# 66_An AI-Driven Blockchain Infrastructure for Secure, Transparent, and Reliable Voting System Report.pdf

My Files

My Files

Gautam Buddha University

## Document Details

Submission ID

trn:oid:::25761:116282656

Submission Date

Oct 11, 2025, 6:43 PM GMT+5:30

Download Date

Oct 11, 2025, 6:50 PM GMT+5:30

File Name

An AI-Driven Blockchain Infrastructure for Secure, Transparent, and Reliable Voting System Rep....pdf

File Size

405.5 KB

9 Pages

6,587 Words

38,303 Characters

# 41% detected as AI

The percentage indicates the combined amount of likely AI-generated text as well as likely AI-generated text that was also likely AI-paraphrased.

**Caution: Review required.**

It is essential to understand the limitations of AI detection before making decisions about a student's work. We encourage you to learn more about Turnitin's AI detection capabilities before using the tool.

## Detection Groups

**32** AI-generated only 41%
Likely AI-generated text from a large-language model.

**0** AI-generated text that was AI-paraphrased 0%
Likely AI-generated text that was likely revised using an AI-paraphrase tool or word spinner.

**Disclaimer**
Our AI writing assessment is designed to help educators identify text that might be prepared by a generative AI tool. Our AI writing assessment may not always be accurate (i.e., our AI models may produce either false positive results or false negative results), so it should not be used as the sole basis for adverse actions against a student. It takes further scrutiny and human judgment in conjunction with an organization's application of its specific academic policies to determine whether any academic misconduct has occurred.

## Frequently Asked Questions

**How should I interpret Turnitin's AI writing percentage and false positives?**
The percentage shown in the AI writing report is the amount of qualifying text within the submission that Turnitin's AI writing detection model determines was either likely AI-generated text from a large-language model or likely AI-generated text that was likely revised using an AI paraphrase tool or word spinner.

False positives (incorrectly flagging human-written text as AI-generated) are a possibility in AI models.

AI detection scores under 20%, which we do not surface in new reports, have a higher likelihood of false positives. To reduce the likelihood of misinterpretation, no score or highlights are attributed and are indicated with an asterisk in the report (*%).

The AI writing percentage should not be the sole basis to determine whether misconduct has occurred. The reviewer/instructor should use the percentage as a means to start a formative conversation with their student and/or use it to examine the submitted assignment in accordance with their school's policies.

**What does 'qualifying text' mean?**
Our model only processes qualifying text in the form of long-form writing. Long-form writing means individual sentences contained in paragraphs that make up a longer piece of written work, such as an essay, a dissertation, or an article, etc. Qualifying text that has been determined to be likely AI-generated will be highlighted in cyan in the submission, and likely AI-generated and then likely AI-paraphrased will be highlighted purple.

Non-qualifying text, such as bullet points, annotated bibliographies, etc., will not be processed and can create disparity between the submission highlights and the percentage shown.

# An AI-Driven Blockchain Infrastructure for Secure, Transparent, and Reliable Voting System

Raghav Kumar
*Department of Computer Science and Engineering*
*Amity University*
Noida, India
kumarraghav580@gmail.com

Bhanu Prakash Lohani
*Department of Computer Science and Engineering*
*Amity University*
Noida, India
bhanuplohani@gmail.com

Deepshikha Bhargava
*Department of Computer Science and Engineering*
*Amity University*
Noida, India
deepshikhabhargava@gmail.com

***Abstract*—Elections today face a tug-of-war between convenience and trust: while digital voting could make casting a ballot as easy as clicking a button, worries about hacking, vote buying, and technical glitches keep many wary. In this work, we sketch a new way forward by combining a permissioned blockchain, self-executing smart contracts, and elegant cryptography so that voters can cast, track, and verify their votes without fear of privacy breaches or tampering. Our design lets election officials spin up district-specific ballot contracts that enforce "one person, one vote" and tally results instantly. Voters log in with government-issued e-IDs and a PIN—no special hardware required—and receive a cryptographic receipt they can check later to confirm their vote was counted. Under the hood, zero-knowledge proofs hide who voted for whom while still proving every ballot is valid, and a two-tier model uses private sidechains for speed, periodically anchoring to a public ledger for a tamper-evident audit trail. We draw on lessons from platforms like Exonum and Quorum and introduce liquid-democracy options for flexible vote delegation. A careful security review shows our approach resists majority-control attacks, bogus node injections, and denial-of-service attempts, and we outline key-recovery and privacy measures to handle real-world hiccups. By putting blockchain's strongest guarantees into a simple, PIN-based workflow, this system aims to make national elections both bulletproof and welcoming to every voter.**

***Keywords*—Blockchain-based voting, permissioned blockchain, smart contracts, zero-knowledge proofs, cryptographic receipts, e-ID and PIN authentication, one person one vote, privacy-preserving elections, liquid democracy, sidechains with public ledger anchoring, vote verification, secure online elections, tamper-evident audit trail, resilience against attacks.**

## I. Introduction

Voting is more than just ticking a box—it's how citizens shape their community and country. Yet traditional paper ballots come with headaches: long waits at polling stations, mistakes in counting, and hefty costs for printing and staff. Moving to electronic voting could solve many of these problems, making it easy for people to vote from home or a mobile kiosk. But with that convenience comes new worries: What if hackers tamper with results? How can voters be sure their vote remains private?

