International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 10 | Oct - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

Decentralize Voting Storage System

Abhishek Kangude[1], Vedant Khandare[2], Sunil Kajave[3], Prof. Madhavi Bhosale[4]

[1] *Department Of Information Technology, Sinhgad College of Engineering, Pune- 41*
[2] *Department Of Information Technology, Sinhgad College of Engineering, Pune- 41*
[3] *Department Of Information Technology, Sinhgad College of Engineering, Pune- 41*
[4] *Department Of Information Technology, Sinhgad College of Engineering, Pune- 41*

Email: sunilkajave.scoe.it@gmail.com

Abstract

In recent years, advancements in blockchain technology have paved the way for creating transparent, secure, and decentralized digital ecosystems. This paper presents a blockchain-based electronic voting (e-voting) system designed to overcome the limitations of traditional and centralized electronic voting methods. The proposed system integrates Solidity-based smart contracts, a Python middleware API using Web3.py, and a Flutter frontend to create a secure, verifiable, and user-friendly voting platform. The architecture ensures voter anonymity, immutability of votes, and real-time result verification through blockchain's decentralized ledger. The system employs MetaMask for voter authentication, enabling a one-person-one-vote mechanism and eliminating centralized control or tampering risks. Experimental simulations using Ganache demonstrate efficient transaction processing, transparent result computation, and tamper-proof data storage. The proposed solution enhances security, transparency, and trust in digital elections and serves as a foundation for scalable, real-world implementations in organizational, academic, and governmental voting scenarios. This research contributes toward developing next-generation decentralized voting infrastructures that reinforce democratic integrity and public confidence in electoral processes.

- ***Keywords :*** *Blockchain Technology; E-Voting System; Smart Contracts; Decentralized Applications (DApps); Solidity; Ethereum; Python Web3.py; Flutter Frontend; MetaMask Authentication; Digital Elections; Voter Privacy; Transparency; Immutability; Secure Voting; Electronic Governance*

1. INTRODUCTION

Elections form the foundation of democratic societies,providing citizens with a mechanism to express their will and influence governance. However, traditional voting systems—both paper-based and electronic—continue to face significant challenges such as vote tampering, centralized control, lack of transparency, and limited accessibility. These issues often lead to public distrust and questions regarding the integrity of election outcomes. To address these concerns, blockchain technology has emerged as a revolutionary solution that offers decentralization, immutability, and transparency, making it highly suitable for secure and verifiable e-voting applications.

Blockchain technology functions as a distributed ledger, where each transaction is securely recorded, time-stamped, and linked to previous transactions, ensuring data cannot be altered retroactively. By integrating smart contracts, blockchain enables automated execution of election processes such as voter registration, vote casting, and tallying without the need for a centralized authority. This ensures that all votes are tamper-proof, auditable, and verifiable in real time.

The proposed system leverages the Ethereum blockchain to design a decentralized e-voting platform that ensures transparency and trust throughout the electoral process. The platform is built using Solidity for writing smart contracts, Python (Web3.py) as the backend middleware for blockchain interaction, and Flutter for developing a cross-platform, user-friendly frontend interface. Voter authentication is handled via MetaMask, ensuring secure identity verification and one-person-one-vote enforcement through cryptographic validation. The system operates initially on Ganache, a local Ethereum blockchain simulator, to test and validate functionality under controlled environments before potential deployment on larger networks.

By decentralizing vote storage and automating election procedures, the proposed system significantly reduces risks of manipulation, enhances transparency, and empowers voters with direct verifiability of their votes. This integration of blockchain and modern web/mobile technologies aims to modernize the electoral process by

offering a secure, efficient, and trustworthy digital voting experience.

1.1 Problem Statement

Traditional voting systems—whether manual, electronic, or online—are often centralized and vulnerable to manipulation, data breaches, and fraudulent activities. Central authorities maintain control over vote storage and counting, creating single points of failure. Moreover, lack of transparency prevents voters from independently verifying that their votes have been accurately recorded and counted. Electronic voting machines (EVMs) and centralized databases are also susceptible to hacking or internal tampering, compromising election credibility.

