

Revisi terdapat di halaman:

- 44
- 45
- 61
- 86



UNIVERSITAS INDONESIA

**IMPLEMENTASI TEKNOLOGI *BLOCKCHAIN* PADA SISTEM
*CUSTOM E-VOTING***

SKRIPSI

Muhamad Fahriza Novriansyah

1706042951

**FAKULTAS TEKNIK
PROGRAM STUDI TEKNIK KOMPUTER**

DEPOK

JUNI 2020



UNIVERSITAS INDONESIA

**IMPLEMENTASI TEKNOLOGI *BLOCKCHAIN* PADA SISTEM
CUSTOM E-VOTING**

SKRIPSI

**Diajukan sebagai salah satu syarat untuk memperoleh gelar
Sarjana Teknik**

Muhamad Fahriza Novriansyah

1706042951

**FAKULTAS TEKNIK
PROGRAM STUDI TEKNIK KOMPUTER**

**DEPOK
JUNI 2020**

HALAMAN PERNYATAAN ORISINALITAS

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

Nama : Muhamad Fahriza Novriansyah

NPM : 1706042951

Tanda Tangan :



Tanggal : 24 Juni 2020

LEMBAR PENGESAHAN

Skripsi dengan judul:

IMPLEMENTASI TEKNOLOGI BLOCKCHAIN PADA SISTEM CUSTOM E-VOTING

dibuat untuk melengkapi sebagian persyaratan menjadi Sarjana Teknik pada Program Studi Teknik Komputer, Departemen Teknik Elektro, Fakultas Teknik Universitas Indonesia dan disetujui untuk diajukan dalam presentasi skripsi.

Depok, 24 Juni 2020

Dosen Pembimbing,



Ruki Harwahu ST.MT,MSc.,Ph.D

NIP.100120910231409891

KATA PENGANTAR

Puji syukur saya panjatkan kehadirat Tuhan Yang Maha Esa, karena berkat rahmat dan karunia-Nya saya dapat menyelesaikan laporan skripsi ini. Penulisan skripsi ini dilakukan dalam rangka memenuhi salah satu kewajiban mahasiswa S1 untuk memenuhi mata kuliah wajib di Departemen Teknik Elektro untuk memperoleh gelar Sarjana Teknik dengan Program Studi Teknik Komputer pada Fakultas Teknik Universitas Indonesia

Dalam proses dan penyusunan laporan ini, penulis mendapat bimbingan, pengarahan, dan dukungan dari berbagai pihak. Oleh karena itu, penulis ingin mengucapkan terima kasih kepada pihak-pihak yang telah membantu pelaksanaan dan penyusunan laporan skripsi ini ini, yaitu :

1. Allah SWT yang telah memberikan rahmat dan hidayah-Nya kepada sehingga laporan skripsi ini dapat disusun.
2. Keluarga yang senantiasa memberikan doa dan dukugan.
3. Dr. Ruki Harwahu, S.T, M.T, M.Sc dosen pembimbing yang telah membimbing dan menyediakan waktu, tenaga dan pikiran dalam penyusunan skripsi ini.
4. Dr. Ir. Dodi Sudiana, M.Eng.selaku dosen pembimbing akademis yang telah membantu dalam akademis dan selalu memberi kesempatan berkembang penulis selama masa perkuliahan.
5. Semua dosen pengajar Departemen Teknik Elektro Universitas Indonesia yang telah memberikan ilmu yang berguna dan bermanfaat.
6. Teman-teman jurusan Teknik Komputer Universitas Indonesia yang telah membantu dalam proses penyelesaian laporan ini.
7. Serta pihak lain yang tidak dapat disebutkan satu persatu yang telah membantu.

Penulis menyadari bahwa dalam penulisan laporan ini masih terdapat banyak kekurangan. Oleh karena itu, penulis mengharapkan kritik dan saran yang dapat membantu penulis demi kesempurnaan penyusunan laporan skripsi ini. Penulis

berharap laporan ini dapat bermanfaat bagi penulis dan pembaca pada umumnya.
Semoga menjadi ilmu yang bermanfaat.

Depok, Juni 2020

Muhamad Fahriza Novriansyah

ABSTRAK

Nama : Muhamad Fahriza Novriansyah
Program Studi : Teknik Komputer
Judul : IMPLEMENTASI TEKNOLOGI *BLOCKCHAIN* PADA
SISTEM CUSTOM E-VOTING

Penggunaan sampah kertas yang berlebih pada saat pemilihan umum merupakan perilaku konsumtif dan dapat merusak kelestarian lingkungan. Pada Pilkada 2018, terdapat 54 juta kertas suara yang tidak terpakai akibat masih minimnya partisipasi masyarakat dalam setiap Pilkada. Selain itu terjadinya berbagai manipulasi surat suara menjadikan berkurangnya integritas data. Aplikasi *Custom E-Voting* merupakan aplikasi sistem pemungutan suara elektronik untuk pemilihan umum dengan penerapan teknologi *Blockchain* untuk menambah keamanan menggunakan platform *Hyperledger Fabric* dan *Hyperledger Composer*. Penerapan teknologi ini dapat menggantikan sistem pemilihan berbasis kertas dan database konvensional. Aplikasi ini menggunakan database yang terdistribusi sehingga database lebih aman dan terjaga integritasnya. Hasil dari *E-Voting* dapat dilihat oleh pemilih sehingga transparansi bagi pemilih. Aplikasi ini dapat di implementasikan di beberapa pemilihan yang membutuhkan transparansi hasil dengan waktu pemilihan dapat diatur sesuai kebutuhan. Adapun masalah yang menjadi fokus pada penelitian ini adalah *proof of concept* teknologi blockchain dengan tools yang ada saat ini dan pengujian performa sistem. Diharapkan hasil penelitian ini dapat memberikan gambaran viabilitas teknologi blockchain untuk menangani kasus seperti Pemilu yang bersifat kritis, masif, dan serentak.

Kata Kunci:

Blockchain; E-Vote; Hyperledger;

ABSTRACT

Name : Muhamad Fahriza Novriansyah
Study Program : Teknik Komputer
Title : IMPLEMENTATION OF *BLOCKCHAIN* TECHNOLOGY IN
CUSTOM E-VOTING SYSTEM

Excessive use of paper waste during the general elections is a consumptive behavior and can damage the environment. In the 2018 Pilkada, there were 54 million unused ballots due to the lack of public participation in each Pilkada. In addition, various ballot manipulations make reduced data integrity. The *Custom E-Voting* Application is an electronic voting system application for general elections with the application of *Blockchain* technology to increase security using the *Hyperledger Fabric* and *Hyperledger Composer* platforms. The application of this technology can replace conventional paper-based voting systems and databases. This application uses a distributed database so that the database is more secure and maintained its integrity. The results of E-Voting can be seen by voters so that transparency for voters. This application can be implemented in a number of elections that require transparency of the results with the election time can be set as needed. The problem that is the focus of this research is the proof of the concept of blockchain technology with existing tools and system performance testing. It is hoped that the results of this research can provide an overview of the viability of blockchain technology to handle cases such as the general elections that are critical, massive, and simultaneous.

Key words:

Blockchain; E-Vote; Hyperledger;

DAFTAR ISI

HALAMAN PERNYATAAN ORISINALITAS	ii
LEMBAR PENGESAHAN	iii
KATA PENGANTAR.....	iv
ABSTRAK	vi
ABSTRACT	vii
DAFTAR ISI.....	viii
DAFTAR GAMBAR.....	xi
DAFTAR TABEL	xv
BAB 1 PENDAHULUAN	16
1.1. Latar Belakang	16
1.2. Rumusan Masalah	18
1.3. Tujuan Penelitian	18
1.4. Batasan Masalah.....	18
1.5. Metodologi Penelitian	19
1.6. Sistematika Penulisan	20
BAB 2 LANDASAN TEORI	21
2.1. Pemilu dan Permasalahanya.....	21
2.2. Blockchain.....	22
2.2.1. Kategorisasi Blockchain.....	22
2.2.1.1 <i>Permissionless</i>	23
2.2.1.2 <i>Permissioned</i>	23
2.2.2. Komponen Blockchain.....	24
2.2.2.1 Fungsi Hash Kriptografis	24
2.2.2.2 Transaksi	25
2.2.2.3 Kriptografi Kunci Asimetrik	26
2.2.2.4 Alamat dan Derivasi Alamat	26
2.2.2.5 Penyimpanan Private Key	27
2.2.2.6 Ledgers	27
2.2.2.7 Blok	28
2.2.2.8 Rantai Blok	28
2.2.3. Model Konsensus	28
2.2.4. <i>Smart Contracts</i>	30
2.3. E-Voting dan Blockchain	31

2.4.	Hyperledger.....	33
2.4.1.	Filosofi Desain Hyperledger	33
2.4.2.	Hyperledger Greenhouse.....	34
2.4.3.	Hyperledger Framework	35
2.4.4.	Hyperledger Tools.....	36
2.5.	Entity Relationship Diagram (ERD)	37
2.5.1.	Notasi ERD	38
2.6.	Unified Modeling Language (UML).....	40
2.6.1.	Diagram UML <i>structural</i>	40
2.6.2.	Diagram UML <i>behavioral</i>	42
2.7.	Angular.....	43
BAB 3 PERANCANGAN SISTEM.....		44
3.1.	Spesifikasi Software dan Hardware	44
3.2.	Diagram Alir, ERD dan UML Sistem.....	46
3.2.1.	Diagram Alir	46
3.2.2.	Entity Relationship Diagram.....	48
3.2.3.	Diagram <i>Use Case</i>	49
3.3.	Perancangan Business Network Blockchain dengan Hyperledger	50
3.3.1.	Penentuan Partisipan	50
3.3.2.	Penentuan <i>Asset</i>	51
3.3.3.	Penentuan Transaksi.....	52
3.3.4.	Penentuan <i>Logic File</i>	53
3.3.5.	Penentuan Akses Kontrol.....	58
BAB 4 IMPLEMENTASI DAN PEMBAHASAN.....		61
4.1.	Antarmuka Sistem dengan Angular	61
4.1.1.	Antarmuka Login untuk Partisipan melakukan	61
4.1.2.	Antarmuka untuk partisipan Admin.....	62
4.1.2.1	Antarmuka Transaksi Inisiasi Pemilihan	62
4.1.2.2	Antarmuka Transaksi Inisiasi Kandidat	63
4.1.2.3	Antarmuka Transaksi Akhiri Pemilihan.....	64
4.1.3.	Antarmuka untuk partisipan Pemilih	65
4.1.3.1	Antarmuka Transaksi Pilih Kandidat	65
4.2.	Pengujian.....	67
4.2.1.	Pengujian Aktivitas Partisipan berdasarkan Hak Akses.	67
4.2.1.1	Pengujian hak akses Admin	67
4.2.1.2	Pengujian hak Akses Pemilih.....	76
4.2.2.	Pengujian Performa berdasarkan Transaksi	84
4.2.3.	Pengujian Performa Memilih Kandidat	86
4.2.4.	Pengujian Performa perintah Create dan Update Pemilih pada Database Blockchain dan Database MySQL	89

BAB 5 PENUTUP.....	95
5.1. Kesimpulan	95
5.2. Saran.....	95
DAFTAR PUSTAKA	97
LAMPIRAN.....	101

DAFTAR GAMBAR

Gambar 2-1 Contoh Transaksi Cryptocurrency [12]	26
Gambar 2-2 Rantai Blok Generik [12].....	28
Gambar 2-3 Struktur Greenhouse Hyperledger [17].....	34
Gambar 2-4 Contoh Entity Relationship Diagram [19]	38
Gambar 2-5 Contoh Hubungan <i>One-to-one</i> [19]	39
Gambar 2-6 Contoh Hubungan <i>One-to-many</i> [19]	39
Gambar 2-7 Contoh Hubungan <i>many-to-many</i> [19]	40
Gambar 2-8 Contoh Diagram <i>Package</i> [21]	41
Gambar 2-9 Contoh Diagram <i>Use Case</i> [21]	42
Gambar 3-1 Topologi <i>Deployment</i> Ideal/ Realistik	44
Gambar 3-2 Topologi Ujicoba	44
Gambar 3-3 Diagram Alir Sistem	47
Gambar 3-4 Entity Relationship Diagram Sistem.....	48
Gambar 3-5 Diagram Use Case Sistem.....	49
Gambar 3-6 Logika fungsi untuk membuat id	53
Gambar 3-7 Logika Fungsi untuk membuat asset Pemilihan	54
Gambar 3-8 Logika Fungsi untuk membuat asset Kandidat.....	55
Gambar 3-9 Logika Fungsi untuk memilih Kandidat dalam Pemilihan	56
Gambar 3-10 Logika untuk mengakhiri Pemilihan.....	57
Gambar 4-1 Antarmuka Login untuk Partisipan sebelum melakukan transaksi...	61

Gambar 4-2 Antarmuka Data Pemilihan sebelum admin melakukan transaksi Inisiasi Pemilihan	62
Gambar 4-3 Antarmuka Data Pemilihan saat Admin akan melakukan transaksi Inisiasi Pemilihan dengan mengisi data yang dibutuhkan	62
Gambar 4-4 Antarmuka Data Pemilihan setelah admin melakukan berhasil transaksi Inisiasi Pemilihan dimana isi dari Data Pemilihan bertambah.....	63
Gambar 4-5 Antarmuka Data Kandidat sebelum admin melakukan transaksi Inisiasi Kandidat	63
Gambar 4-6 Antarmuka Data Kandidat saat Admin akan melakukan transaksi InisiasiKandidat dengan mengisi data yang dibutuhkan.....	63
Gambar 4-7 Antarmuka Data Kandidat setelah admin melakukan berhasil transaksi Inisiasi Kandidat dimana isi dari Data Kandidat bertambah.....	64
Gambar 4-8 Antarmuka Data Pemilihan sebelum dan saat admin akan melakukan transaksi Akhiri Pemilihan	64
Gambar 4-9 Antarmuka Data Pemilihan setelah admin melakukan berhasil transaksi akhiri Pemilihan.	65
Gambar 4-10 Antarmuka Pilih Kandidat sebelum Pemilih melakukan transaksi Pilih Kandidat	65
Gambar 4-11 Antarmuka Pilih Kandidat saat Pemilih saat akan melakukan transaksi Pilih Kandidat denga menekan button Pilih sesuai dengan Kandidat yang ingin dipilih.	66
Gambar 4-12 Antarmuka Pilih Kandidat saat Pemilih berhasil melakukan transaksi Pilih Kandidat ketika pemilihan belum berakhir	66
Gambar 4-13 Antarmuka Pilih Kandidat saat Pemilih berhasil saat pemilihan sudah berakhir	67
Gambar 4-14 Admin Menambah dan Membaca Admin.....	68

Gambar 4-15 Admin Memperbaharui Admin.....	68
Gambar 4-16 Admin Menghapus Admin.....	69
Gambar 4-17 Admin Menambah dan Membaca Partisipan Pemilih	70
Gambar 4-18 Admin Memperbaharui Partisipan Pemilih	70
Gambar 4-19 Admin Menghapus Partisipan Pemilih	70
Gambar 4-20 Menambah dan Membaca Asset Kandidat	71
Gambar 4-21 Admin Memperbaharui Asset Kandidat	71
Gambar 4-22 Admin Menghapus Asset Kandidat	72
Gambar 4-23 Admin Menambah dan Membaca Asset Pemilihan.....	72
Gambar 4-24 Admin Memperbaharui Asset Pemilihan.....	73
Gambar 4-25 Admin Menghapus Asset Pemilihan.....	73
Gambar 4-26 Admin Melakukan Transaksi Inisiasi Pemilihan	74
Gambar 4-27 Admin Melakukan Transaksi Inisiasi Kandidat.....	74
Gambar 4-28 Admin Melakukan Transaksi Pilih Kandidat.....	75
Gambar 4-29 Admin Melakukan Transaksi Akhiri Pemilihan	75
Gambar 4-30 Pemilih Membaca dan Menambah Partisipan Admin	76
Gambar 4-31 Pemilih Menambah dan Menghapus Partisipan Pemilih	76
Gambar 4-32 Pemilih Membaca Partisipan Pemilih	77
Gambar 4-33 Pemilih Memperbaharui Partisipan Pemilih	77
Gambar 4-34 Pemilih Menambah dan Menghapus Asset Kandidat	78
Gambar 4-35 Pemilih Membaca Asset Kandidat.....	78
Gambar 4-36 Pemilih Memperbaharui Asset Kandidat.....	79

Gambar 4-37 Pemilih Menambah, Memperbaharui dan Menghapus Asset Pemilihan	79
Gambar 4-38 Pemilih Membaca Asset Kandidat.....	80
Gambar 4-39 Admin Melakukan Transaksi Inisiasi Pemilihan	80
Gambar 4-40 Pemilih Melakukan Transaksi Inisiasi Kandidat	81
Gambar 4-41 Pemilih Melakukan Transaksi Pilih Kandidat	81
Gambar 4-42 Pemilih Melakukan Transaksi Akhiri Pemilihan.....	82
Gambar 4-43 Jumlah Waktu yang dibutuhkan dengan Percobaan 100 Transaksi untuk Membuat Kandidat.....	85
Gambar 4-44 Perbandingan waktu yang transaksi Pilih Kandidat secara Serial dan Secara Parallel dengan <i>BatchTimeout</i> 250 ms dan 2000ms.....	88
Gambar 4-45 Jumlah Waktu yang dibutuhkan dengan Percobaan 50 Transaksi untuk membuat Pemilih dengan MySQL.....	90
Gambar 4-46 Jumlah Waktu yang dibutuhkan dengan Percobaan 50 Transaksi untuk membuat Pemilih dengan Blockchain.....	91
Gambar 4-47 Perbandingan Jumlah Waktu yang dibutuhkan dengan Percobaan 50 Transaksi untuk membuat Pemilih dengan Blockchain dan MySQL	91
Gambar 4-48 Jumlah Waktu yang dibutuhkan dengan Percobaan 50 Transaksi untuk memperbaharui Pemilih dengan MySQL	92
Gambar 4-49 Perbandingan Jumlah Waktu yang dibutuhkan dengan Percobaan 50 Transaksi untuk memperbaharui Pemilih dengan Blockchain.....	93
Gambar 4-50 Perbandingan Jumlah Waktu yang dibutuhkan dengan Percobaan 50 Transaksi untuk memperbaharui Pemilih dengan Blockchain dan MySQL.....	93

DAFTAR TABEL

Tabel 2-1 Contoh Teks Input dan Nilai-nilai SHA-256 Digest Value.....	25
Tabel 2-2 Solusi berbasis blockchain untuk pemungutan suara [17].....	33
Tabel 3-1 Daftar Spesifikasi <i>Hardware</i> dan <i>Software</i> pada sistem	46
Tabel 3-2 Daftar Partisipan	50
Tabel 3-3 Daftar <i>Asset</i> dan <i>Value</i>	51
Tabel 3-4 Daftar transaksi dan hak partisipan.....	52
Tabel 3-5 Daftar Hak Akses User terhadap partisipan.....	58
Tabel 3-6 Daftar Hak Akses Partisipan terhadap Transaksi	58
Tabel 3-7 Daftar Hak Akses Partisipan terhadap Asset	58
Tabel 3-8 Daftar Aktivitas dan Hak akses Partisipan	60
Tabel 4-1 Hasil Pengujian Fungsional Aktivitas Partisipan	84
Tabel 4-2 Hasil Pengujian Kecepatan 100 Transaksi Membuat Kandidat	85
Tabel 4-3 Hasil Pengujian Waktu Proses setiap Transaksi memilih Kandidat 20 Transaksi secara Serial dan 10 Transaksi Secara Parallel dengan <i>BatchTimeout</i> 250ms	86
Tabel 4-4 Hasil Pengujian Waktu proses setiap Transaksi memilih Kandidat 20 Transaksi secara Serial dan 10 Transaksi Secara Parallel dengan <i>BatchTimeout</i> 2000ms	87
Tabel 4-5 Hasil Pengujian Jumlah Waktu 50 Transaksi untuk Membuat Pemilih dengan Blockchain dan MySQL	90
Tabel 4-6 Hasil Pengujian Kecepatan 50 Transaksi Memperbaharui Pemilih dengan Blockchain dan MySQL	92

BAB 1

PENDAHULUAN

1.1. Latar Belakang

Perkembangan teknologi menerus berkembang pesat dan mengganti sistem tradisional menjadi sistem yang lebih modern. Salah satu yang berganti yaitu proses voting atau pemungutan suara. Voting merupakan salah satu hak dasar setiap warga negara negara demokrasi [1]. Voting dilakukan untuk mengekspresikan pendapat atau membuat keputusan kolektif [2]. Di beberapa negara, pemerintahan dipilih melalui pemungutan suara dimana pemilih biasa memberikan suara mereka dengan menggunakan kertas, kartu punch, dan tuas mekanik. Di Indonesia pemilihan pemerintahan menggunakan sistem tradisional atau manual dimana penggunaan paku untuk mencoblos atau membuat lubang pada kertas suara dan penghitungan manual secara berjenjang itu juga membuat hasil penghitungan atau rekapitulasi berlangsung lama [3].

Sistem tradisional yang digunakan pemerintah Indonesia belum efektif dan memiliki masalah masalah seperti penggunaan kertas yang berlebih, penghitungan suara yang lama, dan rawan kecurangan terhadap surat suara. Penggunaan Sampah Kertas yang berlebih terjadi pada saat pemilihan umum, salah satunya pada Pemilihan Serentak 2018, terdapat 54 Juta kertas suara yang tidak terpakai akibat masih minimnya partisipasi masyarakat dalam setiap pilkada [4]. Masalah kasus surat suara terpilih terjadi di TPS 42 Berlian Indah, Kompleks Berlian Indah, Desa Jenetalasa, Kecamatan Pallangga, Kabupaten Gowa, Sulawesi Selatan dimana seharusnya surat suara masih terjaga [5]. Selain itu, Ada beberapa permasalahan terjadi pada Pemilu Serentak ditahun berikutnya yaitu di tahun 2019 [6], seperti :

1. Penanganan logistik pemilu dimana secara nasional, terdapat 10.520 TPS yang mengalami kekurangan logistik pemilu. Terdapat kotak suara yang diterima KPPS tidak tersegel di 6.474 TPS. Selain itu, terdapat kasus surat suara yang tertukar antar Daerah Pemilihan atau antar-TPS berdasarkan data Bawaslu kasus ini terjadi di 3.411 TPS

2. Penanganan Data Pemilih dimana Pemutakhiran Daftar Pemilih Tetap Hasil Perbaikan Tahap yang dilakukan KPU tuntas pada 8 April 2019 dimana 9 hari sebelum Hari H Pencoblosan. Ini mundur 21 Hari dari perencanaan yang ditetapkan KPU, yaitu 19 Maret 2019. Keterlambatan KPU dikarenakan mengalami kesulitan dalam mengurutkan Data Pemilih secara komprehensif. Dampaknya masih terjadi kesalahan berupa terdaftar 26 ganda, terdaftar yang sudah meninggal, dan tidak ter-updatenya Pemilih yang pindah domisili.
3. Beban kerja KPPS, berdasarkan Data Kementerian Kesehatan, per 16 Mei 2019 sebanyak 527 Petugas KPPS meninggal dunia dan 11.239 jatuh sakit.
4. Kesalahan dalam rekapitulasi penghitungan suara Pemilu Serentak 2019, berdasarkan beberapa organisasi, mencatat ada 708 kasus rekapitulasi tersebut yang utamanya adalah data C1 yang tertukar dan kesalahan input data C1 ke dalam Sistem Perhitungan KPU

Dengan memanfaatkan kemajuan teknologi, dapat mengatasi beberapa masalah dalam sitem pemilu dan membuatnya menjadi lebih modern. Solusi E-Voting bertujuan untuk meningkatkan partisipasi, dan ~~meningkatkan~~ hasil pemilihan dengan mengatasi tantangan yang terkait dengan sistem pemilihan tradisional [7]. Namun penerapan E-Voting masih memiliki kekurangan dalam integritas data. Penerapan Teknologi Blockchain pada E-voting bisa menjadi solusi untuk mengurangi kekhawatiran dalam e-voting. Blockchain adalah basis data ~~read-and-write-only~~ yaitu yang dapat melakukan fungsi membaca dan menulis data, jika ~~ketika~~ data yang sudah ditulis di dalam blockchain maka data tersebut tidak dapat diubah [8]. Fitur-fitur dari blockchain dapat sangat berguna dalam membangun solusi e-voting yang sempurna. Aplikasi yang dibangun menggunakan blockchain didesentralisasi dan dimiliki oleh banyak pihak dan tidak ada yang dapat mengubah atau memperbarui data di blockchain. Sistem pemungutan suara berbasis blockchain dapat ~~menjaga privasi pemilih dan~~ meningkatkan aksesibilitas, dan menjaga sistem pemungutan suara tetap transparan, aman, dan hemat biaya [9]. Melalui penggunaan Blockchain, fokusnya adalah membuat proses pemilihan adil dan tanpa intervensi pihak ketiga..

Penelitian Blockchain sebelumnya oleh Wibowo, Implementasi Blockchain pada Sistem Pencatatan Hasil Rekapitulasi Pemilu berdasarkan pindaian formulir C1, menyimpulkan memiliki kelebihan pada sisi keamanan ketersediaan data dan integritas data [10].

1.2. Rumusan Masalah

Rumusan masalah dari penelitian ini adalah sebagai berikut:

1. Bagaimana perancangan dan implementasi custom e-voting untuk menggantikan sistem tradisional menggunakan teknologi blockchain?
2. Bagaimana implementasi Teknologi blockchain dengan Hyperledger Composer?
3. Bagaimana performa blockchain pada sistem e-voting dengan database blockchain dibanding database biasa?

1.3. Tujuan Penelitian

Tujuan dari penelitian ini adalah sebagai berikut:

1. Merancang dan mengimplementasikan sistem custom e-voting berbasis block chain yang lebih transparan dan lebih ramah lingkungan dengan tujuan menggantikan sistem tradisional.
2. Mengimplementasikan teknologi blockchain pada sistem custom e-voting dengan tools Hyperledger Composer.
3. Menyimulasikan dan membandingkan proses e-voting pada sistem custom e-voting yang menggunakan teknologi blockchain dengan database biasa.

1.4. Batasan Masalah

Batasan masalah dalam penelitian ini adalah sebagai berikut:

1. Implementasi teknologi blockchain pada custom e-voting
2. Sistem custom e-voting dengan teknologi blockchain berbasis aplikasi web. Simulasi dilakukan dalam jaringan lokal.

1.5. Metodologi Penelitian

Metode yang digunakan selama penelitian ini adalah sebagai berikut:

1. Studi literatur

Pada tahap ini merupakan tahap mencari referensi teori yang berhubungan dengan *e-voting*, blockchain, hyperledger, dan teknologi lainya yang berhubungan dengan topik penelitian.