This paper explores a fresh approach that blends familiar government-issued e-IDs and simple PINs with cutting-edge blockchain and cryptography. At its heart is a permissioned sidechain, where only trusted nodes validate votes quickly and efficiently. Every time someone votes, a small "smart contract" on the blockchain checks that the voter hasn't already cast a ballot and then adds the vote to a running tally. Voters walk away with a unique cryptographic receipt they can use later to confirm their vote was counted—without revealing who they voted for.

To keep everything above board, we anchor the sidechain's results periodically to a public blockchain. That way, any attempt to alter past votes is immediately obvious. We also use zero-knowledge proofs, which allow us to prove a vote is valid without ever exposing its content. This means election observers can verify the integrity of the vote without ever linking ballots back to individuals. Our design draws inspiration from real-world tools: Exonum for its speed, Quorum for its permissioned governance, and Go-Ethereum for its developer-friendliness. We even weave in ideas from liquid.

## II. Literature Review

Over the past decade, numerous studies have explored blockchain technology as a foundation for secure and transparent electronic voting. Early implementations, such as Follow My Vote and Helios Voting, demonstrated the potential of public blockchains like Ethereum to enhance transparency and auditability. However, these systems struggled with voter privacy and scalability, as all transactions were visible on-chain.

To address these limitations, researchers shifted toward permissioned and hybrid blockchain models, where only authorized entities manage the network. These systems, inspired by frameworks like Quorum and Exonum, achieve faster consensus and improved privacy while maintaining verifiable results through periodic anchoring to public ledgers.

Another important innovation has been the integration of smart contracts to automate election processes—handling voter eligibility, ballot generation, and real-time vote tallying. Studies such as Hardwick et al. (2018) and Chondros & Yurdakul (2022) emphasize the role of cryptographic protocols like zero-knowledge proofs and ring signatures in maintaining voter anonymity while ensuring verifiability.

In recent works, scholars have also discussed liquid democracy and AI-assisted decision support as emerging paradigms for digital participation. However, large-scale adoption still faces major barriers: limited scalability, complex legal frameworks, accessibility concerns, and the need for stronger voter authentication mechanisms.

In summary, while blockchain voting research has made significant progress—from public transparency models to privacy-preserving permissioned systems—the field continues to evolve. Current efforts focus on combining advanced cryptography, user-friendly interfaces, and regulatory alignment to create secure, inclusive, and trustworthy electronic voting infrastructures.

## III. BACKGROUND AND BASIC IDEAS

Before jumping into how a blockchain-based electronic voting system works, it's helpful to look at what really matters in any voting process and how blockchain can help solve common problems.

### *A. What Makes an Electronic Voting System Work Well?*

For elections to be fair and trustworthy, the system needs to get a few key things right:

*1) Only the Right People Can Vote* - The system should make sure that only people who are registered and eligible get to vote and only once each.

*2) Votes Stay Secret* - It's really important that no one, not even the election officials, can figure out who voted for whom. This protects voters from pressure and keeps the election honest.

*3) Votes Can't Be Changed or Lost* - Once a vote has been cast, it must stay just like it was. No tampering, no erasing. That's what gives people trust in the results.

*4) Voters Can Check Their Vote Was Recorded* - Even though votes are secret, people want a way to double-check that their vote actually counted in the final tally.

*5) No One Can Force or Buy Votes* - The system should make it impossible for someone to prove how they voted to a third party. This helps stop bribery and threats.

*6) Handles Lots of Votes Quickly* - The system should work smoothly, even if millions of people are voting at the same time.

### *B. Why Use a Permissioned Blockchain?*

Blockchain, in simple terms, is like a shared, digital notebook that everyone can trust because it's very hard to change what's written once it's recorded. But regular public blockchains like Bitcoin's have problems: they can be slow, and they show transaction details that might give away private info. That's why many e-voting systems use what's called a permissioned blockchain:

- Only Trusted Groups Run It Instead of anyone joining and validating votes, only pre-approved organizations handle the network. That keeps things fast and safe.
- Faster Vote Recording Because the group is smaller and trusted, votes get recorded quickly—no long waits.
- Proof That No One Cheated Every now and then, the current state of this private network is saved on a public blockchain. Think of it as stamping a document with an official seal to show it hasn't been altered.

### *C. Smart Contracts: The Referees of the Election*

Smart contracts are just small computer programs running on the blockchain that automatically enforce rules, like a referee watching the game:

- Setting Up the Election They create ballots for each voting district and list who the candidates are.
- Making Sure You Don't Vote Twice When you submit your vote, the smart contract checks you haven't already voted and records it.
- Counting Votes as They Come These programs keep track of the votes in real time, so results are ready as soon as voting ends.

### *D. How Votes Stay Private with Zero-Knowledge Proofs*

Keeping votes secret — even while verifying their validity — uses a neat trick called zero-knowledge proofs:

- When you vote, you send a proof that your vote is legitimate but without showing who or what you voted for.
- Other people can check these proofs quickly to ensure only real votes are counted.
- You get a unique code that lets you check your vote is there, without linking it back to you.

### *E. Liquid Democracy: Giving You More Control*

On top of regular voting, this system can let you delegate your vote on specific issues to someone you trust maybe a friend or community leader:

- You decide if, when, and for what topics to give someone else your vote.
- They vote on your behalf, and everything is tracked to make sure it's done right.
- You can always take back your vote and cast it yourself.

Putting It All Together This system uses a mix of trusted networks, automatic enforcement via smart contracts, strong privacy tools, and flexible voting options. All you need is your government-issued digital ID and a PIN to participate securely and easily.

