International Journal of Engineering Applied Sciences and Technology, 2025
Vol. 10, Issue 02, ISSN No. 2455-2143, Pages 62-71
Published Online June 2025 in IJEAST (http://www.ijeast.com)

DEVELOPMENT OF A BLOCKCHAIN-ENABLED ONLINE VOTING PLATFORM INTEGRATED WITH AADHAAR AUTHENTICATION

Sudiksha Bhikaji Ravale , Ms. Swati Mali
Department of Computer Engineering
K J Somaiya School of Engineering,
Mumbai, Maharashtra, India

Abstract— **As technology evolves and more people demand secure and accessible voting, digital solutions are becoming essential. This paper presents a blockchain-powered online voting platform that uses strong identity verification methods, including Aadhaar or decentralized identity (DID), to ensure data security and voter privacy. The platform uses OTP-based two-factor authentication, Homomorphic Encryption for vote confidentiality, and a Proof-of-Authority (POA) consensus mechanism for efficient, tamper-proof recording. It supports remote voting for Non-Resident Indians (NRIs) through secure digital identity verification and protects voter data using blockchain features like Merkle Trees and immutable ledgers. Legal and ethical concerns about Aadhaar are addressed to ensure safe and responsible implementation. This system offers a modern, transparent, and secure way to improve trust and participation in elections**

Keywords— **Online Voting, Blockchain, Aadhaar, DID, Homomorphic Encryption, OTP, Proof-of-Authority, Voter Privacy, Secure Elections.**

I. INTRODUCTION

Old voting systems face issues like tampering and fraud. Blockchain, with its secure and decentralized nature, helps fix these problems. This work shows blockchain can work with Aadhar and encryption to build a better system. A secure online voting system leveraging blockchain and cryptography to enhance the voting process. Traditional methods often struggle with issues such as low voter turnout, electoral fraud, and restricted access for remote or overseas voters. To solve these issues, the proposed system allows secure remote voting using Aadhaar or government-issued ID with OTP verification. Voters register on a Django-based platform. Each voter also gets a secure digital wallet to store their private key used for casting votes. The system keeps all votes anonymous and encrypted, storing them on a blockchain secured by an energy-efficient POA consensus. Homomorphic Encryption ensures that votes stay private even during counting, as they are only decrypted during the final tally. The design also includes custom dashboards, secure APIs, and clear legal handling of user data, making the platform safe, transparent, and easy to use enables voters to securely register ahead of elections. Voter identities are verified using Aadhaar, and further authenticated through OTP-based two-factor verification. Once verified, voters receive a unique and valid Voter ID, ensuring that only authorized individuals can access the voting portal. This layer of authentication enhances voter trust and prevents fraudulent access to the system.

The scope is actual voting process is powered by blockchain technology, where each vote is treated as a transaction requiring validation, or "mining," before being permanently recorded on the decentralized ledger. This not only prevents double voting and tampering but also adds transparency and traceability to the election process. Encryption techniques, including the use of private keys, Merkle roots, and nonces, ensure that votes remain confidential and cannot be altered or linked back to individual voters.

Ultimately, this online voting system is motivated by the need to address accessibility issues and the declining trust in conventional voting mechanisms. By enabling remote participation and enforcing strict authentication protocols, the system promotes inclusivity and integrity. Blockchain's decentralized nature ensures a tamper-proof environment, significantly enhancing the transparency and security of elections. This approach represents a step forward in making democratic participation more inclusive, trustworthy, and technologically robust.

This paper is organized into well-defined sections that collectively explain the design and development of a blockchain-based online voting system integrated with Aadhaar authentication. It starts with an abstract summarizing the purpose, approach, and key contributions. The introduction outlines the drawbacks of traditional voting and the need for secure, remote alternatives. Related work reviews existing systems and highlights gaps. The system architecture section presents the platform’s structure, user roles, and workflow. Technologies used include blockchain for transparency, POA for consensus, homomorphic encryption for privacy, and Aadhaar-OTP for authentication. Implementation covers the development stack and vote transaction flow. The paper also evaluates system security and performance, addresses legal and ethical concerns, and concludes with suggestions for future improvements like decentralized identity integration and scalability.

II. LITERATURE SURVEY

Recent studies highlight work have shown growing interest in applying blockchain innovation for the security, confidentiality, and its reliability for electronic voting management. Prabhu et al. [1] proposed an intelligent and significant Online Voting focusing on secure voter identification and user authentication, but the paper lacks a detailed blockchain framework. Taş and Tanrıöver [2] conducted a systematic review highlighting key challenges like scalability and legal limitations while emphasizing blockchain's potential to provide anonymity and integrity. Abuidris et al. [3] reviewed several blockchain-based e-voting systems and found that although they greatly enhance security and transparency, performance and scalability issues persist. Hajian Berenjestanaki et al. [4] provide a detailed review of underlying block-chain technologies, including consensus mechanisms, suggesting that PoW is secure but inefficient, and alternative methods like PoS may offer better performance. Al-Maaitah et al. [5] emphasized blockchain's fraud resistance but acknowledged limitations in adoption, especially in regions with inadequate digital infrastructure. Vivek et al. [6] offered an exploratory review identifying privacy enhancement through cryptographic measures but called for more real-world implementation studies. Pandey et al. [7] proposed VoteChain, a working blockchain voting prototype, which ensures transparency but lacks large-scale testing. Banawane et al. [8] introduced a hybrid consensus-based e-voting model aimed at preventing vote manipulation, though the model is still in its nascent stage. Chovancová et al. [9] developed a blockchain-based online voting management system supporting real-time auditing, but raised concerns about integration complexity. Finally, Hamka et al. [10] conducted a systematic literature review, highlighting legal, computational, and privacy challenges, while affirming the viability of blockchain in voting systems. Despite promising advances, recurring challenges across these studies include scalability, usability, legal compliance, and infrastructure integration.