2. Konsultasi dengan dosen pembimbing

Pada tahap ini merupakan tahap bimbingan setiap pekan dengan dosen pembimbing untuk mendapatkan ilmu serta berkonsultasi mengenai topik penelitian.

3. Pendefinisian *System Requirement*

Pada Tahap ini merupakan tahap perancangan sistem dengan mendefinisikan sistem dan fungsi sistem berupa aplikasi berdasarkan studi literatur.

4. Desain Sistem

Pada tahap ini merupakan tahap pembuatan desain aplikasi sebagai prototype dan pembuatan UML.

5. Implementasi Sistem

Pada tahap ini merupakan tahap pembuatan sistem sesuai dengan *system requirement* yang sudah dibuat.

6. Simulasi

Pada tahap merupakan tahap simulasi sistem yang dibuat dan memastikan sesuai dengan *system requirement*.

7. Analisis

Pada tahap ini merupakan tahap analisis performa sistem yang dibuat dengan beberapa ujicoba.

8. Kesimpulan

Pada tahap ini merupakan tahap pengambilan kesimpulan dari hasil pengujian dan analisis yang telah dilakukan dan dibuat.

1.6. Sistematika Penulisan

Sistematika Penulisan pada penelitian ini dibagi ke dalam 5 bab, yaitu:

BAB I PENDAHULUAN

Bab ini memuat latar belakang masalah, perumusan masalah, tujuan, batasan masalah, metodologi penelitian, dan sistematika penulisan laporan.

BAB II DASAR TEORI

Bab ini membahas konsep-konsep dan teori dasar tentang *E-voting*, *blockchain*, *UML*, dan *Hyperledger*.

BAB III PERANCANGAN SISTEM

Bab ini membahas tentang rancangan implementasi sistem yang akan digunakan dalam penelitian.

BAB IV IMPLEMENTASI DAN PEMBAHASAN

Bab ini membahas hasil dari implementasi dan analisa hasil sistem yang dibuat.

BAB V PENUTUP

Bab ini memuat kesimpulan penelitian dan saran tentang penelitian untuk penelitian selanjutnya.

BAB 2

LANDASAN TEORI

2.1. Pemilu dan Permasalahannya

Pemilihan Umum (Pemilu) tahun 2019 merupakan pemilu yang pertama kali di Indonesia dilakukan secara serentak, yaitu memilih anggota DPRD kabupaten/kota, DPRD provinsi, DPRRI, DPD serta presiden dan wakil presiden [11]. Pemilu Serentak diadakan pada tanggal 17 April 2019 ini menjadi catatan sejarah pemilu di Indonesia. Terdapat Substansi dari pemilu :

- Sarana kedaulatan rakyat,
- Instrumen untuk memastikan adanya transisi dan rotasi kekuasaan berjalan demokratis.
- Sarana pendorong akuntabilitas dan kontrol publik terhadap negara.

Pemilu mempunyai 4 (empat) fungsi yaitu sebagai sarana membangun legitimasi, sarana penguatan dan sirkulasi elit secara periodik, sarana menyediakan perwakilan, dan sarana pendidikan politik bagi warga masyarakat.

Ada beberapa permasalahan terjadi pada Pemilu Serentak ditahun berikutnya yaitu di tahun 2019 [6], seperti :

1. Penanganan logistik pemilu dimana secara nasional, terdapat 10.520 TPS yang mengalami kekurangan logistik pemilu. Terdapat kotak suara yang diterima KPPS tidak tersegel di 6.474 TPS. Selain itu, terdapat kasus surat suara yang tertukar antar Daerah Pemilihan atau antar-TPS berdasarkan data Bawaslu kasus ini terjadi di 3.411 TPS
2. Penanganan Data Pemilih dimana Pemutakhiran Daftar Pemilih Tetap Hasil Perbaikan Tahap yang dilakukan KPU tuntas pada 8 April 2019 dimana 9 hari sebelum Hari H Pencoblosan. Ini mundur 21 Hari dari perencanaan yang ditetapkan KPU, yaitu 19 Maret 2019. Keterlambatan KPU dikarenakan mengalami kesulitan dalam mengurutkan Data Pemilih secara komprehensif. Dampaknya masih terjadi kesalahan berupa terdaftar 26

ganda, terdaftar yang sudah meninggal, dan tidak ter-updatenya Pemilih yang pindah domisili.

3. Beban kerja KPPS, berdasarkan Data Kementerian Kesehatan, per 16 Mei 2019 sebanyak 527 Petugas KPPS meninggal dunia dan 11.239 jatuh sakit.
4. Kesalahan dalam rekapitulasi penghitungan suara Pemilu Serentak 2019, berdasarkan beberapa organisasi, mencatat ada 708 kasus rekapitulasi tersebut yang utamanya adalah data C1 yang tertukar dan kesalahan input data C1 ke dalam Sistem Perhitungan KPU

2.2. Blockchain

Blockchain adalah *distributed ledgers* atau buku besar digital dari transaksi-transaksi yang ditandatangani secara kriptografis yang dikelompokkan ke dalam blok-blok [12]. Setiap blok dikaitkan secara kriptografis dengan yang blok sebelumnya (membuatnya lebih jelas) setelah validasi dan menjalani keputusan konsensus. Blockchain biasanya tanpa otoritas pusat (mis., bank, perusahaan atau pemerintah).

Pada tahun 2008, ~~Ide-B~~blockchain dikombinasikan dengan beberapa teknologi lain dan konsep komputasi untuk menciptakan *cryptocurrency* modern: uang elektronik yang dilindungi melalui mekanisme kriptografi alih-alih repositori atau otoritas pusat. Teknologi ini mulai dikenal luas pada tahun 2009 dengan diluncurkannya jaringan Bitcoin, yang pertama dari banyak *cryptocurrency* modern. Dalam Bitcoin, dan sistem serupa, transfer informasi digital yang mewakili uang elektronik terjadi dalam sistem terdistribusi.

2.2.1. Kategorisasi Blockchain

Jaringan blockchain dapat dikategorikan berdasarkan model izinnya, yang menentukan siapa yang dapat ~~mempertahankannya-menerbitkan blok~~ [12]. Ada 2 Kategorisasi dalam Jaringan blockchain, yang pertama permissionless dimana semua pengguna bisa menerbitkan blok dan yang kedua permissioned dimana hanya pengguna tertentu yang dapat menerbitkan blok. ~~Jika semua pengguna bisa menerbitkan blok baru, itu permissionless. Jika hanya pengguna tertentu yang dapat~~

~~menerbitkan blok, itu *permissioned*.~~ Dalam istilah sederhana, jaringan blockchain yang diizinkan seperti intranet perusahaan yang dikendalikan, sementara jaringan blockchain yang tanpa izin seperti internet publik, di mana siapa saja dapat berpartisipasi.

2.2.1.1 *Permissionless*

Jaringan blockchain *permissionless* adalah platform ledger terdesentralisasi yang terbuka untuk siapa saja yang menerbitkan blok, tanpa memerlukan izin dari otoritas mana pun [12]. Platform blockchain *permissionless* seringkali merupakan perangkat lunak open source, tersedia secara bebas bagi siapa saja yang ingin mengunduhnya. setiap pengguna jaringan blockchain dalam jaringan blockchain tanpa izin dapat membaca dan menulis ke *ledger* atau buku besar. Jaringan blockchain *permissionless* terbuka untuk semua orang untuk berpartisipasi. Untuk mencegah pengguna jahat menumbangkan sistem dengan mudah, jahat berupaya untuk mempublikasikan blok jaringan blockchain sering menggunakan, penggunaan perjanjian multi-partai banyak pihak atau sistem 'konsensus' yang mengharuskan pengguna mengeluarkan atau mengelola sumber daya ketika ingin menerbitkan blok.

2.2.1.2 *Permissioned*

Jaringan blockchain *permissioned* adalah jaringan di mana pengguna yang mempublikasikan blok harus diotorisasi oleh otoritas tertentu (baik tersentralisasi atau terdesentralisasi) [12]. Karena hanya pengguna yang berwenang yang dapat menggunakan blockchain, dimungkinkan untuk membatasi akses baca dan membatasi siapa yang dapat mengeluarkan transaksi. Jaringan blockchain *permissioned* dapat digunakan dengan perangkat lunak sumber *open source* atau sumber tertutup.

Jaringan blockchain yang *permissioned* dapat memiliki ketertelusuran yang sama dari aset digital ketika menggunakan blockchain, serta sistem penyimpanan data yang terdistribusi, dan redundan yang sama dengan jaringan blockchain *permissionless*. *Permissioned* blockchain juga menggunakan model konsensus

untuk menerbitkan blok, tetapi metode ini biasanya tidak memerlukan biaya atau pemeliharaan sumber daya. Ini terjadi karena pembentukan identitas seseorang diperlukan untuk berpartisipasi sebagai anggota jaringan blockchain *permissioned*. Mereka yang memelihara blockchain memiliki tingkat kepercayaan satu sama lain, karena mereka semua berwenang untuk menerbitkan blok dan karena otorisasi mereka dapat dicabut jika mereka berperilaku tidak pantas. Model konsensus dalam jaringan blockchain yang diijinkan biasanya lebih cepat dan lebih murah secara komputasi.

2.2.2. Komponen Blockchain

Pada tingkat tinggi, teknologi blockchain menggunakan mekanisme ilmu komputer yang sudah terkenal dan primitif kriptografi dicampur dengan konsep penyimpanan catatan [12]. Bagian ini membahas setiap komponen individu utama: fungsi hash kriptografis, transaksi, kriptografi kunci asimetris, alamat, ledgers, blok, dan bagaimana blok menjadi berantai.

2.2.2.1 Fungsi Hash Kriptografis

Komponen penting dari teknologi blockchain adalah penggunaan fungsi hash kriptografis untuk banyak operasi [12]. Hashing adalah metode penerapan fungsi hash kriptografis ke data, yang menghitung output yang relatif unik untuk input dengan ukuran apapun (misal. File, teks, atau gambar). Ini memungkinkan individu untuk secara mandiri melakukan input data, hash data, dan memperoleh hasil yang sama sehingga membuktikan bahwa tidak ada perubahan dalam data.

Fungsi hash kriptografis tertentu yang digunakan dalam banyak implementasi blockchain adalah Secure Hash Algorithm (SHA) dengan ukuran output 256 bit (SHA-256). Banyak komputer mendukung algoritma ini dalam perangkat keras, membuatnya cepat untuk dikomputasi. SHA-256 memiliki output 32 byte (1 byte = 8 bit, 32 byte = 256 bit), umumnya ditampilkan sebagai string heksadesimal 64-karakter.

Teks Input	SHA-256 Digest Value
Muhamad	0x519286eb705fe2ddc39ac3dfd739c1cecaa11c9f632cda0584d9d7f422dcee9c
Fahriza	0xf247637cd294a69422a46ee755deaa0095ec5360bcf3351250b58e628270ce2a
Novriansyah	0xc56b13597bfb8b9b72b03a729936fa02a8b02ab130abc66f6236a67bf52c914f

Tabel 1-22. Contoh Teks Input dan Nilai-nilai SHA-256 Digest Value

Dalam jaringan blockchain, fungsi hash kriptografis digunakan untuk banyak tugas, seperti: Derivasi alamat, Membuat pengidentifikasi unik. Mengamankan data blok, dan Mengamankan header blok.

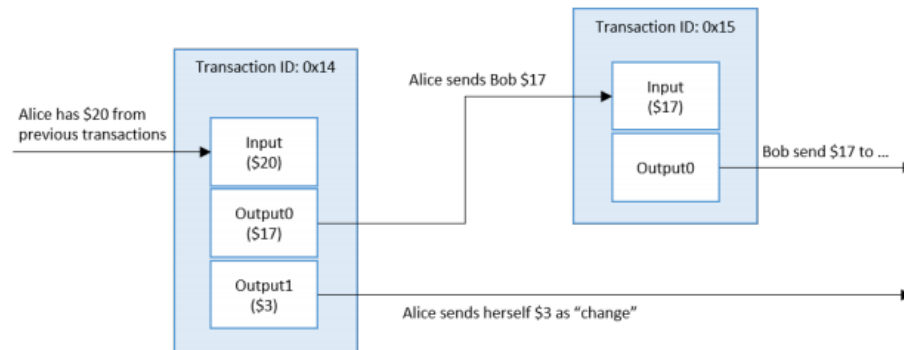
2.2.2.2 Transaksi

Transaksi mewakili interaksi antara pihak-pihak [12]. Dengan *cryptocurrency*, misalnya, transaksi mewakili transfer *cryptocurrency* antara pengguna jaringan blockchain. Untuk skenario bisnis-ke-bisnis, transaksi dapat menjadi cara untuk mencatat aktivitas yang terjadi pada aset digital atau fisik. Setiap blok dalam blockchain dapat berisi nol atau lebih transaksi.

Satu transaksi *cryptocurrency* biasanya memerlukan setidaknya informasi berikut, tetapi bisa mengandung lebih banyak:

1. Input biasanya berupa daftar aset digital yang akan ditransfer.
2. Output biasanya adalah akun yang akan menjadi penerima aset digital beserta berapa banyak aset digital yang akan mereka terima. Setiap output menentukan jumlah aset digital yang akan ditransfer ke pemilik baru, pengidentifikasi pemilik baru, dan seperangkat kondisi yang harus dipenuhi pemilik baru untuk menghabiskan nilai itu. Jika aset digital yang disediakan

lebih dari yang dibutuhkan, dana ekstra harus secara eksplisit dikirim kembali ke pengirim



Gambar 2-1 Contoh Transaksi Cryptocurrency [12]

Transaksi biasanya ditandatangani secara digital oleh private-key terkait pengirim dan dapat diverifikasi kapan saja menggunakan public-key terkait.

2.2.2.3 Kriptografi Kunci Asimetrik

Teknologi Blockchain menggunakan kriptografi kunci asimetrik [12]. Kriptografi Kunci Asimetrik menggunakan sepasang kunci: *public key* dan *private key* yang secara matematis satu sama lain saling terkait. *public key* dibuat publik tanpa mengurangi keamanan proses, tetapi *private key* harus tetap rahasia jika data ingin mempertahankan perlindungan kriptografiknya. Meskipun terdapat hubungan antara dua kunci, *private key* tidak dapat ditentukan secara efisien berdasarkan pengetahuan tentang *public key*. Seseorang dapat mengenkripsi dengan *private key* dan kemudian mendekripsi dengan *public key*. Sebagai alternatif, seseorang dapat mengenkripsi dengan *public key* dan kemudian mendekripsi dengan *private key*.

2.2.2.4 Alamat dan Derivasi Alamat

Beberapa jaringan blockchain menggunakan alamat, yang merupakan string karakter alfanumerik pendek yang berasal dari *public key* pengguna jaringan blockchain menggunakan fungsi hash kriptografi, dengan beberapa data tambahan (misalnya Nomor versi, checksum) [12]. Sebagian besar implementasi blockchain menggunakan alamat sebagai titik akhir "ke" dan "dari" dalam suatu transaksi.

Salah satu metode untuk menghasilkan alamat adalah dengan membuat *public key*, menerapkan fungsi hash kriptografi, dan mengonversi hash ke teks:

public key \rightarrow fungsi hash kriptografi \rightarrow alamat

Setiap implementasi blockchain dapat menerapkan metode yang berbeda untuk mendapatkan alamat. Untuk jaringan blockchain *permissionless*, yang memungkinkan pembuatan akun anonim, pengguna jaringan blockchain dapat menghasilkan banyak pasangan kunci asimetris, dan oleh karena itu alamat sesuai keinginan, memungkinkan untuk berbagai tingkat anonimitas semu.

2.2.2.5 Penyimpanan Private Key

Dengan beberapa jaringan blockchain (terutama dengan jaringan blockchain *permissionless*), pengguna harus mengelola dan menyimpan kunci pribadi mereka sendiri dengan aman [12]. Alih-alih merekam secara manual, pengguna sering menggunakan perangkat lunak untuk menyimpannya dengan aman. Perangkat lunak ini sering disebut sebagai *wallet*. *Wallet* dapat menyimpan *private key*, *public key*, dan alamat terkait. *Wallet* juga dapat menjalankan fungsi lain, seperti menghitung jumlah total aset digital yang mungkin dimiliki pengguna.

Jika pengguna kehilangan *private key*, maka setiap aset digital yang terkait dengan *private key* itu hilang, karena tidak layak secara komputasi untuk membuat ulang *private key* yang sama. Jika *private key* dicuri, penyerang akan memiliki akses penuh ke semua aset digital yang terkait oleh *private key* itu.

2.2.2.6 Ledgers

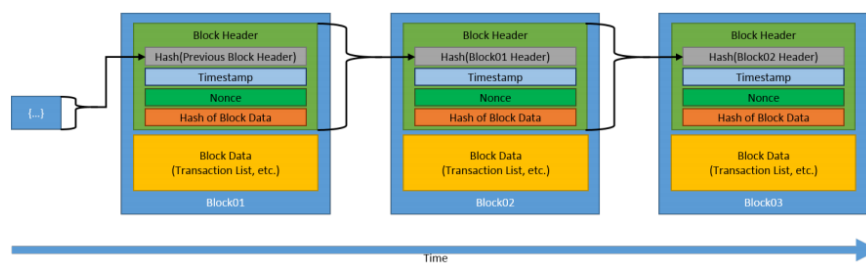
Ledger adalah kumpulan transaksi [12]. Di zaman modern, *ledger* telah disimpan secara digital, seringkali dalam basis data besar, dimiliki dan dioperasikan oleh pihak ketiga yang dipercaya terpusat atas nama komunitas pengguna. *Ledger* ini dengan kepemilikan terpusat dapat diimplementasikan secara terpusat atau terdistribusi (misalnya hanya satu server atau sekelompok server yang terkoordinasi).

2.2.2.7 Blok

Transaksi ditambahkan ke blockchain ketika simpul penerbitan menerbitkan blok [12]. Sebuah Blok dapat berisi blok header dan block data. Blok Header berisi metadata untuk blok. Data blok berisi daftar transaksi yang divalidasi dan otentik yang telah diserahkan ke jaringan blockchain.

2.2.2.8 Rantai Blok

Blok dirantai bersama melalui setiap blok yang berisi intisari blok header sebelumnya, sehingga membentuk blockchain [12]. Jika blok yang diterbitkan sebelumnya diubah, itu akan memiliki hash yang berbeda. Dengan ini memungkinkan untuk dengan mudah mendeteksi dan menolak blok yang diubah.



Gambar 2-2 Rantai Blok Generik [12]

2.2.3. Model Konsensus

Blockchain adalah jenis *distributed ledger* yang didistribusikan yang dapat mencatat transaksi antara pihak-pihak dengan cara yang dapat diverifikasi dan permanen [13]. Aspek kunci dari teknologi blockchain adalah menentukan pengguna mana yang menerbitkan blok berikutnya [12]. Ini diselesaikan melalui penerapan salah satu dari banyak model konsensus. Teknologi Blockchain menggunakan model konsensus untuk memungkinkan sekelompok pengguna yang saling tidak percaya saling bekerja sama. Mekanisme konsensus ini menjadi peran penting dalam teknologi blockchain untuk menjaga buku besar (*ledger*) yang didistribusikan, disinkronkan dengan aman melalui proses kolaboratif [14].

Pada bagian berikut, beberapa model konsensus serta pendekatan resolusi konflik yang paling umum dibahas.

1. Proof of work (PoW)

- Tujuan: Memberikan penghalang pada penerbitan blok dalam bentuk teka-teki yang sulit untuk diselesaikan secara komputasi untuk memungkinkan transaksi antara peserta yang tidak dipercaya
- Keuntungan: Sulit melakukan *denial of service* dengan membanjiri jaringan dengan blok buruk dan Terbuka untuk siapa saja dengan perangkat keras untuk memecahkan teka-teki.
- Kekurangan: Intensif secara komputasi (berdasarkan desain) dan konsumsi daya. Potensi serangan 51% dengan mendapatkan kekuatan komputasi yang cukup.
- Domain: *Permissionless* cryptocurrencies.
- Implementasi: Bitcoin, Ethereum, dan lainnya

2. Proof of stake (PoS)

- Tujuan: Mengaktifkan penghalang yang kurang intensif secara komputasi untuk menerbitkan blok, tetapi tetap memungkinkan transaksi antara peserta yang tidak dipercaya.
- Keuntungan: Komputasi kurang intensif dibandingkan PoW, Terbuka untuk siapa saja yang ingin mempertaruhkan cryptocurrency, dan Stakeholder mengendalikan sistem.
- Kekurangan: Stakeholder mengendalikan sistem, Tidak ada yang mencegah pembentukan kumpulan pemangku kepentingan untuk menciptakan kekuatan terpusat, dan Potensi serangan 51% dengan memperoleh kekuatan finansial yang cukup.
- Domain: *Permissionless* cryptocurrencies.
- Implementasi : Ethereum Casper dan Krypton

3. Delegated PoS

- Tujuan : Mengaktifkan model konsensus yang lebih efisien melalui 'demokrasi liquid' di mana peserta memberikan suara (menggunakan pesan yang ditandatangani secara kriptografi) untuk memilih dan mencabut hak-hak delegasi untuk memvalidasi dan mengamankan blockchain.

- Keuntungan : Delegasi terpilih secara ekonomi diberi insentif untuk tetap jujur, dan Komputasi lebih efisien daripada PoW
 - Kekurangan : Keragaman simpul lebih sedikit daripada implementasi konsensus PoW atau PoS murni, Risiko keamanan yang lebih besar untuk kompromi node karena sekumpulan node operasi terbatas, dan dan kemungkinan ada insentif bagi produsen blok untuk berkolusi dan menerima suap, dengan mengorbankan keamanan sistem
 - Domain: *Permissionless* cryptocurrencies, Sistem *Permissioned*
 - Implementasi : Bitshares, Steem, Cardano, dan EOS
4. Round Robin
- Tujuan : Menyediakan sistem untuk menerbitkan blok di antara node penerbitan yang disetujui / tepercaya
 - Keuntungan : Daya komputasi rendah, dan mudah dimengerti.
 - Kekurangan : Membutuhkan kepercayaan yang besar di antara simpul penerbitan.
 - Domain: Sistem *Permissioned*
 - Implementasi : MultiChain
5. Proof of Authority/Identity
- Tujuan: Msembuat proses konsensus terpusat untuk meminimalkan pembuatan blok dan tingkat konfirmasi
 - Keuntungan: Waktu konfirmasi yang cepat, Mengizinkan tingkat produksi blok dinamis, Dapat digunakan dalam sidechains untuk memblokir jaringan yang menggunakan model konsensus lain
 - Kekurangan: Bergantung pada asumsi bahwa simpul validasi saat ini belum dikompromikan, dan menyebabkan titik kegagalan terpusat.
 - Domain: *Permissioned Systems, Hybrid (sidechain) Systems*
 - Implementasi : Ethereum Kovan testnet, dan POA Chain,

2.2.4. Smart Contracts

Istilah *smart contract* dimulai pada tahun 1994, didefinisikan oleh Nick Szabo sebagai “protokol transaksi terkomputerisasi yang mengeksekusi ketentuan-ketentuan kontrak. Tujuan umum dari desain *smart contract* adalah untuk

memenuhi persyaratan kontrak yang umum (seperti ketentuan pembayaran, hak gadai, kerahasiaan, dan bahkan penegakan), meminimalkan pengecualian berbahaya dan tidak disengaja, dan meminimalkan kebutuhan perantara yang tepercaya” [12]. *smart contract* memperluas dan memanfaatkan teknologi blockchain. *smart contract* adalah kumpulan kode dan data yang disebarluaskan menggunakan transaksi yang ditandatangani secara kriptografi di jaringan blockchain misalnya pada Ethereum’s smart contracts dan Hyperledger Fabric’s chaincode. *smart contract* dijalankan oleh node dalam jaringan blockchain; semua node yang menjalankan *smart contract* harus mendapatkan hasil yang sama dari eksekusi, dan hasil eksekusi dicatat pada blockchain.

2.3. E-Voting dan Blockchain

Sistem pemungutan suara elektronik atau *e-voting* adalah sistem pemungutan suara di mana data pemilu dicatat, disimpan, dan diproses terutama sebagai informasi digital [15]. *E-voting* menggunakan sarana elektronik untuk memberikan dan menghitung suara. Sistem *E-Voting* telah digunakan sejak 1960 ketika sistem *punched card* muncul dan digunakan pada tujuh negara berbeda di AS untuk pemilihan presiden tahun 1964 dan sekarang ini telah menjadi cara pemungutan suara yang sangat praktis. *E-voting* memiliki banyak keunggulan dibanding cara pemungutan suara tradisional. Beberapa keunggulan ini adalah biaya yang lebih rendah, tabulasi hasil yang lebih cepat, akurasi yang lebih tinggi, dan risiko kesalahan manusia dan mekanik yang lebih rendah. Ini menawarkan aksesibilitas yang lebih baik bagi para penyandang cacat, dan menyediakan dukungan multi-bahasa untuk surat suara

Sistem *E-Voting* berbasis blockchain yang menggunakan smart contracts memungkinkan pemilihan yang aman dan efisien serta menjamin privasi pemilih [16]. Dengan teknologi blockchain menawarkan kemungkinan baru untuk mengatasi keterbatasan dan hambatan adopsi sistem pemungutan suara elektronik yang menjamin keamanan dan integritas pemilu dan memberikan landasan bagi transparansi.

Solusi berbasis blockchain telah banyak digunakan untuk pemungutan suara perusahaan, komunitas, kota, dan nasional [17].

