

**IMPLEMENTASI E-VOTING DENGAN KOMBINASI SKEMA
BLIND SIGNATURE DAN ELLIPTIC CURVE CRYPTOGRAPHY**

SKRIPSI

DIFANIE MAYANANDA

201401045



**PROGRAM STUDI S-1 ILMU KOMPUTER
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS SUMATERA UTARA**

MEDAN

2024

**IMPLEMENTASI *E-VOTING* DENGAN KOMBINASI SKEMA *BLIND*
SIGNATURE DAN *ELLIPTIC CURVE CRYPTOGRAPHY***

SKRIPSI

**Diajukan untuk melengkapi tugas dan memenuhi syarat memperoleh ijazah
sarjana ilmu komputer**

DIFANIE MAYANANDA

201401045



**PROGRAM STUDI S1 ILMU KOMPUTER
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS SUMATERA UTARA**

MEDAN

2024

UNIVERSITAS SUMATERA UTARA

PERSETUJUAN

Judul : IMPLEMENTASI *E-VOTING* DENGAN
KOMBINASI SKEMA *BLIND SIGNATURE*
DAN *ELLIPTIC CURVE CRYPTOGRAPHY*

Kategori : SKRIPSI

Nama : DIFANIE MAYANANDA

Nomor Induk Mahasiswa : 201401045

Program Studi : SARJANA (S-1) ILMU KOMPUTER

Fakultas : ILMU KOMPUTER DAN TEKNOLOGI
INFORMASI UNIVERSITAS SUMATERA
UTARA

Medan, 26 Januari 2024

Komisi Pembimbing

Dosen Pembimbing II



Prof. Dr. Syahril Efendi S.Si., M.IT.
NIP 196711101996021001

Dosen Pembimbing I



Amer Sharif S.Si., M.Kom.
NIP 196910212021011001

Diketahui/disetujui oleh Program Studi
SI Ilmu Komputer

Ketua:



Dr. Amalin S.P., M.T.
NIP-197812212014042001

PERNYATAAN
IMPLEMENTASI *E-VOTING* DENGAN KOMBINASI SKEMA *BLIND*
SIGNATURE* DAN *ELLIPTIC CURVE CRYPTOGRAPHY

SKRIPSI

Saya mengakui bahwa skripsi berikut adalah hasil dari karya saya sendiri, kecuali ada beberapa kutipan dan ringkasan yang masing masing sudah saya sebut sumbernya.

Medan, 2 Desember 2023



Difanie Mayananda
NIM. 201401045

UCAPAN TERIMA KASIH

Penulis menyatakan rasa terima kasih kepada Allah SWT. Yang Maha Pengasih dan Maha Penyayang, karena hadirat dan limpahan rahmat-Nya penulis diberikan kemampuan untuk menjalani perkuliahan hingga sampai penyusunan laporan skripsi. Salawat dan salam secara beriringan kepada Nabi Muhammad SAW penulis junjungkan karena beliau telah membawa petunjuk kepada umat manusia. Penulis sadar bahwa pembuatan skripsi ini berisi penuh dukungan orang-orang di sekitar. Maka, pada kesempatan ini, izinkan penulis untuk menyampaikan ucapan terima kasih yang sangat besar kepada:

1. Bapak Dr. Muryanto Amin, S.Sos., M.Si, selaku Rektor Universitas Sumatera Utara.
2. Ibu Dr. Maya Silvi Lydia B.Sc., M.Sc, selaku Dekan Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Sumatera Utara.
3. Ibu Dr. Amalia, S.T., M.T, selaku Ketua Program Studi S1 Ilmu Komputer Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Sumatera Utara.
4. Bapak Amer Sharif S.Si, M.Kom, selaku Dosen Pembimbing I yang telah memberikan banyak bimbingan, arahan, serta meluangkan waktu untuk penulis sampai pada tahap penyusunan laporan skripsi.
5. Bapak Prof. Dr. Syahril Efendi S.Si., M.IT, selaku Dosen Pembimbing II yang telah membimbing penulis sampai pada tahap penyusunan skripsi ini.
6. Bapak Dr. Mohammad Andri Budiman, S.T., M.Comp.Sc., M.E.M. selaku Dosen Pembimbing 1 yang telah memberikan bimbingan, saran, dan ilmu yang membangun kepada penulis.
7. Ibu Dian Rachmawati, S.Si., M.Kom. selaku Dosen Pembimbing 2 yang telah memberikan saran dan kritikan kepada penulis.
8. Seluruh Bapak Ibu Dosen Program Studi S-1 Ilmu Komputer khususnya Bapak Ibu yang telah memberikan sebagian waktunya untuk mengajarkan ilmunya kepada penulis.
9. Kepada kedua Orang Tua yang teramat dicintai, Ibu Sumiati dan Bapak Muhammad Fakhruddin, yang telah memberikan doa dan dukungan moral serta

materi secara tulus. Dukungan tersebut menjadi faktor terbesar bagi penulis dalam menyelesaikan penyusunan skripsi ini.

10. Kakak Dea Syavira dan adik Difqy Aldian yang telah memberika doa dan motivasi yang berharga untuk penulis.
11. Semua anggota keluarga besar penulis telah memberikan motivasi yang tulus dan berarti bagi penulis.
12. Sahabat-sahabat semasa SMP dan SMA penulis yang telah menjadi pendukung serta penyemangat yang tulus untuk penulis.
13. Sahabat-sahabat SISTER yang telah memberikan pengalaman dan penghiburan bagi penulis.
14. Sahabat-sahabat penulis khususnya ciwi-ciwi Kom B yang telah memberikan dorongan yang memotivasi dan berbagi pengalaman yang amat berharga bagi penulis sepanjang perjalanan perkuliahan.
15. IMILKOM yang merupakan tempat berkembangnya penulis selama masa perkuliahan.
16. Stambuk 2020 khususnya Kom B yang memberikan penulis pengalaman belajar yang berharga.

Serta semua pihak yang telah memberikan bantuan kepada penulis, meskipun tidak dapat disebutkan satu per satu, semoga diberkahi oleh Allah SWT. atas segala bantuan yang diberikan.

Medan, 2 Desember 2023

Penulis,



Difanie Mayananda

ABSTRAK

Pemungutan suara atau *voting* merupakan salah satu sistem yang digunakan untuk mengambil keputusan. Namun, ketidakhadiran pemilih merupakan hal yang menjadi hambatan pada kegiatan pemungutan suara. *E-voting* merupakan solusi untuk masalah ketidakhadiran pada pemungutan suara. *E-voting* harus dapat menjaga kerahasiaan isi surat suara, menjamin hak pemilih, dan kerahasiaan identitas dari pemilih. Untuk menjaga kerahasiaan identitas dan menjamin hak pemilih untuk memberikan suara pada penelitian ini, digunakan Skema *Blind Signature* untuk menjaga validitas surat suara walau pun surat suara dikirim secara anonim. Skema ini dikombinasikan dengan *Elliptic Curve Cryptography* (ECC) yang merupakan algoritma asimetris yang memanfaatkan kurva elips sebagai algoritma untuk melakukan enkripsi dan dekripsi yang dapat merahasiakan isi surat suara. Surat suara diolah menggunakan fungsi *hash* yaitu SHA-256 sebelum dilakukan enkripsi. Sistem e-voting pada penelitian ini diimplementasikan dalam bentuk web dengan kunci privat sepanjang 256-bit.

Kata Kunci : *E-voting, Elliptic Curve Cryptography, Blind Signature, Hash, Kunci Privat*

ABSTRACT

Voting is one of the systems used to make decisions. However, Voter absenteeism is an obstacle to voting activities. E-voting is a solution to the problem of absenteeism at the vote. E-voting must be able to maintain the confidentiality of the contents of the ballot, guarantee the rights of *Voters*, and the confidentiality of the identity of *Voters*. To maintain confidentiality of identity and guarantee the right of *Voters* to vote in this study, a Blind Signature Scheme was used to maintain the validity of ballots even if ballots were sent anonymously. This scheme is combined with Elliptic Curve Cryptography (ECC) which is an asymmetric algorithm that utilizes elliptic curves as an algorithm to encrypt and decrypt the contents of ballots. Ballots are processed using the hash function SHA-256 before encryption. The e-voting system in this study was implemented in a web with a 256-bit private key.

Keywords : *E-voting, Elliptic Curve Cryptography, Blind Signature, Hash, Private Key*

DAFTAR ISI

PERSETUJUAN	Error! Bookmark not defined.
PERNYATAAN.....	iii
UCAPAN TERIMA KASIH	iv
DAFTAR ISI.....	viii
DAFTAR TABEL	xi
DAFTAR GAMBAR.....	xii
BAB I.....	1
PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah.....	2
1.4 Tujuan Penelitian.....	2
1.5 Manfaat Penelitian.....	2
1.6 Penelitian Relevan.....	3
1.7 Metodologi Penelitian	4
1.8 Sistematika Penulisan.....	5
BAB II	7
LANDASAN TEORI	7
2.1 E-Voting	7
2.2 Kriptografi	7
2.3 Blind Signature.....	7
2.4 Elliptic Curve Cryptography	9
2.4.1 Enkripsi ECC	10

2.4.2 Dekripsi ECC.....	11
2.4.3 Perubahan Plaintext Menjadi Titik	11
2.4.4 Contoh Kasus.....	13
BAB III.....	15
ANALISIS DAN PERANCANGAN.....	15
3.1.1 Analisis Masalah	15
3.1.2 Analisis Persyaratan	16
3.1.2.1 Analisis Persyaratan Fungsional.....	16
3.1.2.2 Analisis Persyaratan Non-fungsional	16
3.2 Perancangan Sistem.....	17
3.2.1 Diagram Umum	17
3.2.2 <i>Use case</i> Diagram	19
3.2.3 <i>Activity Diagram</i>	19
3.2.4 <i>Sequence Diagram</i>	24
3.2.5 Diagram Alir (<i>Flowchart</i>).....	25
3.2.6 Perancangan Database	32
3.2.7 Perancangan <i>User interface</i>	38
BAB IV	46
IMPLEMENTASI DAN PENGUJIAN SISTEM.....	46
4.1 Implementasi	46
4.1.1 Halaman Registrasi.....	46
4.1.2 Halaman Login	48
4.1.3 Halaman Beranda.....	48
4.1.4 Halaman Menambah <i>Event</i> Pemilihan.....	50
4.1.5 Halaman <i>Event Trustee</i>	51

4.1.6 Halaman <i>Event Voter</i>	52
4.1.7 Halaman <i>Event Signer</i>	53
4.2 Pengujian Sistem Satu <i>Voter</i>	54
4.2.1 Pengujian Membuat Suara	54
4.2.2 Pengujian <i>Signing</i>	56
4.2.3 Pengujian <i>Unblinding</i>	57
4.2.4 Pengujian <i>Verification</i>	58
4.3 Pengujian Sistem Beberapa <i>Voter</i>	60
4.3.1 Pengujian Memberikan Suara	60
4.3.2 Pengujian <i>Signing</i>	61
4.3.3 Pengujian Verifikasi	63
BAB V	65
KESIMPULAN DAN SARAN	65
5.1 Kesimpulan	65
5.2 Saran	65
DAFTAR PUSTAKA	66
LAMPIRAN	68

DAFTAR TABEL

Tabel 2. 1 Tabel Enkoding.....	11
Tabel 3. 1 Tabel Simbol Perhitungan.....	26
Tabel 3. 2 Tabel <i>User</i>	33
Tabel 3. 3 Tabel <i>Voter</i>	33
Tabel 3. 4 Tabel <i>Trustee</i>	33
Tabel 3. 5 Tabel <i>Signer</i>	33
Tabel 3. 6 Tabel Admin	34
Tabel 3. 7 Tabel <i>Key</i>	34
Tabel 3. 8 Tabel <i>Event</i>	34
Tabel 3. 9 Tabel Suara	35
Tabel 3. 10 Tabel Pilihan	36
Tabel 3. 11 Tabel <i>Signing Model</i>	36
Tabel 3. 12 Tabel <i>Unblinding</i>	37

DAFTAR GAMBAR

Gambar 2. 1 Proses Enkripsi dan Dekripsi.....	7
Gambar 2. 2 <i>Blind Signature</i> pada <i>E-voting</i>	9
Gambar 3. 1 <i>Fishbone Diagram</i>	16
Gambar 3. 2 Diagram Umum	18
Gambar 3. 3 <i>Use case Diagram</i>	19
Gambar 3. 4 <i>Diagram Activity</i> Register	20
Gambar 3. 5 <i>Diagram Activity</i> Membuat <i>Event</i>	21
Gambar 3. 6 <i>Diagram Activity</i> Membuat Suara	21
Gambar 3. 7 <i>Diagram Activity</i> Signing	22
Gambar 3. 8 <i>Diagram Activity</i> Unblinding	23
Gambar 3. 9 <i>Diagram Activity</i> Verification	24
Gambar 3. 10 <i>Sequence Diagram</i> <i>Trustee</i>	24
Gambar 3. 11 <i>Sequence Diagram</i> <i>Voter</i>	25
Gambar 3. 12 <i>Sequence Diagram</i> <i>Signer</i>	25
Gambar 3. 13 Flowchart Pembangkitan Kunci.....	27
Gambar 3. 14 Flowchart Blinding.....	28
Gambar 3. 15 Flowchart Signing	29
Gambar 3. 16 Flowchart Unblinding	30
Gambar 3. 17 Flowchart Verification	31
Gambar 3. 18 ERD	32
Gambar 3. 19 Relasi Antar Tabel	38
Gambar 3. 20 <i>User Interface</i> Halaman Register	39
Gambar 3. 21 <i>User Interface</i> Halaman <i>Login User</i>	40
Gambar 3. 22 <i>UserInterface</i> Beranda	40
Gambar 3. 23 <i>User Interface</i> Menambah <i>Event</i>	41
Gambar 3. 24 <i>User Interface</i> <i>Event Voter</i>	42
Gambar 3. 25 <i>User Interface</i> Halaman Utama <i>Event Trustee</i>	43
Gambar 3. 26 <i>User Interface</i> Pengaturan <i>Event</i>	44

Gambar 3. 27	<i>User interface Event Signer</i>	44
Gambar 3. 28	<i>User interface Halaman Beranda Admin</i>	45
Gambar 4. 1	<i>Halaman Registrasi Voter</i>	47
Gambar 4. 2	<i>Halaman Registrasi Signer</i>	47
Gambar 4. 3	<i>Halaman Register Trustee</i>	47
Gambar 4. 4	<i>Halaman Login</i>	48
Gambar 4. 5	<i>Halaman Beranda Admin</i>	49
Gambar 4. 6	<i>Halaman Beranda Trustee</i>	49
Gambar 4. 7	<i>Halaman Beranda Signer</i>	50
Gambar 4. 8	<i>Halaman Beranda Voter</i>	50
Gambar 4. 9	<i>Halaman Menambah Event Pemilihan</i>	51
Gambar 4. 10	<i>Halaman Utama Event Trustee</i>	52
Gambar 4. 11	<i>Halaman Pengaturan Event</i>	52
Gambar 4. 12	<i>Halaman Event Voter</i>	53
Gambar 4. 13	<i>Halaman Event Signer</i>	53
Gambar 4. 14	<i>Memberikan Suara</i>	54
Gambar 4. 15	<i>Hasil Blinding</i>	55
Gambar 4. 16	<i>Detail Blinding</i>	55
Gambar 4. 17	<i>Pengujian Signing</i>	56
Gambar 4. 18	<i>Detail Signing</i>	57
Gambar 4. 19	<i>Voter Menerima Surat yang di-signing</i>	57
Gambar 4. 20	<i>Hasil Unblinding</i>	58
Gambar 4. 21	<i>Sebelum Verifikasi</i>	59
Gambar 4. 22	<i>Hasil Setelah Verifikasi</i>	59
Gambar 4. 23	<i>Detail Verification</i>	60
Gambar 4. 24	<i>Beberapa Voter</i>	61
Gambar 4. 25	<i>Beberapa Voter Memberikan Suara</i>	61
Gambar 4. 26	<i>Pengujian Signing Beberapa Voter</i>	62
Gambar 4. 27	<i>Unblinding Voter</i>	62
Gambar 4. 28	<i>Sebelum Verifikasi Beberapa Voter</i>	63
Gambar 4. 29	<i>Setelah Verifikasi Beberapa Voter</i>	64

BAB I

PENDAHULUAN

1.1 Latar Belakang

Pemungutan suara atau *voting* adalah hal yang umum dilakukan untuk mengambil keputusan atau pun memilih seseorang untuk diberikan amanat. Konsep pemungutan suara yaitu dengan memilih hasil berdasarkan suara terbanyak dari para pemilih. Di Indonesia sendiri pemungutan suara terbesar yang dilakukan salah satunya adalah pemilihan Presiden Indonesia yang diselenggarakan setiap 5 tahun sekali. *Voting* dilakukan untuk menghemat waktu dalam mengambil sebuah keputusan. Akan tetapi pemilih yang tidak dapat berhadir karena alasan tertentu pada sebuah *voting* tidak dapat mengikuti pemungutan suara sehingga sering kali suara orang tersebut tidak dihitung atau bahkan pemungutan suara tidak dapat dilakukan.

Agar dapat menyesuaikan dengan pemilih yang tidak dapat berhadir, maka *e-voting* atau pemungutan suara dengan bantuan teknologi informasi dilakukan. *E-voting* haruslah bersifat seperti *voting* langsung. Untuk bersifat seperti *voting* langsung ada hal-hal yang diperhatikan yaitu memastikan kerahasiaan isi *voting* dan memastikan kerahasiaan identitas pengirim dengan catatan surat suara tersebut sudah divalidasi oleh pihak yang berwenang.

Skema *Blind Signature* atau tanda tangan buta adalah sebuah skema di mana seorang pemilih meminta kepada pihak yang berwenang untuk memvalidasi surat suara yang dikirim tanpa isi surat suara diketahui oleh pihak yang berwenang tersebut. Setelah surat suara divalidasi, pengirim akan mengirimkan surat tersebut secara anonim kepada pihak penghitung suara. Lalu pihak penghitung suara dapat memvalidasi surat suara tersebut melalui pengenalan tanda tangan digital yang dilakukan oleh pihak berwenang. Dengan skema ini pemilih dapat menjaga identitasnya akan tetapi surat suara juga dapat terjaga validitas dan kerahasiannya.

ECC (*Elliptic Curve Cryptography*) merupakan algoritma asimetrik yang menggunakan kunci publik dan privat dalam prosesnya. Sebagian besar algoritma yang menggunakan kunci publik memiliki penyimpanan yang cukup besar, ECC merupakan alternatif yang dapat digunakan karena memiliki ukuran kunci yang lebih kecil tetapi memiliki tingkat keamanan yang sama. Algoritma ini memanfaatkan perhitungan

matematis berdasarkan persamaan kurva elips. Algoritma ini mengalami beberapa kali proses perhitungan sehingga menghasilkan sejumlah titik pada kurva. Keunggulan dari algoritma ini adalah sulitnya menebak berapa kali proses perhitungan dan kurva apa yang digunakan pada prosesnya. Penerapan *Blind Signature* akan dikombinasikan dengan algoritma ECC untuk memaksimalkan kegiatan *e-voting* yang dilakukan.

1.2 Rumusan Masalah

Berdasarkan latar belakang, kegiatan *e-voting* memiliki permasalahan yang dihadapi yaitu bagaimana melakukan validasi terhadap surat suara tanpa pihak yang memvalidasi mengetahui isi surat suara tersebut dan menjaga kerahasiaan identitas dari pemilih. Skema *Blind Signature* yang dipadukan dengan ECC diharapkan dapat mengatasi permasalahan tersebut

1.3 Batasan Masalah

Penelitian ini memiliki beberapa batasan antara lain:

1. Penelitian yang digunakan hanya pada implementasi algoritma *Blind Signature* dan Elliptic Curve Cryptography
2. Penelitian tidak membahas kekurangan dari algoritma yang digunakan
3. Algoritma ini diimplementasikan dengan basis web dengan menggunakan bahasa pemrograman Python dan menggunakan Django sebagai *framework*-nya.

1.4 Tujuan Penelitian

Penelitian ini bertujuan untuk mengembangkan algoritma yang digunakan untuk membangun sebuah sistem *e-voting* yang dapat melindungi kerahasiaan suara dan identitas dari pemilih. Penelitian ini juga bertujuan untuk menguji apakah skema dan algoritma yang diteliti dapat dikombinasikan untuk membangun sebuah sistem *e-voting*.

1.5 Manfaat Penelitian

Harapannya dari penelitian ini, sistem yang dibangun mampu dimanfaatkan masyarakat untuk melakukan *e-voting* tanpa khawatir akan kerahasiaan identitas dan isi pesan suara yang dapat meningkatkan rasa aman pemilih. Lalu dengan dibangunnya sistem ini masyarakat diharap bisa melakukan *voting* dari tempat yang jauh sekalipun.

1.6 Penelitian Relevan

Beberapa studi sebelumnya terkait dengan penelitian yang dilakukan adalah sebagai berikut:

1. Berdasarkan penelitian (James, dkk., 2018) yang berjudul “Pairing Free Identity-Based *Blind Signature* Scheme with Message Recovery”, mereka melakukan kombinasi antara Skema *Blind Signature* dengan pemulihan pesan yang berbasis ID. Skema yang digunakan dinilai aman dan sulit dilawan oleh serangan identitas dengan pembuktian menggunakan *Elliptic Curve Discrete Logarithm Problem* (ECDLP). Teori *Blindness Property* dari Skema PF-IDBS-MR ini dapat diaplikasikan secara efektif pada model *e-voting* dan *e-payment* sistem.
2. Berdasarkan penelitian (Harn, dkk., 2019) yang berjudul “Multiple *Blind Signature* for *e-voting* and e-Cash”, peneliti membangun sistem *e-voting* dan e-cash menggunakan Multiple *Blind Signature* (MBS) yang mana merupakan penggabungan dari *Blind Signature* dan *Dual Signature*. *E-voting* menggunakan MBS dapat membuat pemilih untuk mengosongkan surat suara pada tahap registrasi karena pemilih harus melakukan perhitungan dan interaksi pada masa registrasi. Lalu pada hari pemilihan, pemilih dapat menuliskan pilihannya tanpa membutuhkan perhitungan lagi. Hal ini dapat mengurangi *network traffic* dihari pemilihan.
3. Berdasarkan penelitian (Maulid, 2018) yang berjudul “The Implementation of *Blind Signature* in Digital Cash”, peneliti melakukan dua implementasi yaitu menggunakan *Online - RSA Blind Signatures Digital Cash Scheme* dan *Offline - Brands Blind Signatures Digital Cash Scheme*. Pada skema *online* peneliti menggunakan *RSA Blind Signature* di mana tipe ini dinilai sederhana, praktis, dan tahan terhadap serangan. Akan tetapi sulit

membuktikan ketahanan algoritma ini terhadap serangan karena kunci RSA berdasar pada masalah faktorisasi.

