Online Voting System Using Blockchain

Major Project Report

Submitted in partial fulfillment of the requirement of University of Mumbai

For the Degree of

(Computer Engineering)

By

1) Atharva Birje.	ID No: TU3F2122158
2) Harsh Minde.	ID No: TU3F2122164
3) Ameya Mane.	ID No: TU3F2122206

Under the Guidance of

Prof. Dnyaneshwar Thombre.

Department of Computer Engineering

TERNA ENGINEERING COLLEGE

Plot no.12, Sector-22, Opp. Nerul Railway station, Phase-11, Nerul (w), Navi Mumbai 400706 UNIVERSITY OF MUMBAI

terna
Knowledge to Empower

TERNA ENGINEERING COLLEGE, NERUL,

NAVI MUMBAI

Department of Computer Engineering

Academic Year 2024-2025

CERTIFICATE

This is to certify that the major project entitled "Online Voting System Using Blockchain" is a bonafide work of

1) **Atharva Birje.**	**ID No: TU3F2122158**
2) **Harsh Minde.**	**ID No: TU3F2122164**
3) **Ameya Mane.**	**ID No: TU3F2122206**

submitted to the University of Mumbai in partial fulfillment of the requirement for the award of the Bachelor of Engineering (Computer Engineering).

Guide **Head of Department** **Principal**

Project Report Approval

This Major Project Report – entitled "**Online Voting System Using Blockchain**" by following students is approved for the degree of **B.E. in "Computer Engineering"**.

Submitted by:

1) Atharva Birje.	**ID No: TU3F2122158**
2) Harsh Minde.	**ID No: TU3F2121164**
3) Ameya Mane.	**ID No: TU3F2122206**

Examiners Name & Signature:

1. _____ DCT.

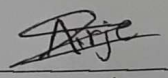
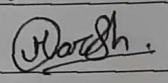
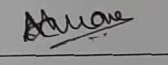
2. _____ Dr. Sujata B

Date: 23/04/2025

Place Nerul

Declaration

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Atharva Birje.	**TU3F2122158**	
Harsh Minde.	**TU3F2122164**	
Ameya Mane.	**TU3F2122164**	

Date: 23/04/2025

Place Nerul

Acknowledgement

We would like to express our sincere gratitude towards our guide **Prof. Dnyaneshwar Thombre**, Project Coordinators **Prof. Pramila Mate** for their help, guidance and encouragement, they provided during the project development. This work would have not been possible without their valuable time, patience and motivation. We thank them for making our stint thoroughly pleasant and enriching. It was great learning and an honor being their student.

We are deeply thankful to **Prof. Kishore Sakure (H.O.D Computer Department)** and entire team in the Computer Department. They supported us with scientific guidance, advice and encouragement, they were always helpful and enthusiastic and this inspired us in our work.

We take the privilege to express our sincere thanks to **Dr. L. K. Ragha** our Principal for providing the encouragement and much support throughout our work.

Atharva Birje. **ID No: TU3F2122158** _____

Harsh Minde. **ID No: TU3F2122164** _____

Ameya Mane. **ID No: TU3F2122206** _____

Date: 23/04/2025

Place: Navi

Index

TABLE OF CONTENTS

Abstract

This report presents VoteChain, a groundbreaking blockchain-integrated, cross-platform mobile voting application designed to modernize and secure electoral processes. Traditional voting systems are often plagued by inefficiencies, high operational costs, and significant security vulnerabilities that can compromise electoral integrity. VoteChain addresses these challenges through a hybrid architecture that exclusively employs blockchain technology for immutable vote recording while delegating ancillary tasks to a robust off-chain database. This dual approach not only ensures the integrity and transparency of vote tallies but also substantially reduces network congestion and operational expenses. Key features of VoteChain include multifactor authentication, cryptographic hashing, and role-based access control, all of which work together to prevent unauthorized access and false voting. The application's intuitive interface, developed using Flutter, facilitates easy navigation and real-time visualization of election results, thereby enhancing voter engagement and trust.

By leveraging the immutable nature of blockchain on the Ethereum network and the scalability of Firebase for non-critical data, VoteChain demonstrates a secure, efficient, and cost-effective solution for conducting large-scale elections. Overall, this system lays a solid foundation for digital democracy, offering a transformative approach to overcome the inherent limitations of traditional electoral methods while ensuring a transparent, tamper-proof, and accessible voting process for diverse democratic societies.

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Introduction:

In a democratic society, the integrity of electoral processes is paramount for ensuring that the collective will of citizens is accurately reflected in governmental decisions. In India—the world's largest democracy—the challenges of conducting free, fair, and transparent elections are magnified by its immense population and diverse regional dynamics. Traditional electoral systems require colossal logistical coordination, significant financial expenditure, and extensive human resources. For instance, national elections such as the Lok Sabha polls (year 2024) incur astronomical costs, with total expenditures in recent cycles estimated to approach ₹1.35 lakh crores. These expenses cover direct government spending on infrastructure, security, administrative logistics, and the procurement and maintenance of millions of Electronic Voting Machines (EVMs), in addition to setting up nearly one million polling stations and mobilizing over 10 lakh security personnel. Despite these investments, inefficiencies remain, including disenfranchisement of migrant workers and citizens unable to vote in their registered constituencies.

Amid these systemic challenges, blockchain technology has emerged as a promising solution due to its decentralized nature, immutability, and robust cryptographic security. Its inherent tamper-proof characteristics ensure that once a vote is cast, it remains unaltered, thereby providing a level of transparency and verifiability that traditional systems struggle to achieve. However, many existing blockchain-based voting systems have been criticized for their high operational costs and inefficiencies, often stemming from redundant on-chain operations that lead to prohibitive gas fees and latency issues. Industry analyses indicate that approximately 60–70% of current blockchain e-voting prototypes face such challenges, underscoring the need for a more balanced and cost-effective approach.

This retort presents VoteChain, a novel, cross-platform mobile application that leverages blockchain technology exclusively for vote recording and tallying while delegating non-critical operations to off-chain solutions. VoteChain's architecture is designed to address key limitations by integrating a minimalist on-chain component—implemented through smart contracts on the Ethereum-Sepolia testnet (accessed via QuickNode endpoints)—with a robust off-chain backend powered by Firebase (as a secondary database). This hybrid architecture minimizes the blockchain load, thereby reducing gas fees by an estimated 40–60%, while still ensuring the integrity and transparency of the vote count.

VoteChain is developed using Flutter, which provides an intuitive user interface and seamless interaction for voters, candidates, party heads and election officials. The system is engineered to support multiple election types—ranging from local (e.g., Gram Panchayat) to state (e.g., Vidhan Sabha) and national (e.g., Lok Sabha) elections—by dynamically managing election hierarchies and participant roles through Firebase. Multifactor authentication is employed to ensure that only eligible voters can participate, and additional off-chain measures are in place to prevent duplicate voting and unauthorized access.

By eliminating the need for extensive physical infrastructure and traditional EVMs, VoteChain aims to drastically reduce the operational costs and manpower required for elections. Moreover, the application incorporates advanced features such as detailed real-time result visualization—through both list-based and graphical formats—and comprehensive data logging for post-election analysis. Notably, once an election is concluded, non-critical vote data is archived off-chain after preserving essential logs and metadata, and a new smart contract instance can be deployed for subsequent elections. This strategy minimizes on-chain storage requirements and enhances long-term data security without compromising auditability or transparency. Additionally, if the underlying root storage is reset or cleared, the blockchain network's capacity can be efficiently repurposed for future elections, ensuring optimal utilization of resources.

This report details the design, development, and evaluation of VoteChain. It discusses how the system integrates blockchain technology with a secondary database to achieve scalability, cost efficiency, and high security, and it outlines the potential of such an approach to transform electoral processes in India and other democracies. Through a comprehensive analysis of VoteChain's architecture, functionality, and performance, this work positions the system as a scalable and practical solution for modernizing elections, ensuring that democratic ideals intersect effectively with technological innovation.

1.2 Organization of the Report:

This report is organized into six comprehensive chapters, each addressing a critical aspect of the project:

- **Chapter 1: Introduction**
  This chapter presents the background and motivation for developing a secure, transparent, and efficient e-voting system. It introduces blockchain technology as a transformative solution for addressing the limitations of traditional voting methods.
- **Chapter 2: Literature Survey**
  This chapter explores the challenges associated with conventional voting systems, such as fraud, tampering, and inefficiencies. It defines the problem statement, outlines the objectives, discusses existing systems, and highlights the need and scope of the proposed solution.
- **Chapter 3: Software Analysis and Design**
  This chapter presents the detailed system design, including the software model, technical architecture, system requirements, and feasibility analysis. It also includes visual design elements such as data flow diagrams, use case diagrams, sequence diagrams, and the flowchart of the proposed e-voting process. A comprehensive Risk Mitigation, Monitoring, and Management Plan is also provided.
- **Chapter 4: Implementation**
  This section describes the implementation of the proposed system, including frontend and backend development, integration of blockchain technology for vote immutability, and Firebase for managing off-chain metadata. It also addresses privacy, security features, performance optimization, and deployment strategies.
- **Chapter 5: Results and Discussion**
  This chapter evaluates the system’s output, discusses the expected outcomes, and reflects on the challenges encountered during development. It also presents solutions to overcome these challenges, along with an analysis of the system's efficiency, scalability, and security.
- **Chapter 6: Conclusion**
  The final chapter summarizes the key findings and impact of the project. It outlines the system’s contributions to improving the electoral process and discusses potential areas for future development, such as incorporating artificial intelligence for fraud detection and enhancing scalability for large-scale elections.

Chapter 2

Literature Survey

2.1 Problem statement:

Elections are the cornerstone of democracy, enabling citizens to choose their representatives through a secure and transparent process. However, conducting large-scale elections in a vast and diverse country like India presents significant challenges. Traditional voting methods, including paper ballots and Electronic Voting Machines (EVMs), demand extensive infrastructure, manpower, and financial resources. For instance, the 2024 Lok Sabha elections incurred an estimated expenditure of ₹1.35 lakh crores, covering security, administrative logistics, and EVM procurement. Despite these investments, inefficiencies persist—manifesting as logistical delays, voter disenfranchisement, and electoral fraud.

Security vulnerabilities remain a critical concern in conventional voting systems. Paper ballots are susceptible to tampering, miscounting, and loss, while EVMs are vulnerable to hacking and unauthorized access. Moreover, centralized digital voting systems introduce significant risks of cyberattacks, data breaches, and vote manipulation. Additionally, geographical constraints hinder millions of Indian citizens, including migrant workers, from casting their votes, further reducing electoral accessibility.

While online voting has been explored as an alternative, existing digital systems face major security and privacy risks—such as identity fraud, vote manipulation, and lack of transparency—that limit their widespread adoption. Although blockchain technology offers promising benefits like decentralization, cryptographic security, and immutability, many blockchain-based voting models suffer from operational inefficiencies, including high gas fees and latency, which hinder scalability and cost-effectiveness.

Given these challenges, there is a pressing need for a secure, transparent, scalable, and cost-effective voting system that overcomes the limitations of both traditional and current digital voting methods. Developing an innovative approach that addresses these issues while maintaining democratic integrity is crucial for modernizing election processes and enhancing trust in the electoral system.

2.2 Existing System Survey:

Blockchain technology offers a secure and transparent platform for digital voting by leveraging advanced cryptography and decentralized authentication. Despite its promise, challenges like scalability and user accessibility remain. This section reviews key research contributions and explores potential improvements in blockchain-enabled voting systems.

The Blockchain Enabled Online-Voting System developed by Akhil Shah, Nishita Sodhia, Shruti Saha, Soumi Banerjee, and Madhuri Chavan (2020) utilizes blockchain to create an immutable and transparent voting system. The project incorporates 128-bit AES encryption and SHA-256 to enhance security, ensuring that the votes cast are secure and tamper-proof. The system employs authentication methods such as unique identification keys and biometric fingerprint verification, ensuring that only authorized voters can participate. Votes are cast and recorded as blockchain transactions, preserving their integrity and transparency throughout the election process [1].

In paper A Privacy-Preserving Voting Protocol on Blockchain, Wenbin Zhang et al. (2018) introduced a decentralized voting protocol leveraging homomorphic encryption and distributed tallying, effectively removing the need for trusted third parties. The protocol ensures voter privacy by encrypting votes and distributing ballots across peers while also detecting and correcting dishonest votes without compromising anonymity. The system uses Hyperledger Fabric, making it particularly suitable for small to medium-scale elections where privacy is a critical concern [2].

The paper by Stephan Neumann, Oksana Kulyk, and Melanie Volkamer (2014) describes a Usable Android Application Implementing Distributed Cryptography for Election Authorities. This Android app is designed to facilitate secure distributed key generation and verifiable vote decryption for non-technical election authorities. While it simplifies the voting process for non experts, the authors highlight that users struggled with understanding complex security concepts, suggesting the need for improved educational tools to assist users in navigating cryptographic security [3].

Jae-Geun Song, Sung-Jun Moon, and Ju-Wook Jang (2021) developed A Scalable Implementation of Anonymous Voting over Ethereum Blockchain to address scalability issues in blockchain voting systems. Their implementation successfully scales to accommodate a larger number of voters and candidates compared to previous models, reducing time complexity and making blockchain-based voting systems more efficient and suitable for large-scale elections [4].

The study by Yulia Bardinova et al. (2018) focused on the impact of blockchain algorithms on mobile devices with their paper Measurements of Mobile Blockchain Execution Impact on Smartphone Battery. The research found that Proof of Work (PoW) algorithms significantly increase battery discharge rates and device temperature, while Proof of Authority (PoA) algorithms have minimal impact on battery performance. Additionally, cellular connections were found to worsen battery discharge rates compared to Wi-Fi, providing essential insights into optimizing blockchain applications for mobile platforms [5].

