Journal of Computer Science

Research Article

Designing the Future: A Blockchain-Based Framework for Transparent and Secure Elections

Jayesh Solanki[1] and Divyakant Meva[2]

[1]*Depertmaent of Computer Science, L. D. Arts College, India*
[2]*Depertmaent of Computer Science, Marwadi University, India*

Article history
Received: 18-02-2026
Revised: 03-04-2026
Accepted: 04-05-2026

Corresponding Author:
Jayesh Solanki
Depertmaent of Computer Science, L. D. Arts College, India
Email: solankijayesh@gmail.com

Abstract: Blockchain-based electronic voting systems have been identified as a solution to enhance the transparency, security, and efficiency of modern electoral processes. However, the existing system has three major problems, which include scalability, privacy issues, and cybersecurity attacks. The researcher proposed an innovative solution to develop an electronic voting system with enhanced security, scalability, and transparency of voters' information. This paper introduced the Hybrid Cryptographic and Enforced Blockchain (HCE VoteChain) framework, which combines Hyperledger Fabric with various sophisticated forms of cryptography, including SHA256 hashing, Advanced Encryption Standard (AES256) encryption, Elliptic Curve Digital Signature Algorithm (ECDSA) and digital signatures, Paillier Homomorphic Encryption, and Zero Knowledge Proof (ZKP) auditing. The experimental evaluation demonstrated that the system achieved a throughput of 288 Transactions Per Second (TPS) while maintaining an average latency of 2.521 seconds, a transaction speed of 0.13 votes per second, and a data immutability score of 0.999 and security resilience of 10000 and fault tolerance of 0.96, which proved its high reliability and robustness across different operational conditions. The results indicate that the framework suggested is a big step up in terms of security, scalability, and transparency over the existing solutions. Besides, it does not compromise the voter's privacy and auditability. The innovation of this work is the combination of multi-layer cryptographic mechanisms with the permission blockchain architecture to not only come up with a balanced compromise between performance and security but also to make the system capable of handling large-scale real-world digital elections.

Keywords: Blockchain, E-Voting, Hyperledger Fabric, Secure Elections, Zero-knowledge Proof

Introduction

Blockchain is a new technology with the potential to bring about more efficient, transparent, and secure electoral systems all over the world. Blockchain uses a decentralized ledger, this means that once a vote is entered, it cannot be changed or tampered with; thus, the chance of electoral fraud is lowered. Besides, smart contracts can powerfully force the automation of crucial election operations like voter identification and the counting of votes. Their correctness and the overall performance of the election can thereby be increased (Johnson, 2019; El Kafhali, 2024). Moreover, election systems with electronic and remote voting features facilitate the rise of voter participation by citizens, especially those voters who are far away geographically and those who are physically limited in their access to the voting stations. Nonetheless, even with these benefits, several issues, such as voter privacy, the ability to implement on a large scale, as well as the security of data, remain the most important ones (Jena and Dash, 2021; Anitha et al., 2023).

The current digital age needs voting systems that provide better access to voters while establishing system reliability and trustworthy operation. The use of traditional paper-based voting systems results in various operational problems because these systems are prone to mistakes and can be hacked (Khan et al., 2018; Qi et al., 2023). The introduction of Electronic Voting Machines (EVMs) and online voting systems has not solved the

SCIENCE Publications

problems because people still doubt the systems' ability to show their voting process and verify their security status. The election process requires secure systems that provide transparent operations and efficient election management while protecting voter privacy (Wang et al., 2023; Boumaiza, 2024; Peelam et al., 2024).

Recent technological developments, such as biometric authentication, real-time monitoring, and mobile-based voting platforms, have greatly enhanced the voting experience. Blockchain technology especially provides major features like decentralization, immutability, and secured data storage, which are the reasons why it is being considered for electoral uses (Alotaibi et al., 2025). Votes can be seen as encrypted transactions that are stored across multiple distributed nodes; therefore, any unauthorized change is impossible. Besides, biometric identification is used to double-check the identity of the voter so that only eligible individuals get to take part in voting (Katkol et al., 2025). All these inventions are a step forward in improving confidence in the electoral processes and encouraging the public to have more interest in voting (Mohanta et al., 2018; Foschini et al., 2020).

Blockchain technology enhances decentralization, immutability, and transparency with secure recording of transactions on multiple nodes. In the context of elections, Blockchain could offer a secure method for capturing, authenticating, and aggregating votes (Hussaini et al., 2023; Kumar et al., 2023). Every vote cast is treated as a record transaction and securely encrypted to avoid malicious manipulation. The decentralized blockchain system prevents any single entity from controlling its entire operation, which decreases data alteration risks. Voters trust elections more because Blockchain enables complete verification of election results (Chafiq et al., 2024; Bhadoria et al., 2022). The e-voting system based on blockchain technology is depicted in Fig. 1.

The current blockchain technology used in election systems faces several challenges, which include scalability problems, privacy issues, and transparency deficiencies. Different parties who monitor blockchain systems can prevent changes to records, which creates an issue for current systems to protect voter privacy while maintaining their ability to verify results. The system faces two major challenges because it requires high transaction costs and network congestion during peak voting times to handle large-scale operations. The research problem aims to develop a new blockchain-based election system that would provide secure and scalable voting solutions with voter confidentiality protection while delivering live election results throughout worldwide voting events. Research objectives are formulated from research problems as follows:

- To develop a scalable and secure blockchain-based election framework that ensures the integrity of votes
- To implement cryptographic techniques to maintain the immutability and transparency of voting records
- To integrate permissioned blockchain models to provide controlled access to election data for authorized entities
- To enhance public trust in election processes by leveraging Blockchain's transparency and security features

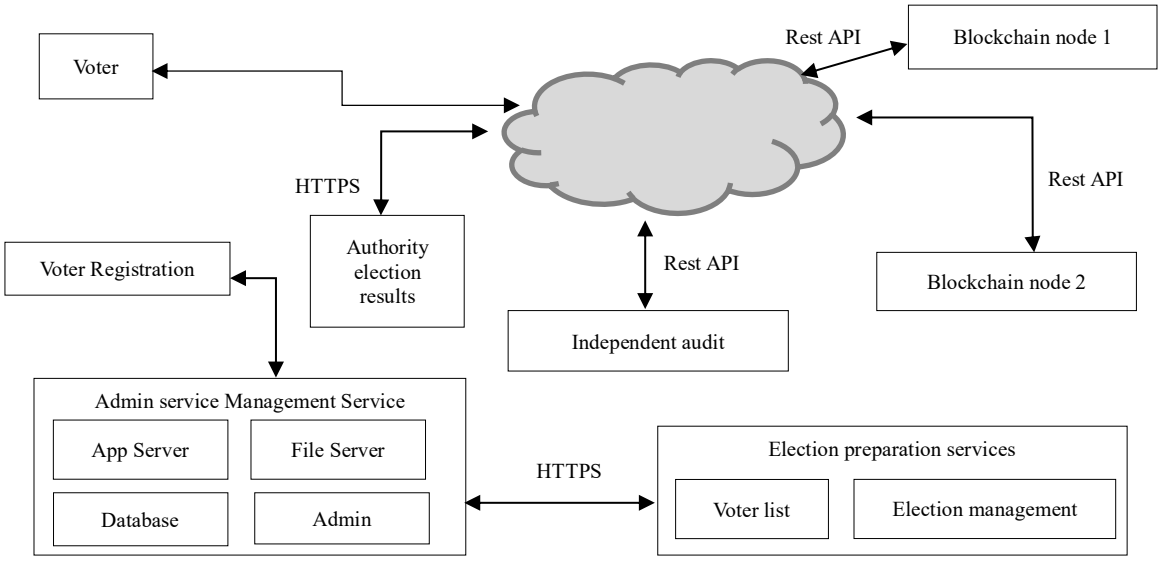

Fig. 1: A Blockchain voting system

This research presents an electronic voting system that achieves both security and transparency through its HCE-VoteChain. The proposed framework uses multiple advanced cryptographic methods to protect voter identification through SHA-256 identity hashing, secure vote data with AES-256 encryption, and maintain private vote counting through Paillier homomorphic encryption and guarantee system security through ECDSA authentication methods. The system uses ZKP to provide auditing capabilities, which maintain user privacy while enabling verifiable results, and its hybrid consensus system combines Proof of Authority **(**PoA) with Delegated Proof of Stake (DPoS) to improve system scalability and operational efficiency. The major contributions of this work can be enumerated as follows:

- Designed a safe and scalable e-voting system based on blockchain with the aid of Hyperledger Fabric
- Adding multi-layered encrypted methods for improved security, privacy, and data integrity
- Application of hybrid consensus protocols for enhanced system performance and minimum latency
- Use of Zero-Knowledge Proofs to enable privacy-preserving vote verification and auditing
- Extensive performance testing shows the benefits in throughput, latency, and fault tolerance

Related Work

This section analyzes earlier works of various scholars focusing on a blockchain-based framework for transparent and secure elections. Recent studies in blockchain-based e-voting systems demonstrated significant advancements in security, scalability, and efficiency.

Galal et al. (2026) proposed a blockchain-based e-voting system that used dual-layer AES and Rivest Shamir Adleman (RSA) encryption to protect voter privacy while providing transparent voting results. The system operated on Hyperledger Fabric, and its capacity was tested through simulations that handled 50,000 voters. The system achieved efficient transaction processing through lightweight processing requirements while maintaining strong protection against privacy breaches. Similarly, Marouan et al. (2026) designed a sustainable hybrid cryptographic framework based on ECDSA to enhance energy-saving and scalability aspects, where, according to the experimental data, energy consumption decreased by almost 50%, and throughput was increased from 120 TPS to 250 TPS, which clearly proved that the system became efficient and scalable. Previously, a blockchain-based application for secure digital transfer of tangible votes in an election using an IoT scanner based on homomorphic encryption and PoSS (proof of secret sharing) has been proposed by Mohammed and Wahab (2025). This implementation showed a 37.84% increase in TPS, an 84.14% decrease in verification time, and an 87.39% decrease in final processing time, as well as less CPU usage, resulting in improved system overall performance and reliability. Besides, Oon and Othman (2025) developed an online voting system architecture based on a permissioned blockchain using Hyperledger Fabric. They also assessed the scalability of the system through Hyperledger Caliper and concluded that the five-peer network setup was the most effective in terms of system performance and scalability while at the same time making sure that voting operations are both secure and transparent.

A blockchain-based voting system with fingerprints was built to improve the integrity of elections, and trust was constructed by Adeniyi et al. (2024). Blockchain technology made the electronic voting process possible through the generation of a public/private key pair based on fingerprint features for signing a transaction. The fingerprints were first taken and then underwent processing and feature scanning to derive a 256-bit private key. An additional public key was produced using an elliptic curve digital signature, which guaranteed the anonymity of the voters. The system has been validated on the Socoprint database containing 400 voters to make sure that every one of them was able to cast a vote without revealing their identity, thus eliminating the chances of identity theft and misuse. VoteChain's architecture was designed to integrate blockchain technology into the current voting framework of Palestine, which was suggested by Daraghmi et al. (2024). Systematic security for this e-voting system encompassed enhanced auditability, verification, and accuracy whilst preserving privacy and transparency. User Interface (UI) was created to improve user engagement and control double voting and impersonation attempts throughout election periods. VoteChain succeeded in regard to privacy, security, and system scalability, which outmatched other conventional and blockchain-integrated e-voting platforms. Additionally, Faruk et al. (2024) worked on the development of an Internet-connected biometric voting system built on blockchain technology. In working with this data, 87% of these people were eligible to preregister, whilst 88% were able to successfully identify themselves via biometric ID card fingerprint or facial recognition authentication.

In addition, Rahman et al. (2024) applied the RSA algorithm to the encryption and decryption of sensitive data in the voting machine that was vulnerable to hacking. MobileFaceNet facial verification devices protected sensitive information utilizing improved fingerprinting and multi-factor authentication. Similarly, Jayakumari et al. (2024) suggested an improvement to the preexisting voting system by proposing the use of a cloud-represented hybrid voting model. The Practical Byzantine Fault Tolerance (PBFT) consensus aided in safeguarding the votes and transactions against any manipulative or fraudulent attempts. The performance test analysis was conducted in comparison to other existing systems by continuously measuring the system and gathering the results.

A blockchain-enabled digital voting system, which is decentralized and immune to fraud, was proposed by Chaudhary et al. (2023). Voting was done through a smart contract deployed in the Remix Integrated Development Environment (IDE), where each candidate had different functions. Blockchain technology was cost-efficient because it was integrated with IPFS for data storage, while a 5G network provided low Latency, high availability, and dependable communication. The evaluation was performed with regard to performance metrics, which include cost per bit, storage costs, gas consumed, and cost. Likewise, Oprea et al. (2023) determined the essential needs of the e-voting system and proposed a solution for university elections. The comparison offered and examined evidence on the significance of the proposed solution compared to current systems. Moreover, Singh et al. (2023b) adopted a blockchain-based voting system that enhances the security, accessibility, and efficiency of the electoral processes. Voting occurred through smart contracts, while biometric identification was performed using Artificial Intelligence (AI) powered facial recognition. The outcomes from all scenarios were adequate to conclude that using the model enhanced the voting process with respect to decentralization and verification security.