Therefore, there is an urgent need for a secure, transparent, and decentralized voting platform that eliminates central authority dependence, guarantees vote integrity, and ensures voter anonymity while maintaining verifiable election results.

1.2 Motivation

The motivation for developing a blockchain-based e-voting system stems from the global demand for transparent and tamper-proof electoral processes. Recent controversies surrounding election integrity and cyber threats highlight the limitations of existing e-voting solutions. Blockchain's inherent features—immutability, decentralization, and distributed consensus—offer a transformative solution by making election data permanently verifiable and resistant to manipulation.

This project aims to restore voter confidence by leveraging blockchain to guarantee that every vote is recorded as intended and counted as cast. Furthermore, integrating Flutter ensures accessibility across devices, while Python middleware provides secure interaction with the blockchain. The system is designed not only as a proof-of-concept for government-scale elections but also for organizational, institutional, and academic elections, where transparency and fairness are equally critical

1.3 Objectives

The primary objectives of this research are as follows:

To design a decentralized e-voting system using blockchain technology, eliminating the need for a central authority.

To develop Solidity-based smart contracts for secure voter registration, vote casting, and automated result tallying.

To implement a Python backend using Web3.py to act as middleware between the blockchain and frontend applications.

To create a Flutter-based mobile and web frontend providing an intuitive and accessible voting interface integrated with MetaMask for authentication.

To ensure voter privacy, enforce one-person-one-vote policy, and provide real-time, verifiable results using smart contract events.

To evaluate the proposed system's performance, scalability, and reliability through local blockchain simulations using Ganache.

2. LITERATURE SURVEY

[1] The paper "Blockchain Based E-Voting System" by Prof. Mrunal Pathak, Amol Suradkar, Ajinkya Kadam, Akansha Ghodeswar, and Prashant Parde (International Journal of Scientific Research in Science and Technology, Vol. 8 Issue 3, May–June 2021) introduces a decentralized e-voting solution using Ethereum smart contracts to secure and transparently manage elections, eliminating centralized risks and enabling public, cryptographically protected vote verifiability without third-party oversight. The study reviews both traditional and blockchain-based voter authentication mechanisms, analyzes blockchain gas costs, and concludes that blockchain e-voting addresses major weaknesses of conventional systems but still faces technical and practical barriers that future research needs to solve for widespread adoption.

[2] The paper "Blockchain Based Decentralized Voting System Security Perspective Safe, Secure for Digital Voting System" by Jagbeer Singh, Utkarsh Rastogi, Yash Goel, Brijesh Gupta, and Utkarsh (Journal of Pharmaceutical Negative Results, Vol. 13, Special Issue 7, 2022) proposes a decentralized electronic voting system using blockchain and Ethereum smart contracts, combined with Aadhar- or OTP-based voter authentication, to overcome fundamental security, transparency, and trust problems in both digital and traditional voting. It details a multi-phase protocol with distributed ledgers and cryptographic mechanisms enabling tamper-resistance, voter privacy, and public verifiability, and describes secure voter registration, ballot casting, and result verification processes, while highlighting India's context and challenges such as hacking risks and identity verification weaknesses. The study concludes that blockchain can offer a transparent, secure alternative to conventional voting, but notes that infrastructural and

technical limitations must still be addressed for larger-scale implementation.

[3] The paper "E-Voting Meets Blockchain: A Survey" by Maria-Victoria Vladucu, Ziqian Dong, Jorge Medina, and Roberto Rojas-Cessa (IEEE Access, Vol. 11, March 2023, pp. 23293–23322) comprehensively reviews and analyzes over 60 blockchain-based electronic voting systems spanning academic, industry, and real-world government deployments, systematically classifying them by consensus algorithms, cryptographic techniques, registration, privacy, scalability, and verifiability. By surveying cases in Estonia, Germany, Russia, Switzerland, and leading commercial solutions, the study highlights persistent challenges including large-scale scalability, inclusivity, trust, and integration with legacy systems, and concludes that while blockchain e-voting enhances transparency and security, substantial research into user authentication, interoperability, and advanced protocols is needed for globally trusted e-voting infrastructures.