In Decentralized Voting Platform Based on Ethereum Blockchain, David Khoury et al. (2020) developed a decentralized voting platform where smart contracts enforce transparency and voting rules, allowing one vote per registered mobile number. The system also achieves voter authentication without relying on a third-party server, enhancing both privacy and security. This approach ensures transparency while maintaining the integrity of the election process by preventing unauthorized access [6].

In the paper Secure Electronic Voting System using Blockchain Technology by D. Dwijesh Kumar, D. V. Chandini, and Dinesh Reddy (2020), the authors propose a system that enhances privacy by storing voter information and votes on two separate blockchains. This ensures the security of sensitive voter data while maintaining transparency. The system uses blockchain transactions for casting votes, with two-step verification via a PIN. Additionally, users can verify that their vote has been correctly recorded. The use of SHA-256 encryption ensures the immutability and security of the voting process [7].

In Implementation of Decentralized Blockchain E-voting, Saad Moin Khan et al. (2018) propose a decentralized e-voting system that leverages blockchain to create a tamper-proof and transparent voting process. The study demonstrates how smart contracts can automate the voting process to secure vote casting, while real-time vote verification builds voter trust. The integration of a user interface via Metamask further simplifies voter interaction, making the system both accessible and secure. However, the authors acknowledge several limitations, including scalability challenges, dependency on continuous internet connectivity, a complex setup process, and the inherent requirement for Ether. They suggest that these issues can be mitigated through system optimizations such as adding offline capabilities, streamlining the setup, and exploring alternatives to traditional cryptocurrency models [8].

In E-Voting System in Smart Phone Using Mobile Application, Kalaiyarasi et al. (2020) present an Android-based e-voting solution that employs AES256 encryption for the secure storage of votes and utilizes Firebase for OTP generation to authenticate voters. This approach facilitates remote voting, thereby reducing the risk of fraud and minimizing human errors associated with manual vote counting. The system's ability to publish results immediately after the election further underscores its operational

efficiency. Nevertheless, the study highlights certain limitations, such as the lack of offline voting support—which may restrict access in remote areas— vulnerability due to reliance on OTP-based authentication, and potential security concerns arising from the use of third-party services like Firebase [9].

In Survey on Blockchain Based Data Storage Security for Android Mobile Applications, Hussam Saeed Musa et al. (2019) investigate how blockchain technology can enhance the security and reliability of data storage within mobile applications. Their proposed BSADS framework, which comprises six comprehensive layers, emphasizes blockchain's advantages over traditional encryption methods by ensuring robust data integrity and auditability. The study also explores innovative solutions to address challenges related to scalability, performance, and cost, including blockchain pruning and the adoption of energy-efficient consensus algorithms, as well as the use of lightweight nodes tailored for mobile environments. Despite these promising approaches, the authors point out significant limitations such as prior unsuccessful implementations in mobile voting projects, scalability issues with platforms like Ethereum Name Service (ENS), privacy concerns due to blockchain transparency, high data storage costs, and the resource-intensive nature of smart contracts that may hinder overall performance [10].

It shows that blockchain-enabled voting systems offer promising security and transparency enhancements for digital elections. However, challenges—ranging from scalability and energy efficiency to user accessibility and connectivity—remain. These gaps underscore the need for further research to refine and optimize blockchain-based solutions for practical, large-scale electoral applications.

2.3 Objectives:

The project aims to develop a Blockchain-Based Online Voting System to enhance security, transparency, scalability, and accessibility in elections. The key objectives are:

1. Secure Voter Authentication & Privacy:

- Implement Multi-Factor Authentication (MFA) with biometrics and OTP to verify voter identity securely.
- Utilize cryptographic hashing to ensure voter anonymity while preventing impersonation and unauthorized voting.

2. Immutable Vote Storage & Integrity:

- Store votes immutably on the Ethereum blockchain, ensuring tamper-proof, decentralized security.
- Leverage smart contracts and cryptographic proofs to prevent vote alteration or manipulation.

3. Automated Vote Validation & Tallying:

- Deploy Ethereum smart contracts to enforce region-based vote validation, preventing cross-constituency fraud.
- Automate vote counting, tallying, and result declaration through decentralized execution.

4. Real-Time Election Results & Analytics:

- Provide instant election results through a mobile app with graphical and tabular representations.
- Offer metadata analytics, including voter demographics, turnout trends, and temporal voting patterns.

5. Scalability & Cost Optimization:

- Reduce blockchain congestion and gas fees by offloading non-sensitive metadata (profiles, logs) to Firebase.
- Optimize transaction costs by 40–60% while ensuring seamless performance at the scale of India's electorate.

6. Multi-Tier Election Support:

- Enable elections across Gram Panchayat, Vidhan Sabha, and Lok Sabha, with hierarchical UI navigation for voters, party heads, candidates, and administrators.
- Support region-wise classification and localized election processes.

7. Transparency & Audit Compliance:

- Maintain immutable on-chain voting records to enhance public trust and verifiability.
- Store off-chain audit logs in Firebase for compliance with election regulations and privacy laws.

8. Fraud Prevention & Security Mechanisms:

- Implement Sybil attack resistance, cryptographic verifications, and AI-driven anomaly detection to prevent fraudulent voting activities.
- Ensure blockchain-based identity verification and activity logging for enhanced security.

9. Role-Based Access Control:

- Define distinct permissions for voters, party heads, candidates, election officials, and administrators to ensure secure and controlled access to the system.
- Prevent unauthorized actions through smart contract-enforced roles and permissions.

10. Edge-Case Handling & Data Management:

- Address critical election scenarios, including migrant voter accessibility, candidates contesting multiple constituencies, and post-election data archiving.
- Optimize blockchain storage through efficient data structuring and on-chain/off-chain data balance.

11. Future Scalability & Technological Enhancements:

- Design a modular architecture to integrate Layer-2 scaling solutions (e.g., Polygon, Optimistic Rollups) for enhanced performance.
- Leverage AI-driven analytics for anomaly detection, voter trend analysis, and security improvements.

12. Accessibility & Inclusivity:

- Ensure mobile and web platform compatibility to accommodate diverse user demographics.
- Implement features for visually impaired voters, including voice-assisted navigation and screen-reader support.

2.4 Need of the Problem:

The integrity and efficiency of electoral processes are fundamental to democratic governance. However, traditional voting systems—whether paper-based or electronic—face significant challenges related to security, transparency, scalability, and accessibility. With growing concerns over electoral fraud, logistical inefficiencies, and the high costs associated with conventional elections, a Blockchain-Based Online Voting System (VoteChain) emerges as a transformative necessity rather than a mere improvement. The key reasons highlighting the urgency and importance of this system are as follows:

1. Electoral Frauds and Security Concerns:

Traditional voting systems are vulnerable to vote tampering, ballot stuffing, duplicate voting, and cyberattacks. Paper ballots can be manipulated or lost, while electronic voting machines (EVMs) are prone to hardware tampering, software hacking, and insider manipulation. Blockchain's decentralized, immutable ledger ensures that votes cannot be altered or deleted, eliminating the risk of fraud and enhancing election security.

2. Transparency & Trust in the Electoral Process:

A major challenge in elections is the lack of transparency in vote counting and result declaration. Opaque processes often lead to public distrust and disputes over election outcomes. Blockchain provides real-time, publicly auditable, and tamper-proof vote records, ensuring a verifiable and transparent electoral process while maintaining voter anonymity.

3. Voter Authentication & Fraud Prevention:

Unauthorized voting due to identity fraud, multiple registrations, and fake voter IDs undermines election integrity. Current authentication methods rely on voter ID cards, which can be forged or misused. VoteChain integrates Multi-Factor Authentication (MFA) with biometric verification, cryptographic identity management, and OTP-based authentication to ensure that only eligible voters cast their votes, eliminating duplicate or fraudulent voting.

4. Accessibility & Remote Voting for All:

Millions of voters, including non-resident Indians (NRIs), migrant workers, and individuals with disabilities, face challenges in reaching polling stations. Postal and proxy voting mechanisms are inefficient and delay results. A blockchain-based online voting system enables secure remote voting**,** ensuring greater voter participation, especially for marginalized groups.

5. High Costs & Resource-Intensive Elections:

Traditional elections demand massive financial investments for infrastructure, security, and logistics. For instance, India's 2024 Lok Sabha elections cost an estimated ₹1.35 lakh crores, covering EVM procurement, polling stations, and personnel deployment. By minimizing physical infrastructure and leveraging blockchain for vote transactions and Firebase for metadata storage, VoteChain significantly reduces operational costs while ensuring efficiency.

6. Instant & Accurate Vote Counting:

Manual vote counting is time-consuming, prone to human errors, and susceptible to manipulation. Even electronic voting requires centralized validation, leading to delays. Smart contracts on Ethereum enable real-time, automated vote tallying, ensuring accurate, tamper-proof results without the risk of miscalculation or data corruption**.**

7. Scalability & Performance Optimization:

India, with 900 million plus eligible voters, faces significant challenges in managing large-scale elections. Existing blockchain voting systems struggle with high gas fees, network congestion, and latency. VoteChain integrates Layer-2 scaling solutions (e.g., Polygon, Optimistic Rollups) and off-chain storage (Firebase) to handle high voter loads efficiently without compromising security or performance**.**

8. Regulatory Compliance & Legal Standards:

Electoral laws mandate secure, verifiable, and fair voting systems. Traditional mechanisms often struggle to meet compliance requirements due to disputes over results and allegations of rigging**.** Blockchain's cryptographic security, role-based access controls, and immutable ledger align with data privacy laws (e.g., GDPR, India's IT Act), ensuring transparency, auditability, and legal compliance.

9. Mitigating Voter Suppression & Coercion:

Political intimidation and voter suppression tactics prevent free and fair elections in various regions. By enabling secure and anonymous remote voting, VoteChain ensures that voters can cast their ballots without fear of external influence or coercion**.**

10. Future-Proofing Elections with Emerging Technologies:

As digital transformation accelerates, election systems must evolve to integrate AI-driven fraud detection, anomaly detection, and predictive analytics to detect irregular voting patterns. VoteChain is designed to be adaptive and scalable**,** incorporating advanced cryptographic security, real-time monitoring, and AI-powered analytics**,** making elections more robust and resilient against emerging threats.

The existing electoral system is plagued by inefficiencies, security risks, and a lack of transparency. A Blockchain-Based Online Voting System (VoteChain) addresses these critical issues by providing a secure, transparent, scalable, and cost-effective solution. By leveraging blockchain's decentralization, cryptographic security, and real-time automation, the system strengthens democracy and ensures trustworthy elections, particularly for large-scale democratic processes like those in India**.**

2.5 Scope of the project:

The Blockchain-Based Online Voting System (VoteChain) is designed to modernize electoral processes in India & beyond by addressing key challenges related to security, transparency, scalability, accessibility, and efficiency. By integrating blockchain technology, smart contracts, and a mobile application, the system ensures a tamper-proof, decentralized, and publicly auditable voting process.

1. Security & Immutability:

- Votes are securely recorded on the Ethereum blockchain (Sepolia testnet) using Solidity-based smart contracts, ensuring tamper-proof, fraud-resistant, and publicly verifiable vote storage.
- Cryptographic hashing techniques preserve voter anonymity and ballot secrecy, preventing identity tracking while ensuring auditability.
- Zero-Knowledge Proofs (ZKPs) enhance privacy, allowing voter authentication without exposing personal information.
- Fraud prevention mechanisms, including public ledger verification and region-based validation, ensure legitimate vote casting and prevent cross-constituency fraud.

2. Smart Contracts & Automation:

- Automated vote validation ensures that only eligible voters can cast votes, eliminating duplicate and fraudulent voting.
- Real-time vote tallying and instant result declaration without manual intervention.
- Region-based validation ensures voters can only participate in elections within their designated constituencies.
- Immutable and self-executing election rules, reducing the risk of manipulation and ensuring unbiased electoral outcomes.

3. Voter Authentication & Privacy:

- Multi-Factor Authentication (MFA), including:
  - Biometric verification (fingerprint/facial recognition) and OTP-based authentication.
  - Aadhaar/PAN-based voter registration to verify eligibility while maintaining privacy.
- Zero-Knowledge Proofs (ZKPs) enable anonymous authentication while ensuring secure voter verification.
- Advanced cryptographic techniques prevent unauthorized tracking and ensure voter anonymity.

4. Mobile Application:

A Flutter-based cross-platform mobile app that offers:

- Secure voter registration, authentication, and voting.
- Intuitive voting interface with accessibility features such as voice-guided navigation for visually impaired voters.
- Real-time election result visualization using dynamic graphical (pie/bar charts) and tabular formats.

5. Hybrid Data Management:

- Blockchain: Stores immutable vote records and audit trails for maximum security and transparency.
- Secondary database (Firebase): Stores non-sensitive metadata (voter profiles, logs, candidate, party head details), optimizing blockchain storage and reducing gas fees by 40–60%.

6. Real-Time Results & Transparency:

- Instant election results are displayed in interactive visual formats (graphs, charts, and tables).
- Publicly auditable blockchain voting records ensure transparency while complying with:
  - General Data Protection Regulation (GDPR).
  - India's IT Act.
  - Election Commission guidelines.

7. Scalability, Performance & Fault Tolerance:

- Optimized architecture to handle up to 900 million voters during national elections.
- Layer-2 scalability solutions (e.g., Polygon integration) reduce network congestion and transaction fees.
- Fault-tolerant design ensures continuous operation even under peak loads, preventing data loss.

8. Comprehensive Election Coverage:

- Supports elections at multiple levels:
  - Gram Panchayat (Village-Level).
  - Vidhan Sabha (State-Level).
  - Lok Sabha (National-Level).
- Region-based voter eligibility checks prevent unauthorized participation and uphold electoral integrity.

9. Fraud Prevention & Auditability:

- Public ledger verification mechanisms prevent electoral fraud.
- Secure audit logs allow authorities and independent observers to monitor election integrity.
- Automated anomaly detection flags suspicious voting behaviours in real-time.

