

A PROJECT REPORT ON ONLINE VOTING SYSTEM

A major project report submitted in partial fulfillment of the requirements
for the award of the Degree of Bachelor of Technology

In

Computer Science and Engineering

Submitted by:

Chinmaya Jena: 2101292073

Under the supervision of

Dr. Dushmanta Kumar Padhi



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

GANDHI ENGINEERING COLLEGE, BHUBANESHWAR

Gandhi Vihar, Madanpur, Bhubaneswar-752054

2021-2025

ACKNOWLEDGEMENT

Our sincere thanks to **Dr. Dushmanta Kumar Padhi**, Professor and Head of the Department of Computer Science & Engineering, Gandhi Engineering College (GEC), Bhubaneswar, for his encouragement and valuable suggestions during the period of our Project work.

No words would suffice to express my regards and gratitude to Dr. Dushmanta Kumar, Department of Computer Science & Engineering, for his inspiring guidance and constant encouragement, immense support, and help during the project work.

Place:_____

Name : _____

Date_____

Reg Number : _____

Abstract

This is a project made for online voting. The basic idea is to make the voting system more reliable and efficient in every aspect. It comprises voter details stored and used for verification during voting, security systems, status, and many more. It can be accessed from anywhere anytime and provides the people of our country the flexibility and reliability of the system(Government). The administrator can add, edit, and delete the candidates and their respective parties and can view the vote count. While on the other hand, the voters can vote in a very user-friendly way. It has an automated vote-counting system that can count the tidal votes and display them in real time. It validates the user by storing their Voter card number, phone number, gender, and other validation details to ensure the safe login and casting of vote. It can remodeled and new functions can be assigned as per the requirements and needs.

The quality provides by the online voting system is that it provides some major features such as :

- Easy Accessubilty
- Secured and Reliable
- Time Efficient
- Centralized control system

CONTENTS

1. Chapter 1: Introduction

1.1 Introduction to Project

1.2 Project Category

1.3 Objectives

1.4 Problem Formulation

1.5 Identification / Reorganisation of Need

1.6 Existing System

1.7 Proposed System

1.8 Unique Features of the System

2. Requirement Analysis and System Specification

2.1 Feasibility Study

2.1.1 Technical Feasibility

2.1.2 Economical Feasibility

2.1.3 Operational Feasibility

2.2 Software Requirement Specification Document

2.2.1 Data Requirement

2.2.2 Functional Requirement

2.2.3 Performance Requirement

2.2.4 Dependability Requirement

2.2.5 Maintainability Requirement

2.2.6 Security Requirement

2.2.7 Look and Feel Requirement

2.3 Validation

2.4 Expected Hurdles

2.5 Software Development Life Cycle (SDLC) Model

3. Chapter 3: System Design

3.1 Design Approach

3.2 System Design Using Structured Analysis And Design Tools :

3.2.1 DFD's

3.2.2 Data Dictionary

3.2.3 Structured Chart

3.2.4 Flowchart or UML

3.3 User Interface Design

3.4 Database Design

3.5 ER Diagram

3.6 Normalization

3.7 Database Manipulation

3.8 Database Connection Controls

3.9 String Methodology

4. Chapter 4: Implementation, Testing, and Maintenance

4.1 Introduction to Languages, IDE, Tools, and Technologies Used

4.2 Coding Standards

4.3 Project Scheduling using various tools

4.4 Testing Techniques and Test Plans

5. Chapter 5: Result and Discussion

5.1 User Interface Representation

5.2 Brief Description of Various Modules of the System

5.3 Snapshots of the System with brief Details of each

5.4 Back End Representantion

5.5 Snapshots of Database Tables with Brief Descriptions

6. Chapter 6: Conclusion & Future Scope

6.1 Conclusion

6.2 Future Scope

7. Bibliography / References

CHAPTER 1

Introduction

1.1 Introduction to Project

As more advancements are made day by day then why not digitalize our central processes as well with the same security and efficiency? One of the most prioritized subjects will be voting. When it comes to voting it is the very right of a citizen of India to vote and select their respective leader who will lead them to an even more brighter future.

This existing idea has yet to be implemented due to its complexities and challenges which can be further broken into various aspects like insecurity, data violation, tampering, and many more. But if most of them are to be dealt with then we can lead to an innovation in the Indian constitution as a centralized system in digitalization will lead to more integrated and modern technologies. It focuses basically on data security and a user-friendly environment as it can connect people easily. It will be efficient as the voting can be done from anywhere anytime due to its flexibility in reach. It will be beneficial to optimize the old-fashioned way of counting and collecting votes.

The Online Voting System is designed to digitize and streamline the voting process. It aims to enhance voter accessibility, minimize administrative burdens, and ensure secure and transparent elections. This project is intended for use by government or organizational bodies conducting elections. The system is robust enough to handle high user traffic while maintaining data integrity.