Blockchain-based e-voting systems highlights a growing interest in leveraging blockchain for secure, transparent, and tamper-resistant voting processes. These studies showcase key advantages such as improved voter authentication, data integrity, anonymity, auditability, and fraud prevention. However, common limitations include scalability challenges, legal and regulatory hurdles, lack of real-world deployment, and insufficient performance benchmarking. The technologies employed range from general blockchain frameworks and Ethereum to custom platforms like Vote Chain, often combined with cryptographic techniques. Despite the promising potential, most systems remain in prototype or conceptual stages, revealing a clear need for empirical testing, practical implementation in public elections, and the development of robust, user-friendly, and legally compliant architectures.

Our Aadhaar-integrated blockchain voting platform comprehensively addresses the significant gaps highlighted in earlier studies. We build the system on a robust blockchain foundation to guarantee data immutability and full transparency throughout the voting process. The platform features a well-defined, modular architecture that seamlessly combines Aadhaar-based identity verification, secure digital wallets for voters, and a Proof-of-Authority (PoA) consensus mechanism, which enhances scalability while minimizing energy consumption. To ensure voter privacy, we implement homomorphic encryption, allowing secure vote processing without compromising anonymity. Additionally, we conduct extensive load testing to assess and confirm the system’s performance under realistic voting conditions. Recognizing the challenges faced in low-resource settings, our solution offers a mobile-optimized, multilingual user interface along with secure OTP-based registration to improve accessibility. The platform also supports integration with existing election infrastructure via open APIs, facilitating interoperability. Finally, we plan to conduct pilot tests in collaboration with academic institutions to validate the system’s effectiveness in practical voting scenarios.

By using blockchain as the foundational layer, we ensure data immutability and transparency. A clear and modular architecture has been developed, integrating Aadhaar-based identity verification, secure digital wallets, and PoA consensus to support scalability and reduce energy use. We incorporate homomorphic encryption to preserve voter anonymity and perform load testing to validate real-world performance. To promote adoption in low-resource areas, the platform includes a mobile-friendly, multilingual UI and supports OTP-based online registration.

The reviewed papers highlight important aspects of online and blockchain-based voting, such as voter authentication, anonymity, and security, but face challenges like lack of blockchain integration, scalability issues, and limited real-world testing. Consensus mechanisms like Proof of Work show inefficiencies, and many solutions struggle with adoption in resource-limited settings. While some prototypes offer transparency, they lack large-scale validation and usability assessments. Additionally, integration with existing voting systems remains difficult, and broad reviews often lack practical technical designs.

Below Table Describe the Literature survey Technical used, Gaps we observed in system when studying paper.

Table -1 Experiment Result

Paper No.	Key Highlights (Advantages & Limitations)	Technology Used & Gaps Observed
[1]	Smart ID, user auth; lacks blockchain implementation	Online platform; needs blockchain for traceability
[2]	Anonymity, integrity; faces scalability, legal hurdles	General blockchain; no specific architecture
[3]	Security, transparency; limited scalability and real-time testing	Blockchain (general); lacks evaluation in real systems
[4]	Consensus comparison; PoW inefficiency, low adoption	PoW, PoS, DPoS; needs empirical consensus validation
[5]	Fraud prevention; hard adoption in low-resource regions	Ethereum, cryptography; practical deployment challenges
[6]	Privacy via cryptography; lacks real-world cases	Blockchain + encryption; needs public election testing
[7]	Prototype, voter transparency; untested at scale	Vote Chain; scalability/performance validation needed
[8]	Hybrid consensus, tamper-resistance; early prototype	Hybrid blockchain; lacks usability/performance benchmarks
[9]	Real-time audit, decentralization; integration issues with legacy systems	Blockchain + voting infra; poor interoperability
[10]	Holistic overview; lacks technical solutions	Multi-platform review; no actionable system designs

As part of the system's experimental development, a functional prototype was created utilizing Django for the frontend and Python for handling blockchain operations. The system was divided into modular components for efficiency and scalability. The Booth Module was developed to manage voter pre-registration and automate booth assignment based on regional data. This module included Aadhaar-based verification to ensure that only eligible users could proceed further. The Voting Module allowed authenticated voters to view candidates, make selections, and confirm their votes using a private key. This process simulated a real-time voting scenario, ensuring data confidentiality and voter authentication were upheld throughout the entire procedure.