4. Berdasarkan penelitian (Nugroho & Painem, 2022) yang berjudul “IMPLEMENTASI ALGORITMA ELLIPTIC CURVE CRYPTOGRAPHY (ECC) UNTUK PENGAMANAN FILE BERBASIS WEB”, peneliti menggunakan ECC untuk mengamankan file. Peneliti menggunakan persamaan $y^2 = x^2 + ax + b \pmod{p}$ pada kuva elips. Hasil implementasi dapat mengamankan data berekstensi xls, doc, jpg, pdf, xlsx, pptx, dan docx.

1.7 Metedologi Penelitian

Adapun tahapan metedologi penelitian ini adalah:

1. Studi Pustaka

Pada tahapan ini, penulis akan mengumpulkan sumber referensi yang didapatkan dari sumber tertulis seperti buku, jurnal, dan *proceeding*. Sumber referensi yang dicari akan berhubungan dengan *Elliptic Curve Cryptography* dan Skema *Blind Signature*.

2. Analisis dan Perancangan Sistem

Pada tahapan ini penulis menganalisis Skema *Blind Signature* dan digabungkan dengan *Elliptic Curve Cryptography* yang akan dirancang untuk malakukan proses *e-voting* dengan aman. Tahapan ini bertujuan untuk melihat sejauh mana keamanan dari proses *e-voting* ketika dijalankan.

3. Implementasi Sistem

Pengimplementasian sistem yang akan dilakukan adalah pembuatan sistem bedasarkan diagram alir (*flowchart*) yang telah dirancang dengan menggunakan kedua algoritma yaitu *Elliptic Curve Cryptography* dan *Blind Signature*.

4. Pengujian Sistem

Pada penelitian yang mengimplementasikan sistem *e-voting* ini, dibutuhkan pengujian untuk menguji apakah pengkombinasian algoritma *Blind Signature* dengan *Elliptic Curve Cryptography* dapat mengoptimalkan keamanan dalam melakukan *e-voting*. Pengujian akan dilakukan dengan mengenkripsi pesan dari *voting* lalu

mengirimkannya kepada validator dan penerima dan pesan akan didekripsi oleh penerima.

5. Dokumentasi Sistem

Setelah sistem dibangun, tahap selanjutnya adalah penyusunan laporan yang akan dilakukan dari tahapan analisis sampai ke pengujian dalam bentuk skripsi.

1.8 Sistematika Penulisan

BAB 1 PENDAHULUAN

Pada bab ini akan dijelaskan latar belakang penelitian, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, penelitian yang relevan, metodologi penelitian, serta sistematika penulisan.

BAB 2 LANDASAN TEORI

Bab ini menjelaskan tentang kajian teoritis terkait penelitian ini khususnya algoritma yang digunakan yaitu Skema Blind Signature dan Elliptic Curve Cryptography.

BAB 3 ANALISIS DAN PERANCANGAN

Menganalisis dan merancang sistem merupakan hal yang umum dilakukan pada penelitian. Proses tersebut akan dijabarkan pada bab ini.

BAB 4 IMPLEMENTASI DAN PENGUJIAN

Bab ini membahas penerapan sistem yang telah dikembangkan dan pengujian hasil implementasinya sesuai dengan desain yang telah disiapkan sebelumnya.

BAB 5 KESIMPULAN DAN SARAN

Bab ini mencakup rangkuman dari beberapa bab sebelumnya dan memberikan rekomendasi serta masukan bagi peneliti di masa mendatang.

BAB II

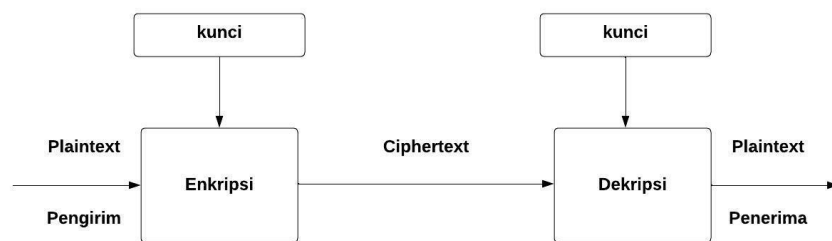
LANDASAN TEORI

2.1 E-Voting

Electronic voting adalah suatu metode pemungutan suara dan penghitungan suara dalam suatu pemilihan dengan menggunakan perangkat elektronik. Tujuan dari electronic voting adalah menyelenggarakan pemungutan suara dengan biaya hemat dan penghitungan suara yang cepat dengan menggunakan sistem yang aman dan mudah untuk dilakukan audit (Risnanto, 2017).

2.2 Kriptografi

Kriptografi merupakan sebuah cabang ilmu yang berkaitan dengan keamanan data dengan menggunakan teknik perhitungan. Maka dari itu pada jaman sekarang, kriptografi merupakan ilmu yang sangat dibutuhkan. Kriptografi biasanya mengalami dua proses yaitu proses enkripsi dan proses dekripsi. Proses enkripsi merupakan tahap untuk mengubah pesan biasa yang dapat dibaca (*plaintext*) menjadi sebuah pesan yang tidak terbaca dan tidak memiliki makna (*chipertext*). Setelah mengalami proses enkripsi, lalu pesan dikirim kepada penerima. Ketika pesan diterima oleh penerima, maka *chipertext* akan diubah kembali menjadi *plaintext*. Proses tersebut dinamakan dekripsi. Proses kriptografi digambarkan pada 2.1.



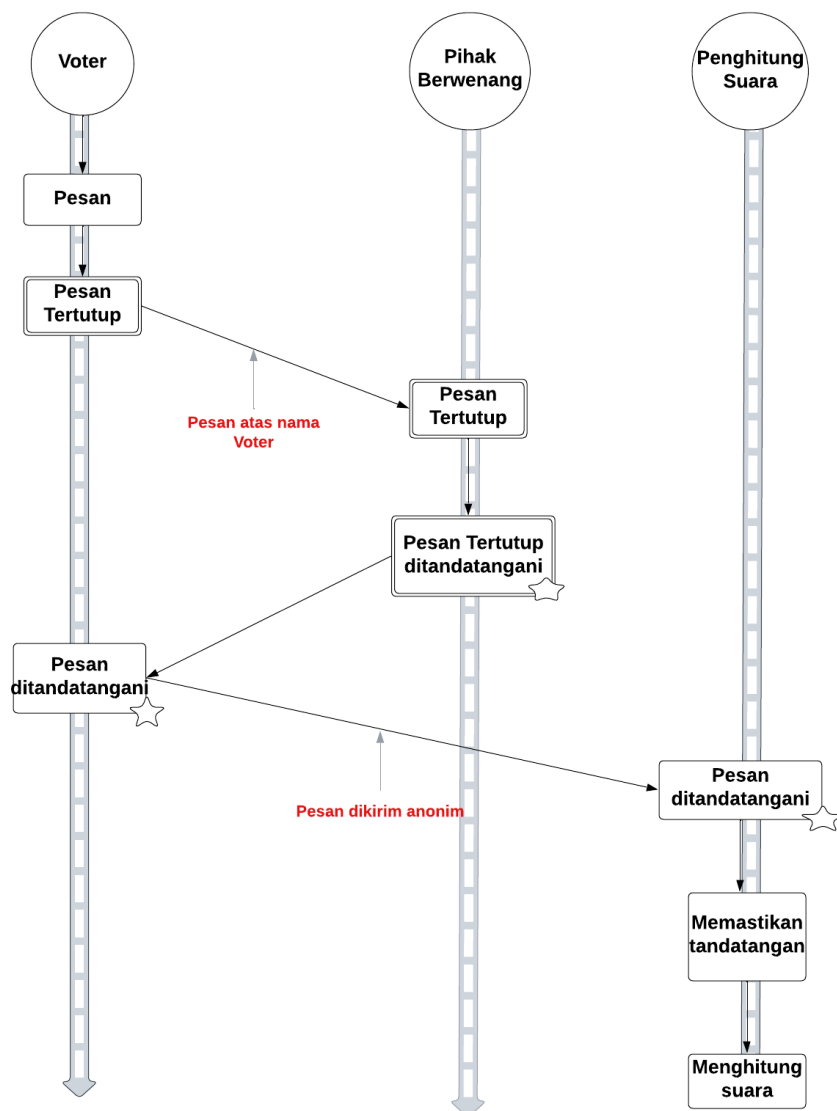
Gambar 2. 1 Proses Enkripsi dan Dekripsi

2.3 Blind Signature

Blind Signature dikenalkan pertama kali oleh David Chaum pada tahun 1982. *Blind Signature* adalah sebuah bentuk tanda-tangan digital di mana isi pesan disembunyikan sebelum ditanda-tangani. Hasil *Blind Signature* dapat diverifikasi secara publik terhadap

pesan asli yang tidak disembunyikan, seperti pada halnya tanda-tangan digital biasa (Prana, 2008).

Ilustrasi *Blind Signature* pada *e-voting* dapat dilihat pada gambar 2.2. *Voter* akan mengirimkan pesan suara kepada penghitung suara. Pertama *Voter* menuliskan isi suaranya dan menutup pesan tersebut. Pesan yang sudah ditutup dikirim kepada pihak yang berwenang atas nama *Voter*. Lalu pihak berwenang akan menerima pesan tersebut dan memastikan identitas *Voter*. Jika identitas *Voter* terdaftar sebagai pemilih, maka pesan tertutup tersebut akan ditandatangani oleh pihak yang berwenang. Akan tetapi pihak berwenang tidak dapat membuka pesan tertutup tersebut. Setelah itu pesan akan dikirimkan lagi ke *Voter*. *Voter* menerima pesan yang sudah ditandatangani tersebut dan membuka pesan untuk memastikan isi pesannya. Lalu *voter* mengirimkan pesan suara yang telah ditandatangani kepada penghitung suara secara anonim untuk melindungi identitasnya. Penghitung suara menerima pesan *Voter* dan memverifikasi kevalidan pesan suara melalui tanda tangan pihak berwenang. Jika tanda tangan valid maka isi pesan suara akan dihitung dan dicatat oleh penghitung suara.



Gambar 2. 2 *Blind Signature* pada *E-voting*

2.4 Elliptic Curve Cryptography

Elliptic Curve Cryptography (ECC) merupakan sebuah algoritma asimetris yang menggunakan kurva elips dalam prosesnya. ECC dinilai lebih baik dari algoritma Rivest Shamir Adleman (RSA) karena memiliki panjang kunci yang lebih kecil dengan tingkat yang sama pada keamanannya. Saat ini ada tiga protokol dari kurva elips, yaitu ECDH (*Elliptic Curve* Diffie-

Helman), EC ElGamal (Elliptic Curve El Gamal), dan ECDSA (*Elliptic Curve Digital Signature*). Kurva elips digunakan untuk menentukan kunci publik dari sistem kriptografi. Protokol yang digunakan pada penelitian ini adalah EC ElGamal (Elliptic Curve El Gamal). Kunci privat n secara acak dipilih dari $[1, p-1]$. n dan p secara berurutan adalah bilangan bulat dan bilangan prima dari sebuah persamaan kurva elips. Lalu kunci publik P dikomputasikan melalui $n \cdot G$, di mana $G = (x_G, y_G)$ merupakan titik pangkal dari $E(F_p)$. $n \cdot G$ merupakan perkalian skalar. Persamaan kurva elips pada bidang prima $E(F_p)$ adalah:

$$y^2 \equiv x^3 + ax + b \pmod{p}$$

$$\text{di mana } 4a^3 + 27b \pmod{p} \neq 0$$

Ada beberapa operasi titik dari ECC antara lain yaitu

1. Penambahan Titik

$$\lambda = (y_k - y_j) / (x_k - x_j) \pmod{p}$$

$$x_R = \lambda^2 - x_j - x_k \pmod{p}$$

$$y_R = \lambda(x_j - x_R) - y_j \pmod{p}$$

2. Penggandaan Titik

$$\lambda = (3x_p^2 + a/2y_p) \pmod{p}$$

$$x_R = (\lambda^2 - 2x_p) \pmod{p}$$

$$y_R = \lambda(x_p - x_R) - y_p \pmod{p}$$

3. Pengurangan Titik

$$P - Q = P + (-Q)$$

$$-Q = (x_Q, -y_Q \pmod{p})$$

2.4.1 Enkripsi ECC

Proses enkripsi *Elliptic Curve Cryptography* adalah sebagai berikut.

$$M' = [(kG), (M+kP)]$$

Dengan keterangan :

M = pesan yang akan dienkripsi dalam bentuk titik

M' = blok data yang telah dienkripsi (*ciphertext*)

k = suatu bilangan random yang akan digunakan sebagai kunci rahasia enkripsi dengan $k \in \{1, K, p-1\}$

P = kunci publik dari perkalian skalar $n \cdot G$

G = titik pangkal

2.4.2 Dekripsi ECC

Proses dekripsi yang dilakukan terhadap M' menggunakan perhitungan sebagai berikut.

$$M = (M' + kP) - [n(kG)]$$

2.4.3 Perubahan Plaintext Menjadi Titik

Untuk mengoperasikan pesan (*plaintext*) pada kurva elips, setiap karakter harus diubah ke dalam titik yang ada pada kurva elips. Untuk mengubahnya maka digunakan Metode Kolbitz dengan langkah sebagai berikut.

1. Bentuk karakter (m) yang terdiri dari angka 0,1,2,...,9 dan huruf A,B,C,...,Z yang berjumlah 35 karakter. Setiap karakter diwakili dengan angka 0-35 yang dapat dilihat pada tabel 2.1.

Tabel 2. 1 Tabel Enkoding

0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
A	10
B	11
C	12

D	13
E	14
F	15
G	16
H	17
I	18
J	19
K	20
L	21
M	22
N	23
O	24
P	25
Q	26
R	27
S	28
T	29
U	30
V	31
W	32
X	33
Y	34
Z	35

2. Pilih sebuah bilangan bulat random k sebagai basis
3. Pada persamaan kurva elips $y^2 \equiv x^3 + ax + b \pmod{p}$ hitung x melalui rumus $x = ms + 1$. Lalu hitung y melalui persamaan tersebut. Jika nilai y tidak memenuhi maka gunakan $x = ms+2$, $x = ms+3$, dan seterusnya sampai nilai y memenuhi.
4. Pada proses dekripsi dari titik mejadi karakter gunakan rumus $[m = (x - 1/s)]$

2.4.4 Contoh Kasus

Misalkan sebuah pesan yang berisi “D” ingin dikirimkan kepada seorang penerima pesan. Pengiriman pesan melalui serangkaian proses sebagai berikut.

1. Pengirim mengubah karakter dengan angka sesuai tabel enkoding “D” = 13.
2. Lalu menentukan $s = 8$. Pada persamaan kurva elips $y^2 \equiv x^3 - 2x + 160 \pmod{911}$
3. Masukkan persamaan $x = ms + 1$. Maka $x = (13 \times 8) + 1 = 105$.
4. Jika dimasukkan $105^3 - 2(105) + 160 \pmod{911} = 605$. Hasilnya akan sama dengan $337^2 \pmod{911} = 605$. Jadi titik dipetakan pada M (105,337)
5. Enkripsi titik tersebut menggunakan G(1,75) pada persamaan kurva
6. Penerima menentukan kunci publik P dengan cara $n \cdot G$ di mana $n=2$ maka $2(1,75)$. $P = (51,532)$ lalu disebar ke setiap pengirim pesan.
7. Lalu pengirim menghitung $M' = [(kG), (M+kP)]$. Di mana $k=3$.
 $M' = [(3(1,75)), ((105,337)+3(51,532))]$.
 $M' = [(338,835), (454,27)]$ dikirimkan ke penerima
8. Penerima melakukan dekripsi menggunakan
 $M = (M + kP) - [n(kG)]$.
9. $(M + kP)$ merupakan titik kedua yang dikirim,
10. (kG) merupakan titik pertama yang akan dikalikan dengan kunci privat dari penerima n yang diketahui penerima $n=2$.
11. Penerima menghitung
 $n(kG) = 2(338,835)$
 $n(kG) = (474,777)$.
12. Lalu penerima menghitung
 $(M + kP) - [n(kG)] = (454,27) - (474,777)$
 $M = (105,337)$
13. Penerima melakukan perubahan kembali dari titik menjadi karakter dengan rumus

$$[m = (x - 1/s)] = \frac{105-1}{8}.$$

$m=13$, maka a_{13} mewakili karakter "D". Penerima mendapat pesan dari pengirim yang berisi "D".

BAB III

ANALISIS DAN PERANCANGAN

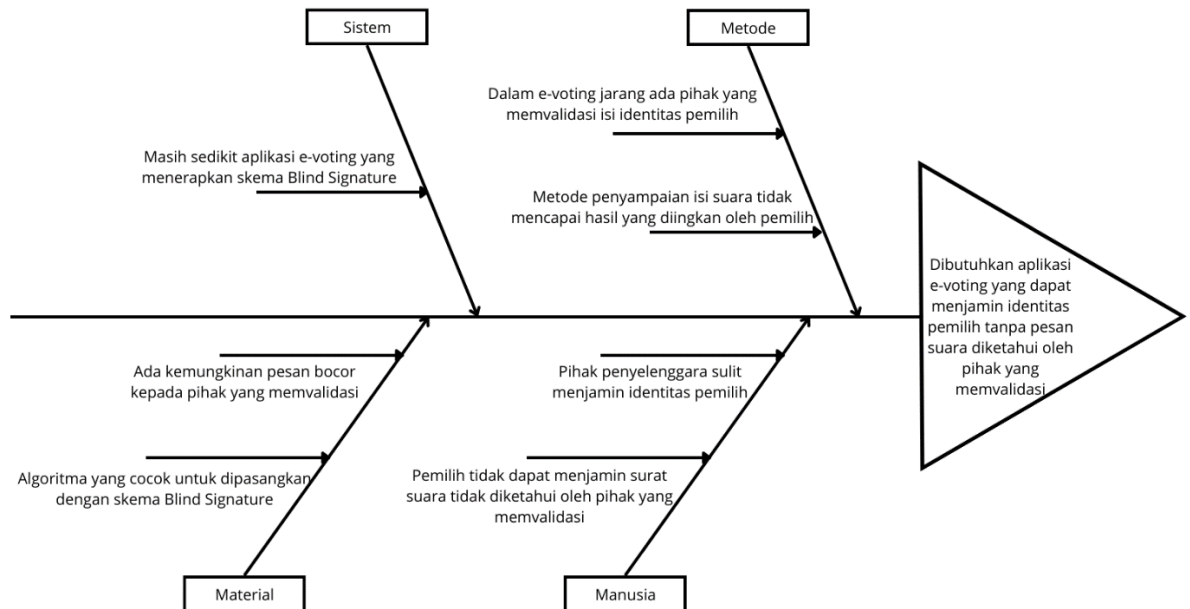
3.1 Analisis Sistem

Analisis sistem merupakan sebuah tahapan menganalisis masalah dalam perancangan sistem yang bertujuan untuk mendapatkan hasil dari pemecahan masalah yang dapat mengoptimalkan kerja sistem. Tahapan ini terbagi menjadi dua proses yaitu analisis masalah dan analisis persyaratan. Analisis masalah merupakan proses mengidentifikasi masalah untuk mendapatkan solusi terbaik dari permasalahan tersebut, sedangkan analisis persyaratan merupakan sebuah proses untuk mengidentifikasi persyaratan apa saja yang harus dipenuhi untuk memecahkan masalah sistem yang dibangun.

3.1.1 Analisis Masalah

Pada suatu pemungutan suara elektronik (e-voting) ada tiga pihak utama yang terlibat yaitu *Trustee* adalah pihak yang menyelenggarakan pemungutan suara, *Voter* adalah pihak yang memberikan suaranya, dan *Signer* adalah pihak yang memastikan bahwa *Voter* adalah pemilih yang memang berhak untuk memberikan suara. Selain itu, ada persyaratan bahwa suara yang diberikan *Voter* harus bersifat rahasia yaitu tidak dapat diketahui oleh *Signer*.

Faktor-faktor penyebab timbulnya masalah yang akan diilustrasikan melalui Diagram Ishikawa (*Fishbone Diagram*) yang terletak pada gambar 3.1. Terlihat bahwa ada beberapa penyebab masalah yang diilustrasikan sebagai tulang ikan. Penyebab masalah ini dibagi menjadi empat kategori yaitu sistem, amterial, metode, dan manusia. Sedangkan kepala ikan mengilustrasikan simpulan permasalahan yang akan diselesaikan.



Gambar 3. 1 *Fishbone Diagram*

3.1.2 Analisis Persyaratan

Analisis persyaratan adalah proses merumuskan syarat-syarat yang harus dipenuhi oleh sistem yang akan dibangun baik, dari segi fungsional maupun non-fungsional.

3.1.2.1 Analisis Persyaratan Fungsional

Adapun beberapa persyaratan fungsional yang dianalisis yaitu.

1. Sistem dapat mengamankan suara menggunakan algoritma *Elliptic Curve Cryptography*
2. Sistem dapat membangkitkan *private key* dan *public key* dari setiap pembuatan akun baru
3. Sistem dapat melakukan validasi terhadap surat suara secara tertutup dengan Skema Blind Signature
4. Surat suara yang telah divalidasi baru dapat dihitung suaranya oleh *Trustee* tanpa mengetahui identitas dari *Voter*

3.1.2.2 Analisis Persyaratan Non-fungsional

Adapun persyaratan non-fungsional yang dianalisis pada sistem yang akan dibangun yaitu.

1. *User-Interface*

Sistem dapat menampilkan antarmuka pengguna yang mudah dimengerti oleh pengguna baru

2. Kualitas

Kualitas yang baik dari sistem yaitu sistem dapat menunjukkan proses enkripsi dan proses dekripsi pada setiap tahapan

3. Performa

Sistem dapat menangani banyaknya jumlah pemilih dan setiap *event* yang diadakan

3.2 Perancangan Sistem

Dalam merancang sistem, dibutuhkan beberapa diagram pada perancangan ini yaitu diagram umum, *use case diagram*, *activity diagram*, *sequence diagram*, diagram alir (*flowchart*), dan *User interface*.