10. Edge-Case Handling:

- Migrant voter accessibility via secure remote voting mechanisms.
- Dual-constituency candidate management, ensuring accurate vote tracking for candidates contesting multiple seats.
- Post-election data archiving optimizes long-term blockchain storage and historical record-keeping.

11. Legal Compliance & Accessibility:

- Fully adheres to Election Commission guidelines, ensuring a legally compliant electoral process.
- Accessibility features support disabled voters, including speech-enabled voting for visually impaired individuals.
- Designed to withstand peak loads and system failures, ensuring continuous and fair election operations.

12. Future-Readiness & Enhancements:

- AI-driven anomaly detection identifies and mitigates election fraud attempts.
- Integration of advanced ZKPs enhances voter privacy and authentication security.
- Layer-2 scaling solutions further improve efficiency and cost-effectiveness.
- Advanced analytics provide insights into voter behaviour, helping optimize future elections.

The Blockchain-Based Online Voting System (VoteChain) establishes a secure, transparent, scalable, and cost-effective electoral framework. By ensuring voter privacy, efficiency, accessibility, and fraud prevention, this system represents the next-generation solution for modern, tamper-proof elections while supporting future technological advancements.

Chapter 3

Software Requirements Specification

3.1 Software Model

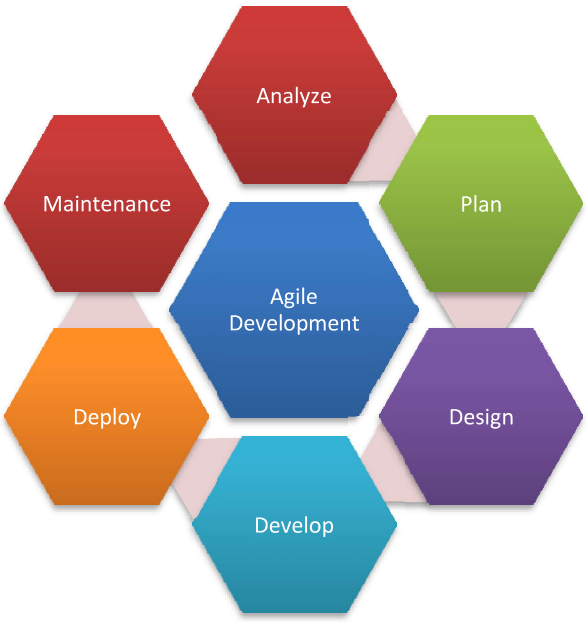


Figure 3.1 Agile Model

Agile Software Development Life Cycle (SDLC) Model:
For the development of VoteChain, an Agile Software Development Life Cycle (SDLC) model has been adopted. Agile is a widely accepted software development approach that emphasizes iterative progress, continuous feedback, and collaboration among stakeholders to ensure efficient, high-quality, and adaptable software development.

Advantages of Agile SDLC Model:
The Agile model offers several advantages, making it an ideal choice for developing a secure, scalable, and dynamic online voting system like VoteChain:

- **Risk Reduction:**
  Agile follows an iterative approach, delivering functional software in smaller increments, which helps identify and mitigate risks early in the development process.
- **Enhanced Customer Satisfaction:**
  Continuous collaboration with stakeholders, election authorities, and end-users ensures that the solution aligns with real-world requirements, leading to higher user satisfaction.

- **Improved Product Quality:**
  Agile emphasizes continuous testing and iterative improvements, ensuring that the final product meets high security, performance, and usability standards.
- **Flexibility and Adaptability:**
  Requirements can evolve dynamically, allowing developers to make adjustments based on user feedback and changing election system regulations.
- **Faster Time-to-Market:**
  Agile enables incremental releases, ensuring that essential functionalities reach users early and frequently, rather than waiting for full-scale development completion.

3.1.1 Phases of Software Model:

Agile development follows a structured yet flexible workflow divided into seven key phases, each contributing to the efficient execution of the project.

1. Planning:

- The project team defines the vision, scope, and objectives of the VoteChain system.
- A product backlog (list of prioritized features and requirements) is created to outline core functionalities, such as user authentication, blockchain-based vote casting, and real-time analytics.
- Agile teams are formed, and a project roadmap with milestones and iterations (sprints) is established.

2. Requirements Analysis:

- A discovery phase is conducted, where stakeholders (election authorities, cybersecurity experts, developers) collaborate to gather and refine requirements.
- The initial product backlog is expanded and categorized, prioritizing features like multi-factor authentication (MFA), voter verification mechanisms, and smart contract implementation.
- Requirements are reviewed and adapted based on real-world election constraints and security considerations.

3. Design:

- The system's architecture, UI/UX, and database structure are conceptualized.
- Prototypes and wireframes are designed for key modules, such as voter registration, election dashboard, and result visualization.
- Security frameworks, such as data encryption and cryptographic hashing techniques, are integrated into the design.

4. Implementation (Development):

- Developers begin coding based on selected technology stack (Flutter for frontend, Firebase for data management, QuickNode for blockchain integration).
- The implementation follows an iterative approach, where each sprint focuses on developing and refining a set of features, such as smart contract-based voting, authentication, and real-time election statistics.
- Agile methodologies, such as Scrum, are used for task management, sprint planning, and tracking progress.

5. Testing:

- Continuous testing ensures that the application remains secure, efficient, and bug-free throughout development.
- **Types of testing performed:**
  - **Unit Testing:** Verifies individual components, such as vote submission and authentication functions.
  - **Integration Testing:** Ensures smooth interaction between blockchain smart contracts, Firebase database, and user interface.
  - **User Acceptance Testing (UAT):** Involves stakeholders and test users to validate system usability and performance.

6. Deployment:

- The application is deployed incrementally, allowing early users (such as test election groups) to provide feedback.
- Smart contracts are deployed on the Sepolia testnet before moving to a production-ready Ethereum or private blockchain.
- Automated deployment pipelines ensure efficient-secure feature rollouts.

7. Maintenance and Support:

- After deployment, the development team provides continuous support, updates, and bug fixes.
- Security audits and real-time monitoring help maintain the integrity and efficiency of the system.
- Feature enhancements are implemented based on feedback from users and regulatory authorities.

3.2 Proposed System:

VoteChain's conceptual design is adapted from the Blockchain-based Secure Android Data Storage (BSADS) framework—a six-layer model originally developed for secure Android applications with blockchain-backed data storage [10]. Although BSADS is not the final production architecture, it provides a robust blueprint for a hybrid on-chain/off-chain voting system that leverages blockchain for critical vote recording and tallying while managing non-sensitive operations off-chain. This approach minimizes transaction fees, reduces network congestion, and enhances scalability (see Fig. 3.2).

3.2.1 Technical Design:

1. **User Interface (UI) Layer:**
   Provides the primary interaction point for all stakeholders (voters, candidates, party heads, election officers, etc.) via an intuitive, multi-screen Flutter application. It supports user registration, multi-factor authentication (MFA), vote casting, and real-time result visualization.
2. **Application Logic Layer:**
   Manages core business workflows by validating user inputs, processing transactions, and ensuring secure communication between the UI, blockchain, and off-chain databases. It protects data in transit through encrypted channels.
3. **Identity Management Layer:**
   Implements decentralized identity (DID) creation, cryptographic key management, and MFA. This layer verifies voter identities, enforces role-based access controls, and prevents false voting while preserving anonymity.
4. **Blockchain Interface Layer:**
   Acts as a gateway to Ethereum (using the Sepolia testnet) for vote submissions and data queries. Libraries such as Web3dart.dart abstract the complexities of interacting with the decentralized network.
5. **Blockchain Network Layer:**
   Utilizes Ethereum's Proof-of-Stake consensus on the Sepolia testnet to immutably record and tally votes. Connectivity services like QuickNode ensure low-latency and reliable transaction processing.
6. **Data Storage Layer:**
   Separates critical and non-critical data storage by recording votes and final tallies on-chain, while maintaining metadata (e.g., user profiles, logs, candidate lists) off-chain in databases like Firebase(Secondary Database). This separation reduces on-chain storage costs and optimizes overall performance.

Scalability and Security:

VoteChain achieves robust security through immutable vote recording and efficient off-chain management. By processing ancillary functions (such as user authentication, candidate/party management, election scheduling, and dynamic result visualization) off-chain, the system minimizes network load and gas fees. Enhanced protection is provided through MFA, end to-end encryption, and rigorous access controls.

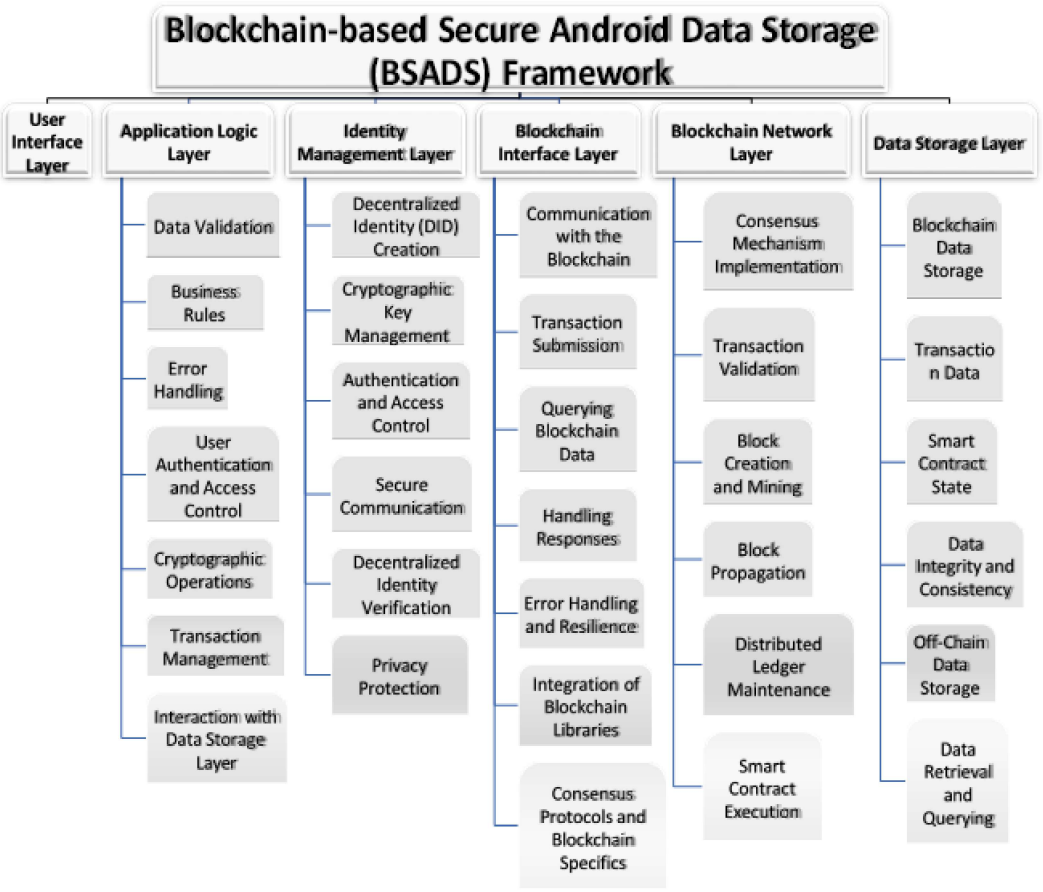


Figure 3.2 Proposed Framework: Blockchain-Based Secure Android Data Storage (BSADS)

3.2.2 System Architecture:

The hybrid architecture of VoteChain is illustrated in Fig. 3.3 and comprises the following core components:

1. **Mobile App (Flutter):**
   Offers a responsive and user-friendly interface for registration, MFA login, vote casting, candidate selection, and real-time result updates (including dynamic graphical visualizations).
2. **Backend System (Firebase + Web3dart.dart + QuickNode API):**
   Handles API requests, input validation, and secure transaction processing. It serves as the communication bridge between the mobile app, the Ethereum network, and off chain databases.
3. **Smart Contracts (Solidity):**
   Deployed on Ethereum Sepolia, these contracts handle vote recording and tallying. They enforce critical voting logic (e.g., verifying constituency eligibility) while minimizing on-chain operations to reduce gas costs.

4. **Blockchain Network (Ethereum Sepolia):**
   Provides an immutable ledger via a Proof-of-Stake consensus mechanism. Services like QuickNode support rapid, cost-effective connectivity and low network congestion.
5. **Off-Chain Database (Firebase):**
   Manages non-critical data such as voter profiles, election metadata, candidate details, and logs. It offers real-time data synchronization and enforces role-based access control, ensuring efficient off-chain operations.

Overall Data Flow:

A user initiates an action (e.g., casting a vote) through the Flutter app. The application logic validates the request and ensures eligibility. The backend (via Firebase and Web3dart.dart with QuickNode API) submits the vote to the smart contract on Ethereum Sepolia. The vote is recorded on-chain, while off-chain metadata is concurrently updated in Firebase. Real-time results are displayed in the app, and administrators manage concurrent elections as needed.

Summary of the Proposed System:

By segregating essential vote-recording functions on-chain from ancillary operations handled off-chain, VoteChain achieves:

- **Immutability and Trust:**
  Permanent, tamper-proof vote recording on the blockchain.
- **High Throughput:**
  Reduced network congestion and latency via off-chain processing.
- **User-Centric Experience:**
  A seamless mobile interface with secure MFA and real-time updates.
- **Dynamic Election Management:**
  Efficient administration of multiple elections with minimal on-chain load.
- **Enhanced Security:**
  Robust cryptographic measures and strict access controls to protect sensitive data.