Voting was a crucial process of democracy, and experts relied on paper ballots, even with the chance of mistakes and abuse, as suggested by Alvi et al. (2022). The proposed system aimed at maintaining anonymity, privacy, security, and fairness at the same time. The system that was implemented on Ethereum 2.0 with Solidity smart contracts proved to increase security infrastructure cost efficiency and, by proxy, improved the safety of voter information, the accuracy of results, and the prevention of election fraud. Farooq et al. (2022) proposed a model using advanced blockchain technology for auditability and reliability, which fostered confidence from the voters and the electoral bodies. Different methods of securing the voting system were considered, including hash encryption, mitigating 51% of attacks, and others. Table 1 shows the existing studies' gaps, merits, and demerits as described below.

Table 1: Existing studies research gaps, merits and demerits

Author Name	Year	Research Gaps	Merits	Demerits
Galal et al.	2026	Limited scalability and latency optimization; key management complexity.	Strong privacy using AES+RSA; supports up to 50,000 voters with low overhead.	Limited real-time scalability; complex key management.
Marouan et al.	2026	High system complexity and limited interoperability of hybrid cryptography.	Reduced energy (~50%) and improved throughput (120→250 TPS).	High implementation complexity; interoperability issues.
Mohammed and Wahab	2025	Focus on vote transfer; it lacks full e-voting implementation.	Improved performance (TPS +37.84%, faster verification 84.14%).	Limited to physical vote transfer; lacks end-to-end e-voting.
Oon and Othman	2025	Limited scalability evaluation under dynamic conditions.	Efficient permissioned blockchain with optimal 5-peer configuration.	Limited large-scale and dynamic performance validation.
Rahman et al.	2024	Vulnerability to cyberattacks in encryption mechanisms for voting systems	Enhanced security using RSA and multi-factor authentication.	Susceptible to advanced cyberattacks; limited robustness.
Jayakumari et al.	2024	Lack of practical solutions to safeguard votes from modifications or corruption	Improved security using PBFT consensus and a hybrid model.	Limited real-world implementation and scalability.
Chaudhary et al.	2023	Insufficient performance evaluation of blockchain-based e-voting systems	Cost-efficient system using smart contracts, IPFS, and 5G.	Lacks detailed scalability and performance validation.
Opera et al.	2023	Limited solutions for university elections using blockchain-based e-voting systems	Tailored solution for university elections with improved usability.	Limited generalizability to large-scale elections.
Singh et al.	2023a	Lack of integration between Blockchain and AI-powered biometric systems for voter identification	Enhanced security using AI-based biometric authentication.	Integration complexity and computational overhead.
Alvi et al.	2022	Over-reliance on traditional paper ballots and the potential for fraud and inaccuracies in elections	Improved security and cost-efficiency using Ethereum smart contracts.	Scalability and performance limitations.
Farooq et al.	2022	Lack of transparency and trust in conventional e-voting systems	Enhanced transparency and auditability using blockchain.	Limited evaluation under real-world scenarios.

Materials and Methods

This section outlines the cryptographic security mechanisms, architectural design, and computational steps employed in the proposed Hybrid Cryptographic and Enforced Blockchain Framework for Transparent and Secure E-Voting system. The method guarantees an effective voter authentication and vote encryption together with secure storing functions and transparent vote counting methods, which enhance election integrity and security and voter privacy. The method applies cryptography and consensus methods with smart contracts to introduce an electoral process that is resilient to tampering, protects the privacy of voters, and is operationally transparent.

The complete voting operations and election procedures demonstrate their function through the process shown in Fig. 2. The election process begins at voter registration, which requires the system to validate each voter's identity before they can start voting. The procedure starts with voter registration and continues to candidate nomination until the process ends with ballot preparation, which requires approval from all necessary parties. The voting phase begins when people vote, and the system prints their ballots, which then undergo a process to verify their validity. The system first verifies and records the votes, which leads to the final stage of collecting votes and announcing the results. The flowchart displays icons that represent users and documents together with the verification systems that operate throughout the various stages of the workflow. The election process diagram establishes decision points that use directional arrows to show how steps depend on each other and follow their correct logical order.

The blockchain-based electronic voting system shown in Fig. 3 is illustrated via a Unified Modeling Language (UML) sequence diagram. It explains the interactions between the Voter, Election System Blockchain Smart Contract, and Voting Authority. The voter first provides identification and biometric data to re-authenticate themselves.

The system employs Blockchain technology for confirming the voter's identity and checking whether they are eligible to vote. Following the voter's choice of candidate, the system creates an encrypted vote, which is then saved in the Blockchain. The vote-counting process is handled by a smart contract, which has complete visibility into the entire operation. Voting Authorities would verify the integrity of all votes cast before releasing final official results using secure methods of communication. Upon verification, the resulting numbers would then be released to the public. The use of smart contracts and blockchain technology would enhance the security, transparency, and integrity of the elections by providing an immutable record for tallying votes and verifying results through automated means.

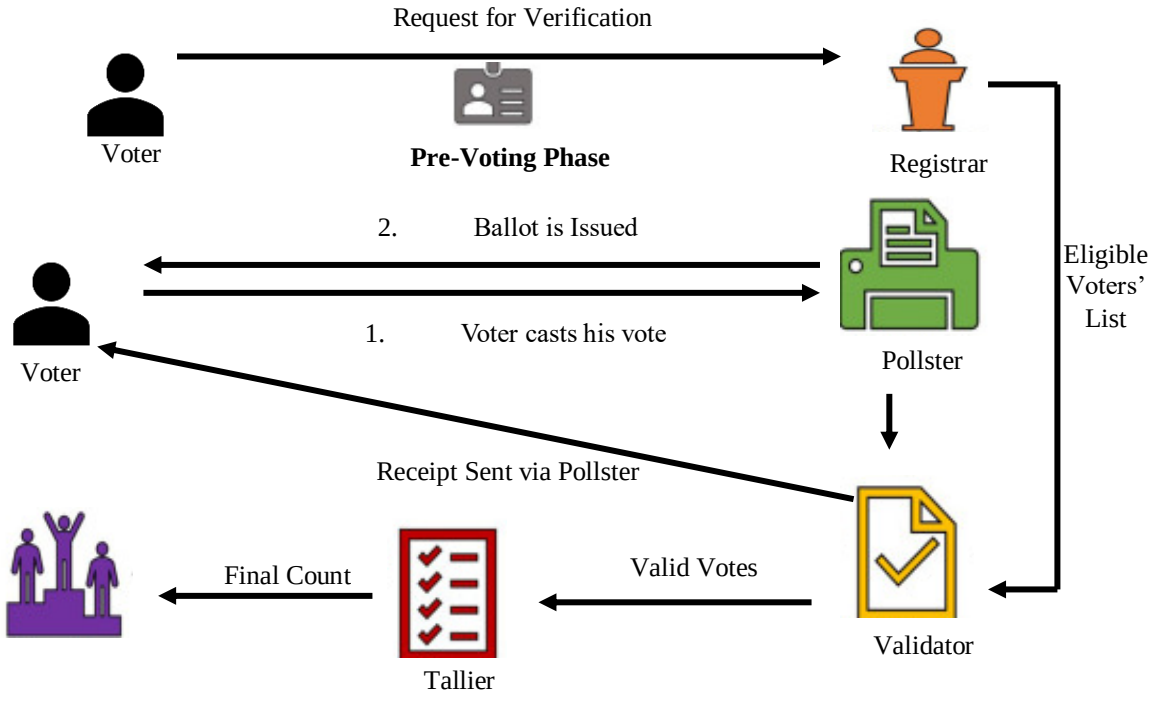

Fig. 2: Flow diagram of the secure Electronic Voting system process

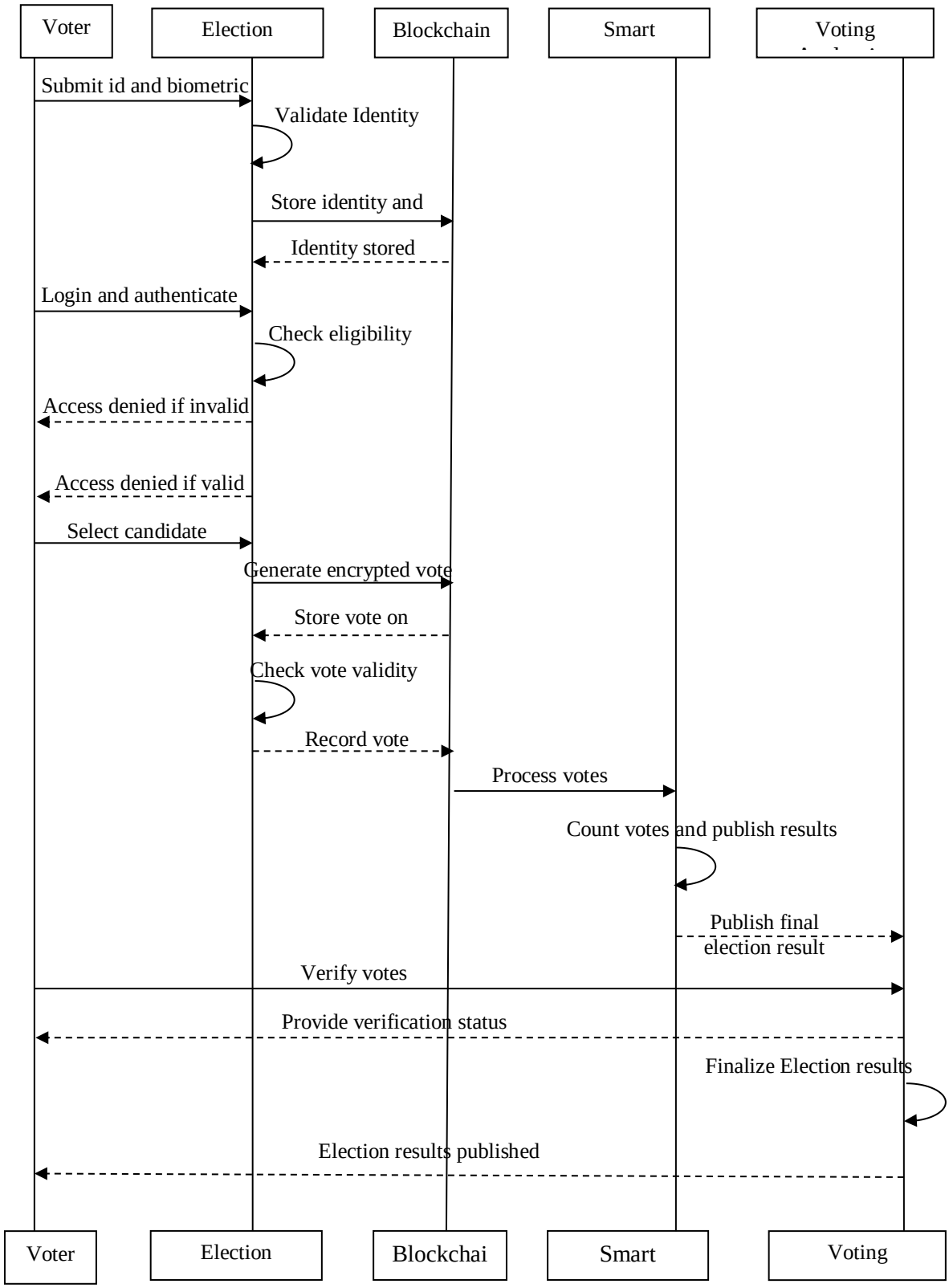

Fig. 3: UML sequence diagram

A blockchain-based digital voting system that provides security, transparency, and voter verification is depicted in Fig. 4. It begins with voter registration, where users enter their ID and biometric details for verification. To make it more secure and avoid fake registrations, the Unique Identification Authority of India (UIDAI) Aadhaar authentication API is implemented, only allowing eligible voters to be registered. A public-private key pair is created if the identity is legitimate, and voter data is hashed and retained on the Blockchain. Smart contracts check the registration; if accepted, the voter is registered. If not accepted, registration is denied.

The voting process starts with voter authentication, where users authenticate using their credentials. When the user is unable to log in for the third time, access to the account is denied. After the voter has successfully logged in, they proceed to select their chosen candidate. This vote is encrypted using AES 256 encryption and ECDSA digital signature. It is then assigned a unique time stamp. A smart contract verifies the integrity of the vote, thus preventing any malicious actions. Blockchain stores the verified votes in its records. This information is then verified through a consensus mechanism. The authorized nodes verify the votes using an audit trail method while keeping the voters' votes private. This information is made public, and the voters can verify their votes using ZKP, thus increasing the transparency and security of the system.

The platform enables secure voting through its ability to cast, store, and count votes while maintaining transparency and confidential voting rights, which is built on Hyperledger Fabric, a permissioned blockchain system. The system uses a Membership Service Provider (MSP) system to control network access because only authorized users are allowed to connect to the network, which creates a security difference with public blockchains. The smart contracts (chaincode) system performs three functions, which include user authentication, vote validation, and result computation, while maintaining unchangeable and trustworthy system operation. The ordering service handles votes in a sequential manner to maintain protection against double voting and vote tampering.

The voting system uses Paillier Homomorphic Encryption during vote aggregation to maintain vote confidentiality and support privacy-protecting vote counting. The system operates by summing encrypted votes and displaying only the final result, which protects system integrity while keeping individual votes secret from unauthorized users. The election system achieves better operational performance and security measures while maintaining system traceability through the modular and scalable framework of Hyperledger Fabric.

Voter Registration

The voter registration process is a significant step in the voting lifecycle on the Blockchain and guarantees that only qualified and authenticated users are inducted into the voting system. Duplicate or fictitious registrations are blocked using state-issued identity verification Application Programming Interfaces (APIs), like Aadhaar in India. A unique cryptographic key pair is generated for each voter to facilitate secure participation in elections without disclosing their identities. The voter's identity is hashed and stored on the Blockchain and, therefore, cannot be modified or changed.

