancestor)
Include		Menspesifikasikan bahwa use case sumber secara <i>eksplisit</i>
Extend		Menspesifikasikan bahwa use case target memperluas perilaku dari use case sumber pada suatu titik yang diberikan
Association		Apa yang menghubungkan antara objek satu dengan objek lainnya
System		Menspesifikasikan paket yang menampilkan sistem secara terbatas
Use Case		Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor
Collaboration		Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya
Note		Elemen fisik yang eksis

		saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputansi
--	---	--

3. Class diagram

Diagram ini menggambarkan struktur koleksi objek dengan dan dengan struktur umum, perilaku umum, hubungan umum dan semantik / kata-kata umum. *Class* ditentukan oleh lokasi objek dalam urutan dan diagram kolaborasi. Berikut ini adalah elemen diagram Kelas.

Tabel 2.3 Elemen – Elemen Class Diagram

Nama	Simbol	Deskripsi
Generalization		Hubungan dimana objek anak (<i>descendant</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>)
Nary Association		Upaya untuk menghindari asosiasi dengan lebih dari 2 objek
Class		Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama
Collaboration		Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor
Realization		Operasi yang benar-benar dilakukan oleh suatu objek

Dependency	----->	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri
Association	_____	Apa yang menghubungkan antara objek satu dengan objek lainnya

2.12 SHA3-256

SHA-3 ini dikenal sebagai Keccak yang dirancang oleh Guido Bertoni, Joan Daemen, Peeters Michael, dan Gilles Van Assche. SHA-3 atau Keccak ini merupakan pemenang dari kompetisi fungsi hash NIST. SHA-3 ini menggunakan sponge construction dimana blok pesan XOR ke dalam bit awal state yang kemudian dilakukan permutasi, dimana input ke state hash pada tingkatan tertentu akan menghasilkan output yang mempunyai tingkatan yang sama. SHA-3 terdiri dari 5 x 5 array 64-bit kata, 1600 bit total.

SHA membutuhkan 5 buah penyangga (buffer) yang masing-masing panjangnya 32 bit (MD5 hanya mempunyai 4 buah penyangga). Total panjang penyangga adalah $5 \times 32 = 160$ bit. Keempat penyangga ini menampung hasil antara dan hasil akhir. Kelima penyangga ini diberi nama A, B, C, D, dan E. Setiap penyangga diinisialisasi dengan nilai-nilai (dalam notasi HEX) sebagai berikut:

A = 67452301

B = EFCDAB89

C = 98BADCFE

D = 10325476

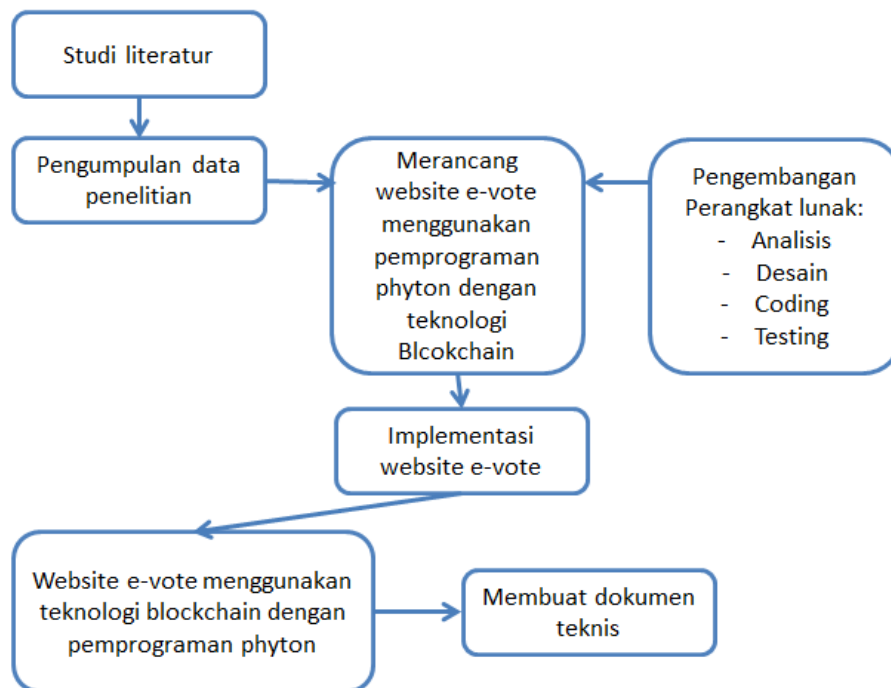
E=C3D2E1F0

BAB III

METODOLOGI PENELITIAN

3.1 Desain Penelitian

Desain penelitian akan digambarkan pada kerangka kerja seperti gambar 3.1 dibawah ini :



Gambar 3.1 Kerangka Kerja Penelitian

Pada gambar 3.1 diatas dijelaskan bagaimana perancangan aplikasi dimulai, yaitu :

1. Studi literatur, pada tahap ini penulis mengumpulkan data dari berbagai macam sumber untuk menjadi refrensi penulisan dan contoh aplikasi yang akan dibuat, kebanyakan sumber berasal dari jurnal dan internet.

2. Pengumpulan data penelitian, pada tahap ini penulis mulai mengumpulkan data yang sudah dicari pada berbagai sumber, dan merangkumnya untuk melanjutkan pada tahap rancangan dan penulisan.
3. Merancang aplikasi, pada tahap ini penulis mulai merancang aplikasi, pada awal mendesain pada mockup, coding hingga hasil jadi, tahap ini didukung oleh metode waterfall.
4. Pengembangan perangkat lunak, pada tahap merancang pengembangan perangkat lunak menggunakan metode waterfall yang akan dijelaskan pada metode penelitian sub bab 2.
5. Implementasi, pada tahap ini penulis telah mengimplementasikan website ini dengan cara menggunakannya dan mencoba melihat kekurangan pada aplikasi sebelum pada tahap hasil akhir.
6. Pada tahap ini, website yang penulis rancang sudah jadi dan siap digunakan, aplikasi juga terus dikembangkan selaras dengan penggunaan aplikasi sesuai kebutuhan aplikasi, mulai dari fitur hingga tampilan.
7. Pada tahap dokumen teknis ini, penulis membuat laporan mengenai jalannya aplikasi seperti screenshot aplikasi dan pengujian blackbox.

3.2 Metode Penelitian

Adapun metode penelitian ini dapat digambarkan sesuai dengan yang telah dirancang pada gambar 3.1 yang terdiri atas :

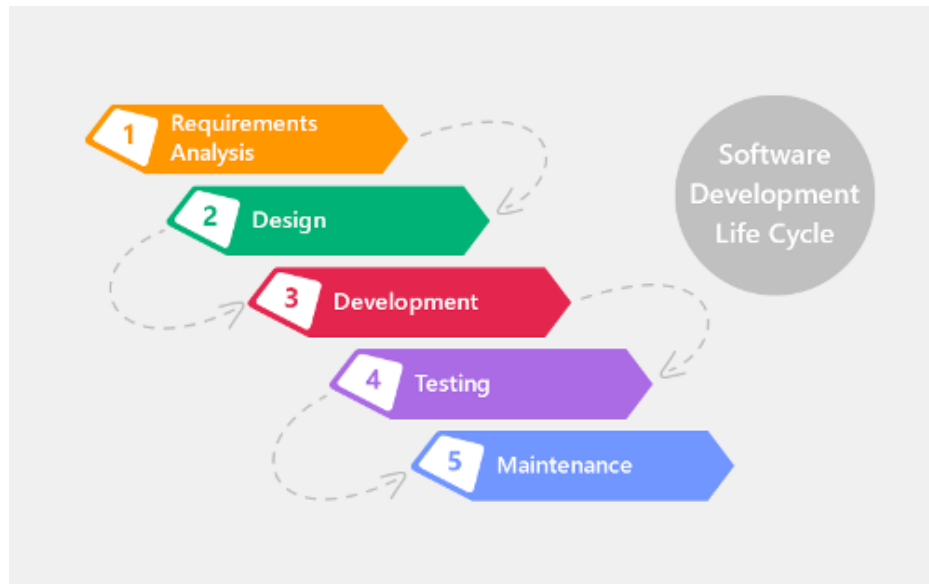
1. Metode Pengumpulan Data

Metode pengumpulan data yang dipakai pada penelitian untuk menyusun aplikasi ini adalah metode studi literatur.

1. Metode literatur, mencari teori-teori pendukung mengenai aplikasi yang dibangun. Teori-teori tersebut dapat dicari melalui buku, jurnal ataupun dari internet.

2. Metode Pengembangan Perangkat Lunak

Model proses yang digunakan dalam pembangunan sistem ini adalah model *waterfall*, yang melingkupi aktivitas sebagai berikut :



Gambar 3.2 Model *Waterfall*

Adapun langkah-langkahnya sebagai berikut:

1. Perencanaan (*planning*) Mengenai kebutuhan pengguna (user's specification), studi-studi kelayakan (*feasibility study*) baik secara teknik maupun secara teknologi serta penjadwalan suatu proyek sistem informasi atau perangkat lunak. pada tahap ini pula, sesuai dengan kakas (tool) yang penulis gunakan yaitu UML.
2. Analisis (*analysis*) Yaitu tahap dimana kita berusaha mengenai segenap permasalahan yang muncul pada pengguna dengan mendekomposisi dan merealisasikan *use case* diagram lebih lanjut, mengenai komponen-komponen sistem atau perangkat lunak, objek-objek, hubungan antar objek dan sebagainya.

3. Perancangan (*design*) Mencari solusi dari permasalahan yang didapat dari tahap analisis dan menetapkan form atau bentuk dari aplikasinya.
4. Implementasi
Mengimplementasikan perencanaan sistem ke situasi nyata yaitu dengan pemilihan perangkat keras dan penyusunan perangkat lunak aplikasi (pengkodean/*coding*).
5. Pengujian (*testing*)
Menentukan apakah sistem atau perangkat lunak yang dibuat sudah sesuai dengan kebutuhan pengguna atau belum, jika belum, proses selanjutnya adalah bersifat iteratif, yaitu kembali ke tahap-tahap sebelumnya. Dan tujuan dari pengujian itu sendiri adalah untuk menghilangkan atau meminimalisasi cacat program (*defect*) sehingga sistem yang dikembangkan benar-benar akan membantu para pengguna saat mereka melakukan aktivitas-aktivitasnya.
6. Pemeliharaan (*maintenance*)
Proses pemeliharaan sistem dan jika diperlukan melakukan perbaikan-perbaikan kecil.

3.3 Alat dan Bahan Penelitian

3.3.1 Alat Penelitian

Pada penelitian ini digunakan alat penelitian berupa perangkat keras dan perangkat lunak sebagai berikut :

Komputer dengan spesifikasi :

1. Processor Intel core i3 , up to 2.53 Ghz
2. RAM 2gb up to 4gb
3. Harddisk 500Gb SATA
4. Monitor dengan resolusi 1366 x 768
5. Mouse keyboard

Perangkat Lunak :

1. Operating system : Windows 10 Pro 64-bit
2. Software pendukung : MySQL, Python 3.8, Django, PyCryptodome
3. Bahasa Pemrograman Python
4. *Software* utama : *pycharm*

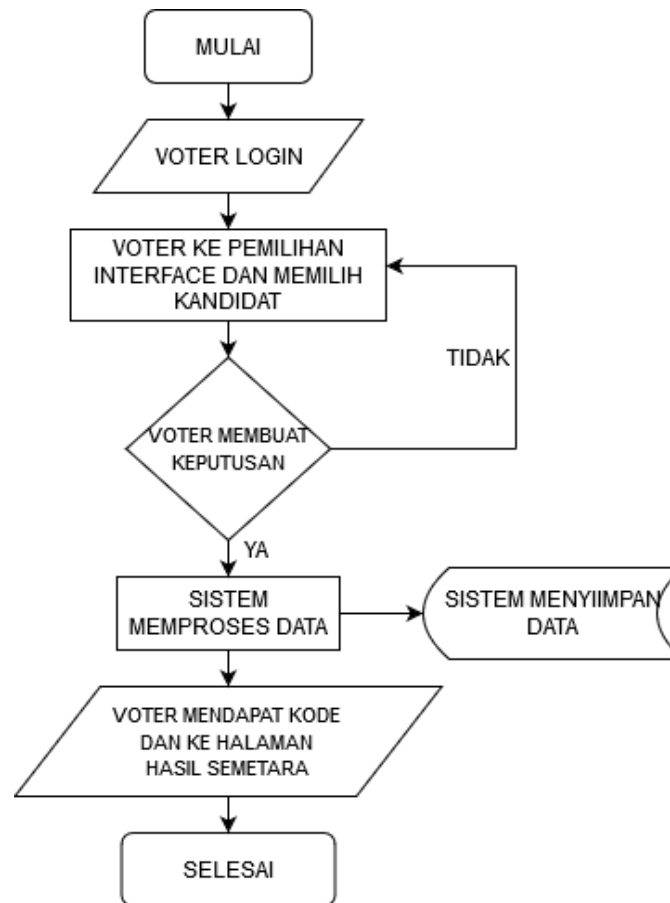
3.3.2 Bahan Penelitian

Bahan-bahan penelitian berupa data pemrograman berupa *website* dengan fitur keamanan *Blockchain* yang dimodifikasi agar dapat digunakan pada *website e-vote*. Data juga dapat berupa modeling dari sistem *e-vote* yang sudah ada dan dikombinasikan. Data-data ini digunakan sebagai dasar kasus dalam pembuatan aplikasi ini.

3.4 Analisa dan Perancangan Aplikasi

Analisis dan Perancangan Sistem merupakan penguraian dari suatu sistem yang utuh ke dalam bagian-bagian untuk mengidentifikasi dan mengevaluasi permasalahan sehingga dapat membangun sistem yang baik. Komponen-komponennya antara lain *flowchart*, *use case diagram* dan *diagram activity*.

3.4.1 Flowchart Diagram

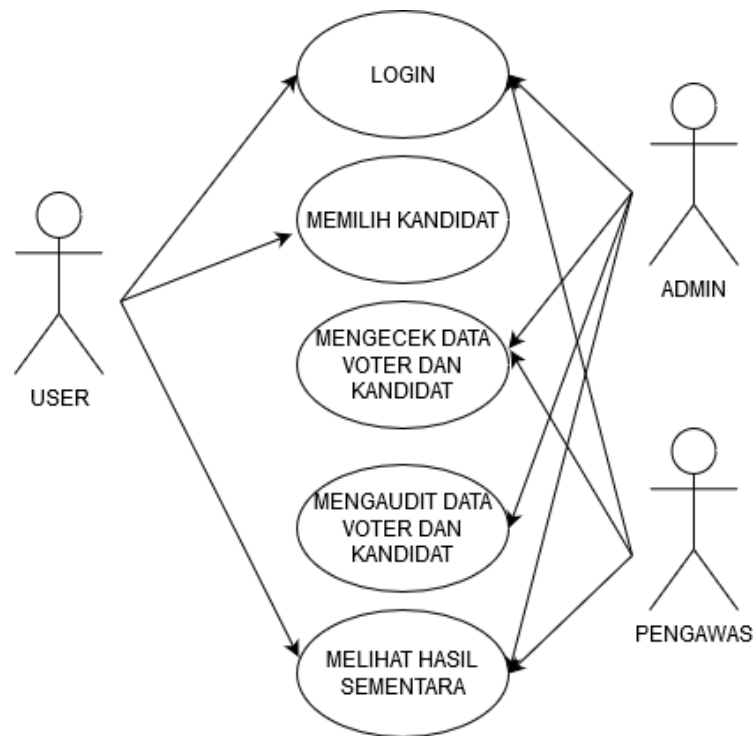


Gambar 3.3 Flowchart Diagram

Pada gambar 3.3 diatas, Proses dari sistem aplikasi yang akan berjalan nantinya yaitu dimulai dari login voter, lalu voter akan diarahkan ke menu *interface* pilihan kandidat, kemudian voter harus memilih salah satu dari kandidat, setelah itu sistem akan memproses data dari voter dan pilihan yang dipilih dan menyimpannya dengan sistem *Blockchain*, setelah itu voter akan mendapat kode transaksi dari sistem untuk validasi dan voter dapat melihat hasil sementara pada laman hasil sementara.

3.4.2 Use Case Diagram

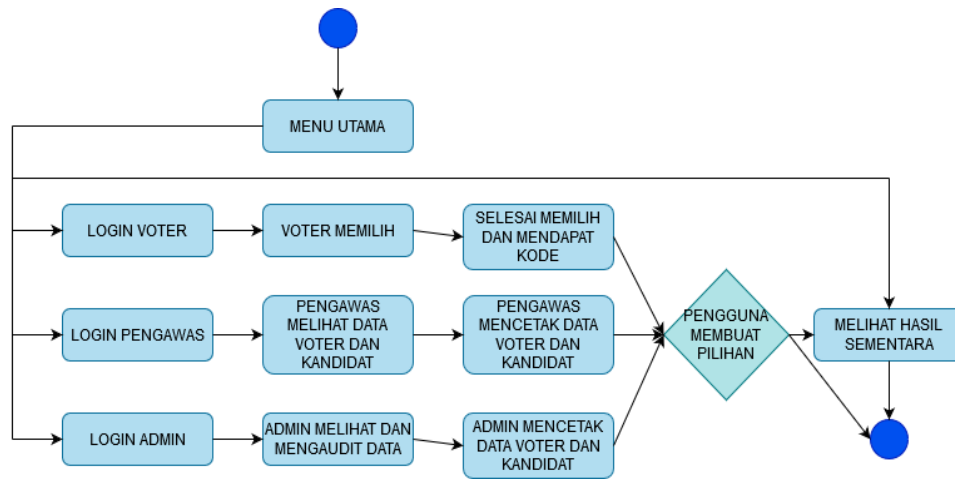
Use case diagram pada aplikasi adalah sebagai berikut :



Gambar 3.4 Use Case Diagram

Pada gambar 3.4 diatas *voter* dapat melakukan login, memilih kandidat, dan juga melihat hasil sementara. Untuk pengawas dapat melakukan login, mengecek dan verifikasi data dari *voter* dan juga melihat hasil sementara. Dan untuk admin dapat login, mengaudit data *voter* dan kandidat, lalu melihat hasil sementara.

3.4.3 Activity Diagram



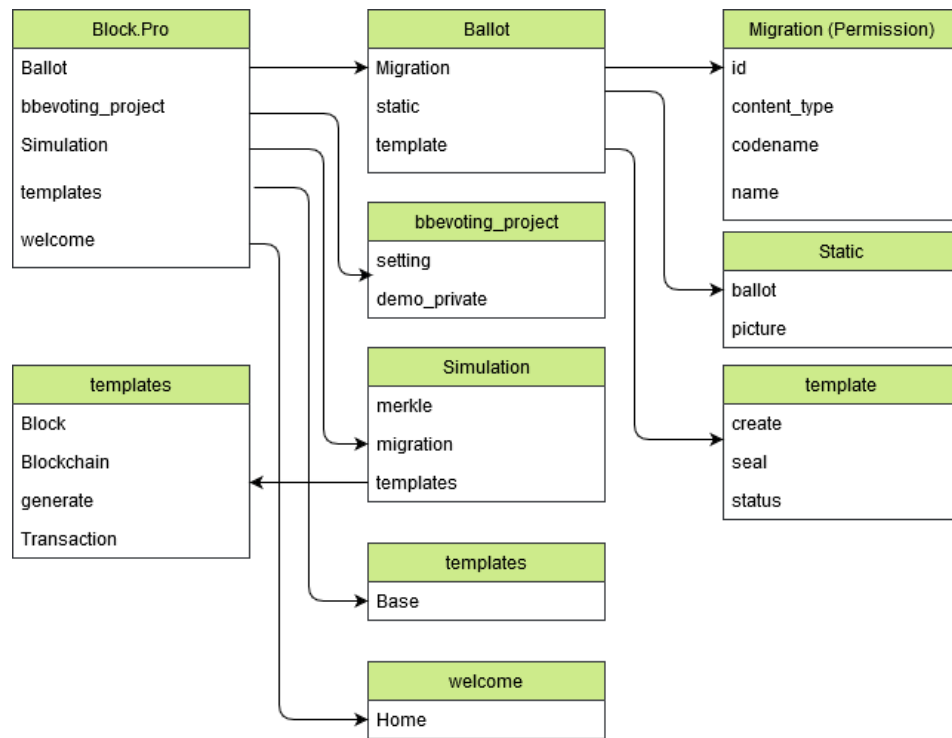
Gambar 3.5 Activity Diagram

Pada gambar 3.5 diatas, disajikan bagaimana aplikasi berjalan dari mulai menu utama, lalu pengguna dapat menggunakan fitur login sesuai apa status mereka, jika mereka *voter* maka mereka dapat memilih kandidat lalu mendapatkan kode, setelah itu *voter* bisa melihat hasil sementara dari *voting*.