Pengaturan	Konteks	Keterangan
Program The city of Moscow's Active Citizen	Pada bulan Desember 2017, program mulai menggunakan blockchain untuk pemungutan suara dan untuk membuat hasil pemungutan suara diaudit secara publik.	Jajak pendapat paling populer dilaporkan memiliki 137.000 hingga 220.000 peserta.. Platform mencapai puncak sekitar 1.000 transaksi per menit
Proyek komunitas provinsi Gyeonggi-do Korea Selatan	Provinsi menggunakan sistem pemungutan suara berbasis blockchain untuk mengumpulkan suara pada proyek komunitas	Startup <i>fin-tech</i> Korea mengembangkan platform blockchain. 9.000 penduduk memberikan suara.
Pertemuan umum tahunan perusahaan teknologi Estonia LVH Group	Pemegang saham dapat login menggunakan ID online nasional terverifikasi dan memberikan suara pada rapat.	Sistem pemungutan suara mengeluarkan aset hak pilih dan aset hak suara kepada pemegang saham. Pengguna dapat melakukan <i>voting</i> untuk memberikan suara pada item agenda rapat memiliki aset hak pilih terkait..
2018 Sierra Leonean general election	Startup Swiss Agora dilakukan penghitungan di dua distrik. Setelah pemungutan suara, tim pengamat terakreditasi dari lokasi yang berbeda secara manual memasukkan sekitar	Tes ini dianggap sebagai <i>deployment</i> parsial blockchain. Pemilihan hanya diverifikasi oleh blockchain, bukan <i>powered</i> blockchain. Agora memberikan penghitungan suara independen, yang

	400.000 surat suara ke sistem blockchain Agora.	dibandingkan dengan penghitungan utama
--	---	--

Tabel 33-4.4 Solusi berbasis blockchain untuk pemungutan suara [17]

2.4. Hyperledger

Hyperledger adalah upaya kolaboratif *open-source* yang diselenggarakan oleh Linux Foundation dibuat untuk memajukan teknologi blockchain lintas-industri [18]. Hyperledger ini merupakan kolaborasi global yang didalamnya ada para pemimpin di bidang perbankan, keuangan, *Internet of Things*, manufaktur, *supply chain*, dan teknologi. Dengan sifatnya yang *open-source*, komunitas pengembang di seluruh dunia dapat bekerja bersama dan berbagi ide, infrastruktur, dan kode untuk mengembangkan Hyperledger.

2.4.1. Filosofi Desain Hyperledger

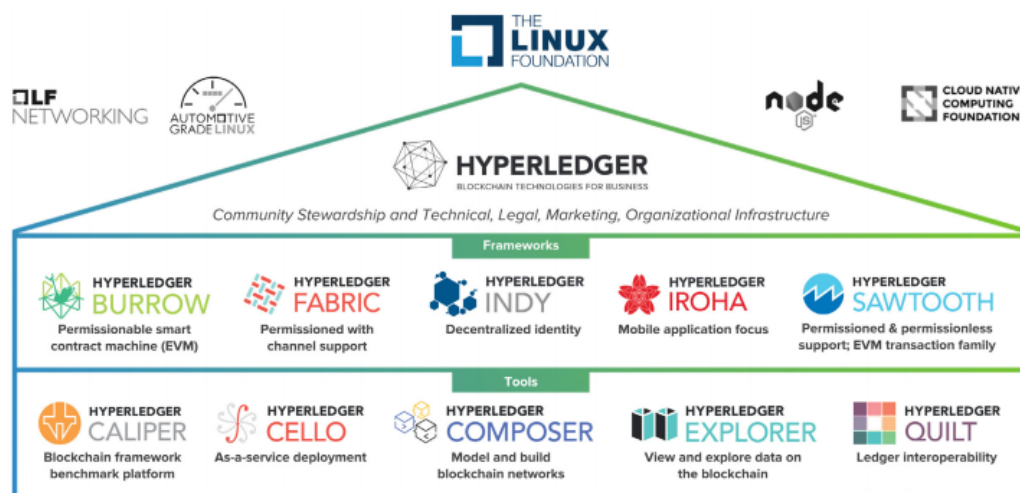
Distributed ledgers dapat memiliki persyaratan yang sangat berbeda untuk berbagai kasus penggunaan [18]. Misalnya, ketika peserta berbagi tingkat kepercayaan yang tinggi seperti antara lembaga keuangan dengan perjanjian hukum, blockchain dapat menambah blok pada rantai dengan waktu konfirmasi yang lebih singkat dengan menggunakan algoritma konsensus yang lebih cepat. Di sisi lain, ketika ada kepercayaan minimal antara peserta, mereka harus mentolerir pemrosesan lebih lambat untuk keamanan tambahan. Hyperledger mencakup spektrum penuh kasus penggunaan. Hyperledger menyadari bahwa berbagai skenario perusahaan memiliki persyaratan yang berbeda untuk waktu konfirmasi, desentralisasi, kepercayaan, dan masalah lainnya, dan bahwa setiap masalah mewakili potensi “titik optimasi” untuk teknologi.

Untuk mengatasi keragaman ini, semua proyek Hyperledger mengikuti filosofi desain yang sama, yaitu:

- Modular
- Sangat aman
- Dapat dioperasikan
- Cryptocurrency-agnostik
- Lengkap dengan API Diagram Alir

2.4.2. Hyperledger Greenhouse

Hyperledger berfungsi sebagai *greenhouse* yang menyatukan pengguna, pengembang, dan vendor dari berbagai sector [18]. Hyperledger menyediakan struktur *greenhouse* yang dapat menetaskan ide-ide baru, mendukung masing-masing dengan sumber daya penting, dan mendistribusikan hasilnya secara luas. Struktur *greenhouse* dapat mendukung banyak varietas yang berbeda walaupun mengonsumsi sumber daya yang jauh lebih sedikit.



Gambar 3-4 Struktur Greenhouse Hyperledger [17]

Sebagai organisasi *greenhouse* untuk pengembangan blockchain *open-source*, Hyperledger memberikan manfaat ini:

- Membantu menjaga perkembangan
- Produktivitas yang lebih baik melalui spesialisasi
- Kolaborasi untuk menghindari upaya duplikat
- Kontrol kualitas kode yang lebih baik

- Penanganan kekayaan intelektual yang lebih mudah

Berdasarkan Struktur *Greenhouse* Hyperledger, Hyperledger dapat dibagi menjadi dua yaitu Hyperledger Frameworks dan Hyperledger Tools.

2.4.3. Hyperledger Framework

Hyperledger memiliki beberapa framework diantaranya [18]:

1. Hyperledger Burrow

Hyperledger Burrow adalah smart contract yang diizinkan. Jenis ini pertama kali dirilis pada bulan Desember 2014, Hyperledger Burrow menyediakan klien blockchain modular dengan penerjemah smart contract yang diizinkan sebagian dibangun untuk spesifikasi Ethereum Virtual Machine (EVM).

2. Hyperledger Fabric

Hyperledger Fabric adalah platform untuk membangun solusi distributed ledger dengan arsitektur modular yang memberikan tingkat kerahasiaan, fleksibilitas, ketahanan, dan skalabilitas yang tinggi. Ini memungkinkan solusi yang dikembangkan dengan Hyperledger Fabric dapat diadaptasi untuk industri apa pun.

3. Hyperledger Indy

Hyperledger Indy adalah distributed ledger yang didistribusikan, dirancang untuk identitas terdesentralisasi. Indy menyediakan alat, perpustakaan, dan komponen yang dapat digunakan kembali untuk membuat dan menggunakan identitas digital independen yang di-root pada blockchain atau distributed ledger lain yang didistribusikan.

4. Hyperledger Iroha

Hyperledger Iroha adalah kerangka kerja blockchain yang dirancang untuk menjadi sederhana dan mudah untuk dimasukkan ke dalam proyek infrastruktur yang membutuhkan teknologi distributed ledger yang didistribusikan

5. Hyperledger Sawtooth

Hyperledger Sawtooth adalah platform modular untuk membangun, menyebarkan, dan menjalankan distributed ledgers. Distributed ledgers

menyediakan catatan digital (seperti kepemilikan aset) yang dikelola tanpa otoritas atau implementasi pusat. Sawtooth bertujuan untuk menjaga distribusi buku besar tetap didistribusikan dan membuat smart contracts aman untuk penggunaan perusahaan.

2.4.4. Hyperledger Tools

Hyperledger memiliki beberapa tools diantaranya [18]:

1. Hyperledger Caliper

Hyperledger Caliper adalah alat benchmark blockchain yang berfungsi untuk mengukur kinerja setiap implementasi blockchain dengan menggunakan sebuah set kasus penggunaan yang telah ditentukan. Caliper menghasilkan laporan yang menunjukkan sejumlah indikator kinerja, seperti:

- Pemanfaatan sumber daya
- Latensi transaksi
- Transaksi Per Detik (TPS)

2. Hyperledger Cello

Hyperledger Cello adalah toolkit modul blockchain yang bertujuan untuk membawa model penyebaran on-demand ke ekosistem blockchain. Tujuannya adalah untuk membantu perusahaan dengan cepat dan mudah mengadopsi teknologi blockchain, dengan menyediakan cara otomatis untuk membuat, mengelola, dan menghentikan blockchain.

3. Hyperledger Composer

Hyperledger Composer adalah *toolset* pengembangan terbuka untuk membuat mudah dan cepat pembuatan *smart contracts* dan aplikasi blockchain untuk menyelesaikan masalah bisnis. Tujuan utama Hyperledger Composer adalah untuk membuat mudah mengintegrasikan aplikasi blockchain dengan sistem bisnis yang ada, dan dengan demikian mempercepat *time-to-value*. Suatu jaringan dapat berisi *asset* seperti barang, jasa berwujud atau tidak berwujud, atau properti dan transaksi yang berkaitan dengannya. sebagai bagian dari model, pengguna dapat menentukan bagaimana transaksi dapat berinteraksi dengan *asset*. *Business*

Network termasuk peserta yang berinteraksi dengan mereka. Dan setiap peserta dapat dikaitkan dengan identitas unik di beberapa *business network* yang berbeda. Hyperledger Composer mendukung infrastruktur dan *runtime* blockchain Hyperledger Fabric. Karena Fabric mendukung protokol konsensus *pluggable*, ini memastikan bahwa transaksi dapat divalidasi sesuai dengan kebijakan yang tepat dari peserta jaringan.

Pada Hyperledger Composer, *Business Network Card* menyediakan semua informasi yang diperlukan untuk terhubung ke jaringan bisnis blockchain. Pada Hyperledger Composer Playground untuk terhubung ke Jaringan Bisnis yang digunakan membutuhkan *Business Network Card*.

4. Hyperledger Explorer

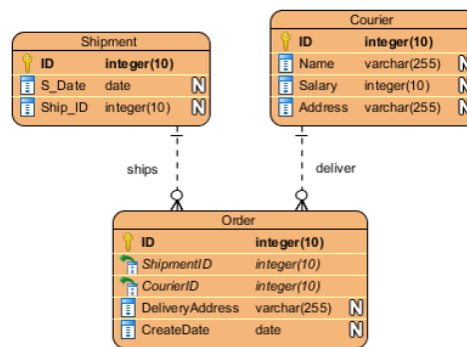
Hyperledger Explorer menyediakan *dashboard* untuk melihat informasi tentang blok, log simpul, statistik, kontrak pintar, transaksi, dan informasi lainnya yang disimpan dalam blockchain. Pengguna dapat meminta blok atau transaksi tertentu untuk melihat detail lengkap.

5. Hyperledger Quilt

Hyperledger Quilt menawarkan interoperabilitas antara sistem ledger dengan mengimplementasikan Interledger Protocol (ILP) dalam Java. ILP adalah protokol *open-source* sederhana yang menetapkan *namespace* global untuk akun untuk membantu melakukan transaksi lintas *ledgers*.

2.5. Entity Relationship Diagram (ERD)

Entity Relationship Diagram, juga dikenal sebagai ERD, ER Diagram atau ER model, adalah jenis diagram struktural untuk digunakan dalam desain database [19]. ERD berisi simbol dan konektor berbeda yang memvisualisasikan dua informasi penting: Entitas utama dalam ruang lingkup sistem, dan hubungan antar entitas-entitas ini.



Gambar 5-6 Contoh Entity Relationship Diagram [19]

2.5.1. Notasi ERD

Diagram ER berisi entitas, atribut, dan hubungan. Berikut simbol-simbol yang ada di ERD [19]:

- Entity** atau Entitas
 Entitas dalam ERD adalah hal atau konsep yang dapat didefinisikan dalam suatu sistem, seperti orang / peran (misalnya Siswa), objek (misalnya Faktur), konsep (misalnya Profil) atau peristiwa (misalnya Transaksi).
- Atribut Entitas**
 Atribut Entitas atau dikenal sebagai kolom, atribut adalah properti atau karakteristik entitas yang menyimpannya. Atribut memiliki nama yang menggambarkan properti dan tipe yang menggambarkan jenis atributnya, seperti varchar untuk sebuah string, dan int untuk integer.
- Primary Key**
Primary Key atau dikenal sebagai PK, *primary Key* adalah jenis khusus dari atribut entitas yang secara unik mendefinisikan catatan dalam tabel database.
- Foreign Key**
Foreign Key atau dikenal sebagai FK, *Foreign Key* adalah referensi ke *Primary Key* dalam sebuah tabel. *Foreign Key* digunakan untuk mengidentifikasi hubungan antar entitas dan tidak harus unik.
- Relationship** atau hubungan
 Hubungan antara dua entitas menandakan bahwa kedua entitas tersebut saling terkait satu sama lain

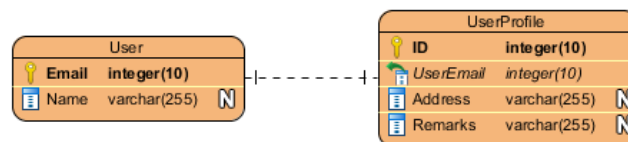
- Kardinalitas

Kardinalitas mendefinisikan kemungkinan jumlah kejadian dalam satu entitas yang dikaitkan dengan jumlah kejadian di entitas lain. Misalnya, satu tim memiliki pemain banyak didalam ERD didefinisikan *one-to-many relationship*.

Ada 3 hubungan kardinal yang umum yaitu *one-to-one*, *one-to-many*, dan *many-to-many*.

- One-to-one

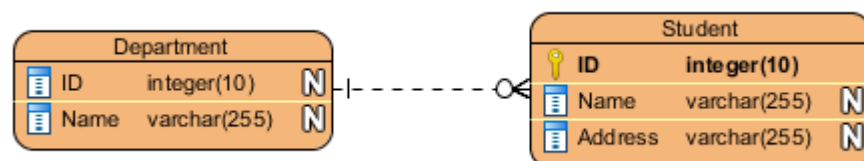
Hubungan *One-to-one* sebagian besar digunakan untuk membagi entitas menjadi dua dimana memberikan informasi secara ringkas dan membuatnya lebih dimengerti.



Gambar 7-8 Contoh Hubungan *One-to-one* [19]

- *One-to-many*

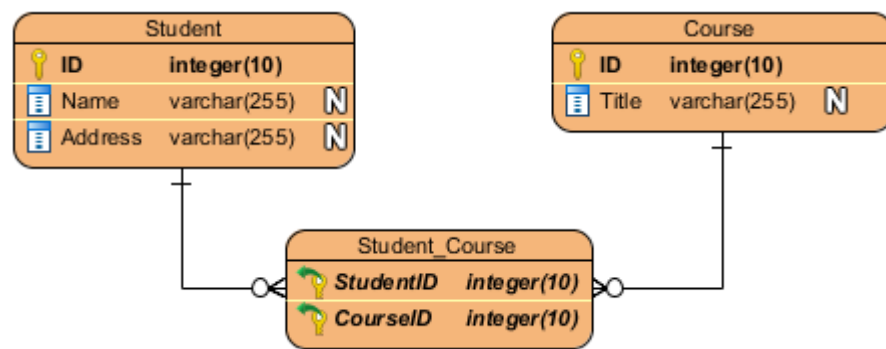
Hubungan *one-to-many* mengacu pada hubungan antara dua entitas X dan Y di mana sebuah *instance* X dapat dihubungkan ke banyak *instance* Y, tetapi sebuah *instance* Y terkait hanya dengan satu *instance* X.



Gambar 9-10 Contoh Hubungan *One-to-many* [19]

- Many-to-many

Hubungan *many-to-many* mengacu pada hubungan antara dua entitas X dan Y di mana X dapat dikaitkan dengan banyak *instance* Y dan sebaliknya.



Gambar 11-12 Contoh Hubungan *many-to-many* [19]

2.6. Unified Modeling Language (UML)

Unified Modeling Language atau UML telah dirancang untuk menjadi notasi standar penuh untuk Pemodelan Berorientasi Objek [20]. UML adalah cara memvisualisasikan program perangkat lunak menggunakan koleksi diagram [21]. UML 2.0 telah berkembang dengan beberapa perkembangan:

- Peningkatan integrasi antara model struktural seperti diagram kelas dan model perilaku seperti diagram aktivitas.
- Menambahkan kemampuan untuk mendefinisikan hierarki dan menguraikan sistem perangkat lunak menjadi komponen dan sub-komponen.
- UML awal menetapkan sembilan diagram; UML 2.x menjadi 13 diagram. Keempat diagram baru disebut: diagram *communication*, diagram *composite structure*, diagram *interaction overview*, dan diagram *timing*. diagram statechart berganti nama menjadi diagram *state machine* atau yang dikenal sebagai diagram *state*.

Diagram ini disusun dalam dua kelompok berbeda: diagram *structural* dan diagram *behavioral* atau *interaction*.

2.6.1. Diagram UML *structural*

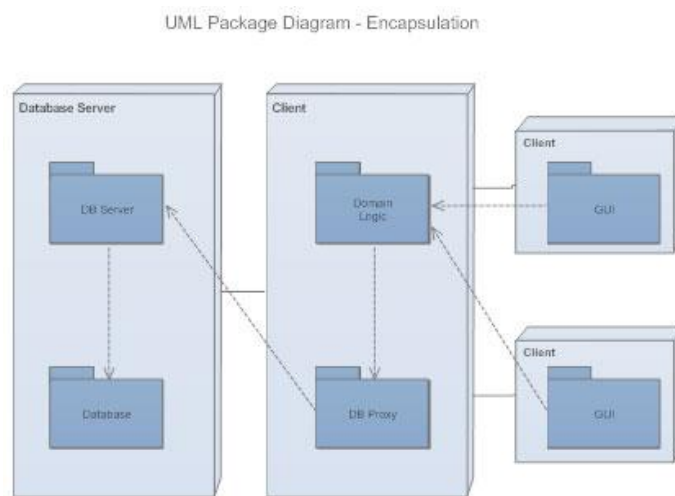
Diagram UML structural diantaranya [21]:

1. Diagram *Class*

Diagram *Class* adalah *backbone* dari hampir setiap metode berorientasi objek, termasuk UML. Diagram *Class* memodelkan struktur statis suatu sistem. Diagram ini menunjukkan hubungan antara kelas, objek, atribut, dan operasi.

2. Diagram *Package*

Diagram *Package* adalah subset diagram *class*, tetapi pengembang terkadang memperlakukannya sebagai teknik terpisah. Diagram *Package* mengatur elemen-elemen suatu sistem ke dalam kelompok terkait untuk meminimalkan ketergantungan antar paket.



Gambar 13-14 Contoh Diagram *Package* [21]

3. Diagram *Object*

Diagram *Object* adalah jenis UML yang menggambarkan struktur statis suatu sistem pada waktu tertentu.

4. Diagram *Component*

Diagram Component menggambarkan organisasi komponen perangkat lunak fisik, termasuk kode sumber, kode *run-time* (*binary*), dan *executable*.

5. Diagram *Composite Structure*

Diagram Composite Structure menunjukkan bagian internal kelas.

6. Diagram *Deployment*

Diagram *Deployment* menggambarkan sumber daya fisik dalam suatu sistem, termasuk node, komponen, dan koneksi

2.6.2. Diagram UML *behavioral*

Diagram UML *behavioral* diantaranya [21]:

1. Diagram *Use Case*

Diagram *Use Case* memodelkan fungsionalitas sistem menggunakan aktor dan *Use case*.

Simbol dan Notasi dalam Diagram *Use Case* [22]:

- *Use Case*: berbentuk oval horizontal yang mewakili berbagai kegunaan yang mungkin dimiliki pengguna.
- Aktor: berbentuk *stick figure* yang mewakili orang-orang yang menggunakan *use cases*.
- Asosiasi: berbentuk garis antara Aktor dan *Use cases*.
- Kotak batas sistem: Kotak yang menetapkan ruang lingkup sistem untuk *use cases*.
- *Packages*: Bentuk UML yang memungkinkan memasukkan elemen yang berbeda ke dalam grup



Gambar 15-16 Contoh Diagram *Use Case* [21]

2. Diagram *Activity*

Diagram *Activity* menggambarkan sifat dinamis sistem dengan memodelkan aliran kontrol dari aktivitas ke aktivitas. Biasanya, Diagram *Activity* digunakan untuk memodelkan alur kerja atau proses bisnis dan operasi internal.

3. Diagram *Sequence*

Diagram *Sequence* diagram menggambarkan interaksi antar kelas dalam hal pertukaran pesan dari waktu ke waktu.

4. Diagram *State*

Diagram *Statechart*, sekarang dikenal sebagai diagram *state machine* dan diagram *state* menggambarkan perilaku dinamis suatu sistem dalam menanggapi rangsangan eksternal

5. Diagram *Communication*

Diagram *Communication* memodelkan interaksi antara objek secara berurutan. Diagram jenis ini menggambarkan struktur statis dan perilaku dinamis suatu sistem.

6. Diagram *Interaction Overview*

Diagram *Interaction Overview* adalah kombinasi diagram *Activity* dan *Sequence*.

7. Diagram *Timing*

Diagram *Timing* adalah jenis UML yang berfokus pada proses yang berlangsung selama periode waktu tertentu.

2.7. Angular

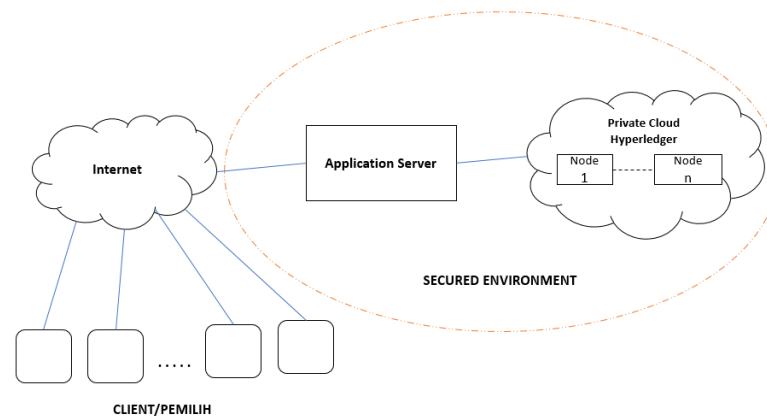
Angular sebuah platform dan kerangka kerja untuk membuat sebuah aplikasi dari sisi klien menggunakan bahasa pemrograman HTML dan TypeScript [23]. Angular ini digunakan untuk mengembangkan *front-end*/tampilan sebuah sistem. Angular ini bersifat *open-source*.

BAB 3

PERANCANGAN SISTEM

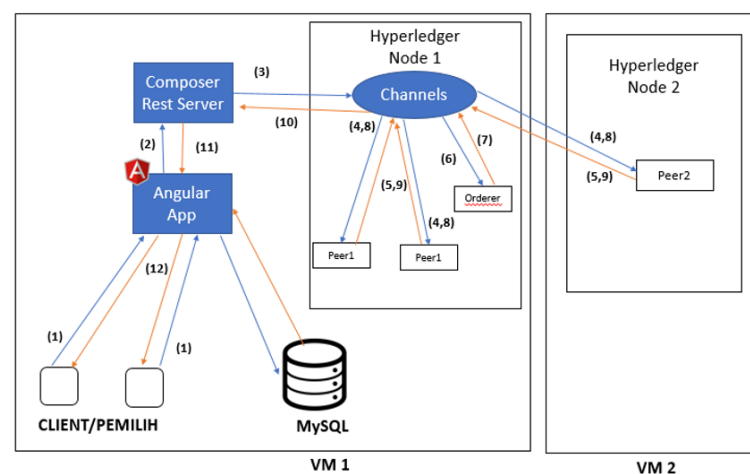
3.1. Spesifikasi Software dan Hardware

Pada Penelitian ini untuk menguji sistem yang digunakan idealnya menggunakan topologi seperti Gambar 3-1.



Gambar 3-1 Topologi *Deployment* Ideal/ Realistik

Topologi gambar diatas merupakan topologi yang ideal/realistic untuk digunakan dimana topologi tersebut menggunakan cloud sehingga dapat diakses lewat internet dan dapat memiliki banyak node yang disesuaikan dengan kebutuhan.



Gambar 3-2 Topologi Ujicoba

Pada Gambar 3-2 merupakan topologi untuk ujicoba yang peneliti akan gunakan. Peneliti menggunakan 2 Node/Perangkat yang digunakan sebagai Virtual Server yang terinstall Hyperledger untuk membuat blockchain.

Terdapat 3 Fase untuk aliran transaksi seperti pada sistem yang dibuat seperti pada Gambar 3-2:

- Fase 1: *Proposal*(1)-(5), tahap ini merupakan tahap ketika pemilih melakukan sebuah transaksi. Pemilih akan mengakses aplikasi *Angular* dan melakukan sebuah transaksi melalui *composer rest-server* ke jaringan blockchain Hyperledger. Transaksi tersebut akan masuk ke *channels*, setelah masuk akan di kirimkan ke semua *peer* yang terkoneksi pada *channels* tersebut dan dilakukan validasi pada *peer*. Jika disetujui maka *peer* tersebut akan mengirim transaksi yang sudah divalidasi ke *Orderer*.
- Fase 2: *Ordering* dan *Packaging*(6)–(7), tahap ini merupakan tahap ketika *Orderer* mengurutkan dan membuat blok dari transaksi yang telah divalidasi *peer*. *Orderer* akan menerima transaksi yang sudah divalidasi oleh *peer*, lalu kumpulan transaksi tersebut diurutkan dan dibuat menjadi sebuah blok. Setelah blok dibuat, blok siap didistribusikan.
- Fase 3: *Validation* dan *commit*(8-9), tahap ini merupakan tahap distribusi blok dan validasinya pada semua *peer* yang terhubung pada *channels*. *Orderer* mendistribusikan blok ke semua *peer* yang terhubung pada channel. Lalu *peer* akan melakukan validasi terhadap blok tersebut sehingga *ledger* konsisten dan melakukan perubahan pada *ledger* tersebut.
- Setelah 3 Fase tersebut selesai(10-12) maka sistem akan mengembalikan sebuah nilai ke pemilih.

Untuk perancangan sistem, Peneliti menggunakan software dan hardware. Tabel III.1 berikut menunjukkan daftar spesifikasi Hardware dan Software yang digunakan pada sistem yang akan diuji.