Mathematically, the identity verification process is expressed as:

$$H(V_i) = SHA256(ID_i || Bi_i) \quad (1)$$

Where:

- V_i represents the unique voter identity
- ID_i is the voter's national identification number
- Bi_i is the biometric data (fingerprint, iris scan, etc.)
- $H(V_i)$ is the hashed voter identity stored on the Blockchain

After confirming a voter's identity, a public-private key pair generated through Elliptic Curve Cryptography (ECC) is allocated to the voter:

$$PK_i, SK_i = KeyGen() \quad (2)$$

Where:

- PK_i is the public key used for vote encryption
- SK_i is the private key used for signing votes securely

A smart contract guarantees the avoidance of duplicate registration by checking if $H(V_i)$ exists in the Blockchain. The registration process follows the rules:

$$SmartContract(V_i) \rightarrow \begin{cases} Register & if\ H(V_i) \notin Blockchain \\ Reject & if\ H(V_i) \in Blockchain \end{cases} \quad (3)$$

Such a mechanism secures individuals from getting unauthorized access while ensuring that each person registers only once.

Voter Authentication

Prior to the voting process, voters must prove their identity to avoid being impersonated, voting multiple times, or being a victim of fraud. The authentication procedure guarantees that voters who claim to access the platform are indeed authenticated. The voting system implements a hashed comparison method whereby the user-supplied credentials are hashed, and the resultant hash is checked against the blockchain-registered hash.

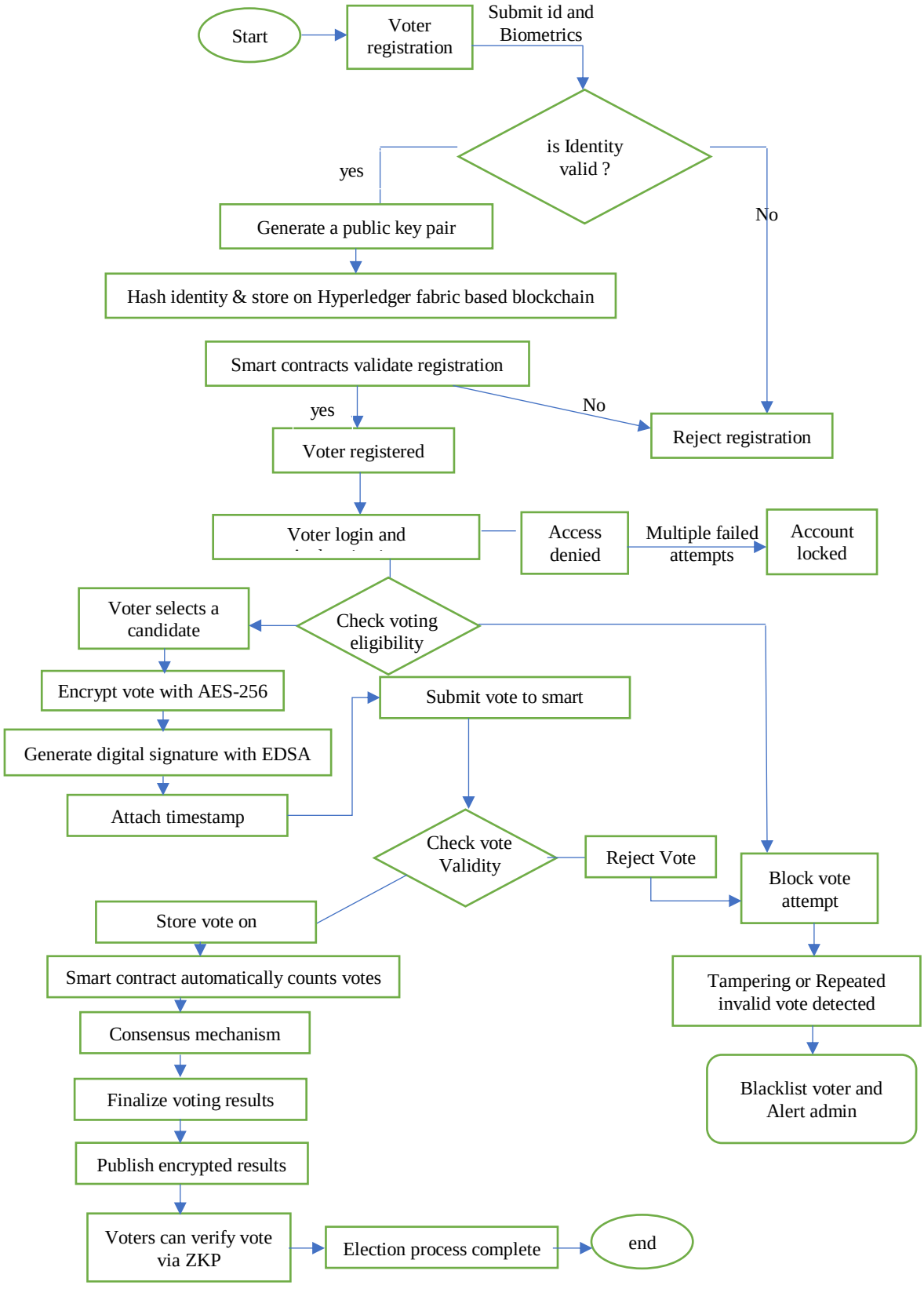

Fig. 4: Prototype of the proposed HCE-VoteChain framework

Mathematically, voter authentication is validated as follows:

$$H(ID_i, Bi_i) \stackrel{?}{\Rightarrow} H(V_i) \quad (4)$$

Where:

- $H(ID_i, Bi_i)$ is the hash generated from the user's input credentials
- $H(V_i)$ is the stored blockchain identity hash

If the hashes match, the voter is authorized; if not, access is denied. To mitigate brute-force attack risks, the system implements a login attempt control that suspends an account after the voter fails to log in within k attempts:

$$LoginAttempts(V_i) > k \Rightarrow Account\ Locked \quad (5)$$

In order to provide additional system security, an auto-logout mechanism is implemented where voter sessions have a limited duration:

$$T_{session} > T_{timeout} \Rightarrow Auto\ Logout \quad (6)$$

Such measures secure the network from external threats, maintaining the confidentiality of voter information and safeguarding system integrity.

Ballot Casting and Vote Encryption

Authenticated voters have the ability to choose their preferred candidate and complete their voting process. The system uses AES-256 encryption to protect the candidate information before the vote is submitted in order to ensure vote confidentiality and vote accuracy. The transmission of the vote remains secure because encryption prevents any unauthorized person from accessing or changing the vote data.

The encryption function is given by:

$$E(V_i) = AES_{256}(PK_i, C_j) \quad (7)$$

Where:

- $E(V_i)$ is the encrypted vote
- PK_i is the voter's public key
- C_j represents the selected candidate

To further ensure that the vote is legitimate and unaltered, a digital signature is generated using the ECDSA:

$$S(V_i) = ECDSA(SK_i, E(V_i) \quad (8)$$

Where:

- $S(V_i)$ is the signed encrypted vote
- SK_i is the voter's private key used for signing

To prevent replay attacks, where attackers might reuse previously cast votes, a timestamp is attached to each vote:

$$T_i = Timestamp() \quad (9)$$

The voting process begins with the encryption, signature, and timestamping of the vote, which proceeds to the Hyperledger Fabric blockchain for its entry and smart contract authentication process before it becomes part of the ledger:

$$SmartContract(E(V_i), S(V_i)) \rightarrow \begin{cases} Store\ on\ Blockchain & if\ valid \\ Reject\ Vote & if\ invalid \end{cases} \quad (1)$$

This ensures transparency, prevents vote manipulation, and guarantees the voter's anonymity.

Vote Storage and Counting

The blockchain-based system securely stores votes while using homomorphic encryption to aggregate votes because the system needs no decryption for individual votes during the tallying process, which protects voter identity. The system enables users to execute mathematical calculations on encrypted information without needing to unlock the secure data.

The encrypted vote sum is computed as:

$$E(Total) = \prod_{i=1}^{N} E(V_i)\ mod\ n^2 \quad (11)$$

Where:

- E(Total) is the encrypted total vote count
- n is the public modulus of the encryption scheme
- N represents the total number of votes cast

Votes are validated using consensus mechanisms, ensuring the integrity of the election.

Proof of Authority (PoA): PoA is a blockchain consensus model that relies on trusted validators (authorities) to authenticate issued transactions instead of mining or staking. Like Proof of Work (PoW) or Proof of Stake (PoS), PoA uses a set of nodes. However, in this case, the nodes are set in advance, simultaneously making them extremely effective, quick, and eco-friendly (Hajian et al., 2023). This approach is often applied in private and consortium blockchains where security and trust are placed above decentralization (Mwansa and Kabaso, 2023). It guarantees scalability with low transaction latency; however, due to its centralized nature, it is vulnerable to validator tampering. Ideal applications include enterprise solutions, supply chain management, and secure voting systems.

PoA relies on trusted validators V_i:

$$T = \sum_{i=1}^{n} V_i(T_x)\ if\ V_i(T_x) \geq 50\% \quad (12)$$

Where:

- T_x is a transaction (vote)
- $V_i(T_x)$ is a validator's approval
- The vote is valid if more than half approve

Delegated Proof of Stake (DPoS): DPoS functions as a blockchain consensus mechanism that enables stakeholders to select delegates who would perform transaction validation and block creation tasks instead of executing these responsibilities by themselves. The DPoS system operates with a selected group of delegates who validate blockchain transactions while all other users of the network maintain complete control over their staked digital assets, which makes it more efficient and energy-saving than traditional proof-of-stake systems (Elisa et al., 2023). This method enhances transaction speed and reduces costs, but its semi-centralized nature introduces trust dependency on elected delegates (Taş and Tanrıöver, 2021). DPoS functions as the primary consensus mechanism that drives efficient operations in blockchain networks that support cryptocurrencies, decentralized applications, and voting systems through its provision of fair and democratic governance systems.

DPoS allows stakeholders to vote for delegates:

$$W_i = \sum_{j=1}^{m} S_j \times P_j \quad (13)$$

Where:

- W_i is the delegate's total weight
- S_j is the stake of voter j
- P_j is the probability of delegation

This process guarantees accuracy, transparency, and confidentiality in vote counting.

Vote Auditing

Vote auditing ensures that the election process remains transparent, verifiable, and tamper-proof. It allows third-party auditors, election authorities, or even voters to verify the integrity of the votes without revealing the actual contents. Anonymity can be preserved, and manipulation can be prevented through cryptography during auditing.

In a system of voting that employs blockchain technology, two methods are used to conduct auditing.

Step 1: Voter-Verifiable Audit

Every voter gets an audit token, a type of receipt that corresponds uniquely to their vote. The vote verification is done using SHA-256. It guarantees that the vote is kept anonymous yet still verifiable:

$$R_i = SHA256(V_i, T_i) \quad (14)$$

Where:

- R_i is the audit receipt for voter i
- V_i is the voter's encrypted vote
- T_i is the timestamp of the vote

Users can confirm that their vote exists on the Blockchain by checking if their audit receipt is recorded on the public or open ledger:

$$R_i\ \varepsilon^{?} Blockchain \quad (15)$$

Once the audit receipt is retrieved, the voter is then free to verify that their vote was indeed counted.

Step 2: ZKP Auditing

The ZKP-based auditing model allows privacy to be maintained while enabling election integrity to be verified.

With the application of the ZKP protocol, the electoral system demonstrates that the total of all the cast votes in retention form equals the decrypted final vote count without disclosing individual choices:

$$\Pi: \exists(V_1, V_2, \ldots, V_N) such\ that\ E(Total) = \prod_{i=1}^{N} E(V_i)\ mod n^2 \quad (16)$$

Where:

- Π represents the ZKP system
- E (Total) is the encrypted total vote count
- E (V_i) are the encrypted individual votes
- N is the encryption modulus

This enables independent third-party verification to ensure that all votes were counted without requiring knowledge of who voted for whom.

Hyperledger Fabric for Secure and Permissioned Blockchain-Based Vote Management

The electoral process maintains its security and transparency and operational efficiency through the implementation of Hyperledger Fabric, which functions as a permissioned blockchain framework. The system implements controlled access through its MSP, which restricts system use to designated users (Haque et al., 2024). Smart contracts (chaincode) authenticate users and verify votes while computing results to maintain data integrity and security against changes. The ordering service executes vote processing in a sequential manner, which stops both fraudulent activities and double-voting incidents. The system uses a modular design, which enables it to grow by processing high volumes of transactions (Jena et al., 2024; Sutradhar et al., 2024).

Hyperledger Fabric increases vote protection and voter privacy through its use of AES-256 encryption together with ECDSA signature cryptographic methods. The introduction of these characteristics leads to substantial enhancements in reliability and operational performance, together with the ability to conduct audits for blockchain-based electronic voting systems.

Secure Hash Algorithm 256-bit (SHA256) for Data Integrity

The SHA-256 technique provides a sophisticated yet effective way of encrypting and storing sensitive information in the context of blockchain voting systems through voter identity and voting records. This approach to hashing information is common due to its ability to be irreversibly and tamper-proof, making sensitive information impossible to alter. For voter information and vote particulars that have already been cast onto the Blockchain, SHA-256 is applied as it prevents turning back the scrutability for these already cast votes (Sahib and Al-Shamery, 2021; Riza Chakim et al., 2023). The SHA-256 being collision resistant makes it more reliable than other hashing methods for voting systems (Nagajothi et al., 2021; Scientific, 2024). Like several different methods, it is extremely popular for ensuring security in blockchain systems because of the effectiveness of these systems against cryptographic attacks and breaches with minimal resources. The general diagram of SHA256 is shown in Fig. 5.