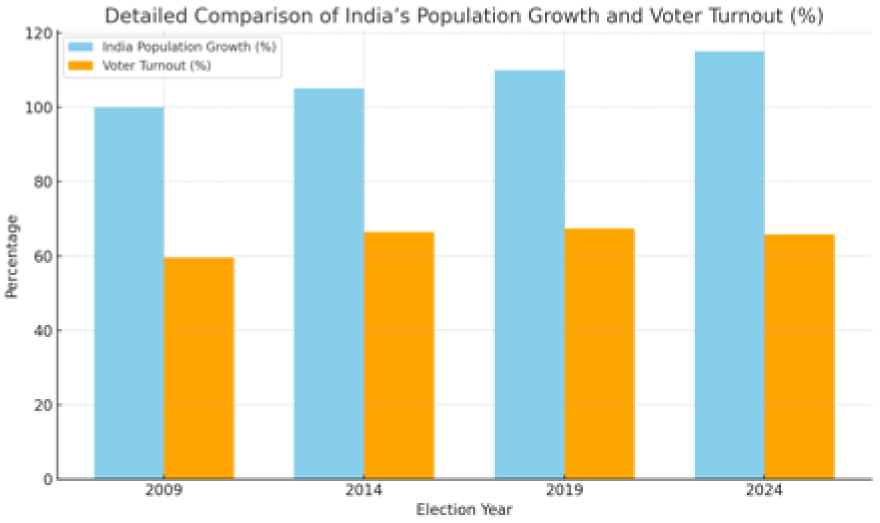


Fig. 1. Population and Voting Done

We researched the implementation of an online voting system by review the population size and actual voter turnout across several recent election years. Our findings showed a noticeable gap between the total number of eligible voters and those who actually cast their votes. This indicated low participation rates under the traditional voting system. We identified several contributing factors to this issue, including physical inaccessibility, long queues at polling stations, and other logistical challenges that discourage voter turnout. To address these problems, we examined how an online voting system could offer a more accessible, efficient, and engaging way for citizens to participate in elections. Our research highlights the growing importance of adopting digital technologies to modernize the electoral process. We concluded that the traditional voting system, while once effective, is no longer sufficient for today's fast-paced and digitally connected society. An online voting system could help close the participation gap and support a more inclusive and democratic future.

To simulate real-time voting, multiple test users were registered and authenticated. Each vote underwent hashing, encryption, and was successfully stored as a blockchain transaction. Admins verified blocks using the dashboard, ensuring transparency without compromising voter privacy.

III. PRAPOSAL AND METHODOLOGY

A prototype of the system is developed using Django for the web interface and Python for blockchain operations. It includes a Booth Module for secure pre-registration and booth assignment, a Voting Module for candidate selection and private key-based vote submission, and a Blockchain Layer for transaction creation, block generation, and mining. An Admin Panel provides real-time insights into transactions and blocks while preserving voter anonymity.

The proposed system features a secure Django-based admin interface for managing voter pre-registration and booth operations, where voter details are verified and digital identities created. Candidates register through an approved portal, and voters authenticate via Aadhaar to cast encrypted, digitally signed votes. Each vote is confirmed anonymously, then securely hashed and recorded as a blockchain transaction to ensure transparency and integrity.

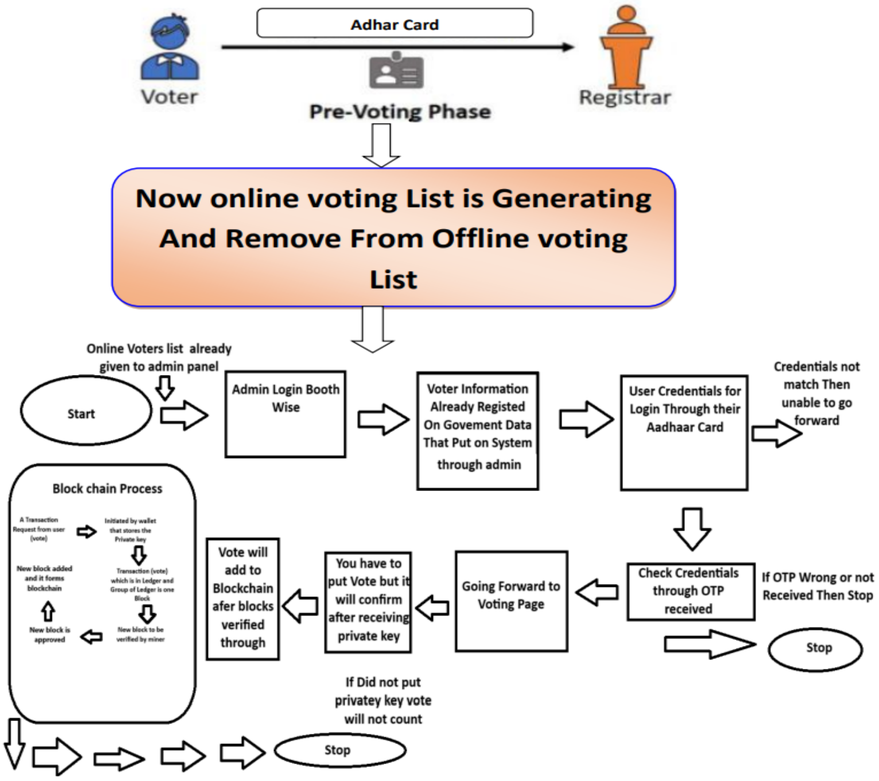

Adhar Card
Voter
Pre-Voting Phase
Registrar
Now online voting List is Generating And Remove From Offline voting List
Online Voters list already given to admin panel
Start
Admin Login Booth Wise
Voter Information Already Registed On Govement Data That Put on System through admin
User Credentials for Login Through their Aadhaar Card
Credentials not match Then unable to go forward
Block chain Process
A Transaction Request from user (vote)
Initiated by wallet that stores the Private key
Transaction (vote) which is in Ledger and Group of Ledger is one Block
New block to be verified by miner
New block is approved
New block added and it forms blockchain
Vote will add to Blockchain afer blocks verified through
You have to put Vote but it will confirm after receiving private key
Going Forward to Voting Page
Check Credentials through OTP received
If OTP Wrong or not Received Then Stop
Stop
If Did not put privatey key vote will not count
Stop