It's designed to give voters confidence that their voices are private and counted, and to help election officials run faster, more reliable elections without mountains of paper or endless manual checks.

By making voting simple, secure, and transparent, this approach helps bring democracy into the digital age—without losing the human trust it depends on. democracy, letting voters delegate their vote on certain issues to someone they trust. A deep dive into security shows we can fend off majority-takeover schemes, block bogus nodes, and keep the network running even under denial-of-service attacks. Finally, we discuss key-recovery options and privacy safeguards to handle the kinds of hiccups that pop up in real elections.

By combining blockchain's unchangeable record, smart contracts' automation, and easy-to-use e-ID login, our system aims to deliver digital voting that is fast, fair, and easy for everyone—no special hardware or tech skills required.

## IV. PROPOSED BLOCKCHAIN-BASED E-VOTING SYSTEM

Creating an electronic voting system that people truly trust isn't just about the tech — it's about building a process that feels secure, private, and simple to use. Inspired by the latest studies and real-world trials, this system blends blockchain technology with everyday tools like digital ID cards and PIN codes to ensure everyone can participate with confidence.

### *A. Who is involved?*

There are a few key players in this election setup:

*1) Election Officials:* They handle the behind-the-scenes work — setting up the election, making sure the voter list is accurate, and keeping an eye on things to prevent any funny business.

*2) Voters:* Regular people who prove who they are with their government-issued digital ID and a PIN, then pick their candidate on a website or app.

*3) District Validators:* Think of these as trustworthy helpers spread out across voting districts. They double-check each vote and add it to a secure, shared record.

*4) Network Helpers:* These folks keep the validators connected and talking smoothly but don't store any votes themselves.

### *B. How Does Voting Work?*

Here's the journey from election start to finish:

*1) Getting Ready:* Election officials use a special app to create ballots for each district, listing candidates and deciding when voting opens and closes. This setup launches small "smart contracts" — little programs on the blockchain that manage the ballots automatically.

*2) Who Votes:* Officials confirm who can vote through trusted identity services. Every eligible voter gets a unique digital "wallet" for the election, keeping their information private but allowing them to participate.

*3) Casting Your Vote:* On election day, you log in with your digital ID and PIN. You see your district's ballot, pick your candidate, and confirm your vote by re-entering your PIN. The district validators check your vote, and once enough agree, your vote is recorded.

*4) Proof Your Vote Counts:* Right after voting, you get a unique code — a receipt. You can use this code later to verify your vote was counted without anyone seeing how you voted.

*5) Counting and Results:* Smart contracts tally votes as they come in. When the election ends, the final numbers are made public right from the blockchain, so anyone can check but no one can cheat.

### *C. Why This Approach Works*

*1) Safety First:* By limiting who runs the network, it's practically impossible for someone to sneak in fake votes or change results after the fact.

*2) Privacy Protected:* Advanced cryptography means no one can link a vote back to the voter, which also stops coercion or vote selling.

*3) Open and Transparent:* Every voter has a way to verify their vote is there. Plus, lots of people can audit the process without exposing any secrets.

*4) Ready for Big Elections:* Dividing the work into districts lets the system handle huge numbers of votes quickly, so nobody is waiting days or weeks for results.

### *D. The Tech Behind It*

This system stands on modern blockchain platforms like Go-Ethereum, Quorum, and Exonum — each offering strengths in speed, security, and programmability to make the whole thing run smoothly.

### *E. Real-World Hurdles*

*1) Protecting Your Login:* Using IDs and PINs makes voting easy but means lost PINs or stolen cards are risks to watch for. Biometrics might be added later to help fix this.

*2) Fighting Attacks:* The design includes ways to fend off attempts to overload or disrupt the network, keeping voting open for everyone.

*3) Following the Rules:* While the tech is solid, laws might need updating to officially recognize blockchain voting and its unique features.

### *F. Wrapping Up*

This system shows that secure digital voting is possible without confusing or expensive tools. By combining proven blockchain features with thoughtful design and everyday authentication methods, it opens the door to elections that are not only trustworthy and private but welcoming to all kinds of voters. It's a big step toward making voting easier, faster, and more secure for the future

## V. SYSTEM DESIGN AND ARCHITECTURE

### *A. Overview*

This section describes the high-level architecture, core components, data flows, security controls, and failure-handling design for a permissioned blockchain-based electronic voting system. The design emphasizes: (a) strong voter privacy, (b) integrity and immutability of ballots, (c) scalability by district partitioning (sidechains), and (d) auditable anchoring to a public ledger.

### *B. High-level components*

TABLE I.

| Component | Description | Primary Responsibility |
|---|---|---|
| Voter Client (Web/Mobile) | UI for voter interactions. Authenticates e-ID + PIN, displays ballots, submits votes | Authentication, vote creation, receipt display |
| Identity Provider (IdP) | Government or trusted service that validates e-identities | Issue/verify e-ID assertions, revoke access |
| District Sidechain | Permissioned blockchain instance per district/region | Fast vote recording, smart contracts enforcement |
| Validator Nodes | Operated by trusted institutions (election commissions, NGOs) | Consensus, block creation, validation |
| Smart Contracts | On-sidechain contracts that enforce election rules | One-person-one-vote, ballot lifecycle, tallies |
| Auditor / Observer Tools | Read-only interfaces to verify vote counts and anchors | Public audits, receipt verification using ZKP proofs |
| Admin Console | Election officials' dashboard | Election configuration, whitelist management |
| Monitoring & DDoS Mitigation | Network and application monitoring | Alerting and throttling, mitigations |
| Aggregator & Anchor Service | Periodically summarizes sidechain state and anchors to public chain | Produce Merkle root and publish proof on public ledger |