Our commitment extends to personalized experiences through algorithms that analyze survey data, offering bespoke end-user experience, suggestions, and tailored content. We prioritize people satisfaction, providing responsive support and continually improving based on valuable user feedback to embrace a digitalized and secure environment.

1.2 Project Category

The project is categorized as an Internet-based Application Development initiative. Influencing the flow of technologies in India and taking a step towards digital India.

1.3 Objectives

There are a few objectives asserted in this project. These are mentioned as follows :

- To create a secure online platform where voters can cast their votes conveniently: Security remains the major concern when it comes to others so it provides a secure environment for the voters to cast their votes with their data being safe from exploitation. It provides convenience by having your personal space and environment.
- To ensure the system is resistant to unauthorized access and fraud: By using a two-factor authentication it prevents unverified bypass. It uses email verification which prevents fraudsters from gaining unwanted access.
- To facilitate real-time vote counting and result announcement: It counts the votes in real-time and shows it to the administrator who can keep monitoring the live feed of the votes being cast and announce the result as soon as voting ends which saves time and gives accurate results by relying on the computer.
- To improve voter turnout by making voting accessible from any location: There are times when some people cannot vote due to many reasons such as - geological reasons, emergencies, personal factors, etc. It counters that factor by providing accessibility anywhere anytime which can result in a rise in votes with time.
- To empower digital India: It shows that India as a developing country is ready to take its digitalization in software and technologies to further integration by implementing a centralized system for a government voting system for India.

1.4 Problem Formulation

Traditional voting methods face challenges like:

- Long queues and time-consuming processes.: We have seen how long we have to wait for to cast a vote. It takes a very much of our time. We are also surrounded by people and can have our personal space violated. It relies on the traditional process for which you have to free some of your time, and in some cases if there is an emergency that time slot is gone and you cannot cast a vote. It effects the number of votes cast.
- Geographical restrictions prevent some voters from participating: Sometimes you get allocated to places you have never been which can result in time-consuming scenarios. It is also a challenge for the people to reach the location on time and u can get lost during the

process. If the location is far away it restricts medical patients or physically disabled persons from reaching the particular location.

- Risks of tampering with paper ballots or electronic voting machines: With physical voting machines there is a high chance of tampering with the votes as people tend to get violent during these times. The papers can be altered with the process and there can be no solid proof for it. It has always been a very critical situation in every voting whether it may be lok sabha, rajya sabha or gram panchayat.

1.5 Identification / Reorganisation of Need

The study delves into the challenges arising from the absence of a centralized platform for governing authorities. In response to the transformative shifts in public preferences, Our online voting website addresses a critical need for the identification and reorganization of key elements in the increasing demand for a reliable, user-friendly online voting platform is driven by several key factors:

1. Digital Adoption:

- **Technological Advancements**: The rapid advancement of technology and the widespread availability of internet access have made it easier for people to interact with digital platforms.
- **Digital Literacy**: As more people become digitally literate, they are increasingly comfortable with online transactions, including voting.
- **Remote Work and Learning**: The global shift towards remote work and online education has normalized the use of digital tools for important tasks, including voting.

2. Transparency and Accountability:

- **Public Trust**: Concerns about the transparency and integrity of traditional voting methods have led to a growing demand for more secure and verifiable voting systems.
- **Accountability**: Online voting platforms can provide a more transparent and accountable process by allowing for easier auditing and verification of results.

3. Voter Convenience:

- **Accessibility:** Online voting can make voting more accessible to people with disabilities or those who live in remote areas.
- **Reduced Barriers:** By eliminating the need for physical polling stations, online voting can reduce barriers to participation and increase voter turnout.
- **Time Efficiency:** Online voting can save time for voters and election officials.

4. Security and Privacy:

- **Secure Infrastructure:** Advanced security measures, such as encryption and biometric authentication, can be implemented to protect the integrity of the voting process.
- **Privacy Protection:** Robust privacy measures can be put in place to safeguard voter information and ensure anonymity.

1.6 Existing System

Existing systems rely heavily on manual processes or electronic voting machines, which are often location-dependent, less accessible, and vulnerable to manipulation.

Manual Processes:

- **Time-Consuming:** Manual processes are time-consuming, particularly during vote counting and tabulation.
- **Error-Prone:** Human error can lead to inaccuracies in vote counting and tabulation.
- **Limited Accessibility:** Manual processes may not be accessible to all voters, especially those with disabilities or those living in remote areas.

Electronic Voting Machines (EVMs):

- **Location Dependency:** EVMs are often tied to specific polling stations, limiting voter accessibility.
- **Vulnerability to Malfunction:** Technical glitches or malfunctions can disrupt the voting process.

- **Concerns about Security and Tampering:** There are concerns about the security of EVMs and the potential for manipulation or hacking.
- **Lack of Transparency:** The voting process may not be transparent, as it is difficult to verify the integrity of the vote count.