Sedangkan jika status mereka admin, maka dapat melakukan login, mengaudit data voter dan kandidat, mencetaknya, dan dapat melihat hasil sementara. Berbeda dengan admin, pengawas hanya bisa melihat data *voter* dan kadidat tanpa bisa merubah isinya, pengawas dapat mencetak data dan melihat hasil sementara juga.

3.4.4 Django Class Diagram

Class diagram dari aplikasi web e-vote ini adalah sebagai berikut:



Gambar 3.6 Class Diagram Django Project

Pada gambar 3.6 diatas disajikan bagaimana struktur dari project website e-vote ini dalam bentuk class diagram, project ini menggunakan framework django sebagai environment nya, dalam project ini terdapat struktur sebagai berikut:

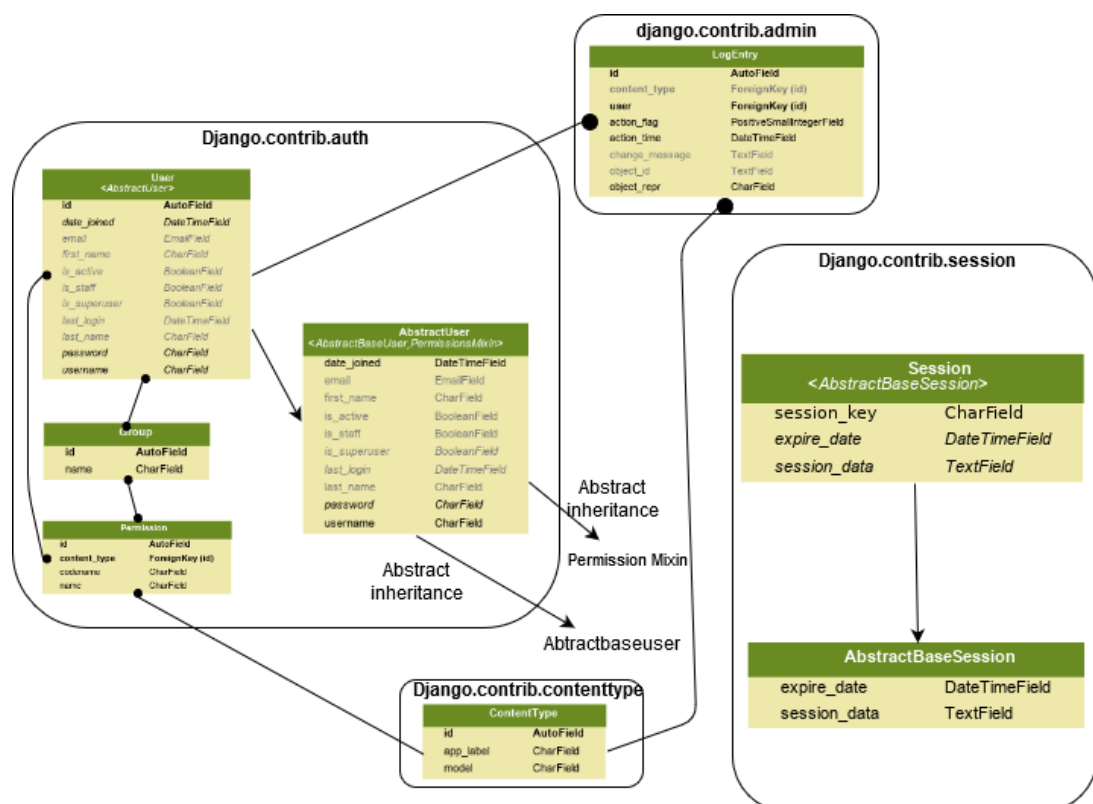
1. **Block.Pro** adalah struktur utama dari website e-voting ini, isi dari project ini antara lain yaitu **Ballot**, **bbevoting_project**, **simulation**, **templates**, dan **welcome**.
2. **Ballot** disini adalah sebuah folder yang mengatur migration database, **static** yang berisi data gambar dari kandidat, dan **picture**, sebagai data gambar cadangan.
3. **Migration (permission)**, adalah tempat dimana program untuk migration data voter yang berisi **id**, **codename**, **content_type**, dan **name**.
4. **Bbvoting_project**, adalah tempat dimana settingan dari aplikasi django ini diatur seperti banyak data dan juga banyak block yang akan dibuat,

dan berisi data private key yang akan dipakai sistem untuk enkripsi dan dekripsi data.

5. Simulation, adalah folder project dimana data program untuk merkle tree, migration dan templatanya dibuat disini, dan didalam template ada data website interface seperti halaman Block, Program Blockchain, Generating signature, dan halaman transaksi.
6. Templates, adalah suatu struktur halaman utama dari website yang didalamnya terdapat file berupa base untuk menu halaman utama.
7. Welcome, adalah page dari awal menu utama yang ditimpa dengan halaman templates, file ini adalah bagian dari halaman menu utama website itu sendiri.

3.4.5 Class Diagram SQLite

Class diagram dari database SQLite pada aplikasi ini adalah sebagai berikut:



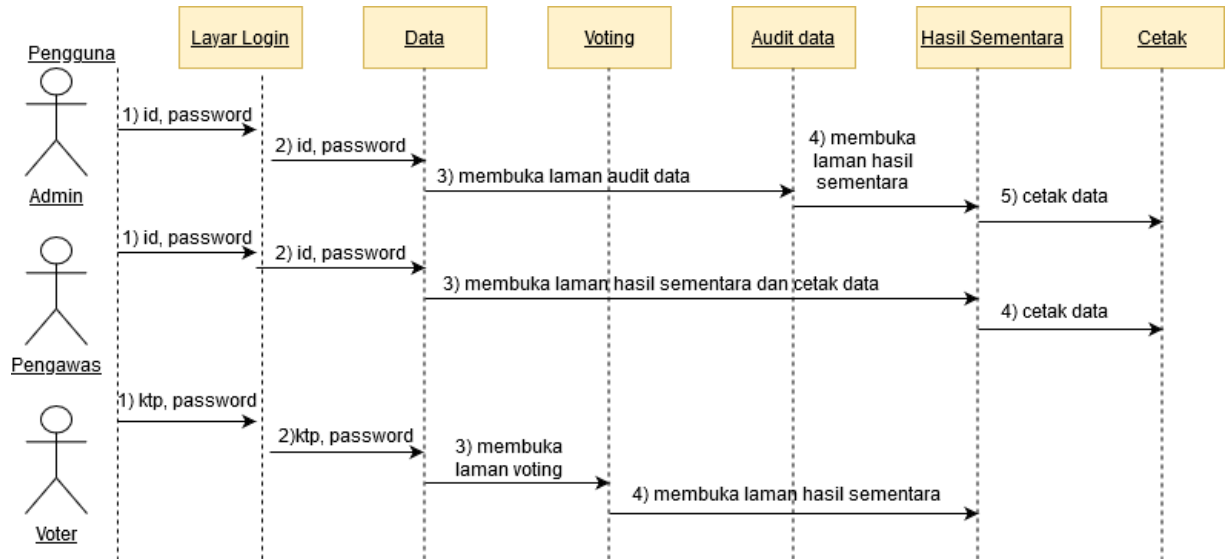
Gambar 3.7 Class Diagram SQLite

Pada Gambar 3.7, disajikan gambar mengenai class diagram dari database sqlite yang digunakan dalam aplikasi website e-vote ini, isi dari class diagram sqlite ini adalah sebagai berikut:

1. Django.contrib.admin yaitu berisi database log.entry yang berisi data berupa id, content_type, user, action_flag, action_time, chane_message, object_id dan object_repr. Django contrib.admin ini berhubungan dengan django.contrib.auth dan django.contrib.contenttype.
2. Django.contrib.auth yaitu berisi database otentifikasi user dengan data id, date_joined, email, first_name, is_active, is_staff, is_superuser, last_login, last_name, password, dan username. Pada tahap ini, contribauth ini berhubungan dengan contenttype dan contribadmin.
3. Django.abstact.user berisi data user seperti pada contrib.auth, data yang dimasukan juga sama karena ini adalah hasil inheritance dari data sebelumnya.
4. Group, adalah database yang berisi otentifikasi dari grup data yang dimasukan dan diproses oleh sistem.
5. Permission, adalah database yang berisi data grup yang sedang diproses oleh sistem untuk melakukan permission data sebelum data diolah ke proses selanjutnya.
6. Django.contrib.contenttype, adalah data konten yang telah diverifikasi oleh sistem pada permission, dan data memiliki model dan label.
7. Django.contrib.session, adalah data dari session pengguna ketika melakukan login atau perpindahan halaman yang memerlukan status login atau diizinkan oleh sistem.

3.4.6 Sequence Diagram

Sequence diagram yang digunakan pada aplikasi website e-vote adalah sebagai berikut:



Gambar 3.8 Sequence Diagram

Gambar 3.8 diatas merupakan rancangan sequence diagram dari aplikasi e-vote tersebut, yang menunjukkan interaksi dari objek-objek dan memberi petunjuk komunikasi dari objek tersebut.

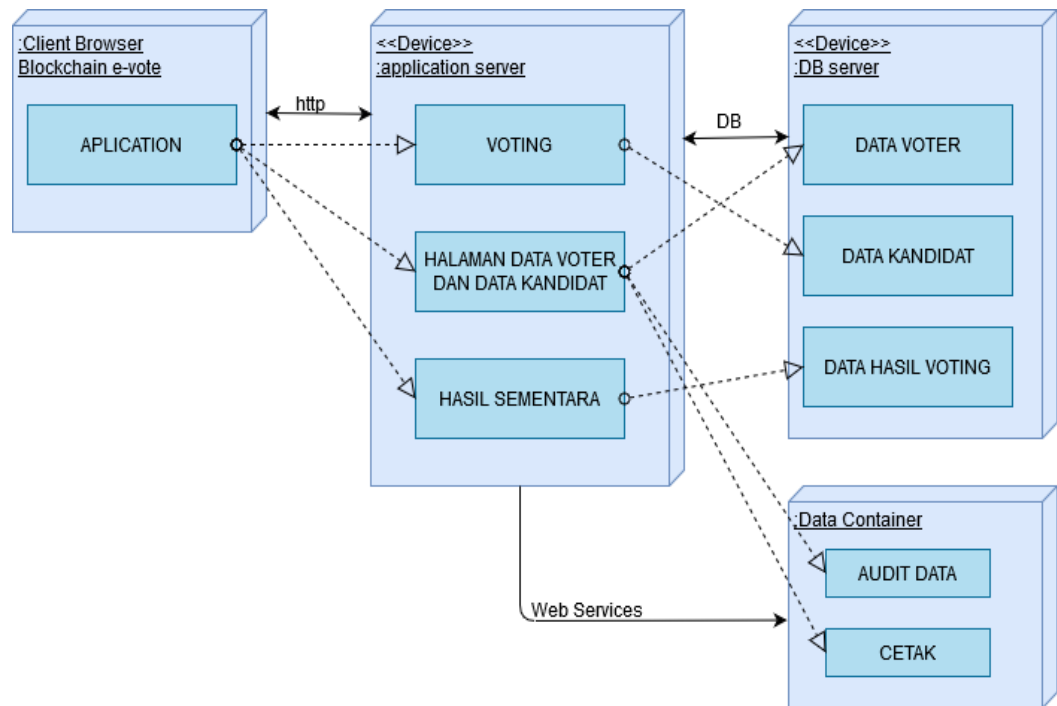
Dalam interaksi yang terjadi pada aplikasi terjadi tiga level pengguna yaitu admin, pengawas, dan voter yang memiliki interaksi berbeda-beda pada objek. Interaksi yang terjadi yaitu sebagai berikut:

1. Admin, pada tahapan pengguna admin akan berinteraksi dengan layar login dengan tahapan data id dan password, lalu interaksi pada data yaitu data login untuk admin id dan password, lalu interaksi dengan audit data dengan membuka laman audit data setelah login, lalu interaksi selanjutnya dengan membuka hasil sementara dan terakhir interaksi dengan cetak data.
2. Pengawas, pada tahapan pengawas akan berinteraksi dengan layar login dengan tahapan data id dan password, lalu interaksi pada data yaitu data login untuk pengawas id dan password, lalu

interaksi dengan hasil sementara dan terakhir interaksi dengan cetak data.

3. *Voter*, pada tahapan voter akan berinteraksi dengan layar login dengan tahapan ktp dan password, lalu interaksi dengan pada data dengan tahapan ktp dan password, lalu interaksi dengan voting dengan memilih kandidat, terakhir interaksi dengan halaman hasil voting sementara.

3.4.7 Deployment Diagram



Gambar 3.9 Deployment Diagram

Pada gambar 3.9 merupakan rancangan dari deployment diagram pada website e-vote. Pada dasarnya deployment diagram ini hanya untuk menggambarkan, menspesifikasikan, dan mendokumentasikan yang terjadi pada aplikasi web dan pada devicenya dan apa saja isinya.

Penjelasan pada deployment diagram diatas adalah sebagai berikut:

1. Client Browser, pada blok ini aplikasi website e-vote berada pada client browser pengguna, aplikasi ini mempunyai relasi dengan device server melalui hosting http dan menghubungkan

dengan laman voting, dengan laman data voter dan data kandidat, dan juga dengan laman hasil sementara.

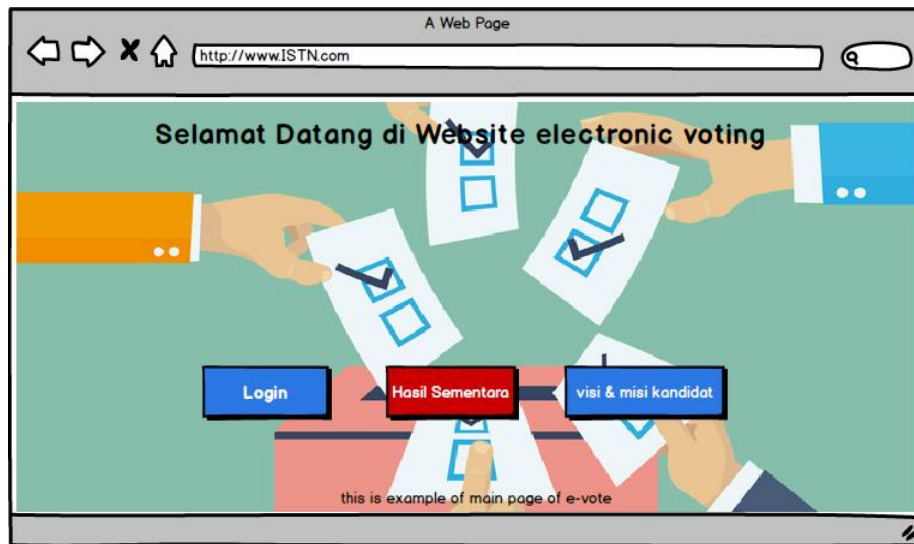
2. Device application server, pada blok ini berisi laman yang dapat dijalankan oleh server pada device, blok ini berisi proyek web itu sendiri, data berupa halaman dari website e-vote yaitu voting, halaman data voter dan kandidat, lalu halaman hasil sementara. Voting memiliki relasi dengan data kandidat, halaman data voter dan kandidat memiliki relasi dengan data voter, dan hasil sementara memiliki relasi dengan data hasil sementara yang ketiganya dihubungkan dengan koneksi database.
3. Device db server, adalah blok berisi database yang tersimpan pada device sehingga aplikasi memiliki akses terhadap database, data berisi data voter yang memiliki relasi dengan halaman voter dan kandidat, data kandidat yang memiliki relasi dengan laman voting, dan data hasil voting yang memiliki relasi dengan laman hasil sementara.
4. Data container, yaitu adalah fitur mengaudit data yang dimiliki oleh admin dan fitur cetak yang bisa digunakan oleh admin dan pengawas yang berada pada aplikasi dan terhubung pada web service.

3.5 Perancangan Antar Muka (*User Interface*)

Perancangan antar muka aplikasi adalah rancangan dari aplikasi web yang akan dibuat. Dalam web ini akan dibagi menjadi web utama *e-vote* dan *background* proses dari web utama e-vote.

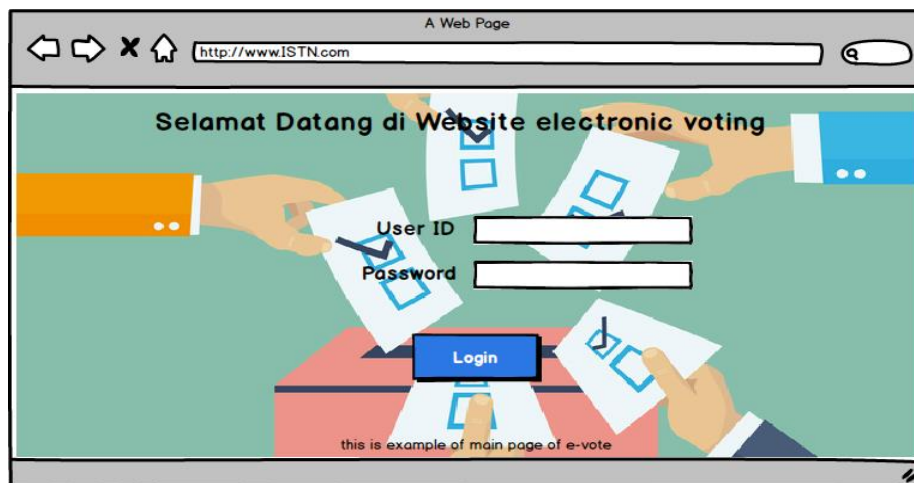
Pada aplikasi terdapat beberapa tampilan yang akan memudahkan pengguna yang dibuat menggunakan *website*. Berikut penjelasan dan gambaran tampilan aplikasinya.

3.5.1 Antar Muka Web Utama (Voter)



Gambar 3.10 Tampilan menu utama

Pada gambar 3.10 merupakan tampilan menu utama dari aplikasi *e-vote*, pada tampilan ini terdapat pilihan login, hasil sementara, visi & misi kandidat. Pada pilihan login dibagi menjadi tiga bagian yaitu, login untuk *voter*, untuk admin, dan untuk pengawas.



Gambar 3.11 Tampilan Login

Pada gambar 3.11 diatas, pengguna aplikasi dapat login sebagaimana status mereka, *voter*, admin, atau pengawas. Data yang dimasukan berupa *User ID* dan *Password*. Pada tahap user ini, voter hanya perlu memasukan nomer ktp dan password.



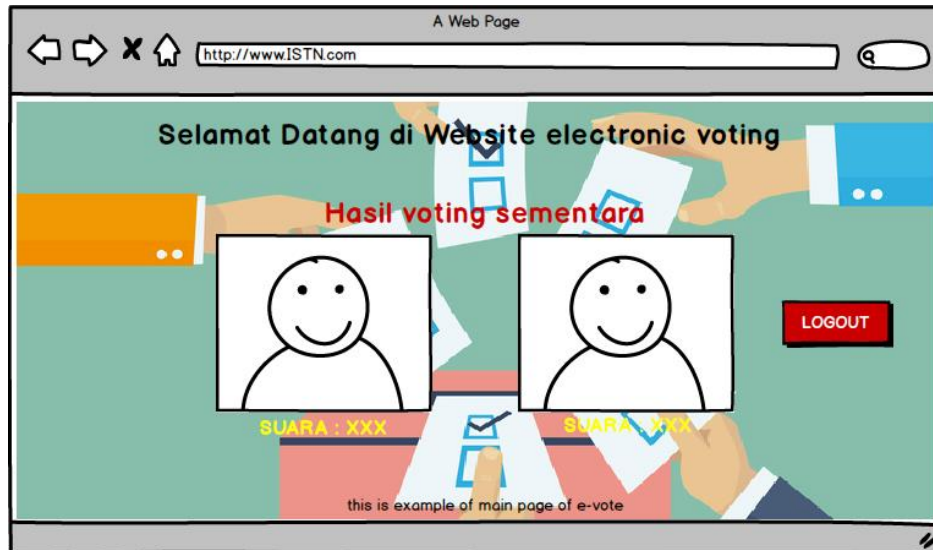
Gambar 3.12 Tampilan memilih kandidat

Pada gambar 3.12 diatas *voter* dapat memilih kandidat yang disediakan dengan memilih tombol pilih yang ada dibawah gambar mereka.