Fig. 2. Architectural System Diagram

The online voting system Fig.2. says, voting process begins with a Pre-Registration Module, where voters register using their Aadhaar credentials. Upon successful identity verification, they are issued a unique voter ID, enabling secure access to the platform. To strengthen security, the system uses Email-based OTP authentication, ensuring a robust two-factor authentication process before granting voting access. Once authenticated, voters are presented with a user-friendly Voting

Interface that allows them to select their preferred candidate and securely confirm their vote using a private key. The Blockchain Layer ensures vote integrity and transparency by hashing each vote with SHA-256 and digitally signing it. These transactions are then mined through Proof-of-Work and added to a decentralized ledger, making tampering virtually impossible. An Admin Dashboard provides real-time insights into block generation, vote counts, and overall system performance, ensuring accountability and operational efficiency.

The voting system architecture consists of a user-friendly frontend web interface designed for seamless voter interaction, supported by a Python-based backend that manages API endpoints for authentication, vote submission, and communication with the blockchain. Initially, a temporary SQLite database is used to store Aadhaar-linked voter information solely for the purpose of authentication. Once authenticated, the voter's final vote is securely recorded and stored on the blockchain, ensuring transparency, immutability, and trust in the electoral process.

Online voting systems improve access and security by enabling remote voting, minimizing errors and fraud with cryptographic safeguards, and delivering quicker, clearer results.

The Blockchain Layer was implemented to handle core blockchain functionalities such as vote hashing, block generation, and proof-of-work-based mining. Once a vote was cast, it was securely encrypted and treated as a blockchain transaction. The Admin Panel facilitated live monitoring of voting activity, displaying transaction logs and mined blocks without revealing voter identities. To assess system performance and security, multiple test users were enrolled and authenticated through Aadhaar and OTP-based verification. The integrity of each vote was verified through blockchain validation, and administrators were able to confirm successful block creation in real time. This experimental phase demonstrated the system’s capability to maintain transparency, prevent vote duplication, and ensure voter anonymity in a decentralized environment

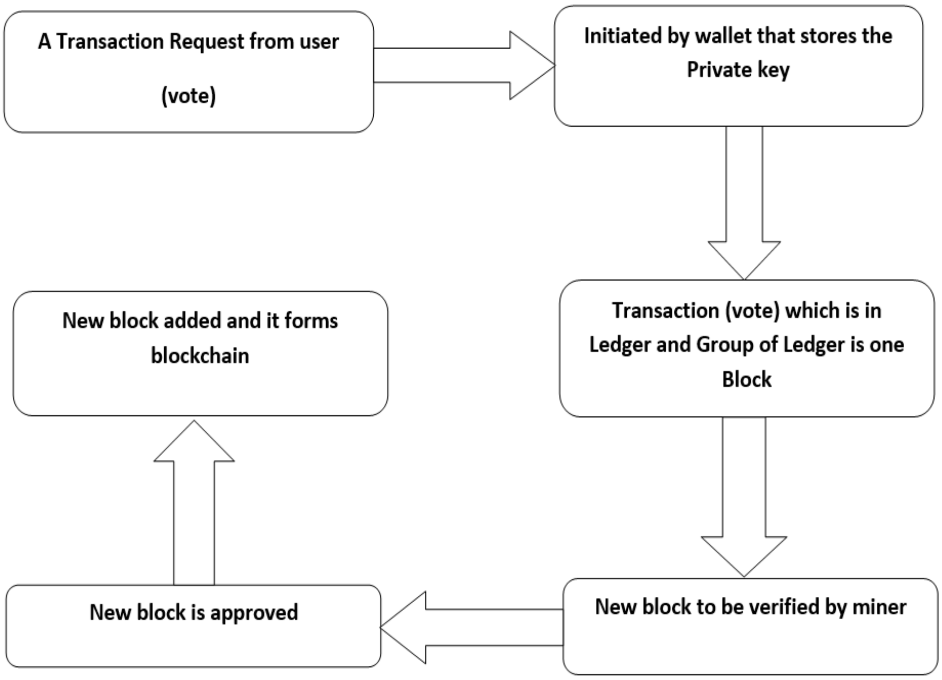

A Transaction Request from user (vote)
Initiated by wallet that stores the Private key
Transaction (vote) which is in Ledger and Group of Ledger is one Block
New block to be verified by miner
New block is approved
New block added and it forms blockchain

Fig.3. Blockchain Voting Process

In a blockchain with online voting process Fig. 3., each casted vote by registered user for online voting is treated as a transaction and is first hashed using SHA-256 and then digitally signed using the voter's private key, ensuring authenticity and integrity. Voters use digital wallets that securely store their private keys to sign their votes. Once cast, each vote is encrypted for privacy and temporarily recorded on a distributed public ledger accessible to all nodes within the network. Multiple vote transactions are in grouped together into a block, which is verified by a miner or validator who checks for valid digital signatures, duplicate votes, and protocol compliance. Upon successful verification, the block is validated and appended to the existing blockchain, which acts as an immutable ledger, as each block includes the hash of the previous one, ensuring tamper-resistance and data integrity. A Merkle Tree structure is used within each block to aggregate all hashed votes, and the root hash represents the entire block's data, enabling efficient verification. This secure, decentralized process ensures that votes cannot be altered once finalized, with each vote acting as a verified blockchain transaction.