1.7 Proposed System

The proposed Online Voting System leverages advanced security technologies to ensure the integrity and transparency of the electoral process. By incorporating these cutting-edge technologies and security measures, the proposed online voting system aims to provide a more secure, efficient, and accessible voting experience for all citizens.

Key Security Features:

- **Encryption:** Sensitive voter information and ballot data are encrypted to prevent unauthorized access and tampering.
- **Authentication Protocols:** Robust authentication protocols, such as multi-factor authentication, are implemented to verify the identity of voters and prevent unauthorized access to the system.
- **Centralized Database:** A centralized database securely stores voter information, ballot data, and election results. This database is protected by advanced security measures, including firewalls, intrusion detection systems, and regular security audits.

Benefits of the Online Voting System:

- **Enhanced Accessibility:** Voters can cast their ballots from anywhere with an internet connection, increasing accessibility for people with disabilities or those living in remote areas.
- **Increased Voter Turnout:** The convenience of online voting can encourage higher voter turnout.
- **Reduced Costs:** Online voting can reduce the costs associated with traditional voting methods, such as printing ballots and operating polling stations.
- **Improved Transparency:** The centralized database allows for real-time monitoring of the voting process, increasing transparency and accountability.

- **Reduced Fraud and Manipulation:** Advanced security measures can help to prevent fraud and manipulation, ensuring the integrity of the electoral process.

1.8 Unique Features of The System

The proposed online voting system incorporates several key features to ensure security, accessibility, and transparency:

Secure Environment:

- **Secure Communication:** All communication between the voter's device and the voting server is encrypted, preventing unauthorized interception and tampering.
- **Data Confidentiality:** Voter information and ballot choices remain confidential throughout the entire voting process.

Multi-Factor Authentication:

- **Enhanced Security:** Multi-factor authentication adds an extra layer of security by requiring voters to provide multiple forms of identification.
- **Reduced Fraud Risk:** By combining multiple authentication factors, the system can effectively mitigate the risk of unauthorized access and identity theft.
- **Personalized Experience:** Multi-factor authentication can provide a personalized voting experience, tailored to the individual voter's preferences.

Real-Time Result Updates:

- **Transparent Process:** Real-time updates on vote counts and trends increase transparency and build trust in the electoral process.
- **Informed Decision-Making:** Voters can stay informed about the election results as they unfold, potentially influencing their voting decisions.
- **Timely Results:** Real-time updates can expedite the announcement of election results, reducing uncertainty and potential disputes.

Compatibility with Desktops and Mobile Devices:

- **Accessibility:** The system is designed to be accessible to voters using various devices, including desktops, laptops, tablets, and smartphones.

- **User-Friendly Interface:** The voting interface is optimized for different screen sizes and operating systems, ensuring a smooth and intuitive user experience.
- **Offline Voting Capability:** For regions with limited internet connectivity, the system can provide offline voting options, allowing voters to cast their ballots and submit them later when an internet connection is available.

CHAPTER 2

Requirement Analysis and System Specification

2.1 Feasibility Study

The feasibility study for our online voting machine venture demonstrates robust Technical, Economic, and Operational viability. Technologically, our platform integrates cutting-edge solutions. Economically, a thorough cost-benefit analysis ensures financial viability. Operationally, streamlined processes and resource optimization underscore sustainable and efficient business operations, which we look further in more detail.

2.1.1 Technical Feasibility

The proposed online voting system is technically feasible, leveraging a combination of well-established and widely-used technologies:

- PHP: A versatile scripting language suitable for web development and database integration.
- MySQL: A robust and efficient database system for storing and managing voter information, ballot data, and election results.
- HTML/CSS: Standard technologies for creating user-friendly web interfaces.
- Java: A dynamic high-level language for enhancing user interaction and providing real-time feedback.

The system's web-based architecture ensures cross-platform compatibility, allowing voters to access the system from various devices, including desktops, laptops, tablets, and smartphones.

2.1.2 Economic Feasibility

The development and deployment of the online voting system can be cost-effective due to the following factors:

- Open-Source Technologies: By utilizing free and open-source software tools, such as PHP, MySQL, and various Java libraries, the development costs can be significantly reduced.
- Reduced Administrative Costs: The system can automate many manual tasks, such as voter registration, ballot distribution, and vote counting, leading to long-term cost savings.

- Scalability: The system can be easily scaled to accommodate a growing number of users and elections.

2.1.3 Operational Feasibility

The system is designed to be user-friendly and intuitive, requiring minimal training for end-users. The following factors contribute to its operational feasibility:

- User-Friendly Interface: A clear and intuitive user interface simplifies the voting process for voters of all ages and technical abilities.
- Secure and Reliable Infrastructure: The system is built on a robust and scalable infrastructure to ensure reliable performance, even under heavy traffic conditions.
- Robust Security Measures: Advanced security measures, such as encryption, authentication protocols, and firewalls, are implemented to protect the system from cyberattacks and unauthorized access.
- Regular Maintenance and Updates: Regular maintenance and updates will be performed to ensure the system's continued reliability and security.