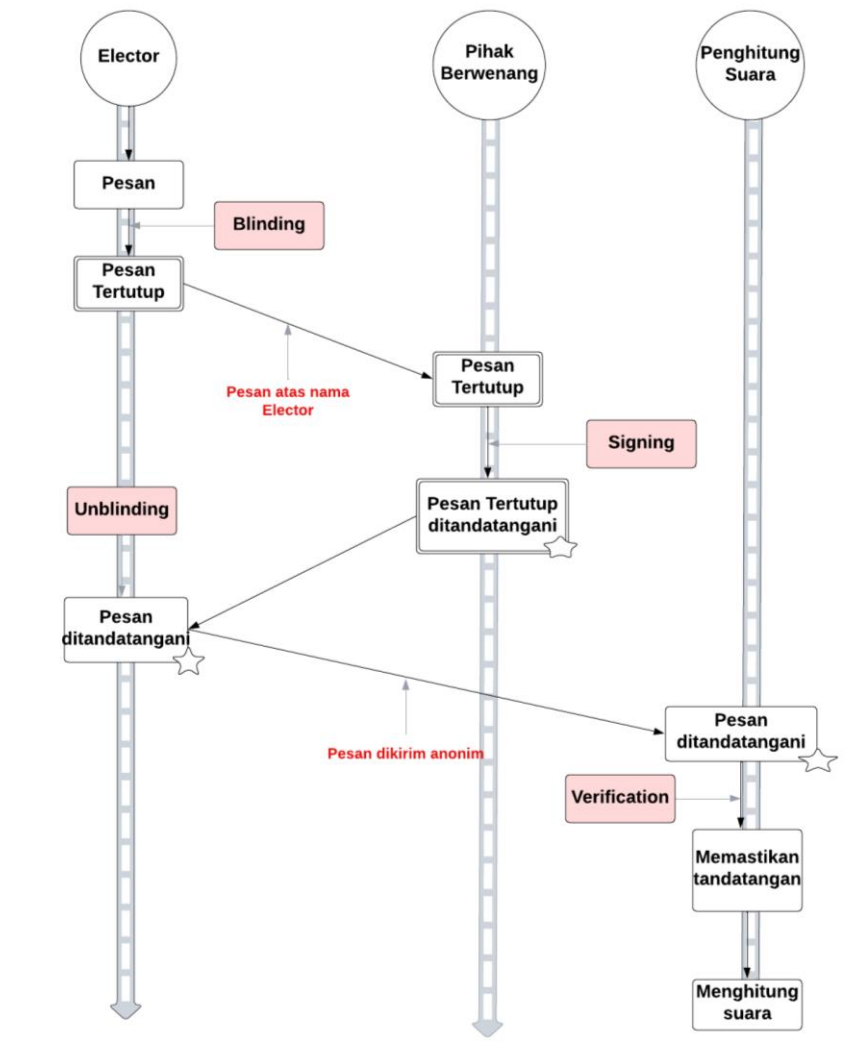
3.2.1 Diagram Umum

Rancangan yang dilakukan pada diagram umum menggunakan *timing diagram* yang terdapat pada Skema *Blind Signature* lalu digabungkan dengan pemrosesan dengan menggunakan algoritma *Elliptic Curve Cryptography*. Diagram umum diilustrasikan pada gambar 3.2. Berikut runtutan proses yang terjadi pada diagram umum.

1. Pertama, *User* akan melakukan registrasi akun. Pada proses ini sistem akan membangkitkan kunci privat *User*. Setelah itu sistem akan menghitung kunci publik dari kunci privat yang telah ditentukan oleh sistem.
2. Pada proses *blinding*, pemilih (*Voter*) akan melakukan enkripsi terhadap pesan suara yang berisi pilihannya menggunakan *Elliptic Curve Elgamal* menggunakan kunci publik *Trustee*. Selain itu pada proses ini *User* juga akan mengenkripsi pesan suaranya menggunakan kunci privat dan publik pengirim untuk dikirim ke *Signer*.
3. Setelah *Signer* menerima pesan dari *Voter*, dia akan melakukan *signing* terhadap pesan menggunakan kunci privat *Signer* tersebut. Setelah itu *Signer* akan mengirimkan kembali pesan kepada *Voter*.
4. Selanjutnya *Voter* yang menerima pesan dari *Signer* akan melakukan *unblinding* terhadap surat suara yang telah di-*signing* menggunakan kunci privat *Voter* dan

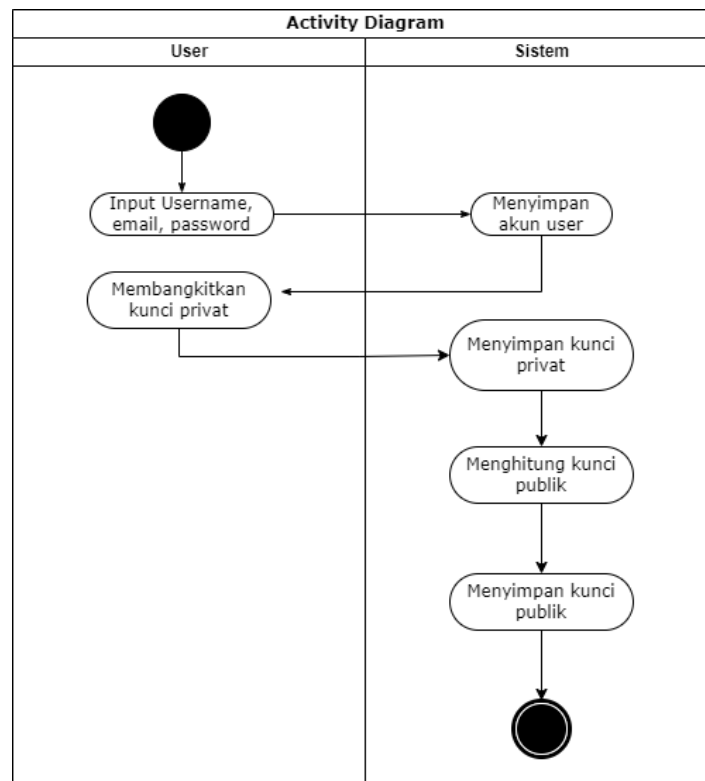
kunci publik *Signer*. Selanjutnya *Voter* akan mengirimkan pesan suara tersebut kepada *Trustee* untuk diverifikasi.

5. *Trustee* akan menerima pesan yang telah di-*unblinding* dan melakukan verifikasi menggunakan kunci publik *Signer*. Jika verifikasi memberi hasil yang sesuai, maka *Trustee* akan melakukan dekripsi terhadap pesan suara menggunakan kunci privat dari *Trustee*.



Gambar 3. 2 Diagram Umum

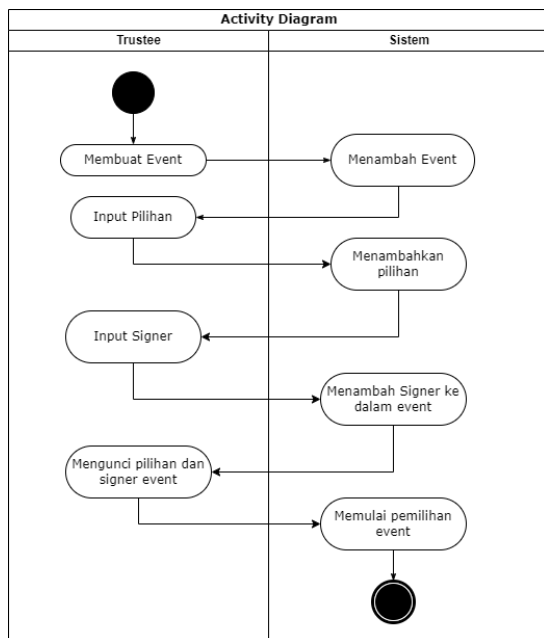
melakukan register, sistem akan membangkitkan kunci privat lalu menghitung kunci publik dari *User*.



Gambar 3. 4 *Diagram Activity Register*

3.2.3.2 *Activity Diagram Membuat Event Pemilihan*

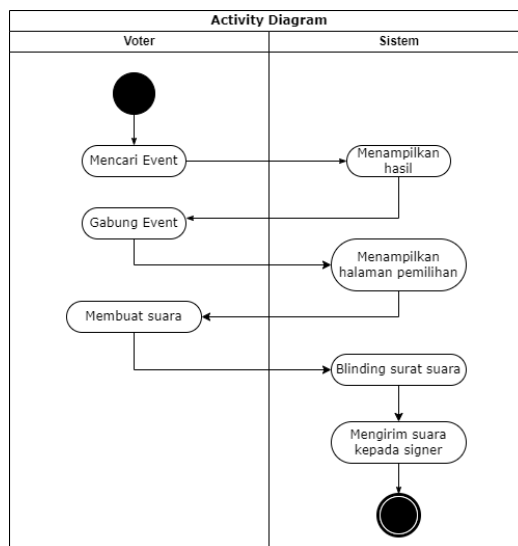
Pemilihan dilakukan melalui sebuah *event*. *Event* hanya dapat dibuat oleh *Trustee*. *Activity diagram* membuat *event* pemilihan ada pada gambar 3.5. Setelah *Trustee* membuat *event*, *Trustee* akan membuat beberapa calon yang akan menjadi pilihan pada *event* tersebut. Lalu *Trustee* akan memasukkan *Signer* yang dipercaya untuk menandatangani surat yang diberikan *Voter*. *Event* pemilihan tidak akan terlaksana sebelum *Trustee* mengunci daftar calon pada *event*. Setelah mengunci daftar calon, *Trustee* tidak dapat mengubah *Signer* dan calon yang telah diinput.



Gambar 3. 5 *Diagram Activity Membuat Event*

3.2.3.3 Activity Diagram Membuat Suara

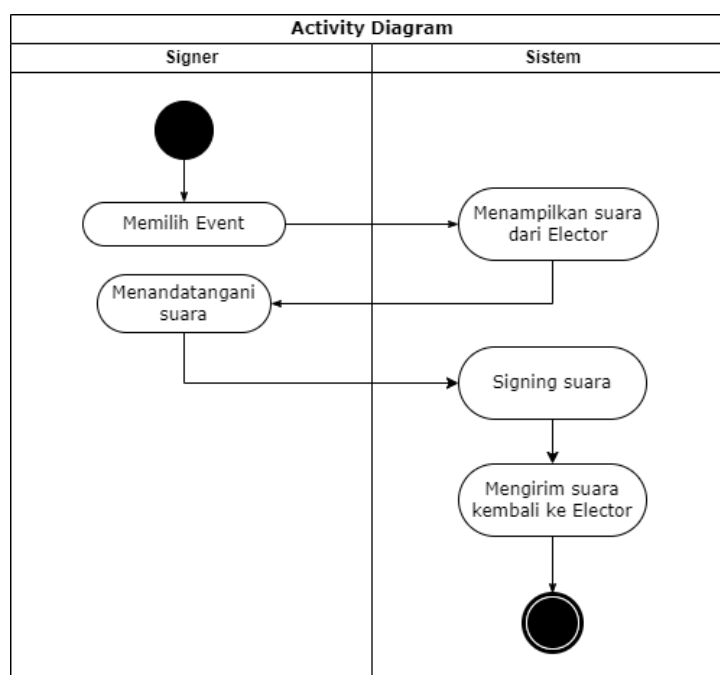
Pada proses ini, *Voter* akan mengirimkan surat suara yang berisi pilihannya. Sebelum itu, *Voter* harus mencari *event* pemilihan yang akan diikuti menggunakan *code event*. Setelah itu *user* dapat mengikut pemilihan lalu pesan akan di-*blinding* dan dikirim ke *Signer event*. Gambar diagram *activity* membuat suara disediakan pada gambar 3.6.



Gambar 3. 6 *Diagram Activity Membuat Suara*

3.2.3.4 Activity Diagram Signing

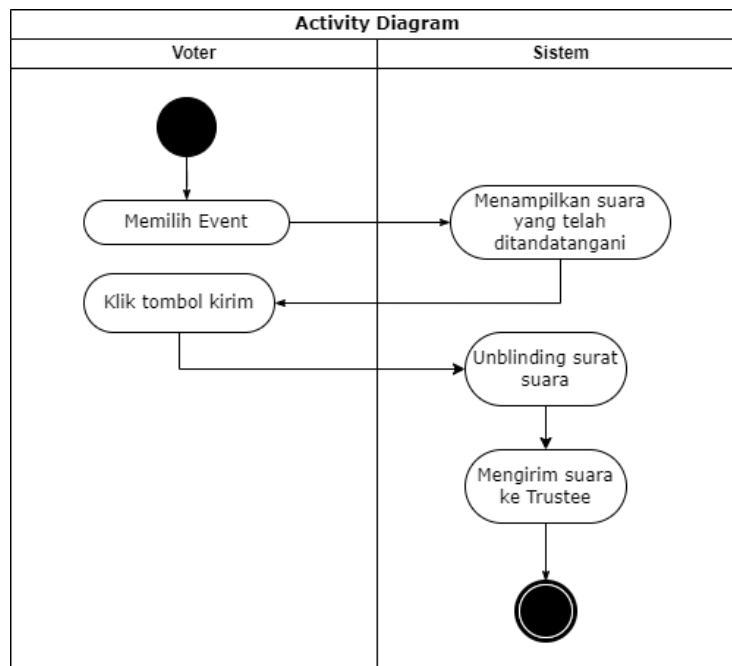
Proses *signing* bertujuan untuk menandatangani surat suara yang telah dibuat oleh *Voter*. *Actor* pada aktivitas ini adalah seorang *Signer*. *Signer* akan memilih *event* yang sudah didaftarkan oleh *Trustee*. Ketika *Voter* telah mengirim surat suaranya pada *event* tersebut, *Signer* dapat melihat apakah *Voter* merupakan orang yang berhak untuk mengikuti pemilihan melalui *Username*, *email*, dan nomor identitas *Voter*. Jika memang *Voter* berhak untuk mengikuti pemilihan maka *Signer* dapat melakukan *signing* terhadap surat suaranya. Setelah itu surat suara yang telah ditandatangani akan dikirim kembali ke *Voter*. Gambar diagram *activity* ini dapat dilihat pada gambar 3.7.



Gambar 3. 7 Diagram Activity Signing

3.2.3.5 Activity Diagram Unblinding

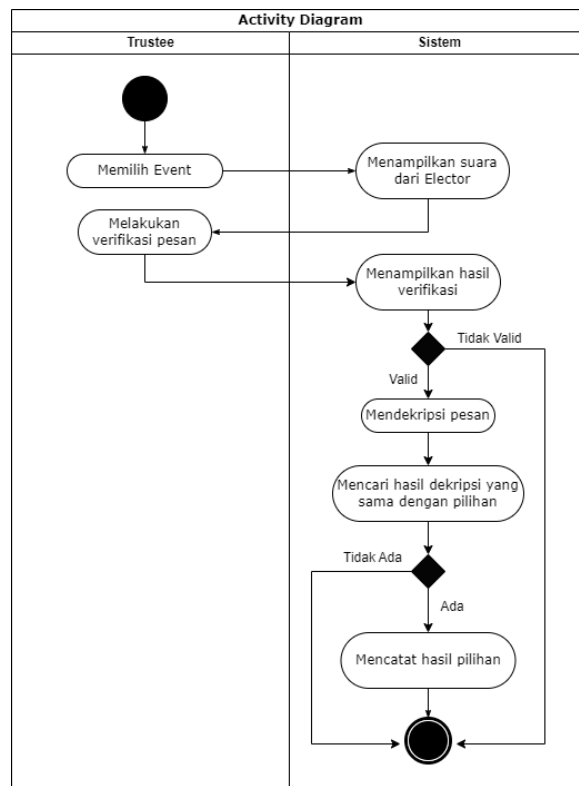
Unblinding adalah proses di mana *Voter* melakukan dekripsi terhadap surat suara yang ditandatangani oleh *Signer*. Proses *unblinding* pesan dilakukan oleh *Voter* setelah ia menerima pesan yang telah ditandatangani oleh *Signer*. Setelah *unblinding* dilakukan pesan akan langsung dikirim ke *Trustee* dengan anonim. Diagram aktivitas ini terdapat pada gambar 3.8..



Gambar 3. 8 *Diagram Activity Unblinding*

3.2.3.6 Activity Diagram Verification

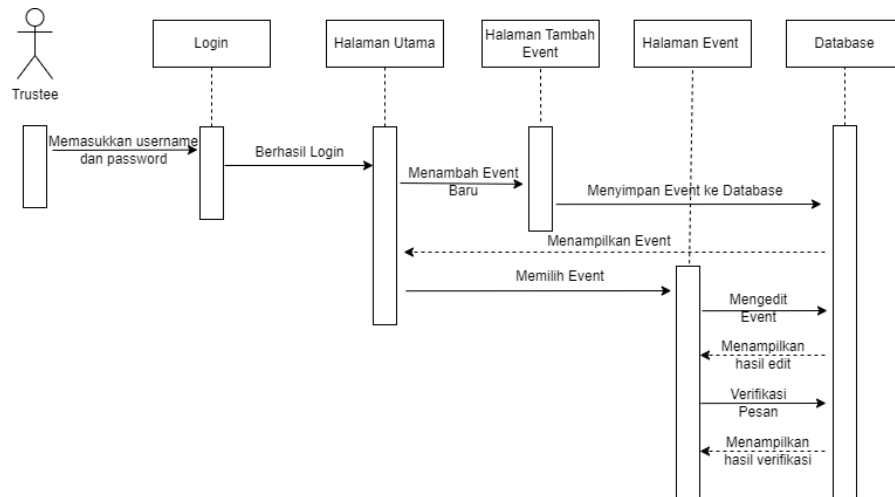
Verification adalah proses memverifikasi tanda tangan *Signer* pada surat suara yang bertujuan untuk melihat apakah *Signer* yang menandatangani surat adalah *Signer* yang ditunjuk oleh *Trustee*. Proses *Verification* dilakukan oleh *Trustee* setelah menerima suara yang dikirim oleh *Voter*. *Trustee* akan melakukan verifikasi terhadap tandatangan *Signer* surat tersebut. Jika verifikasi memberikan hasil yang sesuai, maka sistem akan mengecek apakah isi surat suara ada yang sama dengan pilihan yang tersedia. Jika ada, maka pilihan tersebut yang akan dicatat. Diagram *verification* terdapat di gambar 3.9.



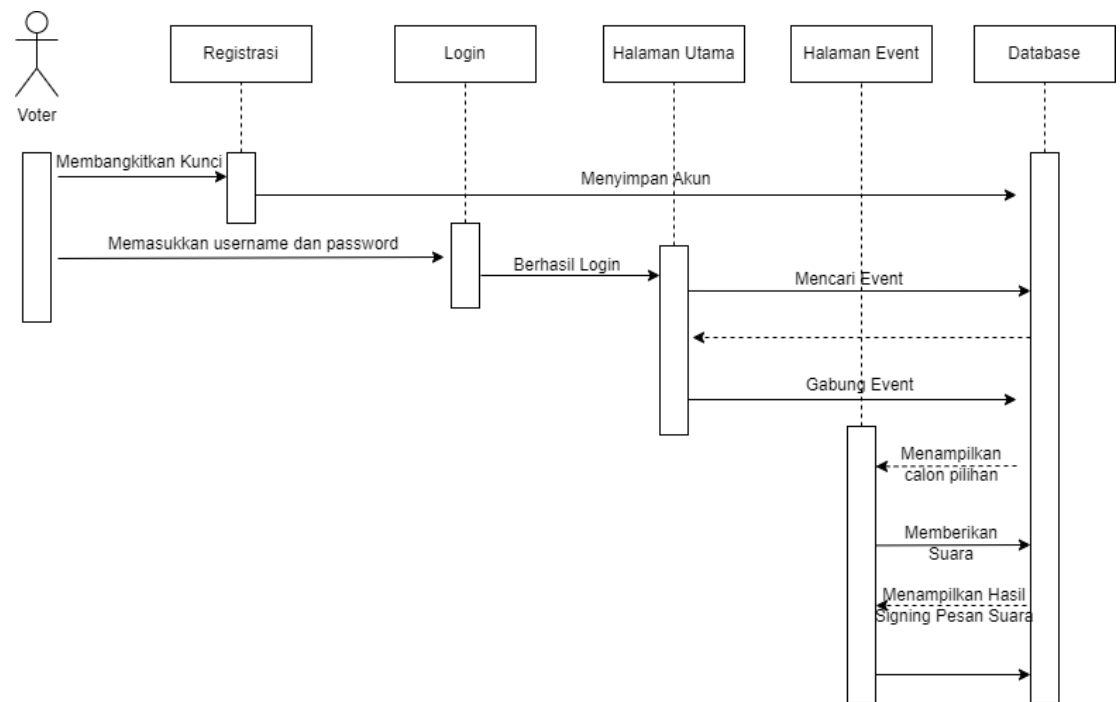
Gambar 3. 9 Diagram Activity Verification

3.2.4 Sequence Diagram

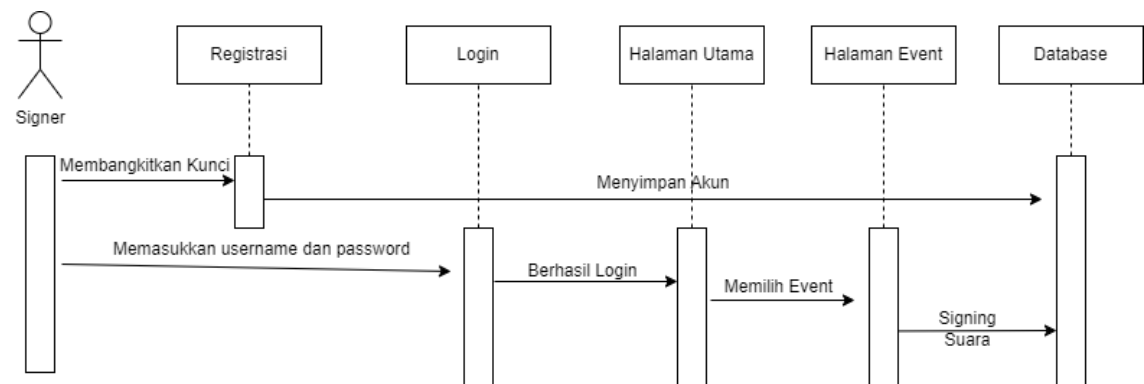
Sequence Diagram menunjukkan interaksi yang dilakukan oleh sistem dan pengguna secara berurutan. *User* yang terdapat pada diagram ini adalah *Voter*, *Signer*, dan *Trustee* yang secara berurutan ada pada gambar 3.10, 3.11, 3.12.