Gambar 3.13 Tampilan selesai memilih

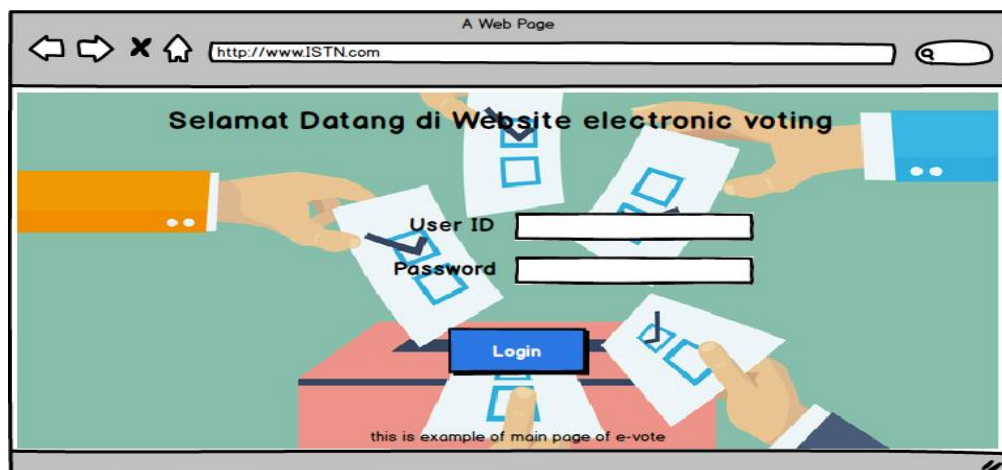
Pada gambar 3.13 diatas, voter telah selesai memilih kandidat, lalu mendapat kode transaksi yang harus disimpan, voter lalu dapat melihat hasil voting sementara.



Gambar 3.14 Tampilan hasil voting sementara

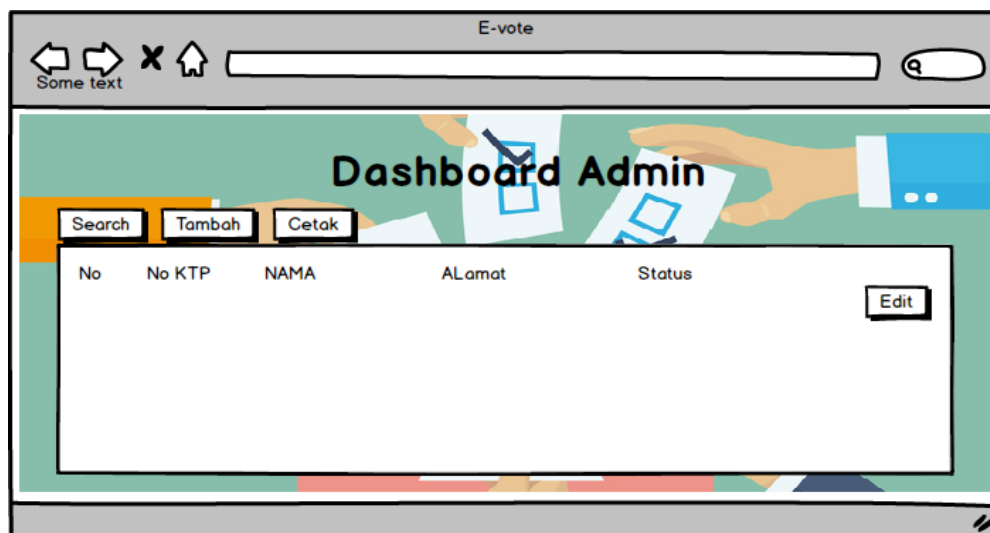
Pada gambar 3.14 diatas, voter maupun pengawas dan admin dapat melihat data hasil voting sementara dari kandidat yang dipilih. Data dari hasil pemilihan tersebut akan disimpan dengan metode blockchain dan tidak dapat diubah.

3.5.2 Antar Muka Web Utama (Admin)



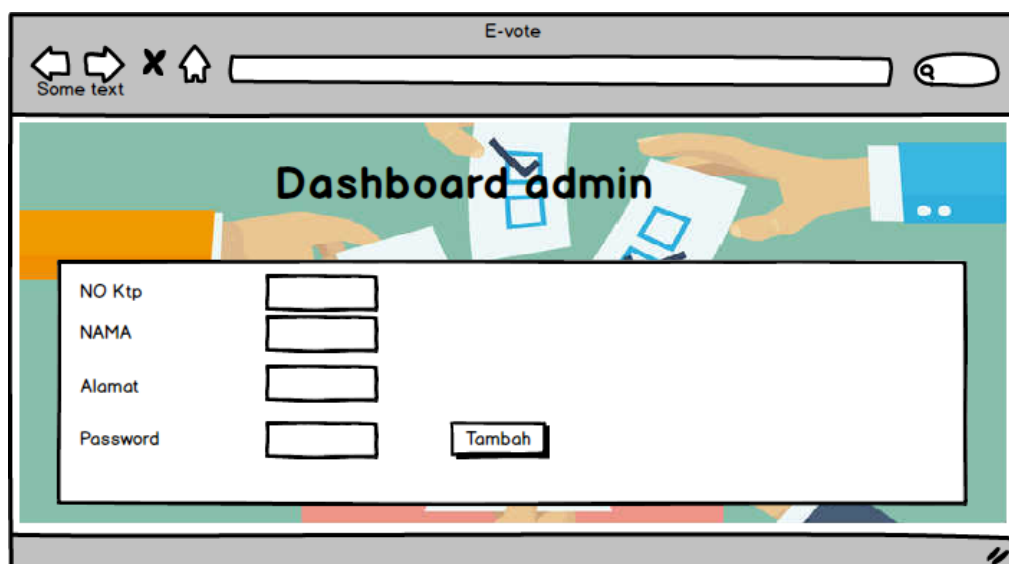
Gambar 3.15 Tampilan Login Admin

Pada gambar 3.15, admin dapat login pada halaman tersebut untuk masuk ke panel milik admin, isi dari admin ID dan Passwordnya tidak menggunakan ktp, admin dapat bebas memilih ID dan passwordnya.



Gambar 3.16 Tampilan Dashboard admin

Pada gambar 3.16, halaman utama admin ini admin dapat mengaudit data *voter* maupun kandidat, pada halaman ini terdapat fitur *search* atau cari data, Tambah berupa tambah data, dan cetak berupa cetak data. Admin dapat mengaudit data pada tombol edit.



Gambar 3.17 Tampilan Tambah data *voter*

Pada gambar 3.17, admin dapat melakukan tambah data *voter* yang tertera pada laman tersebut, admin dapat memasukan data berupa nomer ktp, nama, alamat, dan *password*. *Password* disini dapat diganti oleh *voter* jika ingin menggantinya atau jika ada masalah login.

Karena data voter harus real teridentifikasi oleh banyak pihak termasuk pengawas, maka penjagaan pada data voter yang dimasukan secara tidak sah oleh pihak oknum akan tidak valid, data yang tidak valid ini akan di cek pada pihak lain dan akan dihapus dan dibuat ulang dengan data yang benar.

The screenshot shows a web browser window titled 'E-vote'. The address bar contains 'Some text'. The main content area is titled 'Dashboard admin' and features a form for adding candidate data. The form has the following fields and labels:

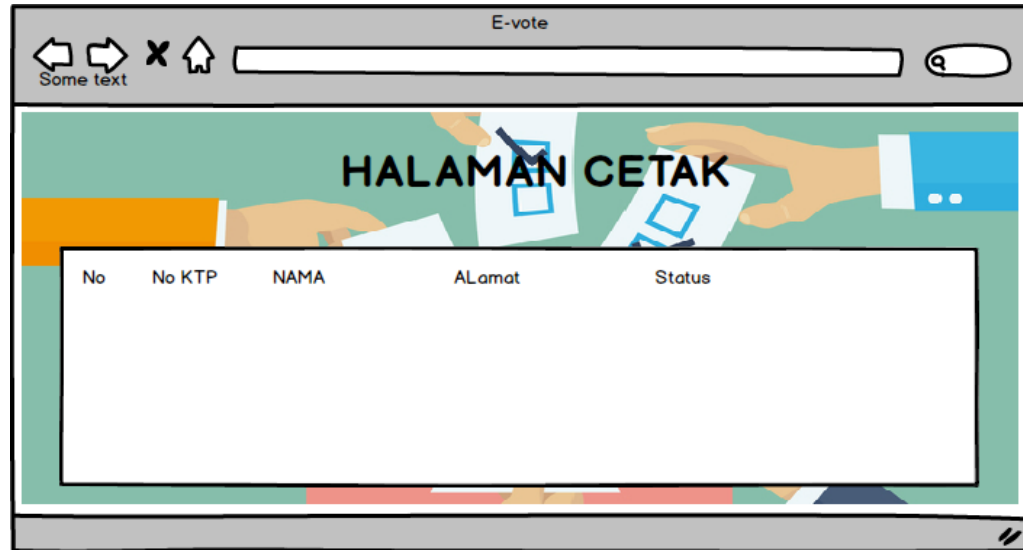
NO	
NAMA	
Nomer Kandidat	
Gambar	
Keterangan	

Below the form is a button labeled 'Tambah'.

Gambar 3.18 Tambah data kandidat

Pada gambar 3.18, admin juga dapat menambahkan data kandidat yang akan dipilih oleh *voter*, data yang dapat ditambahkan berupa no kandidat, nama, gambar dari kandidat, dan keterangan dari kandidat.

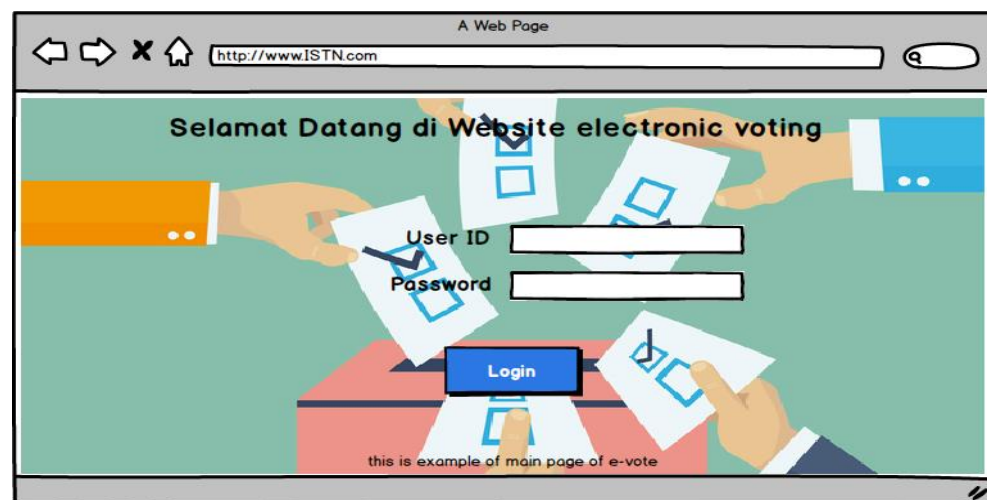
Pada tahap ini juga admin akan menambah data voter yang sudah diverifikasi oleh berbagai pihak yang menyelenggarakan pemilihan umum ini, agar tidak ada data yang tidak valid ditambahkan oleh admin. Didalam databasenya pun sudah ada data tanda terima untuk melakukan verifikasi oleh admin atau pengawas dan pihak lainnya untuk mengecek data valid atau tidak.



Gambar 3.19 Halaman Cetak

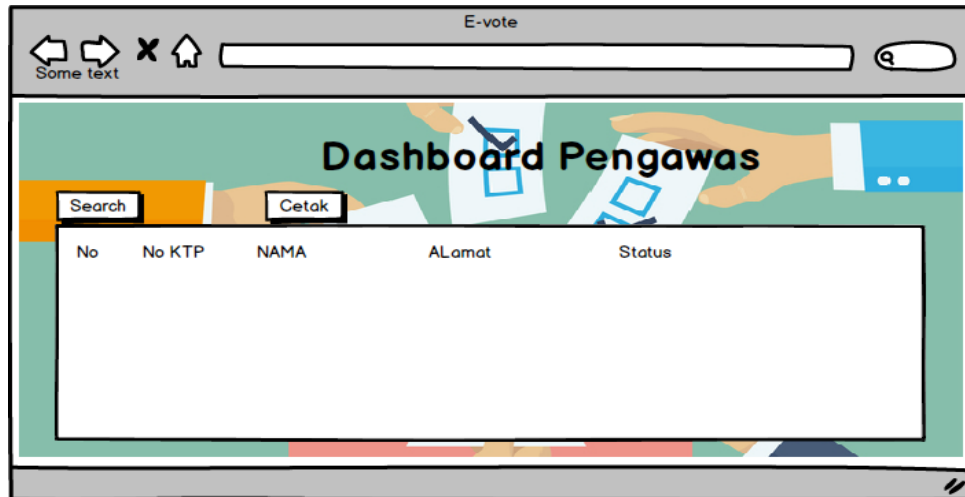
Pada gambar 3.19, admin dapat mencetak data *voter* maupun kandidat kedalam bentuk kertas, hasil cetakan ke dalam bentuk tabel yang berisi data-data *voter* maupun kandidat, data kertas ini juga sebagai bukti fisik dari penyelenggaraan pemilihan umum.

3.5.3 Antar Muka Web Utama (Pengawas)



Gambar 3.20 Login Pengawas

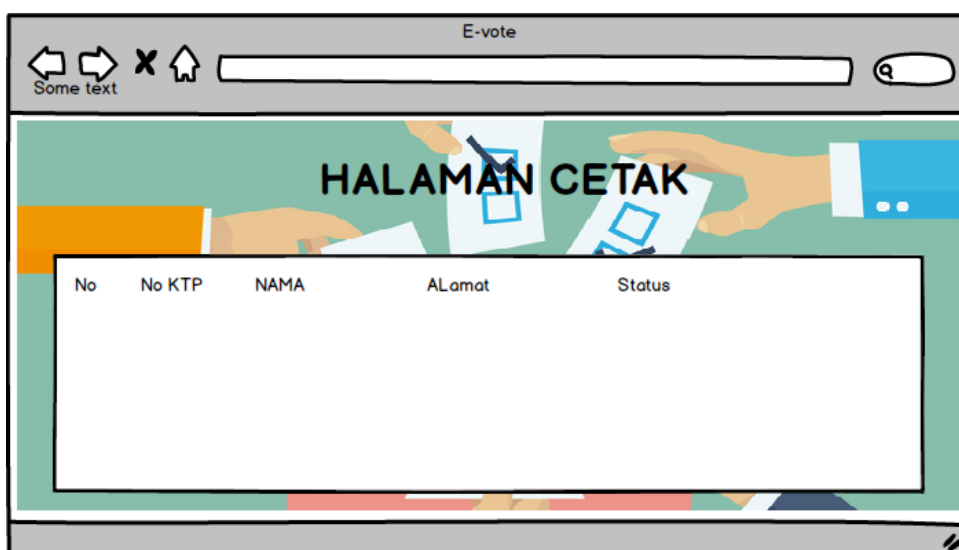
Pada gambar 3.20 pengawas dapat login untuk masuk kedalam dashboard pengawas, pengawas juga tidak menggunakan nomer ktp untuk id, jadi dapat bebas memilih id yang ingin dimasukan.



Gambar 3.21 Dashboard Pengawas

Pada gambar 3.21, pengawas dapat melihat data para *voter* dan kandidat di dashboardnya, hanya saja pengawas tidak dapat akses untuk mengaudit data *voter* dan kandidat, namun hanya bisa mencetak datanya saja sebagai bukti fisik.

Pengawas juga dalam fungsinya sebagai orang yang dipercaya pada penyelenggaraan pemilihan umum ini juga dapan melihat database dari voter yang sudah memilih untuk memvalidasi data voter yang sudah memilih kandidat agar data real dan tidak dimanipulasi oleh orang lain. Data bisa dilihat dari hasil transaksi yang sudah terenripsi

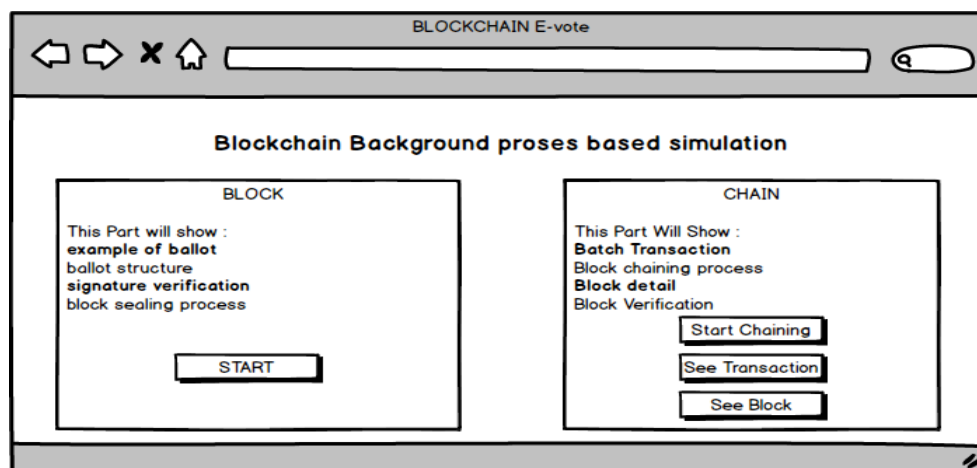


Gambar 3.22 Halaman Cetak Pengawas

Pada gambar 3.22, pengawas dapat melakukan cetak pada data *voter* dan kandidat, bentuk dari hasil cetak berupa tabel yang berisi data *voter* dan kandidat seperti nomer ktp, nama, alamat dan data kandidat berupa nomer kandidat, nama, gambar, dan keterangan.

3.5.4 Antar Muka Web Background Proses

Aplikasi web ini akan menampilkan semua proses pendataan dengan menggunakan metode *blockchain* dikarenakan aplikasi web utama tidak dapat melihat proses *blockchain* terjadi, maka aplikasi ini dibuat.



Gambar 3.23 Tampilan Menu Blockchain background proses

Pada gambar 3.23 diatas adalah tampilan utama dari web, disini pengguna dapat melihat bagaimana cara *block* dibuat, bagaimana proses *chaining* terjadi.

The screenshot shows a web browser window titled "BLOCKCHAIN E-vote". The main heading is "Pemilu Presiden". Below it, there is a field for "YOUR ID :". Under the heading "Kandidat", there are three candidate icons labeled 1, 2, and 3. Below the candidates, there is a field for "Pilihan kandidat" and a field for "Private Key". At the bottom, there is a "SEAL" button.

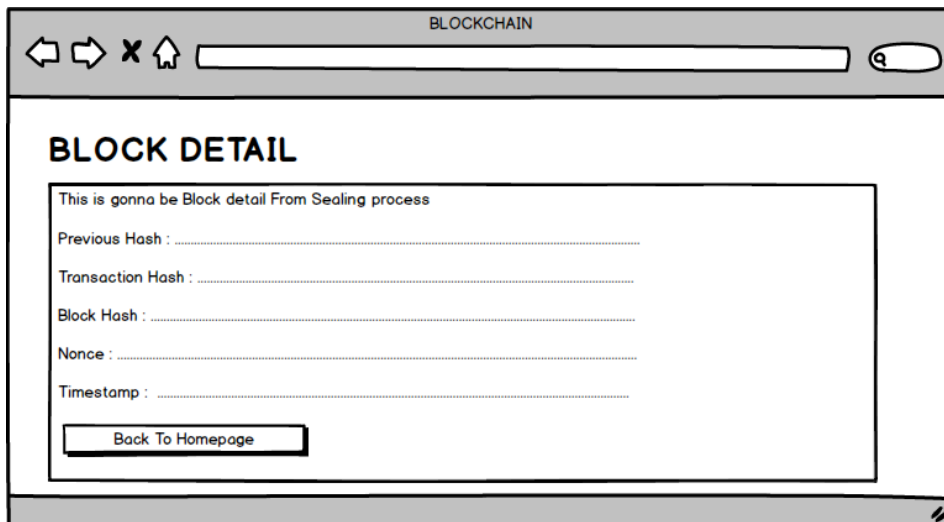
Gambar 3.24 Tampilan simulasi block dibuat

Pada gambar 3.24 diatas pengguna dapat melihat bagaimana cara web membuat *block* dari sebuah data yang dienkripsi dengan memasukan data yang telah dienkripsi tersebut, kandidat, dan *private key* pada pilihan lalu dapat submit datanya.

The screenshot shows a web browser window titled "BLOCKCHAIN". The main heading is "SIGNATURE VERIFICATION". Below it, there is a text box containing the message "This is gonna be ballot information with signature verification detail". Inside the text box, there are two lines of text: "Your Ballot was :" and "Generated Signature :". Below the text box, there is a "Seal The Ballot" button.