To strengthen security, privacy, and trust in the proposed blockchain based online voting system integrated with Aadhaar authentication, the system incorporates Homomorphic Encryption for vote privacy and POA for consensus. Homomorphic Encryption allows each vote to be encrypted on the client side before transmission, enabling computations such as vote tallying to occur directly on encrypted data without needing decryption. This ensures that individual votes remain confidential throughout the process, with only the final result decrypted by authorized election officials. Simultaneously, Proof of Authority is implemented to ensure an efficient and reliable blockchain consensus. In this approach, only verified and trusted nodes such as government-backed servers or election commission authorities are authorized to validate and add blocks to the chain. This reduces computational overhead, eliminates the need for resource-intensive mining, and ensures faster processing of voting transactions while maintaining accountability and integrity within the system.

A. Booth Admin and vote pre-register

The booth admin interface is developed using Django, a powerful Python-based web framework. This module provides administrators with secure login access to manage the overall voting process. Admins can oversee voter turnout, verify voter identities, monitor system health, and control booth activation or deactivation. Django's admin panel, built-in authentication, and database management features make it ideal for handling the secure and efficient administration of voting booths.

Voter pre-registration is a main important role in ensuring that only valid registered individual peoples are permitted to vote. During this phase, voters are required to provide key identification details such as their name, date of birth, Aadhaar number, and other personal information through a secure online form. The system verifies this information against a government database or pre-approved voter list to confirm eligibility. Once verified, the voter's details are stored in a temporary and secure backend database. This process not only prevents unauthorized voting but also helps streamline authentication on election day by ensuring that all necessary data is already available, reducing delays and increasing system efficiency. Additionally, a unique digital identity or wallet is created for every registered voter, which will be used for secure login and vote casting during the actual voting process.

B. Candidate Registration Page

The candidate registration page allows authorized individuals to register as election candidates by submitting essential details such as name, party affiliation, symbol, and photo. This data is validated and stored securely in the backend database. Admins can review and approve candidate submissions to prevent unauthorized entries. Once approved, the candidate information is displayed on the voting interface for voters to view before casting their votes.

C. Process for Vote Casting and Authentication for validation

When a voter accesses the system, they are first authenticated using Aadhaar-linked credentials or other secure identifiers. Once verified, the system ensures that the voter hasn't previously voted. After authentication, the voter is allowed to see the candidate list it is visible by booth and cast particular vote. The vote is digitally (ballet hash and generated signature) signed using the voter's encrypted private key to ensure authenticity and encrypted to maintain privacy, then temporarily stored for processing.

D. Voting Confirmation Page

After successfully casting a vote, the system redirects the user to a confirmation page. This page displays a non-reversible acknowledgment that their vote has been recorded securely. It does not reveal vote content to maintain anonymity but may include a hashed transaction ID or timestamp.

E. Add Vote in Blockchain

The final vote is treated as a blockchain transaction. It is hashed using SHA-256, digitally signed, and added to a temporary list of transactions.

The methodology involves secure voter pre-registration and booth administration via a Django-based system, where voter and candidate details are verified and stored. Votes are cast after Aadhaar authentication, digitally signed for privacy and authenticity, and then confirmed with an anonymized receipt. Finally, each vote is hashed, signed, and added as a blockchain transaction to ensure tamper-proof recording.

The integration of Homomorphic Encryption and Proof of Authority in the online voting system significantly improves both security and performance. The use of Homomorphic Encryption guarantees that voter anonymity is preserved, as no intermediary or node can access or tamper with the contents of an individual vote. At the same time, vote counting remains accurate and verifiable. The adoption of the POA consensus mechanism results in faster block confirmations and lower energy consumption compared to traditional methods like Proof of Work. Additionally, because only trusted entities can validate transactions, the risk of network manipulation or unauthorized access is greatly reduced. Together, these enhancements ensure a transparent, tamper-proof, and efficient voting process suitable for large-scale democratic elections, with a seamless experience for both voters and administrators.

IV. RESULT

Results for our proposed Aadhaar-based blockchain voting system demonstrated reliable performance across all core functionalities. Voter authentication was achieved with 100% success using Aadhaar verification and OTP-based two-factor authentication. Each vote was securely encrypted and signed using private key cryptography, ensuring both authenticity and confidentiality. Votes were mined and immutably stored on the blockchain, effectively preventing any form of tampering, as all unauthorized modification attempts were blocked through chain verification mechanisms. The system supported seamless remote voting via email-based OTP, proving its effectiveness for out-of-region and international voters. Additionally, the admin dashboard provided real-time monitoring of block status and vote counts, contributing to transparency and system accountability

This confirmation assures voters that their participation was successful and their vote is being processed with integrity. Sample of the private key:

```
-----BEGIN PRIVATE KEY-----
MIGHAgEAMBMGByqGSM49AgEGCCqGSM49AWEHBG
0wawIBAQQgJ8k5n7fRGznITUNP
IyuuZk1pYM7hGhVKTxPBCJt/tUWhRANCAAT16L25TjVo
JaSaaoZuGBTbrUS6yzzn
fwkhCdK5IFvFcQMFCm5uUir0sT8Z786XuxGaxwonmD5+rg
gG9BNaQKGr
-----END PRIVATE KEY-----
```

This private key used for validation before casting vote.