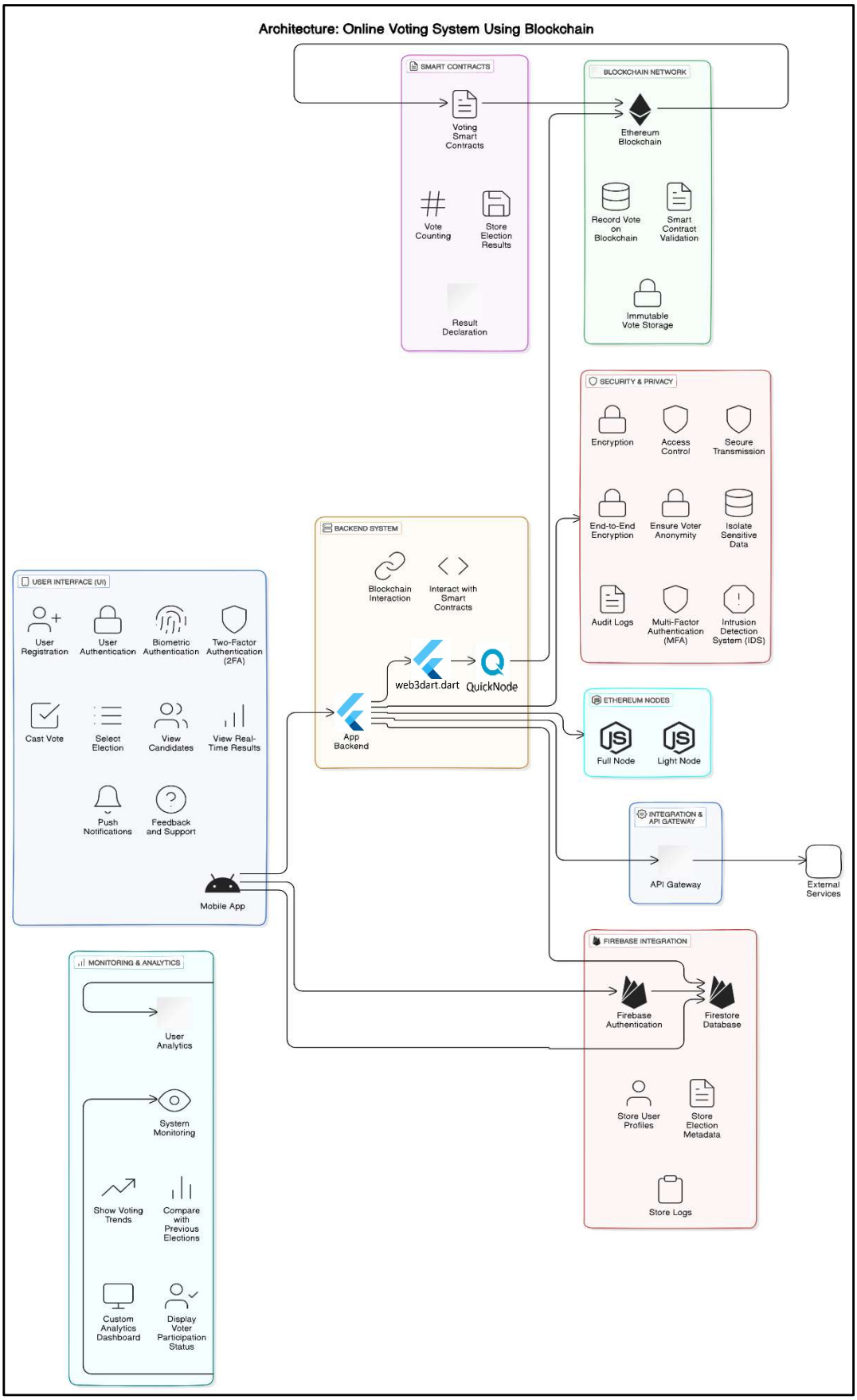


Figure 3.3 System Architecture

3.3 System Requirements Specification:

VoteChain's design is driven by a comprehensive analysis of both functional and non-functional requirements to address modern electoral challenges.

3.3.1 Functional Requirements:

1. **Voter Registration & Authentication:**
   Users register using verifiable credentials and receive unique IDs. MFA (e.g., OTPs, biometrics) reinforces security before voting.
2. **Voting Process:**
   Voters cast ballots for various election types (e.g., Gram Panchayat, Lok Sabha, Vidhan Sabha). Smart contracts ensure eligibility, prevent duplicate voting, and enforce constituency boundaries.
3. **Real-Time Results & Transparent Voting:**
   Votes are tallied on-chain in real time while off-chain systems (e.g., Firebase) capture supplementary metadata for dynamic analytics and graphical visualization.
4. **Administrative Control & Dynamic Election Management:**
   Administrators can create, schedule, and manage multiple elections concurrently. Off-chain candidate and party registrations reduce blockchain load while enabling real time data modifications.
5. **Edge Case Handling & Audit Trails:**
   The system prevents duplicate voting, restricts voting to designated constituencies, and maintains secure audit logs.

3.3.2 Non-Functional Requirements:

1. **Performance & Scalability:**
   The hybrid architecture supports high transaction volumes with minimal latency and is designed to scale for millions of users.
2. **Security & Data Integrity:**
   End-to-end encryption, MFA, blockchain immutability, and regular security audits ensure robust protection against unauthorized access.
3. **Usability & Accessibility:**
   A responsive and intuitive mobile interface accommodates users of varying technical backgrounds through clear visual hierarchies and easy navigation.
4. **Cost Efficiency:**
   Limiting on-chain operations to critical functions minimizes gas fees and overall operational costs.
5. **Compliance & Privacy:**
   The design adheres to data privacy regulations and electoral laws, maintaining secure logs and robust, role-based access controls.
6. **Responsiveness:**
   Sub-second response times provide a smooth and engaging user experience.

3.3.3 Software Requirements:

1. **Blockchain Network:** Ethereum (Sepolia)
2. **Smart Contract (Development):** Solidity
3. **Mobile App Development Framework:** Flutter, Android
4. **Database:** Firebase Database
5. **Development and Testing Tools:** Android Studio, Remix IDE
6. **API Integration:** Web3dart (package), QuickNode

Rationale:

The chosen technologies ensure high performance and scalability while maintaining robust security and cost-effectiveness. Flutter guarantees a consistent user experience; Firebase efficiently handles off-chain data; Ethereum (via QuickNode) provides secure and immutable vote recording; and Web3dart.dart enable seamless integration between all components.

3.3.4 Hardware Requirements:

1. **Mobile Devices:** Smartphone, Tablets (Android and iOS)
2. **Servers:** Blockchain nodes (Sepolia Testnet), Firebase
3. **Development and Testing Hardware:** Laptops/Desktop for developers ,Test devices (various models of smartphones and tablets)

3.3.5 Security Requirements:

The Security Requirements detail the necessary measures to protect user data and maintain the integrity of the voting process.

1. **Data Encryption:**
   All data transmitted between the mobile application, backend, and blockchain must be encrypted using end-to-end encryption protocols (e.g., AES-256) to ensure the confidentiality and integrity of sensitive information.
2. **Access Control:**
   The system must implement role-based access control (RBAC), ensuring that only authorized personnel (e.g., election officials) can access sensitive administrative features while keeping voter data secure and private.
3. **Authentication and Authorization:**
   The application must utilize multi-factor authentication (MFA) to verify voter identities. This will include password protection, OTPs, and biometric verification to enhance security and prevent unauthorized access.
4. **Data Integrity:**
   The system must ensure data integrity through the use of blockchain technology, where all transactions (votes and other) are recorded in an immutable ledger. Additionally, the application should maintain audit logs for all critical operations, allowing election officials to verify the integrity of the voting process without compromising voter privacy.

3.4 Feasibility Analysis:

3.4.1 Technical Feasibility:

1. **Blockchain for Vote Storage:**
   Ethereum or a similar blockchain network ensures secure, immutable, and tamper-proof vote storage. Smart contracts enable vote counting and ensure that no duplicate voting occurs.
2. **Firebase for Metadata:**
   Storing non-sensitive data such as voter logs and metadata in Firebase provides a cost-effective and efficient solution for off-chain operations, reducing blockchain transaction costs.
3. **User-Friendly Authentication:**
   Multi-factor authentication (MFA) with biometric verification provides a secure and easy way for voters to register and authenticate.

3.4.2 Financial Feasibility:

1. **Cost of Blockchain Transactions:**
   Public blockchains like Ethereum have transaction costs, but this can be minimized by using Layer 2 scaling solutions or consortium blockchains. Additionally, operations such as vote recording are optimized to avoid unnecessary costs.
2. **Firebase Usage Costs:**
   Firebase is utilized for handling non-voting-related metadata and logs, which reduces the need for on-chain storage, keeping costs low.

3.4.3 Operational Feasibility:

1. **Ease of Use:**
   The mobile app is designed for ease of use, with a straightforward interface for voters to register, select elections, and cast votes. Multi-factor authentication ensures that even non-technical users can participate securely.
2. **Handling Peak Voting Periods:**
   The system is scalable to handle a large number of users during peak voting times, ensuring that even national elections can be conducted smoothly.

3.5 Design Phase:

3.5.1 Timeline chart/Gantt Chart:

- **Sem 7:**

The following is Timeline of Sem 7 describing the tasks completed in Sem 7 in order to plan for the proposed model:

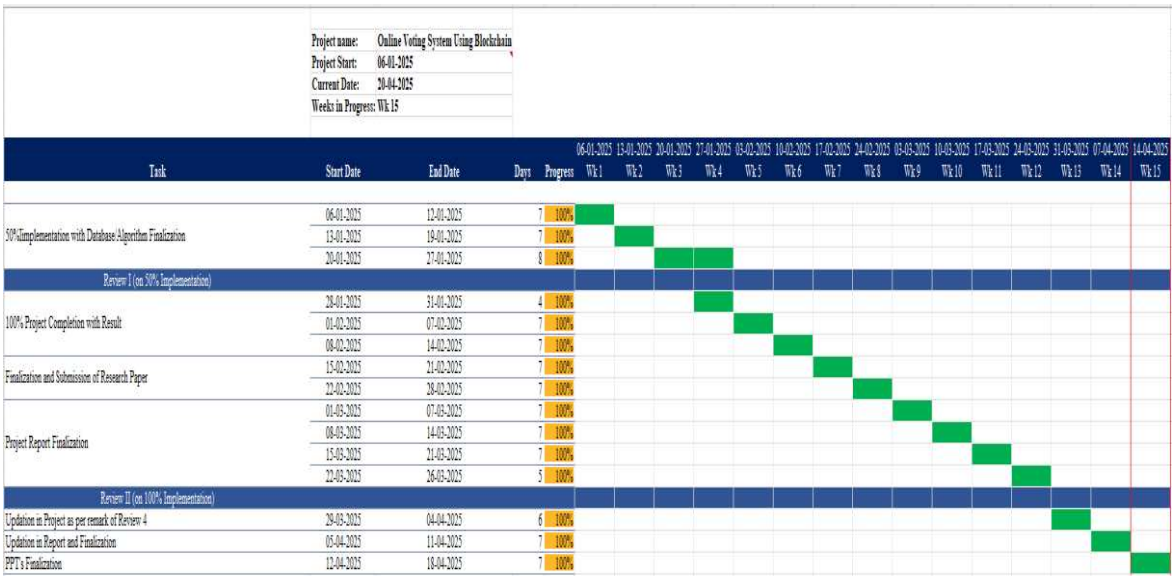
Project name:	Online Voting System Using Blockchain
Project Start:	11-07-2024
Current Date:	16-10-2024
Weeks in Progress:	Wk 14

Task	Start Date	End Date	Days	Progress
Group Finalization Domain	11-07-2024	17-07-2024	6	100%
Topic finalization Project Name submission	18-07-2024	24-07-2024	6	100%
Topic finalization Project Name submission (objectives problem statement scope)	25-07-2024	31-07-2024	6	100%
Literature study & limitation of existing system (refer IEEE ACM Springer Elsevier journal pap	01-08-2024	07-08-2024	6	100%
Abstract, Problem statement scope & objective Finalization.	08-08-2024	14-08-2024	6	100%
Implementation plan /Timeline chart for sem vii & viii	15-08-2024	21-08-2024	6	100%
SRS-Functional & Non functional requirements, S/W Engg Model	22-08-2024	28-08-2024	6	100%
Proposed system architecture/flow chart & Analysis & Design: Use Case, DFD-0,Sequence	29-08-2024	04-09-2024	6	100%
Review I(on Activity 3 to 8)				
Revew Paper Publication (Survey Paper)	05-09-2024	11-09-2024	6	100%
Analysis & Design: Data model/data set, Class diagram,Activity diagram, Deployment Diagra	12-09-2024	18-09-2024	6	100%
Wireframe/Algorithm & flow chart/Input form design/Output Report Design	19-09-2024	25-09-2024	6	100%
RMMM Plan, Feasibility analysis & Test case Design	26-09-2024	02-10-2024	6	100%
Review II (on Activity 3 to 13)				
Project Report Preparation	03-10-2024	09-10-2024	6	100%
Finalisation of PPT's & project report with print & guide signature	10-10-2024	16-10-2024	6	100%

Figure 3.4 Timeline Chart (Sem 7)

- **Sem 8:**

The following is Timeline of Sem 8 describing the tasks completed in Sem 8 in order to build the proposed model:



Project name:	Online Voting System Using Blockchain
Project Start:	06-01-2025
Current Date:	20-04-2025
Weeks in Progress:	Wk 15

Task	Start Date	End Date	Days	Progress
	06-01-2025	12-01-2025	7	100%
50%Implementation with Database Algorithm Finalization	13-01-2025	19-01-2025	7	100%
	20-01-2025	27-01-2025	8	100%
Review I (on 50% implementation)				
	28-01-2025	31-01-2025	4	100%
100% Project Completion with Result	01-02-2025	07-02-2025	7	100%
	08-02-2025	14-02-2025	7	100%
	15-02-2025	21-02-2025	7	100%
Finalization and Submission of Research Paper	22-02-2025	28-02-2025	7	100%
	01-03-2025	07-03-2025	7	100%
	08-03-2025	14-03-2025	7	100%
Project Report Finalization	15-03-2025	21-03-2025	7	100%
	22-03-2025	26-03-2025	5	100%
Review II (on 100% Implementation)				
Updation in Project as per remark of Review 4	28-03-2025	04-04-2025	6	100%
Updation in Report and Finalization	05-04-2025	11-04-2025	7	100%
PPT's Finalization	12-04-2025	18-04-2025	7	100%

Figure 3.5 Timeline Chart (Sem 8)

3.5.2 Data Flow Diagrams:

1. **LEVEL 0:**
   This Data Flow Diagram (DFD) at Level 0 represents an Online Voting System that uses blockchain technology.