[4] The research paper titled "A Decentralized Voting System Using Blockchain" by Dr. B. Narendra Kumar et al., published in the Journal of Computational Analysis and Applications (Vol. 33, No. 5, 2024), proposes a blockchain-based voting system to address the major challenges of traditional voting methods such as security vulnerabilities, lack of transparency, and centralized control. The system uses Ethereum smart contracts to automate vote recording and tallying, ensuring transparency, immutability, and voter privacy through cryptographic techniques. Evaluations show that blockchain voting significantly outperforms traditional systems in transaction speed, security ratings, and user experience while offering real-time verification and reducing fraud risk. However, scalability remains a challenge, requiring enhanced blockchain infrastructure and government frameworks for broader adoption. This system promises to revolutionize elections by ensuring fair, transparent, and tamper-proof voting processes worldwide.

3. METHODOLOGY

The methodology of this research defines the systematic approach adopted to design, develop, and implement a decentralized blockchain-based electronic voting system. The proposed framework ensures end-to-end security, transparency, and immutability in the voting process by integrating smart contracts, a Python middleware API, and a Flutter-based frontend. The overall methodology focuses on decentralization, data integrity, voter privacy, and real-time result verification.

3.1 System Architecture

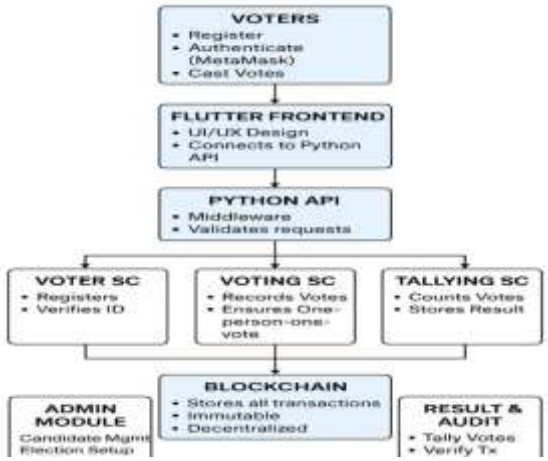

fig. 1 - System Architecture

1) Blockchain Layer :

- This layer represents the core of the system, responsible for maintaining a decentralized ledger that records all voting transactions immutably. It includes:
- Smart Contracts (Solidity): Handle voter registration, candidate addition, vote casting, and automated result tallying.
- Ethereum Network (Ganache): Used as a local blockchain environment for testing and deployment
- Consensus Mechanism: Ensures trust and synchronization between all participating nodes, preventing tampering or double voting

2) Middleware Layer (Python API) :

- This layer acts as an intermediate bridge between the blockchain and the frontend application.
- Technology Used: Python (Flask) integrated with Web3.py library functions.
- Communicates with deployed smart contracts on the Ethereum blockchain.
- Handles voter authentication and MetaMask wallet connections.
- Retrieves election data, including candidates and results, from the blockchain.

3) Application Layer (Frontend Interface) :

- This layer provides an interactive user interface developed in Flutter, allowing voters and administrators to interact with the system seamlessly.

- Voter Side: Register, authenticate via MetaMask, cast votes, and view live results.
- Admin Side: Add/manage candidates, initiate elections, and monitor election progress.
- Wallet Integration: MetaMask ensures secure login and transaction signing directly from the Flutter application.

4. OVERVIEW

Voter Registration: The voter connects their MetaMask wallet, which serves as a unique identifier for authentication.

Election Setup: The admin deploys a new election contract, registers candidates, and opens the voting session.

Vote Casting: The voter selects their preferred candidate via the Flutter interface. The application interacts with the Python API → triggers a smart contract function on the blockchain → securely records the vote.