No	Jenis	Perangkat 1 dan Perangkat 2
----	-------	-----------------------------

1.	Virtual Machine	VMware Workstation 15.0.2 build-10952284
2.	Sistem Operasi	Ubuntu 16.04 LTS
3.	Storage	4 GB
4.	Prosesor	2 Cores
5.	Koneksi Jaringan	NAT
6.	Hyperledger Composer	Docker Engine version 19.03.7, build 7141c199a2
		Docker Compose version 1.13.0, build 1719ceb
		Node v8.17
		Npm v.6.13.4
		Git v2.25.2
		Python v2.7.12
		CLI Tools v0.20.9
7.	Hyperledger Fabric	V1.2

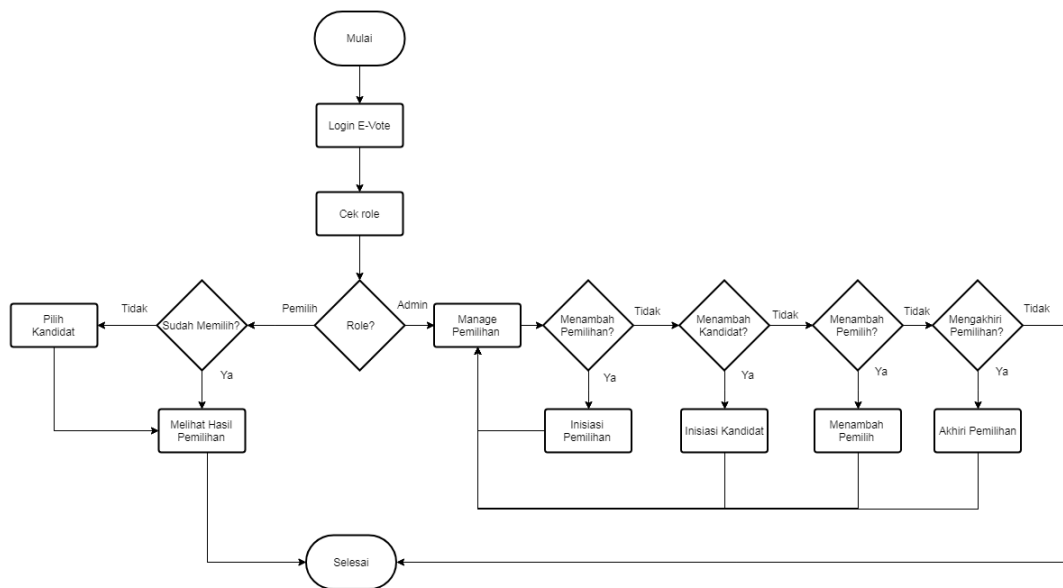
Tabel 3-1 Daftar Spesifikasi *Hardware* dan *Software* pada sistem

Dari tabel diatas, dapat diketahui bahwa terdapat spesifikasi hardware dan software yang digunakan peneliti untuk proses pengujian sistem.

3.2. Diagram Alir, ERD dan UML Sistem

3.2.1. Diagram Alir

Berikut merupakan Diagram Alir Sistem:

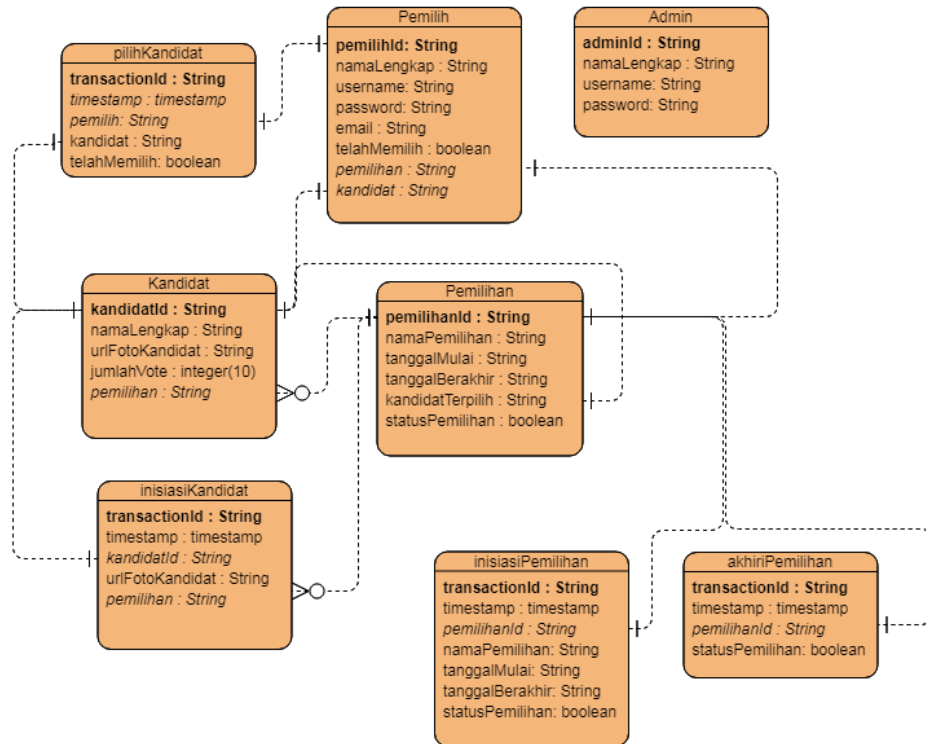


Gambar 3-3 Diagram Alir Sistem

Pertama Pengguna akan melakukan Login ke Sistem, setelah itu sistem akan melakukan Cek Role terhadap pengguna. Jika Role merupakan Admin maka Pengguna dapat melakukan pembuatan pemilihan, apabila sudah dapat membuat pemilihan, Admin dapat membuat kandidat dan pemilih. Jika Role merupakan Pemilih maka pengguna dapat memilih kandidat, jika sudah memilih kandidat maka pengguna dapat melihat hasil Pemilihan. Jika Proses Pemilihan Kandidat telah selesai maka role Admin dapat menyelesaikan Pemilihan.

3.2.2. Entity Relationship Diagram

Berikut merupakan Entity Relationship Diagram Sistem :



Gambar 3-4 Entity Relationship Diagram Sistem

Terdapat 8 Entitas untuk sistem yang akan dirancang yaitu Admin, Pemilih, Pemilihan, Kandidat, inisiasiPemilihan, inisiasiKandidat, pilihKandidat, dan akhiriPemilihan. Terdapat beberapa hubungan antar entitas:

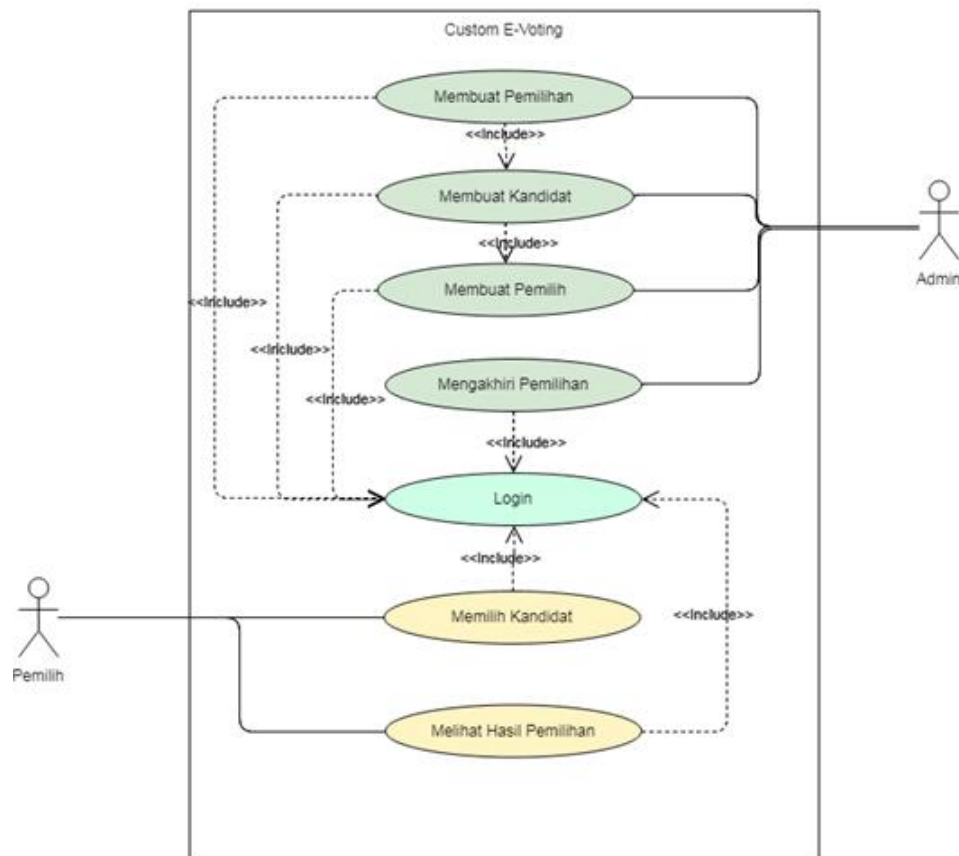
- Satu Pemilihan dapat memiliki banyak kandidat dan memiliki satu kandidat terpilih.
- Satu Pemilih dapat memilih satu Kandidat.
- Untuk membuat satu Pemilihan dapat dibuat dengan satu inisiasiPemilihan dan untuk mengakhirkannya dapat dengan akhiriPemilihan.
- Untuk membuat satu Kandidat dapat dibuat dengan satu inisiasiKandidat.

3.2.3. Diagram Use Case

Sistem yang terdiri dari 2 aktor yaitu Admin dan Pemilih. Setiap Aktor memiliki beberapa *Use Case*:

- Admin dapat Membuat Pemilihan, dapat Membuat Kandidat, dapat membuat pemilih, dan dapat Mengakhiri Pemilihan.
- Pemilih dapat Memilih Kandidat dan dapat Melihat Hasil Pemilihan

Diagram *Use Case* Sistem ditunjukkan pada gambar 3-3.



Gambar 3-5 Diagram Use Case Sistem

3.3. Perancangan Business Network Blockchain dengan Hyperledger

Perancangan *Business Network* atau jaringan bisnis merupakan *key-concept* dari model pemrograman Hyperledger Composer. Business Network bertujuan untuk pembuatan blockchain dengan Hyperledger. Business Network dalam pembuatannya membutuhkan 3 file yaitu:

1. Model file dimana file berjenis .cto berisi pendefinisian partisipan, Asset, dan transaksi.
2. Kontrol Hak Akses terhadap partisipan dimana file berjenis .acl bertujuan untuk membatasi aktivitas partisipan.
3. Logika file berjenis .js bertujuan untuk mendefinisikan logika berupa fungsi pada sistem yang dibuat.

3.3.1. Penentuan Partisipan

Pada bagian ini adalah bagian penentuan partisipan dimana partisipan merupakan aktor yang terlibat di dalam jaringan. Penentuan partisipan dilakukan berdasarkan dari kebutuhan aktor pada Sistem. Tabel 3-2 berikut menunjukkan daftar partisipan selaku aktor yang terlibat di dalam jaringan blockchain custom e-vote.

No	Partisipan
1	Admin
2	Pemilih

Tabel 3-2 Daftar Partisipan

Dari tabel diatas, dapat diketahui terdapat 2 partisipan yang terlibat di dalam jaringan blockchain *custom e-voting*. Partisipan tersebut adalah Admin dan Pemilih. Admin bertindak sebagai pihak yang mengontrol sistem baik *asset*, partisipan, dan beberapa transaksi. Pemilih bertindak sebagai partisipan yang melakukan transaksi dimana memilih kandidat yang ada.

3.3.2. Penentuan *Asset*

Pada bagian ini merupakan bagian penentuan asset dimana *asset* merupakan segala sesuatu yang berharga pada jaringan blockchain custom e-voting. Aset-aset ini berisi data-data dari proses transaksi yang dilakukan. Tabel 3-3 berikut menunjukan daftar *asset* selaku sesuatu yang berharga di dalam jaringan blockchain custom e-vote.

No	Asset	Value
1	Pemilihan	<ul style="list-style-type: none"> - pemilihanId - namaPemilihan - tanggalMulai - tanggalBerakhir - statusPemilihan - kandidatTerpilih
2	Kandidat	<ul style="list-style-type: none"> - kandidatId - namaLengkap - urlFotoKandidat - jumlahVote - pemilihan

Tabel 3-3 Daftar *Asset* dan *Value*

Keterangan *Value* pada setiap *Asset*:

1. Pemilihan :

- a) pemilihanId merupakan identitas unik dari Pemilihan dimana jika id tidak ditentukan maka akan di buat secara acak.
- b) namaPemilihan merupakan nama dari Pemilihan yang dibuat.
- c) tanggalMulai merupakan tanggal dimulainya pemilihan.
- d) tanggalBerakhir merupakan tanggal berakhirnya pemilihan.
- e) statusPemilihan merupakan keterangan status pemilihan berupa tipe boolean menerangkan aktif atau tidak pemilihan.

- f) kandidatTerpilih merupakan kandidat yang memiliki suara terbanyak dalam pemilihan

2. Kandidat :

- a) kandidatId merupakan identitas unik dari Kandidat dimana jika id tidak ditentukan maka akan di buat secara acak.
- b) namaLengkap merupakan nama lengkap dari Kandidat.
- c) urlFotoKandidat merupakan alamat penyimpanan foto Kandidat.
- d) jumlahVote merupakan jumlah suara yang diperoleh Kandidat
- e) pemilihan merupakan keterangan pemilihan yang diikuti Kandidat.

3.3.3. Penentuan Transaksi

Pada bagian ini merupakan bagian penentuan transaksi dimana transaksi merupakan mekanisme partisipan berinteraksi dengan asset. Tabel 3-4 berikut menunjukan daftar transaksi yang ada dalam sistem ini

No	Transaksi	Partisipan	Keterangan
1	Inisiasi Pemilihan	Admin	Transaksi ini untuk membuat <i>asset</i> pemilihan.
2	Inisiasi Kandidat	Admin	Transaksi ini untuk membuat <i>asset</i> kandidat.
3	Pilih Kandidat	Pemilih	Transaksi ini untuk Pemilih memilih Kandidat dan menambah jumlah suara pada kandidat.
4	Akhiri Pemilihan	Admin	Transaksi ini untuk mengakhiri pemilihan.

Tabel 3-4 Daftar transaksi dan hak partisipan

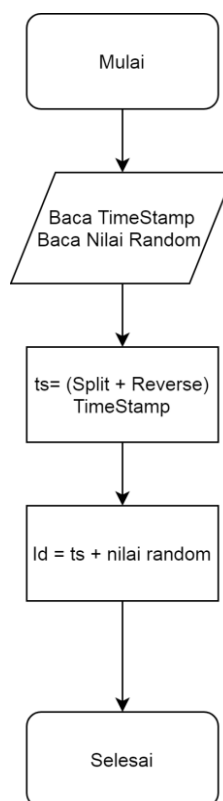
Dari tabel diatas, terdapat 4 transaksi yang ada di dalam jaringan blockchain custom e-voting. Transaksi tersebut adalah Inisiasi Pemilihan, Inisiasi Kandidat Pilih

Kandidat, dan Akhiri Pemilihan dimana dapat dilakukan oleh partisipan untuk berinteraksi dengan asset sesuai dengan hak partisipan.

3.3.4. Penentuan *Logic File*

Pada bagian ini merupakan bagian penentuan logic file yang berisi logika fungsi membuat id secara acak, fungsi membuat asset pemilihan, fungsi membuat asset kandidat, fungsi memperbaharui status telah memilih dari pemilih, dan fungsi untuk memperbaharui jumlah suara kandidat.

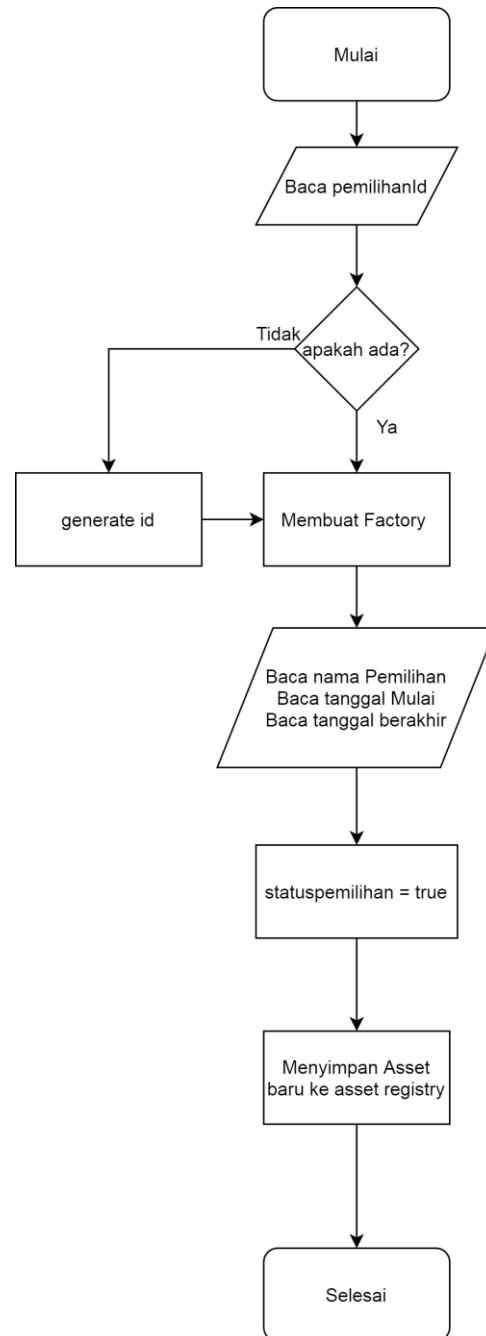
1. Fungsi `generateId` untuk membuat id secara acak yang akan digunakan untuk asset id Pemilihan dan Kandidat



Gambar 3-6 Logika fungsi untuk membuat id

Fungsi `generateId` ini berisi logika untuk membuat Id yang akan dibutuhkan untuk membuat Asset Pemilihan dan Kandidat. Id didapat dari kombinasi dari timestamp yaitu waktu saat digunakan lalu digunakan fungsi *reverse* dan *split* pada timestamp tersebut dan dikombinasikan dengan nilai integer acak.

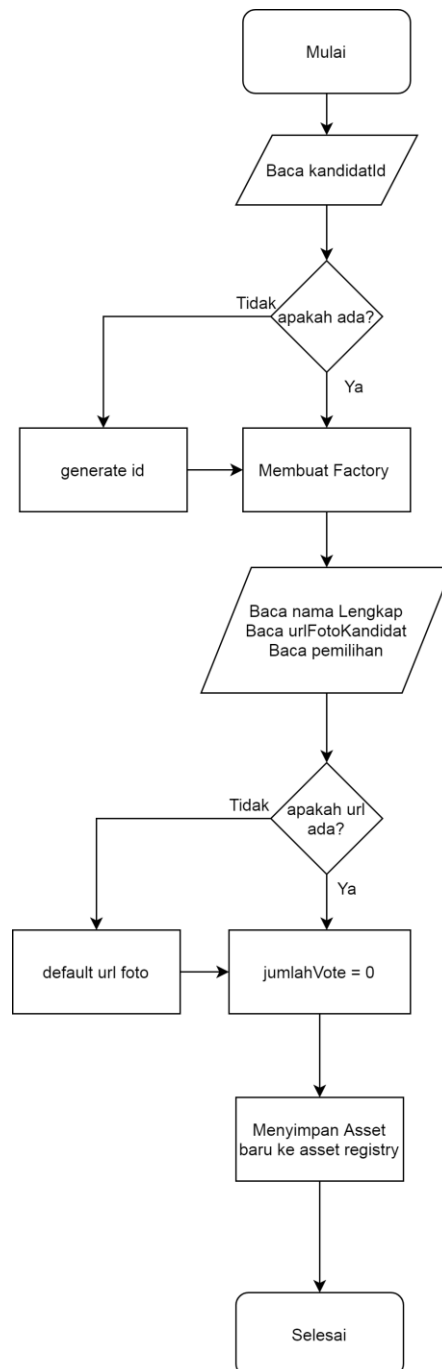
2. Fungsi inisiasiPemilihan untuk membuat Asset Pemilihan



Gambar 3-7 Logika Fungsi untuk membuat asset Pemilihan

Fungsi inisiasiPemilihan ini berisi logika untuk membuat asset Pemilihan di mana membutuhkan pemilihanId yang dapat dibuat secara acak, nama pemilihan, tanggal mulai, tanggal berakhir, dan status pemilihan menjadi otomatis true yang berarti aktif.

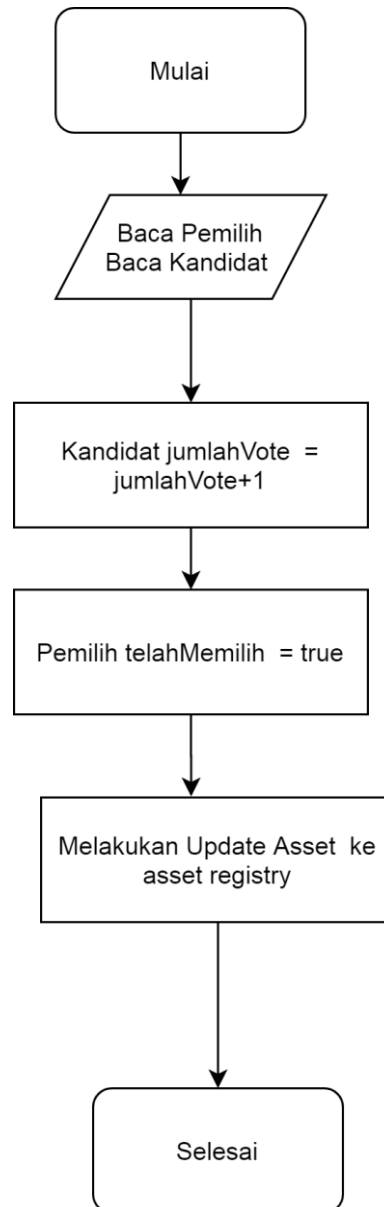
3. Fungsi Membuat Asset Kandidat



Gambar 3-8 Logika Fungsi untuk membuat asset Kandidat

Fungsi inisiasiKandidat ini berisi logika untuk membuat asset Kandidat di mana membutuhkan id kandidat yang dapat dibuat secara acak, nama lengkap, url foto kandidat dimana jika tidak ada maka akan url default, dan jumlah vote otomatis 0.

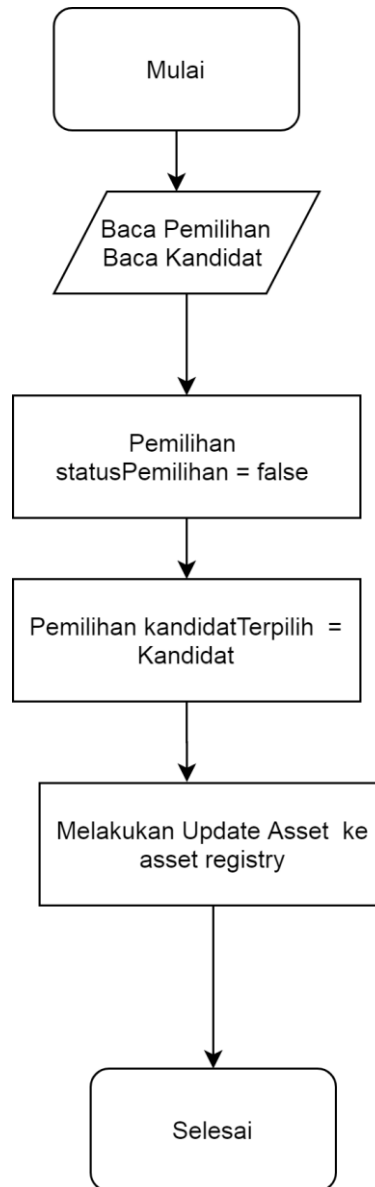
4. Fungsi untuk memilih kandidat



Gambar 3-9 Logika Fungsi untuk memilih Kandidat dalam Pemilihan

Fungsi pilihKandidat ini berisi logika untuk melakukan pemilihan bagi pemilih di mana membutuhkan asset Kandidat dan Pemilih. Asset Kandidat akan diupdate pada jumlah vote yang didapat ditambah satu setiap kali pemilih memilih kandidat tersebut, dan Asset Pemilih akan berubah status pada telahMemilih menjadi true yang mengindikasikan pemilih sudah melakukan pemilihan.

5. Fungsi Mengakhiri Pemilihan



Gambar 3-10 Logika untuk mengakhiri Pemilihan

Fungsi akhiriPemilihan ini berisi logika untuk mengakhir dan menentukan kandidat yang terpilih. Asset Pemilihan akan diupdate pada statusPemilihan dimana menjadi false yang berarti pemilihan berakhir atau sudah tidak aktif dan kandidat terpilih dimana didapat berdasarkan kandidat yang memiliki jumlah vote tertinggi pada pemilihan tersebut.

3.3.5. Penentuan Akses Kontrol

Pada bagian ini merupakan bagian penentuan hak akses bagi partisipan. Penentuan Hak akses berdasarkan dari kebutuhan aktivitas pada Sistem. Tabel 3-5 sampai 3-7 berikut menunjukkan daftar hak akses pada sistem.

No	User	Partisipan	
		Admin	Pemilih
1	Admin	C R U D	C R U D
2	Pemilih	-	R U

Tabel 3-5 Daftar Hak Akses User terhadap partisipan

No	Partisipan	Transaksi			
		Inisiasi Pemilihan	Inisiasi Kandidat	Pilih Kandidat	Akhir Pemilihan
1	Admin	Ya	Ya	Tidak	Ya
2	Pemilih	Tidak	Tidak	Ya	Tidak

Tabel 3-6 Daftar Hak Akses Partisipan terhadap Transaksi

No	Partisipan	Asset	
		Kandidat	Pemilihan
1	Admin	C R U D	C R U D
2	Pemilih	R U	R

Tabel 3-7 Daftar Hak Akses Partisipan terhadap Asset

Keterangan : *C=Create, R=Read, U=Update, D=Delete*

Dari tabel diatas, telah ditentukan hak akses atas asset dan transaksi untuk partisipan. Untuk memperjelas batasan aktivitas yang dapat dilakukan partisipan. Tabel 3-8 berikut menunjukan daftar aktivitas dan hak akses partisipan.