SHA-256 is a cryptographic hash function that transforms input data into a 256-bit fixed-length output:

$$H(M) = SHA - 256(M) \quad (17)$$

Where:

- $H(M)$ is the hashed output
- M is the input message (voter ID, vote, etc.)

AES-256 for Vote Confidentiality

AES-256 functions as a symmetric encryption algorithm to maintain the integrity and confidentiality of the data in the blockchain-based voting system. The algorithm is further regarded as uncrackable as it employs a 256-bit key size, which makes it virtually impossible to brute-force (Amrulloh et al., 2023). Votes must be encrypted before their transmission and storage. The data is locked to the point that, without access to the decryption key, the data is completely incomprehensible, even if it is intercepted (Chaturvedi et al., 2024). Moreover, its structure enhances its resilience to cryptographic attacks by using multiple rounds of encryption through a Substitution-Permutation Network (SPN) (Singh et al., 2023a). These factors apply to sensitive governmental or financial data, making AES-256 the most efficient and widely adopted symmetric key cipher, as illustrated in Fig. 6.

AES-256 is a symmetric encryption algorithm that encrypts plaintext P into ciphertext C:

$$C = E_k(P) \quad (18)$$

Decryption:

$$P = D_k(C) \quad (19)$$

Where:

- E_k is the encryption function using a 256-bit key k
- D_k is the decryption function using the same key k
- P is the plaintext vote
- C is the encrypted vote

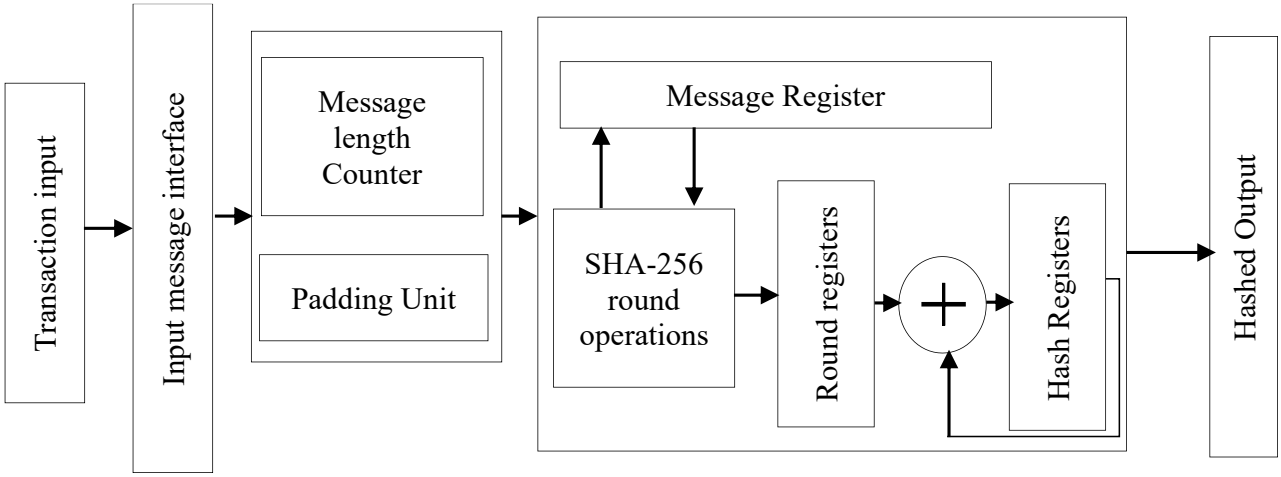

Fig. 4: SHA256

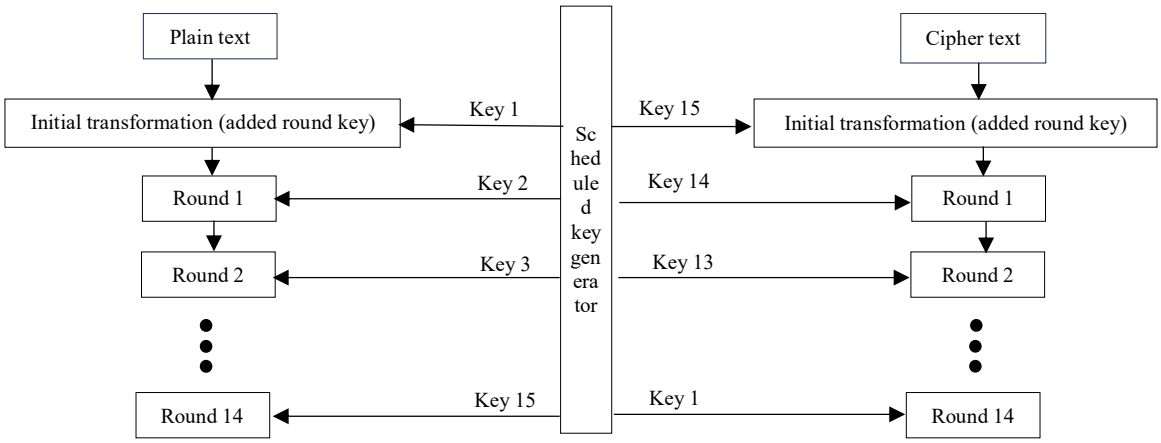

Fig. 5: A flow chart of the generic AES–256 algorithm steps for encryption and decryption

ECDSA for Digital Signature

ECDSA is an asymmetric cryptographic algorithm aimed at ensuring authentication, integrity, and non-repudiation features within the blockchain-based voting system. It uses the voter's private key to sign each vote as a unique digital signature, which allows the system to validate the vote without revealing sensitive information (Kagona, 2022; Marouan et al., 2024). The elliptic curve mathematics used in ECDSA is more efficient than other traditional algorithms, such as RSA, because it can provide more security with fewer keys and is thus more scalable (Gupta and Prasad, 2024). The ECDSA is used to sign the encrypted vote, as shown in Equation 20:

$$S(V_i) = ECDSA_{SKi}(E(V_i) \quad (20)$$

Where:

- Encrypted vote $E(V_i)$
- Digital signature $S(V_i)$

ZKP for Vote Auditing and Verifiability

ZKP is a type of cryptography where one party (the prover) is able to convince another party (the verifier) to have specific information without sharing it with them (Chai et al., 2022). In the context of a blockchain voting system, voters are able to verify that they are eligible to vote by proving their identity without revealing their name and history of voting through the use of ZKP (Mustafa and Waheed, 2021). This is done using cryptographic mathematics, which verifies a statement without degrading the hidden information. This helps ensure that votes that cannot be tracked are anonymous, untraceable, and verifiable, preventing identity stealing, voter coercion, and other nefarious acts (Zhou et al., 2024). ZKP allows a voter to prove their identity without revealing it, as given:

$$P: x \in S, V \leftarrow P \quad (21)$$

$$V \text{ verifies } f(x) = y \quad (22)$$

Where:

- P (prover) knows secret x
- S is a set of eligible voters
- V(verifier) checks without knowing x

Paillier Homomorphic Encryption for Privacy-Preserving Vote Counting

Paillier Homomorphic Encryption is a privacy-protecting encryption method that supports secure computation on encrypted data (Yuhao and Peng, 2024). E-voting supports the aggregation of encrypted votes without decryption, such that individual vote information is kept confidential during the election process. The additive homomorphic nature of Paillier encryption supports the direct summation of encrypted votes, which makes it suitable for vote counting in elections. This method ensures privacy for voters and simultaneously allows election authorities to securely and accurately determine the final vote tally without any unlawful interference on politically sensitive data while preserving and ensuring the quality, together with the openness of the electoral result.

Performance Metrics

To evaluate system performance and security effectiveness:

- Transaction Speed (TPS): Measures votes processed per second:

$$T_{speed} = \frac{Total\ Votes}{Processing\ Time}$$

- Latency: Measures the time taken for vote verification, storage, and confirmation:

$$L = t_{verification} + t_{recording} + t_{confirmation}$$

Total time required for:

- Verification
- Recording
- Confirmation

- Throughput Time: The total time required for all votes to be processed:

$$T_{throughput} = \sum_{i=1}^{n} T_i$$

Total time taken for all votes to be processed.

- Security Resilience: Resistance to cyberattacks:

$$R_{security} = \frac{1}{P_{attack}}$$

Higher security resilience means a lower probability of attack success.

- Fault Tolerance: Evaluate system stability in case of failures:

$$F_{tol} = \frac{T_{up}}{T_{total}}$$

Measures system reliability in case of failures.

This section discusses the proposed algorithm employed in this study.

Algorithm: Blockchain-based Secure e-voting system (HCE-VoteChain)

Step 1. Voter Registration

Identity Hashing

$$H_i = SHA256(V_i || B_i)$$

- The voter's ID V_i and biometric data B_i are concatenated and hashed using SHA-256.
- This ensures a unique, tamper-proof identity stored on the Hyperledger fabric.

Public-Private Key Generation

$$PK_i, SK_i = KeyGen()$$

- Each voter is assigned a cryptographic key pair:
  - PK_i (Public Key) - Used for encrypting votes.
  - SK_i (Private Key) - Used for signing votes.

Identity Verification & Smart Contract Validation

$$SC_i(H_i) = \begin{cases} Valid \Rightarrow R_i = Registered \\ Invalid \Rightarrow R_i = Rejected \end{cases}$$

- A Hyperledger Fabric smart contract (SC_i) checks whether the hashed identity exists in the system.
- If the identity is verified, the voter is registered; otherwise, registration is rejected.

Aadhaar Authentication API Integration

$$Auth(V_{ID}) = \begin{cases} 1, & if\ valid\ Aadhaar\ authentication \\ 0, & if\ invalid \end{cases}$$

- UIDAI's Aadhaar authentication API is integrated to prevent fake registrations and ensure only eligible voters register.

Step 2. Voter Authentication

Credential Hashing and Verification

$$C_i' = H(C_i)$$
$$C_i' \underset{=}{\rightarrow} H_i$$

- The entered credentials C_i are hashed and compared with the stored hash Hi.
- If they match, the voter proceeds; otherwise, the attempt is denied.

Login Failure & Account Locking

$$C_i' = H_i \Rightarrow Access\ Granted$$
$$C_i' \neq H_i \Rightarrow Failed\ Attempt\ Count++$$
$$If\ Failed\ Attempt\ Count > T_{lock}, Account\ Locked$$

- After multiple failed login attempts, the voter's account is locked to prevent brute-force attacks.

Step 3. Ballot Casting & Vote Encryption

Vote Encryption (Confidentiality)

$$E(V_i) = PaillierEncrypt(C_i, PK_i)$$

- The selected candidate C_i is encrypted using Paillier Homomorphic Encryption and the voter's public key PK_i.
- This ensures vote confidentiality while allowing secure vote tallying without decryption.

Digital Signature (Authentication & Integrity)

$$S(V_i) = ECDSA_{SKi}(E(V_i))$$

- The ECDSA is used to sign the encrypted vote.
- This prevents unauthorized modifications.

Timestamping (Replay Attack Prevention)

$$T_i = Current\ Time$$

- A timestamp is attached to each vote to ensure that votes cannot be replayed or duplicated.

Vote Submission

$$Vote_i = \{E(V_i), S(V_i), T_i\}$$

- The vote is submitted to the Hyperledger fabric, including:
  - Encrypted vote $E(V_i)$
  - Digital signature $S(V_i)$
  - Timestamp T_i

Vote Validation by Smart Contract

$$SC(Vote_i) = \begin{cases} Valid \Rightarrow Store\ Vote_i on\ Blockchain \\ Invalid \Rightarrow Reject\ Vote \end{cases}$$

- A Hyperledger fabric smart contract checks vote validity:
  - If valid, the vote is stored on the Hyperledger fabric.
  - If invalid, the vote is rejected.

Step 4. Vote Storage & Counting

Vote Aggregation using Paillier Homomorphic Encryption

$$Count(C) = \sum_{i=1}^{N} I(V_i = C)$$

Votes are collected securely by utilizing Paillier Homomorphic Encryption, which permits the summation of encrypted votes without requiring decryption.

- The system counts votes for each candidate C.
- $I(V_i = C)$ is an indicator function:
  - Returns 1 if vote V_i belongs to candidate C.
  - Returns 0 otherwise.

Consensus Mechanism

$$Consensus = PoA + DPoS$$

- The blockchain consensus mechanism ensures only valid votes are counted:
  - Proof of Authority (PoA): A set of trusted validators authenticates votes.
  - Delegated Proof of Stake (DPoS): Delegates verify votes on behalf of voters.

Final Vote Computation

$$R_f = argmax\ Count(C)$$

- The candidate receiving the highest number of votes is selected as the winner. The outcome is validated and published securely using the Paillier decryption process.

Step 5. Vote Auditing

$$\Pi: \exists (V_1, V_2, \ldots, V_N) such\ that\ E(Total) = \prod_{i=1}^{N} E(V_i)\ mod n^2$$

- ZKP ensures that vote tallying is verifiable without revealing individual votes.
- Any discrepancy in the tally immediately flags an anomaly.