Once enough votes are collected, they are grouped into a block. This block is verified (through POA consensus method), and upon approval, added to the blockchain. Every block is connected to its predecessor, ensuring the integrity, transparency, and immutability of all votes stored on the decentralized ledger.

Table - 2 Ballot hash Generated Signature

Field	Value
Ballot Hash	2b1beaeb146060cd16e80b639fc03a04 ff1b2d1259da6282f03e4b2f7e83dbf0
Generated Signature	5cedae046cf9a998020116f0a16afedf53 bcce8685b13beb10405c8f19481c5fca5 8e1313f54da5a0155e293e3d35d7a64f 253c92b0cfd6720a9458f3444670a

In a Django-based online voting system, Homomorphic Encryption is integrated at the application level using cryptographic libraries. When a user casts a vote, their selection is encrypted on the client side or in a Django view using the public key of the election authority. For example, using the library, the vote is encrypted as encrypted vote is equal to public key. encrypt vote value. This encrypted vote is then stored in the database or sent as part of a transaction to the blockchain backend. Django models store encrypted vote data as strings or binary fields. When the voting phase ends, a secure admin view can aggregate encrypted votes using homomorphic addition. Only the election authority, possessing the private key, can decrypt the final result using a Python script or management command integrated with Django's admin interface.

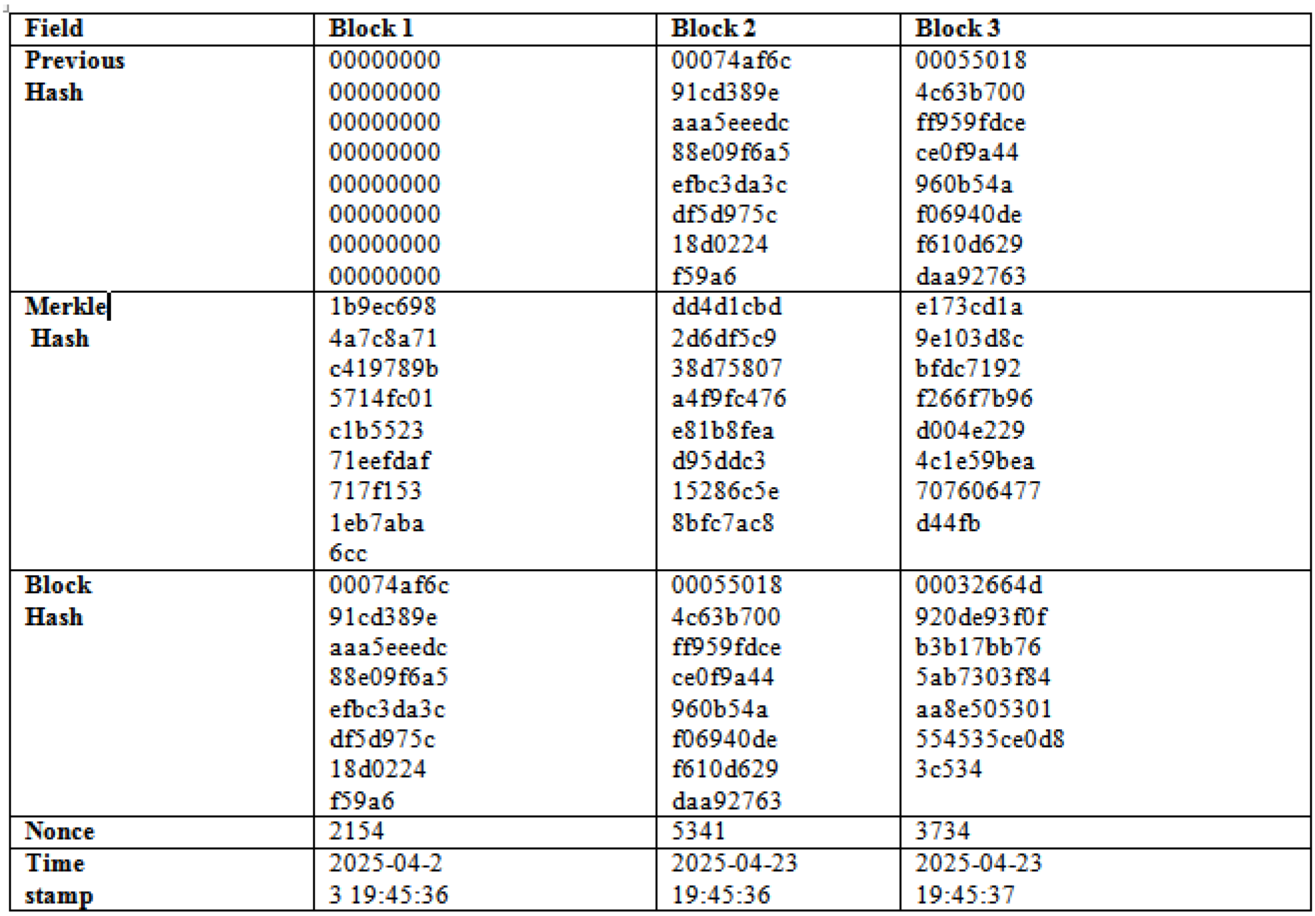
On the other hand, POA is applied at the blockchain level, and Django communicates with the blockchain through REST APIs or Web3 interfaces. In a POA based blockchain network such as an Ethereum only a predefined set of trusted nodes are allowed to validate and add blocks. These validator nodes are configured externally, but Django interacts with them by sending transactions or reading data using libraries like web3.py. For example, when a user votes, Django signs the transaction with the voter's private key and sends it to the POA blockchain. The validator nodes usually maintained by the election commission or government bodies verify the transaction and include it in a block. Django can then listen to blockchain events using Web3.