### C. Logical Architecture

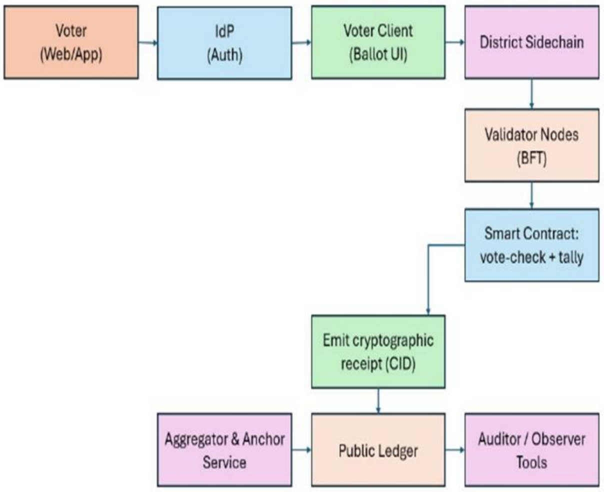


Fig 1.

### D. Data Structures and Tables

#### 1) Key On-chain Data Structures

TABLE II.

| Name | Purpose |
|---|---|
| Voter Commitment | Stores a privacy-preserving commitment that a voter cast a ballot |
| BallotTx | Actual on-chain transaction representing an encrypted vote |
| TallyState | Maintains running tally per election contract |
| AnchorRecord | Maps sidechain state to public ledger proof |

#### 2) Off-chain/Supporting Tables (for Admin/Auditors)

TABLE III.

| Table | Notes |
|---|---|
| VoterRegistry (off-chain, hashed) | Raw identity info is never stored on-chain in cleartext |
| NodeOperators | For governance and accountability |
| AuditLog | Immutable logs anchored regularly |

### E. Sequence of Operations

#### 1) Preparation Phase

a. Admin Console provisions election metadata (districts, candidates, times). Smart contracts are deployed to each district sidechain.

b. IdP issues short-lived e-ID tokens to eligible voters.

c. Validator nodes are registered and governance policies are set.

#### 2) Voting Phase

a. Voter authenticates via IdP and requests ballot from the Voter Client.

b. Client constructs an encrypted vote and generates a ZKP showing vote validity (without revealing selection).

c. BallotTx is submitted to the sidechain; validators run consensus (BFT-style) and append the transaction.

d. Smart contract verifies ZKP, checks registry for double-vote, updates TallyState, and emits VoterCommitment (receipt).

#### 3) Anchoring Phase

a. Aggregator periodically computes a Merkle tree root over recent sidechain blocks and publishes AnchorRecord to the public ledger.

b. Observers verify AnchorRecord to confirm no retroactive changes.

#### 4) Audit/Result Publication

a. At close, final tallies are published on-sidechain and anchored. Auditors validate ZKPs and anchors; voters may verify receipts.

### F. AI Integration and Implementation

To enhance security, usability, and resilience, the proposed system integrates several AI subsystems that operate alongside the permissioned blockchain and cryptographic primitives. The AI layer is designed to (a) detect anomalous behavior and potential fraud in near real-time, (b) strengthen identity verification while preserving privacy, and (c) assist voters with usability and accessibility through intelligent interfaces — all while using privacy-preserving machine learning techniques.

#### 1) Anomaly & Fraud Detection

A federated anomaly detection pipeline runs on validator and monitoring nodes to flag suspicious patterns such as bulk voting from a single IP/subnet, repeated failed authentications, unusual vote submission rates, or abnormal validator behavior. Input features include per-wallet transaction frequency, timestamp entropy, geolocation clusters (coarse, not exact coordinates), device fingerprint hashes, and ZKP validation failure rates. A gradient-boosted decision tree (e.g., XGBoost) or lightweight ensemble is recommended for production due to interpretability and low-latency scoring. Detected anomalies trigger graduated responses: increased verification, temporary rate-limits, or alerts for human auditors.

#### 2) Identity Assurance & Liveness

For stronger authentication, AI models perform liveness and biometric verification when voters opt in (face or fingerprint). These models run client-side (on-device) to avoid sending raw biometric data to servers. Matching is achieved with on-device embeddings and secure verification via the IdP. To avoid privacy leakage, the system uses secure aggregation for server-side model updates and exposes only signed verification tokens to the blockchain layer; raw embeddings never leave the voter device.

#### 3) Privacy-Preserving Model Training

To maintain voter privacy while improving models, the system adopts federated learning with secure aggregation and differential privacy (DP) noise injection during updates. This combination allows validators and IdPs to improve global models (e.g., anomaly detectors and liveness models) without exposing individual vote or biometric data. Model update aggregation uses homomorphic-friendly secure aggregation or trusted execution environments (TEEs) at validator nodes where available.

*4) Voter Assistance & Accessibility*

A lightweight NLP assistant (small transformer or distilled model) provides step-by-step guidance, language translation, and accessibility features (screen-reader friendly prompts). This component runs server-proximal with strict telemetry limits and can be disabled for privacy-conscious voters.

*5) Deployment & Runtime Architecture*

Edge: Client apps run liveness and basic checks locally and produce signed assertions. Validator/Monitoring Nodes: Host anomaly detectors and aggregate model updates via federated endpoints. Critical AI decisions (e.g., temporary suspension) must be explainable and logged. Audit Trail: All AI-triggered events generate signed, immutable audit entries (not voter choices) on the sidechain to preserve transparency of automated interventions.