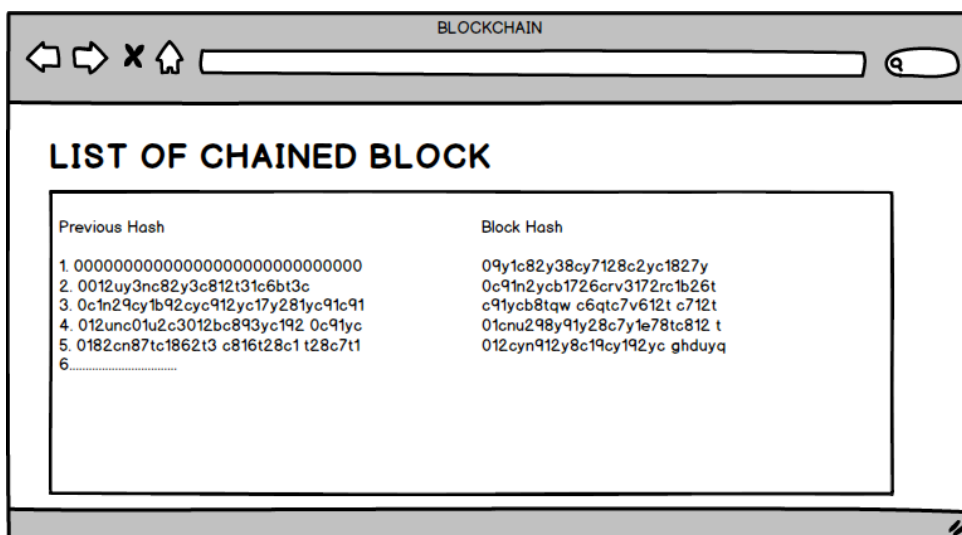
Gambar 3.25 Tampilan setelah data enrkipsi dibuat

Gambar 3.25 menunjukkan bahwa data yang disimpan tadi sudah dienkripsi dan ada data tanda tangan yang merupakan kode yang diberikan pada *voter*, kode ini harus disimpan atau diingat untuk melakukan validasi data.



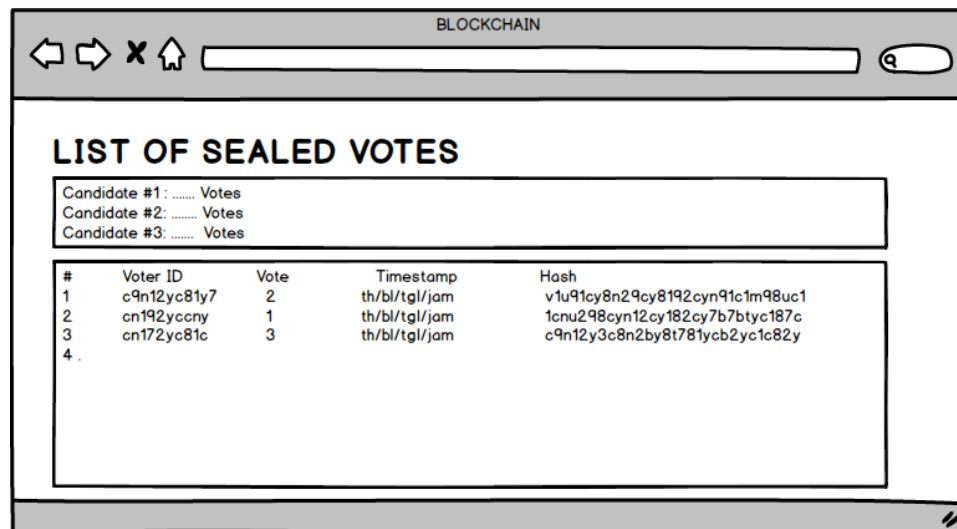
Gambar 3.26 Tampilan detail dari block yang dibuat

Gambar 3.26 diatas adalah hasil enckripsi dari proses membuat *block* yang berasal dari data *voter* yang sudah memilih kandidat, data yang akan ditampilkan nanti berupa *hash*.



Gambar 3.27 tampilan block yang terantai

Gambar 3.27 adalah tampilan setelah proses chaining bekerja pada web dan data tersebut sudah terantai dari satu data ke data yang lain atau yang bisa disebut data *blockchain*.



LIST OF SEALED VOTES

Candidate #1: Votes
 Candidate #2: Votes
 Candidate #3: Votes

#	Voter ID	Vote	Timestamp	Hash
1	c9n12yc81y7	2	th/bl/tgl/jam	v1u91cy8n29cy8192cyn91c1m98uc1
2	cn192ycnny	1	th/bl/tgl/jam	1cnu298cyn12cy182cy7b7btyc187c
3	cn172yc81c	3	th/bl/tgl/jam	c9n12y3c8n2by8t781ycb2yc1c82y
4				

Gambar 3.28 Tampilan data voter yang dienkripsi

Tampilan pada gambar 3.28 memperlihatkan data dari voter yang sudah dienkripsi dan terantai berurutan, data ini berisi data *voter* (Nama, Alamat, Nomer KTP), data yang dipilih *voter* dan timestamp untuk menunjukkan kapan *voter* melakukan pemilihan dan hash dari ketiga data tersebut. Dan data ini berada dalam *block*.

3.6 RANCANGAN DATABASE

Rancangan *database* yang dibuat menggunakan dua jenis *database* yaitu menggunakan MySQL untuk menyimpan data *voter*, kandidat, admin, pengawas dan data kode validasi, dan menggunakan database SQLite untuk autentifikasi, *permission*, ataupun menyimpan data *voter* yang sudah dienkripsi

Rancangan *database* yang digunakan terdapat *database* yang dibuat oleh sistem dan *database* yang dibuat oleh penulis, *database* pada tabel 3.1 sampai 3.13 adalah

buatan sistem namun tetap pada request penulis, dan *database* pada tabel 3.14 sampai 3.17 adalah buatan penulis asli tanpa sangkut paut sistem

Rancangan *database* yang dibuat dapat dilihat pada tabel dibawah ini yaitu :

1. Tabel 3.1 *auth_group* adalah tabel berisi data mengenai id dan nama untuk autentifikasi identitas dari pengguna berupa group.

Tabel 3.1 Tabel *auth_group*

No	Nama Field	Tipe Data	Panjang Data	Keterangan
1	id	integer	11	Primary Key
2	name	varchar	150	

Pada tabel 3.1, adalah data dari *generating* proses *migration* yang dilakukan python pada *framework* django, tabel ini berfungsi untuk akses terhadap grup data yang akan masuk pada tabel data lain, sebelum data masuk, data akan di otentifikasi agar data yang masuk spesifik dan tidak bercampur dengan data yang lain.

2. Table 3.2 *auth_group_permission* adalah tabel berisi data *group* yang sudah mendapat izin untuk pengguna.

Tabel 3.2 Tabel *auth_group_permission*

No	Nama Field	Tipe Data	Panjang Data	Keterangan
1	id	integer	11	Primary Key
2	group_id	integer	11	
3	permission_id	integer	11	

Pada tabel 3.2, *database* pada otentifikasi pertama dan berhasil diotentifikasi akan mendapatkan id berupa grup dan *permission* idnya yang berarti data benar dan bisa melanjutkan ke tahap proses data selanjutnya.

3. Tabel 3.3 *auth_permission* adalah tabel berisi data sesi izin untuk pengguna aplikasi, data hanya tersimpan sementara, jika tidak dipakai akan dihapus.

Tabel 3.3 Tabel auth_permission

No	Nama Field	Tipe data	Panjang Data	Keterangan
1	id	integer	11	Primary Key
2	content_tipe_id	integer	11	
3	codename	varchar	150	
4	name	varchar	150	

Pada tabel 3.3 ini, data yang akan melanjutkan ke proses selanjutnya harus memiliki data individual sendiri, pada *framework* django ini, data untuk *permission single* data berupa id, tipe konten id, kode nama, dan nama dari data itu sendiri.

4. Tabel 3.4 auth_user adalah tabel berisi data user yang memiliki akses terhadap aplikasi, jika user selain dari data ini maka tidak valid.

Tabel 3.4 tabel auth_user

No	Nama Field	Tipe data	Panjang Data	Keterangan
1	id	integer	11	Primary Key
2	password	varchar	128	
3	last_login	datetime	50	
4	is_superuser	bool	50	
5	username	varchar	150	Unique
6	first_name	varchar	150	
7	email	varchar	150	
8	is_staff	bool	50	
9	is_active	bool	50	
10	date_joined	datetime	50	
11	las_name	varchar	150	

Pada tabel 3.4 ini adalah data spesifik dari *user*, yang berarti data yang berada dalam tabel dapat di otentifikasi oleh sistem dan hanya berorientasi pada data *user* saja, data ini adalah data detail pada *framework* django sebagai dasar identitas suatu *user* yang ingin menggunakan *framework* ini.

5. Tabel 3.5 `auth_user_group` adalah tabel untuk menyimpan data id dari user untuk di autentifikasi berapa id yang dimiliki pengguna.

No	Nama Field	Tipe Data	Panjang Data	Keterangan
1	id	integer	11	Primary Key
2	user_id	integer	11	
3	group_id	integer	11	

Tabel 3.5 Tabel `auth_user_group`

Pada tabel 3.5 ini, *user id* dan *group id* digunakan untuk *user* yang sudah berada pada suatu grup data, data ini hanya digunakan untuk memanggil data untuk keperluan pada menu tampilan, contoh untuk penampilkkan data pengguna yang berada pada block untuk ditampilkan pada tabel halaman web.

6. Tabel 3.6 `auth_user_user_permission` adalah tabel untuk menyimpan data permission yang dibutuhkan pada seesion diwebsite ketika sehabis login.

Tabel 3.6 `auth_user_user_permission`

No	Nama field	Tipe Data	Panjang Data	Keterangan
1	id	integer	11	Primary Key
2	user_id	integer	11	
3	permission_id	integer	11	

Tabel 3.6 ini hanya digunakan pada saat kita menuju halaman seperti sehabis login atau ke halaman selanjutnya tanpa meninggalkan identitas dari status apakah sudah otentifikasi atau belum.

7. Tabel 3.7 `django_admin_log` adalah tabel untuk menyimpan laporan dari penggunaan aplikasi yang dilakukan admin atau pengguna.

Tabel 3.7 django_admin_log

No	Nama Field	Tipe Data	Panjang Data	Keterangan
1	id	integer	11	Primary Key
2	action_time	datetime	50	
3	object_id	text	100	
4	object_repr	varchar	150	
5	change_message	text	100	
6	content_type_id	integer	11	
7	user_id	integer	11	
8	action_flag	smallint_unsigned	50	

Tabel 3.7 ini akan menyimpan semua data ketika kita sedang menggunakan aplikasi, data ini hanya disimpan sementara pada suatu *website* khususnya pada *framework* django, agar website memiliki laporan *chace* yang jelas.

8. Tabel 3.8 django_content_type adalah tabel untuk menyimpan data tipe konten dari framework django.

Tabel 3.8 django_content_type

No	Nama Field	Tipe Data	Panjang Data	Keterangan
1	id	integer	11	Primary Key
2	app_label	varchar	100	
3	model	varchar	100	

Pada tabel 3.8, *framework* django juga memilah data yang memiliki model tersendiri dan melabelinya menjadi konten tertentu, tabel ini hanya digunakan ketika menemukan data yang berbeda dari data yang lain.

9. Tabel 3.9 django_migration adalah tabel yang dibuat oleh *framework* django sesuai dengan tabel yang akan kita buat menjadi data.

Tabel 3.9 django_migration

No	Nama Field	Tipe Data	Panjang Data	Keterangan
1	id	integer	11	Primary Key
2	app	integer	11	
3	name	varchar	150	
4	applied	datetime	50	

Pada tabel 3.9, sebelum data yang akan dibuat oleh django, data yang sudah dirancang akan di migrasi dalam sistem untuk dibuat menjadi *database* SQLite, data yang akan diubah akan disusun dengan id, app, name dan applied pada tabel tersebut agar pembuatan data berurutan.

10. Tabel 3.10 *django_session* adalah tabel untuk menyimpan data session pengguna untuk melanjutkan ke page selanjutnya yang menggunakan data tersebut.

Tabel 3.10 *django_session*

No	Nama Field	Tipe Data	Panjang Data	Keterangan
1	<i>session_key</i>	varchar	40	Primary Key
2	<i>session_data</i>	text	100	
3	<i>expire_date</i>	datetime	50	

Pada tabel 3.10, tabel ini akan menyimpan data session dari pengguna aplikasi agar saat melakukan *session* data dapat ditemukan dengan cepat pada tabel ini. Data yang paling penting dari tabel diatas adalah *session_key* itu sendiri.

11. Tabel 3.11 *simulation_block* adalah tabel dari data yang akan dibuatkan *block*, tabel ini berisi *block* dari data *voter* yang sudah memilih kandidat.

Tabel 3.11 Tabel *simulation_block*

No	Nama Field	Tipe Data	Panjang Data	Keterangan
1	<i>id</i>	integer	11	Primary Key
2	<i>prev_h</i>	varchar	64	
3	<i>merkle_h</i>	varchar	64	
4	<i>h</i>	varchar	64	
5	<i>nonce</i>	integer	11	
6	<i>timestamp</i>	real	50	

Pada tabel 3.11, data yang akan dibuatkan *block* akan masuk pada tabel ini, dan *database* ini menjadi kunci dari sistem *blockchain* ini karena *database* inilah yang menyimpan data *chaining* dari data-data yang lainnya,

dan semua data yang masuk pada tabel ini sudah terotentifikasi pada tabel sebelumnya.

12. Tabel 3.12 simulation_vote adalah tabel berisi data simulasi voter yang memilih kandidat disimpan dengan timestamp yang dienkripsi.

Tabel 3.12 Tabel simulation_vote

No	Nama Field	Tipe Data	Panjang Data	Keterangan
1	id	char	32	Primary Key
2	vote	integer	11	
3	timestamp	real	50	
4	block_id	integer	11	

Pada tabel 3.12 ini, menyimpan data dari sebuah *block*, *block* yang sudah mempunyai id, *timestamp* dan *vote* dari kandidat akan disimpan disini, untuk selanjutnya digunakan lagi jika ada data baru yang masuk ke dalam sebuah *block*.

13. Tabel 3.13 simulation_votebackup adalah tabel yang berisi sama dengan tabel simulation_vote, tabel ini menjadi tabel backup jika tabel simulation_vote rusak.

Tabel 3.13 simulation_votebackup

No	Nama Field	Tipe Data	Panjang Data	Keterangan
1	id	char	32	Primary Key
2	vote	integer	11	
3	timestamp	real	50	
4	block_id	integer	11	

Pada tabel 3.13 ini, digunakan sebagai tabel backup jika data yang akan masuk lebih cepat dari satu data yang dimasukkan kedalam *block*, dan data yang disimpan disini adalah data lama (*old*) sebelum data diperbarui lagi.

14. Tabel 3.14 tb_admin adalah tabel yang berisi data admin yang mengelola aplikasi web, dan juga data pengawas. Data dilengkapi dengan level.

Tabel 3.14 tb_admin

No	Nama Field	Tipe Data	Panjang Data	Keterangan
1	user	varchar	16	
2	pass	varchar	8	
3	level	varchar	16	

Pada tabel 3.14 ini, berada pada *database* MySQL, tabel admin ini berisi data admin dengan berbagai level dari admin tersebut, data didalam tabel ini tidak terenkripsi namun ketika masuk ke dalam session akan terenkripsi.

15. Tabel 3.15 tb_pemilih adalah tabel berisi data voter atau pemilih, data berisi nomer ktp, alamat, nama pemilih, id pemilih dan password.

Tabel 3.15 tb_ pemilih

No	Nama Field	Tipe Data	Panjang Data	Keterangan
1	id_pemilih	integer	11	Primary Key
2	ktp	varchar	16	
3	nama_pemilih	varchar	64	
4	alamat	varchar	255	
5	pass	varchar	8	

Pada tabel 3.15 ini, data seorang *voter* disimpan, isi dari tabel tersebut adalah id, ktp, nama pemilih, alamat dan pass, data yang admin input pada website akan tersimpan ditabel ini, ini adalah tabel pertama dan belum terotentifikasi oleh karena belum ada data kandidat yang dipilih.

16. Tabel 3.16 tb_pencalon adalah tabel berisi data kandidat atau pencalon, data berisi id pencalon, kode pencalon, nama pencalon, gambar, dan keterangan.

Tabel 3.16 tb_pencalon

No	Nama Field	Tipe Data	Panjang Data	Keterangan
1	id_pencalon	integer	11	Primary Key
2	kode_pencalon	varchar	5	
3	nama_pencalon	varchar	20	
4	gambar	varchar	255	
5	keterangan	varchar	255	

Pada tabel 3.16 ini, merupakan data kandidat yang diinput oleh admin pada saat penyelenggaraan pemilihan umum, data ini juga disimpan dalam database MySQL, dan disesuaikan dengan profil dari kandidat seperti isi datanya.

17. Tabel 3.17 tb_pilih

Tabel 3.17 tb_pilih

No	Nama Field	Tipe Data	Panjang Data	Keterangan
1	ID	integer	11	Primary Key
2	id_pencalon	integer	5	
3	id_pemilih	integer	5	
4	tanda_terima	varchar	10	

Pada tabel 3.17 ini adalah tabel berisi data dari *voter* yang memilih kandidat dan tanda terima atau transaksi yang dimiliki *voter*, tabel berisi data ID, id pencalon, id pemilih, dan tanda terima. Data tanda terima ini adalah hasil transaksi dari *voter* pada saat melakukan pemilihan umum, data tanda terima ini digunakan untuk memvalidasi data voter.

BAB IV

HASIL DAN PEMBAHASAN

4.1 Hasil Aplikasi

1. Tampilan Menu Utama

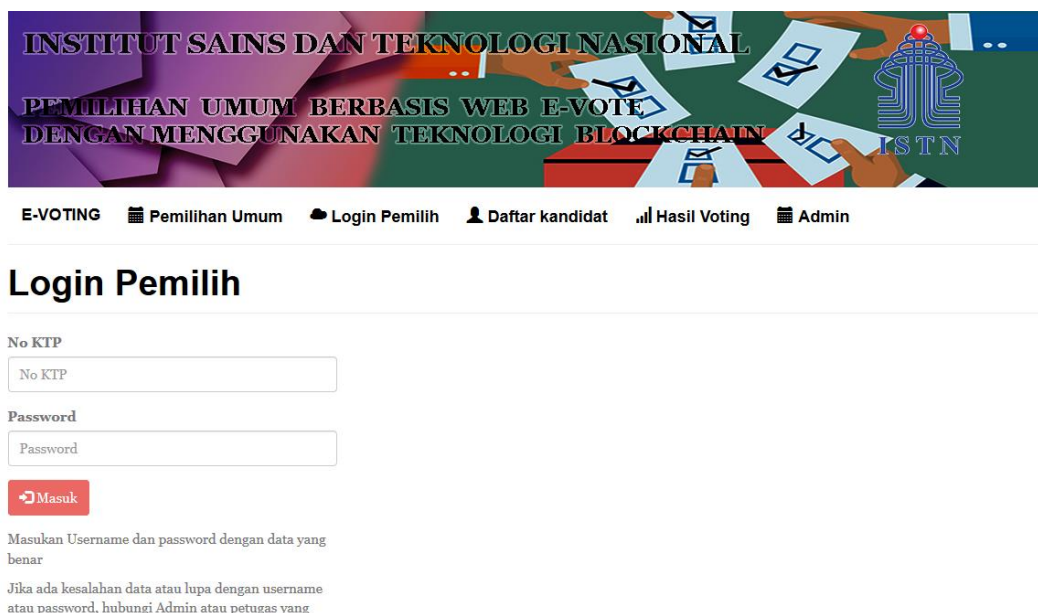
Hasil dari tampilan menu utama ini menampilkan fitur-fitur utama dari website seperti login screen, hasil sementara, dan daftar kandidat. Dan didalam menu utama ada pemberitahuan fitur aplikasi tersebut dan tata cara pemilihan. Tampilan menu utama ini ditunjukkan pada gambar 4.1



Gambar 4.1 Tampilan Menu Utama

2. Tampilan Login Pemilih (Voter)

Tampilan login pemilih adalah tempat dimana para pemilih atau voter melakukan login menggunakan Id atau nomer ktp mereka dan *password voter*. Tampilan login pemilih ditunjukkan pada gambar 4.2 dibawah