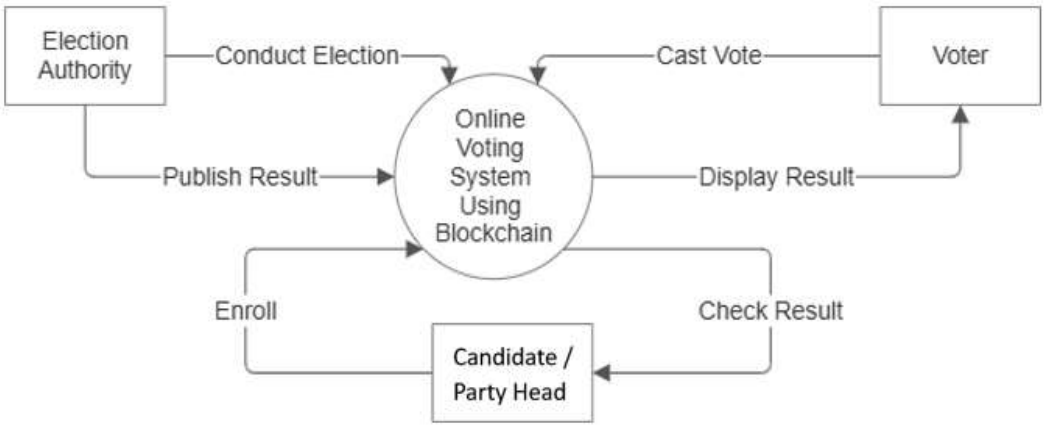


Figure 3.6 DFD Level 0

1. **Admin:**
   - **Conduct Election:** The admin initiates and manages the election process.
   - **Publish Result:** After the election, the admin publishes the results through the system.
2. **Voter:**
   - **Cast Vote:** Voters cast their vote securely in the system.
   - **Display Result:** Voters can view the results once they are published.
3. **Candidate and Party Head:**
   - **Enrol**: Can register or enrol in the election system to be part of the election.
   - **Check Result:** After the election, can check the results.
4. **Online Voting System Using Blockchain (central process):**
   This is the core system where all election activities are managed. The use of blockchain ensures security, transparency, and immutability of the voting process and results.

The system connects the Admin, Voter, Candidate, and Part head through interactions such as voting, enrolling, publishing, and viewing results, all secured by blockchain technology.

2. Data Flow Diagram (LEVEL 1):

This Level 1 DFD for the Online Voting System using Blockchain provides a detailed overview of the system's core functions:

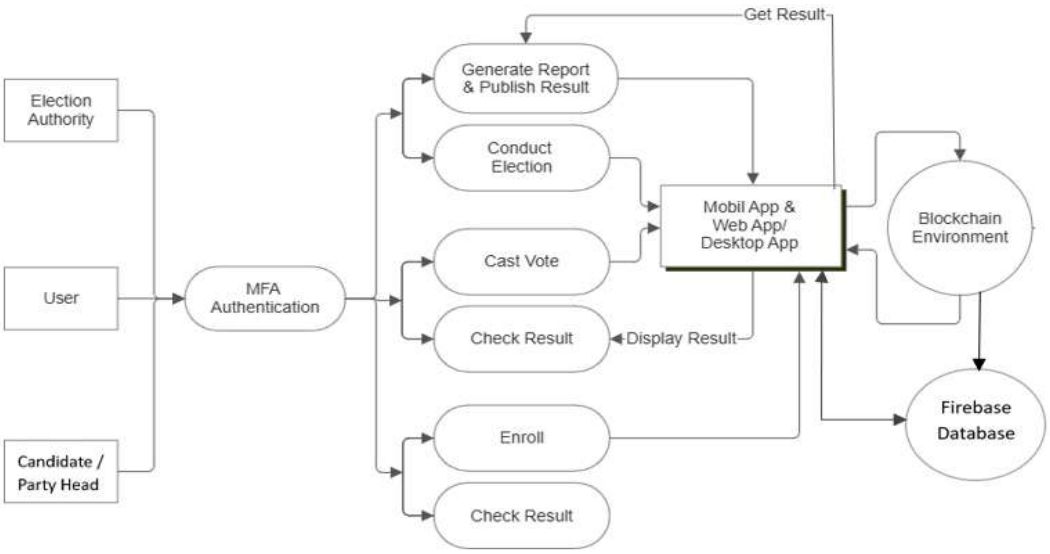


Figure 3.7 DFD Level 1

1. **Election Authority:**
   - Responsible for managing the election process, including conducting the election, generating reports, and publishing results.
2. **User:**
   - A voter or participant who interacts with the system after completing MFA Authentication (Multi-Factor Authentication).
   - The user can cast their vote, check election results, and access other relevant features once authenticated.
3. **Candidate and Party Head:**
   - A candidate/party head for the election who can enrol in the system and later check the election results after the voting process.
4. **Mobile/Web/Desktop App:**
   - Serves as the user interface for both voters and candidate/party head to interact with the system. It facilitates tasks such as casting votes, checking results, and enrolling in the election.
   - The app interacts directly with the Blockchain Environment for secure data processing.
5. **Blockchain Environment:**
   - Ensures all votes, results, and election data are securely stored and tamper-proof, guaranteeing transparency and integrity.
   - The system retrieves results from the blockchain and displays them through the app interface.

In essence, the diagram depicts a secure, blockchain-based voting system where election authorities, users, and candidate/party head interact with the system through various steps, all secured by multi-factor authentication and blockchain technology.

3.5.3 Use Case Diagram:

This Use Case Diagram shows how the Online Voting System Using Blockchain works:

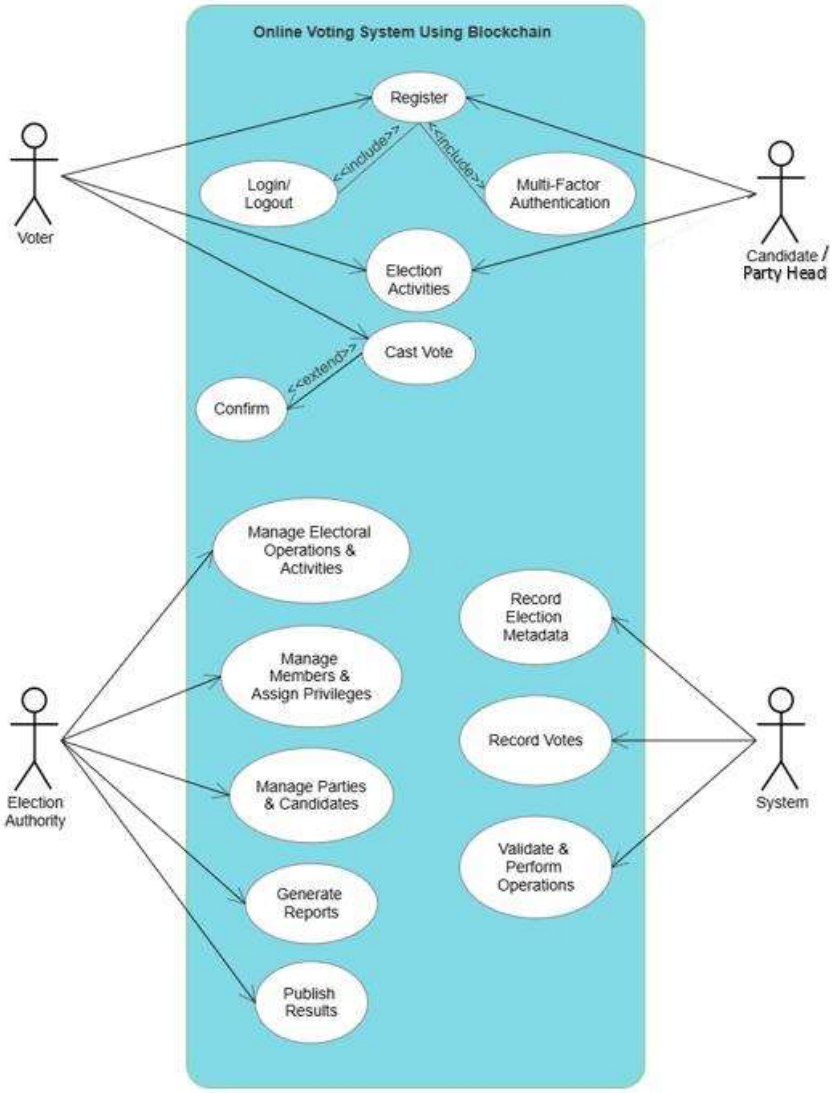


Figure 3.8 Use Case Diagram

- **Voter:**
  Registers, logs in, authenticates with multi-factor, views election activities, casts votes, and confirms voting.
- **Candidate and Party Head:**
  Registers/logs in, and views election activities and can take part in it.
- **Election Authority:**
  Manages operations, members, parties, generates reports, and publishes results.
- **System:**
  Records election data, votes, and validates operations using blockchain.

3.5.4 Flowchart Diagram:

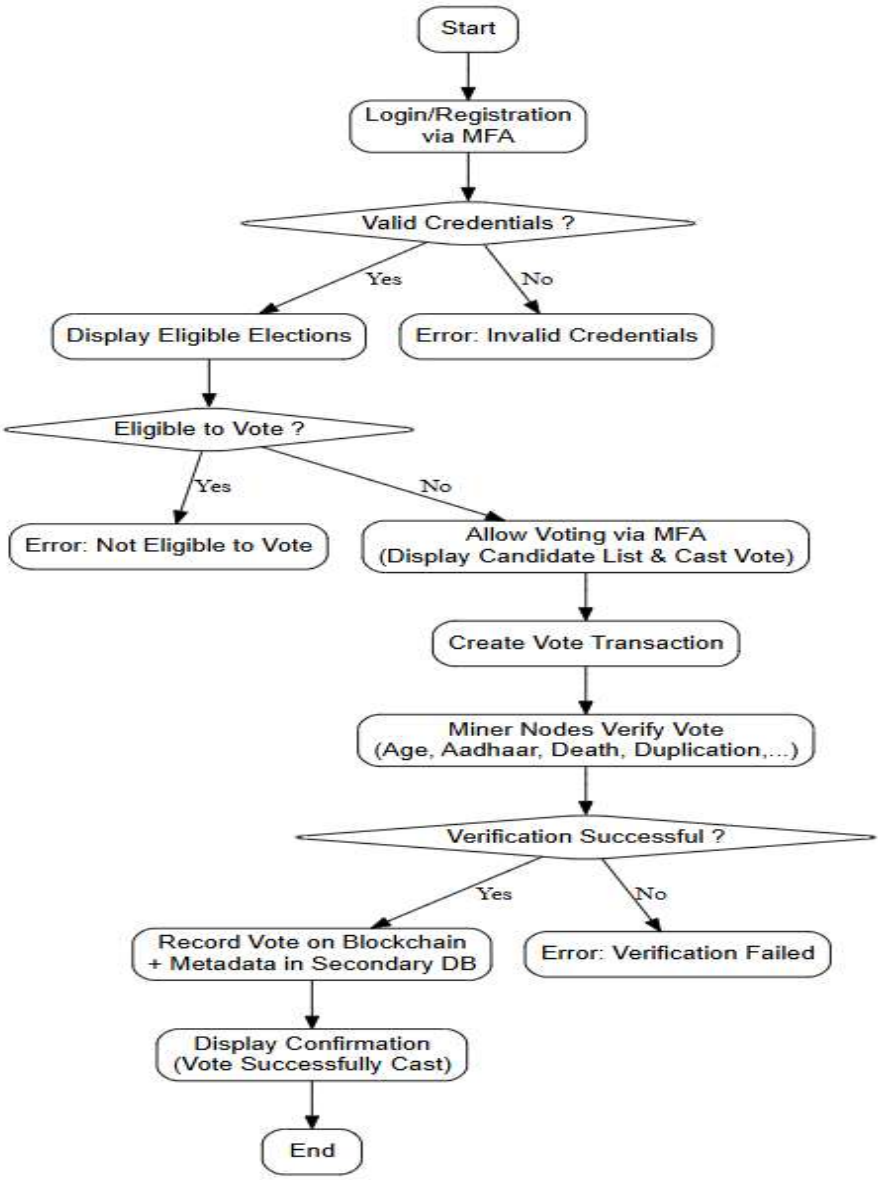


Figure 3.9 Flowchart Diagram (Citizen - Voting)

3.5.5 Sequence Diagram:

This sequence diagram represents the interaction flow of an Online Voting System Using Blockchain.

1. **User Registration:**
   The user registers into the system (invoking the Register () function).
2. **MFA (Multi-Factor Authentication) :**
   After registration, the user undergoes an authentication process (Authenticate () function) to ensure secure access.
3. **View Candidate:**
   The authenticated user can view candidates for an election (View () function).