Gambar 3. 10 Sequence Diagram Trustee



Gambar 3. 11 *Sequence Diagram Voter*



Gambar 3. 12 *Sequence Diagram Signer*

3.2.5 Diagram Alir (*Flowchart*)

Diagram alir (*flowchart*) adalah bentuk visual dari serangkaian langkah atau proses dari sebuah sistem atau algoritma. Melalui penggunaan symbol-simbol yang saling terhubung, diagram alir mewakili perkembangan dari aktivitas suatu proses. *Flowchart* membantu memahami bagaimana informasi dari satu langkah ke langkah lainnya. *Flowchart* yang dibuat pada penelitian ini adalah *flowchart* pembangkitan

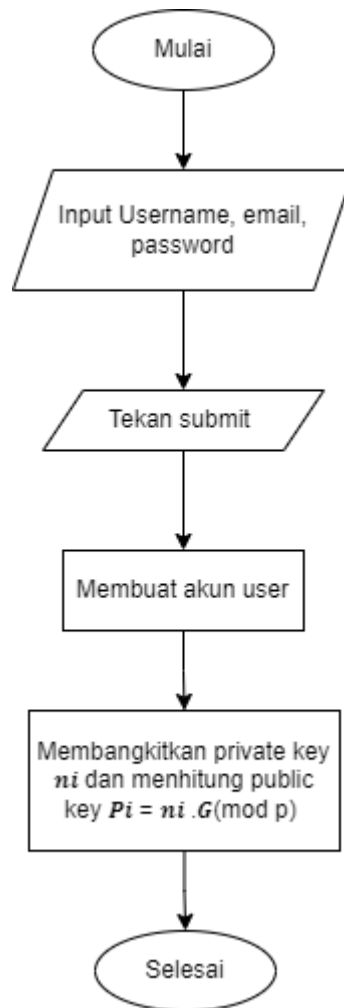
kunci, *flowchart blinding*, *flowchart unblinding*, *flowchart verification*. Ada beberapa simbol dari perhitungan yang dituliskan pada tabel di bawah ini.

Tabel 3. 1 Tabel Simbol Perhitungan

Simbol	Penjelasan
n_i	Kunci Privat Voter
P_i	Kunci Publik Voter
n_s	Kunci Privat Signer
n_v	Kunci Rahasia Signer
P_s	Kunci Publik Signer
n_t	Kunci Privat Trustee
P_t	Kunci Publik Trustee
G	Titik basis dari persamaan
p	Modular P dari persamaan kuva p-256
α	Pesan yang di-blinding
m	Pesan yang diubah ke dalam bentuk <i>hash</i>
M	Bilangan m yang diubah ke dalam titik
k	Kunci rahasia pesan
K	k yang diubah ke dalam bentuk titik
C	Ciphertext pesan
r	Pesan yang di-signing oleh Signer
s	Tandatangan Signer
s'	Hasil dekripsi s
m'	Pasangan s' yang dibutuhkan untuk verifikasi

3.2.5.1 *Flowchart* Pembangkitan Kunci

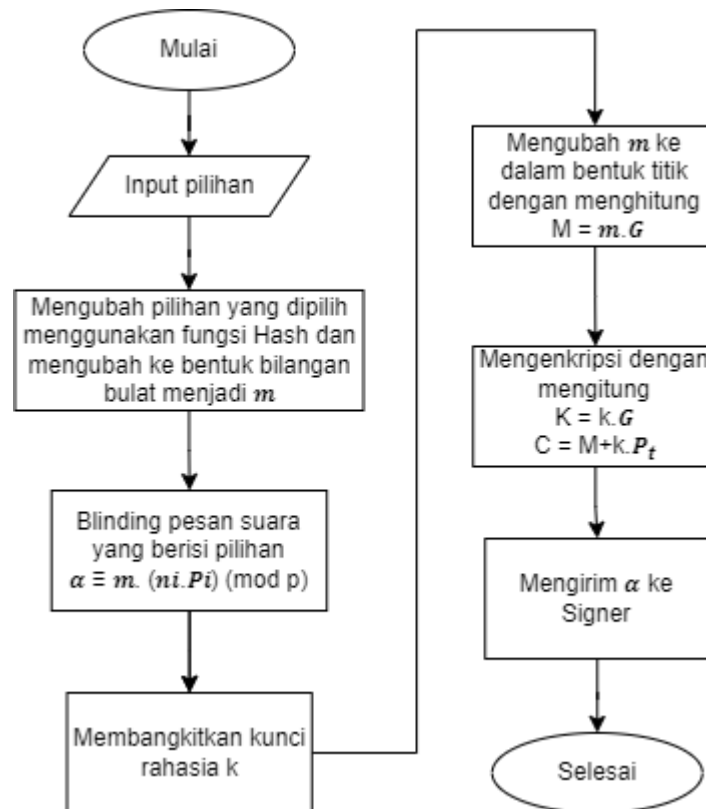
Gambar 3.13 merupakan *flowchart* untuk pembangkitan kunci pada setiap *User*. *User* melakukan pembangkitan kunci ketika pada tahap registrasi akun. Pertama kunci privat dibangkitkan. Lalu sistem akan menghitung kunci publik menggunakan operasi perkalian antara kunci privat dan base dari persamaan kurva elips.



Gambar 3. 13 *Flowchart* Pembangkitan Kunci

3.2.5.2 *Flowchart Blinding*

Proses *blinding* terjadi ketika *Voter* ingin membuat suara yang berisi pilihannya. Surat suara akan diubah menggunakan fungsi SHA-256 dan diubah menjadi bilangan bulat. Setelah itu pesan akan dikalikan dengan base dari kurva elips untuk mendapatkan titik pada kurva. Proses *blinding* menghasilkan dua enkripsi yaitu pesan yang dienkripsi dengan menggunakan *Elliptic Curve Elgamal* dan publik *Trustee* dan pesan yang dienkripsi untuk dilakukan *signing* yang akan dikirim ke *Signer*. Proses ini diilustrasikan di gambar 3.14.



Gambar 3. 14 *Flowchart Blinding*

3.2.5.3 Flowchart Signing

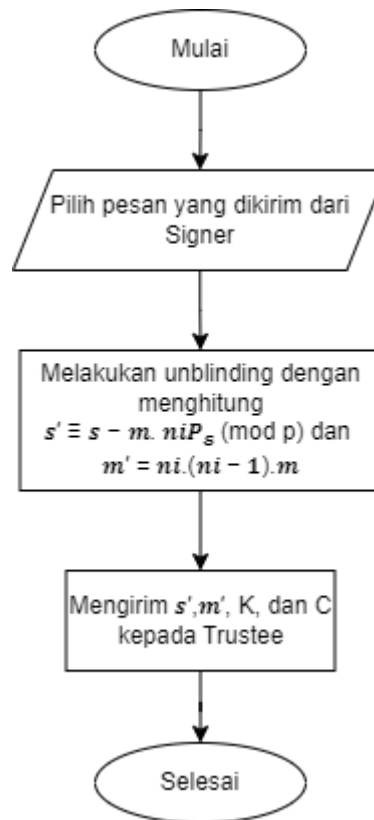
Pada tahap ini, *Signer* melakukan *signing* dengan menghitung r menggunakan kunci privatnya dan menghitung s menggunakan kunci rahasia yang akan dibangkitkan oleh *Signer* yaitu n_v . Setelah itu, r dan s akan dikirimkan kembali ke *Voter*. Gambar 3.15 menunjukkan *flowchart* proses *signing*.



Gambar 3. 15 *Flowchart Signing*

3.2.5.4 *Flowchart Unblinding*

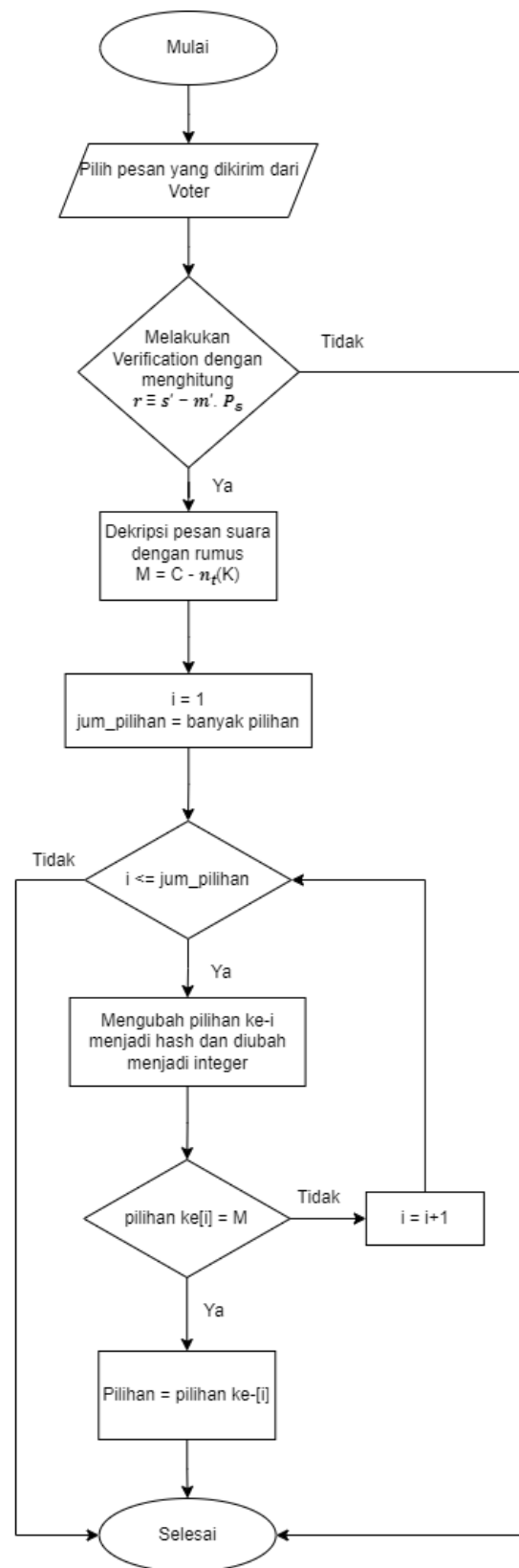
Setelah *User* mendapatkan surat suaranya yang telah ditandatangani, *Voter* akan melakukan *unblinding* terhadap surat suara tersebut. Proses *unblinding* akan menghasilkan s' dan m' . *Voter* menghitung s' menggunakan *Public Key Signer* P_s dan *Private Key Voter* n_i . Lalu s' dan m' akan dikirim kepada *Trustee*. *Flowchart unblinding* dapat dilihat pada gambar 3.16.



Gambar 3. 16 *Flowchart Unblinding*

3.2.5.5 *Flowchart Verification*

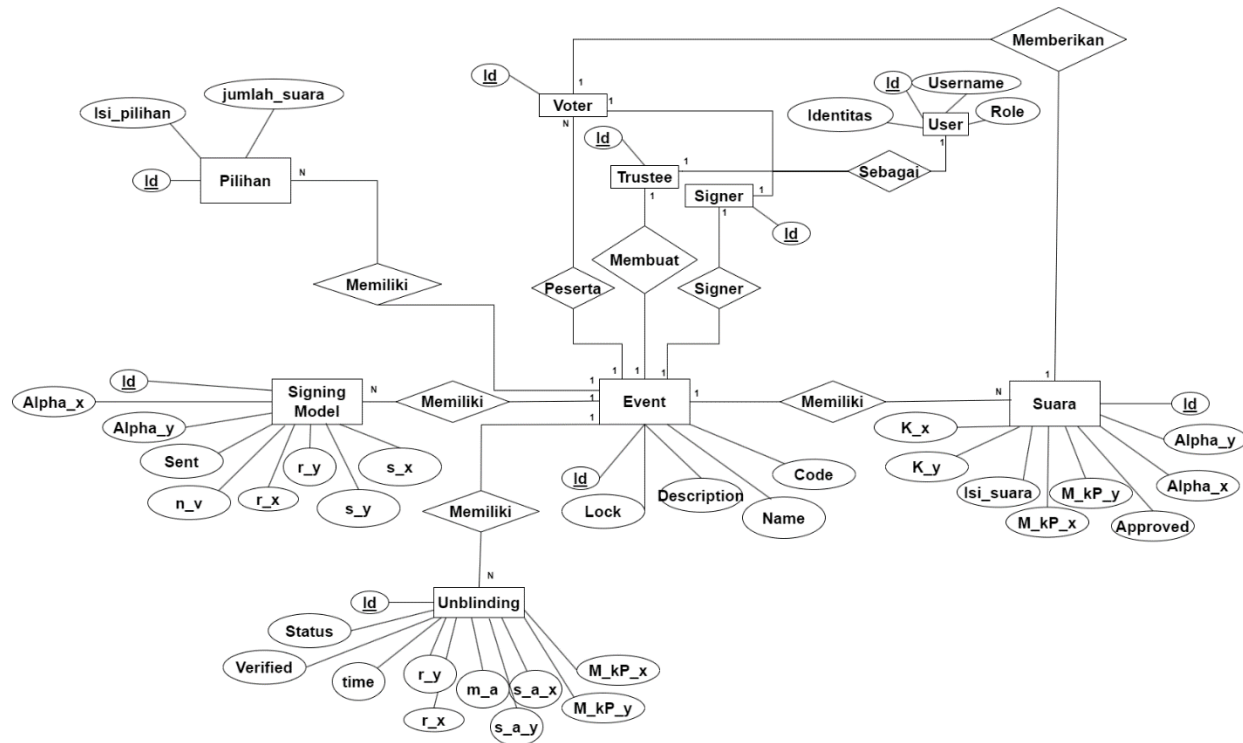
Trustee melakukan verifikasi terhadap pesan yang telah dikirimkan oleh *Voter* secara anonim. *Trustee* akan memastikan tanda tangan *Signer* menggunakan kunci publik *Signer* P_s . Jika $r \equiv s' - m' \cdot P_s$, maka *Signer* yang memberikan *signing* adalah *Signer* yang tepat. Lalu *Trustee* akan melakukan dekripsi terhadap pesan menggunakan kunci privatnya lalu mencocokkan isi pesan dengan pilihan yang tersedia. Jika ada, maka pilihan dicatat. *Flowchart* ada pada gambar 3.17.



Gambar 3. 17 *Flowchart Verification*

3.2.6 Perancangan Database

Pada tahapan ini, penulis melakukan perancangan database untuk merancang data-data apa saja yang disimpan dan bagaimana hubungan antar tabel. Pada sistem yang dibangun, penulis membuat beberapa tabel yaitu Tabel *User*, *Voter*, *Trustee*, *Signer*, *Admin*, *Key*, *Event*, *Pilihan*, *Suara*, *Signing Model*, dan *Unblinding*.



Gambar 3. 18 ERD

Penjelasan gambar 3.18 adalah sebagai berikut.

1. Setiap *User* hanya dapat memilih satu *role* sebagai *Voter*, *Trustee*, atau *Signer*
2. Seorang *Trustee* dapat membuat banyak *event*
3. Sebuah *event* dapat memiliki beberapa *voter*, *pilihan*, *suara*, *signing model*, dan *unblinding*
4. Sebuah *event* hanya dapat memiliki sebuah *Signer*
5. Seorang *Voter* yang telah bergabung dengan sebuah *event* dapat memberikan sebuah *suara*.

Tabel 3. 2 Tabel *User*

Nama	Tipe Data	Ukuran	Keterangan
<u>id</u>	Integer	2147483647	Menyimpan id <i>User</i> sebagai kunci primer
username	Varchar	150	Menyimpan <i>Username User</i>
role	Varchar	10	Menyimpan role <i>User</i>
identitas	Integer	20	Menyimpan nomor identitas <i>User</i>

Tabel 3. 3 Tabel *Voter*

Nama	Tipe Data	Ukuran	Keterangan
<u>id</u>	Integer	2147483647	Menyimpan id <i>Voter</i> sebagai kunci primer
user_id	Integer	2147483647	Menyimpan id <i>User</i>

Tabel 3. 4 Tabel *Trustee*

Nama	Tipe Data	Ukuran	Keterangan
<u>id</u>	Integer	2147483647	Menyimpan id <i>Trustee</i> sebagai kunci primer
user_id	Varchar	2147483647	Menyimpan id <i>User</i>

Tabel 3. 5 Tabel *Signer*

Nama	Tipe Data	Ukuran	Keterangan
<u>id</u>	Integer	2147483647	Menyimpan id <i>Signer</i> sebagai kunci primer
user_id	Integer	2147483647	Menyimpan id <i>User</i>

Tabel 3. 6 Tabel Admin

Nama	Tipe Data	Ukuran	Keterangan
<u>id</u>	Integer	2147483647	Menyimpan id Admin sebagai kunci primer
user_id	Integer	2147483647	Menyimpan id <i>User</i>

Tabel 3. 7 Tabel Key

Nama	Tipe Data	Ukuran	Keterangan
<u>id</u>	Integer	2147483647	Menyimpan id kunci sebagai kunci primer
user_id	Integer	2147483647	Menyimpan id <i>User</i>
private_key_user	Varchar	66	Menyimpan kunci privat
public_key_user_x	Varchar	66	Menyimpan kunci publik titik x
public_key_user_y	Varchar	66	Menyimpan kunci publik titik y

Tabel 3. 8 Tabel Event

Nama	Tipe Data	Ukuran	Keterangan
<u>id</u>	Integer	2147483647	Menyimpan id event sebagai kunci primer
name	Varchar	20	Menyimpan nama event
description	Varchar	200	Menyimpan deskripsi event
code	Varchar	20	Menyimpan kode event
voter_id	Integer	2147483647	Menyimpan id <i>Voter</i> yang bergabung dalam event
trustee_id	Integer	2147483647	Menyimpan id <i>Trustee</i> event
signer_id	Integer	2147483647	Menyimpan id <i>Signer</i> event

lock	Boolean		Parameter untuk menyatakan calon pilihan pada event telah dikunci
------	---------	--	---

Tabel 3. 9 Tabel Suara

Nama	Tipe Data	Ukuran	Keterangan
<u>id</u>	Integer	2147483647	Menyimpan id surat suara yang diberikan oleh <i>Voter</i> sebagai kunci primer
isi	Varchar	100	Menyimpan isi surat suara yang berisi nama calon pilihan
event_id	Integer	2147483647	Menyimpan id event
approved	Boolean	20	Parameter surat suara sudah ditandatangani atau belum
voter_id	Integer	2147483647	Menyimpan id <i>Voter</i> yang memberikan suara
alpha_x	Varchar	66	Menyimpan hasil perhitungan α pada titik x
alpha_y	Varchar	66	Menyimpan hasil perhitungan α pada titik y
K_x	Varchar	66	Menyimpan hasil perhitungan K pada titik x
K_y	Varchar	66	Menyimpan hasil perhitungan K pada titik y
M_kP_x	Varchar	66	Menyimpan hasil perhitungan C pada titik x
M_kP_y	Varchar	66	Menyimpan hasil perhitungan C pada titik y

Tabel 3. 10 Tabel Pilihan

Nama	Tipe Data	Ukuran	Keterangan
<u>id</u>	Integer	2147483647	Menyimpan id pilihan sebagai kunci primer
event_id	Integer	2147483647	Menyimpan id event
isi	Varchar	100	Menyimpan nama calon pilihan
jumlah_suara	Integer	2147483647	Menyimpan jumlah suara yang didapat calon pilihan

Tabel 3. 11 Tabel *Signing Model*

Nama	Tipe Data	Ukuran	Keterangan
<u>id</u>	Integer	2147483647	Menyimpan id surat suara yang telah ditandatangani oleh <i>Signer</i> sebagai kunci primer
sent	Boolean		Parameter apakah surat suara telah dikirim kepada <i>Trustee</i>
event_id	Integer	2147483647	Menyimpan id event
n_v	Varchar	66	Menyimpan kunci rahasia untuk melakukan signing
voter_id	Integer	2147483647	Menyimpan id <i>Voter</i> yang memberikan suara
alpha_x	Varchar	66	Menyimpan α pada titik x dari tabel suara
alpha_y	Varchar	66	Menyimpan α pada titik y dari tabel suara
r_x	Varchar	66	Menyimpan hasil perhitungan r pada titik x

r_y	Varchar	66	Menyimpan hasil perhitungan r pada titik y
s_x	Varchar	66	Menyimpan hasil perhitungan s pada titik x
s_y	Varchar	66	Menyimpan hasil perhitungan s pada titik y

Tabel 3. 12 Tabel *Unblinding*

Nama	Tipe Data	Ukuran	Keterangan
<u>id</u>	Integer	2147483647	Menyimpan id surat suara setelah di-unblinding sebagai kunci primer
verified	Boolean		Parameter apakah surat suara telah diverifikasi oleh <i>Trustee</i>
event_id	Integer	2147483647	Menyimpan id event
status	Varchar	20	Menyimpan status hasil verifikasi
r_x	Varchar	66	Menyimpan r pada titik x dari tabel signing model
r_y	Varchar	66	Menyimpan r pada titik y dari tabel signing model
s_a_x	Varchar	66	Menyimpan hasil perhitungan s' pada titik x
s_a_y	Varchar	66	Menyimpan hasil perhitungan s' pada titik y
m_a	Varchar	66	Menyimpan hasil perhitungan m'
K_x	Varchar	66	Menyimpan K pada titik x dari tabel suara

K_y	Varchar	66	Menyimpan K pada titik y dari tabel suara
M_kP_x	Varchar	66	Menyimpan C pada titik x dari tabel suara
M_kP_y	Varchar	66	Menyimpan C pada titik y dari tabel suara



Gambar 3. 19 Relasi Antar Tabel

3.2.7 Perancangan *User interface*

User interface merupakan sebuah rancangan antarmuka untuk menunjukkan seperti apa bentuk implementasi yang akan dibangun. Rancangan ini berbentuk wireframe yang hanya berupa tata letak dari fungsi yang tersedia. *User interface* ini terdiri dari halaman login, halaman registrasi, halaman beranda, halaman *event*

Voter, halaman *event Signer*, halaman *event Trustee*, halaman beranda admin dan halaman menambah *event*.

3.2.7.1 Rancangan Halaman Register

Setiap *User* diharuskan melakukan register jika belum memiliki akun. Pada penelitian ini *User* hanya memasukkan *Username*, *email*, dan password. Lalu *User* dapat menekan tombol register untuk melakukan registrasi.