No.	Partisipan	Aktivitas	Izin
1.	Admin	Melakukan fungsi <i>create, read, update, delete</i> pada partisipan Admin.	Bisa
2.	Admin	Melakukan fungsi <i>create, read, update, delete</i> pada partisipan Pemilih.	Bisa
3.	Admin	Melakukan fungsi <i>create, read, update, delete</i> pada asset Kandidat.	Bisa
4.	Admin	Melakukan fungsi <i>create, read, update, delete</i> pada asset Pemilihan.	Bisa
5.	Admin	Melakukan fungsi <i>create</i> pada transaksi Inisiasi Pemilihan.	Bisa
6.	Admin	Melakukan fungsi <i>create</i> pada transaksi Inisiasi Kandidat.	Bisa
7.	Admin	Melakukan fungsi <i>create</i> pada transaksi Pilih Kandidat.	Tidak Bisa
8.	Admin	Melakukan fungsi <i>create</i> pada transaksi Akhir Pemilihan.	Bisa
9.	Pemilih	Melakukan fungsi <i>create, read, update, delete</i> pada partisipan Admin.	Tidak Bisa
10.	Pemilih	Melakukan fungsi <i>create, delete</i> , pada partisipan Pemilih.	Tidak Bisa

11.	Pemilih	Melakukan fungsi <i>read, update</i> , pada partisipan Pemilih.	Bisa
12.	Pemilih	Melakukan fungsi <i>create, delete</i> , pada asset Kandidat.	Tidak Bisa
13.	Pemilih	Melakukan fungsi <i>read, update</i> , pada asset Kandidat.	Bisa
14.	Pemilih	Melakukan fungsi <i>create, update, delete</i> , pada asset Pemilihan.	Tidak Bisa
15.	Pemilih	Melakukan fungsi <i>read</i> pada asset Pemilihan.	Bisa
16.	Pemilih	Melakukan fungsi <i>create</i> pada transaksi Inisiasi Pemilihan.	Tidak Bisa
17.	Pemilih	Melakukan fungsi <i>create</i> pada transaksi Inisiasi Kandidat.	Tidak Bisa
18.	Pemilih	Melakukan fungsi <i>create</i> pada transaksi Pilih Kandidat.	Bisa
19.	Pemilih	Melakukan fungsi <i>create</i> pada transaksi Akhir Pemilihan.	Tidak Bisa

Tabel 3-8 Daftar Aktivitas dan Hak akses Partisipan

Dari tabel diatas, telah ditentukan daftar aktivitas dan hak akses untuk masing-masing Partisipan. Pembuatan hak akses dibuat agar sumber daya pada jaringan dapat dikontrol sesuai dengan perancangan.

BAB 4

IMPLEMENTASI DAN PEMBAHASAN

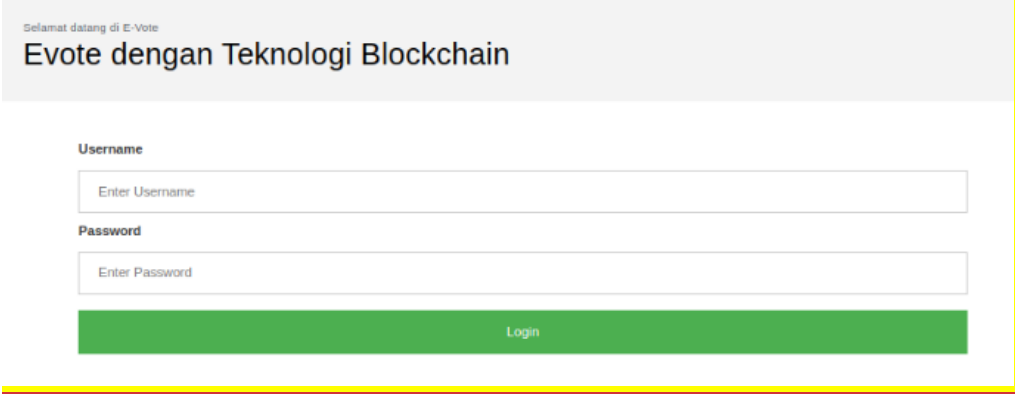
Pada bab pembahasan ini menjelaskan pengujian dan evaluasi terhadap rancangan sistem yang telah diimplementasikan sebelumnya. Pengujian dan evaluasi ini dilakukan dengan tujuan untuk membuktikan konsep dan mengukur performa dari sistem *Custom E-Voting*. Konsep yang ingin dibuktikan adalah apakah teknologi blockchain dapat menangani kasus seperti Pemilu yang bersifat kritikal, masif, dan serentak.

4.1. Antarmuka Sistem dengan Angular

Antarmuka ini dibuat dengan Angular dengan tujuan untuk memudahkan partisipan dalam melakukan transaksi dalam sistem.

4.1.1. Antarmuka Login untuk Partisipan melakukan

Sebelum partisipan melakukan transaksi di dalam sistem, Partisipan diharuskan untuk login terlebih dahulu.



Gambar 4-1 Antarmuka Login untuk Partisipan sebelum melakukan transaksi

4.1.2. Antarmuka untuk partisipan Admin

4.1.2.1 Antarmuka Transaksi Inisiasi Pemilihan

Antarmuka Data Pemilihan untuk admin melakukan transaksi Inisiasi Pemilihan

Selamat Datang di Evote

Data Pemilihan

Logout Pemilihan User

Pemilihan + | Pemilihan

pemilihanId	Nama Pemilihan	Lihat Kandidat	Lihat Pemilih	Tanggal Mulai	Tanggal Berakhir	Status Pemilihan	kandidatTerpilih	Aksi
-------------	----------------	----------------	---------------	---------------	------------------	------------------	------------------	------

Gambar 4-2 Antarmuka Data Pemilihan sebelum admin melakukan transaksi Inisiasi Pemilihan

Selamat Datang di Evote

Data Pemilihan

Logout Pemilihan User

Pemilihan + | Pemilihan

Tambah Pemilihan

Harap isi semua fields

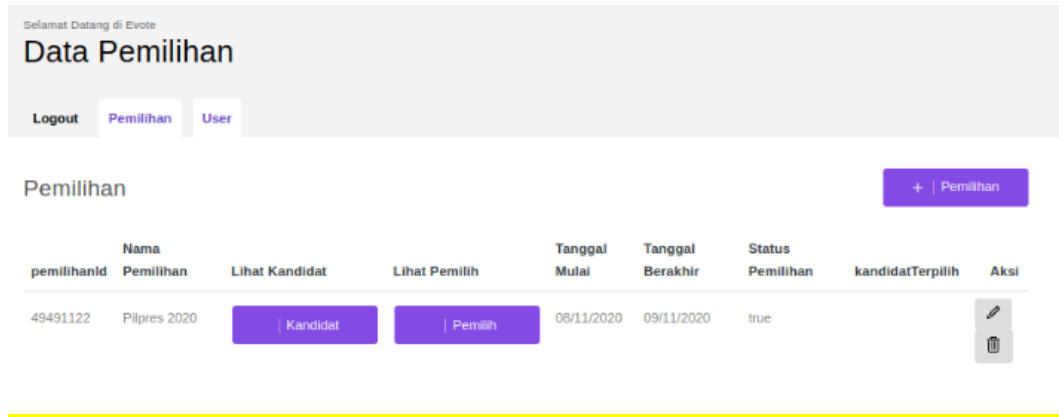
namaPemilihan
Pilpres 2020

tanggalMulai
08/11/2020

tanggalBerakhir
09/11/2020

Tidak Ya

Gambar 4-3 Antarmuka Data Pemilihan saat Admin akan melakukan transaksi Inisiasi Pemilihan dengan mengisi data yang dibutuhkan



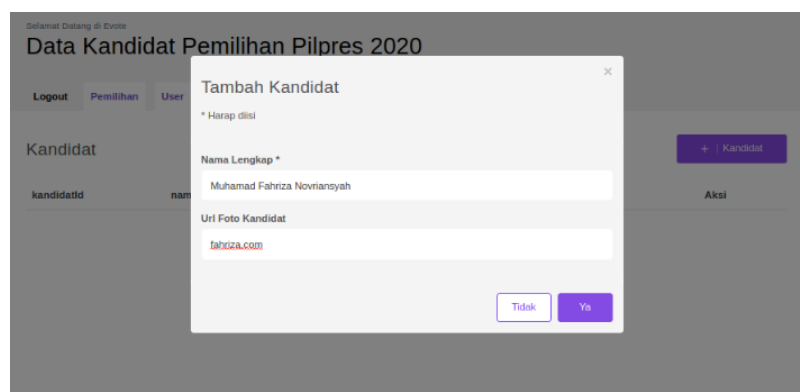
Gambar 4-4 Antarmuka Data Pemilihan setelah admin melakukan berhasil transaksi Inisiasi Pemilihan dimana isi dari Data Pemilihan bertambah

4.1.2.2 Antarmuka Transaksi Inisiasi Kandidat

Antarmuka Data Kandidat untuk admin melakukan transaksi Inisiasi Kandidat



Gambar 4-5 Antarmuka Data Kandidat sebelum admin melakukan transaksi Inisiasi Kandidat



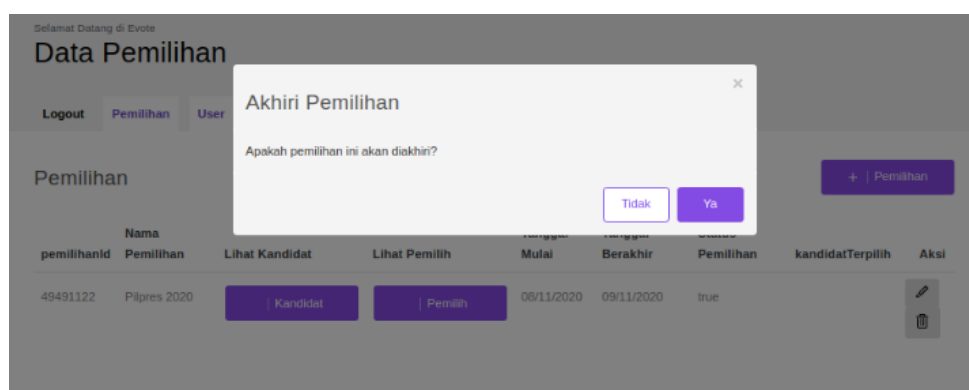
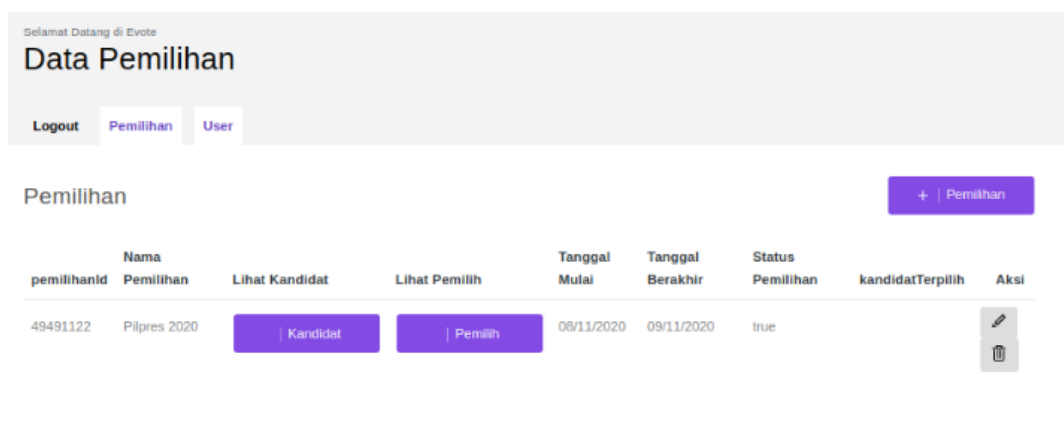
Gambar 4-6 Antarmuka Data Kandidat saat Admin akan melakukan transaksi InisiasiKandidat dengan mengisi data yang dibutuhkan.



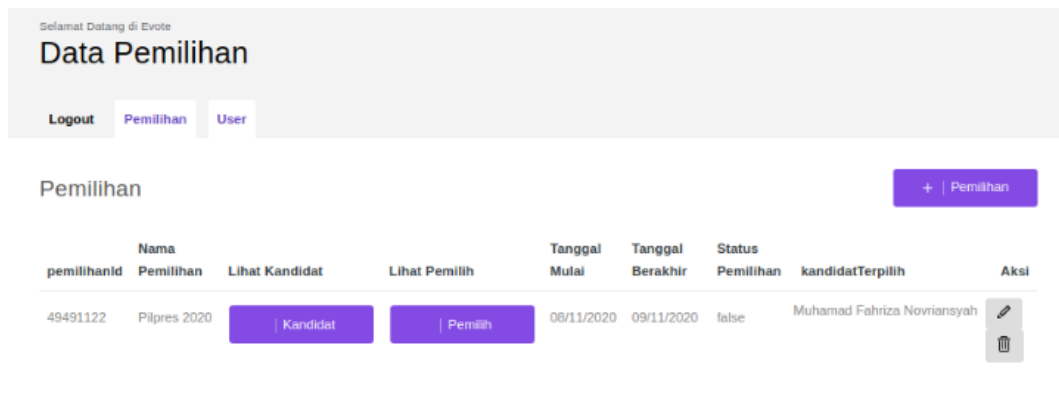
Gambar 4-7 Antarmuka Data Kandidat setelah admin melakukan berhasil transaksi Inisiasi Kandidat dimana isi dari Data Kandidat bertambah.

4.1.2.3 Antarmuka Transaksi Akhiri Pemilihan

Antarmuka Data Pemilihan untuk admin melakukan transaksi Akhiri Pemilihan.



Gambar 4-8 Antarmuka Data Pemilihan sebelum dan saat admin akan melakukan transaksi Akhiri Pemilihan



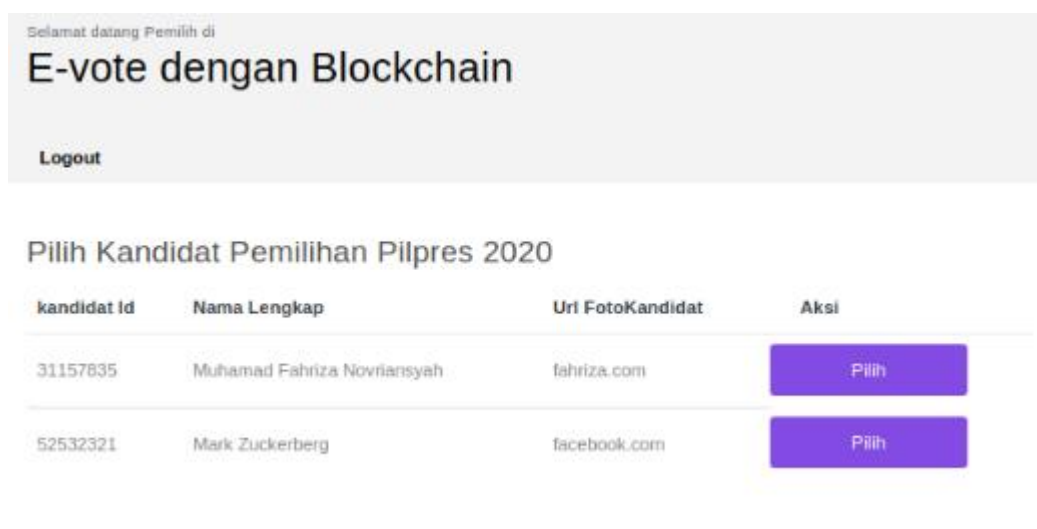
Gambar 4-9 Antarmuka Data Pemilihan setelah admin melakukan berhasil transaksi akhiri Pemilihan.

Setelah Admin berhasil melakukan transaksi Akhir Pemilihan maka status Pemilihan akan berubah menjadi false dan Kandidat Terpilih akan otomatis terisi dengan Nama Kandidat yang memiliki jumlah vote tertinggi.

4.1.3. Antarmuka untuk partisipan Pemilih

4.1.3.1 Antarmuka Transaksi Pilih Kandidat

Antarmuka Pilih Kandidat untuk Pemilih melakukan transaksi Pilih Kandidat



Gambar 4-10 Antarmuka Pilih Kandidat sebelum Pemilih melakukan transaksi Pilih Kandidat

Selamat datang Pemilih di

E-vote dengan Blockchain

Logout

Pilih Kandidat Pemilih

kandidat Id	Nama Lengkap
31157835	Muhamad Fahriza Novriansyah
52532321	Mark Zuckerberg

Pilih Kandidat

Yakin dengan memilih kandidat berikut?

ID kandidat

31157835

Nama Lengkap

Muhamad Fahriza Novriansyah

Tidak Ya

Gambar 4-11 Antarmuka Pilih Kandidat saat Pemilih saat akan melakukan transaksi Pilih Kandidat dengan menekan button Pilih sesuai dengan Kandidat yang ingin dipilih.

Selamat datang Pemilih di

E-vote dengan Blockchain

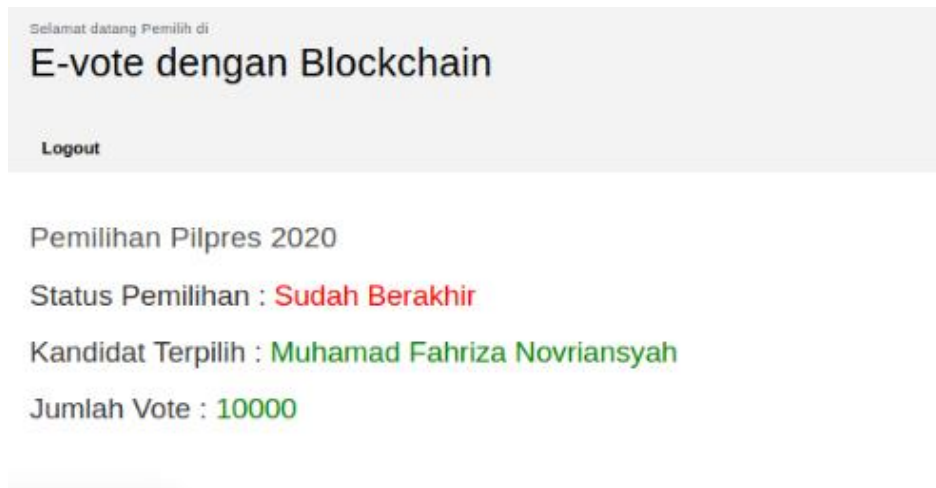
Logout

Pilih Kandidat Pemilihan Pilpres 2020

kandidat Id	Nama Lengkap	Url FotoKandidat	Aksi
31157835	Muhamad Fahriza Novriansyah	fahriza.com	Sudah Memilih
52532321	Mark Zuckerberg	facebook.com	Sudah Memilih

Gambar 4-12 Antarmuka Pilih Kandidat saat Pemilih berhasil melakukan transaksi Pilih Kandidat ketika pemilihan belum berakhir

Setelah Pemilih berhasil melakukan transaksi Pilih Kandidat dan Jika Pemilihan belum berakhir maka Pemilih tidak dapat melakukan Pilih Kandidat lagi dengan keterangan sudah memilih.



Gambar 4-13 Antarmuka Pilih Kandidat saat Pemilih berhasil saat pemilihan sudah berakhir

Jika Pemilihan sudah berakhir maka Pemilih akan melihat Hasil Pemilihan berupa Nama Kandidat yang terpilih dan Jumlah Vote yang didapat dari kandidat tersebut.

4.1.1.—

4.2. Pengujian

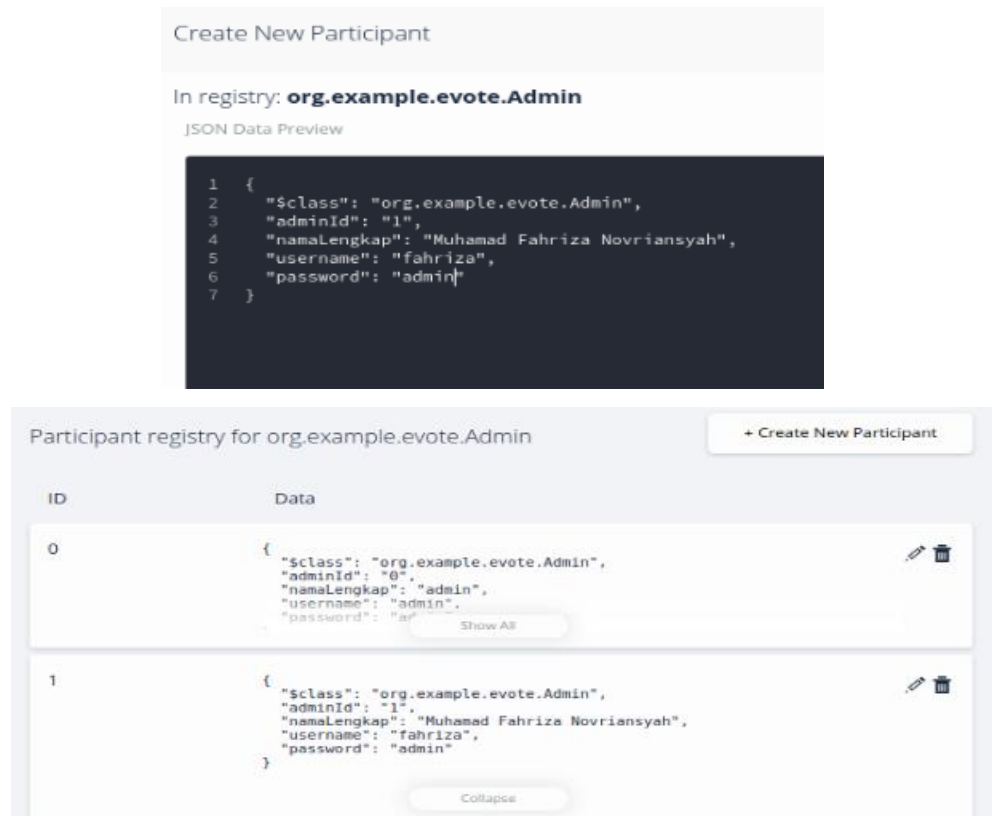
4.2.1. Pengujian Aktivitas Partisipan berdasarkan Hak Akses.

4.1.1.1 Pengujian hak akses Admin

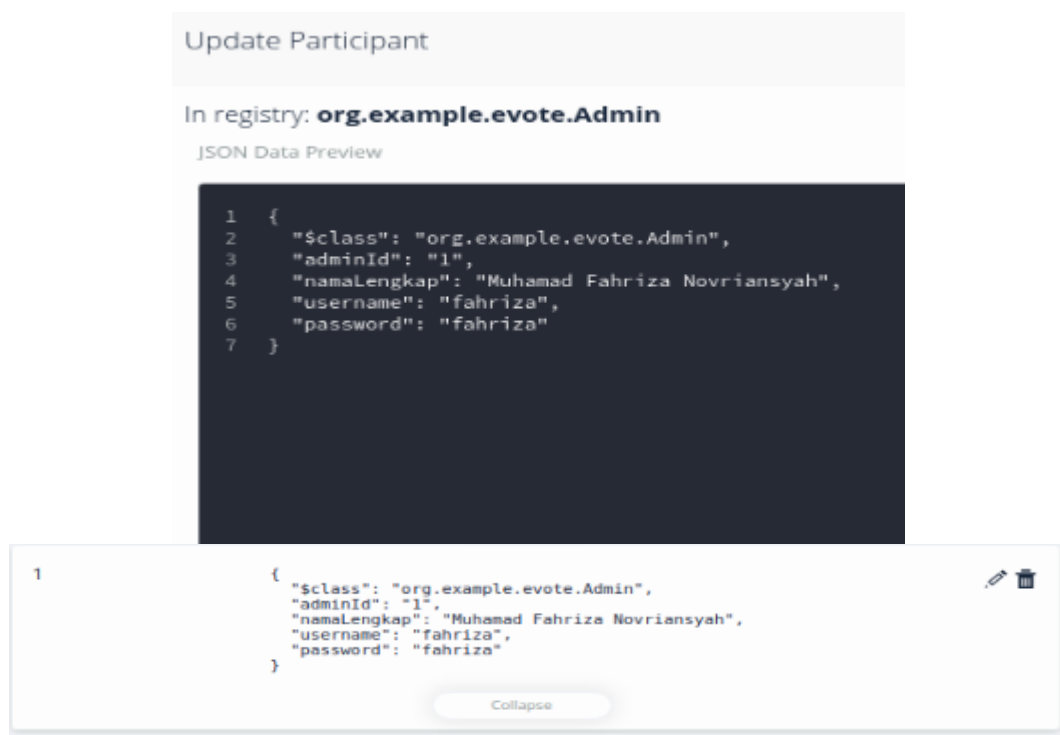
Pengujian berdasarkan aktivitas Admin menggunakan network card Admin.

Pengujian ini bertujuan untuk membuktikan apakah akun pengguna dengan role sebagai Admin benar-benar dapat melakukan tugas/fungsi yang direncanakan, yaitu sebagai berikut:

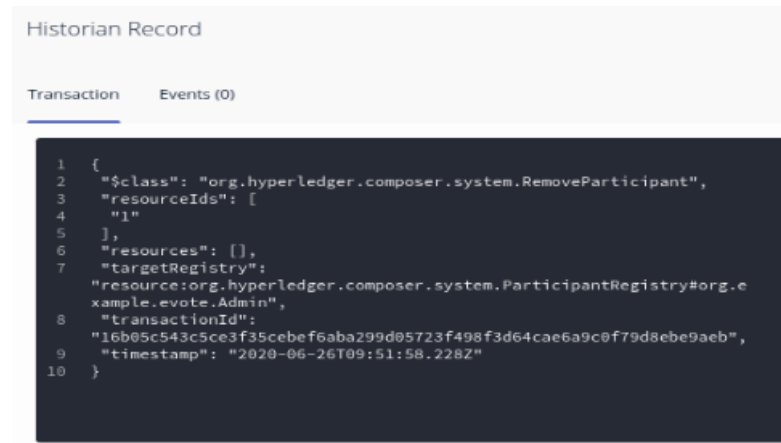
1. Melakukan fungsi *create*, *read*, *update*, *delete* pada Admin.



Gambar 4-14 Admin Menambah dan Membaca Admin



Gambar 4-15 Admin Memperbaharui Admin



Gambar 4-16 Admin Menghapus Admin

Berdasarkan ketiga gambar diatas, ditunjukkan bahwa aktivitas Admin terhadap Admin telah sesuai seperti perancangan. Admin dapat melakukan fungsi Create,Read,Update,Delete pada Admin.

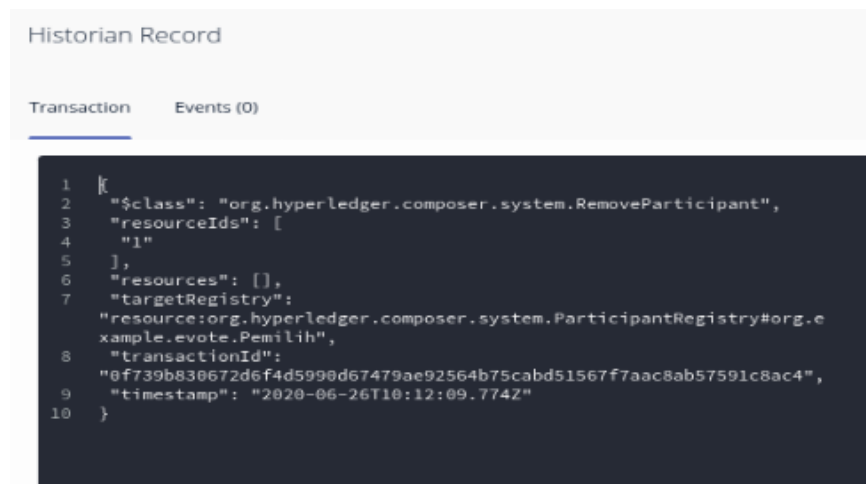
2. Melakukan fungsi *create*, *read*, *update*, *delete* pada partisipan Pemilih.