INSTITUT SAINS DAN TEKNOLOGI NASIONAL

PEMILIHAN UMUM BERBASIS WEB E-VOTE
DENGAN MENGGUNAKAN TEKNOLOGI BLOCKCHAIN

E-VOTING Pemilihan Umum Login Pemilih Daftar kandidat Hasil Voting Admin

Login Pemilih

No KTP

Password

Masuk

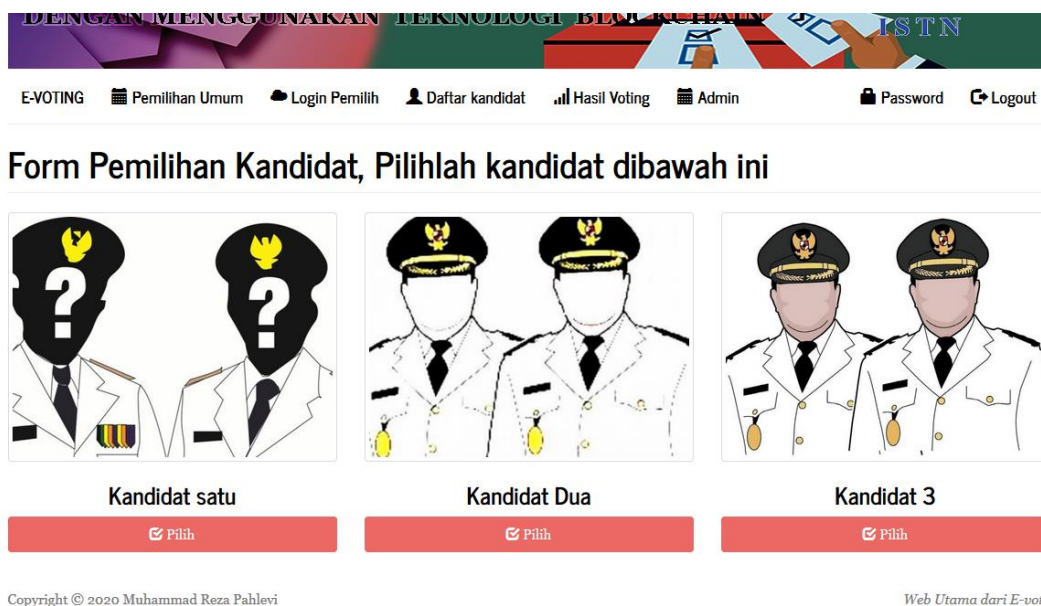
Masukan Username dan password dengan data yang benar

Jika ada kesalahan data atau lupa dengan username atau password, hubungi Admin atau petugas yang

Gambar 4.2 Tampilan Login

3. Tampilan Pemilihan Kandidat

Pada tampilan ini, pemilih atau *voter* dapat memilih kandidat yang tersedia pada halaman ini, *voter* dapat menekan tombol pilih pada bawah gambar kandidat masing masing. Tampilan pemilihan kandidat dapat dilihat pada gambar 4.3



DENGAN MENGGUNAKAN TEKNOLOGI BLOCKCHAIN

E-VOTING Pemilihan Umum Login Pemilih Daftar kandidat Hasil Voting Admin Password Logout

Form Pemilihan Kandidat, Pilihlah kandidat dibawah ini



Kandidat satu

Pilih



Kandidat Dua

Pilih



Kandidat 3

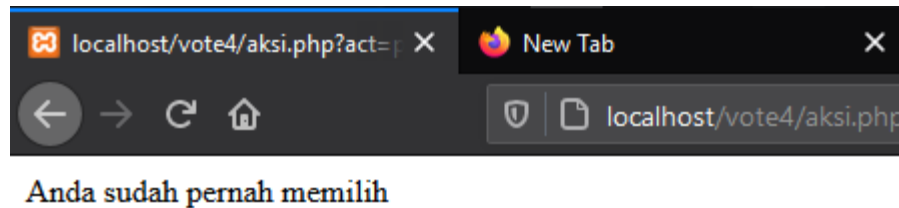
Pilih

Copyright © 2020 Muhammad Reza Pahlevi

Web Utama dari E-vote

Gambar 4.3 Tampilan pemilihan kandidat

Voter tidak dapat memilih kandidat lebih dari satu kali, jika *voter* mencoba memilih lagi maka *voter* akan ke halaman anda sudah pernah memilih. Gambar 4.4 yang akan disajikan seperti berikut:



Gambar 4.4 Sudah Memilih

4. Tampilan Tanda Terima

pada tampilan ini setelah *voter* memilih kandidat yang mereka pilih sendiri, *voter* akan mendapatkan kode, kode ini adalah kode transaksi dalam *blockchain* yang harus *voter* simpan agar suara *voter* dapat dipertanggungjawabkan. Tampilan tanda terima ditunjukkan pada gambar 4.5 dibawah.

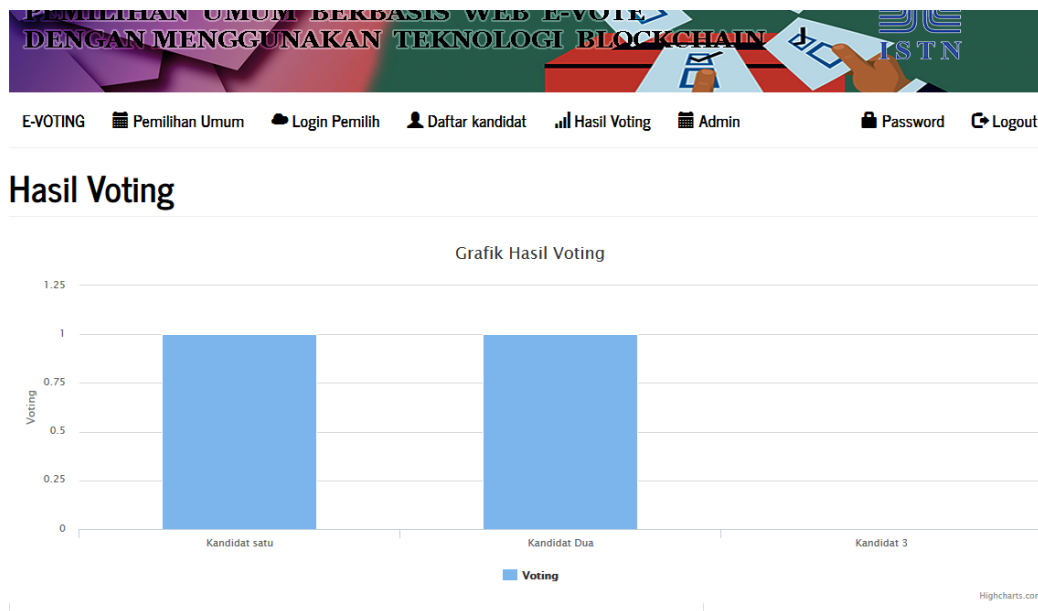


Gambar 4.5 Tampilan Tanda Terima

5. Tampilan Hasil Voting

Pada halaman ini menunjukkan berapa banyak suara yang sudah masuk, dan berapa banyak suara yang didapatkan para kandidat yang tersedia, tampilan

data berupa grafik yang pada data menunjukkan suara yang didapat kandidat dengan kandidat itu sendiri. Tampilan hasil *voting* ditunjukkan pada gambar 4.6 dibawah.



Gambar 4.6 Tampilan Hasil Sementara

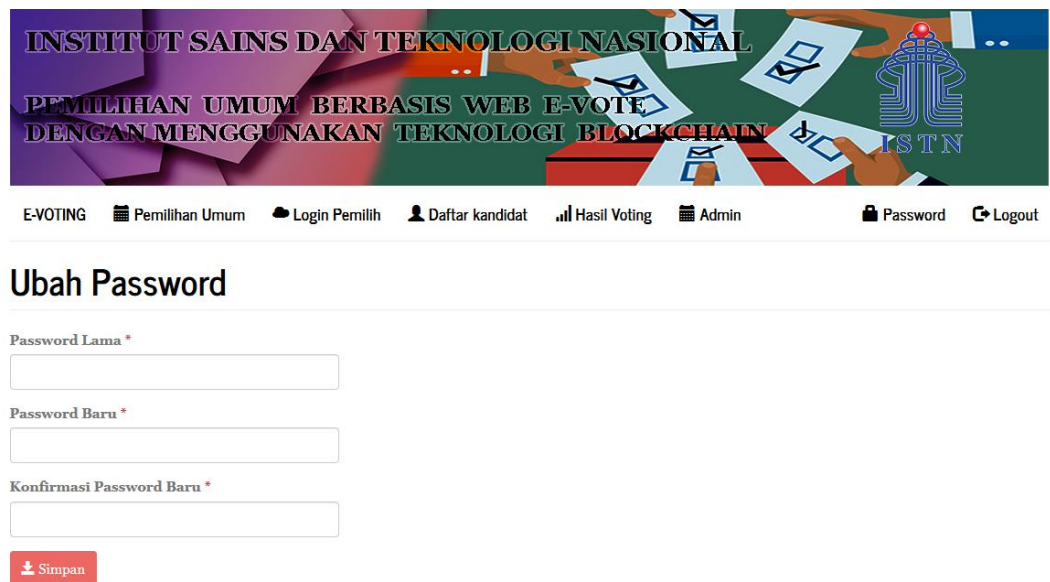
Selain dari data grafik seperti gambar 4.5 diatas, data yang diterima oleh kandidat juga ditampilkan dalam bentuk tabel dibawah grafik tersebut, agar suara yang didapat oleh kandidat menjadi jelas. Tampilan dari tabel tersebut ditunjukkan pada gambar 4.7 dibawah.

Kandidat	Total
	1
	1
	0

Gambar 4.7 Tampilan tabel suara kandidat

6. Tampilan Ubah Password

Fitur ini berfungsi untuk mengubah kata sandi dari pengguna seperti *voter*, pengawas, maupun admin. Fitur ini dapat digunakan setelah pengguna dari aplikasi ini telah login terlebih dahulu. Tampilan dari ubah *password* ditunjukkan pada gambar 4.8 dibawah.



INSTITUT SAINS DAN TEKNOLOGI NASIONAL

PEMILIHAN UMUM BERBASIS WEB E-VOTE
DENGAN MENGGUNAKAN TEKNOLOGI BLOCKCHAIN

E-VOTING Pemilihan Umum Login Pemilih Daftar kandidat Hasil Voting Admin Password Logout

Ubah Password

Password Lama *

Password Baru *

Konfirmasi Password Baru *

 Simpan

Gambar 4.8 Tampilan Ubah Password

7. Login Admin

Sama seperti tampilan login *voter*, login admin berfungsi untuk masuk kedalam aplikasi untuk menggunakannya, hanya bedanya dalam halamn login ini khusus untuk admin dan juga pengawas pemilihan umum. Tampilan dari login admin ini ditunjukkan pada gambar 4.9 dibawah.

INSTITUT SAINS DAN TEKNOLOGI NASIONAL
PEMILIHAN UMUM BERBASIS WEB E-VOTE
DENGAN MENGGUNAKAN TEKNOLOGI BLOCKCHAIN

E-VOTING Pemilihan Umum Login Pemilih Daftar kandidat Hasil Voting Admin

Login Admin

Username

Password

[Masuk](#)

Masukan Username dan password dengan data yang benar
 Jika ada kesalahan data atau lupa dengan username atau password, hubungi Admin atau petugas yang bersangkutan

Gambar 4.9 Login admin

8. Tampilan Menu Utama Admin

Pada tampilan ini, admin dapat memasukan data *voter* pada bagian panel pemilih, dan admin juga dapat memasukan data pencalon atau kandidat pada bagian panel pencalon. Tampilan menu utama admin ditunjukkan pada gambar 4.10 dibawah.

INSTITUT SAINS DAN TEKNOLOGI NASIONAL
PEMILIHAN UMUM BERBASIS WEB E-VOTE
DENGAN MENGGUNAKAN TEKNOLOGI BLOCKCHAIN

E-VOTING Pencalon Pemilih Password Logout

Fitur Aplikasi dan Tata Cara Pemilihan

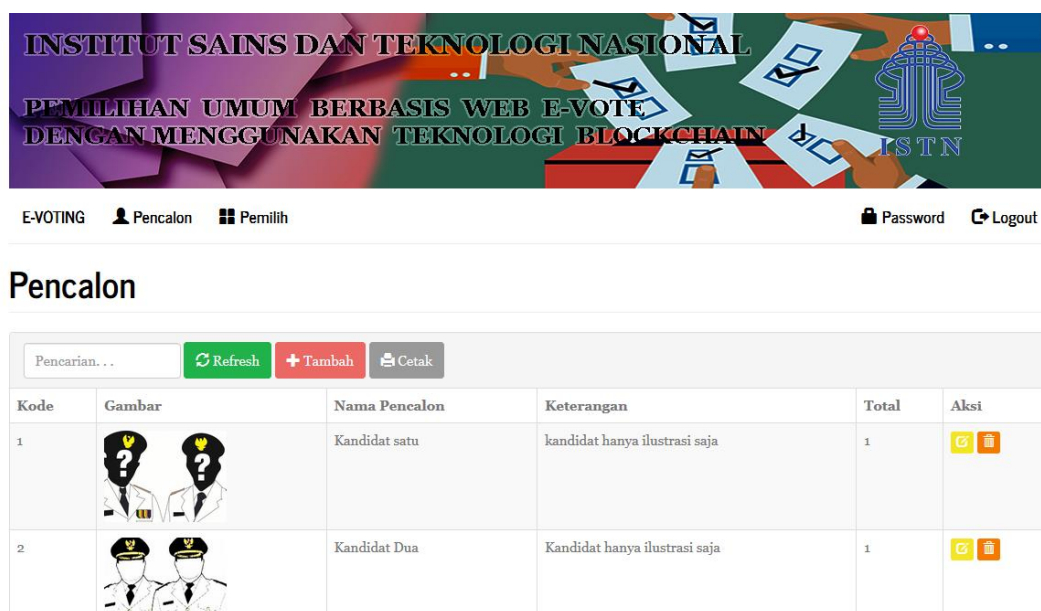
Website ini dikembangkan untuk memenuhi kebutuhan akan media tempat untuk memilih suatu kandidat dari berbagai macam pemilihan umum. website ini masih dalam tahapan pengembangan, dalam kata lain masih prototype, karena itu beberapa fitur masih dalam tahap pengembangan, fitur-fitur tersebut adalah :

1. Fitur Login
 Fitur login ini dibuat agar identitas dari pengguna aplikasi web ini menjadi jelas dan keamanan dari sistem ini dapat dijaga
2. Fitur memilih kandidat
 Fitur ini hanya berlaku untuk pengguna aplikasi yang akan memilih kandidat yang disediakan oleh admin, nantinya setelah pengguna memilih kandidat, data akan dienkripsi dan pengguna akan mendapat kode transaksi yang dapat dipergunakan sebagai alat pertanggung jawaban pemilih
3. Fitur Melihat hasil kandidat yang dipilih (Hasil sementara)
 Fitur ini berfungsi untuk pengguna aplikasi untuk melihat hasil voting sementara dari hasil pemilihan yang dilakukan, fitur ini akan berupa bar atau Grafik dari hasil suara masing-masing kandidat
4. Fitur Pengawas

Gambar 4.10 Menu Utama Admin

9. Tampilan Data Kandidat

Halaman ini hanya bisa diakses oleh admin yang memiliki fitur untuk menambah kandidat pemilihan umum. Data dari kandidat tersebut adalah kode, gambar kandidat, nama kandidat, dan keterangan. Pada halaman ini juga admin dapat mengedit atau menghapus data kandidat. Tampilan dari data kandidat ini ditunjukkan pada gambar 4.11 dibawah.



Kode	Gambar	Nama Pencalon	Keterangan	Total	Aksi
1		Kandidat satu	kandidat hanya ilustrasi saja	1	 
2		Kandidat Dua	Kandidat hanya ilustrasi saja	1	 

Gambar 4.11 Data Kandidat

10. Tambah Data Kandidat

Pada halaman ini, admin dapat menambah data dari kandidat dengan memasukan data kandidat berupa kode, nama kandidat, gambar dari kandidat, dan keterangan. Kode yang dimasukan admin harus berurutan dengan jumlah kandidat yang dimasukan, missal kandidat satu dengan kode 1, dan kandidat dua dengan kode 2, dan seterusnya. Keterangan adalah catatan untuk kandidat, seperti apakah kandidat sudah memenuhi syarat untuk menjadi kandidat atau belum. Tampilan dari tambah data kandidat ditunjukkan pada gambar 4.12 dibawah.

Tambah Pecalon

Kode *

Nama Pecalon *

Gambar *

Browse... No file selected.

Keterangan

Simpan Kembali

Gambar 4.12 Tambah Kandidat

11. Tampilan Data Pemilih (Voter)

Pada halaman ini menampilkan data pemilih atau voter, data pemilih berupa nomer ktp, nama pemilih, alamat dan status. Status menunjukkan apakah voter sudah memilih atau belum akan ditampilkan pada status ini. Tampilan dari data pemilih ditunjukkan pada gambar 4.13 dibawah.

Pemilih

Pencarian... Refresh Tambah Cetak

No	KTP	Nama Pemilih	Alamat	Status	Aksi
1	1234567890	Alfreds Futterkiste	Obere Str. 57		
2	3216549870	Ana Trujillo Emparedados y helados	Avda. de la Constitucion 2222		
3	0987654321	Antonio Moreno Taqueria	Mataderos 2312		
4	1324567980	Around the Horn	120 Hanover Sq.		
5	0897564231	Berglunds snabbkop	Berguvsvagen 8		

Gambar 4.13 Data Pemilih

12. Tambah Data Pemilih

Pada halaman ini, admin dapat memasukan data dari *voter* atau pemilih, data dari *voter* tersebut berupa nomer ktp, nama, alamat, dan *password*. Tampilan tambah data pemilih ditunjukkan pada gambar 4.14 dibawah.

PEMILIHAN UMUM BERBASIS WEB E-VOTE
DENGAN MENGGUNAKAN TEKNOLOGI BLOCKCHAIN

E-VOTING Pencalon Pemilih Password Logout

Tambah Pemilih

No KTP *

Nama Pemilih *

Alamat

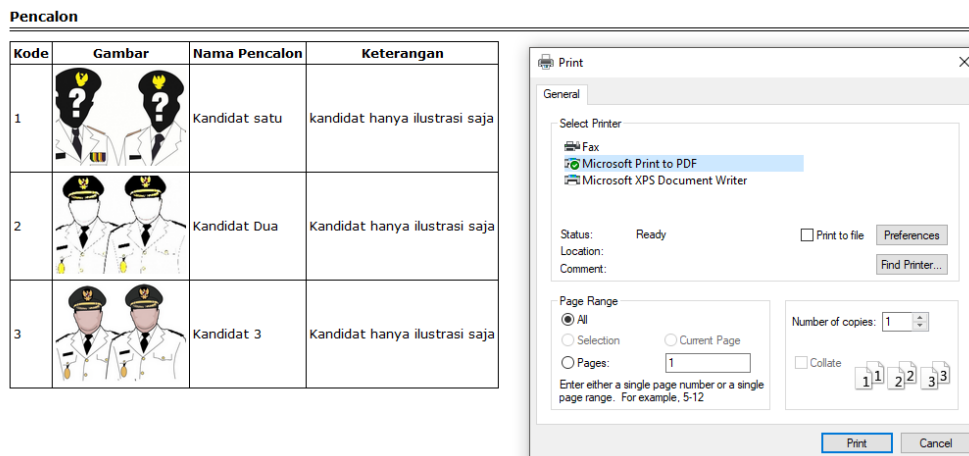
Password *

Simpan Kembali

Gambar 4.14 Tambah Data Pemilih (*Voter*)

13. Cetak Data Kandidat

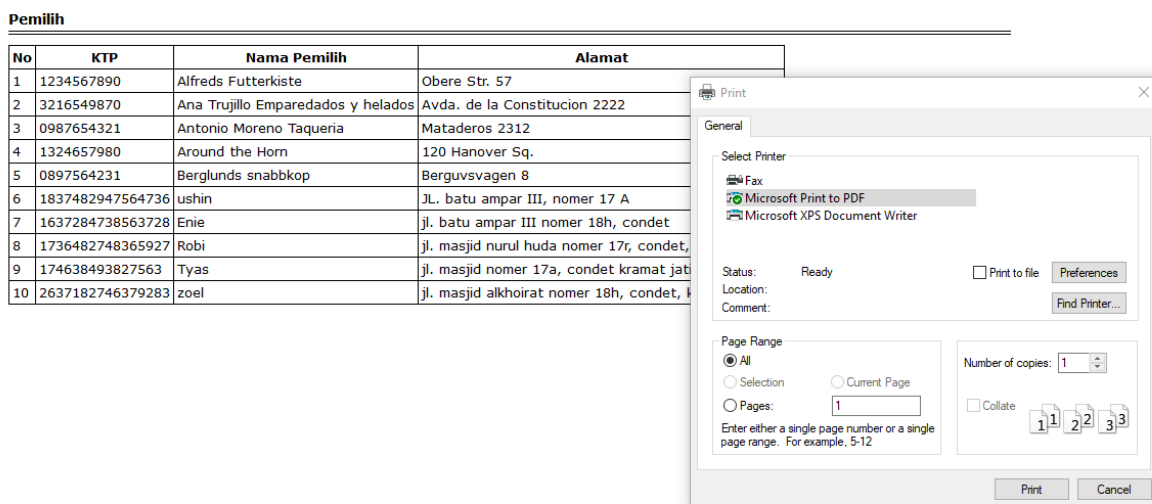
Pada fitur ini, admin dan pengawas dapat mencetak data dari *voter* maupun kandidat untuk bukti fisik akan data yang nyata selain pada *database*, saat mencetak pada kertas, data yang ditampilkan berupa tabel yang berisi kode pencalon, Gambar dari kandidat, dan keterangan dari kandidat. Tampilan dari cetak kandidat dapat dilihat dari gambar 4.15 dibawah.