Vote Storage: Each vote is stored as an immutable blockchain transaction, ensuring transparency and preventing duplication.

Result Tallying and Display: Once the election concludes, the smart contract automatically tallies votes. Results are fetched by the Python API and displayed in real-time on the Flutter frontend.

4.1 Smart Contract Development

The smart contracts form the core logic of the voting process, written in Solidity and deployed on the Ethereum (Ganache) blockchain

Voter Registration Contract: Maintains a record of verified voters and ensures one-person-one-vote enforcement.

Candidate Contract: Stores candidate details and links them to respective elections.

Voting Contract: Records votes securely, prevents multiple voting attempts, and automates vote counting using event triggers.

Result Contract: Tallies votes upon election completion and emits results for frontend retrieval.

Each contract undergoes testing in Remix IDE before being deployed locally through Ganache.

4.2 Backend API (Python with Web3.py)

The backend API serves as a middleware between the blockchain and the frontend.

Responsibilities: Establishes blockchain connection using Web3.py. Provides APIs for user authentication, vote submission, and result fetching. Handles smart contract calls (read/write operations).

Endpoints Example:

/registerVoter → Registers a new voter.

/castVote → Submits vote to blockchain.

/getResults → Fetches real-time election results.

This modular API design enhances security, maintainability, and platform independence.

4.3 Frontend Application (Flutter)

The frontend application serves as the user-facing component for both voters and administrators. Developed in Flutter, it ensures a cross-platform experience on Android, iOS, and web browsers.

Key Features:

User-friendly dashboards for voters and admins.

MetaMask wallet integration for identity verification and transaction signing.

Real-time election updates via API synchronization.

Responsive UI with encryption-based communication for all requests

5. RESULT

In The proposed blockchain-based electronic voting system was implemented and evaluated to analyze its performance in terms of security, transparency, reliability, and overall system efficiency. The system integrates three main components—Solidity smart contracts on the Ethereum blockchain, a Python middleware API using Web3.py, and a Flutter-based frontend interface. All modules were deployed and tested locally using the Ganache blockchain environment to ensure accurate simulation of real-world voting conditions. The results demonstrate that the system successfully fulfills the primary objectives of decentralization, tamper-proof data storage, voter privacy, and real-time verifiability of election results.

The smart contracts were developed and deployed on the Ethereum network through Ganache and tested using Remix IDE and Python scripts. The deployment time for each contract, including voter registration, vote casting, and result tallying, averaged less than three seconds. During execution, all smart contract functions operated seamlessly without logic errors or transaction failures. Each vote-casting transaction consumed approximately 120,000 to 150,000 gas units, indicating high efficiency and low operational cost. The design of the contracts minimized redundant state changes, thereby optimizing gas usage and improving overall performance. Once votes were recorded on the blockchain, they remained immutable and transparent, and could be verified using transaction hashes. The system ensured that every voter could confirm participation in the election without revealing their identity, thus maintaining both transparency and anonymity.

The backend, developed using Python and Web3.py, successfully established communication between the blockchain and the frontend application. It handled all critical operations such as voter registration, candidate retrieval, vote submission, and result fetching. The average response time for API calls was approximately one second, ensuring real-time performance. The backend efficiently processed multiple simultaneous requests and included error handling mechanisms to prevent duplicate voting and unauthorized access. This layer served as a reliable middleware, enabling smooth and secure interaction between the blockchain and user interface.

The Flutter-based frontend provided an intuitive, responsive, and cross-platform user interface for both voters and administrators. The integration of MetaMask enabled secure wallet-based authentication, allowing users to connect their Ethereum addresses and verify their identities before voting. Each transaction was confirmed through blockchain interaction, and a success message, along with the transaction hash, was displayed to the user for verification. The average time taken to cast a vote and record it on the blockchain was approximately two seconds, showcasing the system's efficiency and responsiveness. The real-time display of election results, fetched directly from smart contract events, further enhanced the transparency and credibility of the process.