Gambar 4-17 Admin Menambah dan Membaca Partisipan Pemilih



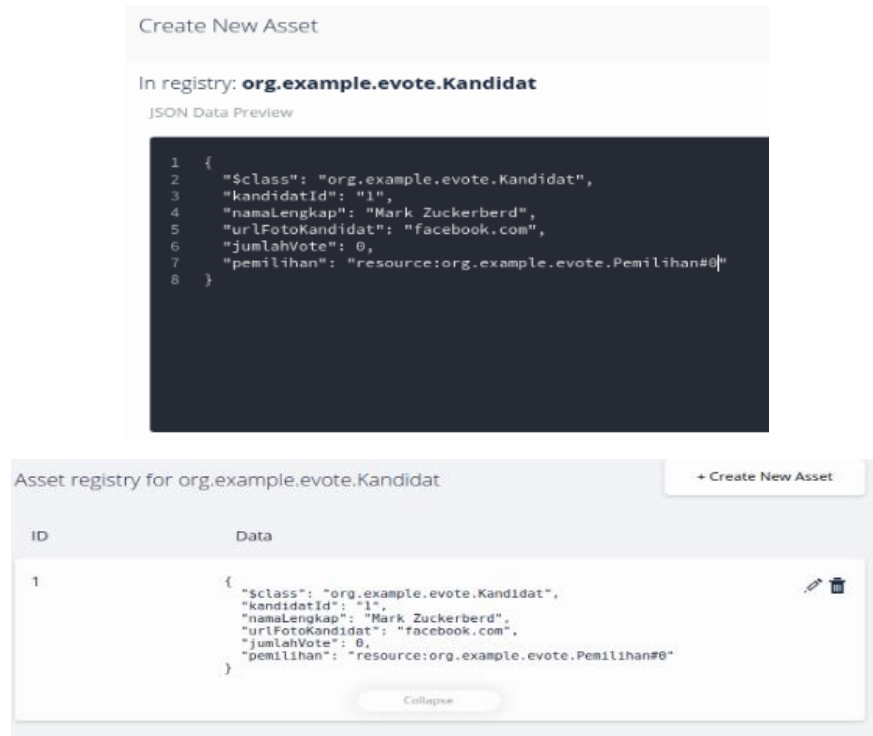
Gambar 4-18 Admin Memperbaharui Partisipan Pemilih



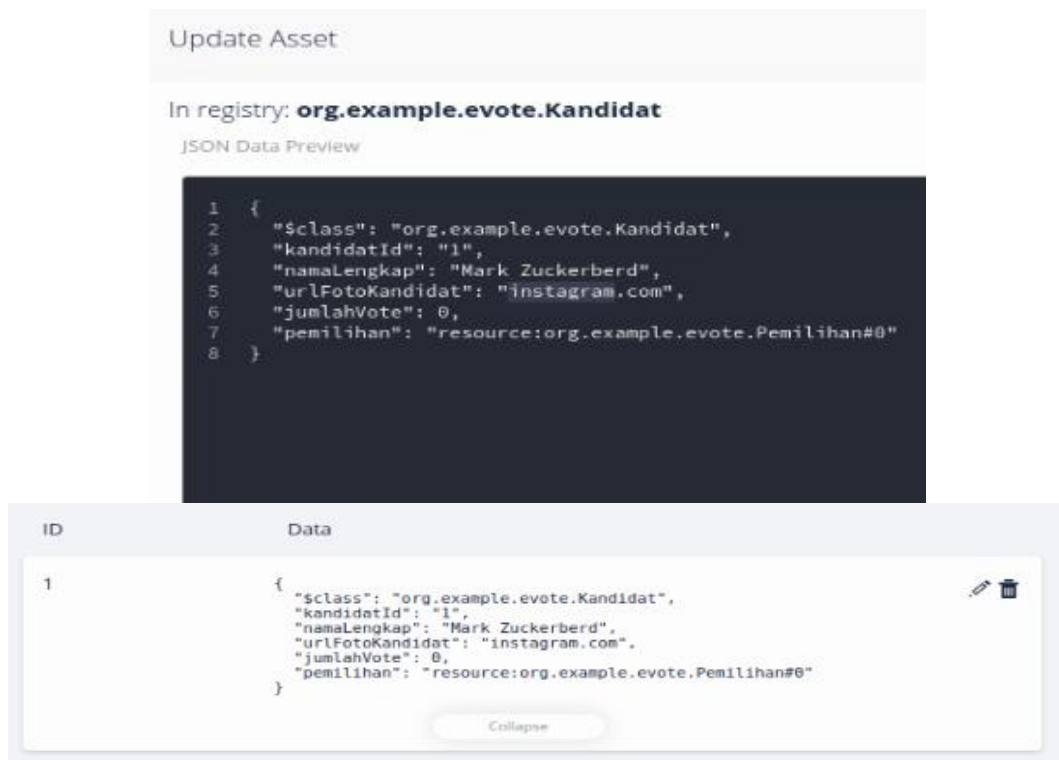
Gambar 4-19 Admin Menghapus Partisipan Pemilih

Berdasarkan ketiga gambar diatas, ditunjukkan bahwa aktivitas Admin terhadap partisipan Pemilih telah sesuai seperti perancangan. Admin tidak dapat melakukan fungsi Create, Read, Update, Delete pada partisipan Pemilih.

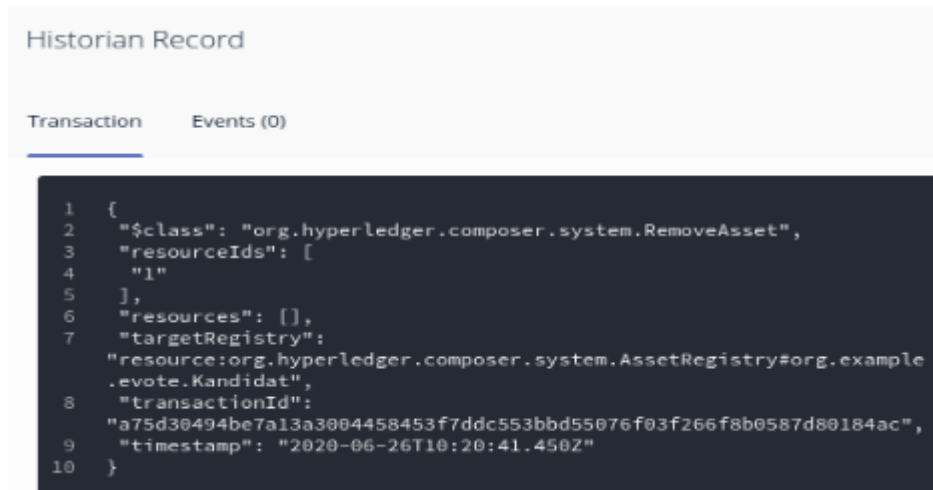
3. Melakukan fungsi *create*, *read*, *update*, *delete* pada Asset Kandidat.



Gambar 4-20 Menambah dan Membaca Asset Kandidat



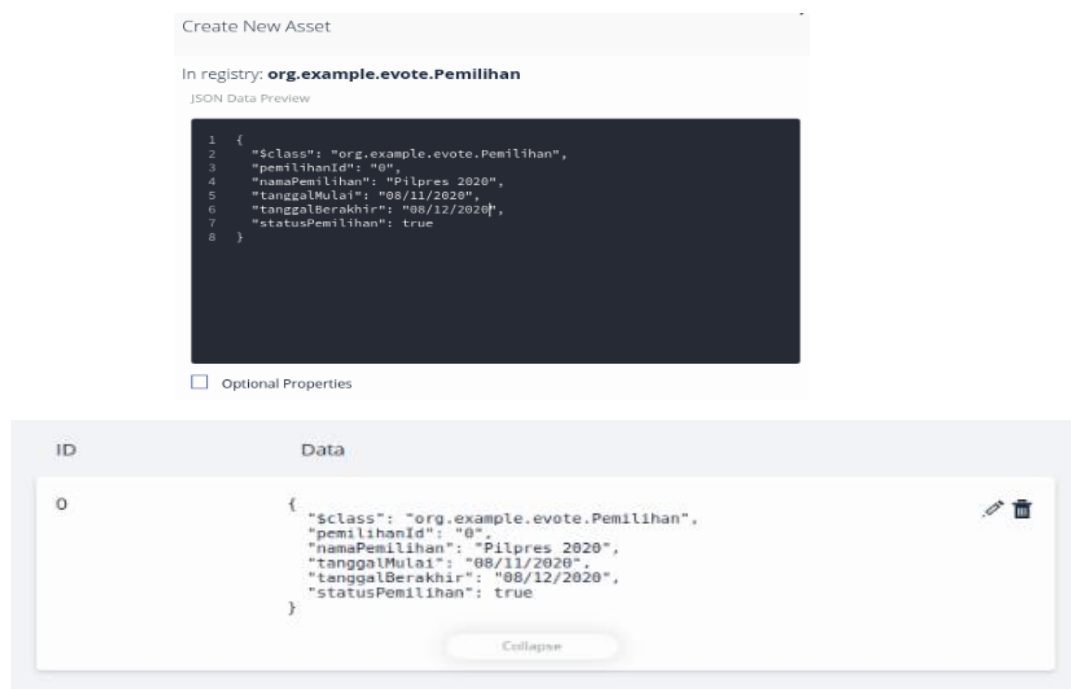
Gambar 4-21 Admin Memperbaharui Asset Kandidat



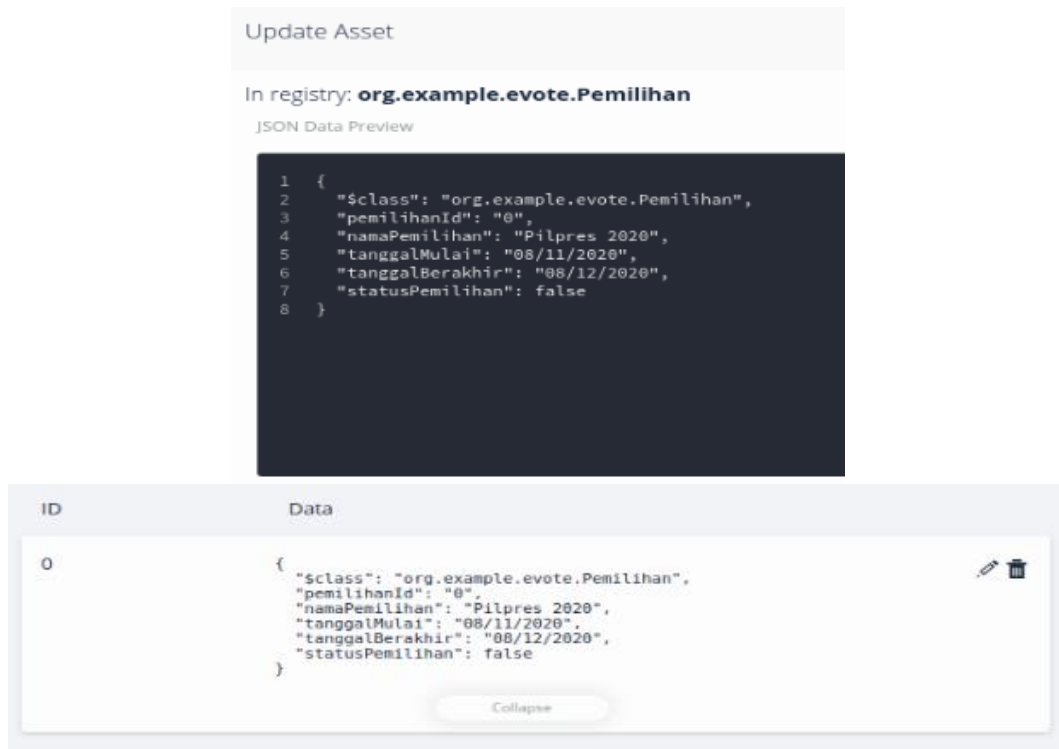
Gambar 4-22 Admin Menghapus Asset Kandidat

Berdasarkan ketiga gambar diatas, ditunjukkan bahwa aktivitas Admin terhadap Asset Kandidat telah sesuai seperti perancangan. Admin dapat melakukan fungsi Create,Read,Update,Delete pada Asset Kandidat.

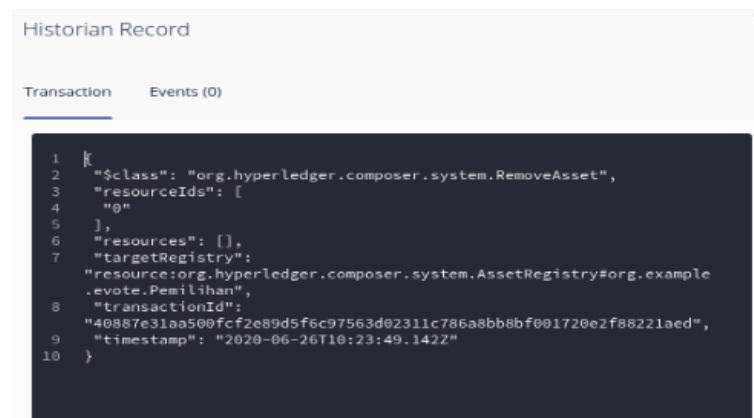
4. Melakukan fungsi *create*, *read*, *update*, *delete* pada Asset Pemilihan.



Gambar 4-23 Admin Menambah dan Membaca Asset Pemilihan



Gambar 4-24 Admin Memperbaharui Asset Pemilihan



Gambar 4-25 Admin Menghapus Asset Pemilihan

Berdasarkan ketiga gambar diatas, ditunjukkan bahwa aktivitas Admin terhadap Asset Pemilihan telah sesuai seperti perancangan. Admin dapat melakukan fungsi Create,Read,Update,Delete pada Asset Pemilihan.

5. Melakukan fungsi *create* pada transaksi Inisiasi Pemilihan.

Submit Transaction

Transaction Type: **InisiasiPemilihan**

JSON Data Preview

```

1 {
2   "$class": "org.example.evote.inisiasiPemilihan",
3   "namaPemilihan": "Pilkada 2020",
4   "tanggalMulai": "08/11/2020",
5   "tanggalBerakhir": "08/12/2020",
6   "statusPemilihan": true
7 }

```

☐ Optional Properties

ID	Data
26118129	<pre> { "\$class": "org.example.evote.Pemilihan", "pemilihanId": "26118129", "namaPemilihan": "Pilkada 2020", "tanggalMulai": "08/11/2020", "tanggalBerakhir": "08/12/2020", "statusPemilihan": true } </pre>

Collapse

Gambar 4-26 Admin Melakukan Transaksi Inisiasi Pemilihan

Berdasarkan gambar diatas, ditunjukkan bahwa aktivitas Admin terhadap transaksi Inisiasi Pemilihan telah sesuai seperti perancangan. Admin dapat melakukan fungsi *Create* pada transaksi Inisiasi Pemilihan.

6. Melakukan fungsi *create* pada transaksi Inisiasi Kandidat.

Submit Transaction

Transaction Type: **InisiasiKandidat**

JSON Data Preview

```

1 {
2   "$class": "org.example.evote.inisiasiKandidat",
3   "namaLengkap": "Muhamad Fahriza Novrinsyah",
4   "pemilihan": "resource:org.example.evote.Pemilihan#26118129"
5 }

```

ID	Data
51717071	<pre> { "\$class": "org.example.evote.Kandidat", "kandidatId": "51717071", "namaLengkap": "Muhamad Fahriza Novrinsyah", "urlFotoKandidat": "url.com", "jumlahVote": 0, "pemilihan": "resource:org.example.evote.Pemilihan#26118129" } </pre>

Gambar 4-27 Admin Melakukan Transaksi Inisiasi Kandidat

Berdasarkan gambar 4-14, ditunjukkan bahwa aktivitas Admin terhadap transaksi Inisiasi Kandidat telah sesuai seperti perancangan. Admin dapat melakukan fungsi *Create* pada transaksi Inisiasi Pemilihan.

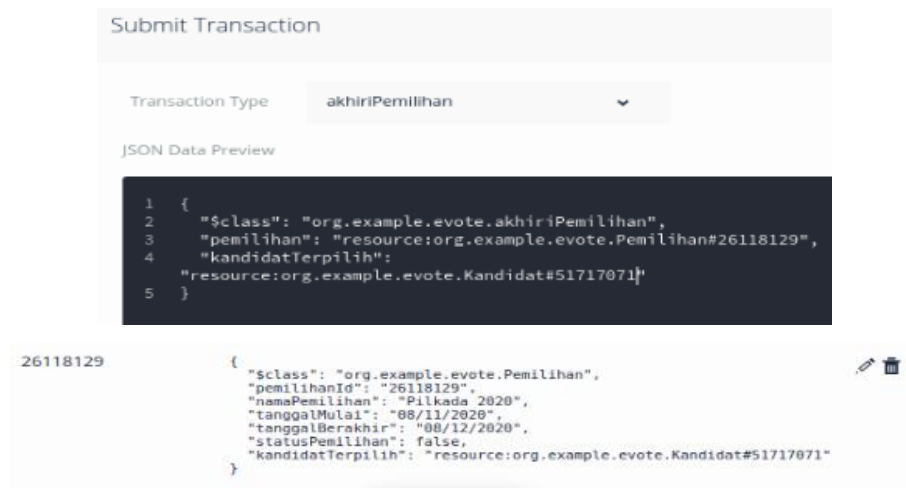
7. Melakukan fungsi *create* pada transaksi Pilih Kandidat.



Gambar 4-28 Admin Melakukan Transaksi Pilih Kandidat

Berdasarkan gambar diatas, ditunjukkan bahwa aktivitas Admin terhadap transaksi Pilih Kandidat telah sesuai seperti perancangan. Admin tidak dapat melakukan fungsi *Create* pada transaksi Pilih Kandidat.

8. Melakukan fungsi *create* pada transaksi Akhir Pemilihan.



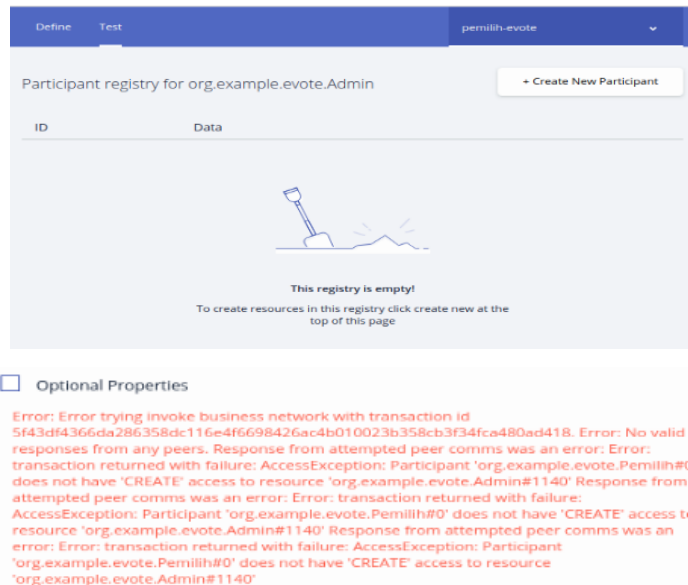
Gambar 4-29 Admin Melakukan Transaksi Akhir Pemilihan

Berdasarkan gambar diatas, ditunjukkan bahwa aktivitas Admin terhadap transaksi Akhir Pemilihan telah sesuai seperti perancangan. Admin dapat melakukan fungsi *Create* pada transaksi Pilih Kandidat.

4.1.1.2 Pengujian hak Akses Pemilih

Pengujian berdasarkan aktivitas Pemilih:

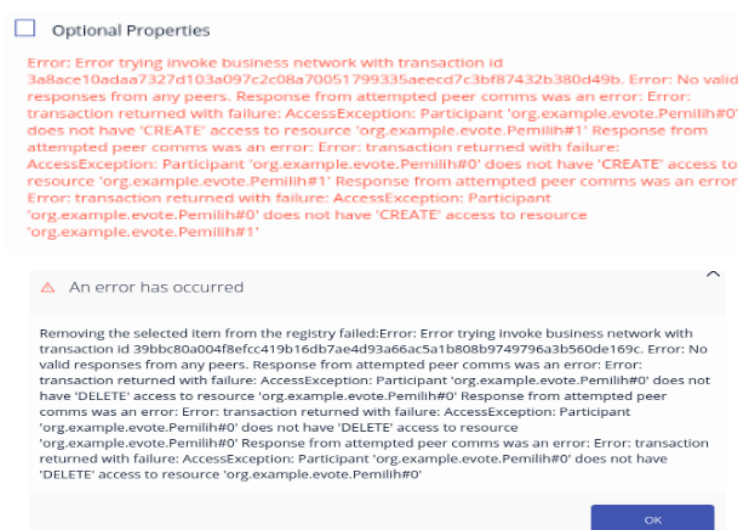
1. Melakukan fungsi *create*, *read*, *update*, *delete* pada partisipan Admin.



Gambar 4-30 Pemilih Membaca dan Menambah Partisipan Admin

Berdasarkan gambar 4-17, ditunjukkan bahwa aktivitas Pemilih terhadap partisipan Admin telah sesuai seperti perancangan. Pemilih tidak dapat melakukan fungsi Create, Read, Update, Delete pada partisipan Admin.

2. Melakukan fungsi *create*, *delete* pada partisipan Pemilih.



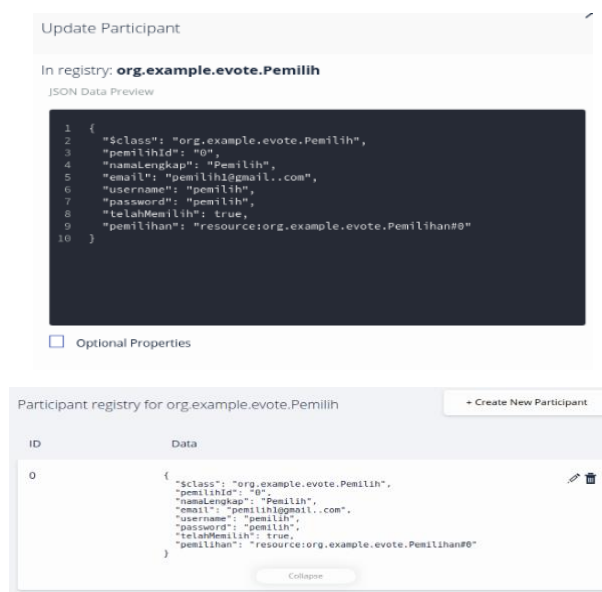
Gambar 4-31 Pemilih Menambah dan Menghapus Partisipan Pemilih

Berdasarkan gambar diatas, ditunjukkan bahwa aktivitas Pemilih terhadap Partisipan Pemilih telah sesuai seperti perancangan. Pemilih tidak dapat melakukan fungsi *Create*, *Delete* pada Partisipan Pemilih.

3. Melakukan fungsi *read*, *update* pada partisipan Pemilih.



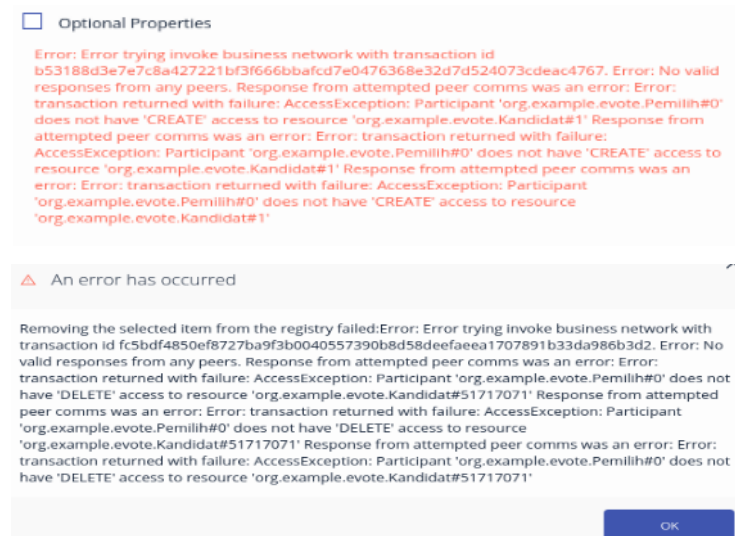
Gambar 4-32 ~~Grafik~~ Pemilih Membaca Partisipan Pemilih



Gambar 4-33 ~~Grafik~~ Pemilih Memperbaharui Partisipan Pemilih

Berdasarkan kedua gambar diatas, ditunjukkan bahwa aktivitas Pemilih terhadap Partisipan Pemilih telah sesuai seperti perancangan. Pemilih dapat melakukan fungsi *Read*, *Update* pada Partisipan Pemilih.

4. Melakukan fungsi *create*, *delete* pada asset Kandidat.



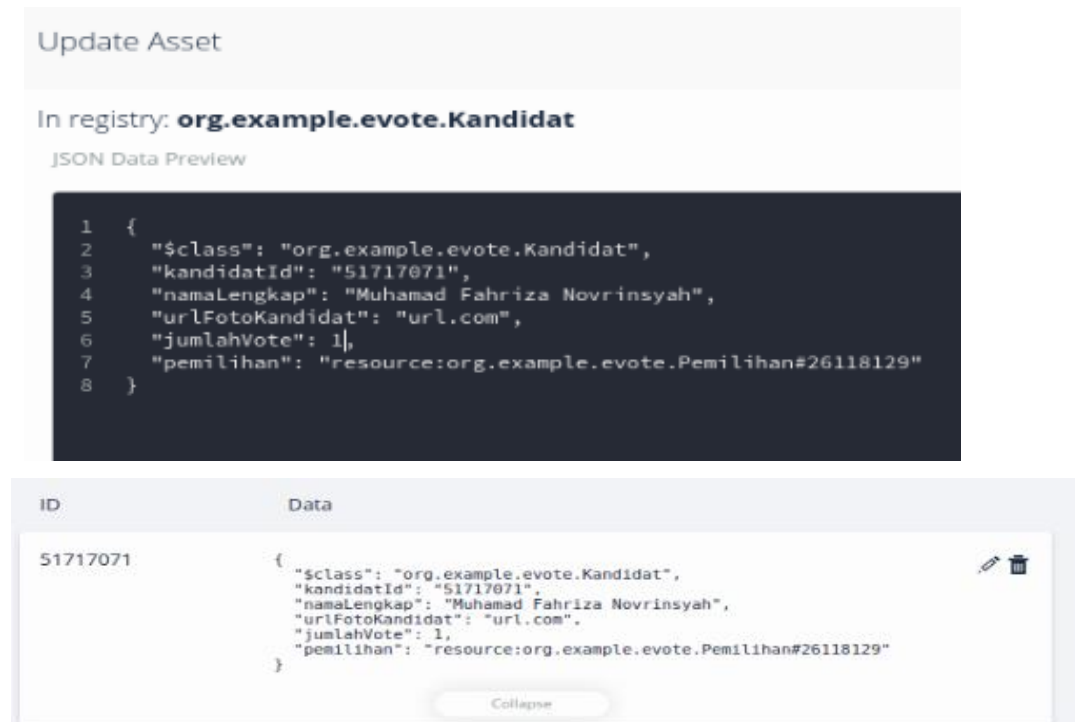
Gambar 4-34 ~~Grafik~~ Pemilih Menambah dan Menghapus Asset Kandidat

Berdasarkan gambar diatas, ditunjukkan bahwa aktivitas Pemilih terhadap Asset Kandidat telah sesuai seperti perancangan. Pemilih tidak dapat melakukan fungsi *Create*, *Delete* pada Asset Kandidat.