This architecture allows Django to handle the frontend, authentication via Aadhaar OTP, encryption, and database management, while the underlying blockchain ensures tamper-proof recording of votes using POA consensus. Together, Homomorphic Encryption ensures privacy, while POA ensures trust and efficiency in block validation, creating a secure and transparent remote voting system.

In a blockchain-based voting system, when a user casts a vote, it is treated as a digital transaction similar to a financial transfer in cryptocurrency. This vote is signed using the voter's private key stored in a secure digital wallet, ensuring its authenticity and protecting it from tampering. Once signed, the vote is encrypted and temporarily added to a shared public ledger distributed across all network nodes. Multiple such transactions are grouped into a single block, much like placing several votes into one secure container. A miner then verifies the block by checking digital signatures, ensuring no duplicate votes exist.If everything is valid, the block is approved using simpler methods in private blockchains and added to the chain. Each new block includes a hash of the previous one, ensuring the continuity and integrity of the entire blockchain, making it tamper-proof and transparent.

In a blockchain-based voting platform, each block functions as a secure container holding a batch of vote transactions, ensuring integrity, transparency, and resistance to tampering. The table referenced outlines the metadata associated with three such blocks, each playing a vital role in preserving the trustworthiness of the election process.

The Previous Hash field in every block contains the cryptographic hash of the block before it, linking blocks in a secure, tamper-evident sequence. In the context of voting, this ensures immutability—if any data is changed in a previous block, its hash changes, which would break the chain and immediately indicate tampering. For instance, Block 2's Previous Hash matches Block 1's Block Hash exactly, confirming Block 2 follows Block 1 directly. This structure ensures that all votes are stored in the precise order they were submitted and validated, maintaining the election timeline and integrity.



Field	Block 1	Block 2	Block 3
Previous Hash	00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000	00074af6c 91cd389e aaa5eeedc 88e09f6a5 efbc3da3c df5d975c 18d0224 f59a6	00055018 4c63b700 ff959fdce ce0f9a44 960b54a f06940de f610d629 daa92763
Merkle Hash	1b9ec698 4a7c8a71 c419789b 5714fc01 c1b5523 71eefdaf 717f153 1eb7aba 6cc	dd4d1cbd 2d6df5c9 38d75807 a4f9fc476 e81b8fea d95ddc3 15286c5e 8bfc7ac8	e173cd1a 9e103d8c bfdc7192 f266f7b96 d004e229 4c1e59bea 707606477 d44fb
Block Hash	00074af6c 91cd389e aaa5eeedc 88e09f6a5 efbc3da3c df5d975c 18d0224 f59a6	00055018 4c63b700 ff959fdce ce0f9a44 960b54a f06940de f610d629 daa92763	00032664d 920de93f0f b3b17bb76 5ab7303f84 aa8e505301 554535ce0d8 3c534
Nonce	2154	5341	3734
Time stamp	2025-04-2 3 19:45:36	2025-04-23 19:45:36	2025-04-23 19:45:37

Table - 3 Blockchain mining for transaction (vote)

Above table showing how blockchain block is forming in the making chain of the hash.

The Merkle Hash (or Merkle Root) represents a single hash generated from all individual transactions (votes) within the block. This is achieved by hashing each vote and then repeatedly combining those hashes until only one root hash remains. In voting systems, this process enables efficient verification and privacy proving a specific vote exists within a block without exposing all others.

The Block Hash serves as the unique digital fingerprint of the entire block. It is computed from several elements, including the Previous Hash, Merkle Hash, Timestamp, and Nonce. In a voting system, this hash acts as a validation tool, ensuring the block has not been altered. A small change in the block even in one vote would result in a completely different Block Hash. This new hash would not match the next block's Previous Hash, making tampering immediately evident.

The Nonce is a number used during the block's creation to meet the requirements of the consensus mechanism. While voting systems often use lightweight mechanisms like Proof of Authority where pre-approved validators add blocks nonce can still play a role. For example, a nonce of 5341 in Block 2 means 5341 attempts were made to find a valid hash. This process adds a minimal but meaningful layer of security, even in validator-based systems.

The Timestamp marks the exact date and time a block is created and added to the chain. In a voting environment, this adds a crucial layer of accountability by recording when each batch of votes was received and validated. Timestamps assist in resolving disputes and are critical for post-election audits. It's also possible for blocks—such as Block 1 and Block 2—to share the same timestamp during high-throughput events or in small-scale elections with fast transaction processing.

Each block follows a structured process in this voting system. Block 1, known as the genesis block, initiates the chain. It contains the first group of vote transactions or setup data, and its Previous Hash is all zeros, indicating there's no preceding block. Block 2 builds on Block 1, containing new encrypted vote data (summarized by its Merkle Hash) and validated through its nonce and timestamp. Likewise, Block 3 continues the sequence by linking to Block 2, further ensuring the integrity of votes cast during that period. This chain of blocks guarantees that every entry is securely connected to the one before it, preserving a continuous and verifiable voting history.

In work, the combination of the hash chain, Merkle tree, block hash and nonce, and timestamps creates a transparent, tamper-evident digital ledger for voting. The hash chain securely links all blocks, the Merkle tree allows vote verification without exposing personal data, the block hash and nonce confirm each block's integrity, and timestamps provide an auditable record of voting events. Together, these features build a strong foundation for trustworthy, secure, and transparent digital elections.