2.2 Software Requirement Specification

2.3.1 Data Requirements

The online voting system requires the following data:

- **Voter Details:**
 - Name
 - Unique Identifier (e.g., voter ID, Aadhaar number)
 - Password or biometric/OTP data for authentication
- **Candidate Details:**
 - Name
 - Constituency
 - Party Affiliation
- **Vote Data:**
 - Timestamp of the vote
 - Voter's Unique Identifier

- Selected Candidate
- **Admin Data:**
 - Admin user name and password
 - Authentication method

2.3.2 Functional Requirements

The system must provide the following functionalities:

- **Voter Registration:** A user-friendly registration process allowing eligible voters to create accounts.
- **Voter Login:** Secure login mechanisms using passwords, biometrics, or OTPs.
- **Secure Voting Process:** A secure voting process that ensures the confidentiality and integrity of votes.
- **Real-Time Tallying:** Real-time updates on vote counts to provide transparent and timely results.

2.3.3 Performance Requirements

The system must meet the following performance requirements:

- **Response Time:** The system should respond to user requests within 3 seconds or less to ensure a smooth user experience.
- **Scalability:** The system should be able to handle a large number of concurrent users during peak voting times.
- **Load Balancing:** Implement load balancing techniques to distribute the workload across multiple servers.

2.3.4 Dependability Requirements

The system must be highly reliable and available during election periods:

- **24/7 Availability:** The system should be accessible 24/7 during the voting period.
- **Automated Failover:** Implement automated failover mechanisms for database servers to ensure continuous operation in case of hardware or software failures.
- **Regular Maintenance:** Regular maintenance and updates are essential to address security vulnerabilities and improve performance.

2.3.5 Maintainability Requirements

The system should be easy to maintain and update:

- **Modular Design:** The system should be designed in a modular way, making it easier to modify and update specific components without affecting the entire system.
- **Regular Security Patches:** Regular security updates should be applied to address vulnerabilities and protect the system from attacks.

2.3.6 Security Requirements

The system must implement robust security measures to protect voter data and the integrity of the election process:

- **HTTPS Protocol:** All communication between the client and server should be encrypted using HTTPS to ensure secure data transmission.
- **Input Validation:** Input validation should be implemented to prevent malicious input, such as SQL injection and cross-site scripting (XSS) attacks.
- **DDoS Protection:** Implement measures to mitigate Distributed Denial of Service (DDoS) attacks, such as rate limiting and intrusion detection systems.

2.3.7 Look and Feel Requirements

The system should have a user-friendly and visually appealing interface:

- **Clean and Minimalist Design:** A clean and minimalist design can enhance user experience and reduce cognitive load.
- **Responsive Design:** The system should be responsive and adaptable to different screen sizes, ensuring a seamless user experience on desktops, tablets, and mobile devices.
- **Accessibility:** The system should be accessible to users with disabilities, adhering to accessibility standards like WCAG

2.4 Validation

To ensure the quality and reliability of the online voting system, a comprehensive testing strategy will be implemented, involving the following methodologies:

Unit Testing

Unit testing focuses on testing individual software components or modules in isolation. This helps to identify and fix bugs early in the development process.

Key benefits of unit testing:

- **Early defect detection:** By testing individual components, defects can be identified and fixed before they propagate to higher levels of integration.
- **Improved code quality:** Unit tests help to ensure that code is well-structured, maintainable, and adheres to coding standards.
- **Increased confidence:** Well-tested components contribute to a more reliable and robust system.

Integration Testing

Integration testing involves testing the interaction between different software components or modules. This ensures that components work together seamlessly and communicate effectively.

Key benefits of integration testing:

- **Identifying interface defects:** It helps uncover issues related to data exchange and communication between components.
- **Ensuring system functionality:** It verifies that the system meets functional requirements as a whole.
- **Validating system behavior:** It tests the system's behavior under various conditions and scenarios.

Stress Testing

Stress testing evaluates the system's performance under extreme load conditions. It helps to identify bottlenecks, performance issues, and system failures.

Key benefits of stress testing:

- **Identifying performance bottlenecks:** It helps pinpoint areas of the system that may not be able to handle high loads.
- **Evaluating system stability:** It assesses the system's ability to handle peak loads and unexpected surges in traffic.

- **Improving system scalability:** It provides insights into how to scale the system to accommodate future growth.