5. Melakukan fungsi *read*, *update* pada asset Kandidat.



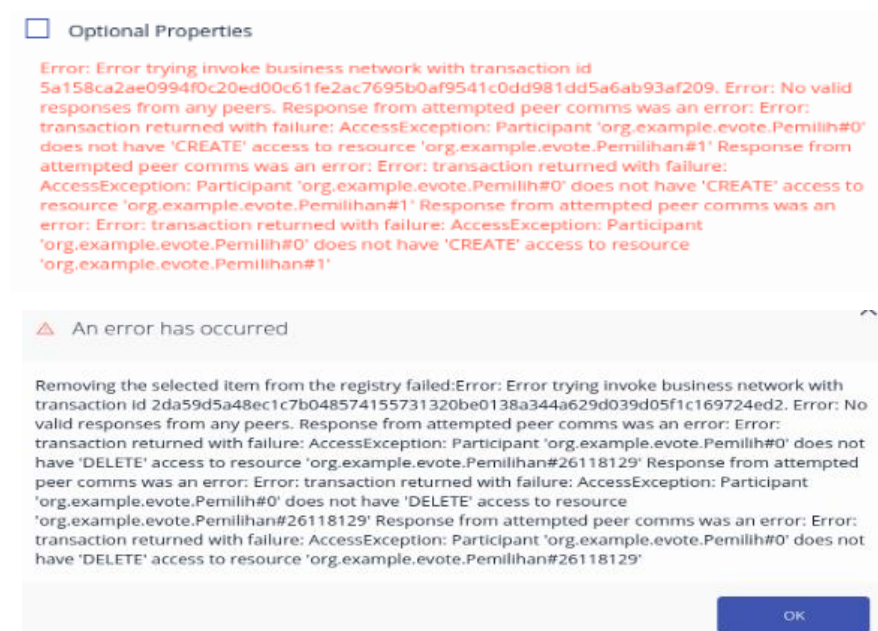
Gambar 4-35 ~~Grafik~~ Pemilih Membaca Asset Kandidat



Gambar 4-36 ~~Grafik~~ Pemilih Memperbaharui Asset Kandidat

Berdasarkan gambar 4-22 dan 4-23, ditunjukkan bahwa aktivitas Pemilih terhadap Asset Kandidat telah sesuai seperti perancangan. Pemilih dapat melakukan fungsi *Read*, *Update* pada Asset Kandidat.

6. Melakukan fungsi *create*, *update*, *delete* pada asset Pemilihan.



Gambar 4-37 ~~Grafik~~ Pemilih Menambah, Memperbaharui dan Menghapus Asset Pemilihan

Berdasarkan gambar 4-24, ditunjukkan bahwa aktivitas Pemilih terhadap Asset Pemilihan telah sesuai seperti perancangan. Pemilih tidak dapat melakukan fungsi *Create*, *Update*, *Delete* pada Asset Pemilihan.

7. Melakukan fungsi *read* pada asset Pemilihan.



Gambar 4-38 ~~Grafik~~ Pemilih Membaca Asset Kandidat

Berdasarkan gambar diatas, ditunjukkan bahwa aktivitas Pemilih terhadap Asset Pemilihan telah sesuai seperti perancangan. Pemilih dapat melakukan fungsi *Read* pada Asset Pemilihan.

8. Melakukan fungsi *create* pada transaksi Inisiasi Pemilihan.



Gambar 4-39 ~~Grafik~~ Admin Melakukan Transaksi Inisiasi Pemilihan

Berdasarkan gambar diatas, ditunjukkan bahwa aktivitas Pemilih terhadap transaksi Inisiasi Pemilihan telah sesuai seperti perancangan. Pemilih tidak dapat melakukan fungsi *Create* pada transaksi Inisiasi Pemilihan.

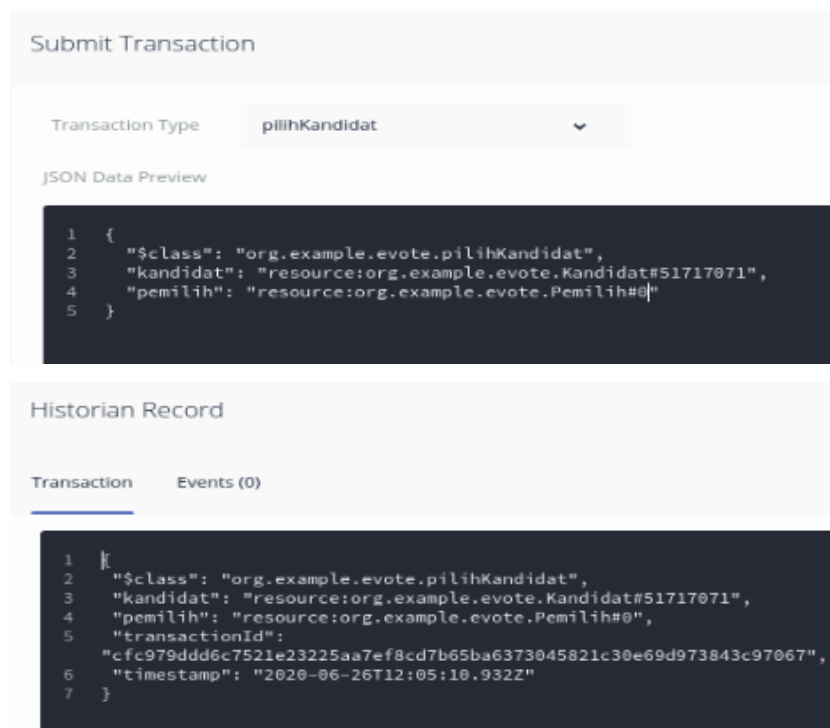
9. Melakukan fungsi *create* pada transaksi Inisiasi Kandidat.



Gambar 4-40 ~~Grafik~~ Pemilih Melakukan Transaksi Inisiasi Kandidat

Berdasarkan gambar diatas, ditunjukkan bahwa aktivitas Pemilih terhadap transaksi Inisiasi Kandidat telah sesuai seperti perancangan. Pemilih tidak dapat melakukan fungsi *Create* pada transaksi Inisiasi Pemilihan.

10. Melakukan fungsi *create* pada transaksi Pilih Kandidat.



Gambar 4-41 ~~Grafik~~ Pemilih Melakukan Transaksi Pilih Kandidat

Berdasarkan gambar 4-28, ditunjukkan bahwa aktivitas Pemilih terhadap transaksi Pilih Kandidat telah sesuai seperti perancangan. Pemilih dapat melakukan fungsi Create pada transaksi Pilih Kandidat.

11. Melakukan fungsi *create* pada transaksi Akhir Pemilihan.



Gambar 4-42 Pemilih Melakukan Transaksi Akhiri Pemilihan

Berdasarkan gambar diatas, ditunjukkan bahwa aktivitas Pemilih terhadap transaksi Akhiri Pemilihan telah sesuai seperti perancangan. Pemilih tidak dapat melakukan fungsi Create pada transaksi Pilih Kandidat.

No.	Partisipan	Aktivitas	Izin	Hasil
1.	Admin	Melakukan fungsi <i>create</i> , <i>read</i> , <i>update</i> , <i>delete</i> pada partisipan Admin.	Bisa	Sesuai
2.	Admin	Melakukan fungsi <i>create</i> , <i>read</i> , <i>update</i> , <i>delete</i> pada partisipan Pemilih.	Bisa	Sesuai
3.	Admin	Melakukan fungsi <i>create</i> , <i>read</i> , <i>update</i> , <i>delete</i> pada asset Kandidat.	Bisa	Sesuai
4.	Admin	Melakukan fungsi <i>create</i> , <i>read</i> , <i>update</i> , <i>delete</i> pada asset Pemilihan.	Bisa	Sesuai

5.	Admin	Melakukan fungsi <i>create</i> pada transaksi Inisiasi Pemilihan.	Bisa	Sesuai
6.	Admin	Melakukan fungsi <i>create</i> pada transaksi Inisiasi Kandidat.	Bisa	Sesuai
7.	Admin	Melakukan fungsi <i>create</i> pada transaksi Pilih Kandidat.	Tidak Bisa	Sesuai
8.	Admin	Melakukan fungsi <i>create</i> pada transaksi Akhir Pemilihan.	Bisa	Sesuai
9.	Pemilih	Melakukan fungsi <i>create</i> , <i>read</i> , <i>update</i> , <i>delete</i> pada partisipan <i>Admin</i> .	Tidak Bisa	Sesuai
10.	Pemilih	Melakukan fungsi <i>create</i> , <i>delete</i> , pada partisipan <i>Pemilih</i> .	Tidak Bisa	Sesuai
11.	Pemilih	Melakukan fungsi <i>read</i> , <i>update</i> , pada partisipan <i>Pemilih</i> .	Bisa	Sesuai
12.	Pemilih	Melakukan fungsi <i>create</i> , <i>delete</i> , pada asset <i>Kandidat</i> .	Tidak Bisa	Sesuai
13.	Pemilih	Melakukan fungsi <i>read</i> , <i>update</i> , pada asset <i>Kandidat</i> .	Bisa	Sesuai
14.	Pemilih	Melakukan fungsi <i>create</i> , <i>update</i> , <i>delete</i> , pada asset <i>Pemilihan</i> .	Tidak Bisa	Sesuai
15.	Pemilih	Melakukan fungsi <i>read</i> pada asset <i>Pemilihan</i> .	Bisa	Sesuai

16.	Pemilih	Melakukan fungsi <i>create</i> pada transaksi Inisiasi Pemilihan.	Tidak Bisa	Sesuai
17.	Pemilih	Melakukan fungsi <i>create</i> pada transaksi Inisiasi Kandidat.	Tidak Bisa	Sesuai
18.	Pemilih	Melakukan fungsi <i>create</i> pada transaksi Pilih Kandidat.	Bisa	Sesuai
19.	Pemilih	Melakukan fungsi <i>create</i> pada transaksi Akhir Pemilihan.	Tidak Bisa	Sesuai

Tabel 4-1 Hasil Pengujian Fungsional Aktivitas Partisipan

Tabel diatas menunjukkan pengujian fungsionalitas aktivitas partisipan telah dilakukan dan mendapatkan hasil yang sesuai dengan perancangan.

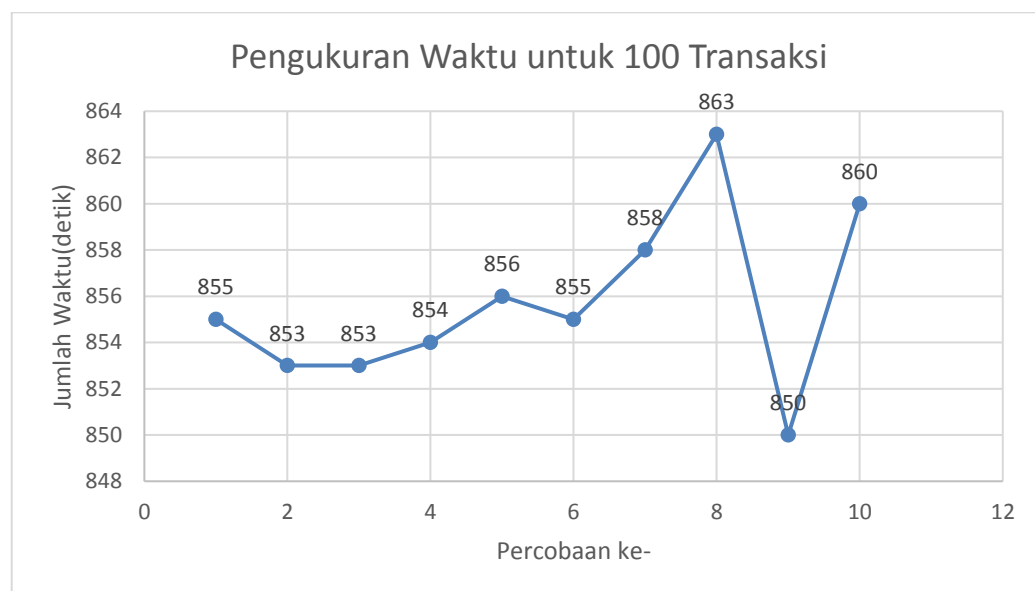
4.2.2. Pengujian Performa berdasarkan Transaksi

Pengujian Performa 100 transaksi untuk membuat Asset Kandidat. Setiap percobaan dilakukan dengan membuat 100 kandidat baru, lalu dicatat lama waktu yang dibutuhkan. percobaan diulang 10 kali dan dihitung rata-rata lama waktunya. Hasilnya ditampilkan pada Tabel 4-2.

Percobaan	Jumlah Waktu 100 Transaksi	
	(Menit dan Detik)	Detik
1	14 Menit 15 Detik	855
2	14 Menit 13 Detik	853
3	14 Menit 13 Detik	853
4	14 Menit 14 Detik	854
5	14 Menit dan 16 Detik	856

6	14 Menit dan 15 Detik	855
7	14 Menit dan 18 Detik	858
8	14 Menit dan 23 Detik	863
9	14 Menit dan 10 Detik	850
10	14 Menit dan 20 Detik	860
Rata-Rata	14 menit 16 detik	856

Tabel 4-2 Hasil Pengujian Kecepatan 100 Transaksi Membuat Kandidat

Gambar 4-43 ~~Grafik~~ Jumlah Waktu yang dibutuhkan dengan Percobaan 100 Transaksi untuk Membuat Kandidat

Percobaan untuk 100 transaksi untuk membuat Asset Kandidat membutuhkan waktu rata-rata 856 detik atau 14 menit 16 detik yang berarti 8,56 detik per transaksi. Kecepatan transaksi ini sangat bergantung pada spesifikasi perangkat yang digunakan. Semakin baik spesifikasi komputer semakin cepat transaksi yang di selesaikan.

4.2.3. Pengujian Performa Memilih Kandidat

Pengujian ini dilakukan untuk mengukur performa saat setiap pemilih memilih salah satu kandidat dalam suatu pemilihan secara serial dan parallel untuk membuat sebuah transaksi dengan variasi *BatchTimeout* yang merupakan waktu tunggu untuk membuat sebuah block dalam blockchain. Selain itu pengujian ini juga menguji apakah sistem dapat mengeksekusi proses tersebut secara parallel. Pengujian Pertama dilakukan dengan menguji 20 Transaksi untuk Pemilih Memilih Kandidat 20 Transaksi secara serial di 1 Perangkat dan 10 Transaksi secara Parallel di 2 Perangkat secara bersamaan, dan diulang sebanyak 35 kali percobaan dengan *BatchTimeout* 250ms. Pengujian Kedua dengan menguji 20 Transaksi Transaksi untuk Pemilih memilih Kandidat secara Parallel, dimana 10 transaksi diproses di mesin pertama dan 10 transaksi diproses di mesin kedua, dan diulang sebanyak 5 kali percobaan.

Percobaan Ke-	BatchTimeout (ms)	Parallel		Serial
		Perangkat 1	Perangkat 2	
		Delay (Detik)	Delay (Detik)	Delay (Detik)
1	250	9.3	9.3	8.9
2	250	9.4	9.5	8.9
3	250	9.5	9.6	10.15
Rata- Rata	250	9.4	9.47	9.32

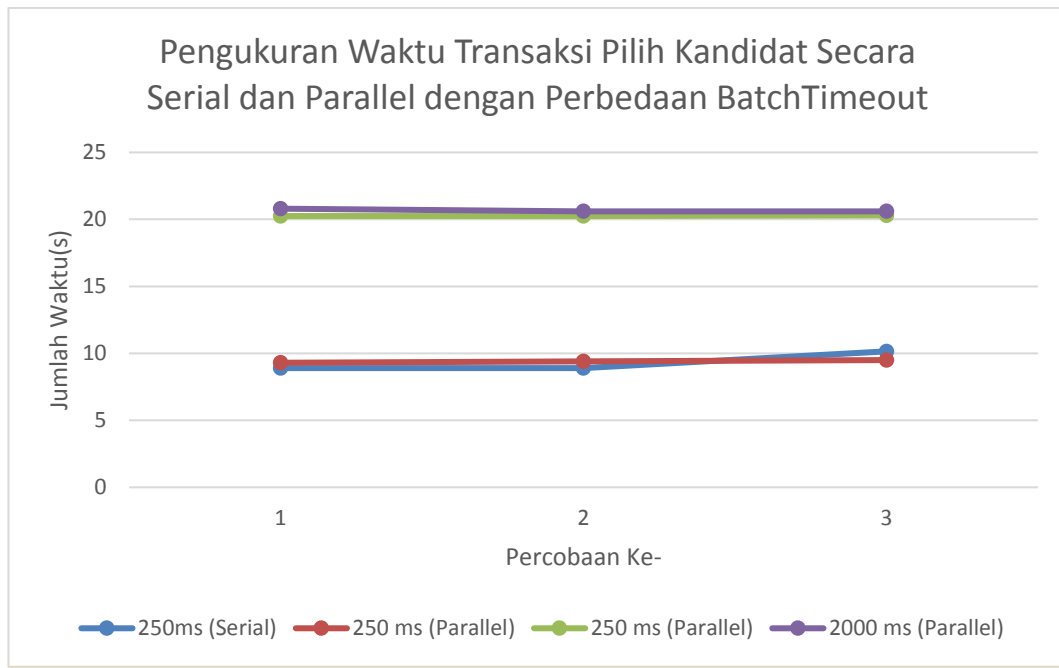
Tabel 4-3 Hasil Pengujian Kecepatan 20 Transaksi Memilih Kandidat Secara Serial Hasil Pengujian Waktu Proses setiap Transaksi memilih Kandidat 20 Transaksi secara Serial dan 10 Transaksi Secara Parallel dengan *BatchTimeout* 250ms

Pengujian Kedua dengan menguji Transaksi untuk Pemilih memilih Kandidat 20 Transaksi secara serial di Satu Perangkat dan 10 Transaksi Secara Parallel di 2 Perangkat secara bersamaan diulang sebanyak 3 kali percobaan dengan BatchTimeout 2000s.

Percobaan untuk 20 transaksi untuk memilih Kandidat membutuhkan waktu rata-rata 121 detik atau 2 menit 1 detik yang berarti 6,05 detik per transaksi. Kecepatan transaksi ini sangat bergantung pada spesifikasi perangkat yang digunakan.

Percobaan Ke-	BatchTimeout (ms)	Parallel		Serial
		Perangkat 1	Perangkat 2	
		Delay (Detik)	Delay (Detik)	Delay (Detik)
1	2000	20.8	21.1	20.25
2	2000	20.6	20.9	20.25
3	2000	20.6	20.9	20.3
Rata- Rata	2000	20.67	20.97	20.27

Tabel 4-4 Hasil Pengujian Waktu proses setiap Transaksi memilih Kandidat 20 Transaksi secara Serial dan 10 Transaksi Secara Parallel dengan BatchTimeout 2000ms Hasil Pengujian Kecepatan 20 Transaksi Memilih Kandidat Secara Parallel



Gambar 4-44 Grafik jumlah waktu yang dibutuhkan dengan Percobaan 20 Transaksi untuk Memilih Kandidat Secara Parallel Perbandingan waktu yang transaksi Pilih Kandidat secara Serial dan Secara Parallel dengan *BatchTimeout* 250 ms dan 2000ms

Pengujian Pertama menguji Transaksi untuk Pemilih memilih Kandidat dengan *BatchTimeout* 250ms secara serial di satu Perangkat membutuhkan waktu rata rata tiap transaksi 8.9 detik sedangkan untuk 10 Transaksi Secara Parallel di 2 Perangkat secara bersamaan membutuhkan waktu rata rata tiap transaksi di Perangkat 1 detik 9.4 dan di Perangkat 2 detik 9.47

Pengujian Kedua menguji Transaksi untuk Pemilih memilih Kandidat dengan *BatchTimeout* 2000ms secara serial di satu Perangkat membutuhkan waktu rata rata tiap transaksi 20.27 detik sedangkan untuk 10 Transaksi Secara Parallel di 2 Perangkat secara bersamaan membutuhkan waktu rata rata tiap transaksi di Perangkat 1 detik 20.67 dan di Perangkat 2 detik 20.97.

Berdasarkan dua pengujian diatas, Adanya pengaruh *BatchTimeout* terhadap waktu yang dibutuhkan untuk membuat transaksi dimana semakin kecil batch timeout semakin cepat transaksi dibuat. Untuk transaksi secara serial dan parallel, meskipun rata-rata untuk membuat transaksi secara serial lebih cepat tetapi transaksi secara parallel dapat memproses lebih banyak kuantitas transaksinya.

~~Percobaan untuk 20 transaksi. Memilih untuk memilih Kandidat dimana transaksi yang berhasil hanya 10 transaksi di satu mesin dan gagal di mesin lainnya. Untuk 10 Transaksi membutuhkan waktu rata-rata 60 detik atau 1 Menit yang berarti 6 detik per transaksi. Pengujian ini membuktikan bahwa proses blockchain bersifat tidak parallel. Berikut ini adalah hasil stack trace dari error tersebut:~~

~~Proses Error terjadi ketika menjalankan file testing10PilihKandidat.sh pada Perangkat 1 dengan pemilih id 1-10 dan Perangkat 2 dengan pemilih id 11-20 secara bersamaan untuk menguji eksekusi transaksi Pilih Kandidat secara parallel. Error yang didapat yaitu MVCC_READ_CONFLICT dikarenakan ketidakcocokan versi, ini terjadi ketika melakukan submit transaksi secara bersamaan. Hyperledger Fabric menggunakan MVCC (multiversion concurrency control) yang berarti menggunakan dokumen atau catatan yang bervariasi dengan tujuan memastikan tidak ada pengeluaran ganda atau ketidakkonsistenan dalam data ada. Akibatnya menggunakan MVCC ketika Memilih melakukan transaksi secara bersamaan satu waktu maka satu pemilih akan berhasil dan satu lainnya akan gagal dengan error MVCC_READ_CONFLICT.~~

4.2.4. Pengujian Performa perintah Create dan Update Pemilih pada Database Blockchain dan Database MySQL

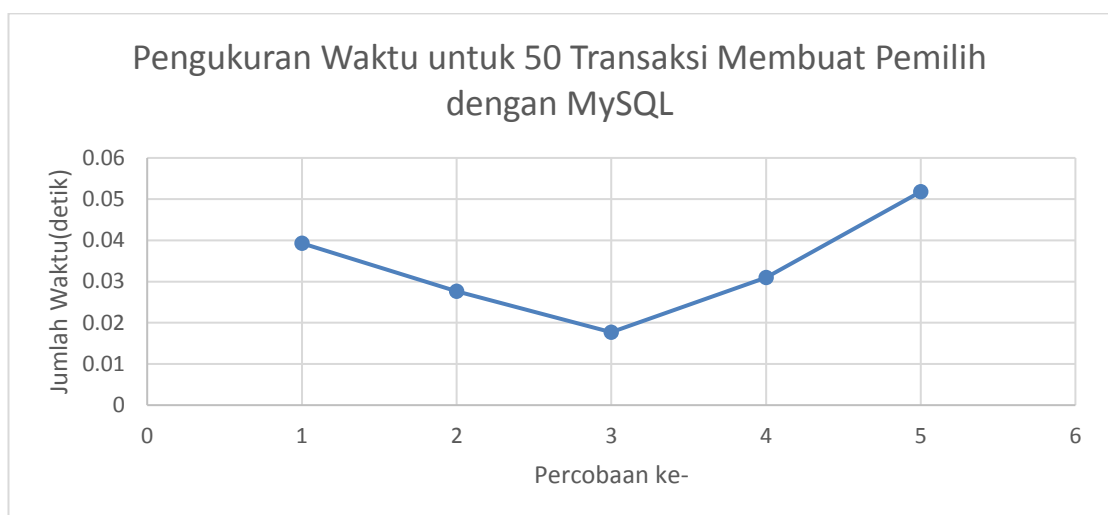
Pengujian ini untuk mengukur dan membandingkan performa saat membuat dan memperbaharui pemilih pada database blockchain dan database RDBMS biasa. Pada percobaan ini database MySQL dipilih karena merupakan database RDBMS yang paling umum digunakan. Pengujian Pertama dengan menguji 50 Transaksi untuk Membuat Pemilih dengan 5 kali percobaan. Pengujian Kedua dengan menguji 50 Transaksi untuk Memperbaharui Pemilih dengan 5 kali percobaan.