Gambar 4.15 tampilan cetak kandidat

14. Cetak Data Pemilih

Halaman ini memperlihatkan bagaimana admin mencetak data *voter*, data *voter* yang dicetak dikertas berupa tabel yang berisi nomer ktp, nama pemilih, dan alamat dari. Tampilan dari cetak data pemilih ditunjukkan pada gambar 4.16 dibawah.



Gambar 4.16 tampilan cetak data pemilih

15. Tampilan Menu Utama Pengawas

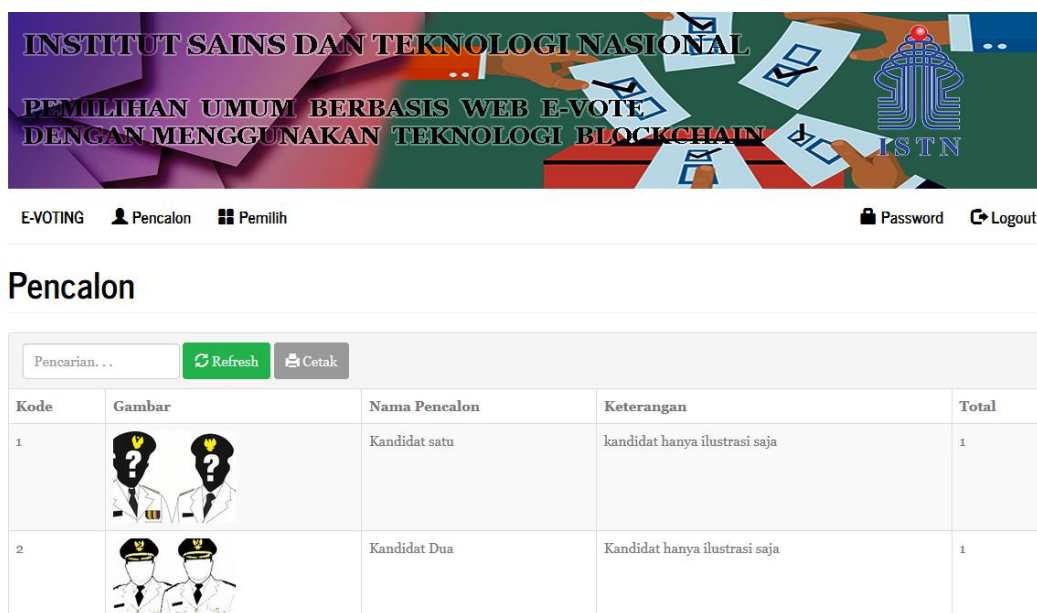
Tampilan halaman ini hanya bisa diakses oleh pengawas pemilihan umum, pengawas hanya diberi akses seperti tugasnya yaitu melihat data *voter* yang sudah terdaftar, melihat data kandidat yang terdaftar, mencetak data *voter* dan kandidat yang terdaftar tanpa bisa mengubah isi dari data *voter* dan kandidat. Tampilan dari menu utama pengawas ditunjukkan pada gambar 4.17 dibawah.



Gambar 4.17 Menu utama pengawas

16. Tampilan Data Kandidat Oleh Pengawas

Pada halaman ini, pengawas dapat melihat data kandidat dan mencetaknya, tidak seperti admin, pengawas hanya dapat melihat tanpa bisa mengedit data dari kandidat, namun pengawas juga diberikan akses untuk mencetak data kandidat. Tampilan dari data kandidat oleh pengawas ditunjukkan pada gambar 4.18 dibawah.

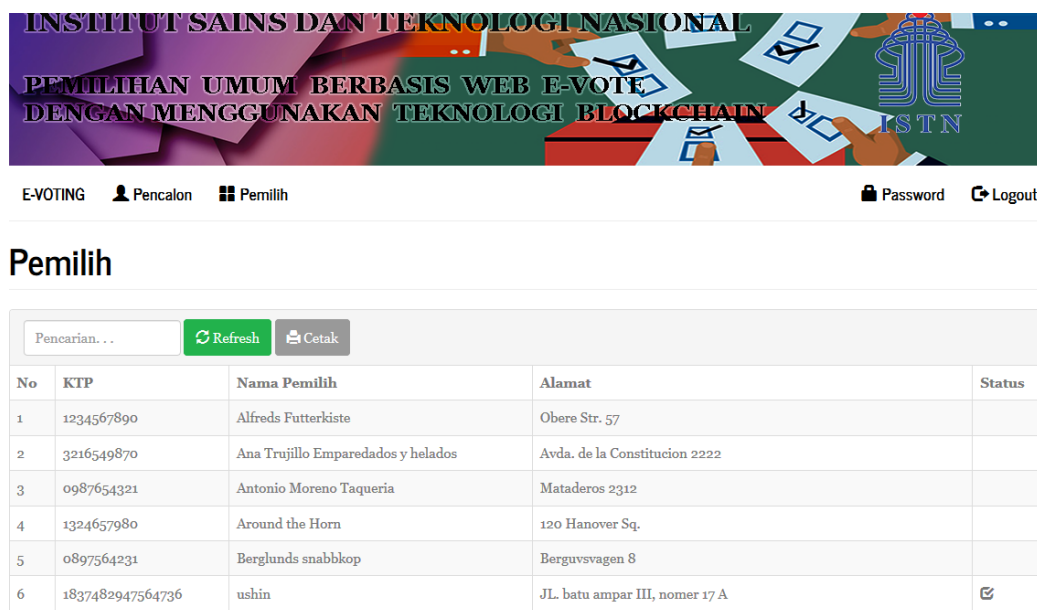


Kode	Gambar	Nama Pencalon	Keterangan	Total
1		Kandidat satu	kandidat hanya ilustrasi saja	1
2		Kandidat Dua	Kandidat hanya ilustrasi saja	1

Gambar 4.18 Tampilan data kandidat oleh pengawas

17. Tampilan Data Voter Oleh Pengawas

Halaman ini juga menunjukkan data *voter*, namun pengawas tidak bisa mengubah datanya, dan hanya bisa melihat dan mencetak data. Tampilan data *voter* oleh pengawas ditunjukkan pada gambar 4.19 dibawah.

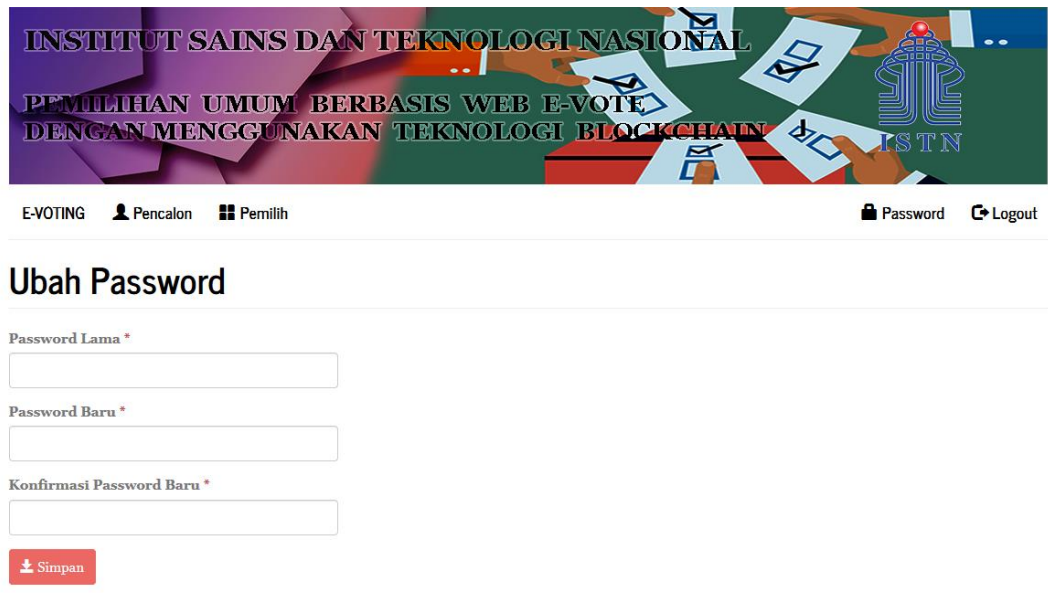


No	KTP	Nama Pemilih	Alamat	Status
1	1234567890	Alfreds Futterkiste	Obere Str. 57	
2	3216549870	Ana Trujillo Emparedados y helados	Avda. de la Constitucion 2222	
3	0987654321	Antonio Moreno Taqueria	Mataderos 2312	
4	1324657980	Around the Horn	120 Hanover Sq.	
5	0897564231	Berglunds snabbkop	Berguvsvagen 8	
6	1837482947564736	ushin	JL. batu ampar III, nomer 17 A	

Gambar 4.19 Data Voter Oleh Pengawas

18. Ubah Password Admin dan Pengawas

Halaman ini menunjukkan seperti voter, admin dan pengawas jika ingin mengubah password untuk login dapat diubah pada halaman ini. Tampilan dari ubah password admin dan pengawas ditunjukkan pada gambar 4.20 dibawah.



INSTITUT SAINS DAN TEKNOLOGI NASIONAL

PEMILIHAN UMUM BERBASIS WEB E-VOTE
DENGAN MENGGUNAKAN TEKNOLOGI BLOCKCHAIN

E-VOTING Pencalon Pemilih Password Logout

Ubah Password

Password Lama *

Password Baru *

Konfirmasi Password Baru *

 Simpan

Gambar 4.20 Ubah password admin dan pengawas

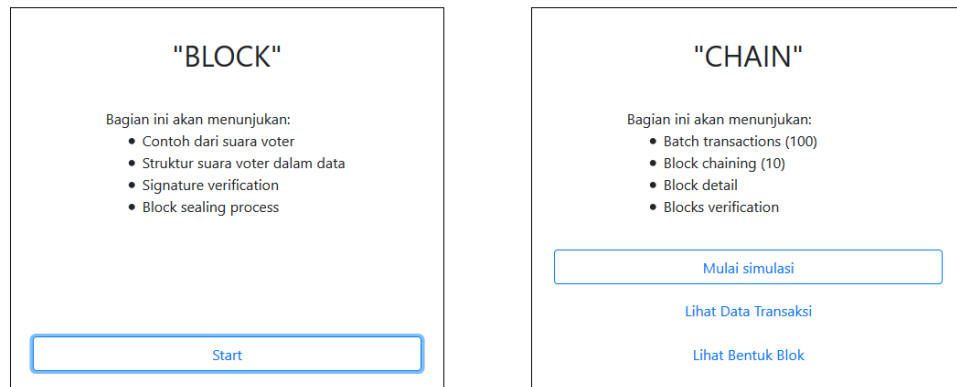
19. Tampilan Background Menu Utama

Pada tahap tampilan ini, aplikasi hanya untuk pengembangan penelitian, aplikasi berikut menunjukkan bagaimana proses yang terjadi pada aplikasi utama tahap demi tahap.

Tampilan background manu utama ini menampilkan fitur yang sedikit berbeda karena hanya untuk melihat bagaimana data dibentuk berupa *Block*, dan merantai data yang dibentuk sehingga terciptalah proses *Blockchain*.

Menu utama dari aplikasi ini menunjukkan bagaimana membuat *block* dari sebuah data dan merantainya. Tampilan dari background menu utama ditunjukkan pada gambar 4.21 dibawah.

Blockchain background process based e-voting simulation



Gambar 4.21 Menu Utama Background Proses

20. Pembuatan Block Data

Pada tampilan ini, proses pembuatan data dilakukan, data yang dibuat *block* disini adalah data dari *voter* yang telah memilih kandidat namun dalam bentuk enkripsi, id disini adalah data *voter* yang dienkripsi RSA, data voter-nya yaitu nomer ktp, nama, alamat, dan password.

Private key disini adalah hasil dari generating antara data *voter* tadi dan kandidat yang dipilih dan menghasilkan *private key*, proses ini dilakukan komputer sebagai syarat pembuatan *block* data. Tampilan dari pembuatan *block* data ditunjukkan pada gambar 4.22 dibawah.

Gambar 4.22 Pembuatan Block Data

Pada simulasi ini halaman ini, *private key* berada pada sistem setting milik *framework* django. *Private key* ini berfungsi untuk *hash* tambahan yang dibuat oleh *framework* ini yang berfungsi untuk membuat *hash* baru dan membuat *signature* atau tanda terima.

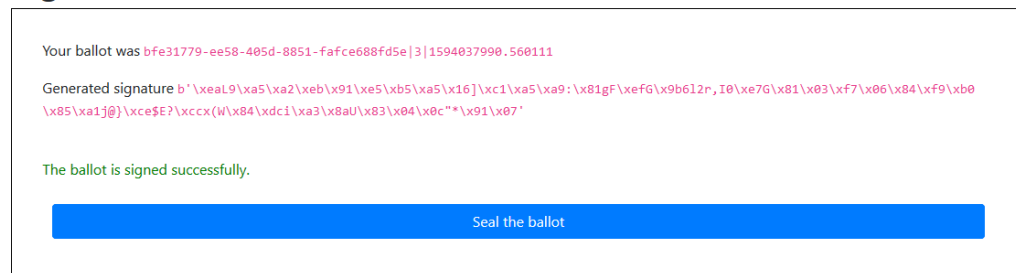
Private key ini akan saling berkesinambungan dengan *public key* yang berada pada *framework* ini. Karena untuk melakukan enkripsi pada suatu data diperlukan *private key*, dalam hal ini menggunakan metode RSA.

Sebaliknya *public key* disini berfungsi untuk mendekrip data yang dienkrpsi sebelumnya dengan menggunakan *private key*. Dengan adanya *public key* ini, sistem tidak perlu repot mencari *key* baru yang akan memakan waktu yang sangat lama, ditambah dengan jumlah data dari para *voter* yang bisa sangat banyak jika pemilihan dilakukan secara besar.

21. *Signature Verification*

Disini setelah data diproses, data akan di-verifikasi apakah data yang sudah dimasukan tadi telah benar atau tidak, dan dalam proses ini kode tanda tangan akan dibuat oleh sistem, data ini berupa kode yang akan diberikan kepada *voter* untuk menunjukan bahwa *voter* telah memilih kandidat. Tampilan pada gambar 4.23 dibawah.

Signature Verification



Created by Muhammad Reza Pahlevi
1 July 2020
Aplikasi ini adalah konsep dari penggunaan Blockchain pada website E-Vote

Gambar 4.23 Signature Verification

22. *Block Detail*

Pada tampilan ini, data dari *voter* sudah menjadi sebuah *block*, isi dari data *block* ini adalah *previous hash*, atau *hash* dari data yang sebelumnya, jika data tidak terantai maka hanya akan keluar data *genesis*, *transaction hash* adalah enkripsi data dari *voter* yang sudah memilih, data tersebut berisi nomer ktp, nama, alamat, *password*, dan status.

Block hash disini adalah enkripsi dari data *voter* yang sudah memilih, *transaction hash*, *nonce*, dan *timestamp*. *Nonce* disini adalah data hasil kerja dari sebuah sistem *blockchain* untuk menunjukan bahwa suatu block data sudah dibuat atau disebut juga *proof of work*.

Timestamp disini adalah data berupa waktu yang dibuat oleh sistem, waktu disini diambil pada saat sistem membuat *block*, bukan pada saat *voter*

melakukan pemilihan. Tampilan dari *block detail* ditunjukkan pada gambar 4.24 dibawah.

Block detail

Previous hash	GENESIS
Transaction hash	7a75bf57a204bcfba0545a46df6d3c18c220a389e6dd2e4654dbc5072e572005
Block hash	0000613cb33ab7db4649be9450e7fd5a08d123e3a7ceb05426018bac2a8f9675
Nonce	16772
Timestamp	1594038067.594395 (2020-07-06 19:21:07)
Back to homepage	

Created by Muhammad Reza Pahlevi
1 July 2020
Aplikasi ini adalah konsep dari penggunaan Blockchain pada website E-Vote

Gambar 4.24 Block Detail

23. Tampilan *Pycharm Command Block*

Tampilan ini untuk memperjelas bahwa sistem benar-benar melakukan pembuatan *block* dengan benar. Pada saat aplikasi dijalankan dan melaksanakan perintah untuk membuat *block*, maka tampilan dari *command line* aplikasi akan tetap terlihat. Tampilan pada saat aplikasi selesai membuat *block*, tampilan pada *command line* juga berhasil membuat *block*, dengan kata lain data sebuah *block* sudah dibuat. Tampilan *command line block* ditunjukkan pada gambar 4.25 dibawah.

```

Run: manage x
trial hash: b978652bf837fbd2439fbe8cd927fd0b8fc8405fe6e5f08c3317300de12b65f7
trial hash: 0000613cb33ab7db4649be9450e7fd5a08d123e3a7ceb05426018bac2a8f9675
block is sealed in 4.953281879425049 seconds
[06/Jul/2020 19:21:15] "POST /ballot/seal/ HTTP/1.1" 200 2416
[06/Jul/2020 19:22:43] "GET / HTTP/1.1" 200 2868

```

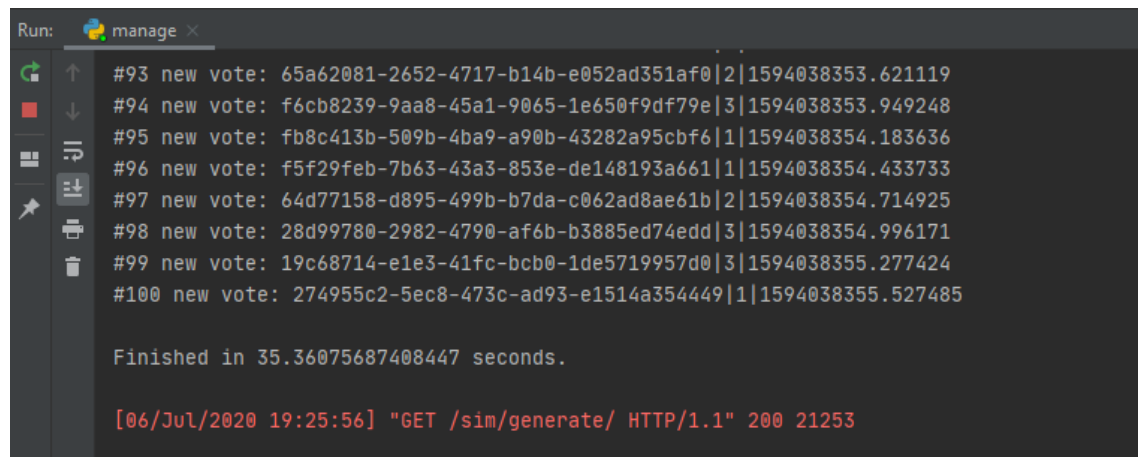
Gambar 4.25 Command line block

24. Proses *Chaining*

Pada proses ini akan menunjukkan bagaimana proses *chaining* atau merantai data satu dengan data seterusnya, data yang dirantai berbentuk enkripsi RSA dari data *voter* yang sudah memilih kandidat dan *timestamp* dari kapan proses data ini dibuat.

Karena proses ini terjadi spontan ketika seorang *voter* melakukan pemilihan lalu menekan tombol pilih dan sistem melakukan penyimpanan *database*, maka proses pembuatan data tidak terlihat diinterface.