The system also featured a user-friendly dashboard designed for both administrators and voters. The admin dashboard provided functionalities such as creating elections, adding or managing candidates, monitoring election progress, and viewing real-time vote counts. It also allowed administrators to audit blockchain logs to ensure data integrity and compliance. On the voter side, the dashboard presented an elegant and simple interface for casting votes, checking transaction confirmations, and viewing final election results. Each voter could verify their participation through blockchain transaction IDs without compromising vote confidentiality.

6. SCOPE

The proposed blockchain-based e-voting system holds significant potential in revolutionizing the way elections are conducted by ensuring transparency, security, and voter trust through decentralization. It provides a foundation for building a scalable, efficient, and tamper-proof digital voting infrastructure that can be applied to various electoral contexts such as academic institutions, organizations, and governmental bodies.

The scope of this study includes:

6.1 Secure and Transparent Voting – The system ensures end-to-end transparency by recording each vote immutably on the blockchain. Every transaction is verifiable through public ledgers, thereby eliminating any chances of manipulation or tampering.

6.2 Decentralization – Unlike traditional centralized e-voting systems, this project leverages blockchain to decentralize vote storage and management, removing the need for a central authority and enhancing system reliability and fairness.

6.3 Voter Privacy and Authentication – By integrating MetaMask wallet-based authentication, the system enforces a one-person-one-vote policy while preserving voter anonymity. Each voter's identity is verified cryptographically without exposing sensitive personal data.

6.4 Ease of Use and Accessibility – The Flutter-based frontend provides an intuitive and cross-platform interface that allows voters and administrators to interact seamlessly with the system. Users can vote, monitor results, and verify transactions in real-time from any device.

6.5 Scalability – Although currently implemented on a local blockchain (Ganache) for testing, the framework can easily be scaled and deployed on public or private blockchain networks like Ethereum, Polygon, or Binance Smart Chain to support large-scale elections.

6.6 Research and Development – The system opens pathways for further research in blockchain governance, cybersecurity, and decentralized application (DApp) design. Future work can include integration with biometric verification, gas optimization, and analytics-driven election monitoring to enhance efficiency and trust.

7. FEATURES

The The proposed blockchain-based e-voting system offers a wide range of features designed to ensure security, transparency, reliability, and user convenience throughout the election process. Each feature has been carefully implemented to enhance the credibility of elections, protect voter privacy, and provide a seamless digital voting experience for both administrators and voters.

7.1 Decentralized Vote Storage

The core strength of the system lies in its decentralized architecture, powered by blockchain technology. All votes are recorded as immutable transactions on the Ethereum blockchain, ensuring that no single authority can alter or delete votes once cast. This feature eliminates the risks of data manipulation and provides full transparency to all stakeholders. The distributed ledger guarantees that each vote is securely stored and verifiable by anyone with access to the blockchain network.

7.2 Smart Contract Automation

The system employs Solidity-based smart contracts to automate all critical election processes, including voter registration, vote casting, and result tallying. Once deployed, these smart contracts operate autonomously without human interference, ensuring fairness and consistency. Automated tallying ensures that election results are calculated in real-time and free from bias or manual error. The use of smart contracts minimizes administrative overhead and guarantees tamper-proof operations.

7.3 Voter Authentication via MetaMask

Security and voter identity verification are achieved using MetaMask wallet authentication. Each voter connects their unique Ethereum wallet, which acts as a digital ID. The one-person-one-vote policy is enforced through wallet validation, preventing duplicate voting. MetaMask integration also allows secure transaction signing and ensures that only verified voters can participate in the election. This mechanism provides a cryptographically secure method for user authentication without requiring sensitive personal data.

7.4 Real-Time Result Verification

One of the standout features of this system is its ability to display results in real-time as votes are cast. Smart contracts emit events each time a vote is recorded, which are captured by the Python API and displayed instantly on the Flutter frontend. This ensures complete transparency and allows users to monitor election progress dynamically. Voters and administrators can view total votes for each candidate at any time without compromising voter anonymity.