*6) Metrics & Evaluation*

Key evaluation metrics include true positive/false positive rates for anomaly detection, model latency (ms), on-device inference time, privacy budget ε for DP, and system throughput impact. Continuous evaluation with synthetic and historical (anonymized) data, combined with offline red-team testing, should be adopted before deployment.

*7) Explainability & Governance*

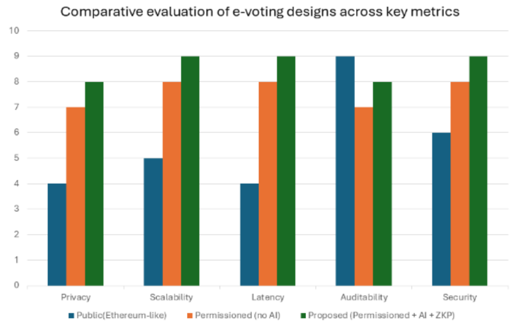
Given the high stakes, AI decisions must be auditable. Use model explainers (SHAP/LIME) for post-hoc analysis and keep a human-in-the-loop for recovery and appeals. Governance includes model update review by independent auditors and a public description of features and privacy protections in user documentation.

### *G. Security Design & Protocol Choices*

- Consensus: BFT (Practical Byzantine Fault Tolerance) or permissioned PoA to ensure liveness and throughput with a limited node set.
- Privacy: Use cryptographic commitments + zero-knowledge proofs (e.g., zk-SNARKs or Bulletproofs) so validators can prove ballot correctness without learning choice.
- Encryption: End-to-end encryption of ballot payloads with ephemeral keys; only commitments are on-chain.
- Anchoring: Anchors on a public chain provide tamper-evidence; keep anchor frequency balanced (e.g., every 5–15 minutes) to reduce cost and improve audit windows.
- Key Recovery & PIN: Offer social or multi-party recovery for lost credentials—avoid single-person recovery to reduce coercion risks.
- DDoS & Availability: Rate-limit client requests, geo-redundant validator nodes, and use CDN for voter-facing UI.

### *H. Performance & Scalability Considerations*

- Partitioning: One sidechain per district (or group of districts) enables parallel processing and reduces consensus load.
- Batch Anchors: Group sidechain blocks into batches for anchoring to the public ledger to lower gas/fees.
- State Channels / Layer-2: Consider temporary ephemeral channels for very high-frequency use-cases (e.g., instant polls).



### *I. Failure Modes & Recovery*

TABLE IV.

| Failure Mode | Mitigation |
|---|---|
| Node crash | Automatic failover to other validators; health checks and alerts |
| Double-spend/double-vote attempt | Smart contract validation rejects duplicates; ZKP & registry checks |
| Compromised IdP | Multi-factor IdPs, cross-check with alternate identity sources, revoke tokens quickly |
| Malicious validator | Multi-org governance with slashing or removal after audit |

## VI. SECURITY AND LEGAL ISSUES

Building a digital voting system is about more than just technology; it's about making sure people feel safe and confident that their vote counts and stays private. This section dives into the main security risks we need to watch out for and the legal hurdles a blockchain voting system might face.

### *A. What Could Go Wrong?*

- DDoS Attacks Imagine someone trying to jam the system by flooding it with fake requests. Because our network only lets trusted nodes handle votes, it's tough for attackers to overwhelm the system. If anyone tries, it's easy to spot and stop them quickly.
- Keeping Voter Identity Safe Voting depends on making sure you are who you say you are. We use your government-issued digital ID and a PIN you know. But if someone else finds out your PIN, they could try to vote in your place. Adding things like fingerprint or face recognition might help fix this down the road.
- Fake Nodes Trying to Take Over (Sybil Attacks) Some bad actors might try to fake a bunch of validators to

control the network. Because only trusted, verified groups can run nodes in our system, it's basically impossible for anyone to sneak in fake players.

- What If One Group Controls Everything (51% Attack)? One fear with blockchains is if one person or group controls most of the validators, they could mess with the votes. Our solution spreads control across different trustworthy organizations, making sure no one side has too much power.
- Keeping Votes Secret, Stopping Pressure Privacy is key. Using smart cryptography, we make sure no one can link a vote back to the voter. That stops people from being forced to prove how they voted or selling their vote.

### B. *What About The Law?*

- Following Election Policies Most election rules were written for paper ballots. New laws will need to catch up so digital votes are recognized and protected under the law.
- Balancing Openness with Privacy Everyone should be able to see election results and trust them, but voter choices must remain secret. Our design uses math magic to make both happen at once.
- Checking and Fixing Mistakes With blockchain's "can't be changed" record, audits are easier than ever. But rules must be in place for handling disagreements or errors, so everyone plays fair.
- Making Sure Everyone Can Vote The system should work for all voters—even those who aren't tech-savvy or don't have easy internet—so laws should ensure no one gets left out.

### C. *Wrapping Up*

This isn't just about coding a system; it's about building trust. By limiting who runs the network and using strong privacy tools, we protect against many common threats. Still, the tech needs to fit into existing laws and social expectations. Updating policies and educating the public will be as important as any technical fix. When security and legality go hand in hand, we'll have a voting system people actually believe in.