Namun dalam proses ini ketika dimenu utama sebelah kanan pada tab “CHAINING” dan setelah menekan tombol start maka proses *chaining* dimulai. Tampilan dari proses *chaining* dilakukan ditunjukkan pada gambar 4.26 dibawah



```
Run: manage x
#93 new vote: 65a62081-2652-4717-b14b-e052ad351af0|2|1594038353.621119
#94 new vote: f6cb8239-9aa8-45a1-9065-1e650f9df79e|3|1594038353.949248
#95 new vote: fb8c413b-509b-4ba9-a90b-43282a95cbf6|1|1594038354.183636
#96 new vote: f5f29feb-7b63-43a3-853e-de148193a661|1|1594038354.433733
#97 new vote: 64d77158-d895-499b-b7da-c062ad8ae61b|2|1594038354.714925
#98 new vote: 28d99780-2982-4790-af6b-b3885ed74edd|3|1594038354.996171
#99 new vote: 19c68714-e1e3-41fc-bcb0-1de5719957d0|3|1594038355.277424
#100 new vote: 274955c2-5ec8-473c-ad93-e1514a354449|1|1594038355.527485

Finished in 35.36075687408447 seconds.

[06/Jul/2020 19:25:56] "GET /sim/generate/ HTTP/1.1" 200 21253
```

Gambar 4.26 Proses Chaining

25. Data Enkripsi Voter

Setelah proses chaining selesai, maka data siap dibuat menjadi block-block data sesuai dari pengembang ingin membuat berapa block dari data voter yang sudah valid. Data tersebut seperti nomer ktp, nama, alamat, nomer kandidat yang dipilih dan timestamp. Tampilan data enkripsi voter ditunjukkan pada gambar 4.27 dibawah.

List of unconfirmed votes

Showing last 100 transactions (or all if <=100)

Voter ID	Vote	Timestamp
274955c2-5ec8-473c-ad93-e1514a354449	1	1594038355.527485
19c68714-e1e3-41fc-bcb0-1de5719957d0	3	1594038355.277424
28d99780-2982-4790-af6b-b3885ed74edd	3	1594038354.996171
64d77158-d895-499b-b7da-c062ad8ae61b	2	1594038354.714925
f5f29Feb-7b63-43a3-853e-de148193a661	1	1594038354.433733
...	...	...

Mine into blocks

Created by Muhammad Reza Pahlevi

1 July 2020

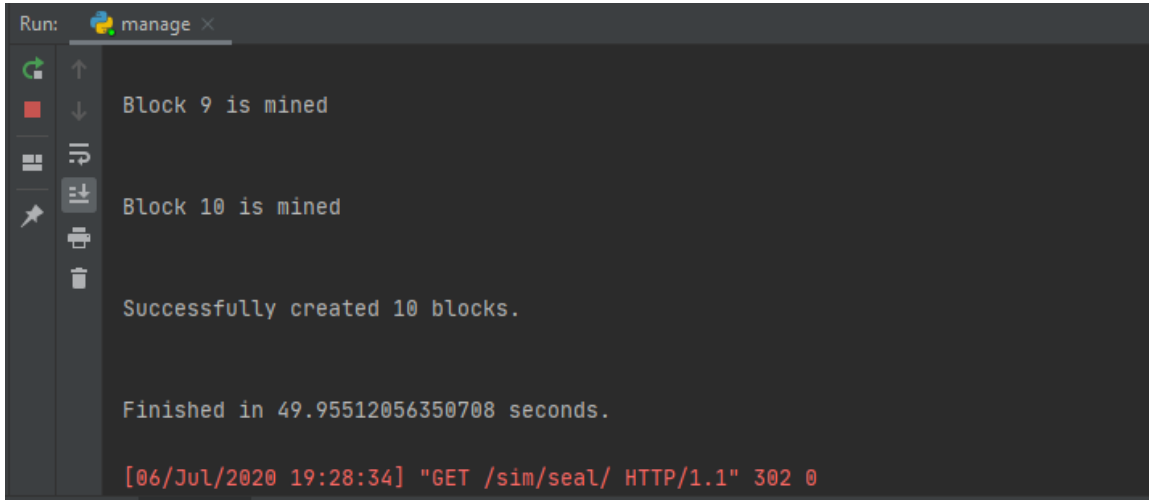
Aplikasi ini adalah konsep dari penggunaan Blockchain pada website E-Vote

Gambar 4.27 Data Enkripsi Voter

26. Proses Mining Block

Proses ini adalah proses paling penting dari aplikasi ini yaitu proses mining block yang sudah dirantai atau bisa disebut inilah inti dari proses dari teknologi Blockchain bekerja.

Data dari para voter yang sudah dienkripsi tadi dibuatkan block-block data, block data ini juga akan dirantai dengan block data seterusnya yang juga disebut proses chaining tadi. Tampilan pada web hanya akan meloading data, sedangkan proses pada command line atau sistem akan terlihat pada gambar 4.28 dibawah.



Gambar 4.28 Proses Mining Block

27. List Of Block

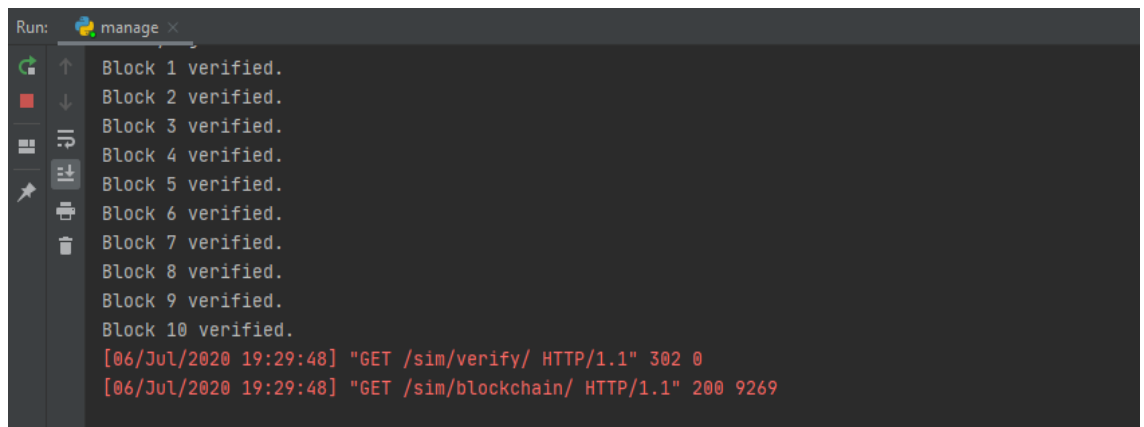
Setelah *block* dibuat, tampilan akan menunjukkan daftar dari *block-block* yang dibuat tersebut. Data dari *block* tersebut adalah berupa tabel yang berisi *previous hash* adalah enkripsi SHA3-256 dari sebuah *block*, lalu ada *block hash* atau isi dari *block* tersebut yang dienkripsi, lalu *merkle root hash* atau enkripsi dari mekanisme *block* ini dibuat, *nonce* atau *proof of work* dari setiap *block* dan *timestamp* dari kapan *block* ini dibuat. Tampilan dari list of *block* tersebut ditunjukkan pada gambar 4.29 dibawah.

List of blocks					
See all transactions		Back to homepage			
#	Previous Hash	Block Hash	Merkle Root Hash	Nonce	Timestamp
1	0000000000000000000000000000...	00007784a7d1e54cfde734ddaef59...	0b3ec542d2762764b33ad2e5c62e2...	27005	2020-07-06 19:27:44
2	00007784a7d1e54cfde734ddaef59...	00002385fa655665cd5dc68c9f91...	9824bf1bf2d2e3d03493296e98a31...	16065	2020-07-06 19:27:46
3	00002385fa655665cd5dc68c9f91...	0000afb00a3fc03654b6568e2f10a...	8e8e1153efcb465c9b0a521f8e45...	22119	2020-07-06 19:27:47
4	0000afb00a3fc03654b6568e2f10a...	00001d771ec1964123a74d09984ee...	a426b22fac89ecbc57a3d1139b874...	25602	2020-07-06 19:27:49
5	00001d771ec1964123a74d09984ee...	0000a7bd358edd9602e698b1a36d7...	b6983ea8ca1b1edca3061c40fde3...	52000	2020-07-06 19:27:51
Verify					
Sync					

Gambar 4.29 List Of Block

28. Proses Verifikasi Block

Proses ini untuk verifikasi apakah *block* data yang dibuat tadi datanya benar dan tidak berubah. Proses ini akan mengecek rantai data dari hash *block* satu dengan *block* seterusnya agar memastikan semua *block* data terantai dan tidak bisa diubah disalah satu *block*. Tampilan dari varifikasi *block* berupa *command* ditunjukkan pada gambar 4.30 dibawah.

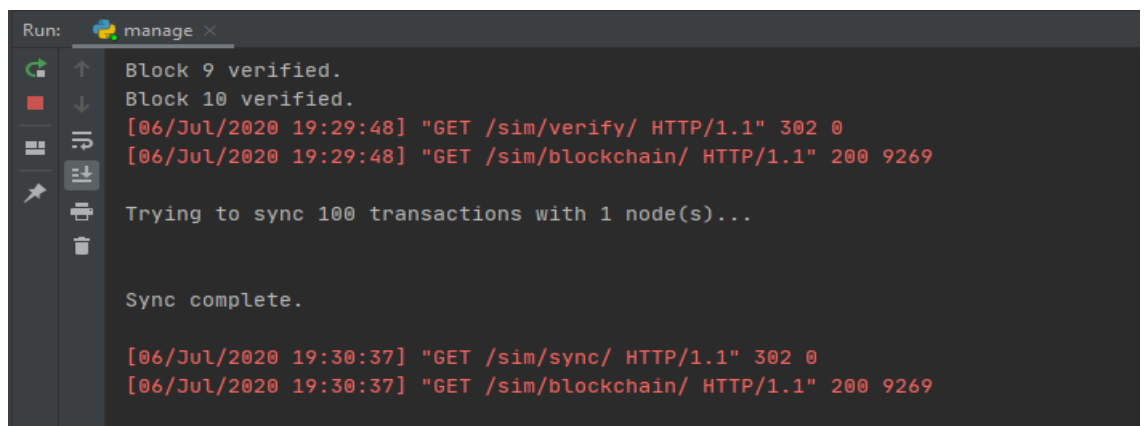


```
Run: manage x
Block 1 verified.
Block 2 verified.
Block 3 verified.
Block 4 verified.
Block 5 verified.
Block 6 verified.
Block 7 verified.
Block 8 verified.
Block 9 verified.
Block 10 verified.
[06/Jul/2020 19:29:48] "GET /sim/verify/ HTTP/1.1" 302 0
[06/Jul/2020 19:29:48] "GET /sim/blockchain/ HTTP/1.1" 200 9269
```

Gambar 4.30 Proses Verifikasi Block

29. Proses Sinkron Block

Proses mensinkronkan data ini adalah untuk memastikan bahwa data masih sama dan belum berubah. Proses ini sangat krusial jika tidak dilakukan data pada *block* akan berantakan satu dengan seterusnya karena tidak sinkron. Tampilan *command line* ditunjukkan pada gambar 4.31 dibawah.



```
Run: manage x
Block 9 verified.
Block 10 verified.
[06/Jul/2020 19:29:48] "GET /sim/verify/ HTTP/1.1" 302 0
[06/Jul/2020 19:29:48] "GET /sim/blockchain/ HTTP/1.1" 200 9269
Trying to sync 100 transactions with 1 node(s)...
Sync complete.
[06/Jul/2020 19:30:37] "GET /sim/sync/ HTTP/1.1" 302 0
[06/Jul/2020 19:30:37] "GET /sim/blockchain/ HTTP/1.1" 200 9269
```

Gambar 4.31 Proses Sinkron Block

30. *List Of Confirmed Votes*

Ini adalah tampilan data dari para *voter* yang sudah memilih tadi dan sudah dibentuk dalam bentuk *block* data. Pada tampilan ini menunjukkan data berupa *Voter ID* adalah data *voter* yang sudah memilih kandidat yang berbentuk enkripsi, lalu nomer kandidat yang dipilih, timestamp atau waktu pembuatan data, dan hash dari data-data sebelumnya.

Pada halaman ini juga menampilkan berapa banyak suara *voter* yang didapatkan oleh kandidat-kandidat yang tersedia. Data ini background dari halaman hasil sementara pada aplikasi utama di website *e-vote*. Tampilan dari *list of confirmed votes* ditunjukkan pada gambar 4.32 dibawah.

List of confirmed votes

See all blocks Back to homepage

Candidate #1: 36 votes
 Candidate #2: 32 votes
 Candidate #3: 32 votes

Showing page 1 of 1

#	Voter ID	Vote	Timestamp	Hash	Block
1	a1311dfc-17c1-4c76-abf3-f3260722bfe9	2	2020-07-06 19:25:20	1c13f73e6ef0244e9a1...	go to block
2	7fc0c550-292e-4653-b552-ee3434c61709	1	2020-07-06 19:25:20	77455f81112f7055aed...	go to block
3	94739648-163e-48c1-8103-8963eba444b4	1	2020-07-06 19:25:21	7527ac967235f1aad5...	go to block
4	d9ebc69d-7a39-4ecf-ab3b-4bd8a5f83592	2	2020-07-06 19:25:21	aa68674a26799c05946...	go to block
5	32c72918-e5fc-497f-878b-9952e536df1a	2	2020-07-06 19:25:21	e5ef9844408dfbec20d...	go to block

Gambar 4.32 List Of Confirmed Votes

31. *Blockchain Data Detail*

Pada halaman ini, data dari sebuah block yang sudah terantai akan terlihat, data yang ditampilkan yaitu:

1. *Previous hash*, atau enkripsi dari data sebelumnya, jika data merupakan data pertama maka hanya akan terlihat angka nol sepanjang angka enkripsi block.
2. *Merkle root hash* atau enkripsi dari mekanisme sebuah block dibuat.

3. *Block hash* atau enkripsi dari data voter.
4. *Nonce* atau proof of work dari block.
5. *Timestamp* atau waktu pada saat block dibuat.
6. Dan *confirmation* atau *block* yang dikonfirmasi dari *block* sebelumnya, jika *block* sebelumnya ada 5 *block*, maka *confirmation* hanya akan muncul 4 karena *block* yang sedang dilihat sudah di konfirmasi.

Tampilan dari *blockchain* data detail ini ditunjukkan pada gambar 4.33 dibawah.

Block #1

[Back to blocks list](#)
[Back to homepage](#)

Previous hash	00
Merkle root hash	0b3ec542d2762764b33ad2e5c62e2def4b2e688e9c05469858738e720fb7c234
Block hash	00007784a7d1e54cfde734ddaef594270b2a0a5ce74a863251a057fca4e1256c
Nonce	27005
Timestamp	2020-07-06 19:27:44 (4 minutes ago)
Confirmations	10 confirmations

[Next block](#)

Transactions

Re-calculated merkle root hash: 0b3ec542d2762764b33ad2e5c62e2def4b2e688e9c05469858738e720fb7c234

Transactions are not tampered

Showing page 1 of 1

#	Voter ID	Cand	Timestamp	Hash
1	1	1	1	1

Gambar 4.33 Blockchain Data Detail

32. Data Transaksi

Tampilan data transaksi ini adalah isi dari jumlah *voter* yang ada dalam *block*, transaksi hash bisa juga disebut mekanisme dari sebuah *block* dibuat dalam bentuk enkripsi atau disebut juga dengan *merkle root hash*. Tampilan ditunjukkan pada gambar 4.34 dibawah.

Transactions

Re-calculated merkle root hash: 0b3ec542d2762764b33ad2e5c62e2def4b2e688e9c05469858738e720fb7c234

Transactions are not tampered

Showing page 1 of 1

#	Voter ID	Cand	Timestamp	Hash
1	a1311dfc-17c1-4c76-abf3-f3260722bfe9	2	2020-07-06 19:25:20	1c13f73e6ef0244e9a1e2fb0bafa16e...
2	7fc0c550-292e-4653-b552-ee3434c61709	1	2020-07-06 19:25:20	77455f81112f7055aedb5ad52040c84...
3	94739648-163e-48c1-8103-8963eba444b4	1	2020-07-06 19:25:21	7527ac967235f11aad575608afb6217...
4	d9ebc69d-7a39-4ecf-ab3b-4bd8a5f83592	2	2020-07-06 19:25:21	aa68674a26799c059464142a3d836cb...
5	32c72918-e5fc-497f-878b-9952e536df1a	2	2020-07-06 19:25:21	e5ef9844408dfbec20daa2789bfb1c...

Created by Muhammad Reza Pahlevi

1 July 2020

Aplikasi ini adalah konsep dari penggunaan Blockchain pada website E-Vote

Gambar 4.34 Data Transaksi

33. Data Detail 9 Confirmation

Pada halaman ini, data dari sebuah *block* yang sudah terantai akan dengan 9 konfirmasi dari data sebelumnya, data yang ditampilkan yaitu:

1. *Previous hash*, atau enkripsi dari data sebelumnya, jika data merupakan data pertama maka hanya akan terlihat angka nol sepanjang angka enkripsi *block*.
2. *Merkle root hash* atau enkripsi dari mekanisme sebuah *block* dibuat.
3. *Block hash* atau enkripsi dari data *voter*.
4. *Nonce* atau *proof of work* dari *block*.
5. *Timestamp* atau waktu pada saat *block* dibuat.
6. Dan *confirmation* atau *block* yang dikonfirmasi dari *block* sebelumnya, jika *block* sebelumnya ada 9 *block*, maka *confirmation* hanya akan muncul 8 karena *block* yang sedang dilihat sudah di konfirmasi.

Tampilan dari *blockchain* data detail ini ditunjukkan pada gambar 4.35 dibawah.

Block #2

[Back to blocks list](#)
[Back to homepage](#)

Previous hash	00007784a7d1e54cfde734ddae594270b2a0a5ce74a863251a057fca4e1256c
Merkle root hash	9824bf1b6f2dde3d0349326e98a3112735ec0dfcc877e40951044af0230284ad
Block hash	00002385fa655665cd5cd68c9f9162f1c4b5aec957e871581aa78f096642b9f
Nonce	16065
Timestamp	2020-07-06 19:27:46 (9 minutes ago)
Confirmations	9 confirmations

[Previous block](#)
[Next block](#)

Transactions

Re-calculated merkle root hash: 9824bf1b6f2dde3d0349326e98a3112735ec0dfcc877e40951044af0230284ad

Transactions are not tampered

Showing page 1 of 1

#	Voter ID	Cand	Timestamp	Hash

Gambar 4.35 Data 9 Confirmation

34. Data Detail 1 Confirmation

Pada halaman ini, data dari sebuah *block* yang sudah terantai akan dengan 9 konfirmasi dari data sebelumnya, data yang ditampilkan yaitu:

1. *Previous hash*, atau enkripsi dari data sebelumnya, jika data merupakan data pertama maka hanya akan terlihat angka nol sepanjang angka enkripsi *block*.
2. *Merkle root hash* atau enkripsi dari mekanisme sebuah *block* dibuat.
3. *Block hash* atau enkripsi dari data voter.
4. *Nonce* atau *proof of work* dari *block*.
5. *Timestamp* atau waktu pada saat *block* dibuat.
6. Dan *confirmation* atau *block* yang dikonfirmasi dari *block* sebelumnya, jika *block* sebelumnya ada 9 *block*, maka *confirmation* hanya akan muncul 8 karena *block* yang sedang dilihat sudah di konfirmasi

7. Tampilan dari *blockchain* data detail ini ditunjukkan pada gambar 4.36 dibawah.

Block #10

[Back to blocks list](#) [Back to homepage](#)

Previous hash	0000fc8efdfac47bb241e4db546ebbd83129b78e9d894f9fca9b8fe51de24c57
Merkle root hash	8bc0e325e4b0b25d7240e84902a734ead43b8416f0093f55daa7a11bc38a6be1
Block hash	0000a16ab2760ee0f2b245068d5b301a064beee2a1e616423df8b1f0bee3227d
Nonce	134849
Timestamp	2020-07-06 19:28:23 (11 minutes ago)
Confirmations	1 confirmations

[Previous block](#)

Transactions

Re-calculated merkle root hash: 8bc0e325e4b0b25d7240e84902a734ead43b8416f0093f55daa7a11bc38a6be1

Transactions are not tampered

Showing page 1 of 1

#	Voter ID	Cand	Timestamp	Hash
1	...	...	...	...

Gambar 4.36 Data 1 Confirmation

35. Validasi Data

Setelah semua *voter* melakukan *voting*, mereka akan mendapat kode dimana kode itu adalah hasil transaksi dari proses voting yang mereka lakukan. Kode ini juga berfungsi sebagai data validasi apakah *voter* sudah melakukan voting atau belum, apakah *voter* memilih kandidat sesuai dengan pilihannya atau belum, dan apakah voter memilih kandidat lebih dari satu atau tidak.

Kode ini dapat dicek oleh admin atau petugas untuk pengecekan data dan validasi dengan cara melihat kedalam databasenya, pengawas akan mensinkronkan data *voter* yang sudah memilih dengan data transaksi yang ada.

Gambar dari kode transaksi disajikan dalam gambar 4.37 sebagai berikut.