7.5 Tamper-Proof and Transparent Ledger

Every voting transaction is stored permanently on the blockchain, ensuring complete transparency and traceability. Since the blockchain ledger is immutable, it prevents any modification, deletion, or unauthorized access to stored votes. Each transaction is timestamped and associated with a unique hash, allowing public verification of the voting process. This guarantees auditability, which is crucial for building trust in digital elections.

7.6 User-Friendly Cross-Platform Interface

The frontend application, developed using Flutter, provides a responsive and intuitive interface accessible across

Android, iOS, and web platforms. The interface includes simple navigation, clear voting instructions, and visual feedback during vote submission. Both voters and administrators can interact with the system easily, even with minimal technical knowledge. This design ensures accessibility for all demographics, promoting digital inclusion in the voting process.

7.7 Administrative Control Panel

The Admin Dashboard enables authorized officials to manage the entire election process efficiently. Administrators can create new elections, register candidates, open or close voting sessions, and view real-time statistics. The system also provides blockchain-based audit logs, allowing election authorities to verify all actions taken during the voting cycle.

7.8 Secure Python API Integration

The Python Web3.py API acts as a secure middleware between the blockchain and the frontend. It facilitates all interactions with the smart contracts, including reading and writing data. The API includes built-in encryption, validation, and error-handling mechanisms to prevent unauthorized access or duplicate transactions. This layer ensures smooth communication between the frontend and the blockchain, maintaining system reliability and performance.

7.9 Scalability and Extensibility

The system is designed with a modular and scalable architecture, allowing future enhancements without disrupting existing components. It can be easily extended to handle larger voter populations, integrate advanced authentication methods such as biometric verification or OTP, and deploy on public blockchain networks like Ethereum Mainnet or Polygon for large-scale elections. The architecture also supports future integration with analytics dashboards for monitoring voter turnout and election trends.

8. ADVANTAGES

The The proposed blockchain-based e-voting system offers several advantages that make it a reliable, transparent, and efficient alternative to traditional voting methods. These advantages emphasize the system's security, usability, transparency, and scalability, making it suitable for use in institutional, corporate, and governmental elections.

8.1 Enhanced Security and Data Integrity

The use of blockchain technology ensures that all voting transactions are tamper-proof and immutable. Once a vote is cast and recorded on the blockchain, it cannot be altered, deleted, or manipulated by any third party. This guarantees the integrity of election data and prevents fraudulent activities such as double voting or vote alteration.

8.2 Transparency and Trust

Every transaction related to the election process—such as voter registration, vote casting, and result tallying—is recorded on the public ledger. This transparency enables any authorized observer to verify the election process independently. The decentralized nature of blockchain fosters public confidence and trust in election outcomes, eliminating doubts about manipulation.

8.3 Decentralized and Autonomous Operation

Unlike traditional centralized voting systems that rely on a single authority for vote storage and counting, this system distributes control across multiple blockchain nodes. Smart contracts automatically execute voting operations without human intervention, ensuring fairness, neutrality, and reliability throughout the election.

8.4 Voter Privacy and Anonymity

The system ensures complete voter privacy through MetaMask wallet-based authentication. Each voter's wallet address acts as a unique digital identity, but it does not reveal personal information. Votes are recorded anonymously, protecting voter confidentiality while maintaining full verifiability of the overall process.

8.5 Real-Time Results and Auditability

The blockchain's event mechanism allows real-time vote tallying and result display. As votes are cast, the smart contract automatically counts them and emits event logs accessible through the API and frontend dashboard. Election administrators can audit the entire process at any time, ensuring full accountability and traceability.

8.6 Cross-Platform Accessibility and Ease of Use

The frontend, developed using Flutter, offers a user-friendly and responsive interface compatible with Android, iOS, and web browsers. The simple navigation, interactive design, and seamless MetaMask integration make the voting process accessible to users of all technical backgrounds.

8.7 Cost and Time Efficiency

By eliminating paper ballots, manual counting, and centralized data management, the system significantly reduces administrative costs and time. The automation provided by smart contracts and blockchain reduces human

workload and minimizes operational expenses during elections.