2.5 Expected Hurdles

Creating and managing an online voting website can be a rewarding venture, but it comes with its own set of challenges. Here are some expected hurdles we might encounter in this industry to ensure the security and reliability of the online voting system, several measures must be implemented:

Cybersecurity Measures

- **Strong Encryption:** Employing strong encryption algorithms to protect sensitive voter data, including personal information and voting choices.
- **Secure Authentication:** Implementing robust authentication mechanisms, such as multi-factor authentication, to verify the identity of users and prevent unauthorized access.
- **Regular Security Audits:** Conduct regular security audits and vulnerability assessments to identify and address potential security¹ weaknesses.
- **Intrusion Detection Systems:** Deploying intrusion detection systems to monitor network traffic for suspicious activity and promptly respond to threats.
- **Web Application Firewalls (WAF):** Using WAFs to filter and block malicious traffic, protecting the system from common web attacks like SQL injection and cross-site scripting (XSS).
- **Regular Software Updates:** Keeping the system's software and libraries up-to-date with the latest security patches to address vulnerabilities.
- **User Awareness and Training:** Educating users about cybersecurity best practices, such as avoiding phishing attacks and strong password usage.
 - **Server Reliability**
- **Redundancy:** Implementing redundant servers and network components to ensure high availability and minimize downtime.
- **Load Balancing:** Distributing the workload across multiple servers to prevent overloading and improve performance.
- **Disaster Recovery Planning:** Developing a comprehensive disaster recovery plan to minimize the impact of potential disasters, such as natural disasters or cyberattacks.

- **Regular Monitoring and Maintenance:** Monitoring system performance and proactively addressing any issues to maintain optimal performance.
- **Scalability:** Designing the system to handle increasing loads and future growth.

2.6 SDLC Model to be Used

The Iterative Waterfall Model is used for the development.

Phase 1: Planning

- **Identify Requirements:**
 - Define the core functionalities of the system, such as user registration, voter authentication, ballot casting, vote counting, and result publication.
 - Identify stakeholders, including voters, election officials, and system administrators.
- **Feasibility Study:**
 - **Technical Feasibility:** Assess the availability of necessary technologies, such as secure authentication protocols, encryption algorithms, and reliable database systems.
 - **Economic Feasibility:** Evaluate the cost-effectiveness of the project, including hardware, software, and personnel costs.
 - **Operational Feasibility:** Consider the operational challenges, such as system uptime, scalability, and security.

Phase 2: Requirement Analysis

- **Gather Requirements:**
 - Collect functional requirements, such as the system's specific features and capabilities.
 - Identify non-functional requirements, such as performance, security, usability, and scalability.
- **Analyze Requirements:**
 - Prioritize requirements based on their importance and feasibility.
 - Refine and clarify requirements to ensure a clear understanding.
- **Create Software Requirements Specification (SRS):**

- Document detailed requirements, including use cases, data flows, and system interactions.
- Use a structured format like the IEEE Standard for Software Requirements Specification (IEEE Std 830).

Phase 3: System Design

- **High-Level Design (HLD):**

- Define the overall system architecture, including the client-server architecture and the database schema.
- Create data flow diagrams to visualize the data flow between different system components.

- **Detailed Design (LLD):**

- Design the user interface, including wireframes and mockups.
- Develop detailed design specifications for each module, including data structures, algorithms, and interfaces.
- Design the database schema, defining entities, attributes, and relationships.
- Plan the security mechanisms, such as authentication, authorization, and encryption.

Phase 4: Development

- **Setup Development Environment:**

- Install and configure necessary software, such as a web server, database server, and development tools.
- Set up a version control system like Git to manage code changes.

- **Code Development:**

- Develop the backend components, including server-side logic for user authentication, vote casting, and result calculation.
- Create the front-end user interface using HTML, CSS, and JavaScript.
- Implement database operations to store and retrieve voter information, candidate information, and vote data.

- **Code Review:**

- Conduct code reviews to ensure code quality, adherence to coding standards, and security best practices.

Phase 5: Testing

- **Unit Testing:** Test individual modules to ensure they function correctly.
- **Integration Testing:** Test the interaction between different modules to ensure they work together seamlessly.
- **System Testing:** Test the entire system to verify that it meets all functional and non-functional requirements.
- **Security Testing:** Test the system's security measures, including vulnerability scanning and penetration testing.
- **Performance Testing:** Test the system's performance under load to identify bottlenecks and optimize performance.
- **User Acceptance Testing (UAT):** Involve end-users to test the system's usability and functionality.

Phase 6: Deployment

- **Deployment Environment Setup:**
 - Configure servers, databases, and network infrastructure.
 - Install and configure necessary software and security measures.
- **Deployment:**
 - Deploy the system to the production environment, including transferring code, databases, and configuration files.
 - Perform final testing and validation.
- **Data Migration:** If applicable, migrate data from existing systems to the new system.

Phase 7: Maintenance

- **Bug Fixes:** Monitor the system for bugs and issues, and fix them promptly.
- **Security Updates:** Apply security patches and updates to address vulnerabilities.
- **Performance Optimization:** Monitor system performance and optimize it as needed.
- **Feature Enhancements:** Add new features and functionalities based on user feedback and evolving requirements.
- **Regular Maintenance:** Perform regular maintenance tasks, such as backups, database optimization, and system updates.