ID	id_pencalon	id_pemilih	tanda_terima
3	12	8	3BZUNWXFCY
4	13	9	4CATLJQDHK
5	12	1	5NVRALQJPU
6	14	2	6UEQJVZNKG
7	14	3	7HVSMCIYPW
8	14	4	8DERLOTYZN
9	14	10	9ULAIPXBHD
10	14	13	10ZCSUBTFD

Gambar 4.37 Data Tanda Terima

4.2 Pengujian

Pengujian *Black Box* ini mencakup semua fungsi dan tampilan pengguna yang ada dalam aplikasi. Hasil pengujian yang telah dilakukan ditunjukkan pada tabel 4.1

Tabel 4.1 Hasil Pengujian

Objek	Skenario	Hasil Yang Diharapkan	Hasil Pengujian	Keterangan
Halaman Utama	pengguna dapat masuk ke halaman utama dan membaca fitur dan tata cara pemilihan	pengguna memasuki halaman utama dan melihat fitur aplikasi dan tata cara pemilihan	pengguna dapat memasuki halaman utama dan melihat fitur aplikasi dan tata cara pemilihan	Berhasil
Halaman Login Admin	Admin dapat melakukan login ke dalam sistem	halaman utama admin dapat ditampilkan	halaman utama admin dapat ditampilkan dan semua fitur berfungsi	Berhasil
Halaman Login Voter	Voter dapat login ke dalam sistem	voter dapat login dan masuk ke halaman pemilihan kandidat	voter dapat login dan masuk ke halaman pemilihan kandidat	Berhasil

Halaman login Pengawas	pengawas dapat login ke dalam sistem	pengawas dapat login dan masuk halaman utama pengawas	pengawas dapat login dan masuk halaman pengawas	Berhasil
Halaman ubah password	pengguna masuk kehalaman ubah password dan mengubah password lama	pengguna masuk ke halaman ubah password dan dapat mengubah passwordnya	pengguna dapat masuk ke halaman ubah password dan dapat mengubah passwordnya	Berhasil
Halaman Logout	pengguna dapat keluar halaman utama masing-masing pengguna	pengguna kembali ke halaman login	pengguna dapat kembali ke halaman login	Berhasil
Halaman Lihat hasil sementara	pengguna memasuki halaman hasil sementara dan melihat data	pengguna memasuki halaman hasil sementara dan melihat data	pengguna dapat memasuki halaman hasil sementara dan melihat data	Berhasil
VOTER				
Halaman Pemilihan kandidat	Voter memasuki halaman pemilihan dan memilih kandidat	voter dapat memilih kandidat yang diinginkan	voter dapat memilih kandidat yang diinginkan	Berhasil
Halaman setelah memilih kandidat	setelah memilih voter ke halaman selanjutnya dan mendapat kode	voter dapat kehalaman selanjutnya dan mendapat kode	voter dapat ke halaman selanjutnya dan mendapat kode	Berhasil
ADMIN				
Halaman data voter	admin memasuki halaman data voter dan mengaudit data voter	admin memasuki halaman data voter dan dapat mengaudit data voter dan mencetaknya	admin dapat memasuki halaman data voter dan dapat mengaudit data voter dan mencetaknya	Berhasil
Halaman data kandidat	admin memasuki halaman data kandidat dan mengaudit data kandidat	admin memasuki halaman data kandidat dan mengaudit data kandidat	admin dapat memasuki halaman data kandidat dan dapat mengaudit data dan mencetaknya	Berhasil
PENGAWAS				

Halaman data voter	pengawas memasuki halaman data voter dan mengeceknya	pengawas memasuki halaman data voter dan mengeceknya	pengawas dapat masuk ke halaman data voter dan mengeceknya dan bisa mencetak data voter	Berhasil
Halaman data kandidat	pengawas memasuki halaman data kandidat dan mengeceknya	pengawas memasuki halaman data kandidat dan mengeceknya	pengawas dapat memasuki halaman data kandidat dan mengeceknya, lalu mencetak datanya	Berhasil
Background Process				
Halaman utama	pengguna masuk halaman utama dan melihat fitur dihalaman utama	pengguna memasuki halaman utama dan melihat fitur utama	pengguna dapat memasuki halaman utama dan melihat fitur utama	Berhasil
Halaman Block	pengguna melihat proses pembuatan block	pengguna dapat membuat block dari data yang dimasukan	pengguna dapat membuat block cari data yang dimasukan	Berhasil
Halaman Signature Verification	Pengguna memasuki halaman verifikasi setelah submit data	pengguna dapat melihat data	pengguna dapat melihat data dan melakukan segel data menjadi block	Berhasil
Halaman data voter	pengguna memasuki halaman data voter dan melihat data voter	pengguna memasuki halaman data voter dan melihat data voter	pengguna dapat memasuki halaman data voter dan melihat data voter	Berhasil
Halaman data Blockchain	pengguna memasuki halaman data Blockchain	pengguna memasuki halaman data Blockchain dan melihat datanya	pengguna dapat memasuki halaman data blockchain dan melihat datanya	Berhasil

Berdasarkan pengujian *Black Box* yang telah dilakukan didapat hasil bahwa semua halaman yang dibuat pada aplikasi *e-vote* utama dapat diakses dengan baik oleh pengguna dari berbagai level pemakai seperti *voter*, admin, dan pengawas. Begitu juga dengan aplikasi *background* proses yang dapat digunakan dengan baik dan seluruh sistem berjalan tanpa ada kesalahan sistem.

Hasil akhir dari aplikasi ini juga diuji di beberapa perangkat lain. Hasil dari pengujian di beberapa perangkat lainnya ini ditampilkan pada tabel 4.2 di bawah.

Tabel 4.2 Pengujian Pada Perangkat Lain

No	Nama Device	Spesifikasi	Hasil Pengujian
1	Komputer	Processor Intel Hd G 4400, RAM 4 Gb, Harddisk 500 Gb, Windows 10	Berjala Dengan Baik
2	Laptop HP	Processor Intel I3, RAM 2 Gb, Harddisk 500 Gb, Windows 10	Berjala Dengan Baik
3	Laptop asus	Processor Intel i3, RAM 6GB, harddisk 500gb, Windows 10	Berjala Dengan Baik

BAB V

PENUTUP

5.1. Kesimpulan

Berdasarkan hasil implementasi dan pengujian yang dilakukan terhadap sistem yang dibuat maka dapat ditarik kesimpulan sebagai berikut:

1. Aplikasi website e-vote memudahkan dalam membuat pemilihan umum secara online
2. Teknologi BlockChain yang diterapkan membuat aplikasi menjadi lebih aman
3. Teknologi BlockChain ini dapat digunakan pada aplikasi website electronic voting
4. Aplikasi e-vote membuat pengembangan teknologi BlockChain ini menjadi lebih baik dari segi penerapan

5.2. Saran

Penelitian ini masih dapat dikembangkan. Beberapa saran yang dapat dikembangkan untuk penelitian ini sebagai berikut:

1. Membuat panel informasi pada kandidat
2. Membuat tabel cetak data lebih teratur
3. Penjelasan pada halaman utama lebih interaktif
4. Membuat enkripsi hanya pada data block saja

DAFTAR PUSTAKA

- [1] “Pengertian Teknologi Blockchain”.[duniafintech.com](https://duniafintech.com/pengertian-teknologi-blockchain-adalah/).Maret 21, 2018.
<https://duniafintech.com/pengertian-teknologi-blockchain-adalah/>
- [2] “apa itu Distributed Database Management System dan Keamanan”.[medium.com](https://medium.com/@ibay26bf/apa-itu-distributed-database-management-system-dan-keamanan-dalam-sistem-informasi-12f8a40d1556).Juni 8, 2018.<https://medium.com/@ibay26bf/apa-itu-distributed-database-management-system-dan-keamanan-dalam-sistem-informasi-12f8a40d1556>
- [3] Yusrianto Toisutta, Eka. 2013. Pengaksesan Daun Secara Random Pada Hash Tree. Jurnal, Bandung. Institut Teknologi Bandung
- [4] Ayed, A. B. 2017. A Conceptual Secure Blockchain-Based Electronic Voting System. International Journal of Network Security & Its Applications, 9(3): 1-9.[doi:10.5121/ijnsa.2017.9301](https://doi.org/10.5121/ijnsa.2017.9301)
- [5] Conte de Leon, Daniel & Q. Stalick, Antonius& Jillepalli, Ananth & A. Haney, Michael & Sheldon, F.T. 2017. Blockchain: properties and misconceptions. Asia Pacific Journal of Innovation and Entrepreneurship. 11: 286-300. [doi:10.1108/APJIE-12-2017-034](https://doi.org/10.1108/APJIE-12-2017-034).
- [6] Liu, Y., & Wang, Q. 2017. An E-voting Protocol Based on Blockchain. IACR Cryptology ePrint Archive, 2017, 1043.
- [7] Maya Kadarina, Trie. 2019. Pengenalan Bahasa Pemrograman Python Menggunakan Aplikasi Games Untuk Siswa/I di Wilayah Kembangan Utara. Jurnal, Jakarta. Universitas Mercu Buana Jakarta
- [8] Destiningrum, Mara. 2017. SISTEM INFORMASI PENJADWALAN DOKTER BERBASIS WEB DENGAN MENGGUNAKAN FRAMEWORK CODEIGNITER(STUDI KASUS: RUMAH SAKIT YUKUM MEDICAL CENTRE). Jurnal, Bandar Lampung. Universitas Teknokrat Indonesia
- [9] Pengenalan Dasar Django.[www.petanikode.com](https://www.petanikode.com/django-untuk-pemula/).24 agustus 2016.
<https://www.petanikode.com/django-untuk-pemula/>

- [10] PyCryptodome. [pycryptodome.readthedocs.io](https://pycryptodome.readthedocs.io/en/latest/src/introduction.html). 25 desember 2019.
- [11] Yoan Rizaldi, Dimas. 2019. Implementasi Multichain sebagai Alternatif Solusi Keamanan dan Privasi Data pada Komunikasi Perangkat Pintar Rumah. Jurnal, Surabaya. Universitas Negeri Surabaya
- [12] Setiadi, Angga. Harihayati, Tati. 2016. PENERAPAN SQLITE PADA APLIKASI PENGATURAN WAKTU UJIAN DAN PRESENTASI. Jurnal, Jakarta. Universitas Komputer Indonesia
- [13] Azwanti, N. 2017. Perancangan E-Voting berbasis Web. Jurnal Komputer Terapan, 3(2): 119-132.
- [14] Noizat, P. 2015. Chapter 22 -Blockchain Electronic Vote. Chuen, D. L.K. (Ed.) Handbook of Digital Currency (pp 453-461). San Diego: Academic Press.
- [15] Pinsard, L. 2018. Deploy Your First Ethereum Smart Contract on a Blockchain!Retrieved from <https://blog.theodo.fr/2018/01/deploy-first-ethereum-smart-contract-blockchain/>
- [16] Public, Private and Consortium Blockchains. 2019. Retrieved from <https://www.draglet.com/blockchain-services/blockchain-technology/private-or-public-blockchain/>
- [17] Nakamoto.S , “Bitcoin : A Peer-to-Peer Electronic Cash System,” pp. 1–9, 2008.
- [18] Miller.P , “The cryptocurrency enigma,” in Digital Forensics: Threatscape and Best Practices, no. August 2015, Elsevier Inc., 2015, pp. 1–25.
- [19] J. Lopez, R. Rios, F. Bao, and G. Wang, “Evolving privacy : From sensors to the Internet of Things,” Futur. Gener. Comput. Syst., 2017.
- [20] Weigand.J , “Understanding the BlockchainUsing Enterprise Ontology,” no. May, 2017.

Lampiran 1:

Metode penelitian yang dibuat dalam penelitian ini mengikuti bagan gambar 3.1, pada bagan tersebut dijelaskan beberapa tahap pada proses pengembangan penelitian, proses tersebut dibagi menjadi tahap yaitu:

1. Studi literatur, pada tahap ini penulis mengumpulkan data dari berbagai macam sumber untuk menjadi referensi penulisan dan contoh aplikasi yang akan dibuat, kebanyakan sumber berasal dari jurnal dan internet.
2. Pengumpulan data penelitian, pada tahap ini penulis mulai mengumpulkan data yang sudah dicari pada berbagai sumber, dan merangkumnya untuk melanjutkan pada tahap rancangan dan penulisan.
3. Merancang aplikasi, pada tahap ini penulis mulai merancang aplikasi, pada awal mendesain pada mockup, coding hingga hasil jadi, tahap ini didukung oleh metode waterfall.
4. Pengembangan perangkat lunak, pada tahap merancang pengembangan perangkat lunak menggunakan metode waterfall yang akan dijelaskan pada metode penelitian sub bab 2.
5. Implementasi, pada tahap ini penulis telah mengimplementasikan website ini dengan cara menggunakannya dan mencoba melihat kekurangan pada aplikasi sebelum pada tahap hasil akhir.
6. Pada tahap ini, website yang penulis rancang sudah jadi dan siap digunakan, aplikasi juga terus dikembangkan selaras dengan penggunaan aplikasi sesuai kebutuhan aplikasi, mulai dari fitur hingga tampilan.
7. Pada tahap dokumen teknis ini, penulis membuat laporan mengenai jalannya aplikasi seperti screenshot aplikasi dan pengujian blackbox.

Lampiran 2 : Source Code

Halaman utama (Manage.py)

```
#!/usr/bin/env python
import os
import sys
```

```

if __name__ == '__main__':
    os.environ.setdefault('DJANGO_SETTINGS_MODULE',
        'bbevoting_project.settings')
    try:
        from django.core.management import execute_from_command_line
        except ImportError as exc:
            raise ImportError(
                "Couldn't import Django. Are you sure it's installed and "
                "available on your PYTHONPATH environment variable? Did you "
                "forget to activate a virtual environment?"
            ) from exc
        execute_from_command_line(sys.argv)

```

Halaman Utama (Home.py)

```

{% extends 'base.html' %}
{% load static %}
{% block title_bl %}Home{% endblock %}
{% block body_bl %}

<div class="container">
    <div>
        <h1 style="padding-bottom:5%;">Blockchain background process based e-voting
simulation</h1>
    </div>
    <div class="row">
        <div class="col-md-5 menu">
            <h2>"BLOCK"</h2>
            <p>Bagian ini akan menunjukkan:</p>
            <ul>
                <li>Contoh dari suara voter</li>
                <li>Struktur suara voter dalam data</li>
                <li>Signature verification</li>
                <li>Block sealing process</li>
            </ul>
            <div class="btn-menu">
                <a class="btn btn-outline-primary btn-block btn-menu" href="{% url
'ballot:create' %}" role="button">Start</a>
            </div>
        </div>
        <div class="col-md-5 menu">
            <h2>"CHAIN"</h2>
            <p>Bagian ini akan menunjukkan:</p>

```

```

<ul>
  <li>Batch transactions ({{ tx }})</li>
  <li>Block chaining ({{ bl }})</li>
  <li>Block detail</li>
  <li>Blocks verification</li>
</ul>
<div class="btn-menu">
  <a class="btn btn-outline-primary btn-menu" href="{% url
'simulation:generate' %}" role="button">Mulai simulasi</a>
  <a class="btn btn-link btn-menu" href="{% url 'simulation:transactions' %}"
role="button">Lihat Data Transaksi</a>
  <a class="btn btn-link btn-menu" href="{% url 'simulation:blockchain' %}"
role="button">Lihat Bentuk Blok</a>
</div>
</div>
</div>
</div>
{% endblock %}

```

Halaman membuat block (create.py)

```

{% extends 'base.html' %}
{% load static %}
{% block title_bl %}Create ballot{% endblock %}
{% block body_bl %}
<div class="container" style="border: 1px solid black; background-color: #F5F5F5;">
  <h1 style="text-align:center;">Pemilihan Umum</h1>
  <form action="{% url 'ballot:create' %}" method="POST">
    {% csrf_token %}
    <div class="form-group">
      <label for="voter-id-input">Your ID:</label>
      <input name="voter-id-input" type="text" class="form-control" id="voter-
id-input" placeholder="fe021e59-c29d-4532-9e99-5c42f1910af3" value="{{ voter_id
 }}" required autofocus>
    </div>
    <div id="candidate" style="padding:20px 0;">
      <p style="text-align:center;">Daftar Kandidat akan terlihat dibawah:</p>
      <div class="row">
        <div class="col-md-4">
          <p style="text-align: center;"></p>
          <p style="text-align: center;">Kandidat satu (1)</p>
        </div>
        <div class="col-md-4">
          <p style="text-align: center;"></p>

```

```

        <p style="text-align: center;">Kandidat Dua (2)</p>
    </div>
    <div class="col-md-4">
        <p style="text-align: center;"></p>
        <p style="text-align: center;">Kandidat Tiga (3)</p>
    </div>
</div>
<div class="form-group">
    <label for="vote-input">Your candidate choice:</label>
    <input name="vote-input" type="number" class="form-control"
id="voter-input" placeholder="Enter valid candidate ID" min="1" max="3" required>
</div>
<div class="form-group">
    <label for="private-key-textarea">Your private key:</label>
    <textarea name="private-key-input" class="form-control" id="private-key-
textarea" rows="10" required></textarea>
    <small id="privatekeyHelpBlock" class="form-text text-muted">
        The matching public key must already be registered.
    </small>
</div>
<button type="submit" class="btn btn-primary btn-block">Submit ballot</button>
</form>
</div>
{% endblock %}

```

Halaman Block Data (seal.py)

```

{% extends 'base.html' %}
{% load static %}
{% load sim_filters %}
{% block title_bl %}Block detail{% endblock %}
{% block body_bl %}
    <div class="container">
        <h1>Block detail</h1>
        <div style="padding:3%; border: 1px solid black;">
            <dl class="row">
                <dt class="col-sm-3">Previous hash</dt>
                <dd class="col-sm-9">{{ prev_hash }}</dd>

                <dt class="col-sm-3">Transaction hash</dt>
                <dd class="col-sm-9"><code>{{ transaction_hash }}</code></dd>

                <dt class="col-sm-3">Block hash</dt>
                <dd class="col-sm-9"><code>{{ block_hash }}</code></dd>
            </dl>
        </div>
    </div>
{% endblock %}

```

```

        <dt class="col-sm-3">Nonce</dt>
        <dd class="col-sm-9">{{ nonce }}</dd>

        <dt class="col-sm-3">Timestamp</dt>
        <dd class="col-sm-9">{{ timestamp }} ({{ timestamp|unix_to_date|date:"Y-
m-d H:i:s" }})</dd>
    </dl>
    <a class="btn btn-primary btn-small" href="{% url 'welcome:home' %}"
role="button">Back to homepage</a>
</div>
</div>
{% endblock %}

```

Halaman Signature Verification (status.py)

```

{% extends 'base.html' %}
{% block title_bl %}Status{% endblock %}

{% block body_bl %}
<div class="container">
    <h1>Signature Verification</h1>
    <div style="padding:3%; border:1px solid black;">
        <p>Your ballot was <code>{{ ballot }}</code></p>
        <p>Generated signature <code>{{ signature }}</code></p><br>
        <p style="color:{% if error %}red{% else %}green{% endif %};">
            {{ status }}
        </p>

        {% if not error %}
        <form action="{% url 'ballot:seal' %}" method="POST">
            {% csrf_token %}
            <input name="ballot_input" type="hidden" value="{{ ballot }}">
            <button type="submit" class="btn btn-primary btn-block">Seal the
ballot</button>
        </form>
        {% endif %}
    </div>
</div>

{% endblock %}

a class="btn btn-light btn-sm" href="?page={{ votes_pg.paginator.num_pages }}"
role="button">Last &raquo;</a>
    {% endif %}
</span>

```

```

</div>

<table class="table table-hover">
  <thead>
    <tr>
      <th scope="col">#</th>
      <th scope="col">Voter ID</th>
      <th scope="col">Vote</th>
      <th scope="col">Timestamp</th>
      <th scope="col">Hash</th>
      <th scope="col">Block</th>
    </tr>
  </thead>
  <tbody>
    {% for ballot, h, bh in votes %}
      <tr>
        <td>{{ forloop.counter }}</td>
        <td><code>{{ ballot.id }}</code></td>
        <td>{{ ballot.vote }}</td>
        <td>{{ ballot.timestamp|unix_to_date|date:"Y-m-d H:i:s" }}</td>
        <td title="{{ h }}"><code>{{ h|truncatechars:20 }}</code></td>
        <td><a class="btn btn-light btn-sm" href="{% url 'simulation:block_detail'
block_hash=bh %}" role="button">go to block</a></td>
      </tr>
      {% empty %}
        <p>No record.</p>
      {% endfor %}
    </tbody>
  </table>
</div>
{% endblock %}

```