Step 6. Cryptographic Security Techniques

Paillier Homomorphic Encryption (Privacy-Preserving Vote Tallying)

$$E(V) = PaillierEncrypt(M, K)$$

- The Paillier Homomorphic Encryption System can encrypt votes, allowing for the private counting of votes. The election authorities can compute the total votes from the encapsulated votes without disclosing the vote breakdown while maintaining the confidentiality of the votes.
- This homomorphic encryption enables the secure aggregation and verification of votes, which protects the anonymity of voters while ensuring the security of the election.

AES-256 Encryption (Confidentiality)

$$E(V) = AES_{256}(M, K)$$

- AES-256 encryption ensures votes remain confidential.

SHA-256 Hashing (Data Integrity)

$$H(M) = SHA256(M)$$

- SHA-256 hashing prevents vote tampering.

ECDSA Digital Signature (Authentication & Integrity)

$$S(V) = ECDSA_{SK}(E(V))$$

- The ECDSA signature ensures votes remain unaltered.

Step 7. Performance Metrics

Transaction Speed (Votes Processed Per Second)

$$T_{speed} = \frac{Total\ Votes}{Processing\ Time}$$

- Measures system speed in processing votes.

Latency (Time Delay in Vote Processing)

$$L = t_{verification} + t_{recording} + t_{confirmation}$$

- Total time required for:
  - Verification
  - Recording
  - Confirmation

Throughput Time (Total Processing Time)

$$T_{throughput} = \sum_{i=1}^{n} T_i$$

- Total time taken for all votes to be processed.

Data Immutability (Tamper Resistance)

$$P_{immut} = 1 - P_{tamper}$$

- Ensures votes remain unchanged after being cast.

Security Resilience (Resistance to Attacks)

$$R_{security} = \frac{1}{P_{attack}}$$

- Higher security resilience means a lower probability of attack success.

Fault Tolerance (System Stability)

$$F_{tol} = \frac{T_{up}}{T_{total}}$$

- Measures system reliability in case of failures.

Results and Discussion

The HCE-VoteChain system was developed through the implementation of Hyperledger Fabric, which combined with Paillier Homomorphic Encryption, ECDSA, and SHA-256 to establish a secure voting system that guaranteed complete voting transparency.

The upgraded experimental setup now displays the simulation environment through a Chinese language demonstration of Hyperledger Fabric, which operates with 10 validator nodes and Raft consensus and ≥2/3 endorsement policy, CouchDB, and multi-channel configuration for scalability. The system evaluation involved testing with synthetically generated voting transactions, which used 100 to 500 transactions per block and 2 to 5 second block time to assess latency and throughput capabilities. The system achieved security and privacy through its multi-layer cryptographic framework, which included SHA-256, ECDSA, AES-256, Paillier, and ZKP, while the evaluation metrics measured system performance through latency and throughput, cryptographic expenses, and scalability. The new additions to the HCE-VoteChain system enhance both experimental clarity and reproducibility and experimental testing standards.

The secure e-voting system, which has been developed, shows its complete operational capacity through the visual interface, which appears in Fig. 7. The front-end interface developed using Python enables voters to vote by choosing a political party and providing their Voter ID through a valid sample. The proposed blockchain-based e-voting framework provides a secure and transparent platform for casting and recording votes. Voters can use the user-friendly interface to choose their political party while entering their Voter ID information. The system uses Aadhaar authentication to verify voters by checking their identity against a hashed registry. Every vote gets protected through Paillier Homomorphic Encryption, which also uses ECDSA signing to ensure integrity and timestamps to defend against replay attacks.

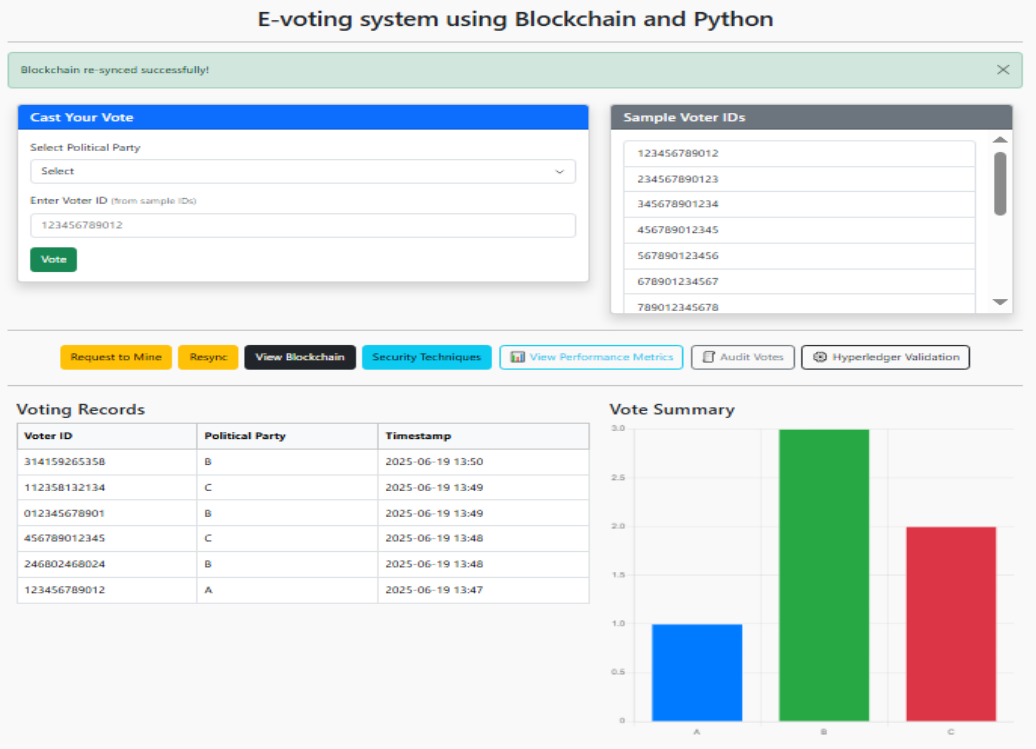

Fig. 6: Proposed HCE-VoteChain framework for vote casting

The encrypted vote becomes validated through smart contracts after its submission and then gets permanent storage on the Hyperledger Fabric blockchain. The system shows active voter participation through its real-time voter activity tracking feature, which displays a vote summary bar chart that updates with current voting results while keeping voter identities confidential. The system guarantees voter privacy while enabling verification through auditing processes and maintaining protection against unauthorized alterations, which results in a secure and trustworthy voting procedure. The blockchain-based e-voting system begins with a seamless voter registration process, where users input their personal details (Full Name, Voter ID, Aadhaar number, etc.) through a clean and intuitive interface, as shown in Fig. 8.

The Aadhaar verification system establishes authentic voter registration because it blocks all unauthorized registrations, which prevents fraudulent voter registration. The verification process assigns each voter a distinct public-private key combination, which enables the voter to encrypt their vote using the public key and to digitally sign their vote using the private key, as shown in Fig. 9, which displays the registered public key in Privacy-Enhanced Mail (PEM) format.

After registration, the voter's identity is hashed using SHA-256, combining their Voter ID and Aadhaar data, to create a unique identifier. The hashed identity gets stored on the Hyperledger Fabric blockchain because it creates an unchangeable, transparent record of the voter, which prevents any modification of their personal information. The registration information saved on the Smart Contact, together with the voter hash identity and registration validation timestamp, appears in Fig. 10.

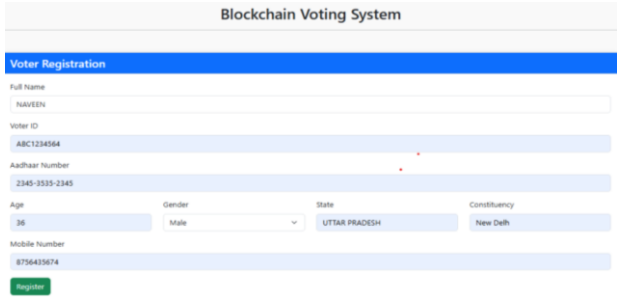

Fig. 7: Voter registration

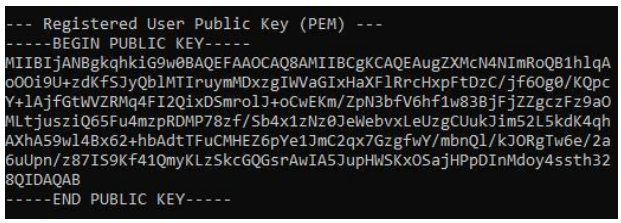

Fig. 8: Public key generated

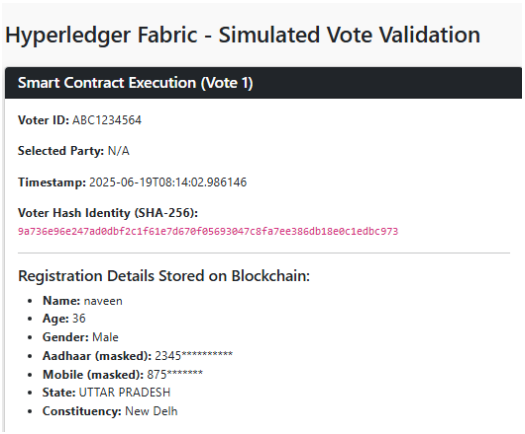

Fig. 9: Hash identity & store on Hyperledger fabric-based Blockchain

The Hyperledger Fabric blockchain uses smart contracts to verify voters' registration by matching their hashed identity with the existing registration records. The system achieves secure storage of voter registration details together with a confirmation message after completing successful validation. The registration process for the voter named "Naveen" has been completed according to the notification shown in Fig. 11.

After registration, the user login process redirects to the interface, which allows voters to choose their political party from three options (A, B, C) and enter their Voter ID. Fig. 12, displays a clean, simple form where the voter selects their desired party and enters a Voter ID, providing an intuitive and user-friendly experience. The process verifies voter identity to allow only verified voters to vote, while it stops unauthorized individuals from accessing the system.

Once the vote is cast, the system proceeds with the encryption and validation process.

The voting process starts with the system processing a voter's choice for Party B, as shown in Fig 13. The system starts a multi-layered encryption and validation process to protect three core security elements, which are confidentiality, integrity, and authenticity. First, the vote is encrypted using AES-256 to encode the vote data securely. The system creates a SHA-256 hash from the encrypted ballot to establish a mechanism that detects any attempts to tamper with the ballot. The hash receives digital signing through the ECDSA algorithm, which uses the voter's private key to establish authenticity while blocking any modifications to the vote. The system adds a digital timestamp (2025-06-19T08:35:29), which prevents replay attacks by ensuring every vote maintains its individual identity while requiring actual time voting. The cryptographic measures protect the voting process from the moment it starts until it reaches the Blockchain.

Fig. 10: Smart contracts validate registration

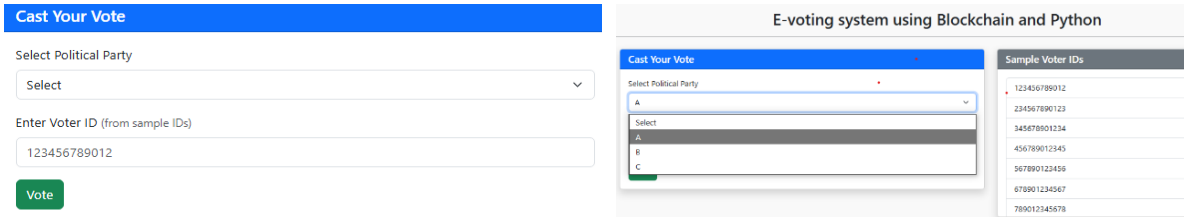

Fig. 11: Vote casting

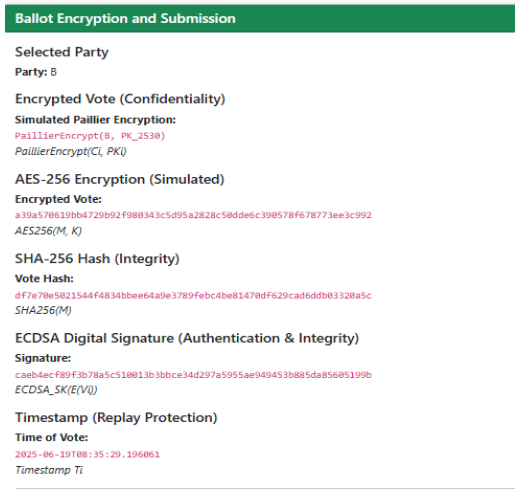

Fig. 12: Multi-layered encryption and validation

Once the voter, like ID: 123456789012, selects a candidate Party A, the encrypted vote, timestamp, and digital signature are submitted to the Hyperledger Fabric smart contract. The system validates the voter's registration, Aadhaar linkage, and uniqueness of the vote using the chaincode logic. All endorsing peers (PeerOrg1, PeerOrg2) approve the transaction, and the vote is committed to the blockchain ledger as Block #1685. The final status displays Vote accepted and stored on the Hyperledger Fabric as shown in Fig. 14.

In this case, a vote by another voter ID: 314159265358 is cast for Party B. Although the format is correct and the endorsement phase proceeds normally, the smart contract detects an issue, likely duplicate voting, unverified identity, or registration status as N/A. As a result, the vote is rejected during the chaincode logic validation, and the block is marked as a vote rejected by the smart contract, as shown in Fig. 15.