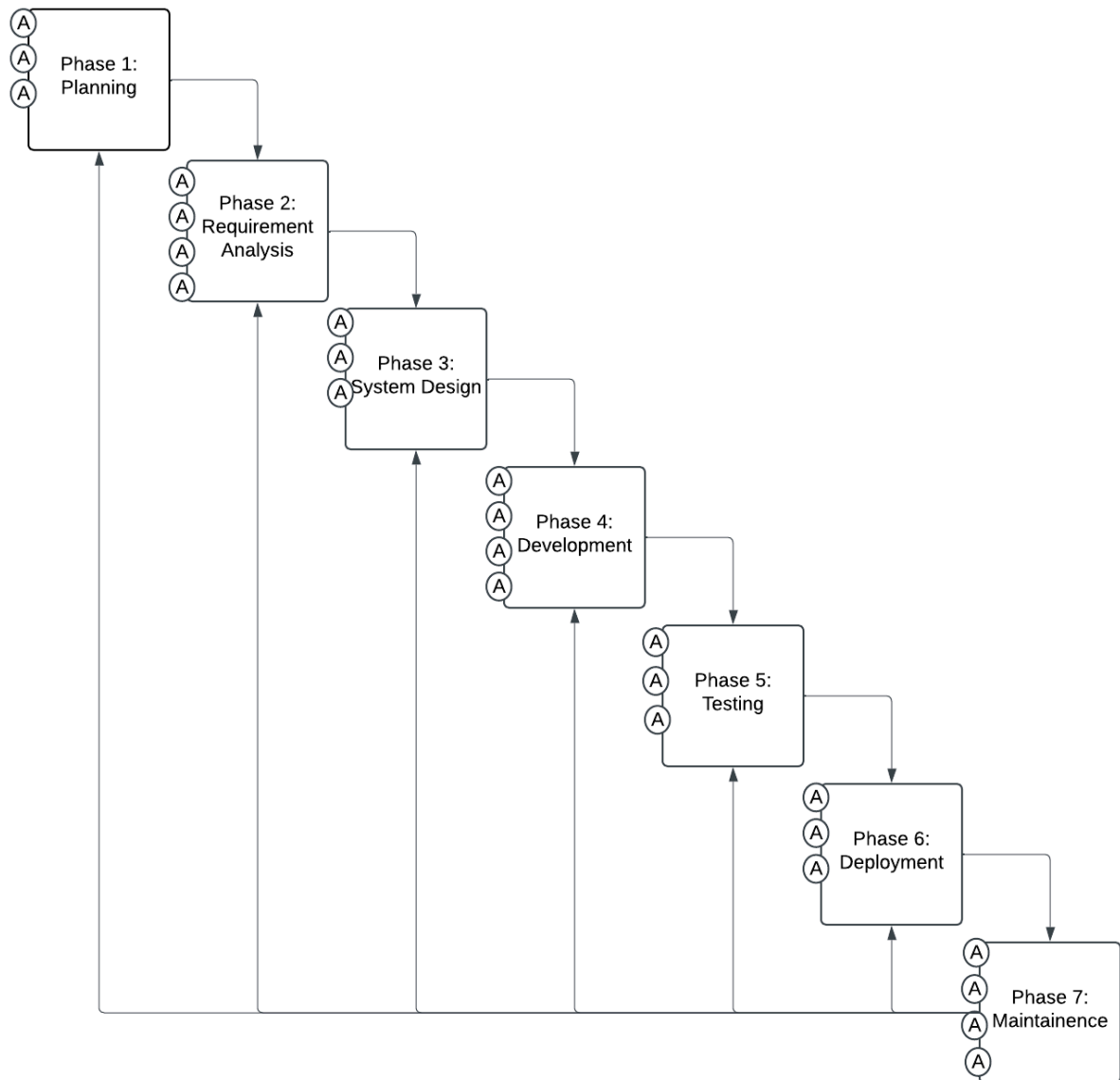


Fig2.1: SDLC Model Diagram

CHAPTER 3

System Design

3.1 Design Approach

Employing an Object-Oriented design paradigm in the development of our online voting platform is pivotal. This approach adeptly manages intricate relationships among diverse entities, fostering modularity, encapsulation, and scalability. Through encapsulating functionality within distinct components representing real-world entities (votes, candidates) the design ensures maintainable and scalable code. Leveraging inheritance and polymorphism enhances code reusability, optimizing efficiency. This methodology establishes a cohesive, intuitive system architecture, crucial for handling complexities intrinsic to online voting interactions. The Object-Oriented design proves indispensable in navigating the dynamic landscape of e-commerce, ensuring a resilient and user-friendly platform.

