

BALLOT Secure E-Voting System

Enhancing electoral transparency and security.

Presented By :-

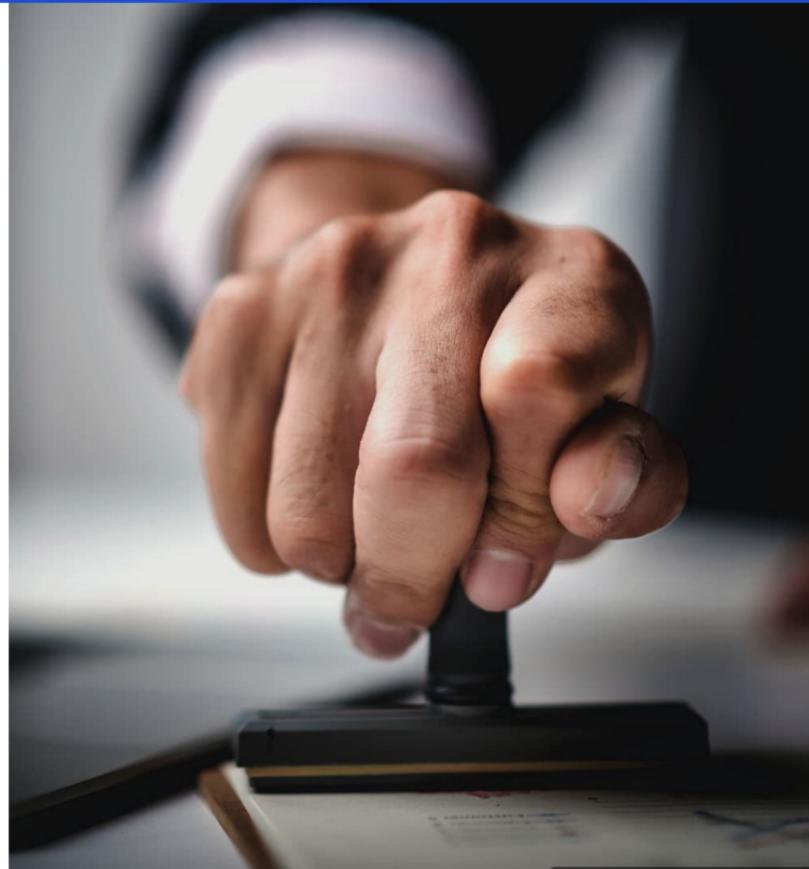
NIKHIL NAIKADE (B400440161)

SHIVRAJ BABAR(B400440110)

ANIRUDDHA KHONDE(B400440152)

YASH DESHMUKH (B400440204)

Under The Guidance of
PROF.TRUPTI RAJPUT





Developed with leading technologies

Utilizes **Next.js**, **Node.js**, and **MySQL** for robust performance and scalability.



Ensures secure, transparent voting

Guarantees **voter confidentiality** and **transparent processes** to uphold democratic integrity.



Prevents double voting

Incorporates mechanisms to detect and **prevent duplicate votes**, ensuring each vote counts only once.



Real-time results monitoring

Provides **instant access to results** and allows administrators to control the voting process efficiently.

Secure E-Voting System Overview

An advanced voting platform ensuring integrity and security

Challenges in Manual Voting

Addressing issues in traditional voting processes for improved e-voting systems



Error-prone manual voting

The traditional manual voting process is susceptible to mistakes, leading to incorrect results and undermining the integrity of elections.



Time and resource intensive

Manual voting requires significant time and resources for setup, execution, and counting, which can delay results and increase costs.



Transparency and trust issues

A lack of transparency in manual voting can erode public trust, resulting in skepticism about election outcomes and democratic legitimacy.



Authentication and confidentiality needs

Effective voting systems must ensure robust voter authentication and maintain the confidentiality of votes to protect voter privacy.

Key Objectives of E-Voting System

Ensuring secure, accessible, and efficient voting experiences for all users.



Remote voting access

Facilitates participation from any location, enabling voters to cast their votes conveniently and securely.



End-to-end encryption

Ensures that all data transmitted during the voting process remains confidential and tamper-proof, safeguarding voter privacy.



Admin-managed elections

Allows administrators to oversee the electoral process, ensuring compliance with regulations and smooth operation.



Reduce Voting Time

- Speed up the voting and counting process



One vote per user

Guarantees the integrity of the election by ensuring that each participant can only vote once, preventing fraud.



Scalable architecture

Modular & Cloud-Based Design – Easily handles growing voter numbers with flexible, distributed systems.

Load Balancing & Redundancy – Ensures smooth performance and no downtime, even under high traffic.

Comprehensive Technology Stack Overview

An overview of technologies used in BALLOT secure E-Voting system



Frontend framework utilized

The frontend of the BALLOT secure E-Voting system is developed using **Next.js**, which enables server-side rendering and static site generation for improved performance.



Backend architecture details

The backend is built with **Node.js** and **Express.js**, allowing for efficient handling of asynchronous requests and a robust API structure.



Database management system

For data storage, **MySQL** is employed, ensuring structured data organization and efficient querying capabilities.



Security measures implemented

To safeguard user data, the system employs **JWT** for authentication, **bcrypt** for password hashing, and ensures secure communications via **HTTPS**.



Development tools used

The development process is streamlined using tools such as **Git** for version control, **Postman** for API testing, and **Vercel** for deployment.