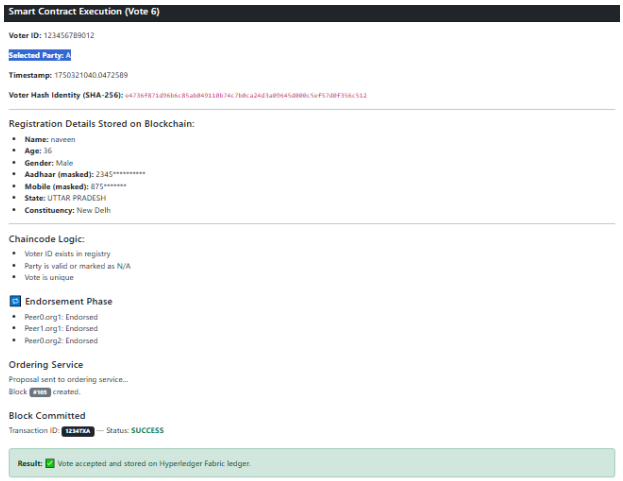

Fig. 13: Valid Vote Submitted and Accepted

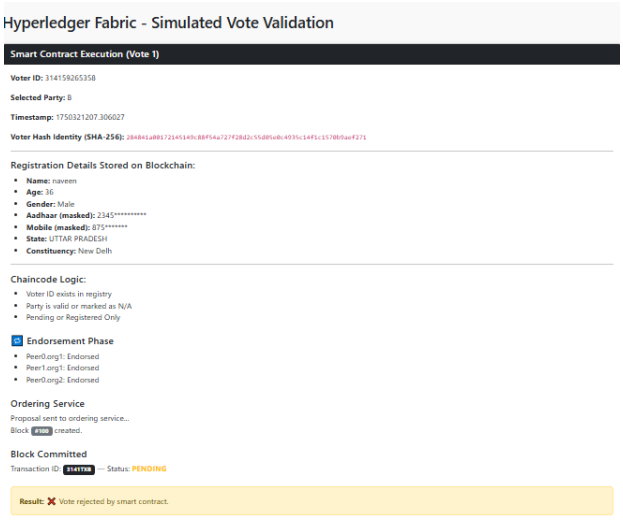

Fig. 14: Invalid Vote Rejected

After continuous invalid or fraudulent vote attempts using the same voter ID (123456789012), the system identifies it as a replay attack or duplicate submission, as shown in Fig. 16. The account gets locked automatically as a security mechanism to prevent brute-force attacks. The front-end displays: "Your ID is blocked or invalid!".

The e-voting system ensures secure and transparent voting through smart contract validation on Hyperledger Fabric. The system stores valid votes on the Blockchain after encrypting, signing, and verifying them. At the same time, it blocks any attempts to submit identical or false votes, and it secures accounts after multiple failed attempts to access the system. The system stores each encrypted and signed vote on the blockchain ledger after it is submitted to the voting process, which is displayed in Fig. 17. Each vote is recorded with a unique block index, timestamp, and hash, which guarantees that the data must remain unchanged and secure in storage. The process creates a system for transparent vote recording, which records every transaction as an unchangeable entry in the distributed ledger system.

The voting records display three types of information according to Fig. 18, which shows the voter ID, selected political party, and timestamp for each successfully cast vote. The records generate a vote summary bar chart, which shows the ongoing vote count for each political party. The results show that Party B leads with 3 votes, followed by Party C with 2, and Party A with 1.

The system demonstrates its smart contract execution ability through its use of smart contract-based systems to determine final results in Fig. 19. The system uses homomorphic encryption-based voting methods to count votes while ZKP technology verifies the voting results. The system verifies total voting results through encrypted methods, which keep all individual votes secret to protect voter privacy. The visual audit chart displays the election results, which show Party B as the election winner, thus allowing public verification of results and increasing election transparency.

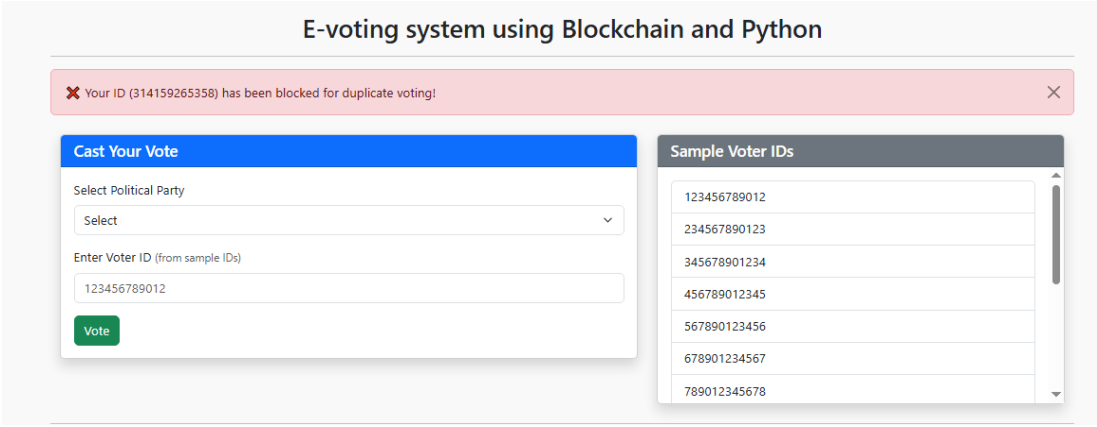

Fig. 15: Repeated Invalid Attempts: Account Blocked

DOI: 10.3844/jcssp.2026.1841.1865

{"length": 7, "chain": [{"index": 0, "transactions": [], "timestamp": 0, "previous_hash": "0", "nonce": 0, "hash": "6dbf23122cb5046cc5c0c1b245c75f8e43c59ca8ffeac292715e5078e631d0c9"}, {"index": 1, "transactions": [{"voter_id": "123456789012", "party": "A", "timestamp": 1750321040.0472589}], "timestamp": 1750321081.813717, "previous_hash": "6dbf23122cb5046cc5c0c1b245c75f8e43c59ca8ffeac292715e5078e631d0c9", "nonce": 151, "hash": "000658a81b43d621eee51ed71fd5634efbf52c7c12286b303a381661368df477"}, {"index": 2, "transactions": [{"voter_id": "246802468024", "party": "B", "timestamp": 1750321101.224485}], "timestamp": 1750321106.5041835, "previous_hash": "000658a81b43d621eee51ed71fd5634efbf52c7c12286b303a381661368df477", "nonce": 96, "hash": "0020001c400ad71443773381f6849381409673db0db709c3a0a807eef3afd7f3"}, {"index": 3, "transactions": [{"voter_id": "456789012345", "party": "C", "timestamp": 1750321118.7172852}], "timestamp": 1750321129.0183616, "previous_hash": "0020001c400ad71443773381f6849381409673db0db709c3a0a807eef3afd7f3", "nonce": 70, "hash": "00af8bedebc5499729c4cd764f296c9bef138e1515adddd6f30303721ea9e311"}, {"index": 4, "transactions": [{"voter_id": "012345678901", "party": "B", "timestamp": 1750321144.084157}], "timestamp": 1750321148.234263, "previous_hash": "00af8bedebc5499729c4cd764f296c9bef138e1515adddd6f30303721ea9e311", "nonce": 965, "hash": "0039b6a51d357320bc782f1b3b08624ad5252756e351e5ceda4a3a90d4579854"}, {"index": 5, "transactions": [{"voter_id": "112358132134", "party": "C", "timestamp": 1750321178.3866017}], "timestamp": 1750321187.8354056, "previous_hash": "0039b6a51d357320bc782f1b3b08624ad5252756e351e5ceda4a3a90d4579854", "nonce": 275, "hash": "00bc9043d5a1c29c172a7bdce78e23a3083da04ae7dfcfbd2a6bfdf1ef47ab4f"}, {"index": 6, "transactions": [{"voter_id": "314159265358", "party": "B", "timestamp": 1750321207.306027}], "timestamp": 1750321208.9554143, "previous_hash": "00bc9043d5a1c29c172a7bdce78e23a3083da04ae7dfcfbd2a6bfdf1ef47ab4f", "nonce": 106, "hash": "00365eaea0250854ba25d4fdb029c3d99b39b91e2a18e54ea3508faf9ade76cc"}], "peers": []}

Fig. 16: Blockchain Vote Storage Log

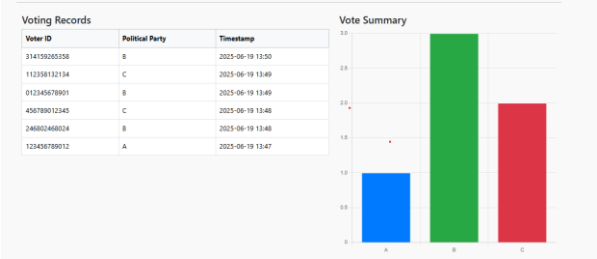

Fig. 17: Vote Records and Vote Summary

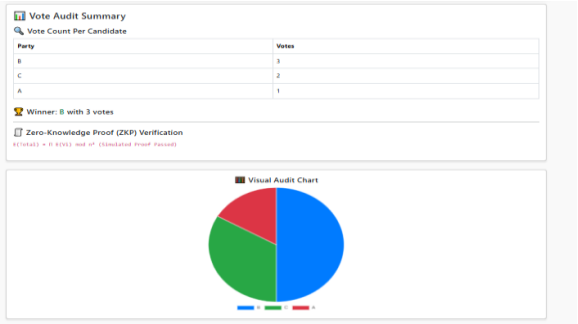

Fig. 18: Final Vote Result, Audit, and ZKP Verification

Table 2: Performance Metrics

Metric	Value
Transaction Speed	0.13 votes/sec
Average Latency	2.521 sec
Average Throughput	288 TPS
Data Immutability	0.999
Security Resilience	10,000
Fault Tolerance	0.96

The system was able to attain an average throughput of 288 TPS, which shows the system's capability to process high-scale, real-time voting transactions. Additionally, the system's transaction speed was 0.13 votes/sec. This shows the secure cryptographic processes

System Performance

The HCE-VoteChain system was developed through its implementation on Hyperledger Fabric, which underwent testing with different transaction volumes to assess its performance in efficiency and scalability, and system strength testing. This section presents the key performance metrics observed during system evaluation, including transaction speed, throughput, Latency, data immutability, security resilience, and fault tolerance. The system demonstrates its ability to handle high-volume voting operations through its security and reliability mechanisms, which are essential for conducting national and state elections. Table 2 summarizes the core performance outcomes of the system (AES-256, ECDSA, Paillier, and ZKP) the system undergoes in processing a single vote. These processes add to the system's processing time but provide the necessary security. The system's average Latency was 2.521 seconds. This shows the system's capability to provide timely responses to voters and confirm votes even in moderate to high loads.

Additionally, the system's immutability score was 0.999. This shows the system's capability to ensure that a recorded vote cannot be altered in the future, which is a requirement in a voting system. The system's security resilience was 10,000. This shows the system's low chances of being attacked due to the robustness of the system's cryptographic processes. Finally, the system's fault tolerance was 0.96. This shows the system's capability to maintain functionality and integrity even in the event of failure or partial network downtime.

In this case, metrics mainly indicate that the system is very secure, stable, and scalable. Immutability score confirms the integrity of the blockchain. Furthermore, high resilience and tolerance values are signs of a reliable system that can withstand attacks or failures.

Throughput Analysis

The system performance assessment required testing with different transaction loads to measure its scalability limits. The system throughput, which measures the successful transaction processing rate, shows continuous growth as more transactions are executed, as shown in Fig. 20. The system demonstrates throughput performance that matches the incoming transaction rate because the system maintains high capacity even during peak operation times. The system achieved 219 processed transactions at 200 TPS while it maintained 785 processed transactions at 500 TPS, which demonstrates that the system architecture can handle multiple tasks simultaneously while meeting horizontal scaling requirements. The Hyperledger Fabric platform achieves its efficiency through two main processes, which include peer endorsement policies and smart contract execution and parallel transaction validation.

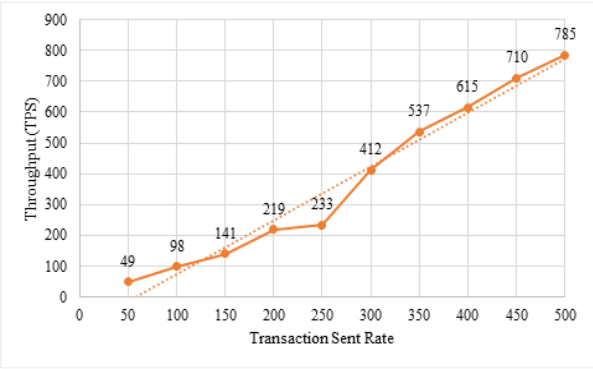

Fig. 19: Throughput vs Transaction sent rate

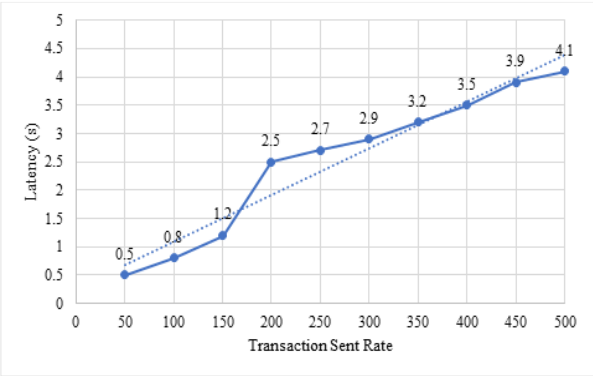

Fig. 20: Latency vs Transaction Sent Rate

The throughput increases with the rising input rate because it tracks the input rate development, which results in only minor reductions. The system demonstrates its ability to manage peak voting hours because it can handle high traffic without causing congestion or bottlenecks, which proves the effectiveness of smart contracts, parallel execution, and consensus processing.