## VII. RELATED WORK AND COMPARATIVE OVERVIEW

To truly appreciate the blockchain-based voting system we propose, it helps to take a quick look at what others have done and how our approach fits in. Over the years, researchers and developers have explored various ways to use blockchain technology for elections, each with its own strengths and challenges.

### A. *Early Blockchain Voting Attempts*

Some of the first experiments used public blockchains like Ethereum to record votes openly. Projects such as "Follow My Vote" aimed to make voting transparent and verifiable. However, because these blockchains are open to everyone, voter privacy was hard to protect. Also, public chains can slow down during busy times and aren't always easy to scale for big elections.

### B. *Moving to Permissioned and Hybrid Models*

To get around those issues, newer systems use permissioned blockchains. These are private or semi-private networks where only trusted groups—like government agencies or universities—validate votes. This makes voting faster and helps keep voter information private. Some systems combine permissioned chains with public blockchains by periodically submitting secure summaries to public ledgers, giving the best of both worlds: privacy, speed, and transparency.

### C. *Smart Contracts as Election Managers*

Smart contracts—automated programs on blockchains—have changed the game by handling tasks like creating ballots, enforcing voting rules, and counting votes automatically. Different platforms use different programming languages to build these contracts, but they all aim to make the election process smoother and more trustworthy.

### D. *Keeping Votes Private with Cryptography*

Another big focus has been on ways to keep your vote secret while still proving it was counted correctly. Techniques like zero-knowledge proofs let voters do just that, without revealing who they voted for. Researchers have worked hard to make these methods efficient and suitable for real elections.

### E. *What's Next?*

While a lot of progress has been made, there's still work to do. Making sure anyone can use the system easily, integrating biometric logins, and getting legal approval in various countries will take time. Our approach brings together many of today's best ideas to create a voting system that's not only secure and private but also user-friendly and flexible.

## VIII. CHALLENGES IN IMPLEMENTING BLOCKCHAIN-BASED E-VOTING SYSTEMS

Even though blockchain technology offers great promise for making elections more transparent, secure, and trustworthy, putting it into real-world use isn't without its hurdles. Let's talk about some of the main challenges that show up when trying to build and run blockchain voting systems for large-scale elections, based on what researchers and experts have found.

### A. *Handling Large Numbers of Votes (Scalability)*

One major problem is how to manage a huge volume of votes at once—something national elections throw at a system. Traditional public blockchains like Bitcoin and Ethereum weren't built to process millions of transactions quickly. This leads to slowdowns and even high fees during busy times, which is a big no-no when people expect fast, reliable results. Permissioned blockchain networks and hybrid systems help speed things up, but we still need better ways—like breaking the ledger into parts (called sharding) or speeding up consensus—to keep everything running smoothly as the election grows.

### B. Keeping Votes Secret While Being Transparent (Privacy vs. Openness)

It might sound like a paradox, but voters want their choices to be private while also being confident that votes aren't tampered with. Blockchain, by nature, shares transaction details publicly, which could risk exposing private info. That's where clever math comes in—techniques like zero-knowledge proofs let the system prove that each vote is valid without showing who the vote was for. Getting these privacy protections solid and foolproof is tricky, but absolutely essential to keep voters safe from intimidation or vote-buying.

### C. Defending Against Cyberattacks (Security Risks)

Blockchain itself is tough to hack, but the other parts of the voting ecosystem—like the apps people use or their devices—can be vulnerable. Past blockchain voting pilots have shown security weaknesses, especially with mobile voting applications. Things like malware, phishing scams, or even bugs in the software can open doors to fraud. That means a lot of attention has to go into building strong defenses, testing thoroughly, and keeping voters' devices secure.

### D. Making It Easy for Everyone to Vote (Accessibility and Usability)

For blockchain voting to work widely, the system needs to be simple and clear for all voters, regardless of their tech skills. Not everyone is comfortable with digital tools, and some places might lack fast internet or access to electronic ID cards. Educating voters, designing straightforward apps, and ensuring alternatives for those without technology will be key to making sure nobody is left out.

### E. Legal and Policy Challenges

Election laws around the world mostly focus on paper ballots or older technology, so they don't always cover digital or blockchain voting. To make blockchain elections official and accepted, lawmakers will need to update regulations, define standards for privacy and auditability, and clearly handle how to settle disputes or recounts. This process can be slow and requires political will.

### F. Costs and Resources

While blockchain can save money in the long run by cutting out middlemen and paper handling, setting up a blockchain election system isn't cheap. Developing the software, running validator nodes, training staff, and educating voters all require money and expertise. Some places might hesitate or struggle to cover those costs without outside help or phased rollouts.

### G. Building Public Trust (Transparency and Openness)

Even with all the good tech, voters need to trust the system. If parts of the process are hidden behind closed doors or rely on proprietary software that no one can examine, skepticism will grow. To keep trust high, open-source code, independent audits, and clear communication about how the system works are essential.

In short, while blockchain offers powerful tools to make elections safer and more transparent, we still face many real-world challenges in scaling, privacy, security, accessibility, law, cost, and trust. Addressing these thoughtfully will be crucial to turning blockchain voting from a promising idea into a reliable, everyday reality for voters everywhere.

## IX. FUTURE WORK AND RESEARCH DIRECTIONS

Blockchain-based electronic voting has shown great potential, but there are still many areas where further work and innovation are needed before it can become mainstream. This section explores some promising paths forward, inspired by findings from the research studies and real-world experiments.