Pengujian Pertama untuk 50 Transaksi Membuat Pemilih dengan 5 Percobaan:

Percobaan	Jumlah Waktu 50 Transaksi (Detik)

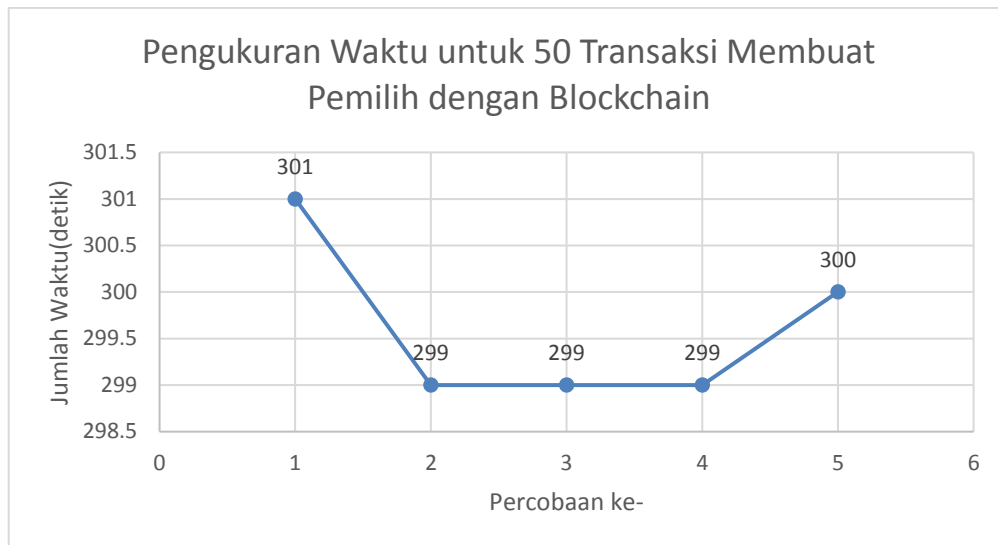
	Blockchain	Database Biasa (MySql)
1	301	0.0393
2	299	0.0276
3	299	0.0177
4	299	0.0310
5	300	0.0518
Rata-Rata	299.6	0.0335

Tabel 4-5 Hasil Pengujian Jumlah Waktu 50 Transaksi untuk Membuat Pemilih dengan Blockchain dan MySQL



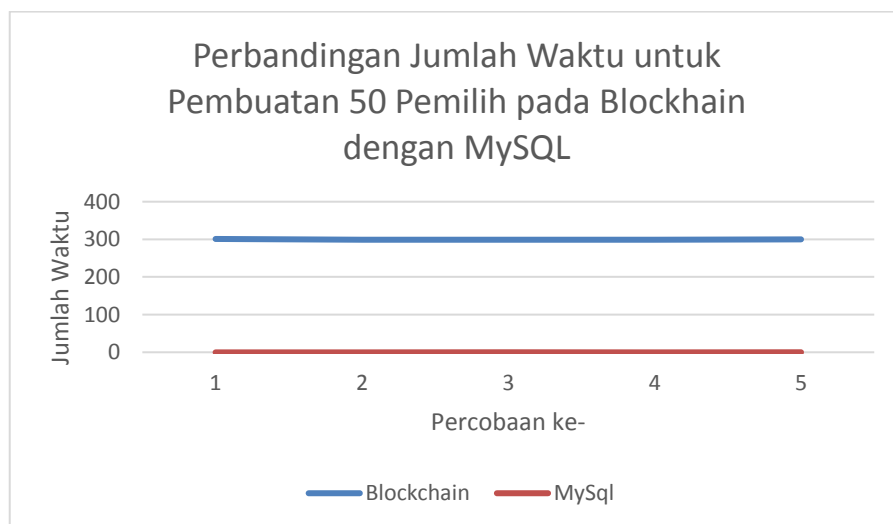
Gambar 4-45 ~~Grafik~~ Jumlah Waktu yang dibutuhkan dengan Percobaan 50 Transaksi untuk membuat Pemilih dengan MySQL

Pengujian pertama dengan 5 percobaan untuk 50 transaksi membuat Pemilih dengan MySQL membutuhkan waktu rata-rata 0.0335 detik.



Gambar 4-46 ~~Grafik~~ Jumlah Waktu yang dibutuhkan dengan Percobaan 50 Transaksi untuk membuat Pemilih dengan Blockchain

Pengujian pertama dengan 5 percobaan untuk 50 transaksi membuat Pemilih dengan Blockchain membutuhkan waktu rata-rata 299.6 detik.

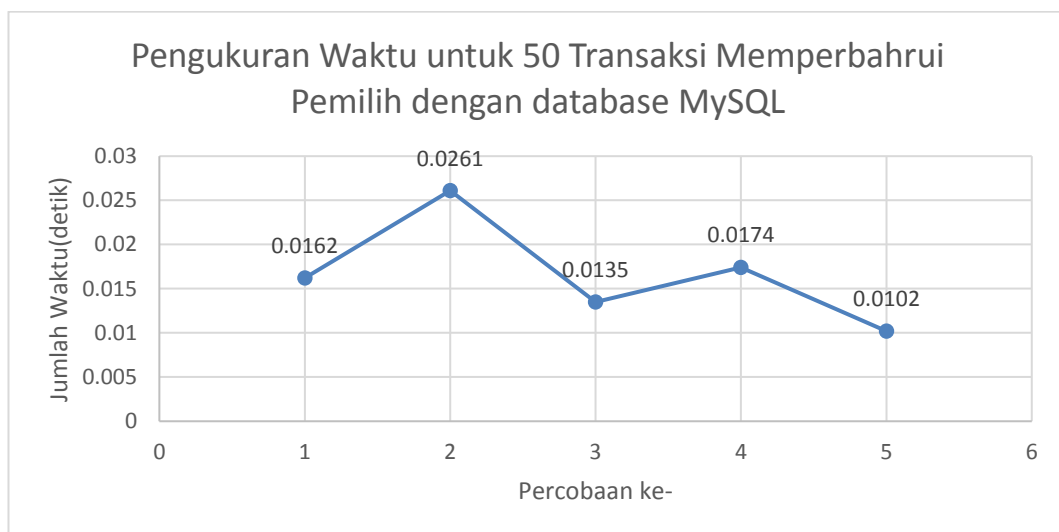


Gambar 4-47 ~~Grafik~~ Perbandingan Jumlah Waktu yang dibutuhkan dengan Percobaan 50 Transaksi untuk membuat Pemilih dengan Blockchain dan MySQL

Pengujian Kedua untuk 50 Transaksi Memperbaharui Pemilih pada variable telahMemilih menjadi true dan kandidat yang dipilih dengan 5 Percobaan.

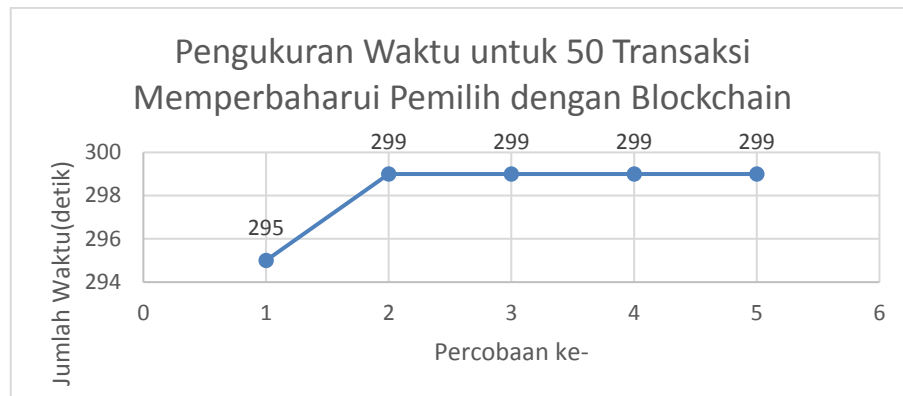
Percobaan	Jumlah Waktu 50 Transaksi (Detik)	
	Blockchain	Database Biasa (MySql)
1	295	0.0162
2	299	0.0261
3	299	0.0135
4	299	0.0174
5	299	0.0102
Rata-Rata	298.2	0.0167

Tabel 4-6 Hasil Pengujian Kecepatan 50 Transaksi Memperbaharui Pemilih dengan Blockchain dan MySQL



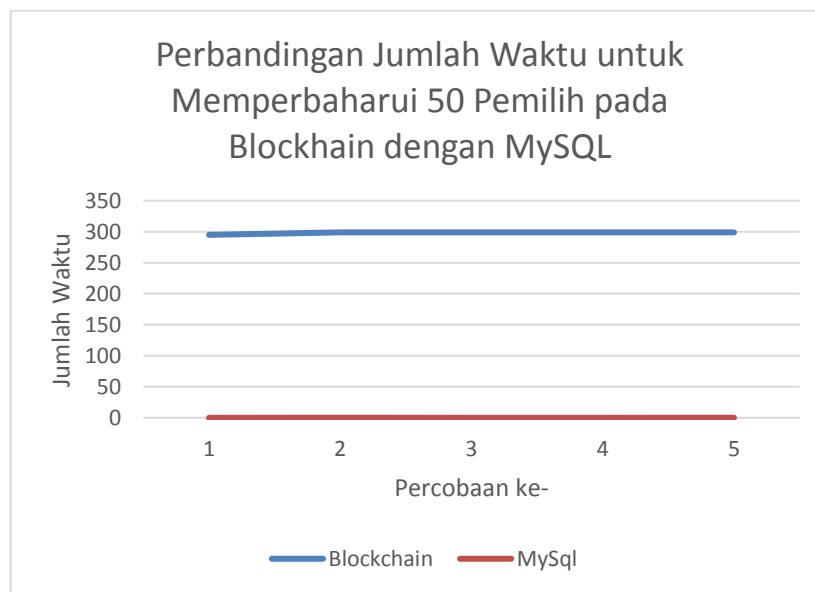
Gambar 4-48 ~~Grafik~~ Jumlah Waktu yang dibutuhkan dengan Percobaan 50 Transaksi untuk memperbaharui Pemilih dengan MySQL

Pengujian kedua dengan 5 percobaan untuk 50 transaksi memperbaharui Pemilih dengan MySQL membutuhkan waktu rata-rata 0.0167 detik.



Gambar 4-49 ~~Grafik~~ Perbandingan Jumlah Waktu yang dibutuhkan dengan Percobaan 50 Transaksi untuk memperbaharui Pemilih dengan Blockchain

Pengujian Kedua dengan 5 percobaan untuk 50 transaksi Memperbaharui Pemilih dengan Blockchain membutuhkan waktu rata-rata 298.2 detik.



Gambar 4-50 ~~Grafik~~ Perbandingan Jumlah Waktu yang dibutuhkan dengan Percobaan 50 Transaksi untuk memperbaharui Pemilih dengan Blockchain dan MySQL

Pengujian pertama membuktikan bahwa database biasa yaitu MySQL lebih cepat dibandingkan database blockchain, dimana untuk 50 transaksi membuat Pemilih, database biasa hanya memerlukan waktu 0.0335 detik sedangkan database blockchain membutuhkan waktu 299.6 detik. Pengujian kedua juga membuktikan bahwa database biasa lebih cepat dibandingkan database blockchain, dimana untuk

50 transaksi memperbaharui Pemilih, database biasa hanya memerlukan waktu 0.0167 detik sedangkan database blockchain membutuhkan waktu 298.2 detik

BAB 5

PENUTUP

5.1. Kesimpulan

Setelah dilakukan Pengujian dan Implementasi Teknologi *Blockchain* pada Sistem *Custom E-Voting*, dapat disimpulkan:

1. Implementasi *Blockchain* pada Sistem *Custom E-Voting* dapat diimplementasikan dengan platform Hyperledger mendukung *permissioned blockchain*. Implementasi Hyperledger Composer pada Hyperledger Fabric membutuhkan pendefinisian partisipan, asset, dan transaksi.

2. Adanya pengaruh *BatchTimeout* terhadap waktu yang dibutuhkan untuk membuat transaksi, dimana semakin kecil *BatchTimeout* semakin cepat transaksi dibuat. Selain itu, *Blockchain* memungkinkan dapat memproses Transaksi secara bersamaan dalam satu waktu.

2.3. Implementasi Teknologi *Blockchain* dapat menunjang Sistem *Custom E-voting* dengan menggunakan multi server. Meskipun memiliki tingkat keamanan yang lebih baik dibanding database biasa, tetapi untuk kecepatan dalam menjalankan perintah, Database biasa jauh lebih cepat dibanding database blockchain.

3.4. Pengujian secara kinerja dan fungsional didapatkan sesuai hasil tetapi membutuhkan spesifikasi komputer yang lebih tinggi agar mendapat hasil yang lebih baik dari sisi kinerja.

5.2. Saran

Setelah penulis menjalani penelitian tentang Implementasi Teknologi *Blockchain* pada Sistem *Custom E-Voting*, penulis memberikan saran untuk pengembangan lebih lanjut:

1. Pengujian menggunakan spesifikasi komputer untuk komputasi yang tinggi.

2. Penguji dapat menguji sistem *e-voting* dalam skala yang lebih besar yang meyerupai sistem yang ada di kondisi sekarang.

DAFTAR PUSTAKA

- [1] R. Rezwan, H. Ahmed, M. R. N. Biplob, S. M. Shuvo and M. A. Rahman, "Biometrically secured electronic voting machine," in *IEEE Region 10 Humanitarian Technology Conference (R10-HTC)*, Dhaka, 2017.
- [2] A. Jagtap, V. Kesarkar and A. Supekar, "Electronic Voting System using Biometrics, Raspberry Pi and TFT module," in *2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI)*, Tirunelveli, India, 2019.
- [3] "Pemilu 2019: Pemungutan suara Indonesia paling 'rumit' dan 'menakutkan' di dunia," BBC News Indonesia, 2019 April 15. [Online]. Available: <https://www.bbc.com/indonesia/indonesia-47879833>. [Accessed 17 April 2020].
- [4] D. Sandi, "Pilkada Serentak 2018, 54 Juta Lembar Kertas Suara Potensi Mubazir," Kompasiana, 9 January 2018. [Online]. Available: <https://www.kompasiana.com/diansandi/5a548401cf01b47187382cd2/pilkada-serentak-2018-54-juta-lembar-kertas-suara-potensi-mubazir>. [Accessed 17 April 2020].
- [5] "Soal Surat Suara Tercoblos di beberapa Daerah, KPU Sebut Sedang Didalami," Kompas.com, 17 April 2019. [Online]. Available: <https://nasional.kompas.com/read/2019/04/17/15324251/soal-surat-suara-tercoblos-di-beberapa-daerah-kpu-sebut-sedang-didalami>. [Accessed 17 April 2020].
- [6] A. Ardipandanto, "Permasalahan Penyelenggaraan Pemilu Serentak Tahun 2019," *Info Singkat*, vol. XI No. 11/I/P3DI/Juni/2019, no. Pusat Penelitian Badan Keahlian DPR RI, pp. 25-30, 2019.

- [7] P. S. Naidu, R. Kharat, R. Tekade, P. Mendhe and V. Magade, "E-voting system using visual cryptography & secure multi-party computation," in *2016 International Conference on Computing Communication Control and automation (ICCUBEA)*, Pune, 2016.
- [8] K. Garg, P. Saraswat, S. Bisht, S. K. Aggarwal, S. K. Kothuri and S. Gupta, "A Comparative Analysis on E-Voting System Using Blockchain," in *2019 4th International Conference on Internet of Things: Smart Innovation and Usages (IoT-SIU)*, Ghaziabad, India, 2019.
- [9] G. G. Dagher, P. B. Marella, M. Milojkovic and J. Mohler, "BroncoVote: Secure Voting System Using Ethereum's Blockchain," in *Proceedings of the 4th International Conference on Information Systems Security and Privacy*, Funchal, Madeira, Portugal, 2018.
- [10] D. F. H. S. Wibowo, "Perancangan dan Implementasi Teknologi Blockchain pada Sistem Pencatatan Hasil Rekapitulasi Pemilu Berdasarkan Formulir C1 Pindaian KPU," in *Master's Thesis*, Bandung, 2019.
- [11] "Pemilu 2019 Pertama Kali Pemilu Serentak di Indonesia," KPU Sukoharjo, 08 November 2018. [Online]. Available: <https://kpu-d-sukoharjokab.go.id/berita-335-pemilu-2019-pertama-kali-pemilu-serentak-di-indonesia.html>. [Accessed 18 April 2020].
- [12] D. Yaga, P. Mell, N. Roby and K. Scarfone, "Blockchain Technology Overview," National Institute of Standards and Technology, October 2018. [Online]. Available: <https://nvlpubs.nist.gov/nistpubs/ir/2018/NIST.IR.8202.pdf>. [Accessed 18 April 2020].
- [13] J. Yoo, Y. Jung, D. Shin, M. Bae and E. Jee, "Formal Modeling and Verification of a Federated Byzantine Agreement Algorithm for Blockchain Platforms," in *2019 IEEE International Workshop on Blockchain Oriented Software Engineering (IWBOSE)*, Hangzhou, China, 2019.

- [14] B. D. C. PUTRI, "Perancangan dan Implementasi Teknologi Blockchain Pada Sistem Pelayanan KJP Plus (Kartu Jakarta Pintar) Menggunakan Hyperledger Composer," in *Master Thesis*, Depok, 2019.
- [15] M. H. Sedky and E. M. R. Hamed, "A secure e-Government's e-voting system," in *2015 Science and Information Conference (SAI)*, London, 2015.
- [16] F. Þ. Hjálmarsson, G. K. Hreiðarsson, M. Hamdaqa and G. Hjálmtýsson, "Blockchain-Based E-Voting System," in *2018 IEEE 11th International Conference on Cloud Computing (CLOUD)*, San Francisco, CA, 2018.
- [17] N. Kshetri and J. Voas, "Blockchain-Enabled E-Voting," *IEEE Software*, vol. 35, no. 4, pp. 95-99, 2018.
- [18] T. Blummer, S. Bohan, M. Bowman, C. Cachin, N. Gaski, N. George, G. Graham, D. Hardman, R. Jagadeesan, T. Keith, R. Khasanshyn, M. Krishna, T. Kuhrt, A. L. Hors, J. Levi, S. Liberman and E. Mende, "An Introduction to Hyperledger White Paper," Hyperledger, 2018.
- [19] "What is Entity Relationship Diagram (ERD)?," Visual Paradigm, [Online]. Available: <https://www.visual-paradigm.com/guide/data-modeling/what-is-entity-relationship-diagram/>. [Accessed 24 April 2020].
- [20] M. N. Alanazi, "Basic Rules to Build Correct UML Diagrams," in *2009 International Conference on New Trends in Information and Service Science*, Beijing, 2009.
- [21] "UML Diagram," SmartDraw, [Online]. Available: <https://www.smartdraw.com/uml-diagram/>. [Accessed 24 April 2020].
- [22] "UML Use Case Diagram," Lucidchart, [Online]. Available: <https://www.lucidchart.com/pages/uml-use-case-diagram>. [Accessed 3 Mei 2020].

- [23] "Angular - Introduction to Angular concepts," Google , 18 February 2020. [Online]. Available: <https://angular.io/guide/architecture#introduction-to-angular-concepts>. [Accessed 20 July 2020].

LAMPIRAN

KUESIONER PENGUJIAN SISTEM

SISTEM CUSTOM EVOTING DENGAN TEKNOLOGI BLOCKCHAIN

Nama : Febrian Adhi Patria

Tanggal : 23 Juli 2020

PETUNJUK PENGISIAN:

Mohon kesediaan anda untuk mengisi kuisioner ini dengan baik. Berikan penilaian dengan memberikan tanda silang(X) pada salah satu kolom yang disediakan

No	Pernyataan	Bisa	Tidak Bisa
1	Dapat melakukan Login sebagai Admin	X	
2	Dapat melakukan Login sebagai User	X	
3.	Admin dapat membuat Pemilihan	X	
4	Admin dapat membuat Kandidat	X	
5	Admin dapat membuat Pemilih	X	
6	Admin dapat mengakhiri Pemilihan	X	
7.	Pemilih dapat memilih Kandidat	X	
8.	Pemilih dapat melihat Hasil Pemilihan	X	

Berikan penilaian dengan memberikan tanda silang(X) pada salah satu kolom yang disediakan(1 = sangat buruk; 2 = buruk; 3 = cukup; 4= baik; 5= sangat baik)

No	Pernyataan	1	2	3	4	5
1	Kemudahan penggunaan sistem ini sebagai upaya menggantikan sistem tradisional					X

Kritik & Saran :

.....

Terima Kasih atas kesediaan dan waktu yang diberikan untuk turut serta dalam pengujian Sistem ini.

KUESIONER PENGUJIAN SISTEM

SISTEM CUSTOM EVOTING DENGAN TEKNOLOGI BLOCKCHAIN

Nama : Farhana Syuaib

Tanggal : 23/07/2020

PETUNJUK PENGISIAN:

Mohon kesediaan anda untuk mengisi kuisisioner ini dengan baik. Berikan penilaian dengan memberikan tanda silang(X) pada salah satu kolom yang disediakan

No	Pernyataan	Bisa	Tidak Bisa
1	Dapat melakukan Login sebagai Admin	X	
2	Dapat melakukan Login sebagai User	X	
3.	Admin dapat membuat Pemilihan	X	
4	Admin dapat membuat Kandidat	X	
5	Admin dapat membuat Pemilih	X	
6	Admin dapat mengakhiri Pemilihan	X	
7.	Pemilih dapat memilih Kandidat	X	
8.	Pemilih dapat melihat Hasil Pemilihan	X	

Berikan penilaian dengan memberikan tanda silang(X) pada salah satu kolom yang disediakan(1 = sangat buruk; 2 = buruk; 3 = cukup; 4= baik; 5= sangat baik)

No	Pernyataan	1	2	3	4	5
1	Kemudahan penggunaan sistem ini sebagai upaya menggantikan sistem tradisional					X

Kritik & Saran : *

.....

Terima Kasih atas kesediaan dan waktu yang diberikan untuk turut serta dalam pengujian Sistem ini.

KUESIONER PENGUJIAN SISTEM

SISTEM CUSTOM EVOTING DENGAN TEKNOLOGI BLOCKCHAIN

Nama : Muhammad Ilham Akbar

Tanggal : 23/07/2020

PETUNJUK PENGISIAN:

Mohon kesediaan anda untuk mengisi kuisisioner ini dengan baik. Berikan penilaian dengan memberikan tanda silang(X) pada salah satu kolom yang disediakan

No	Pernyataan	Bisa	Tidak Bisa
1	Dapat melakukan Login sebagai Admin	X	
2	Dapat melakukan Login sebagai User	X	
3.	Admin dapat membuat Pemilihan	X	
4	Admin dapat membuat Kandidat	X	
5	Admin dapat membuat Pemilih	X	
6	Admin dapat mengakhiri Pemilihan	X	
7.	Pemilih dapat memilih Kandidat	X	
8.	Pemilih dapat melihat Hasil Pemilihan	X	

Berikan penilaian dengan memberikan tanda silang(X) pada salah satu kolom yang disediakan(1 = sangat buruk; 2 = buruk; 3 = cukup; 4= baik; 5= sangat baik)

No	Pernyataan	1	2	3	4	5
1	Kemudahan penggunaan sistem ini sebagai upaya menggantikan sistem tradisional					X

Kritik & Saran : Secara umum sudah sangat baik dan mudah digunakan (user friendly), karenesemua fungsinya telah diimplementasikan dengan baik. Namun, masih tedapat beberapa bagian yang perlu diperbaiki, seperti tampilan situs web (front-end) dan pemilihan icon yang lebih tepat.

Terima Kasih atas kesediaan dan waktu yang diberikan untuk turut serta dalam pengujian Sistem ini.

KUESIONER PENGUJIAN SISTEM

SISTEM CUSTOM EVOTING DENGAN TEKNOLOGI BLOCKCHAIN

Nama : Ahmad Naufal Hilmy

Tanggal : 23 Juli 2020

PETUNJUK PENGISIAN:

Mohon kesediaan anda untuk mengisi kuisioner ini dengan baik. Berikan penilaian dengan memberikan tanda silang(X) pada salah satu kolom yang disediakan

No	Pernyataan	Bisa	Tidak Bisa
1	Dapat melakukan Login sebagai Admin	X	
2	Dapat melakukan Login sebagai User	X	
3.	Admin dapat membuat Pemilihan	X	
4	Admin dapat membuat Kandidat	X	
5	Admin dapat membuat Pemilih	X	
6	Admin dapat mengakhiri Pemilihan	X	
7.	Pemilih dapat memilih Kandidat	X	
8.	Pemilih dapat melihat Hasil Pemilihan	X	

Berikan penilaian dengan memberikan tanda silang(X) pada salah satu kolom yang disediakan(1 = sangat buruk; 2 = buruk; 3 = cukup; 4= baik; 5= sangat baik)

No	Pernyataan	1	2	3	4	5
1	Kemudahan penggunaan sistem ini sebagai upaya menggantikan sistem tradisional					X

Kritik & Saran :

.....

Terima Kasih atas kesediaan dan waktu yang diberikan untuk turut serta dalam pengujian Sistem ini.

KUESIONER PENGUJIAN SISTEM

SISTEM CUSTOM EVOTING DENGAN TEKNOLOGI BLOCKCHAIN

Nama : Armelia Ramandha

Tanggal : 23 Juli 2020

PETUNJUK PENGISIAN:

Mohon kesediaan anda untuk mengisi kuisioner ini dengan baik. Berikan penilaian dengan memberikan tanda silang(X) pada salah satu kolom yang disediakan

No	Pernyataan	Bisa	Tidak Bisa
1	Dapat melakukan Login sebagai Admin	V	
2	Dapat melakukan Login sebagai User	V	
3.	Admin dapat membuat Pemilihan	V	
4	Admin dapat membuat Kandidat	V	
5	Admin dapat membuat Pemilih	V	
6	Admin dapat mengakhiri Pemilihan	V	
7.	Pemilih dapat memilih Kandidat	V	
8.	Pemilih dapat melihat Hasil Pemilihan	V	

Berikan penilaian dengan memberikan tanda silang(X) pada salah satu kolom yang disediakan(1 = sangat buruk; 2 = buruk; 3 = cukup; 4= baik; 5= sangat baik)

No	Pernyataan	1	2	3	4	5
1	Kemudahan penggunaan sistem ini sebagai upaya menggantikan sistem tradisional					V

Kritik & Saran :

Mungkin untuk username dan pass pemilih agar lebih aman bisa dibuat otomatis setelah admin menginput nama panjang atau berdasarkan no.KTP

Terima Kasih atas kesediaan dan waktu yang diberikan untuk turut serta dalam pengujian Sistem ini.

KUESIONER PENGUJIAN SISTEM

SISTEM CUSTOM EVOTING DENGAN TEKNOLOGI BLOCKCHAIN

Nama : Muhammad Fairus Adzka

Tanggal : 23/07/2020

PETUNJUK PENGISIAN:

Mohon kesediaan anda untuk mengisi kuisisioner ini dengan baik. Berikan penilaian dengan memberikan tanda silang(X) pada salah satu kolom yang disediakan

No	Pernyataan	Bisa	Tidak Bisa
1	Dapat melakukan Login sebagai Admin	X	
2	Dapat melakukan Login sebagai User	X	
3.	Admin dapat membuat Pemilihan	X	
4	Admin dapat membuat Kandidat	X	
5	Admin dapat membuat Pemilih	X	
6	Admin dapat mengakhiri Pemilihan	X	
7.	Pemilih dapat memilih Kandidat	X	
8.	Pemilih dapat melihat Hasil Pemilihan	X	

Berikan penilaian dengan memberikan tanda silang(X) pada salah satu kolom yang disediakan(1 = sangat buruk; 2 = buruk; 3 = cukup; 4= baik; 5= sangat baik)

No	Pernyataan	1	2	3	4	5
1	Kemudahan penggunaan sistem ini sebagai upaya menggantikan sistem tradisional					X

Kritik & Saran :

Terima Kasih atas kesediaan dan waktu yang diberikan untuk turut serta dalam pengujian Sistem ini.