Transaction Latency

The latency of the system increases steadily with the increasing system load, as shown in Fig. 21. The system maintains low latency during moderate load conditions, which reach 250 TPS, because it enables almost real-time vote recording. The system maintains a latency level below 5 seconds during high-load conditions, which reach 500 TPS, showing that it remains responsive during its most demanding operational periods.

Comparative Analysis

A comparative analysis was carried out with established electronic voting mechanisms to explore the proposed e-voting scheme leveraging blockchain technology.

Throughput performance assessment serves as the core evaluation method, which determines system capacity to handle maximum transaction loads during vital operations such as voter registration and vote casting. The analysis given in Table 3 considers throughput for 1,000 transactions with a transaction send rate of 200 TPS, using standardized benchmarking environments.

The comparative analysis based on throughput demonstrates that the proposed HCE-VoteChain system significantly outperforms existing techniques in the voter registration and vote casting phases. With a registration throughput of 219 TPS, surpassing the 142.2 TPS of a permissioned blockchain system (Oon and Othman, 2025) and the 185.2 TPS of the IBM Cloud-based approach (Clarke et al., 2023), the improvement is attributed to efficient cryptographic hashing and optimized smart contract execution. Similarly, the proposed system achieved 214 TPS during vote casting, far exceeding the score-voting method (Alshehri et al., 2023), owing to the integration of Paillier homomorphic encryption, ECDSA digital signatures, and timestamping within Hyperledger Fabric's parallel execution environment. In contrast, the score-based voting system showed the lowest scalability due to its computational complexity. Overall, these results affirm that the proposed framework is not only secure and transparent but also highly scalable, making it a strong candidate for real-world deployment in large-scale, time-sensitive electoral applications requiring high throughput, resilience, and auditability.

The latency comparison for 1,000 transactions at a transaction send rate of 200 TPS shows that the e-voting system achieves the lowest total latency among all tested methods. The HCE-VoteChain system demonstrates a registration latency of 1.095 seconds, which combines with a vote casting latency of 1.426 seconds to produce an overall latency of 2.521 seconds according to Table 4.

In comparison, AES and RSA dual layer system by (Galal et al., 2026) obtains a total latency of 2.48 seconds, the permissioned blockchain system by (Oon and Othman, 2025) reports a total latency of 4.08 seconds, while the IBM Cloud-based Hyperledger Fabric approach by Clarke et al., 2023) exhibits a higher latency of 5.5 seconds, primarily due to heavier cloud processing overhead during vote casting. Notably, the score-based voting model by Alshehri et al. (2023) shows a significantly higher delay of 50.71 seconds, making it unsuitable for real-time election scenarios. The improved latency of the proposed system can be ascribed to the lightweight operations of the cryptographic processes, parallel processing, and efficient smart contract execution mechanisms, which collectively guarantee timely recording and confirmation of votes even under moderate to high transactional loads. This positions the system as a reliable and responsive solution for real-time digital elections.

Table 3: Comparison based on Throughput

Author	E-voting system techniques	Throughput	
		Register Throughput	Vote Casting Throughput
Oon and Othman (2025)	Permissioned Blockchain + Hyperledger Caliper	142.2	142.4
Clarke et al. (2023)	Blockchain with IBM Cloud-Based Hyperledger Fabric	185.2	49
Alshehri et al. (2023)	Blockchain +score-voting	10	32
Proposed study	HCE-VoteChain	219	214

Table 4: Comparison based on Latency

Author	e-voting Technique	Latency	Total Latency (s)
		Register + Vote casting.	
Galal et al. (2026)	AES + RSA dual-layer	1.77+1.07	2.84
Oon and Othman (2025)	Permissioned Blockchain + Hyperledger Caliper	2.04+2.04	4.08
Clarke et al. (2023)	Blockchain with IBM Cloud-Based Hyperledger Fabric	1.2+4.3	5.5
Alshehri et al. (2023)	Blockchain +score-voting	34.41+16.3	50.71
Proposed study	HCE-VoteChain	1.095 + 1.002	2.521

Table 5: Comparison of Security Properties across E-voting Systems

Security Property	(Sheer Hardwick et al. 2018)	(Liu & Wang 2017)	(Chaieb et al., 2019)	(Zhang et al. 2018)	(Sun et al. 2019)	Proposed Algorithm
Eligibility	✔	✔	✔	✔	✔	✔
Anonymity	✘	✔	✘	✔	✔	✔
Fairness	✔	✘	✔	✔	✔	✔
Auditability	✘	✔	✔	✔	✘	✔
Individual Verifiability	✔	✔	✔	✘	✔	✔
Universal Verifiability	✔	✔	✔	✘	✔	✔
Vote Privacy	✘	✔	✔	✔	✔	✔
Coercion resistance	✔	✘	✘	✘	✘	✔
Replay Protection	✘	✘	✘	✘	✘	✔
Tamper-Evident Logging	✘	✘	✘	✘	✘	✔

The security comparison, which Table 5 presents, demonstrates that the proposed e-voting protocol achieves better results than all fundamental security requirements, which include eligibility, anonymity, fairness, and vote privacy and verifiability. The system provides Replay Protection and Tamper Evident Logging, which uses timestamping and SHA-256 hashing through Hyperledger Fabric to prevent vote reproduction and Vote tampering.

The mechanisms need to demonstrate their proof through resistance to coercion, universal verification, and auditability. The upgraded HCE-VoteChain method operates as a leading candidate for practical election implementation because it combines Paillier encryption and ECDSA and smart contracts as advanced cryptographic technologies to establish a publicly secure and fully verifiable voting system.

Discussion

The method used in this research, which combines Hyperledger Fabric with multiple cryptographic layers like SHA-256, AES-256, ECDSA Paillier homomorphic encryption, and Zero-Knowledge Proofs, was successful in providing a secure, transparent, and scalable e-voting system. According to the experimental assessment, the HCE-VoteChain system that was suggested in this research was able to process up to 288 transactions per second and had an average delay of 2.521 seconds, showing that it can handle live voting activities even when the number of transactions is moderate to high. The average transaction rate of 0.13 transactions per second, as well as the impact of using multiple encryption techniques to enhance privacy, security, and ensure the integrity of the data, was measured to be substantially slower than the average USD amount of money transferred through a third-party provider. In addition, this solution has an average immutability score of 0.999 and a high security resilience rating of 10,000, indicating robust protection against data manipulation, cyber-attacks, and partial failures in the entire system, a typical fault tolerance score of 0.96, along with an average system stability/effectiveness score of 99.93%.

The architectural design of the proposed framework leads to these results because it uses Hyperledger Fabric as a permissioned blockchain to process transactions through its ability to execute multiple transactions in parallel while controlling who can access the system. The implementation of Paillier homomorphic encryption protects voting data during aggregation because it enables secure processing without the need for decryption, which enables efficient tallying of votes while keeping voter information confidential. The combination of ECDSA and SHA-256 provides strong authentication together with data protection, while Zero-Knowledge Proofs enable auditing that maintains privacy and verification, which results in the system displaying advanced security and permanent data protection according to the outcomes. The system achieves better reliability and fault tolerance through its endorsement policy, which requires at least two-thirds of peers to approve, and its Raft consensus system.

The proposed framework demonstrated better performance than existing e-voting systems because it achieved higher throughput and lower latency through its efficient smart contract execution, parallel transaction processing, and Hyperledger Fabric optimized consensus mechanisms. The combination of homomorphic encryption and Zero-Knowledge Proofs achieved a solution that protected voter privacy while enabling election verification because it solved the main problems found in previous voting systems. The use of multiple cryptographic layers creates a situation where computation requirements increase, which can lead to performance issues during national elections that experience exceptionally high voter turnout. The HCE-VoteChain framework demonstrates its value as a trustworthy and secure system that can support modern digital elections, while its large-scale implementation needs further improvements.

Conclusion

The proposed HCE-VoteChain framework presents an end-to-end, future-resistant approach towards secure, transparent, and scalable electronic voting. Incorporation of the use of blockchain technology with advanced cryptographic techniques such as Paillier encryption, AES-256, ECDSA, and Zero-Knowledge Proofs ensures the integrity of the information, the secrecy of the voters, and end-to-end verifiability. The election process gains increased trustworthiness and auditability through the implementation of smart contracts together with replay defense and tamper-evident logging. The assessment results demonstrate that the framework achieves its performance objectives through a system capacity of 288 transactions per second, a system latency of 2.52 seconds, a fault tolerance rating of 0.96, and its ability to maintain security. The results demonstrate that HCE-VoteChain demonstrates theoretical strength while also demonstrating practical efficiency for implementation in actual high-risk electoral environments. The framework provides states and institutions with a solution that enables secure voting while satisfying the increasing requirements of digital democracy through its adaptable and fault-tolerant design.

Future Recommendations

The framework needs additional development because mobile voting interfaces can improve accessibility for people living in remote rural regions:

- Develop secure mobile-based voting interfaces to enhance accessibility for users in remote and rural areas
- Integrate AI-driven behavioral anomaly detection mechanisms to identify suspicious activities and prevent fraudulent voting attempts
- Enable multilingual interfaces to support international e-voting and improve usability across diverse populations
- Optimize the system for large-scale national and cross-border elections with higher transaction loads
- Incorporate biometric and multi-factor authentication to further strengthen voter verification
- Implement real-time dashboards for monitoring voting activities and system performance
- Explore quantum-resistant cryptographic techniques to ensure long-term security
- Enable integration with international electoral infrastructures for standardized and secure global voting frameworks

Acknowledgment

We are extremely grateful to all those who contributed to the completion of this research. Initially, I would like to express my sincere gratitude to Dean, my research guide, for his invaluable insights, encouragement, and guidance throughout the research process. The dedication and collaboration of the members of my research team are greatly appreciated. The distinctive perspectives and contributions of each member have significantly influenced the ideas outlined in this paper.

Funding Information

No funding was pursued by the authors for this project. This research on the Microsoft Azure platform was entirely self-funded by the individual involved in the project. Nevertheless, authors have the option to

investigate grant opportunities that may provide encouragement for our future research endeavors.

Author's Contributions

Jayesh Solanki: Formulated the research problem, developed the proposed HCE-VoteChain system architecture, and constructed the comprehensive study framework. He prepared the initial draft of the manuscript, conducted experimental evaluations, performed data analysis, and implemented the blockchain and cryptographic components.

Divyakant Meva: The research was supervised, who also provided critical guidance on research design and methodological rigor, validated the experimental findings, and made significant contributions to the interpretation of results and the refinement of the discussion and conclusion sections.

Ethics

The authors agree to comply with all ethical standards and to address any ethical concerns that may arise after publication.

Conflict of Interest

The researchers have declared that they have no conflicts of interest in connection with this investigation.

References

Adeniyi, J. K., Ajagbe, S. A., Adeniyi, E. A., Mudali, P., Adigun, M. O., Adeniyi, T. T., & Ajibola, O. (2024). A biometrics-generated private/public key cryptography for a blockchain-based e-voting system. *Egyptian Informatics Journal*, *25*, 100447. https://doi.org/10.1016/j.eij.2024.100447

Alotaibi, E. M., Issa, H., & Codesso, M. (2025). Blockchain-based conceptual model for enhanced transparency in government records: a design science research approach. *International Journal of Information Management Data Insights*, *5*(1), 100304. https://doi.org/10.1016/j.jjimei.2024.100304

Alshehri, A., Baza, M., Srivastava, G., Rajeh, W., Alrowaily, M., & Almusali, M. (2023). Privacy-Preserving E-Voting System Supporting Score Voting Using Blockchain. *Applied Sciences*, *13*(2), 1096. https://doi.org/10.3390/app13021096

Alvi, S. T., Uddin, M. N., Islam, L., & Ahamed, S. (2022). DVTChain: A blockchain-based decentralized mechanism to ensure the security of digital voting system voting system. *Journal of King Saud University - Computer and Information Sciences*, *34*(9), 6855–6871. https://doi.org/10.1016/j.jksuci.2022.06.014

Amrulloh, F. N., & Asriningtias, Y. (2023). Implementation of AES-256 Algorithm in Android-Based E-Voting Data Security. *Jurnal Penelitian Pendidikan IPA*, *9*(9), 7757–7766. https://doi.org/10.29303/jppipa.v9i9.4543

Anitha, V., Marquez Caro, O. J., Sudharsan, R., Yoganandan, S., & Vimal, M. (2023). Transparent voting system using blockchain. *Measurement: Sensors*, *25*, 100620. https://doi.org/10.1016/j.measen.2022.100620

Bhadoria, R. S., Das, A. P., Bashar, A., & Zikria, M. (2022). Implementing Blockchain-Based Traceable Certificates as Sustainable Technology in Democratic Elections. *Electronics*, *11*(20), 3359. https://doi.org/10.3390/electronics11203359

Boumaiza, A. (2024). A blockchain-centric P2P trading framework incorporating carbon and energy trades. *Energy Strategy Reviews*, *54*, 101466. https://doi.org/10.1016/j.esr.2024.101466

Chafiq, T., Azmi, R., & Mohammed, O. (2024). Blockchain-based electronic voting systems: A case study in Morocco. *International Journal of Intelligent Networks*, *5*, 38–48. https://doi.org/10.1016/j.ijin.2024.01.004

Chai, W., Liu, M., Zhang, Z., & Lv, L. (2022). Blockchain-based privacy-preserving electronic voting protocol. *International Journal of Network Security*, *24*(2), 230–237.