The image shows a registration form titled "Halaman Registrasi". It contains four input fields and one button, each with a numbered red circle callout:

- 1: Username input field
- 2: Email input field
- 3: Password input field
- 4: Password input field (for confirmation)
- 5: Register button

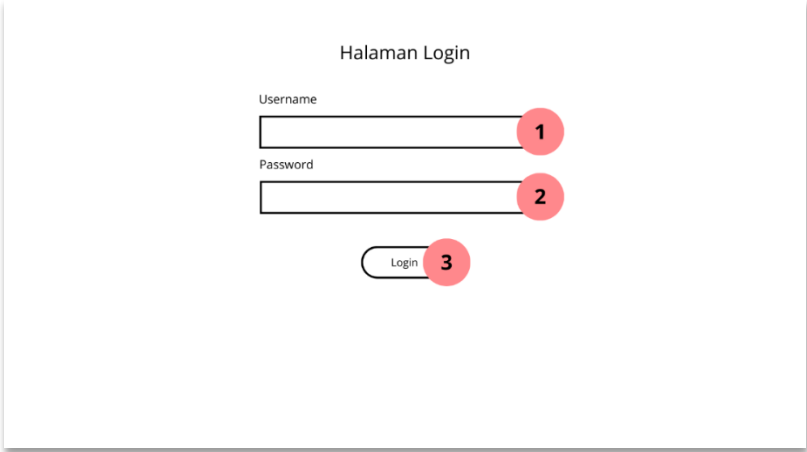
Gambar 3. 20 *User Interface* Halaman Register

Keterangan gambar 3.20 yaitu sebagai berikut.

1. Textbox, digunakan untuk input *Username User*
2. Textbox, digunakan untuk input *email User*
3. Textbox, digunakan untuk input password *User*
4. Textbox, digunakan untuk memastikan input password *User*
5. Button, digunakan untuk mendaftarkan akun

3.2.7.2 Rancangan Halaman Login

Bila *User* telah memiliki akun, maka ia dapat melakukan login untuk mengakses fungsi lainnya pada sistem. *User* cukup memasukkan password dan *Username* ketika login pada sistem.



The image shows a login page titled "Halaman Login". It contains two text input fields: "Username" and "Password". Below these fields is a "Login" button. Red circles with numbers 1, 2, and 3 are placed next to the Username field, Password field, and Login button respectively, indicating their functions as described in the text.

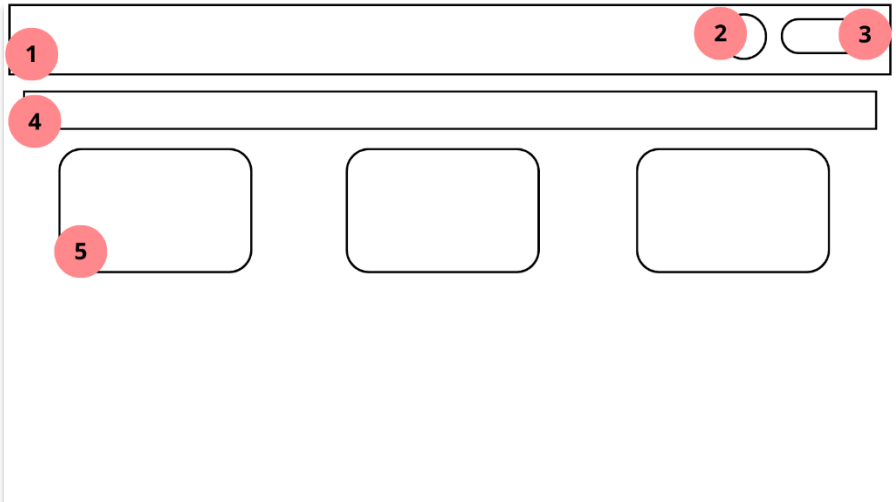
Gambar 3. 21 *User Interface Halaman Login User*

Keterangan gambar 3.21 adalah sebagai berikut.

1. Textbox, digunakan untuk input *Username User*
2. Textbox, digunakan untuk input password *User*
3. Button, digunakan untuk melakukan login ke sistem

3.2.7.3 Rancangan Halaman Beranda

Pada halaman beranda *Voter*, *Signer*, dan *Trustee* secara umum sama. Akan tetapi pada beranda *Voter* terdapat kolom pencarian sedangkan pada halaman *Trustee* terdapat tombol tambahan untuk menambah *event*.



The image shows a dashboard layout. At the top, there is a search bar (labeled 1) and a button (labeled 3). Below the search bar is a horizontal bar (labeled 4). At the bottom, there are three rounded rectangular boxes (labeled 5). Red circles with numbers 1, 2, 3, 4, and 5 are placed next to these elements respectively, indicating their functions as described in the text.

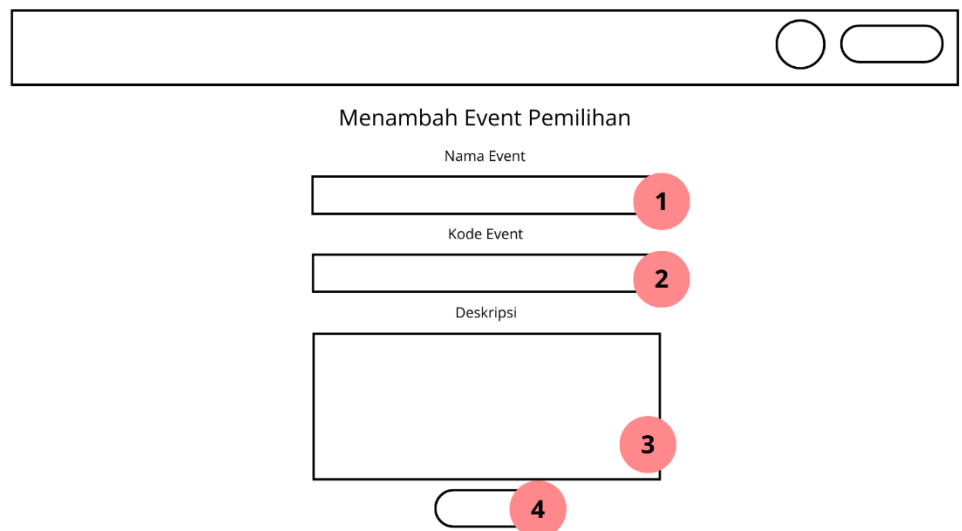
Gambar 3. 22 *UserInterface Beranda*

Keterangan gambar 3.22 adalah sebagai berikut.

1. Navbar, sebagai menu navigasi *User*
2. Button, digunakan untuk menambah *event* yang hanya tersedia pada beranda *Trustee*
3. Button, digunakan untuk logout *User*
4. Textbox, digunakan untuk mencari *event* yang hanya tersedia pada beranda *Voter*
5. Card, digunakan untuk informasi *event* yang mana *User* telah bergabung

3.2.7.3 Rancangan Halaman Menambah *Event*

Halaman menambah *event* hanya dapat diakses oleh *Trustee*. Halaman ini akan muncul jika *User* menekan tombol menambah *event* yang berada di beranda.



The image shows a user interface for adding an event. At the top, there is a horizontal bar containing a circle icon and a rounded rectangle icon. Below this bar, the title "Menambah Event Pemilihan" is centered. The form consists of four labeled input fields, each with a red circle containing a number indicating its function:

- 1. "Nama Event" (Text input field)
- 2. "Kode Event" (Text input field)
- 3. "Deskripsi" (Text area)
- 4. A rounded rectangle button at the bottom right.

Gambar 3. 23 *User Interface Menambah Event*

Keterangan gambar 3.23 adalah sebagai berikut.

1. Textbox, digunakan untuk input nama *event*
2. Textbox, digunakan untuk input kode *event*
3. Textbox, digunakan untuk input deskripsi *event*
4. Button, digunakan untuk menyimpan *event*

3.2.7.4 Rancangan Halaman *Event Voter*

Pada halaman ini *event voter*, *Voter* dapat mengakses fungsinya yaitu proses *blinding* dan *unblinding*.

The diagram illustrates the User Interface for an Event Voter. It features a header bar with a login button on the right. The main content area is divided into three vertical panels. The left panel contains a form with a title field (1), a table with columns 'Nama' and 'Jumlah' (2), a 'Buat Suara' button (3), and a card area (4). The middle panel is a large empty box (5), and the right panel is another large empty box (6).

Gambar 3. 24 *User Interface Event Voter*

Keterangan gambar 3.24 adalah sebagai berikut.

1. Label, digunakan untuk meletakkan judul dan deskripsi *event*
2. Tabel, digunakan untuk menampilkan calon dan jumlah suara yang dihasilkan
3. Button, digunakan untuk membuat suara
4. Card, digunakan untuk *unblinding* jika pesan telah di-*signing*
5. Label, digunakan untuk menampilkan *timing diagram* untuk memperjelas tahap apa yang sedang dilakukan
6. Label, digunakan untuk menampilkan proses yang sedang berjalan

3.2.7.5 Rancangan Halaman *Event Trustee*

Rancangan ini terdiri dari dua halaman yaitu halaman utama di mana *Trustee* dapat melakukan *verification* pada rancangan gambar 3.24 dan halaman untuk pengatur *event* pada rancangan gambar 3.25.

1	
Nama	Jumlah

3 Pengaturan

Waktu	Status	Approve

4

5

6

Gambar 3. 25 *User Interface Halaman Utama Event Trustee*

Keterangan gambar 3.25 adalah sebagai berikut.

1. Label, digunakan untuk meletakkan judul dan deskripsi *event*
2. Tabel, digunakan untuk melihat pilihan dan jumlah pilihannya
3. Button, digunakan untuk mengakses pengaturan *event*
4. Tabel, digunakan untuk melakukan verifikasi suara yang dikirim *Voter*
5. Label, digunakan untuk menampilkan *timing diagram* untuk memperjelas tahap apa yang sedang dilakukan
6. Label, digunakan untuk menampilkan proses yang sedang berjalan

The diagram shows a window with a title bar containing a circle and a rounded rectangle. Below the title bar, there are two buttons: 'Hapus Pilihan' (labeled 1) and 'Tambah Signer' (labeled 2). Below these buttons is a table with the header 'Pilihan' (labeled 3) and three empty rows. Below the table is a button labeled 'Kunci Pilihan' (labeled 4).

Gambar 3. 26 *User Interface Pengaturan Event*

Keterangan gambar 3.26 adalah sebagai berikut.

1. Button, digunakan untuk menambah pilihan pada *event*
2. Button, digunakan untuk menentukan *Signer* pada *event*
3. Tabel, digunakan untuk melihat pilihan yang telah dibuat
4. Button, digunakan untuk mengunci pilihan

3.2.7.6 Rancangan Halaman *Event Signer*

Halaman ini merupakan halaman utama di mana *User* dapat melakukan *signing* terhadap pesan suara yang telah dikirim oleh *Voter*.

The diagram shows a window with a title bar containing a rounded rectangle. Below the title bar, there is a table with three columns: 'Nama', 'Email', and 'Approve' (labeled 1). Below the table are two large empty rectangular areas, labeled 2 and 3.

Gambar 3. 27 *User interface Event Signer*

Keterangan gambar 3.27 adalah sebagai berikut.

1. Tabel, digunakan untuk *signing* pesan dari *Voter*
2. Label, digunakan untuk menampilkan *timing diagram* untuk memperjelas tahap apa yang sedang dilakukan
3. Label, digunakan untuk menampilkan proses yang sedang berjalan

3.2.7.7 Rancangan Halaman Beranda Admin

Akun *Trustee* hanya dapat dibuat oleh admin. Ketika admin menambah akun *Trustee*, halaman yang muncul adalah halaman register yang terdapat pada gambar 3.16. Pada halaman beranda admin hanya terdapat tombol untuk menambah akun *Trustee* dan menampilkan seluruh akun *Trustee* yang pernah dibuat.

Username	Email

Gambar 3. 28 *User interface* Halaman Beranda Admin

Keterangan gambar 3.28 adalah sebagai berikut.

1. Button, digunakan untuk menambah akun *Trustee*
2. Tabel, digunakan untuk melihat akun *Trustee* yang tersedia

BAB IV

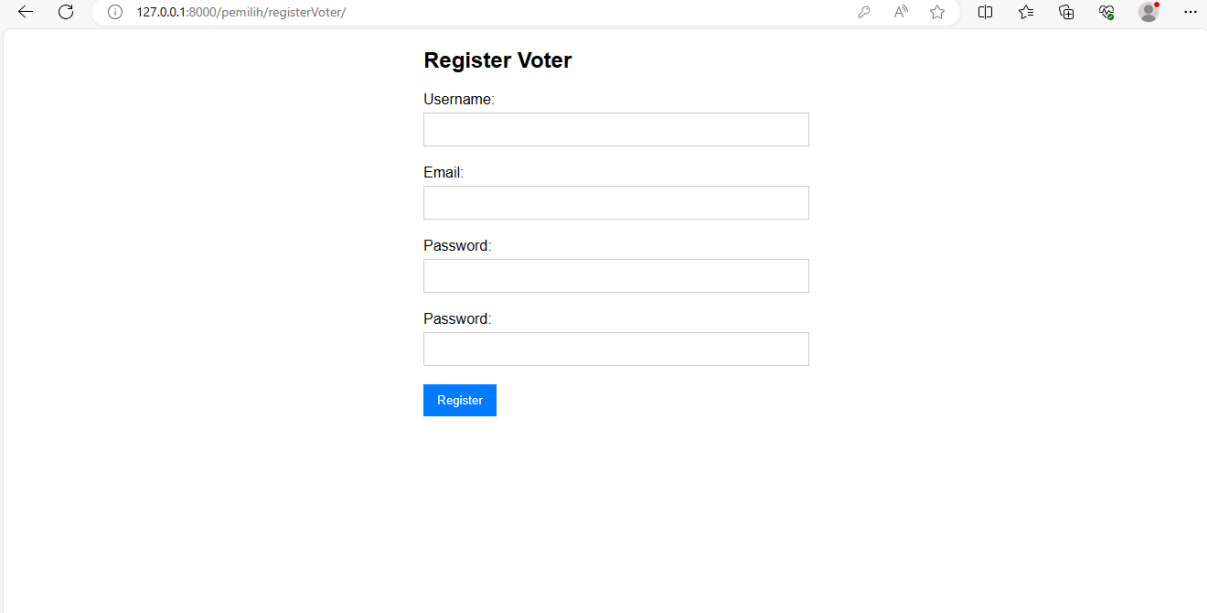
IMPLEMENTASI DAN PENGUJIAN SISTEM

4.1 Implementasi

Algoritma diimplementasikan dalam sebuah web yang dibuat dalam Bahasa Pemrograman Python juga menggunakan *framework* Django. Pada implementasinya, kurva elips yang digunakan adalah kurva P-256 yang direkomendasikan oleh NIST (*National Institute of Standards and Technology*). Seperti yang sudah dibuat pada *User interface*, implementasi sistem ini dibangun atas beberapa halaman yaitu halaman login, halaman registrasi, halaman beranda, halaman menambah *event*, halaman *event Trustee*, halaman *event Voter*, dan halaman *event Signer*.

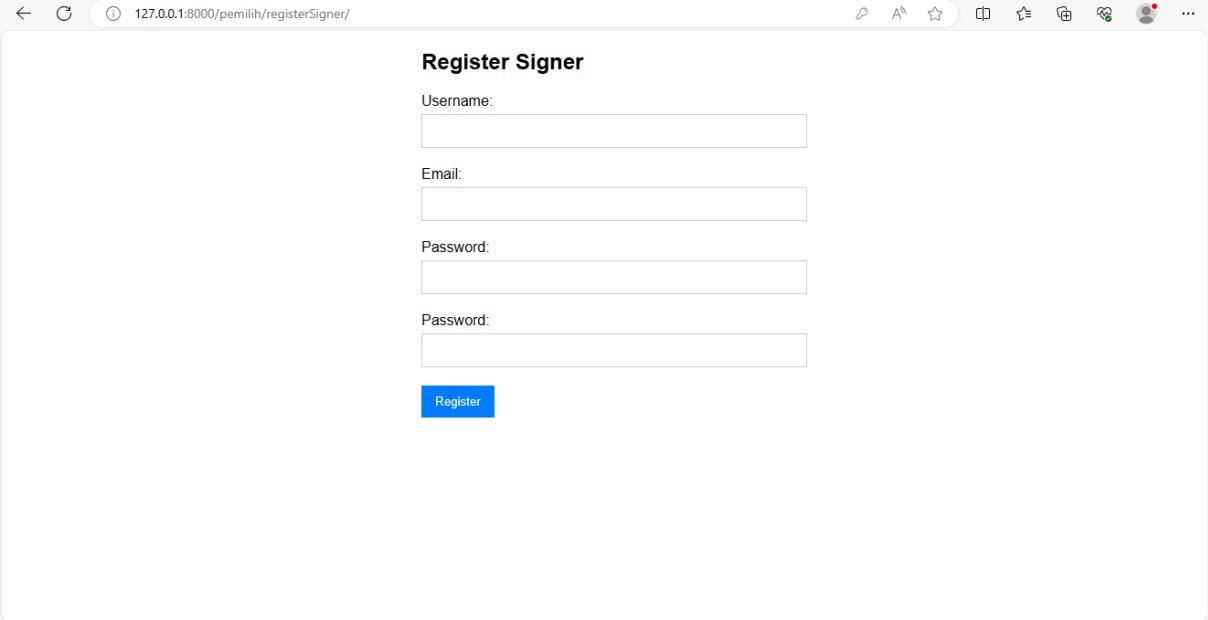
4.1.1 Halaman Registrasi

Sebelum dapat mengakses akun, *user* diharuskan untuk mengakses halaman registrasi terlebih dahulu. Halaman ini terdiri dari tiga URL yaitu, pertama untuk register akun *Voter* yang terlihat pada gambar 4.1, kedua untuk register akun *Signer* yang terlihat pada gambar 4.2, dan terakhir register akun *Trustee* yang terlihat pada gambar 4.3.



The screenshot shows a web browser window with the address bar displaying "127.0.0.1:8000/pemilih/registerVoter/". The page title is "Register Voter". The form contains four input fields: "Username:", "Email:", "Password:", and "Password:". Below the second password field is a blue "Register" button.

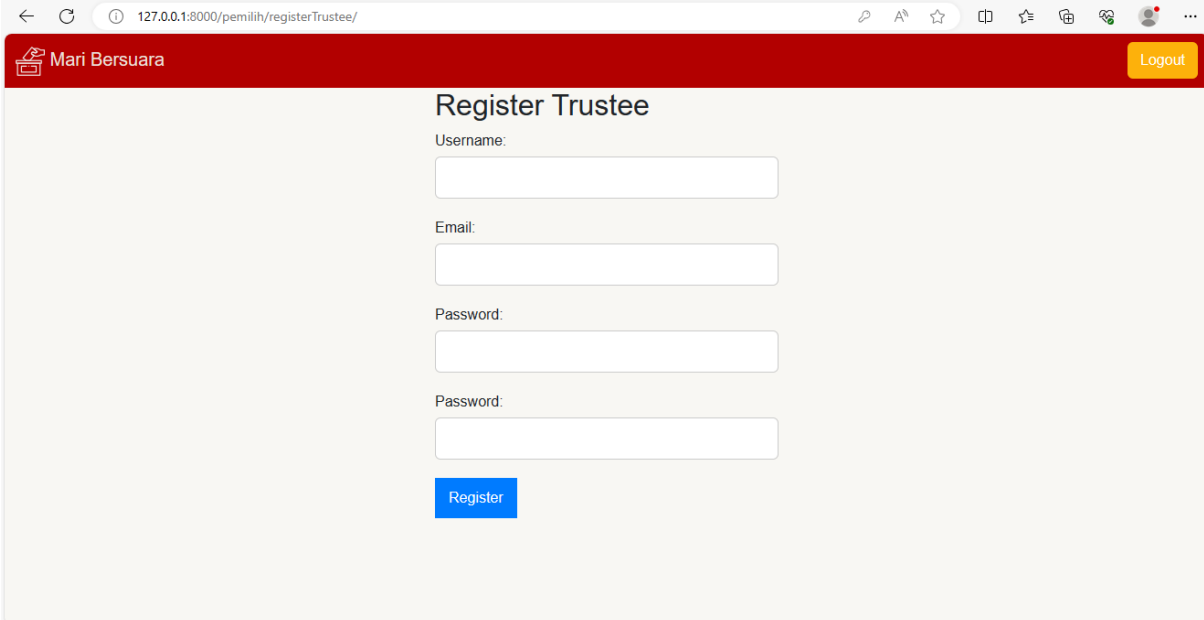
Gambar 4. 1 *Halaman Registrasi Voter*



A screenshot of a web browser displaying the "Register Signer" page. The browser's address bar shows the URL "127.0.0.1:8000/pemilih/registerSigner/". The page has a white background and contains the following elements:

- Register Signer**: A heading centered at the top of the form area.
- Username:** A text label followed by a white input field.
- Email:** A text label followed by a white input field.
- Password:** A text label followed by a white input field.
- Password:** A second text label followed by a second white input field.
- Register**: A blue button with white text located below the password fields.

Gambar 4. 2 *Halaman Registrasi Signer*



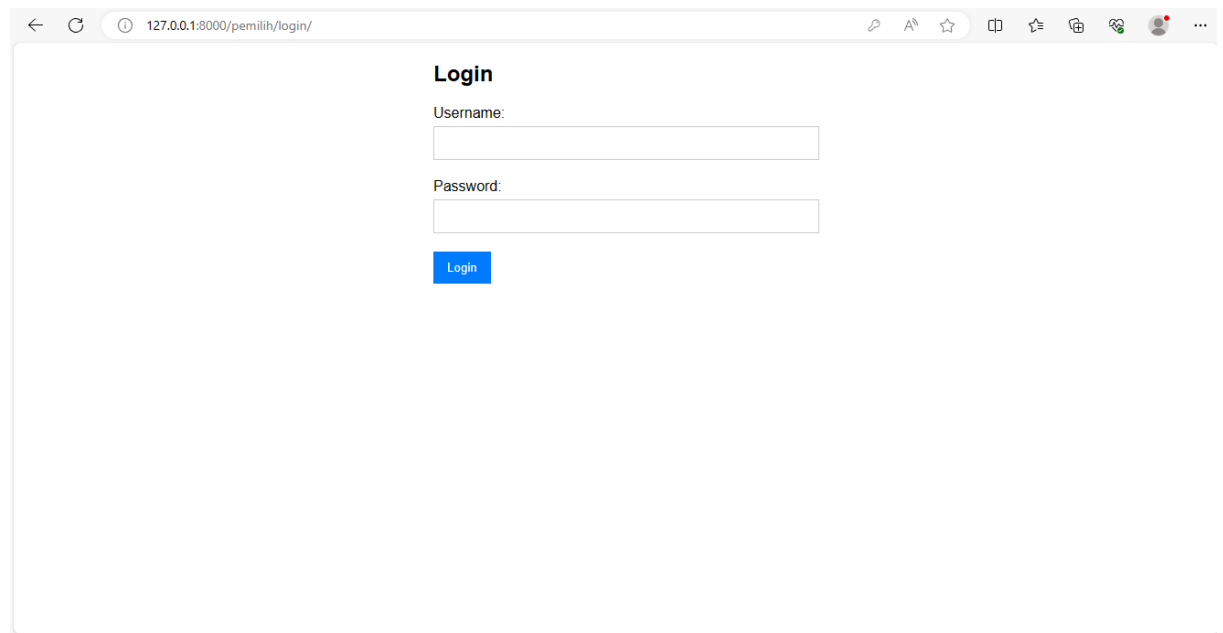
A screenshot of a web browser displaying the "Register Trustee" page. The browser's address bar shows the URL "127.0.0.1:8000/pemilih/registerTrustee/". The page features a red header bar with the text "Mari Bersuara" and a "Logout" button. The main content area has a light beige background and includes the following elements:

- Register Trustee**: A heading centered at the top of the form area.
- Username:** A text label followed by a white input field.
- Email:** A text label followed by a white input field.
- Password:** A text label followed by a white input field.
- Password:** A second text label followed by a second white input field.
- Register**: A blue button with white text located below the password fields.