### A. Improving Authentication and Usability

One key area is making voter authentication both more secure and user-friendly. Today's systems often rely on government-issued electronic IDs and PINs—but these can be lost, stolen, or forgotten. Integrating biometrics like fingerprints or facial recognition could offer stronger security while keeping the process simple. Additionally, developing intuitive interfaces that guide voters smoothly through the process—especially those with little tech experience—is critical to encourage widespread adoption.

### B. Enhancing Privacy Protections

While zero-knowledge proofs and similar cryptographic tools have come a long way, there's still room to make these methods faster and easier to implement on a large scale. Research into new privacy-enhancing technologies can help ensure that votes remain secret without sacrificing verification or audit capabilities.

### C. Scalability and Performance Optimizations

Handling national-scale elections means processing millions of votes quickly and reliably. To reach this goal, future systems may adopt techniques such as sharding, sidechains, or layer-two protocols that allow parallel processing of votes. Exploring these innovations will improve transaction speed and reduce costs.

### D. Addressing Accessibility and Inclusion

Ensuring everyone can participate, regardless of digital literacy or internet access, is a priority. Future developments should explore offline voting options, simple devices for vote casting, and educational programs that empower underserved communities to engage safely and confidently.

### *E. Security Audits and Formal Verification*

Continuous security testing and formal verification of smart contracts and blockchain protocols are essential. As more real-world deployments occur, lessons learned will inform stricter standards and better tooling to catch vulnerabilities before they become problems.

### *F. Exploring Liquid Democracy and New Voting Paradigms*

Liquid democracy—letting voters delegate their vote on specific issues—offers exciting possibilities for more active and nuanced participation. Future work can refine how delegation is managed securely and transparently on the blockchain, and how this model can coexist alongside traditional voting.

### *G. Public Education and Trust Building*

Even the best technology won't succeed without public confidence. Future efforts must include education campaigns to demystify blockchain voting, explain its benefits and safeguards, and listen to voter concerns. Transparency, openness, and community involvement will be key pillars.

In summary, the future for blockchain-based elections is bright but calls for ongoing innovation across technology, policy, and outreach. By focusing on making systems easier to use, faster, more secure, and legally sound, while building trust with voters, researchers and practitioners can help bring the promise of digital democracy to life—fairly and confidently—for everyone.

## X. CONCLUSION

Electronic voting powered by blockchain technology holds incredible promise to reshape how elections are conducted. It offers a path toward faster, more transparent, and more secure voting processes—one where each vote is recorded immutably, counted accurately, and remains private. However, as promising as the technology is, turning this vision into reality requires carefully balancing technical innovation with real-world considerations like usability, legal frameworks, and voter trust.

Drawing from current research and practical experiments, this paper has outlined a comprehensive blockchain-based voting system that combines permissioned networks, smart contracts, and cryptographic tools like zero-knowledge proofs. Together, these elements provide a strong foundation for secure and verifiable digital elections. By using government-issued electronic IDs and PINs, the system ensures only eligible voters participate and allows voters to verify their vote was counted without revealing how they voted.

Despite the many benefits, several challenges remain: achieving sufficient scalability to handle nationwide elections smoothly, protecting voter privacy while maintaining auditability, securing voter devices against attacks, and updating election laws to accommodate digital methods. Additionally, user interface simplicity and inclusiveness are key to making sure every eligible voter can easily and confidently use the system. Future work should focus on improving authentication, integrating biometrics, increasing throughput, and aligning digital solutions with existing electoral processes and regulations. Building public trust through transparency, open-source development, and education will be just as important as the technology itself.

In conclusion, blockchain-based e-voting is not merely a futuristic idea but a feasible evolution of democratic participation. By addressing technical, legal, and social challenges together, we can move towards elections that are not only quicker and cheaper but fundamentally more trustworthy and accessible. The road ahead calls for collaboration between computer scientists, policymakers, election officials, and the public—a shared journey toward modernizing democracy for the digital age.

## REFERENCES

[1] B. Adida, O. de Marneffe, O. Pereira, and J.-J. Quisquater, "Electing a University President using Open-Audit Voting: Analysis of real-world use of Helios," IACR, 2011. [Online]. Available: https://www.usenix.org/legacy/events/evtwote10/tech/full_papers/Adida.pdf

[2] S. Akinbohun, S. Apeh, E. Olaye, and D. Ogbeide, "Literature review of blockchain-based voting systems: Framework and concept," Int. J. Eng. Technol., vol. 20, no. 1, pp. 72–83, 2023. [Online]. Available: https://www.researchgate.net/publication/375462419

[3] S. Bartolucci, P. Bernat, and D. Joseph, "SHARVOT: Secret SHARe-based VOTing on the blockchain," arXiv:1803.04861, 2018. [Online]. Available: https://arxiv.org/abs/1803.04861

[4] J.-M. Bohli, J. Müller-Quade, and S. Röhrich, "Bingo Voting: Secure and coercion-free voting using a trusted random number generator," in VOTE-ID, 2007, pp. 111–124. [Online]. Available: https://en.wikipedia.org/wiki/Bingo_voting

[5] V. Buterin, "Ethereum whitepaper: A next-generation smart contract and decentralized application platform," Ethereum Foundation, 2014. [Online]. Available: https://ethereum.org/en/whitepaper/

[6] O. Chondros and A. Yurdakul, "ElectAnon: A blockchain-based, anonymous, robust and scalable ranked-choice voting protocol," arXiv:2204.00057, 2022. [Online]. Available: https://arxiv.org/abs/2204.00057

[7] Exonum, "Blockchain framework for business," Bitfury Group. [Online]. Available: https://exonum.com