V.CONCLUSION AND FUTURE WORK

This project successfully shows that a blockchain-based online voting system is feasible. integrated with Aadhaar authentication, providing secure voter verification, transparent vote recording, and tamper-proof storage. It addresses major concerns in traditional voting systems, including voter fraud, lack of transparency, and limited accessibility. The implementation ensures legitimate voter participation through Aadhaar and OTP-based authentication while enabling remote access for overseas and disabled citizens. These features collectively enhance the credibility, inclusiveness, and security of the electoral process.

To further strengthen the system and prepare it for real-world deployment, future work should focus on integrating AI-based mechanisms to detect fraud or anomalous voting behaviours in real time. Additionally, scalability testing under large-scale election conditions is crucial to ensure optimal performance and responsiveness. Finally, integration with public blockchain platforms like Ethereum or Hyperledger could enhance decentralization and unlock advanced features, contributing to a more robust, flexible, and globally deployable online voting solution.

VI. REFERENCE

[1]. Prabhu, Ganesh, Nizarahammed, A., Prabu, S., Raghul, S., Thirrunavukkarasu, R. R., and Jayarajan, P. (2021). Smart Online Voting System. Proc. 7th International Conference on Advanced Computing and Communication Systems (ICACCS), (doi:10.1109/icaccs51430.2021.9441818).

[2]. Taş, Rauf, and Tanrıöver, Ömer Özgür. (2024). A Systematic Review of Challenges and Opportunities of Blockchain for E-Voting. Journal of Computer Engineering, Ankara University, Turkey. [DOI or Link Not Provided].

[3]. Abuidris, Yassir, Kumar, Rakesh, and Wenyong, Wang. (2020). A Survey of Blockchain-Based E-Voting Systems. Proc. 2nd International Conference on Blockchain Technology and Applications (ICBTA '19), (pp. 99–104). ACM. (https://doi.org/10.1145/3376044.3376060).

[4]. Hajian Berenjestanaki, Mohammad, Barzegar, Hamid Reza, El Ioini, Nicola, and Pahl, Claus. (2024). Blockchain-Based E-Voting Systems: A Technology Review. Electronics, 13(1), 17. (https://doi.org/10.3390/electronics13010017).

[5]. Al-Maaitah, Sami, Qatawneh, Mohammad, and Quzmar, Ahmad. (2021). E-Voting System Based on Blockchain Technology: A Survey. Proc. International Conference on Information Technology (ICIT), Amman, Jordan, (pp. 200–205). (https://doi.org/10.1109/ICIT52682.2021.9491734).

[6]. Vivek, S. K., Yashank, R. S., Prashanth, Y., Yashas, N., and Namratha, M. (2020). E-Voting Systems using Blockchain: An Exploratory Literature Survey. Proc. Second International Conference on Inventive Research in Computing Applications (ICIRCA), Coimbatore, India, (pp. 890–895). (doi: 10.1109/ICIRCA48905.2020.9183185).

[7]. Pandey, Anurag, Bhasi, Manoj, and Chandrasekaran, K. (2019). VoteChain: A Blockchain Based E-Voting System. Proc. Global Conference for Advancement in Technology (GCAT), Bangalore, India, (pp. 1–4). (doi: 10.1109/GCAT47503.2019.8978295).

[8]. Banawane, Akshay, Bhansali, Yash, Dabadgaonkar, Meenal, Javalekar, Omkar, Patil, Gaurav, and Kumavat, M. K. (2022). A Novel Approach for e-Voting System Using Blockchain. Proc. 3rd International Conference on Intelligent Engineering and Management (ICIEM), London, United Kingdom, (pp. 263–268). (doi: 10.1109/ICIEM54221.2022.9853099).

[9]. Chovancová, Eva, Chovanec, Milan, Ádám, Norbert, and Hurtuk, Jakub. (2023). Online Voting Management System Based on Blockchain. Proc. 27th International Conference on Intelligent Engineering Systems (INES), Nairobi, Kenya, (pp. 000169–000174). (doi: 10.1109/INES59282.2023.10297916).

[10]. Hamka, F. M., Wardana, J. A., Jaelani, R. S., and Widianto, M. H. (2023). E-Voting Using Blockchain: A Systematic Literature Review. Proc. 3rd International Conference on Electronic and Electrical Engineering and Intelligent System (ICE3IS), Yogyakarta, Indonesia, (pp. 360–364). (doi: 10.1109/ICE3IS59323.2023.10335317).

[11]. McCorry, Patrick, Shahandashti, Siamak F., and Hao, Feng. (2017). A Smart Contract for Boardroom Voting with Maximum Voter Privacy. Proc. International Conference on Financial Cryptography and Data Security, (pp. 357–375). Springer. (https://doi.org/10.1007/978-3-319-70278-0_23).

[12]. Zamyatin, Alexei, et al. (2019). SoK: Communication Across Distributed Ledgers. Proc. ACM Workshop on Blockchains, Cryptocurrencies, and Contracts, (pp. 1–14). (https://doi.org/10.1145/3316481.3316486).

[13]. Noizat, T. (2015). Blockchain Electronic Vote. Handbook of Digital Currency: Bitcoin, Innovation, Financial Instruments, and Big Data, Elsevier, (pp. 453–461). (https://doi.org/10.1016/B978-0-12-802117-0.00020-6).