Gambar 4. 3 *Halaman Register Trustee*

4.1.2 Halaman Login

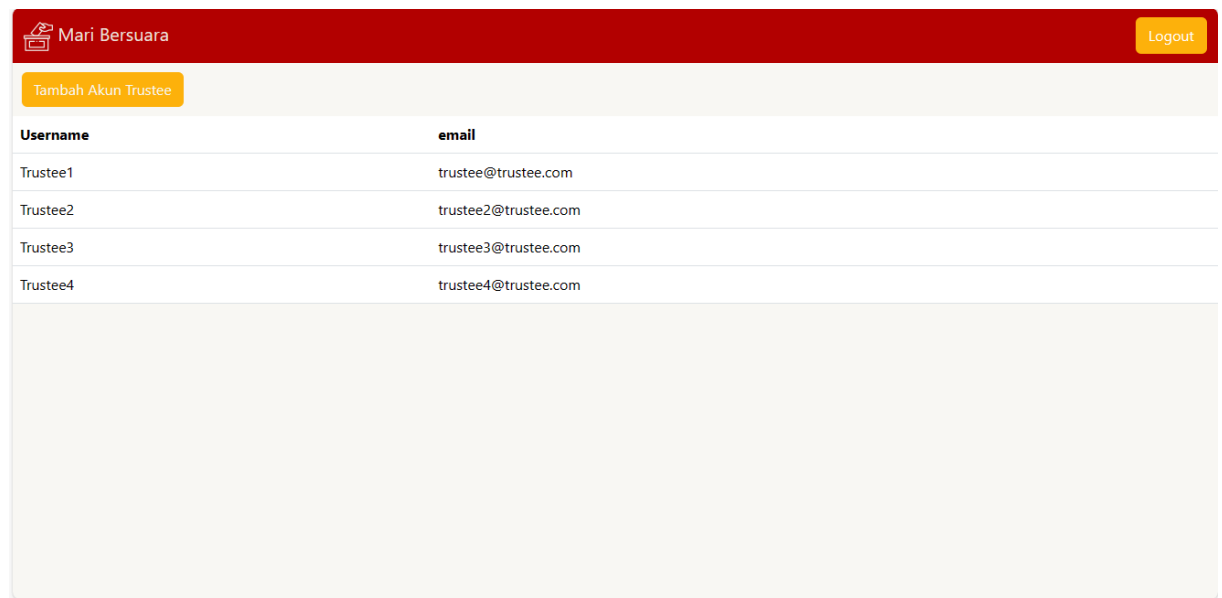
Setelah memiliki akun, *user* dapat melakukan login. Setiap *User* memiliki halaman login yang sama. *User* hanya menginputkan *username* lalu *password* yang dimiliki untuk melakukan masuk. Jika salah satu dari *username* atau *password* salah, maka *user* tidak dapat mengakses halaman selanjutnya dan dikembalikan ke halaman login. Halaman login terlihat pada gambar 4.4.



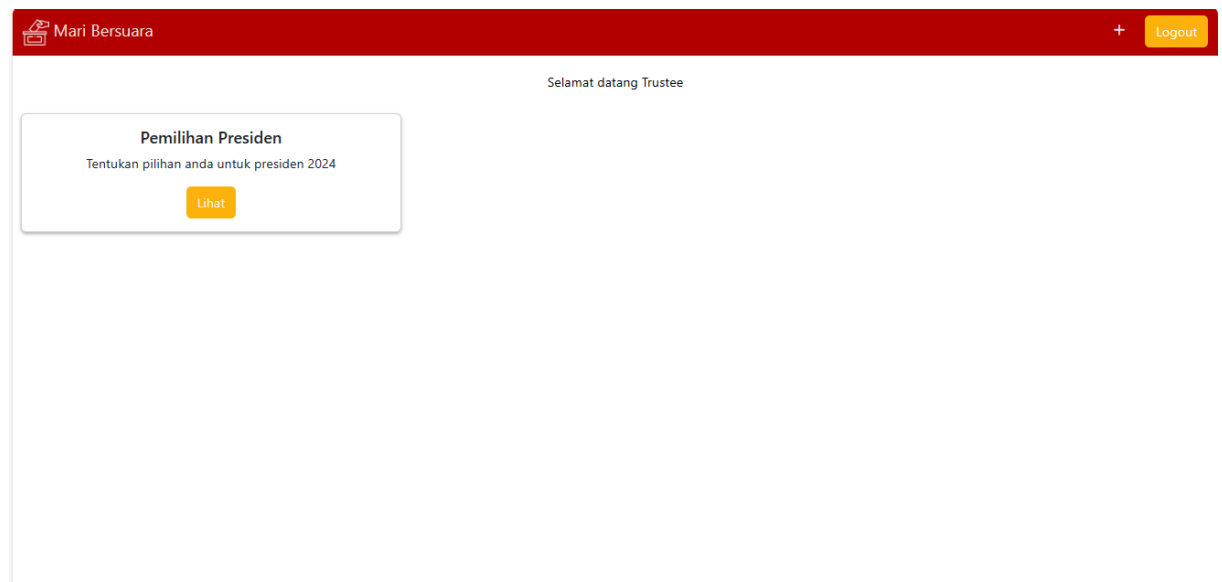
Gambar 4. 4 Halaman *Login*

4.1.3 Halaman Beranda

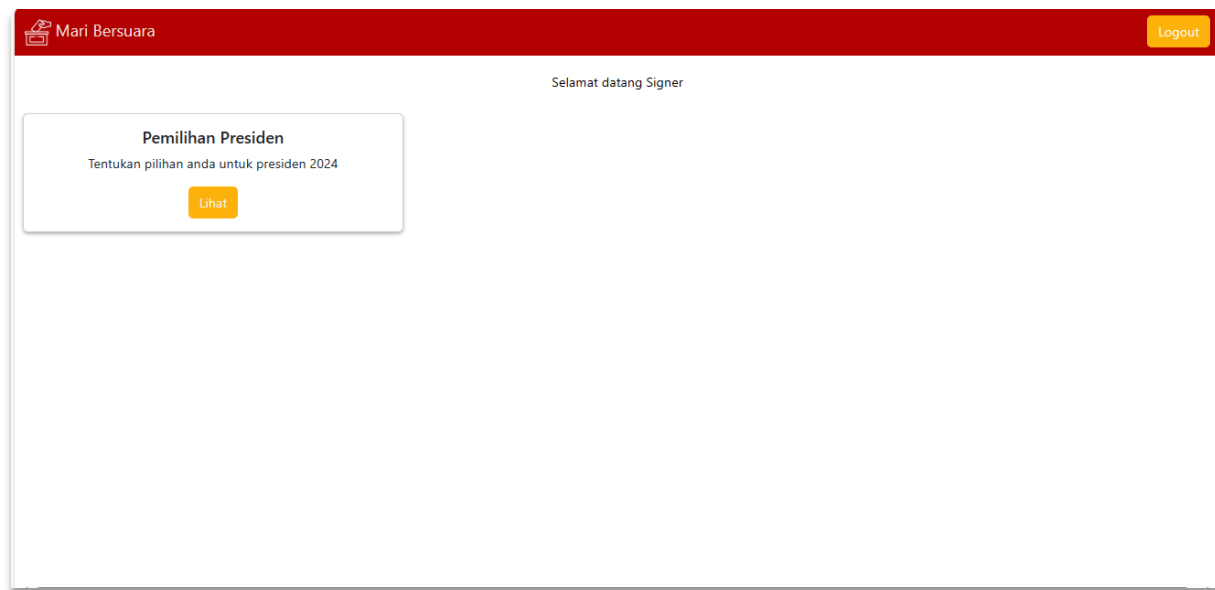
Setiap role *user* memiliki beranda yang sedikit berbeda. Perbedaannya terletak pada halaman beranda *Voter* terdapat kolom untuk mencari *event* yang akan diikuti, pada halaman beranda *Trustee* yang memiliki tombol untuk menambah *event* pada navbarnya, pada halaman beranda admin terdapat tombol untuk menambah akun *Trustee*. Beranda admin ada pada gambar 4.5, beranda *Trustee* ada pada gambar 4.6, beranda *Signer* ada pada gambar 4.7, dan beranda *Voter* ada pada gambar 4.8.



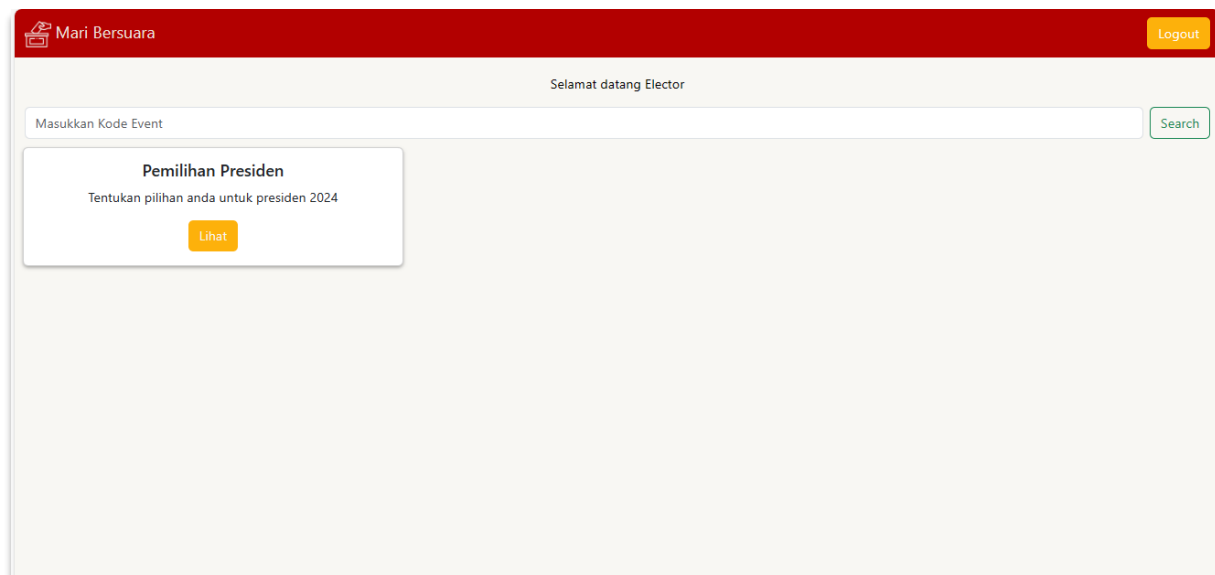
Gambar 4. 5 Halaman Beranda Admin



Gambar 4. 6 Halaman Beranda *Trustee*



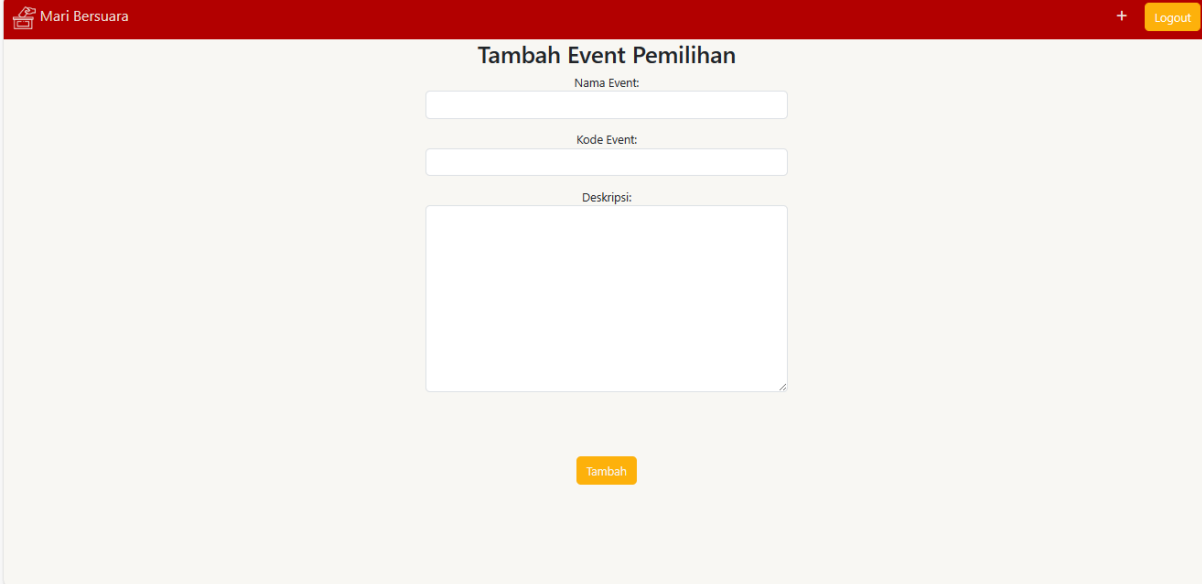
Gambar 4. 7 Halaman Beranda *Signer*



Gambar 4. 8 Halaman Beranda *Voter*

4.1.4 Halaman Menambah *Event* Pemilihan

Setelah *Trustee* menekan tombol menambah *event* yang tersedia pada beranda *Trustee*. Kode *event* bersifat unik agar *Voter* dapat mencari *event* berdasarkan kode yang telah dibuat. Halaman menambah *event* dapat dilihat di gambar 4.9

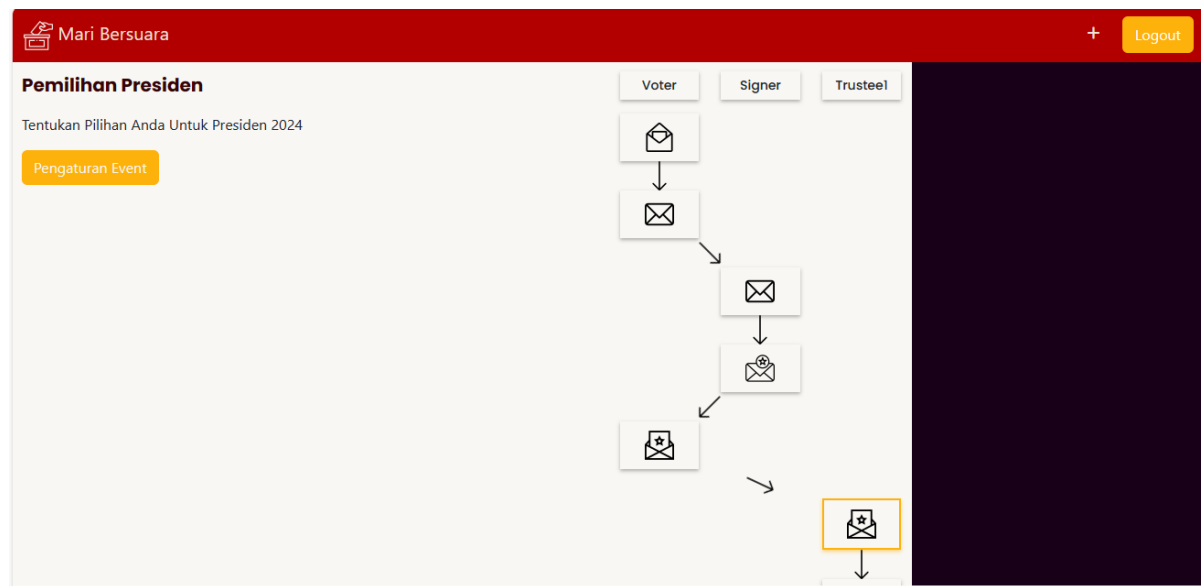


The screenshot shows a web application interface for adding an election event. At the top, there is a red header bar with the text 'Mari Bersuara' on the left and a '+ Logout' button on the right. The main content area has a light beige background. In the center, the title 'Tambah Event Pemilihan' is displayed. Below the title are three input fields: 'Nama Event:' (a single-line text box), 'Kode Event:' (a single-line text box), and 'Deskripsi:' (a multi-line text area). At the bottom center of the form is an orange button labeled 'Tambah'.

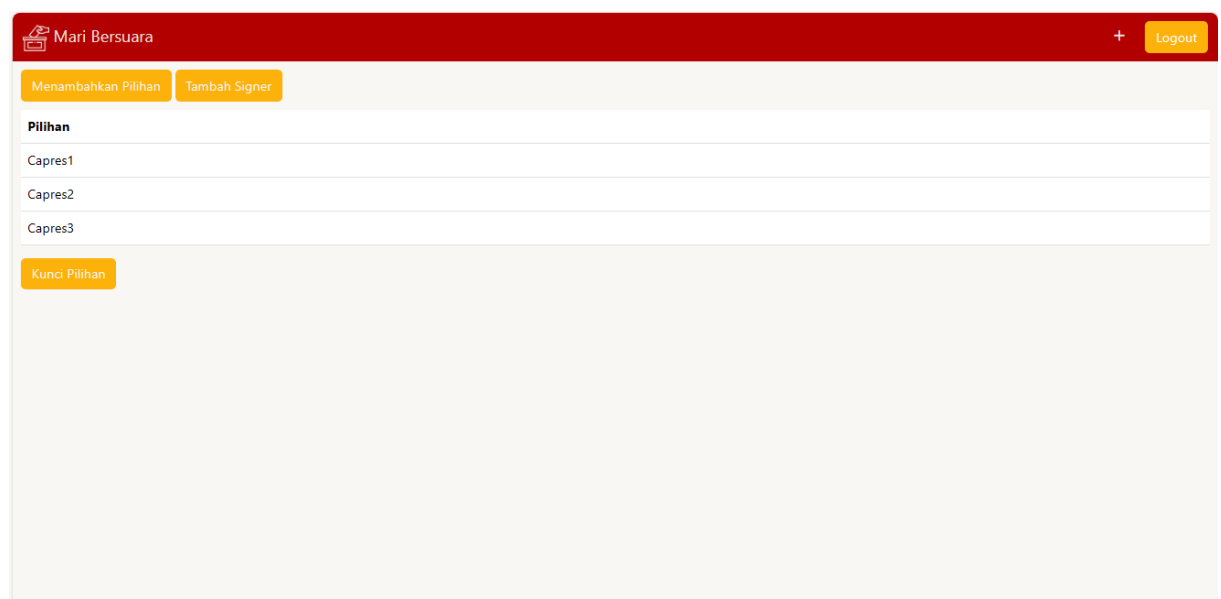
Gambar 4. 9 Halaman Menambah *Event* Pemilihan

4.1.5 Halaman *Event Trustee*

Halaman ini terdiri dari dua halaman yaitu halaman utama *event Trustee* dan halaman pengaturan *event*. Halaman utama pada gambar 4.10 berfungsi untuk *Trustee* dapat melakukan verifikasi, sedangkan halaman pengaturan *event* pada gambar 4.11 berfungsi untuk *Trustee* menambah pilihan dan mengatur *Signer* untuk *event* tersebut.



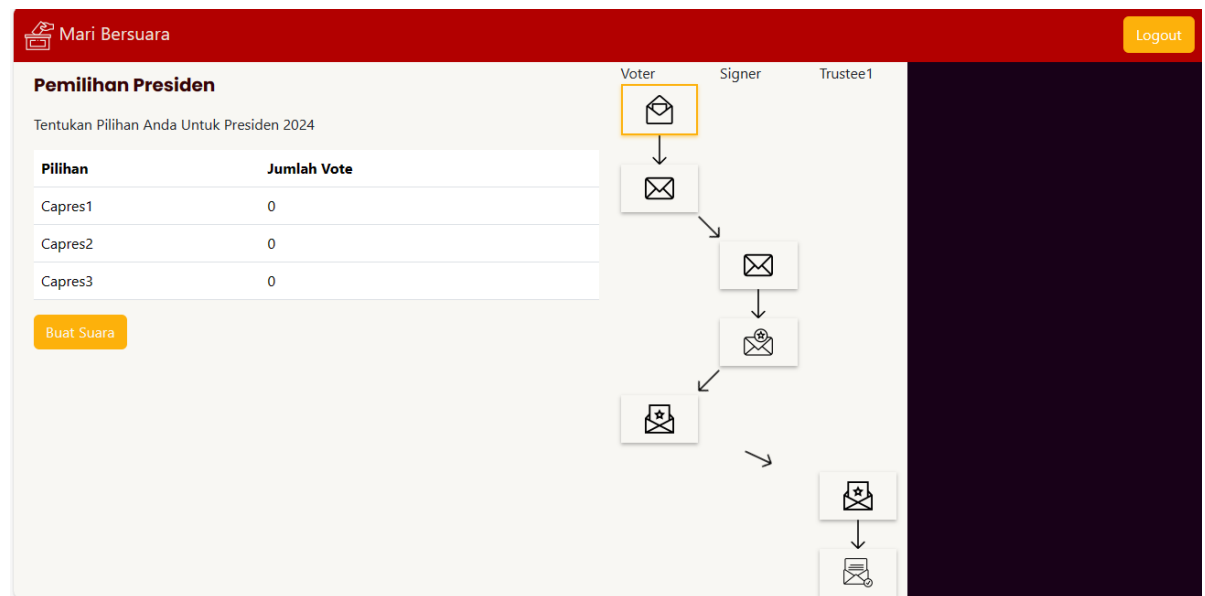
Gambar 4. 10 Halaman Utama *Event Trustee*



Gambar 4. 11 Halaman Pengaturan *Event*

4.1.6 Halaman *Event Voter*

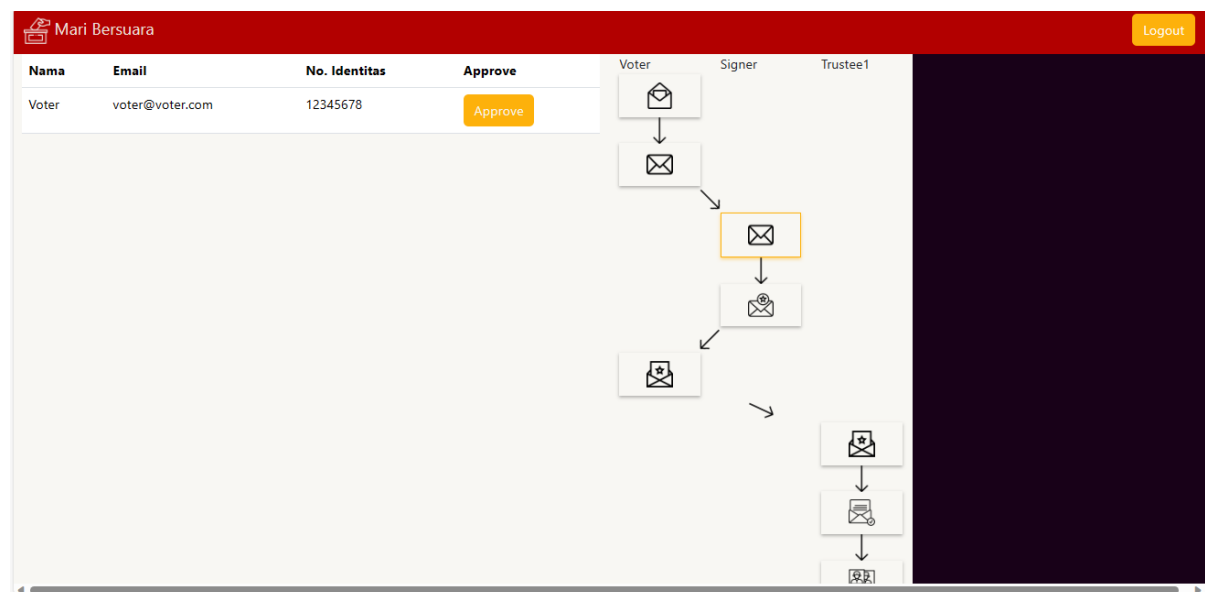
Halaman *event Voter* tempat di mana *Voter* melakukan *blinding* dan *unblinding*. Jika pilihan pada *event* tersebut telah dikunci oleh *Trustee* maka *Voter* dapat membuat suara. Gambar halaman *event Voter* dapat terlihat pada gambar 4.12.