Overview of System Architecture

Detailed architecture of the BALLOT E-Voting System



Frontend: User interface

The **frontend** of the system serves as the user interface, allowing voters to interact with the **e-voting** platform seamlessly. It is designed to be intuitive and accessible, ensuring a positive user experience.



Backend: APIs for logic

The **backend** is responsible for the core logic and **validation** of votes. It utilizes robust APIs to ensure that all operations are processed efficiently and securely, maintaining system integrity.



Database: Encrypted storage

A secure **database** is vital for the storage of encrypted votes and user information. This ensures that all sensitive data is protected against unauthorized access and breaches.



Secure layer for authentication

To safeguard the system, a **secure authentication layer** is implemented, verifying user identities before allowing access to vote, thus enhancing overall security.

Essential Modules of E-Voting

An overview of the critical components in BALLOT system



Registration & Login

This module ensures users can **securely register** and log into the system, establishing a **trusted identity** for voters.



Admin Dashboard

An interface for administrators to manage the **overall system**, including user accounts and system settings, ensuring smooth operation.



Voter Dashboard

Designed for voters to view their **voting status**, upcoming elections, and access necessary voting functions in a user-friendly layout.



Secure Voting Panel

This module provides a **secure environment** for casting votes, ensuring confidentiality and integrity during the voting process.



Super Admin Control

Grants high-level administrators the ability to oversee all system operations, manage users, and enforce security protocols effectively.



Vote Tallying Module

Facilitates the **accurate and efficient counting** of votes post-election, providing real-time results and ensuring transparency.

Essential Security Features in E-Voting

Key security measures for secure e-voting implementation

JWT-based session tokens

Utilizing **JSON Web Tokens (JWT)** ensures secure, stateless user sessions, reducing the risk of session hijacking.



Encrypted vote storage (e.g., RSA)

Votes are securely stored using **encryption** methods like **RSA**, ensuring data confidentiality and integrity during transmission.



Role-based access (Voter/Admin)

Implementing **role-based access control** ensures that users can only perform actions pertinent to their assigned roles, enhancing security.



Secure audit logging

Comprehensive **audit logging** tracks all actions within the system, providing transparency and accountability for all operations.



One vote per user enforcement

Mechanisms are in place to ensure that each user can cast **only one vote**, preventing multiple submissions and maintaining the integrity of the election process.



E-Voting System Use Cases

Flow of interactions among Voters,
Admin, and System



Voter Interaction Sequence

The voter begins by **registering** on the e-voting system, then **logging in** to their account, and finally **casting their vote** during the election.

Admin Roles and Tasks

The admin is responsible for **creating elections**, **adding candidates** to the ballot, and **monitoring the votes** to ensure a fair process.

System Authentication Process

The system performs **user authentication** to verify identities, **encrypts** data for security, and **stores** and **counts** votes accurately.

Comprehensive Testing Strategy

A detailed overview of testing methodologies for e-voting systems



Unit Testing: Critical functions

Focus on individual software components such as registration, login, and voting processes to ensure they perform as expected.



System Testing: Full workflow validation

Conduct end-to-end testing of the entire e-voting system to ensure all functionalities operate correctly in unison from start to finish.



Integration Testing: Component interaction

Evaluate the interaction between various components, including the database, user interface, and APIs, to verify they work together seamlessly.



User Acceptance Testing: Real-world feedback

Gather feedback from actual users to validate that the e-voting system meets their needs and expectations prior to deployment.

E-Voting System Results Overview

Key components and features of the BALLOT secure e-voting system



Dashboard Overview

The **Dashboard** provides a comprehensive view of the voting process, displaying real-time statistics and user-friendly navigation for administrators and voters alike.



Candidate List Display

The **Candidate List** showcases all candidates participating in the election, including their profiles, party affiliations, and campaign information for informed voter decisions.



Voting Panel Features

The **Voting Panel** facilitates the actual voting process, ensuring a secure and intuitive interface for voters to cast their votes efficiently and accurately.



Admin Control Panel

The **Admin Control Panel** allows administrators to manage the election process, including candidate management, voter authentication, and monitoring of the voting activity.



Live Results Tracking

The **Live Results** feature provides instant updates on voting outcomes, reflecting real-time data to keep stakeholders informed throughout the voting period.

E-Voting System Deployment & Maintenance

Ensuring secure and reliable operation of e-voting services

	Deployed using Heroku/Vercel Utilizes Heroku or Vercel for efficient deployment and management of the e-voting application.		HTTPS enabled Ensures all communications are securely encrypted using HTTPS to protect user data during transactions.		Regular backups Implements a strategy for regular backups to ensure data integrity and availability in case of failures.
	Auto-scaling for load Employs auto-scaling capabilities to adjust resources dynamically based on traffic demands, enhancing performance.		Secure logs and patching Maintains secure logs and timely patching of vulnerabilities to safeguard against potential threats.		

Advantages

- Convenient Voting – Vote anytime, anywhere.
- Low Cost – Saves money on paper and manpower.
- Easy Access – Helpful for elderly, disabled, and remote users.
 - High Accuracy – Reduces human errors.
- Strong Security – Uses encryption and verification.
 - Eco-Friendly – No paper waste.
- More Transparent – Digital records improve trust



Conclusion

Secure, transparent e-voting system

- Modular, scalable architecture
- Real-time result generation
- Future-ready: MFA, blockchain support
- Enhances trust in digital democracy