Chaieb, M., Yousfi, S., Lafourcade, P., & Robbana, R. (2019). Verify-Your-Vote: A Verifiable Blockchain-Based Online Voting Protocol. *Information Systems*, *341*, 16–30. https://doi.org/10.1007/978-3-030-11395-7_2

Chaturvedi, M., Sharma, G., & Sharma, A. (2024). *Blockchain-Based Electronic Voting System*.

Chaudhary, S., Shah, S., Kakkar, R., Gupta, R., Alabdulatif, A., Tanwar, S., Sharma, G., & Bokoro, P. N. (2023). Blockchain-Based Secure Voting Mechanism Underlying 5G Network: A Smart Contract Approach. *IEEE Access*, *11*, 76537–76550. https://doi.org/10.1109/access.2023.3297492

Clarke, R., McGuire, L., Baza, M., Rasheed, A., & Alsabaan, M. (2023). Online Voting Scheme Using IBM Cloud-Based Hyperledger Fabric with Privacy-Preservation. *Applied Sciences*, *13*(13), 7905. https://doi.org/10.3390/app13137905

Daraghmi, E., Hamoudi, A., & Abu Helou, M. (2024). Decentralizing Democracy: Secure and Transparent E-Voting Systems with Blockchain Technology in the Context of Palestine. *Future Internet*, *16*(11), 388. https://doi.org/10.3390/fi16110388

El Kafhali, S. (2024). Blockchain-Based Electronic Voting System: Significance and Requirements. *Mathematical Problems in Engineering*, 1–17. https://doi.org/10.1155/2024/5591147

Elisa, N., Yang, L., Chao, F., & Cao, Y. (2023). A framework of blockchain-based secure and privacy-preserving E-government system. *Wireless Networks*, *29*(3), 1005–1015. https://doi.org/10.1007/s11276-018-1883-0

Farooq, M. S., Iftikhar, U., & Khelifi, A. (2022). A Framework to Make Voting System Transparent Using Blockchain Technology. *IEEE Access*, *10*, 59959–59969. https://doi.org/10.1109/access.2022.3180168

Foschini, L., Gavagna, A., Martuscelli, G., & Montanari, R. (2020). Hyperledger Fabric Blockchain: Chaincode Performance Analysis. *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*, 1–6. https://doi.org/10.1109/icc40277.2020.9149080

Galal, M., Reheem, E. A. E., & Guirguis, S. (2026). Enhancing privacy and transparency in electronic voting: a blockchain-based cryptographic framework. *Journal of Cloud Computing*, *15*(1), 39 https://doi.org/10.1186/s13677-026-00839-z

Gupta, S. V. C., & Prasad, K. S. (2024). Neural-Based Secured Decentralized E-Voting Framework using Blur Image Broadcasting. *Annals of Emerging Technologies in Computing*, *8*(4), 77. https://doi.org/10.33166/aetic.2024.04.004

Hajian, M. B., Barzegar, H. R., El Ioini, N., & Pahl, C. (2023). Blockchain-Based E-Voting Systems: A Technology Review. *Electronics*, *13*(1), 17. https://doi.org/10.3390/electronics13010017

Haque, E. U., Shah, A., Iqbal, J., Ullah, S. S., Alroobaea, R., & Hussain, S. (2024). A scalable blockchain based framework for efficient IoT data management using lightweight consensus. *Scientific Reports*, *14*(1), 7841. https://doi.org/10.1038/s41598-024-58578-7

Faruk, M. J. H., Alam, F., Islam, M., & Rahman, A. (2024). Transforming online voting: a novel system utilizing blockchain and biometric verification for enhanced security, privacy, and transparency. *Cluster Computing*, *27*(4), 4015–4034. https://doi.org/10.1007/s10586-023-04261-x

Jayakumari, B., Sheeba, S. L., Eapen, M., Anbarasi, J., Ravi, V., Suganya, A., & Jawahar, M. (2024). E-voting system using cloud-based hybrid blockchain technology. *Journal of Safety Science and Resilience*, *5*(1), 102–109. https://doi.org/10.1016/j.jnlssr.2024.01.002

Jena, A. K., & Dash, S. P. (2021). Blockchain Technology: Introduction, Applications, Challenges. *Blockchain Technology: Applications and Challenges*, *203*, 1–11. https://doi.org/10.1007/978-3-030-69395-4_1

Jena, S. K., Kumar, B., Mohanty, B., Singhal, A., & Barik, R. C. (2024). An advanced blockchain-based hyperledger fabric solution for tracing fraudulent claims in the healthcare industry. *Decision Analytics Journal*, *10*, 100411. https://doi.org/10.1016/j.dajour.2024.100411

Johnson, D. (2019). Blockchain-Based Voting in the US and EU Constitutional Orders: A Digital Technology to Secure Democratic Values? *European Journal of Risk Regulation*, *10*(2), 330–358. https://doi.org/10.1017/err.2019.40

Kagona, E. (2022). Blockchain-Based Convolutional Neural Network E-Voting Scheme. *Advanced Journal of Science, Technology and Engineering*, *1*(1), 52–90. https://doi.org/https://doi.org/10.52589/ajste-x4asiqpq

Katkol, S., Dhulavvagol, P. M., & Totad, S. G. (2025). Performance Optimization in Blockchain Networks for Healthcare Systems Using Adaptive Sharding. *Procedia Computer Science*, *252*, 873–882 https://doi.org/10.1016/j.procs.2025.01.048

Khan, K. M., Arshad, J., & Khan, M. M. (2018). Secure Digital Voting System Based on Blockchain Technology. *International Journal of Electronic Government Research*, *14*(1), 53–62. https://doi.org/10.4018/ijegr.2018010103

Kumar, M., Raj, H., Chaurasia, N., & Gill, S. S. (2023). Blockchain inspired secure and reliable data exchange architecture for cyber-physical healthcare system 4.0. *Internet of Things and Cyber-Physical Systems*, *3*, 309–322. https://doi.org/10.1016/j.iotcps.2023.05.006

Liu, Y., & Wang, Q. (2017). *An e-voting protocol based on Blockchain*.

Marouan, A., Badrani, M., Kannouf, N., Zannou, A., & Chetouani, A. (2024). Blockchain-based e-voting system in a university. *Indonesian Journal of Electrical Engineering and Computer Science*, *34*(3), 1915. https://doi.org/10.11591/ijeecs.v34.i3.pp1915-1923

Marouan, A., Badrani, M., Zannou, A., Kannouf, N., & Chetouani, A. (2026). A sustainable hybrid cryptographic framework for energy optimization and scalability in blockchain-based e-voting. *Scientific African*, *31*, e03191. https://doi.org/10.1016/j.sciaf.2026.e03191

Mohammed, M. A., & Wahab, H. B. A. (2025). Blockchain-based Physical Election Votes Digitally Secure Transfer. *Journal of Soft Computing and Computer Applications*, *2*(1), 1018. https://doi.org/10.70403/3008-1084.1018

Mohanta, B. K., Panda, S. S., & Jena, D. (2018). An Overview of Smart Contract and Use Cases in Blockchain Technology. *2018 9th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, 1–4. https://doi.org/10.1109/icccnt.2018.8494045

Mustafa, M. K., & Waheed, S. (2021). An E-Voting Framework with Enterprise Blockchain. *Advances in Distributed Computing and Machine Learning*, *127*, 135–145. https://doi.org/10.1007/978-981-15-4218-3_14

Mwansa, P., & Kabaso, B. (2023). An Exploration of Blockchain Protocols for Trusted Vote Aggregation: A Consensus Algorithm Approach. *International Conference on Artificial Intelligence and Its Applications*, *2023*, 107–113. https://doi.org/10.59200/icarti.2023.015

Nagajothi, S., Shamrutha, S. S., Sreeja, D., & Tejeswini, S. J. (2021). A Transparent Immutable and More Secured Voting System Using Blockchain. *Journal of Physics: Conference Series*, *1964*(4), 042099. https://doi.org/10.1088/1742-6596/1964/4/042099

Oon, W. C., & Othman, S. H. (2025). Design and Scalability Performance Evaluation of Permissioned Blockchain Architecture for Online Voting System. *International Journal of Innovative Computing*, *15*(1), 87–94. https://doi.org/10.11113/ijic.v15n1.527

Oprea, S.-V., Bâra, A., Andreescu, A.-I., & Cristescu, M. P. (2023). Conceptual Architecture of a Blockchain Solution for E-Voting in Elections at the University Level. *IEEE Access*, *11*, 18461–18474. https://doi.org/10.1109/access.2023.3247964

Peelam, M. S., Kumar, G., Shah, K., & Chamola, V. (2024). DemocracyGuard: Blockchain-based secure voting framework for digital democracy. *Expert Systems*, *42*(2), 13694. https://doi.org/https://doi.org/10.1111/exsy.13694

Qi, W., Xia, Y., Zhu, P., Zhang, S., Zhu, L., & Zhang, S. (2023). Secure and efficient blockchain-based consensus scheme for MWSNs with clustered architecture. *Pervasive and Mobile Computing*, *94*, 101830. https://doi.org/10.1016/j.pmcj.2023.101830

Rahman, K. N., Hridoy, M. W., Mizanur Rahman, M., Islam, M. R., & Banik, S. (2024). Highly secured and effective management of app-based online voting system using RSA encryption and decryption. *Heliyon*, *10*(3), e25373. https://doi.org/10.1016/j.heliyon.2024.e25373

Riza Chakim, M. H., Aliyah, Yuda, M. A. D., Fahrudin, R., & Apriliasari, D. (2023). Secure and Transparent Elections: Exploring Decentralized Electronic Voting on P2P Blockchain. *ADI Journal on Recent Innovation (AJRI)*, *5*(1Sp), 54–67. https://doi.org/10.34306/ajri.v5i1sp.959

Sahib, R. H., & Al-Shamery, E. S. (2021). A Review on Distributed Blockchain Technology for E-voting Systems. *Journal of Physics: Conference Series*, *1804*(1), 012050. https://doi.org/10.1088/1742-6596/1804/1/012050

Scientific, L. L. (2024). The Development of a Blockchain-Based System for Electronic Voting. *Journal of Theoretical and Applied Information Technology*, *102*(17), 6468–6481.

Sheer Hardwick, F., Gioulis, A., Naeem Akram, R., & Markantonakis, K. (2018). E-Voting With Blockchain: An E-Voting Protocol with Decentralisation and Voter Privacy. *2018 IEEE International Conference on Internet of Things (IThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, 1561–1567. https://doi.org/10.1109/cybermatics_2018.2018.00262

Singh, A., Ganesh, A., Patil, R. R., Kumar, S., Rani, R., & Pippal, S. K. (2023a). Secure Voting Website Using Ethereum and Smart Contracts. *Applied System Innovation*, *6*(4), 70. https://doi.org/10.3390/asi6040070

Singh, A., Ganesh, A., Patil, R. R., Kumar, S., Rani, R., & Pippal, S. K. (2023b). Secure Voting Website Using Ethereum and Smart Contracts. *Applied System Innovation*, *6*(4), 70. https://doi.org/10.3390/asi6040070

Sun, X., Wang, Q., Kulicki, P., & Sopek, M. (2019). A Simple Voting Protocol on Quantum Blockchain. *International Journal of Theoretical Physics*, *58*(1), 275–281. https://doi.org/10.1007/s10773-018-3929-6

Sutradhar, S., Karforma, S., Bose, R., Roy, S., Djebali, S., & Bhattacharyya, D. (2024). Enhancing identity and access management using Hyperledger Fabric and OAuth 2.0: A block-chain-based approach for security and scalability for healthcare industry. *Internet of Things and Cyber-Physical Systems*, *4*, 49–67. https://doi.org/10.1016/j.iotcps.2023.07.004

Taş, R., & Tanrıöver, Ö. Ö. (2021). A Manipulation Prevention Model for Blockchain-Based E-Voting Systems. *Security and Communication Networks*, *2021*, 1–16. https://doi.org/10.1155/2021/6673691

Wang, Y., Li, Y., Suo, Y., Qiang, Y., Zhao, J., & Li, K. (2023). A scalable, efficient, and secured consensus mechanism for Vehicle-to-Vehicle energy trading blockchain. *Energy Reports*, *10*, 1565–1574. https://doi.org/10.1016/j.egyr.2023.07.035

Hussaini, W., Hanggoro, D., Salman, M., & Sari, R. F. (2023). PoIR: A Node Selection Mechanism in Reputation-Based Blockchain Consensus Using Bidirectional LSTM Regression Model. *Computers, Materials & Continua*, *77*(2), 2309–2339. https://doi.org/10.32604/cmc.2023.041152

Yuhao, H., & Peng, S. (2024). A Decentralized Voting System on the Polygon Blockchain. *Procedia Computer Science*, *247*, 1304–1313. https://doi.org/10.1016/j.procs.2024.10.156

Zhang, W., Yuan, Y., Hu, Y., Huang, S., Cao, S., Chopra, A., & Huang, S. (2018). A Privacy-Preserving Voting Protocol on Blockchain. *2018 IEEE 11th International Conference on Cloud Computing (CLOUD)*, 401–408. https://doi.org/10.1109/cloud.2018.00057

Zhou, L., Diro, A., Saini, A., Kaisar, S., & Hiep, P. C. (2024). Leveraging zero knowledge proofs for blockchain-based identity sharing: A survey of advancements, challenges and opportunities. *Journal of Information Security and Applications*, *80*, 103678. https://doi.org/10.1016/j.jisa.2023.103678