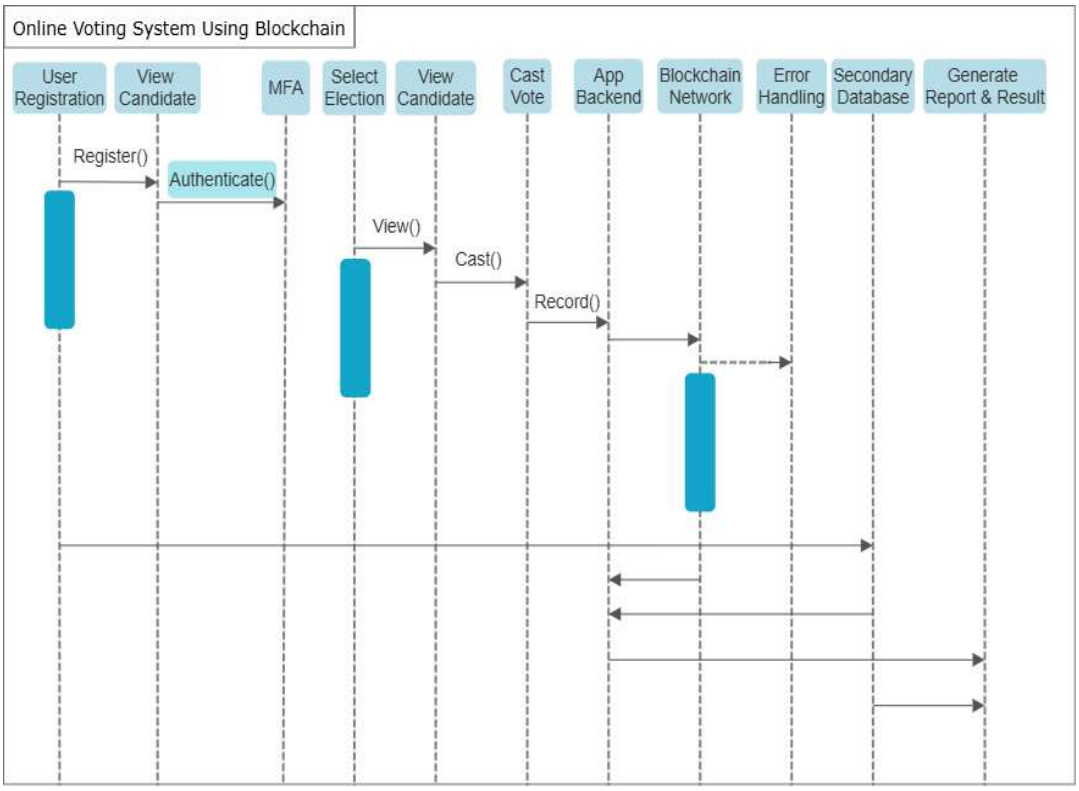


Figure 3.10 Sequence Diagram (Citizen - Voting)

4. **Select Election:**
   The user selects an election in which they wish to participate.
5. **Cast Vote:**
   After viewing the candidates and selecting the election, the user casts their vote (Cast () function).
6. **App Backend:**
   The application backend records the user's vote.
7. **Blockchain Network:**
   The vote is then recorded onto the blockchain network, ensuring immutability and security (Record () function).
8. **Error Handling:**
   If any issues arise, error handling processes are triggered.
9. **Secondary Database:**
   A secondary database may store non-critical data.
10. **Generate Report & Result:**
    Once voting is complete, reports and results are generated and displayed.

Each entity in the system (user, app backend, blockchain, etc.) interacts to maintain the flow and integrity of the voting process.

3.6 RMMM Plan:

Table 3.1 RMMM Summary

Sr No.	Risk	Category	Probability	Impact	RMMM Code
1.	Smart contract vulnerabilities & blockchain network outages	Technical	20%	High	RMMM 1
2.	Blockchain network congestion & high gas fees	Technical/ Financial	30%	High	RMMM 2
3.	Backend (Firebase/QuickNode) downtime or performance issues	Technical	20%	Moderate	RMMM 3
4.	Cybersecurity breaches affecting app, blockchain, and backend systems	Technical	20%	Moderate	RMMM 4
5.	Authentication failures and duplicate voting	Operational	20%	High	RMMM 5
6.	Data inconsistency between blockchain records and off-chain (Firebase) metadata	Technical	10%	Moderate	RMMM 6
7.	Regulatory and legal changes impacting blockchain usage in elections	External	20%	Low	RMMM 7
8.	Integration issues between Flutter app, QuickNode endpoints, and Firebase	Technical	10%	Low	RMMM 8
9.	Errors in vote tallying and result visualization	Operational	10%	High	RMMM 9
10.	Voter privacy and anonymity risks	Security	20%	High	RMMM 10

Table 3.2 RMMM Detailed Description

Risk ID	Description	Mitigation	Monitoring	Management
RMMM 1	Vulnerabilities in the smart contract or blockchain network outages could lead to vote manipulation or service disruption.	• Conduct thorough audits and security testing of smart contracts. • Utilize established security libraries and best practices.	Regular smart contract audits. Continuous network and transaction monitoring.	Implement immediate patching and deploy fallback mechanisms if vulnerabilities are exploited.
RMMM 2	Blockchain network congestion may result in high gas fees and delayed transactions, affecting real-time voting.	• Optimize contract functions and minimize on-chain operations. • Schedule non-critical transactions during off-peak hours.	Monitor blockchain network performance and gas fee trends in real time.	Adjust transaction strategies dynamically and consider temporary off-chain solutions for vote aggregation.
RMMM 3	Downtime or performance issues with Firebase or QuickNode endpoints could disrupt voting operations and metadata logging.	• Scale backend infrastructure and incorporate redundancy. • Maintain backup servers and database replicas.	Use real-time performance monitoring tools and set alert thresholds.	Switch to backup systems promptly and communicate issues to stakeholders.
RMMM 4	Cyberattacks on the app, blockchain, or backend may compromise sensitive data or	• Implement robust cybersecurity measures including encryption and	Deploy intrusion detection systems and monitor network traffic	Activate incident response plans and conduct post-incident reviews to improve defenses.

	disrupt service.	multi-factor authentication. • Regularly update and patch all components.	for anomalies.	
RMMM 5	Weak authentication systems might allow unauthorized access or duplicate voting, undermining election integrity.	• Enforce strict multifactor authentication and implement duplicate vote-checking mechanisms. • Use secure session management practices.	Audit user sessions and review authentication logs continuously.	Provide rapid user support and update authentication protocols as needed.
RMMM 6	Inconsistencies between on-chain vote records and off-chain metadata can lead to disputes over election results.	• Establish regular data reconciliation routines between blockchain and Firebase. • Implement checksums or hash comparisons for critical data sets.	Automate consistency checks and generate regular reports.	Initiate manual audits and corrective procedures if discrepancies are detected.
RMMM 7	New or changing regulations could affect the use of blockchain in voting systems and require system adaptations.	• Stay updated with regulatory developments through legal advisories. • Design the system with flexibility for legal compliance adjustments.	Monitor government and legal announcements on electoral technology.	Revise policies and update system architecture to comply with new regulations as required.

RMMM 8	Integration challenges between the Flutter app, QuickNode endpoints, and Firebase may result in operational disruptions.	• Perform comprehensive integration and system testing before deployment. • Use modular architecture to isolate components for easier troubleshooting.	Continuously log and monitor integration performance and error reports.	Deploy rapid troubleshooting protocols and iterative fixes when integration issues are identified.
RMMM 9	Errors in vote tallying or result visualization could misrepresent election outcomes.	• Implement redundancy checks and independent verification processes for vote counts. • Utilize both list-based and graphical formats to cross-check results.	Use real-time analytics dashboards and periodic audits of results.	Cross-check with backup systems and engage in manual reviews to correct any discrepancies.
RMMM 10	Compromise of voter anonymity or data privacy could erode trust in the system.	• Employ advanced encryption, anonymization techniques, and strict access controls. • Design privacy-preserving data handling practices.	Regularly conduct privacy audits and vulnerability assessments.	Update privacy policies and execute immediate remedial actions in response to breaches.

Chapter: 4

Implementation

VoteChain is a hybrid, cross-platform online voting application that leverages blockchain exclusively for immutable vote recording, while off-chain systems (using Firebase) handle supporting functions. This chapter details the end-to-end implementation of the system, explaining the technical decisions, design flows, and verification protocols that ensure a secure, scalable, and efficient online voting process.

4.1 Frontend Development:

Framework and UI Design:

- **Platform:**
  The mobile application is developed using Android-Flutter, ensuring a modern, responsive, and intuitive interface.
- **User Screens:**
  - **User Registration and Authentication:**
    Incorporates multifactor authentication (password, biometric, OTP) to ensure secure login.
  - **Election Dashboard:**
    Displays a hierarchical view of elections at local, state, and national levels with dedicated views for voters, candidates, party heads, and election officials.
  - **Voting Interface:**
    Allows voters to view party, candidate details, cast their votes, and receive real-time feedback.
  - **Result Visualization:**
    Provides both tabular and graphical representations of election outcomes, ensuring transparency and ease of understanding.

Figure 4.1
Login Screen

Figure 4.2
Citizen Dashboard

Figure 4.3
Voting Screen

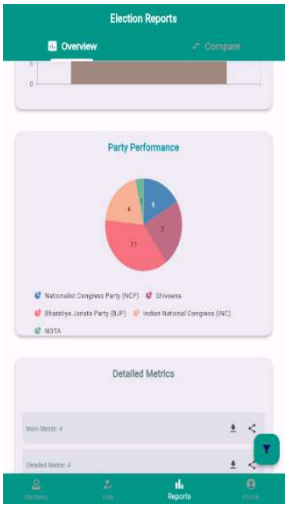


Figure 4,4
Reports Screen

Figure 4.5
Admin Dashboard

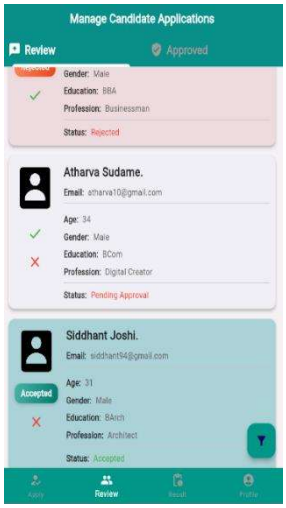


Figure 4.6
Party Head Dashboard

Figure 4.7
Citizen Dashboard

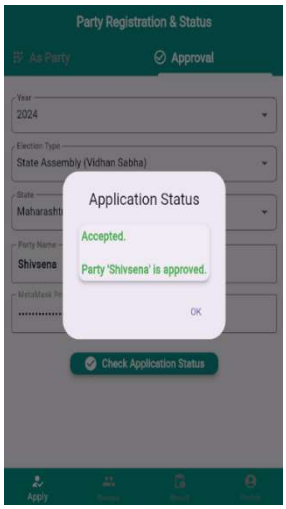


Figure 4.8
Party-Application-Status Screen

4.2 Backend and Middleware Integration:

Middleware Role:

- Acts as the secure intermediary between the Flutter frontend, the blockchain network, and the off-chain data store (Firebase).
- Validates all user actions to ensure that only authenticated and authorized transactions are processed.

Data Exchange and API Integration:

- Secure endpoints facilitate data synchronization and logging.
- A Level 1 Data Flow Diagram (see Fig 3.7) illustrates how user inputs are processed, routed to the blockchain for vote recording via QuickNode endpoints (see below Fig 4.9), and sent to Firebase for metadata management.

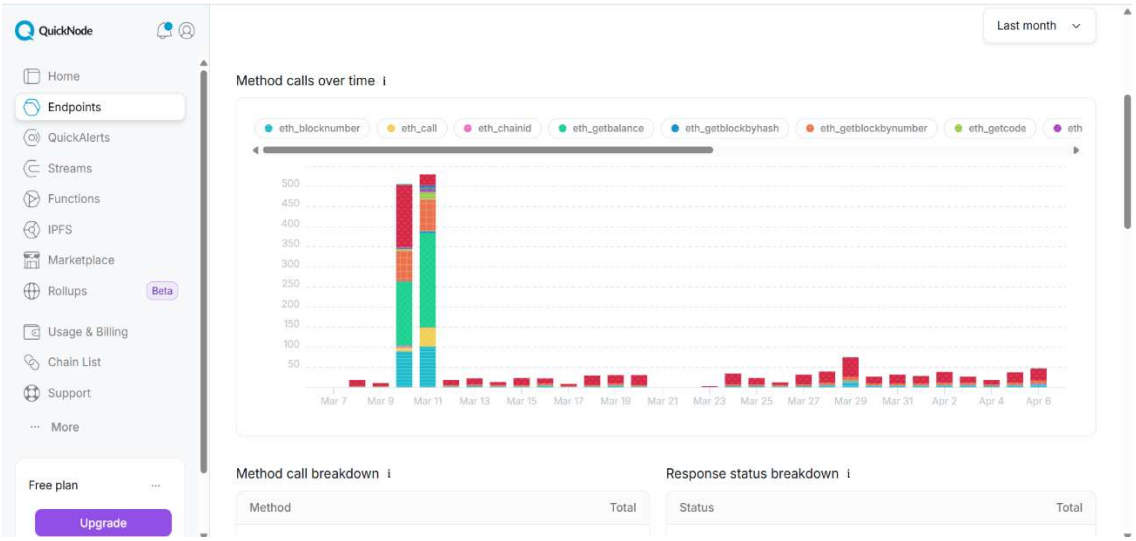


Figure 4.9 QuickNode Dashboard

4.3 Off-Chain Data Management:

Data Handling Strategy:

- All non-critical data (e.g., user profiles, election configurations, candidate/party applications, detailed logs) are managed off-chain using Firebase.
- This strategy minimizes the load on the blockchain, reducing gas fees and ensuring high performance during large-scale elections (see Fig 4.11).

Security and Scalability:

- Robust access controls and encryption safeguard sensitive data (see Fig 4.10).
- The system supports simultaneous elections and real-time updates without exposing internal data structures.

Integration with Blockchain:

- A blockchain library (e.g., web3dart) is used within the mobile app to interact with the deployed smart contract for operations like vote casting and vote count retrieval.