[8] Follow My Vote, "Online voting with blockchain technology." [Online]. Available: https://followmyvote.com

[9] F. S. Hardwick, A. Gioulis, R. N. Akram, and K. Markantonakis, "E-Voting with Blockchain: An E-Voting Protocol with decentralisation and voter privacy," arXiv:1805.10258, 2018. [Online]. Available: https://arxiv.org/abs/1805.10258

[10] Helios Voting, Wikipedia. [Online]. Available: https://en.wikipedia.org/wiki/Helios_Voting

[11] X. Huang, J. Xu, Y. Wang, and H. Zhang, "Blockchain-based e-voting: A survey," IEEE Access, vol. 9, pp. 124765–124781, 2021. doi: 10.1109/ACCESS.2021.3109886

[12] M. Jafar and M. Ab Aziz, "Blockchain-based e-voting systems: Benefits and challenges," Int. J. Comput. Appl., vol. 176, no. 1, pp. 1–8, 2020. doi: 10.5120/ijca2020919954

[13] B. J. D. Kalyani, J. K. Modadugu, and S. Neelima, "Blockchain-based decentralized voting system with SHA-256 algorithm and facial recognition," 2025. [Online]. Available: https://www.researchgate.net/publication/342932007

[14] G. Kappos, H. Yousaf, A. Piotrowska, S. Kanjalkar, and S. Meiklejohn, "Blockchains and voting: A framework for analysis and evaluation," IEEE Secur. Privacy, vol. 19, no. 5, pp. 28–37, 2021. doi: 10.1109/MSEC.2021.3082237

[15] LiquidFeedback, Wikipedia. [Online]. Available: https://en.wikipedia.org/wiki/LiquidFeedback

[16] M. Pawlak, P. Teisseyre, and A. Misztal, "Security of blockchain-based electronic voting systems," Future Internet, vol. 13, no. 5, p. 127, 2021. doi: 10.3390/fi13050127

[17] Quorum, "Enterprise blockchain platform," ConsenSys. [Online]. Available: https://consensys.net/quorum/

[18] A. Russo, A. F. Anta, M. I. G. Vasco, and S. P. Romano, "Chirotonia: A scalable and secure e-voting framework based on blockchains and

linkable ring signatures," arXiv:2111.02257, 2021. [Online]. Available: https://arxiv.org/abs/2111.02257

[19] T. Sasaki and S. Matsuo, "A survey on blockchain consensus with a performance comparison of PoW, PoS, and BFT," J. Internet Serv. Inf. Secur., vol. 9, no. 3, pp. 1–20, 2019. [Online]. Available: https://jisis.org/vol9/no3/p1.html

[20] B. Shahzad and J. Crowcroft, "Trustworthy electronic voting using adjusted blockchain technology," IEEE Access, vol. 7, pp. 24477–24488, 2019. doi: 10.1109/ACCESS.2019.2895670

[21] S. Siri, "Radical blockchain democracy," Wired, 2018. [Online]. Available: https://www.wired.com/story/santiago-siri-radical-plan-for-blockchain-voting/

[22] "Blockchain for securing electronic voting systems: A survey of architecture, trends, challenges," Cluster Comput., 2024. doi: 10.1007/s10586-024-04709-8

[23] O. Taş and M. D. Tanrıöver, "A survey of blockchain-based electronic voting," Digital Gov.: Res. Pract., vol. 1, no. 2, pp. 1–21, 2020. doi: 10.1145/3396959

[24] "West Virginia's mobile voting pilot and blockchain," Time, 2019. [Online]. Available: https://time.com/5717479/mobile-voting-accessibility/

[25] "Australia's Flux party blockchain democracy," Time, 2016. [Online]. Available: https://time.com/4375991/flux-blockchain-bitcoin-democracy-politics-australia/

[26] Voatz, Wikipedia. [Online]. Available: https://en.wikipedia.org/wiki/Voatz

[27] "Why you can't just vote on your phone," New Yorker, 2020. [Online]. Available: https://www.newyorker.com/tech/annals-of-technology/why-you-cant-just-vote-on-your-phone-during-the-pandemic

[28] G. Zyskind, O. Nathan, and A. Pentland, "Decentralizing privacy: Using blockchain to protect personal data," in IEEE Secur. Privacy Workshops, 2015, pp. 180–184. doi: 10.1109/SPW.2015.27

[29] "Blockchain-based e-voting systems: A technology review," Electronics, vol. 13, no. 1, p. 17, 2023. doi: 10.3390/electronics13010017

[30] "A systematic literature review and meta-analysis on scalable blockchain-based electronic voting systems," Frontiers in Blockchain, 2022. [Online]. Available: https://pmc.ncbi.nlm.nih.gov/articles/PMC9572428/

[31] "Blockchain-based electronic voting systems: A case study in Morocco," J. Inf. Secur. Appl., vol. 77, p. 103620, 2024. doi: 10.1016/j.jisa.2023.103620

[32] "A comprehensive analysis of blockchain-based voting systems," in Proc. ACM Int. Conf., 2025. doi: 10.1145/3723178.3723275

[33] "Blockchain for electronic voting system: Review and open challenges," PeerJ Comput. Sci., 2021. [Online]. Available: https://pmc.ncbi.nlm.nih.gov/articles/PMC8434614/

[34] "Blockchain democracy: Sovereign system's risks," Axios, 2018. [Online]. Available: https://www.axios.com/2018/08/18/dreams-of-democracy-on-the-blockchain