Gambar 4. 12 Halaman *Event Voter*

4.1.7 Halaman *Event Signer*

Halaman *Signer* terdapat tabel yang berisi nama dan *email Voter* yang telah memberikan suara pada *event* yang dipilih. Jika *Signer* ingin menandatangani suara tersebut, *Signer* dapat menekan tombol approve dan langsung mengirim kembali suara ke *Voter*. Gambar halaman *Signer* ada pada gambar 4.13.



Gambar 4. 13 Halaman *Event Signer*

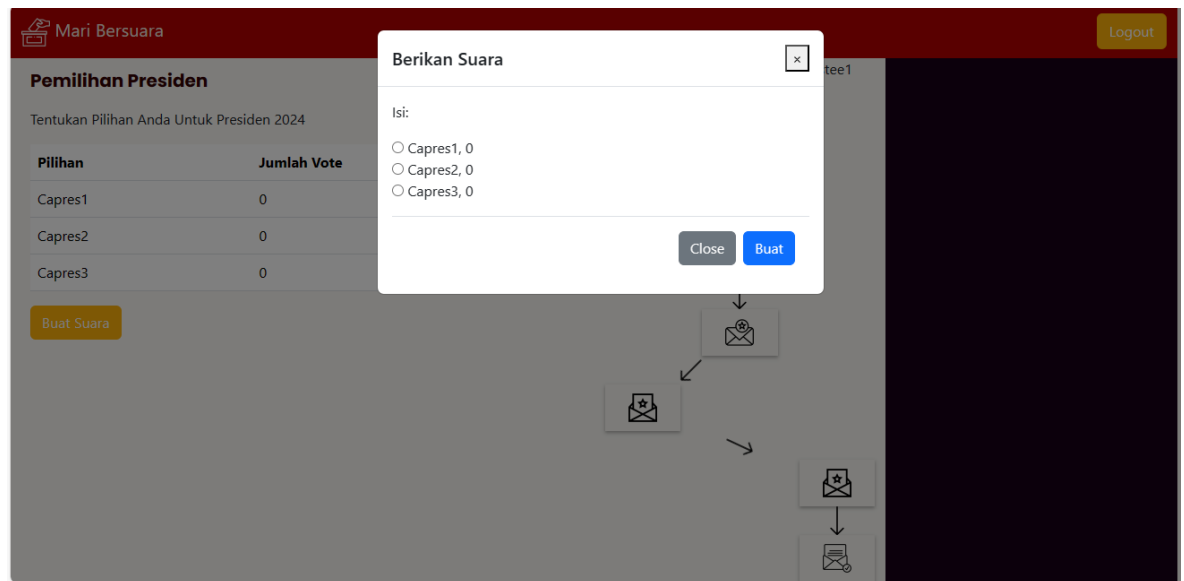
4.2 Pengujian Sistem Satu Voter

Tahap pengujian sistem ini menguji apakah sistem *e-voting* dapat dilakukan dengan memadukan skema *Blind Signature* dengan *Elliptic Curve Cryptography*. Kriteria dari pengujian sistem adalah sebagai berikut.

1. *Voter* dapat membuat surat suara
2. *Signer* dapat melakukan *signing* terhadap surat yang dikirim *Voter*
3. *Voter* dapat melakukan *unblinding* dari surat suara yang telah ditandatangani
4. *Trustee* dapat melakukan verifikasi terhadap surat suara yang telah di-*unblinding*
5. Hasil suara yang tercatat adalah pilihan yang telah dipilih elektor sebelumnya

4.2.1 Pengujian Membuat Suara

Pada tahap ini akan menguji apakah *User* dapat membuat suara dan melakukan *blinding* terhadap surat suaranya. Pada gambar 4.14 terlihat bahwa *User* mengisi surat suaranya dengan pilihan “Capres1”. Lalu pada gambar 4.15 terlihat dari *timing diagram* *User* melakukan proses *blinding* menggunakan kunci privat dan publiknya yang terlihat pada kolom paling kanan dan menghasilkan surat yang telah di-*blinding* (alpha). Selain itu *Voter* juga menghasilkan *K* dan *C* untuk dikirim ke *Trustee* nantinya jika pesan telah di-*signing*.



Gambar 4. 14 Memberikan Suara

Logout

Pemilihan Presiden

Tentukan Pilihan Anda Untuk Presiden 2024

Pilihan	Jumlah Vote
Capres1	1
Capres2	0
Capres3	0

Voter

Signer

Trustee1

```

--Blinding--
Private Key Voter :
0x8ee2328a8cf70da047bbf6ab314b734653b49ad64bdae07826052ea8b4ce104c

Public Key Voter :
0xf0eef8ed1b6eb9da3b0772c9f8126d7f8f46a7c5fc216d87bf7e8c54b2eb4942
0xbc4c9d3013001ca0fb99298939e9d166a96234c1b66b0ef72bc49ee718b711

Alpha yang dihasilkan :
0xfc771c3a90d5c962e7195ca12ee7845e3785670f95feb2bc92a57df0d51fd568
0x304ecb607d7e0e321e80a41525f1025c2a7f398c6e27b1cb569ee2966c56ede

isi suara : Capres1

isi hash
24812844050003666004672578584256034070493047553046031070438546878447856326089

K
0x9ee06423065f6e955d157afdb317e34f16390eda05478b0fa3d6cf188b678bc2
0xacc57e85589f10f450f8cdecee9540740e5ad00c3a8aee25da5fa36ea41b9e65

C
0x10a663e5fb36d167d0c2aadaad3c9e71fe96c1b103ba081406c95bc4cfdc45c1
0x5d22fcd728182802194c5ef6dd8646272cbe28d82dcd08451d3ecd8397e4733c

```

Gambar 4. 15 Hasil *Blinding*

```

--Blinding--
Private Key Voter
0x8ee2328a8cf70da047bbf6ab314b734653b49ad64bdae07826052ea8b4ce104c

Public key Voter
0xf0eef8ed1b6eb9da3b0772c9f8126d7f8f46a7c5fc216d87bf7e8c54b2eb4942
0xbc4c9d3013001ca0fb99298939e9d166a96234c1b66b0ef72bc49ee718b711

Alpha yang dihasilkan
0xfc771c3a90d5c962e7195ca12ee7845e3785670f95feb2bc92a57df0d51fd568
0x304ecb607d7e0e321e80a41525f1025c2a7f398c6e27b1cb569ee2966c56ede

isi suara : Capres1

isi hash
24812844050003666004672578584256034070493047553046031070438546878447856326089

K
0x9ee06423065f6e955d157afdb317e34f16390eda05478b0fa3d6cf188b678bc2
0xacc57e85589f10f450f8cdecee9540740e5ad00c3a8aee25da5fa36ea41b9e65

C
0x10a663e5fb36d167d0c2aadaad3c9e71fe96c1b103ba081406c95bc4cfdc45c1
0x5d22fcd728182802194c5ef6dd8646272cbe28d82dcd08451d3ecd8397e4733c

```

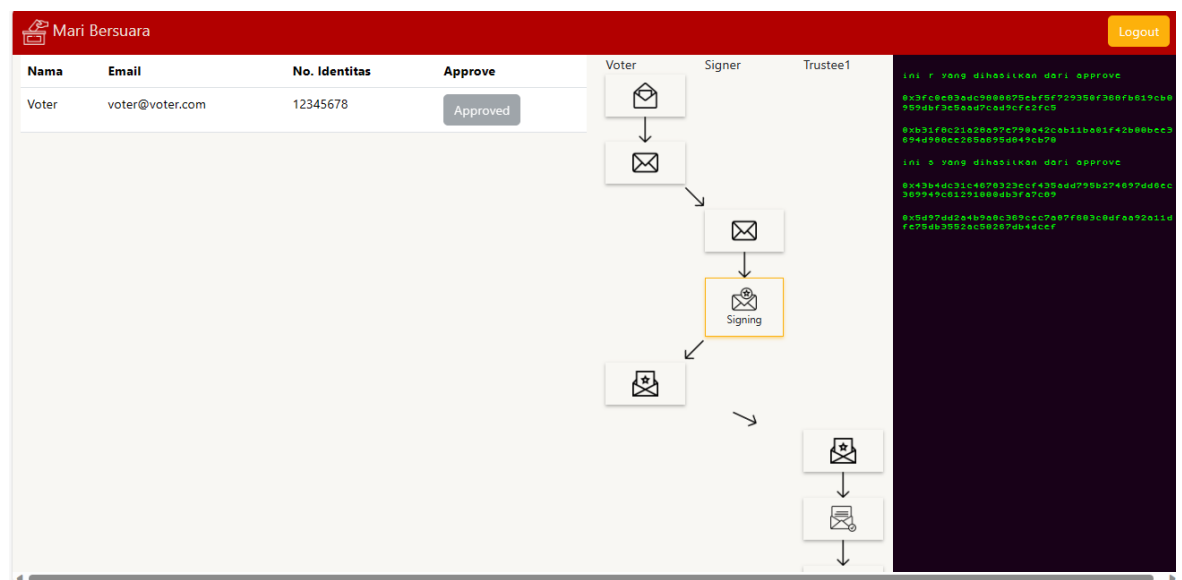
Gambar 4. 16 Detail *Blinding*

Penjelasan gambar 4.16 adalah sebagai berikut.

1. Private Key *Voter* merupakan kunci privat yang telah dibangkitkan ketika proses registrasi *Voter* yang diubah ke dalam hexadesimal
2. Public Key *Voter* merupakan kunci publik yang telah dihitung oleh sistem dan diubah ke dalam hexadesimal
3. Alpha merupakan hasil dari blinding yang akan dikirim ke *Signer*
4. Isi *hash* merupakan isi suara *Voter* yang diubah menggunakan SHA-256
5. K dan C merupakan elemen perhitungan yang dihitung untuk mengenkripsi pesan

4.2.2 Pengujian *Signing*

Setelah *Voter* mengirim suara, *Signer* akan melihat pesan suara pada halaman *event Signer*. Pada gambar 4.17, *Signer* melakukan *signing* terhadap surat suara milik *Voter*. Ketika *Signer* menekan tombol “approve”, maka *timing diagram* akan langdung menunjuk pada proses *signing* di mana *Signer* akan menghasilkan “s” dan “r” yang dapat terlihat pada kolom paling kanan. Lalu pesan akan dikembalikan ke *Voter*.



Gambar 4. 17 Pengujian *Signing*

```
--Signing--
Ini r yang dihasilkan dari approve
0x3fc0e83adc9808675ebf5f729350f368fb619cb0959dbf3e5aad7cad9cfe2fc5
0xb31f8c21a28a97e790a42cab11ba01f42b00bee3694d988ee265a695d849cb70

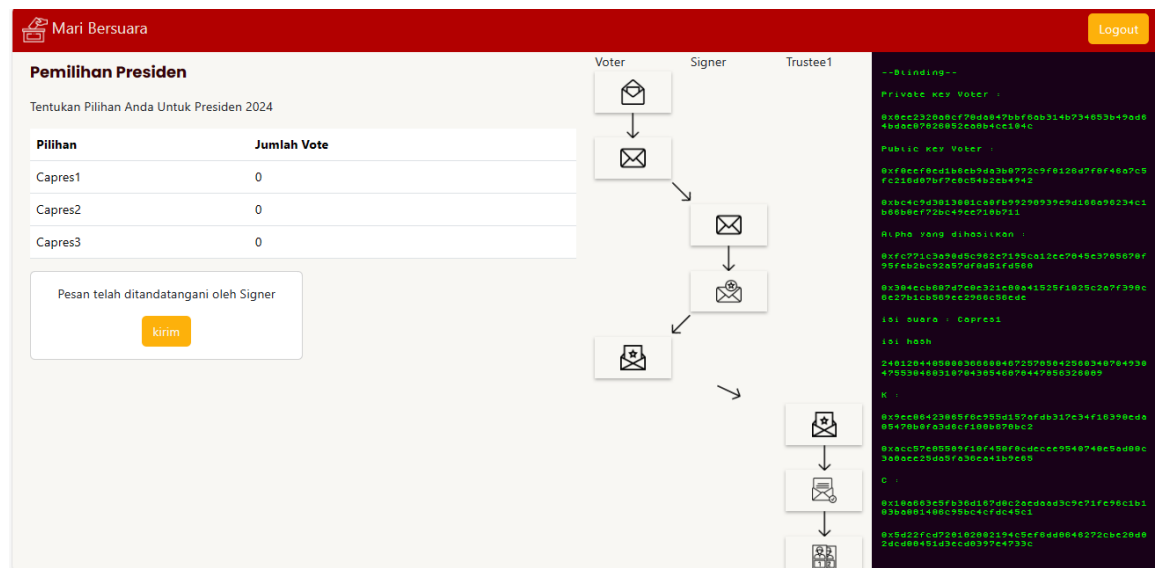
Ini s yang dihasilkan dari approve
0x43b4dc31c4678323ecf435add795b274697dd6ec369949c6129100db3fa7c89
0x5d97dd2a4b9a8c369cec7a87f603c0dfa9a211dfe75db3552ac50267db4dcef
```

Gambar 4. 18 Detail *Signing*

Ketika *Signer* klik tombol *Approve* maka sistem langsung melakukan proses *signing*. Pada gambar 4.18 terlihat bahwa sistem menghitung *r* dan *s* sesuai dengan *flowchart* yang tersedia pada gambar 3.15. Kedua variabel ini disimpan dalam database dan ditampilkan dalam bentuk hexadesimal.

4.2.3 Pengujian *Unblinding*

Kembali ke halaman *Voter*, di mana *Voter* telah menerima pesan yang telah ditandatangani yang ada pada gambar 4.17. Ketika *Voter* ingin mengirim surat suara tersebut ke *Trustee* maka *Voter* akan melakukan proses *unblinding* ada pada gambar 4.18. Proses *unblinding* akan menghasilkan “ m ’ ” dan “ s ’ ” yang terlihat pada kolom paling kanan pada gambar 4.18.



Gambar 4. 19 Voter Menerima Surat yang di-signing

```
--Unblinding--
r
0x3fc0e83adc9808675ebf5f729350f368fb619cb0959dbf3e5aad7cad9cfe2fc5
0xb31f8c21a28a97e790a42cab11ba01f42b00bee3694d988ee265a695d849cb70

s'
0xc578d274a97d8b895335b2970c3346926f1e63939167ac35729e5acbaabaf87f
0x974592c8de060b9243bad9bde8fc35b5349817105ad18fc4a34c11f0d1314800

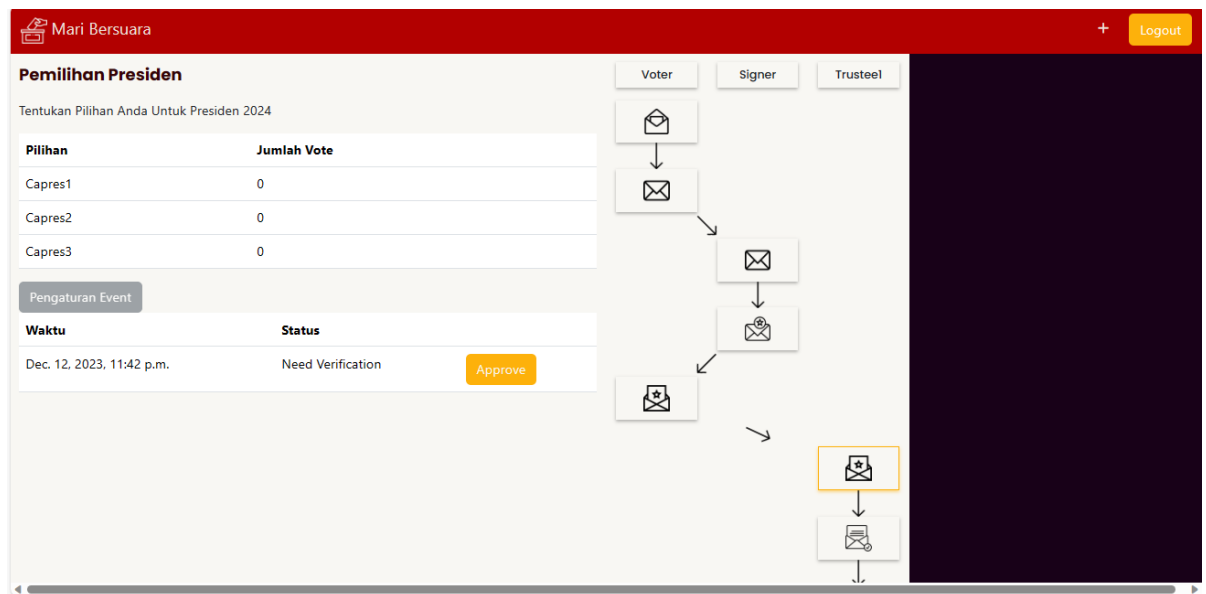
m'
0x1116d76d81cca1ac1a667cb63c2fec7fae6c09efc66e78f55502c521b71f252ba
8d347779add13dc8d25c3f973c3cf63b94494fef1ed55b30e77d75e01e35a73773c
9454132f0b8355de0e1a8e808f900d9389446198ab28176223930edbfff64|
```

Gambar 4. 20 Hasil *Unblinding*

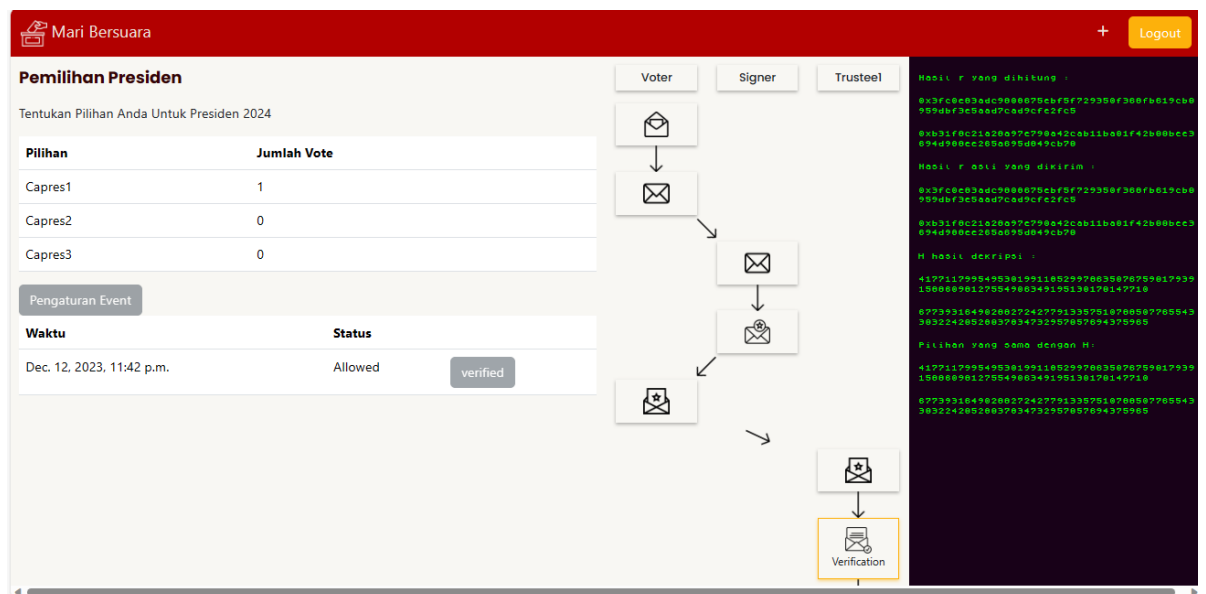
Ketika *Voter* menekan tombol kirim, sistem lalu melakukan unblinding dan menghasilkan s' yang merupakan hasil dekripsi s dan m' . Semua variabel ini akan dikirim kepada *Trustee*. Pada gambar 4.20 variabel ditampilkan dalam bentuk hexadesimal.

4.2.4 Pengujian *Verification*

Ketika *Trustee* telah menerima pesan dari *Voter* maka *Trustee* dapat melakukan *verification*. Sebelum diverifikasi, status surat suara tersebut adalah “*need verification*” seperti yang ada pada gambar 4.21. Jika telah diverifikasi dan hasilnya benar maka akan menghasilkan status “*allowed*” dan apabila hasil verifikasi salah status akan berubah menjadi “*Rejected*”. Ketika status “*allowed*” maka sistem akan menyamakan isi jawaban M yang berupa *hash* dengan seluruh pilihan yang diubah juga menjadi *hash*. Jika ada yang sama maka hasil tersebut yang akan dicatat. Proses hasil ada pada gambar 4.22.