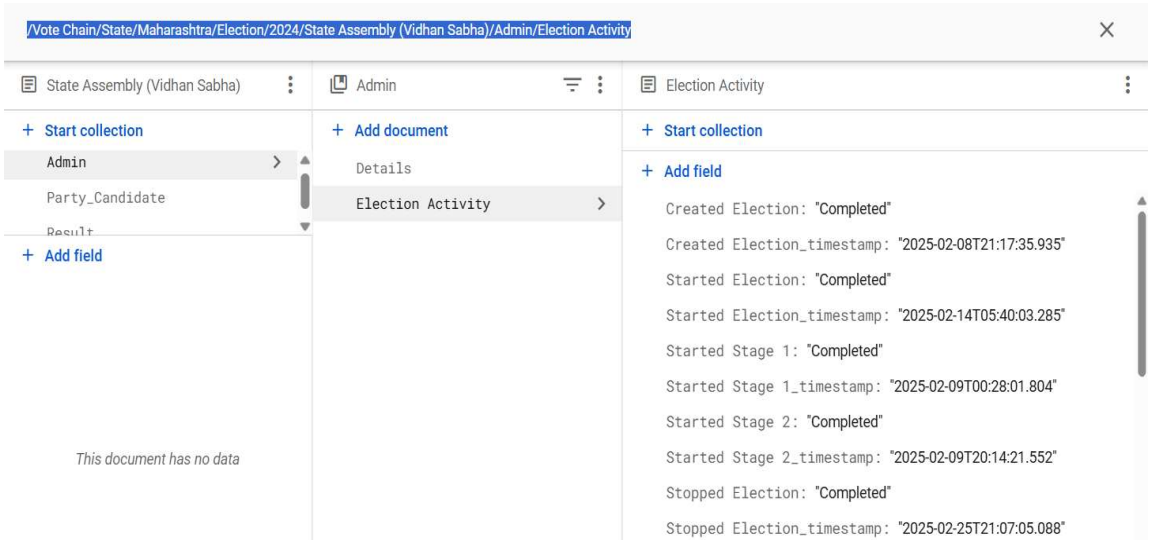


Figure 4.10 Firebase Dashboard (Admin - Activity)

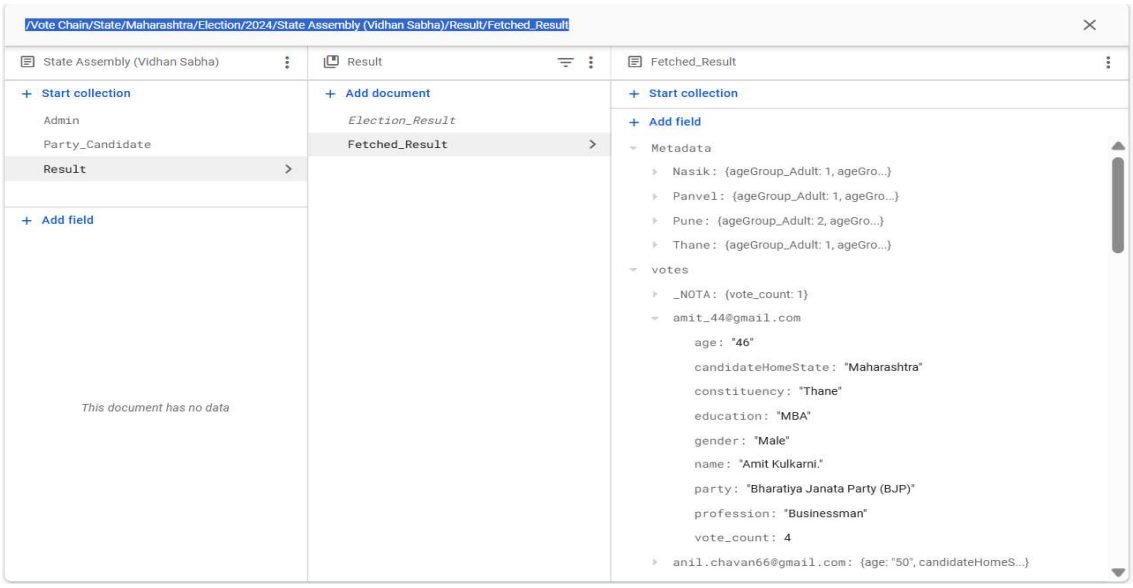


Figure 4.11 Firebase Database (Fetched-Data from Blcokchain)

4.4 Blockchain Integration:

Smart Contract Deployment:

- A Solidity-based smart contract is deployed on the Ethereum Sepolia testnet via QuickNode.
- Role: The smart contract is solely responsible for recording votes and tallying results, thereby ensuring transparency and reducing on-chain operations to minimize gas fees.

Interaction Flow:

- The mobile app interacts with the smart contract via the middleware, ensuring that only validated votes are recorded on the blockchain (see Fig 4.12).
- A Sequence Diagram (see Fig. 3.10) details the flow between the mobile app, middleware, and the smart contract.

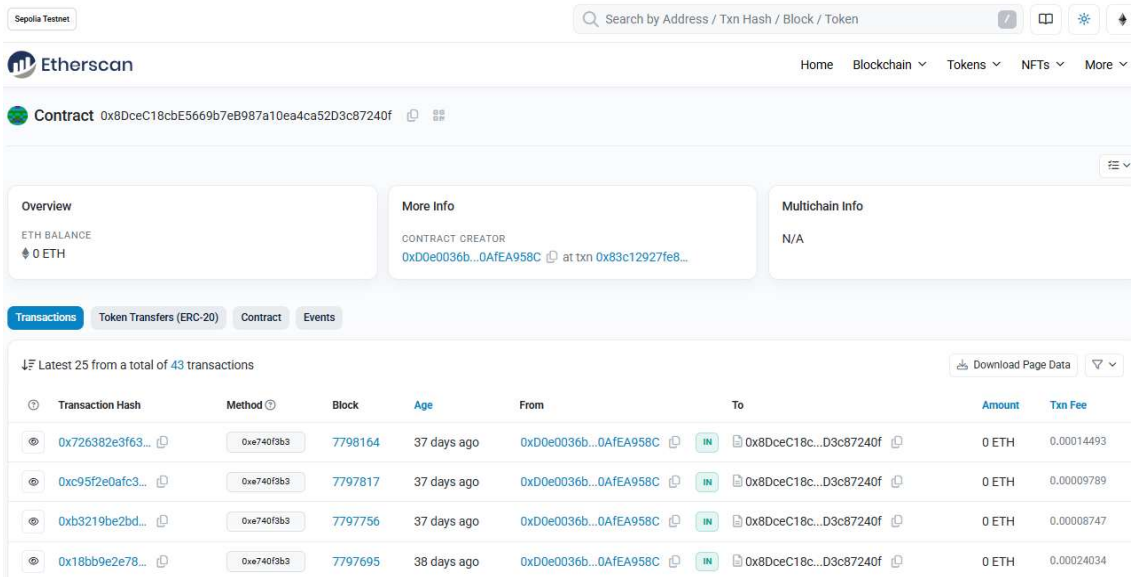


Figure 4.12 Ethereum Dashboard

4.5 Security and Privacy Measures:

Authentication and Access Control:

- Multifactor authentication (MFA) is implemented for all users.
- Role-based access control ensures that only authorized users can perform specific actions (e.g., vote casting, election management).

Data Encryption:

- End-to-end encryption is used for data transmission between the app, middleware, and the blockchain network.
- Sensitive data stored on Firebase is encrypted at rest to prevent unauthorized access.

Smart Contract Security:

- The smart contract undergoes thorough security audits and formal verification to identify and mitigate vulnerabilities.
- On-chain operations are minimized to reduce the attack surface.

Continuous Monitoring:

- Regular security audits, real-time network traffic monitoring, and periodic penetration testing are conducted to identify and address any emerging threats.

4.6 Performance Optimization:

System Scalability:

- The architecture leverages off-chain data management to relieve the blockchain network from processing non-critical operations, thereby improving transaction speed and reducing gas fees.
- Dynamic scaling strategies are employed to handle high loads during peak voting periods.

Transaction Batching:

- Votes are batched for efficient processing on the blockchain, reducing latency and cost per transaction.

Caching and Load Balancing:

- Data caching and load balancing mechanisms are implemented in the middleware to optimize API response times and ensure high availability.

Performance Monitoring:

- Real-time dashboards monitor transaction throughput, server load, and network latency.
- Regular stress tests simulate high concurrent user loads to validate system performance under peak conditions.

4.7 Testing and Deployment:

Testing is a critical component of VoteChain's implementation. A comprehensive testing protocol ensures that the system meets security, performance, and reliability requirements.

4.7.1 Authentication Test:

- **Description:** Verify the proper functioning of the multifactor authentication system (password, biometric, OTP).
- **Input:** User enters credentials and biometric data, followed by an OTP.
- **Expected Output:** Successful login, vote casting by accessing dashboard.

4.7.2 Vote Casting Test:

- **Description:** Ensure that votes are securely cast and recorded on the blockchain.
- **Input:** User selects an election and candidate, then confirms the vote.
- **Expected Output:** Vote is recorded on the blockchain, a transaction hash is returned, and a confirmation message is displayed.

4.7.3 Duplicate Voting Prevention:

- **Description:** Prevent users from double/false voting in the same election.
- **Input:** User attempts to cast a second/false vote in the same election.
- **Expected Output:** An error message indicates that the user has already voted, and no additional transaction is recorded.

4.7.4 Result Display Test:

- **Description:** Validate that real-time results are correctly fetched and displayed after the voting period ends.
- **Input:** User requests to view election results.
- **Expected Output:** Real-time results are fetched from the blockchain, displaying vote counts and overall turnout.

4.7.5 Data Encryption and Security Test:

- **Description:** Verify that all sensitive data is encrypted during transmission and storage.
- **Input:** Data submission during vote casting.
- **Expected Output:** Data remains encrypted and secure across all layers of the system.

4.7.6 Scalability Test:

- **Description:** Assess the system's ability to handle a high volume of concurrent users.
- **Input:** Simulated load of thousands of users casting votes simultaneously.
- **Expected Output:** The system processes all votes with minimal latency and no service degradation.

4.7.7 Failover and Recovery Test:

- **Description:** Ensure the system can recover from failures, such as server crashes or blockchain node disconnections.
- **Input:** Simulate a backend server crash during the voting process.
- **Expected Output:** The system automatically switches to a backup server with no data loss, maintaining the voting process uninterrupted.

Deployment Strategy:

- A CI/CD pipeline automates the testing and deployment process, ensuring that new releases are rigorously validated before production deployment.
- Scalable cloud infrastructure and continuous monitoring/logging enable proactive performance management and rapid incident response.

This chapter outlines the full implementation strategy for VoteChain. The approach leverages a robust combination of frontend design, secure middleware integration, efficient off-chain data management, and blockchain-based vote recording. Critical measures for security, performance, and thorough testing ensure that the system is ready for real-world elections with high scalability, transparency, and reliability.

Chapter 5

Result and Discussion

This chapter presents a comprehensive evaluation of the VoteChain prototype—an innovative online voting system that leverages blockchain technology alongside a secondary database (Firebase) for auxiliary functions. The discussion details the system's expected outcomes and the challenges encountered during development, along with the targeted solutions implemented. The hybrid architecture of VoteChain is designed to deliver a secure, transparent, scalable, and cost-effective electoral process while ensuring an intuitive user experience.

5.1 Result:

VoteChain is engineered to transform the electoral process by achieving the following key outcomes:

5.1.1 Secure and Transparent Voting Process:

- **Immutable Vote Recording:**
  Votes are recorded using a Solidity smart contract deployed on the Ethereum Sepolia testnet (accessed via QuickNode). This ensures that every vote is tamper-proof, permanently stored, and auditable.
- **Robust Authentication:**
  The system employs multifactor authentication (MFA), including biometric verification and OTPs, alongside strict role-based access control. These measures prevent unauthorized access and ensure that each voter casts only one vote.

5.1.2 Real-Time Results and Auditability:

- **Immediate Election Insights:**
  Real-time textual and graphical representations of election results provide transparent updates to voters and election officials, ensuring that results are continuously monitored as votes are tallied.
- **Comprehensive Audit Logs:**
  Election-related data—such as voter participation, metadata, and activity logs—are securely stored in Firebase. This off-chain storage facilitates thorough post-election audits while maintaining voter anonymity.

5.1.3 Cost Efficiency:

- **Optimized Blockchain Usage:**
  By confining on-chain operations to essential vote recording and tallying, and offloading non-critical tasks (e.g., user management and metadata storage) to Firebase, the system reduces gas fees by an estimated 40–60% and overall election expenses by up to 70–80%.

- **Reduction of Physical Infrastructure:**
  Leveraging blockchain technology minimizes the need for traditional electronic voting machinery and extensive manpower, further cutting operational costs for large-scale elections.

5.1.4 Scalability and Performance:

- **Dual-Layer Architecture:**
  The hybrid model supports diverse election types—from local to national—by combining blockchain's security with Firebase's real-time, hierarchical data structure. This approach efficiently manages high user loads and complex queries during peak voting periods.
- **Transaction Batching:**
  Grouping multiple transactions into batches minimizes the number of on-chain interactions, thereby reducing latency and enhancing overall system performance.

5.1.5 Enhanced Security and User Privacy:

- **End-to-End Encryption:**
  All communications between the mobile app, backend, and blockchain are encrypted, safeguarding sensitive data throughout the election process.
- **Data Segregation and Encryption:**
  Sensitive vote data is isolated on the blockchain for immutability, while non-sensitive information (such as voter profiles and metadata) is encrypted both at rest and in transit in Firebase, ensuring comprehensive data security.

5.1.6 Improved User Experience and Accessibility

- **Intuitive Interface:**
  Developed using Flutter, the mobile application offers a clean, responsive, and visually appealing interface. Detailed dashboards, smooth animations, and real-time visual analytics facilitate easy navigation for all stakeholders, including voters, candidates, party heads, and election officials.
- **Dynamic Adaptability:**
  The system's modular design accommodates evolving election configurations and candidate-party relationships, ensuring that VoteChain remains relevant for future electoral events.