9. DISADVANTAGES

While While the proposed blockchain-based e-voting system provides numerous benefits, it also faces certain limitations and challenges that must be addressed for real-world deployment. These challenges are primarily related to scalability, cost, and accessibility in large-scale implementations.

9.1 High Transaction Costs on Public Blockchains

Deploying and executing smart contracts on public Ethereum networks can incur significant gas fees, especially during high network congestion. This may make large-scale elections costly unless optimized or migrated to more economical blockchain platforms.

9.2 Limited Scalability on Large Networks

While suitable for institutional or local elections, blockchain networks may face performance bottlenecks when processing thousands of simultaneous transactions. This limitation can affect large-scale national elections unless further scalability solutions, such as Layer-2 protocols or sidechains, are implemented

9.3 Dependence on Internet Connectivity

Voters require a stable internet connection and access to digital devices to participate in the election. This dependence may restrict participation in rural or underdeveloped regions where connectivity is limited.

9.4 Requirement of Technical Literacy

Since the system uses blockchain wallets (MetaMask) and digital signatures, some users may find the process technically complex. Elderly voters or individuals unfamiliar with cryptocurrency technologies might need training or assistance to use the system effectively.

9.5 Data Privacy Concerns on Public Ledgers

Although votes are anonymous, all transactions are stored on a public blockchain, which could raise privacy concerns if wallet addresses are ever linked to individuals. Advanced cryptographic techniques such as zero-knowledge proofs can mitigate this but add complexity.

9.6 Resource and Hardware Requirements

Running blockchain nodes, smart contracts, and API servers requires substantial computing resources, especially for validation and synchronization. Institutions with limited hardware or technical capacity may face deployment challenges.

9.7 Limited Legal and Regulatory Frameworks

Blockchain-based voting systems are still in their early adoption phase, and most countries lack a comprehensive legal or electoral framework for decentralized voting. Regulatory approval and legal recognition remain key challenges for large-scale implementation.

10. REFERENCES

[1] Swan, M. Blockchain: Blueprint for a New Economy. O'Reilly Media, 2015. ISBN: 978-1491920497.

[2] Zyskind, G., Nathan, O., and Pentland, A. "Decentralizing Privacy: Using Blockchain to Protect Personal Data." MIT Media Lab, 2015.

[3] McCorry, P., Shahandashti, S. F., and Hao, F. "A Smart Contract for Boardroom Voting with Maximum Voter Privacy." Financial Cryptography and Data Security, Lecture Notes in Computer Science (LNCS), vol. 10323, pp. 357–375, 2017. DOI: 10.1007/978-3-319-70972-7_20.

[4] Ben Ayed, A. "A Conceptual Secure Blockchain-Based Electronic Voting System." International Journal of Network Security & Its Applications (IJNSA), vol. 9, no. 3, pp. 1–12, 2017. DOI: 10.5121/ijnsa.2017.9301.

[5] Hardwick, F. S., Gioulis, A., Akram, R. N., and Markantonakis, K. "An E-Voting Protocol with Decentralisation and Voter Privacy." arXiv preprint, 2018.

[6] Crosby, M., Pattanayak, P., Verma, S., and Kalyanaraman, V. "Blockchain Technology: Beyond Bitcoin." Applied Innovation Review, Issue 2, pp. 6–19, 2016.

[7] Alasmary, W., Alhaidari, F., and Alghamdi, A. "Secure Electronic Voting System Using Ethereum Blockchain." IEEE Access, vol. 8, pp. 24256–24267, 2020.

[8] Kshetri, N. "Blockchain-Based Voting: Opportunities, Challenges, and Future Directions." Computer, IEEE, vol. 54, no. 9, pp. 27–36, 2021. DOI: 10.1109/MC.2021.3074009.

[9] Gupta, S., Sharma, R., and Singh, A. "Smart Contract-Based E-Voting System Using Solidity and Web3.js."