Gambar 4. 21 Sebelum Verifikasi



Gambar 4. 22 Hasil Setelah Verifikasi

```
--Verification--
Hasil r yang dihitung
0x3fc0e83adc9808675ebf5f729350f368fb619cb0959dbf3e5aad7cad9cfe2fc5
0xb31f8c21a28a97e790a42cab11ba01f42b00bee3694d988ee265a695d849cb70

Hasil r asli yang dikirim
0x3fc0e83adc9808675ebf5f729350f368fb619cb0959dbf3e5aad7cad9cfe2fc5
0xb31f8c21a28a97e790a42cab11ba01f42b00bee3694d988ee265a695d849cb70

M hasil Dekripsi
41771179954953019911852997863507675901793915886098127554906349195130170147710
67739316490288272427791335751078850776554330322420528037834732957057694375965

Pilihan yang sama dengan M :
41771179954953019911852997863507675901793915886098127554906349195130170147710
67739316490288272427791335751078850776554330322420528037834732957057694375965
```

Gambar 4. 23 Detail *Verification*

Pada gambar 4.23 sistem akan menyamakan r yang dihitung menggunakan rumus $r = s' - m' \cdot P_s$ apabila r yang dihitung sama dengan r yang dikirim dari *Voter* yang terdapat pada gambar 4.18, maka verifikasi benar dan menghasilkan status “Allowed”. Lalu *Trustee* akan mendekripsi hasil dari K dan C yang dikirim menjadi M dalam bentuk *hash*. Setiap pilihan yang tersedia akan diubah ke dalam bentuk *hash* juga dan dikalikan dengan Base menjadi sebuah titik. Lalu M akan disamakan dengan pilihan yang sudah berbentuk titik tersebut. Jika M ada yang sama dengan pilihan maka pilihan tersebut yang akan dicatat. Hasil terbukti benar dengan hasil yang bertambah adalah Capres1 sesuai dengan pilihan *Voter* pada gambar 4.16.

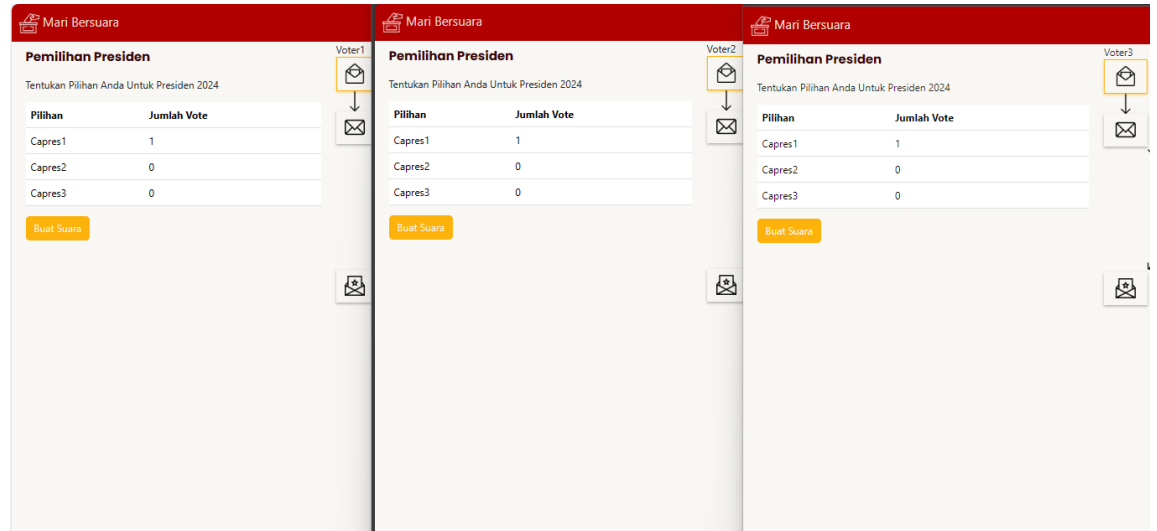
4.3 Pengujian Sistem Beberapa *Voter*

Tujuan pengujian ini untuk melihat bagaimana *Trustee* dan *Signer* menangani bagaimana jika *Voter* lebih dari satu. Pengujian ini masih menggunakan event yang sama pada pengujian sebelumnya.

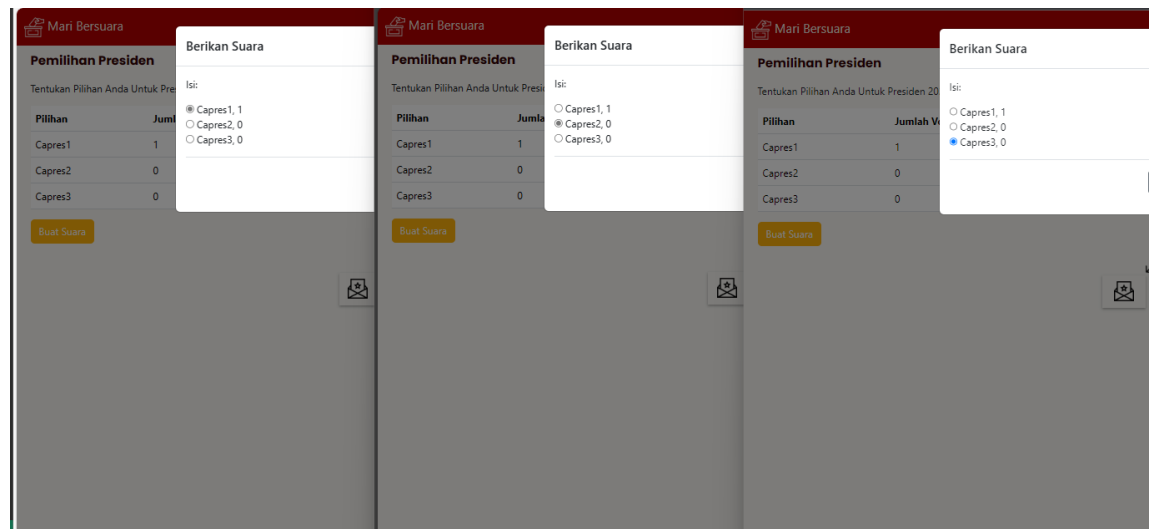
4.3.1 Pengujian Memberikan Suara

Pada pengujian diberikan contoh tiga orang *Voter* memberikan suaranya. Dapat dilihat di gambar 4.24, tab di sisi kiri adalah *Voter1*, lalu tab yang berada di tengah adalah *Voter2*, dan yang terakhir tab di sisi kanan adalah *Voter3*. Mereka ingin

memberikan suara yang terlihat pada gambar 4.25. Lalu suara akan terkirim kepada *Signer* yang bertugas.



Gambar 4. 24 Beberapa *Voter*

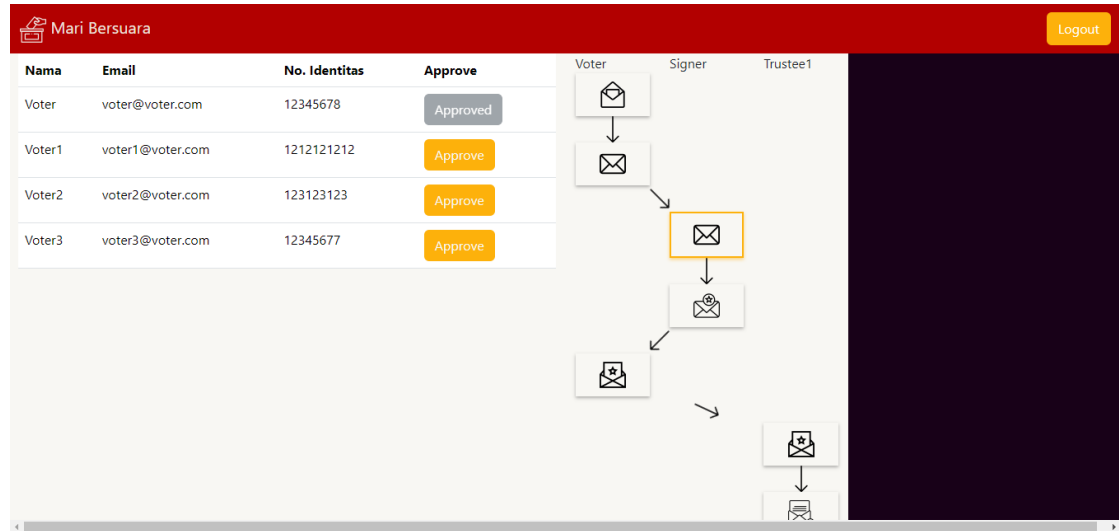


Gambar 4. 25 Beberapa *Voter* Memberikan Suara

4.3.2 Pengujian Signing

Ketika beberapa *Voter* telah membuat suara pada event yang sama, suara mereka akan masuk ke halaman *Signer* seperti yang digambarkan pada gambar

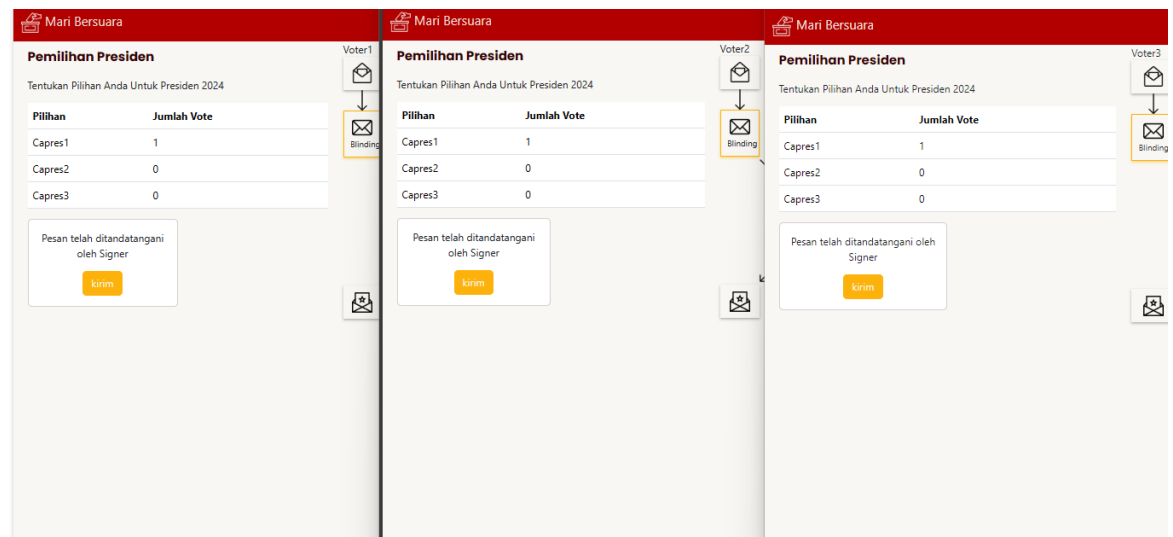
4.26. *Signer* memilih pesan mana yang akan di approve untuk diberikan persetujuannya dan dikembalikan ke *Voter* yang membuat suara.



Gambar 4. 26 Pengujian Signing Beberapa Voter

4.3.3 Pengujian Unblinding

Saat *Signer* telah melakukan *signing* dan dikembalikan kepada *Voter*, maka *Voter* dapat melakukan *unblinding* dan mengirimkannya kepada *Trustee*. Proses ini terdapat pada gambar 4.27.



Gambar 4. 27 Unblinding Voter

4.3.3 Pengujian Verifikasi

Ketika beberapa *Voter* telah mengirimkan surat suaranya kepada *Trustee* pada sebuah event, maka surat tersebut akan muncul pada halaman *Trustee* yang ada pada gambar 4.27. Lalu *Trustee* akan memilih surat suara mana yang akan di verifikasi terlebih dahulu. Saat surat suara yang dipilih selesai di verifikasi, akan muncul hasil suara pada tabel jumlah vote. Di gambar 4.28, diperlihatkan hasil pengujian proses verifikasi. Pengujian ini memberikan hasil yang sama dengan pilihan *Voter* pada gambar 4.25

Pemilihan Presiden

Tentukan Pilihan Anda Untuk Presiden 2024

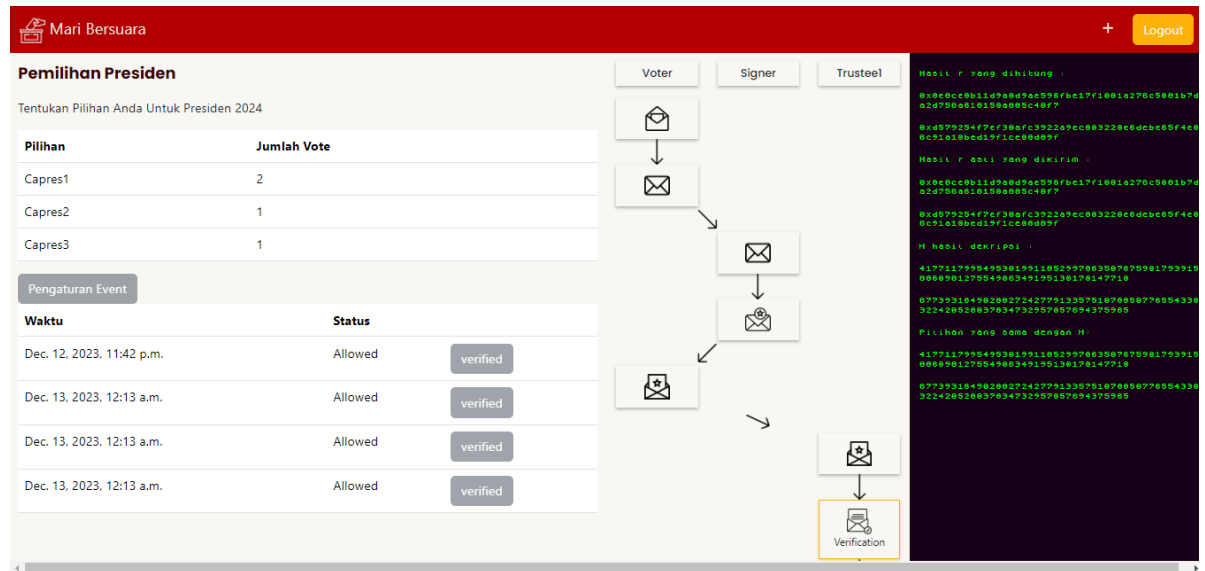
Pilihan	Jumlah Vote
Capres1	1
Capres2	0
Capres3	0

Pengaturan Event

Waktu	Status
Dec. 12. 2023, 11:42 p.m.	Allowed verified
Dec. 13. 2023, 12:13 a.m.	Need Verification Approve
Dec. 13. 2023, 12:13 a.m.	Need Verification Approve
Dec. 13. 2023, 12:13 a.m.	Need Verification Approve

The flowchart on the right illustrates the verification process: Voter (envelope icon) → Signer (envelope icon) → Trustee (envelope icon) → verified (verified icon) → Approve (Approve button). The 'verified' status is highlighted with a yellow border.

Gambar 4. 28 Sebelum Verifikasi Beberapa *Voter*



Gambar 4. 29 Setelah Verifikasi Beberapa *Voter*

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Bedasarkan dari penelitian yang telah dilakukan pada sistem *e-voting* menggunakan Skema *Blind Signature* dan *Eliptic Curve Cryptography*, penulis dapat mengambil beberapa kesimpulan yaitu sebagai berikut.

1. *Skema Blind Signature* yang dikombinasikan dengan *Elliptic Curve Cryptography* berhasil diimplementasikan menjadi sistem sistem pemungutan suara elektronik (*e-voting*) yang memungkinkan suara pemilih dapat diverifikasi dan dihitung secara aman, tertutup, dan rahasia.
2. Kunci privat yang dibangkitkan pada sistem ini memiliki panjang 256 bit dengan menggunakan kurva elips P-256 yang merupakan rekomendasi kurva elips dari NIST (*National Institute of Standards and Technology*).
3. Panjang hasil enkripsi tidak bergantung pada panjang karakter pilihan yang diinputkan karena setiap pilihan yang diinput akan diubah menjadi *hash* dengan menggunakan fungsi SHA-256.

5.2 Saran

Setelah melakukan penelitian, penulis menuliskan beberapa masukan yang bisa menjadi pertimbangan untuk melanjutkan penelitian ini yaitu.

1. Pada sistem ini, Pihak Berwenang masih manual dalam mengidentifikasi pemilih. Diharapkan pada penelitian selanjutnya dapat membuat identifikasi Pemilih secara otomatis.
2. Penelitian ini dibangun berbasis *web* untuk semua *User*. Diharapkan pada penelitian selanjutnya dapat mengembangkan sistem yang berbasis *mobile* untuk Pemilih sehingga Pemilih dapat lebih fleksibel mengakses aplikasi.
3. Pada sistem yang dibangun, Penyelenggara Pemilihan masih memilih Pihak Berwenang berdasarkan *Username* sehingga besar kemungkinan dapat terjadi kesalahan Penyelenggara Pemilihan dalam memilih Pihak Berwenang. Maka dari itu diharapkan pada penelitian selanjutnya dapat memberikan parameter tambahan untuk mengurangi terjadinya kesalahan.

DAFTAR PUSTAKA

- A., R. P. (2008). Penggunaan dan Perbandingan Blind RSA Signature pada Digital Credential. *Program Studi Teknik Informatika, Institut Teknologi Bandung*.
- Aminudin, Aditya, G. P., & Arifianto, S. (2020). Algoritme RSA menggunakan pembangkit kunci ESRKGS untuk enkripsi pesan chat dengan protokol TCP/IP. *Jurnal Teknologi dan Sistem Komputer*.
- Bashir, M. Z., & Ali, R. (2021, 10 25). Cryptanalysis and improvement of blind signcryption scheme based on elliptic curve. Islamabad, Pakistan.
- Chaum, D. (1982). Blind signatures for untraceable payments.
- Damanik, P. S. (2019). Implementasi Algoritma Elliptic Curve Cryptography (ECC) Untuk Penyandian Pesan Pada Aplikasi Chatting Client Server Berbasis Desktop. *Jurnal Riset Komputer (JURIKOM)*.
- Harn, L., Hsu, C., Xia, Z., & Li, Z. (2022). Multiple Blind Signature for e-Voting and e-Cash. *The Computer Journal*.
- James, S., Gayathri, N., & Reddy, P. V. (2019). Pairing Free Identity-Based Blind Signature Scheme with Message Recovery. *MDPI*.
- Latifah, U. W., & Prasetyo, P. W. (2021). IMPLEMENTASI KRIPTOGRAFI KURVA ELIPTIK ELGAMAL DI LAPANGAN GALOIS PRIMA PADA PROSES ENKRIPSI DAN DEKRIPSI BERBANTUAN SOFTWARE PYTHON. *JOURNAL OF FUNDAMENTAL MATHEMATICS AND APPLICATIONS (JFMA)*.
- Maulid, H. (2018). The Implementation of Blind Signature in Digital Cash. *Department of Informatics Engineering, School of Applied Science Telkom University*.
- Nugroho, Y., & P. P. (2022). Implementasi Algoritma Elliptic Curve Cryptography (Ecc) untuk Pengamanan File Berbasis Web. *Seminar Nasional Mahasiswa Fakultas Teknologi Informasi (SENAFTI)*.
- Risnanto, S. (2017). Aplikasi Pemungutan Suara Elektronik/E-Voting Menggunakan Teknologi Short Message Service dan At Command.

JURNAL TEKNIK INFORMATIKA.

- Sibarani, E. B., Zarlis, M., & Sembiring, R. W. (2017). ANALISIS KRIPTO SISTEM ALGORITMA AES DAN ELLIPTIC CURVE CRYPTOGRAPHY (ECC) UNTUK KEAMANAN DATA. *Jurnal Nasional Informatika dan Teknologi Informasi*.
- THU, A. A., & MYA, K. T. (2015). an Efficient Blind Signature Scheme for E-Voting System. *International Journal of Advanced Computational Engineering and Networkin*.

LAMPIRAN

Operasi ECC-Penggandaan Titik

```
def ganda(x,y):
    if x == -1 or y == -1 :
        xr = -1
        yr = -1
        return (xr,yr)
    else :
        m = (((3*pow(x,2))+a)*(pow(2*y,-1,p)))%p
        xr = (pow(m,2) - 2*x)%p
        yr = ((m*(x-xr))-y)%p
        return (xr,yr)
```

Operasi ECC-Penjumlahan Titik

```
def jumlah(x1,y1,x2,y2):
    if x1==x2 and y1==y2 :
        return ganda(x1,y1)
    elif x1==x2 :
        xr = -1
        yr = -1
        return (xr,yr)
    elif x1 == -1 :
        return add_infinity_to_point(x2,y2)
    elif x2 == -1 :
        return add_infinity_to_point(x1,y1)
    else :
        m = ((y2-y1)*(pow(x2-x1,-1,p)))%p
        xr = (pow(m,2)-x1-x2)%p
        yr = ((m*(x1-xr))-y1)%p
        return(xr,yr)
```

Operasi ECC-Pengurangan Titik

```
def kurang(x1,y1,x2,y2) :
    y2_min = (-y2)%p
    return jumlah(x1,y1,x2,y2_min)
```

Operasi ECC-Operasi Perhitungan

```
def operasi(n,x,y) :
    if n <=0 :
        return "n is not valid"

    if n == 1 :
        return x,y
    elif n%2 == 0 :
        n//=2
        return ganda(*operasi(n,x,y))
    else :
        n-=1
        n//=2
        return jumlah(*ganda(*operasi(n,x,y)),x,y)
```

Pembangkitan Kunci

```
def private_key() :
    private_n = secrets.randbelow(p)
    while private_n.bit_length() != 256 :
        private_n = secrets.randbelow(p)

    priv_hex = hex(private_n)
    return(priv_hex)
```

Penentuan Base Persamaan Kurva Elips

```
def base() :
    for x in range(p) :
        y_sq = (x**3 + a*x + b) % p
        y = y_sq**0.5
        if y == int(y) :
            base_x = x
            base_y = int(y)
            break
    return base_x, base_y
```

Perhitungan Kunci Publik

```
def public_key(private_n) :

    x_base, y_base = base()

    private_n_hex = int(private_n,16)
    public_key = operasi(private_n_hex,x_base,y_base)
    public_key_hex_x = hex(public_key[0])
    public_key_hex_y = hex(public_key[1])
    public_key_hex = (public_key_hex_x, public_key_hex_y)

    return public_key_hex
```