5.1.7 Enhanced User Engagement and Participation:

- **Real-Time Participation Tracking:**
  The platform provides tools for monitoring voter turnout and engagement in real-time, enabling election officials to identify participation trends and implement strategies to boost voter turnout.

5.2 Challenges and Solutions:

The development of VoteChain involved addressing several technical and operational challenges. The following solutions were implemented to ensure system robustness and efficiency:

5.2.1 Blockchain Scalability and Transaction Efficiency:

- **Challenge:**
  Public blockchains like Ethereum face scalability limitations that can lead to network congestion, increased transaction times, and higher gas fees—especially during large-scale elections.
- **Solutions:**
  - **Layer 2 Scaling Solutions:**
    Integration of rollups or sidechains processes transactions off-chain before final settlement, reducing fees and improving speed.
  - **Transaction Batching:**
    Aggregating multiple votes into single transactions minimizes on-chain interactions.
  - **Selective On-Chain Storage:**
    Only essential vote data is stored on-chain while ancillary data is managed off-chain, optimizing resource usage.
  - **Private/Permissioned Blockchain Option:**
    For high-capacity national elections, a private blockchain could be utilized to better control transaction throughput and costs.

5.2.2 Security Threats and Data Integrity:

- **Challenge:**
  The system must be resilient against cybersecurity threats—including DDoS attacks, unauthorized access, and data breaches—while preserving voter privacy.
- **Solutions:**
  - **Strong End-to-End Encryption:**
    Encrypting all communications ensures that sensitive data is secure during transmission.
  - **Enhanced Authentication Measures:**
    Multifactor authentication and role-based access control protect voter identities and prevent duplicate voting.
  - **DDoS Mitigation:**
    Rate limiting on API calls, regular penetration testing, and security audits mitigate the risk of cyberattacks.
  - **Smart Contract Audits:**
    Rigorous audits of the smart contracts ensure that vulnerabilities are identified and addressed before deployment.

5.2.3 Prevention of Duplicate Voting and User Privacy:

- **Challenge:**
  Ensuring that each voter casts only one vote while maintaining the anonymity and privacy of voter data.
- **Solutions:**
  - **Secure Vote Tracking:**
    A secure flag maintained in Firebase tracks voting status, ensuring each voter can only participate once.
  - **Data Segregation:**
    Sensitive vote data is isolated on the blockchain, while non-sensitive data is stored in Firebase with robust encryption.
  - **Comprehensive Audit Logs:**
    Detailed logs enable transparent audits without compromising voter privacy.

5.2.4 User Interface Complexity and Usability

- **Challenge:**
  Delivering a professional, user-friendly interface that meets needs of a diverse user base, including voters, candidates, party heads, and election officials.
- **Solutions:**
  - **Flutter-Based Design:**
    The mobile app's intuitive design and consistent visual theme simplify navigation and enhance user engagement.
  - **Real-Time Visual Analytics:**
    Interactive dashboards and graphical representations provide immediate feedback, ensuring that users can easily monitor election progress.

5.2.5 Post-Election Data Management:

- **Challenge:**
  Permanently storing all vote data on-chain could lead to high costs and potential security vulnerabilities.
- **Solutions:**
  - **Off-Chain Metadata Archiving:**
    Non-critical logs and metadata are stored in Firebase, while the essential vote counts remain secured on the blockchain. This balance preserves auditability and reduces on-chain storage costs.
  - **Root Storage Reset:**
    Once the election process concludes and all vote data and metadata have been fetched, the root storage of all blockchain network servers can be cleared or reset. By doing so, the same storage can be reused for further elections, significantly reducing storage costs over time.

The VoteChain prototype demonstrates a robust and innovative approach to modernizing the electoral process. By integrating blockchain for secure, immutable vote recording with Firebase for efficient data management and real-time processing, the system achieves enhanced security, transparency, scalability, and cost efficiency. The comprehensive solutions implemented to address scalability, security, and usability challenges make VoteChain a viable platform for conducting secure and efficient elections on both local and national scales. This hybrid architecture positions VoteChain as a promising solution for transforming traditional voting systems into modern, digitally secure electoral processes.

This chapter synthesizes the technical and operational strengths of VoteChain, outlining how its hybrid design effectively meets the evolving demands of modern elections while ensuring a seamless user experience and robust data security.

Chapter 6

Conclusion

6.1 Conclusion:

The development of VoteChain—a blockchain-based online voting system—marks a significant advancement in modernizing electoral processes, especially for large-scale democracies such as India. By strategically leveraging Ethereum's blockchain (via the Sepolia testnet and QuickNode) solely for immutable vote recording and offloading ancillary operations to a Firebase-backed off-chain system, VoteChain achieves a robust balance between security, scalability, and cost efficiency.

Hybrid Architecture and Cost Optimization:

- **Blockchain for Core Voting:**
  VoteChain uses Ethereum exclusively for recording votes, ensuring that every vote is immutable and tamper-proof. This focused use of blockchain reduces on-chain load and cuts gas fees by an estimated 40–60%.
- **Off-Chain Management:**
  Non-critical data—such as metadata, user profiles, and dynamic election hierarchies (spanning local Gram Panchayat, state Vidhan Sabha, to national Lok Sabha elections)—is managed via Firebase. This separation optimizes performance during peak voting periods and minimizes operational overhead.

Enhanced Security and Accessibility:

- **Robust Authentication and Authorization:**
  Multifactor authentication (MFA) and role-based access control ensure that only eligible voters participate, effectively preventing false voting and unauthorized access.
- **Comprehensive Cryptographic Measures:**
  End-to-end encryption, cryptographic hashing, and detailed audit logs secure the voting process, maintaining transparency and integrity throughout the election cycle.
- **Remote Voting Capability:**
  VoteChain enables migrant and out-of-station voters to cast their vote, addressing geographical challenges and promoting inclusivity.

Operational Efficiency and Scalability:

- **Efficient Resource Utilization:**
  Offloading non-sensitive tasks to Firebase reduces the computational burden on the blockchain, leading to faster transaction processing and real-time result visualization in both list-based and graphical formats.
- **Scalable System Design:**
  Techniques such as transaction batching, modular architecture, and preparedness for Layer-2 scaling solutions equip the system to handle high volumes of transactions, ensuring smooth operation during peak periods.

Real-World Impact:

- **Economic Benefits:**
  By reducing reliance on expensive physical infrastructure (like EVMs, Polling Stations) and extensive manpower, VoteChain has the potential to save substantial electoral costs, thereby making the voting process more affordable and sustainable.
- **Enhanced Transparency:**
  Secure, verifiable record keeping and transparent auditing bolster public trust, paving the way for a future-proof digital democracy that aligns with modern governance needs.

In summary, VoteChain not only addresses the shortcomings of traditional and fully on-chain voting systems but also establishes a strong foundation for secure, transparent, and scalable electoral processes in the digital era.

6.2 Future Scope:

While VoteChain demonstrates significant promise as a transformative electoral solution, further enhancements and research are needed to realize its full potential and adaptability in various contexts. Key areas for future work include:

Technical Enhancements:

- **Smart Contract Optimization:**
  Further refine the smart contract logic to minimize gas fees and reduce transaction latency. Evaluating advanced Layer-2 scaling solutions (such as Polygon or Optimism) and alternative consensus mechanisms could significantly improve throughput.
- **Enhanced Privacy Measures:**
  Investigate the implementation of privacy-preserving techniques such as zero-knowledge proofs (ZKPs) or homomorphic encryption to further protect voter anonymity without sacrificing transparency.
- **Cross-Chain Interoperability:**
  Explore the integration of VoteChain with other blockchain networks to support multi-jurisdiction or cross-border elections.

Scalability and Performance Testing:

- **Empirical Validation:**
  Conduct extensive pilot studies and stress tests simulating millions of users to validate system performance, identify edge cases, and ensure resilience under real-world conditions.
- **Data Lifecycle Management:**
  Develop sophisticated off-chain archival strategies to manage election data efficiently, preserving essential metadata while optimizing blockchain storage for future elections.

Policy, Regulation, and Integration:

- **Regulatory Frameworks:**
  Collaborate with national and regional election authorities and independent security auditors to establish standardized evaluation frameworks and regulatory guidelines that ensure legal compliance and best practices.
- **System Integration:**
  Enhance interoperability with existing election management infrastructures to facilitate broader adoption and smoother integration into current governmental systems.

User-Centric and Ecological Considerations:

- **Enhanced Accessibility:**
  Incorporate additional features such as biometric authentication, multi-language support, and voice-based interfaces to improve usability for diverse voter populations, particularly in rural areas.
- **AI-Driven Analytics:**
  Integrate artificial intelligence for real-time fraud detection and anomaly monitoring, providing deeper insights into electoral data and system performance.
- **Sustainable Practices:**
  Transition to more energy-efficient, proof-of-stake (PoS) blockchains or other green consensus mechanisms to reduce the system's ecological footprint.

VoteChain's innovative hybrid approach has been recognized in academic circles as a paradigm shift in the design of digital voting systems. The published research underscores its effective cost reduction strategies, real-time auditability, and balanced integration of blockchain with off-chain management. This contribution not only enriches the academic discourse but also provides a practical framework for implementing secure, transparent, and accessible voting systems worldwide. Future interdisciplinary collaborations and iterative user testing will be crucial in evolving this promising model into a globally scalable solution.

References

1. Akhil Shah, Nishita Sodhia, Shruti Saha, Soumi Banerjee, Madhuri Chavan, "Blockchain Enabled Online-Voting System", ITM Web of Conferences, 32, 03018, 2020.
2. Wenbin Zhang, Sheng Huang, Yuan Yuan, Yanyan Hu, Shaohua Huang, Shengjiao Cao, Anuj Chopra, "A Privacy Preserving Voting Protocol on Blockchain", Journal of Information Security, 9(1), 54-67, 2018.
3. Stephan Neumann, Oksana Kulyk, Melanie Volkamer, "A Usable Android Application Implementing Distributed Cryptography for Election Authorities", Journal of Cryptography, 7(2), 120-130, 2014.
4. Jae-Geun Song, Sung-Jun Moon, Ju-Wook Jang, "A Scalable Implementation of Anonymous Voting over Ethereum Blockchain", IEEE Access, 9, 37930-37942, 2021.
5. Yulia Bardinova, Konstantin Zhidanov, Sergey Bezzateev, Mikhail Komarov, Aleksandr Ometov, "Measurements of Mobile Blockchain Execution Impact on Smartphone Battery", Journal of Mobile Computing, 8(1), 58-67, 2019.
6. David Khoury, Elie F. Kfoury, Ali Kassem, Hamza Harb, "Decentralized Voting Platform Based on Ethereum Blockchain", International Conference on Decentralized Applications and Infrastructures (DAPPS), 65-70, 2020.
7. D. Dwijesh Kumar, D. V. Chandini, Dinesh Reddy, "Secure Electronic Voting System using Blockchain Technology", Proceedings of the 2020 Blockchain Conference, 112-119, 2020.
8. Saad Moin Khan, Aansa Arshad, Gazala Mushtaq, Aqeel Khalique, Tarek Husein, "Implementation of Decentralized Blockchain E voting", International Journal Applications, 182(20), 1-5, 2018.
9. G. Kalaiyarasi, T. Narmadha, K. Balaji, V. Naveen, "E-Voting System in Smart Phone Using Mobile Application", International Journal of Advanced Research in Computer Science, 11(4), 41 48, 2020.
10. Hussam Saeed Musa, Moez Krichen, Adem Alpaslan Altun, Meryem Ammi, "Survey on Blockchain-Based Data Storage Security for Android Mobile Applications", Journal of Blockchain Research, 5(3), 102-110, 2019.

Publications

1. Ameya Mane, Atharva Birje, Harsh Minde, Jyotiraditya Patil, Dnyaneshwar Thombre, "Online Voting System Using Blockchain: A Comprehensive Survey", International Research Journal of Modernization in Engineering Technology and Science, vol. 7, issue 3, 70300268560, March 2025.
2. Harsh Minde, Atharva Birje, Ameya Mane, Jyotiraditya Patil, Dnyaneshwar Thombre, "Online Voting System Using Blockchain: A Comprehensive Approach", International Journal of Creative Research Thoughts, vol. 13, issue 3, 317, March 2025.

IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Ref No : IJCRT/Vol 13/ Issue 3 /317

To,
Harsh Anil Minde

Subject: Publication of paper at International Journal of Creative Research Thoughts.

Dear Author,

With Greetings we are informing you that your paper has been successfully published in the International Journal of Creative Research Thoughts - IJCRT (ISSN: 2320-2882). Thank you very much for your patience and cooperation during the submission of paper to final publication Process. It gives me immense pleasure to send the certificate of publication in our Journal. Following are the details regarding the published paper.

About IJCRT	:	Scholarly open access journals, Peer-reviewed, and Refereed Journals, Impact factor 7.97 (Calculate by google scholar and Semantic Scholar | AI-Powered Research Tool) , Multidisciplinary, Monthly, Indexing in all major database & Metadata, Citation Generator, Digital Object Identifier(DOI) | UGC Approved Journal No: 49023 (18)
Registration ID	:	IJCRT_280830
Paper ID	:	IJCRT25A3317
Title of Paper	:	Online Voting System Using Blockchain: A Comprehensive Approach
Impact Factor	:	7.97 (Calculate by Google Scholar) | License by Creative Common 3.0
Publication Date	:	30-March-2025
DOI	:	
Published in	:	Volume 13 | Issue 3 | March 2025
Page No	:	I481-I491
Published URL	:	http://www.ijcrt.org/viewfull.php?&p_id=IJCRT25A3317
Authors	:	Harsh Anil Minde, Atharva Sudhir Birje, Ameya Avinash Mane, Jyotiraditya Anil Patil, Dnyaneshwar Vitthalrao Thombre
Notification	:	UGC Approved Journal No: 49023 (18)

Thank you very much for publishing your article in IJCRT.

Editor In Chief
International Journal of Creative Research Thoughts - IJCRT
(ISSN: 2320-2882)