3.2 Detail Design

3.2.1 Data Flow Diagram(DFD's)-

A data flow diagram (DFD) is a graphical representation of the "flow" of data through an information system, modeling its process aspects. A DFD is often used as a preliminary step to create an overview of the system, which can later be elaborated.

0 Level DFD -

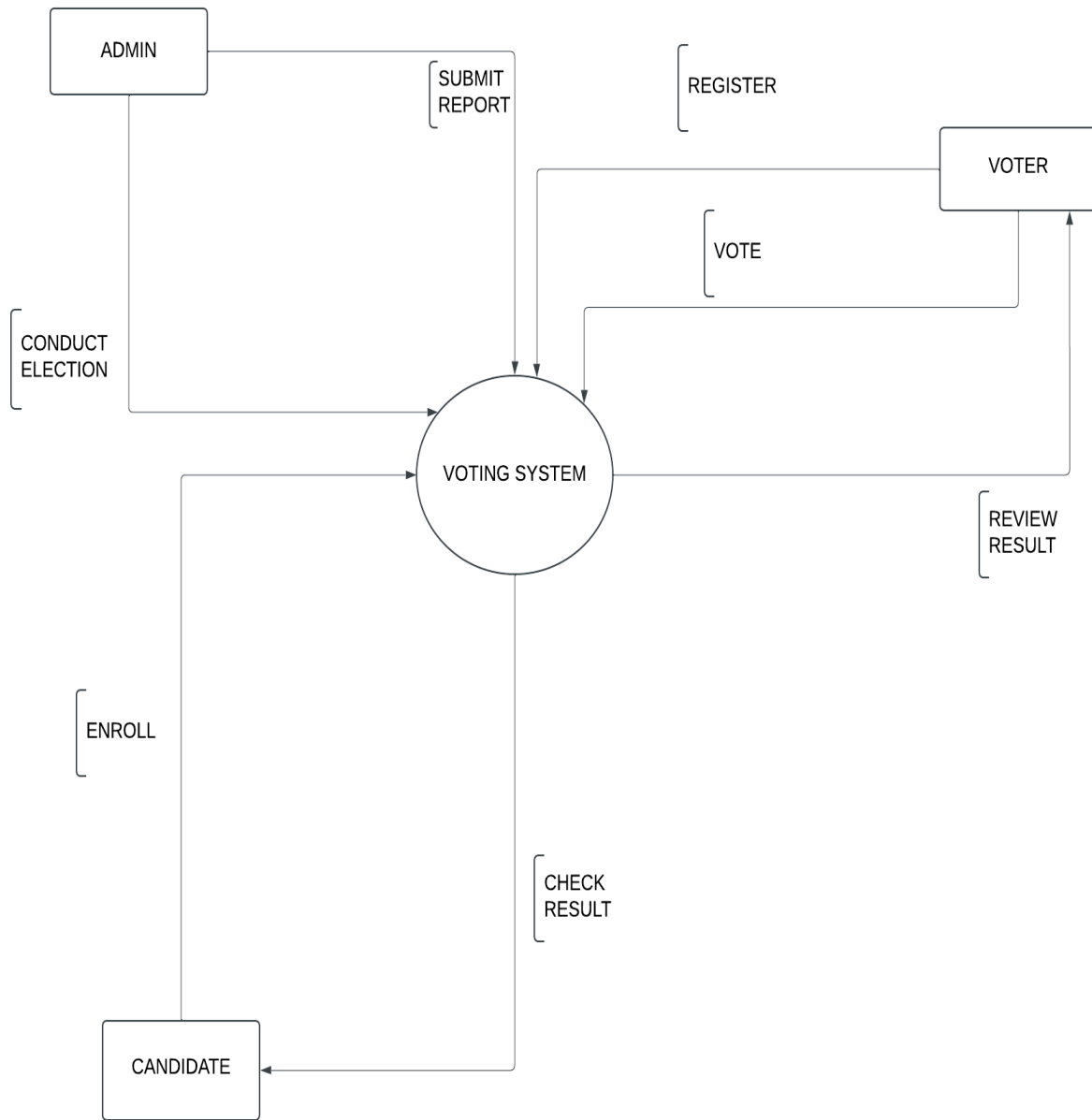


Fig 3.1: 0 Level DFD

1 Level DFD -

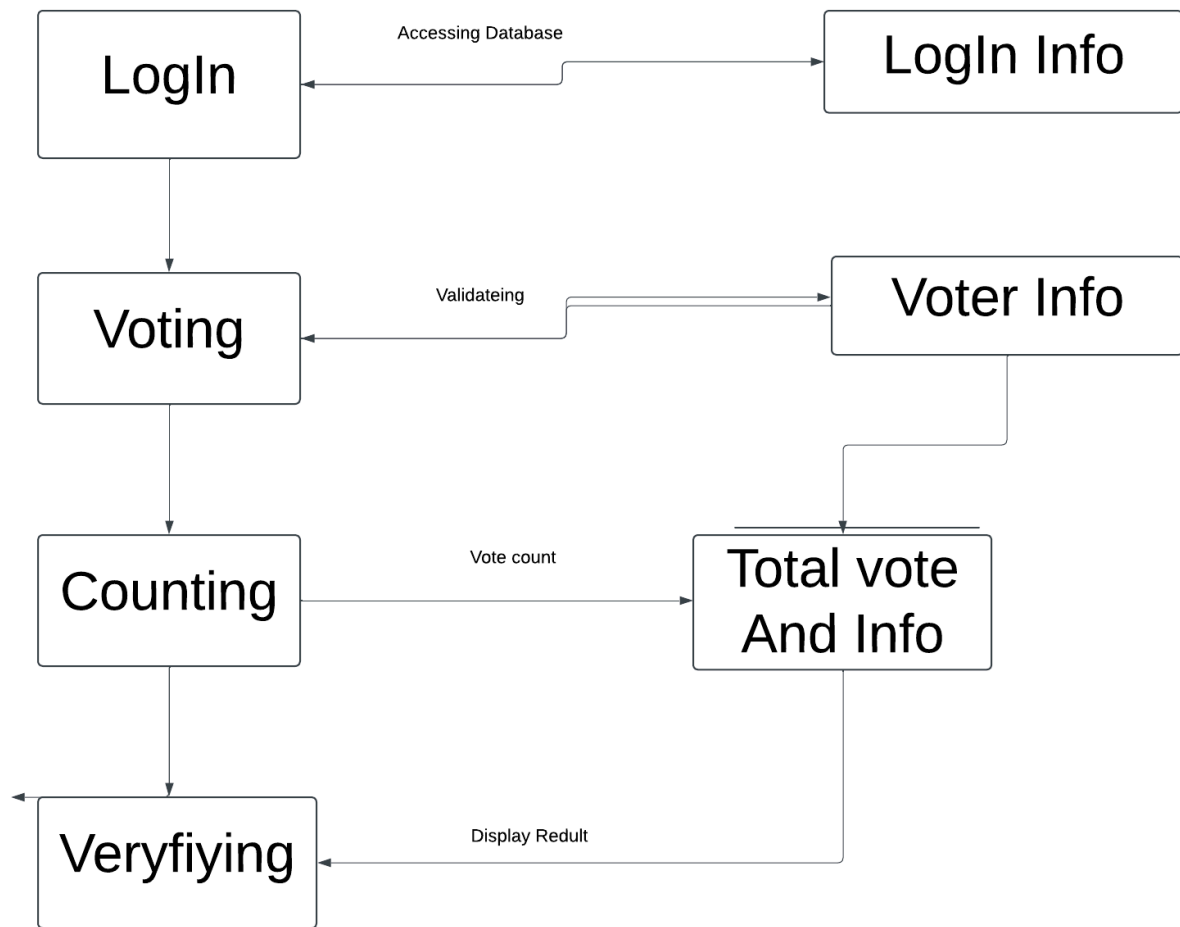


Fig 3.2 : 1 Level DFD

3.2.2 Data Dictionary-

1. Table: Admin

Column Name	Data Type	Description	Key
adminID	INT	Unique identifier for admin users.	Primary key
username	VARCHAR(50)	Unique username for admin authentication.	Unique key
password	VARCHAR(100)	Encrypted password for admin login.	-

2. Table Login

Column Name	Data Type	Description	Key
voter_card_number	VARCHAR(30)	Unique voter ID card number	Primary key
name	VARCHAR(100)	Full of the voter	-
username	VARCHAR(50)	Unique username for voter login	Foreign Key
gender	ENUM('Male', 'Female', 'Other')	Gender of the voters	-
dob	DATE	Date of birth of voter	-
email	VARCHAR(100)	Unique email ID of the voter	Unique key
password	VARCHAR(100)	Encrypted password for voter authentication.	-

3. Table Voter

Column Name	Data Type	Description	Key
voter_card_number	VARCHAR(30)	Unique voter ID card	Primary Key
voter	VARCHAR(100)	Candidate or party voted by the voter	-

4. Table Partyable

Column Name	Data Type	Description	Key
pid	INT	Unique identifier for a political party	Primary Key
partycode	VARCHAR(20)	Unique code for the political party	Primary key
partyname	VARCHAR(100)	Full name of the political party	
photo	BLOB	Party logo or representative image	

5. Table contact

Column Name	Data Type	Description	Key
contactid	INT	Unique Identifier for each message	Primary key
name	VARCHAR(100)	Name of the person sending the message	
company	VARCHAR(100)	(Optional) company name of the sender	
email	VARCHAR(100)	Email address of the sender	
message	TEXT	Content of the message	

3.2.3 Structured Charts -

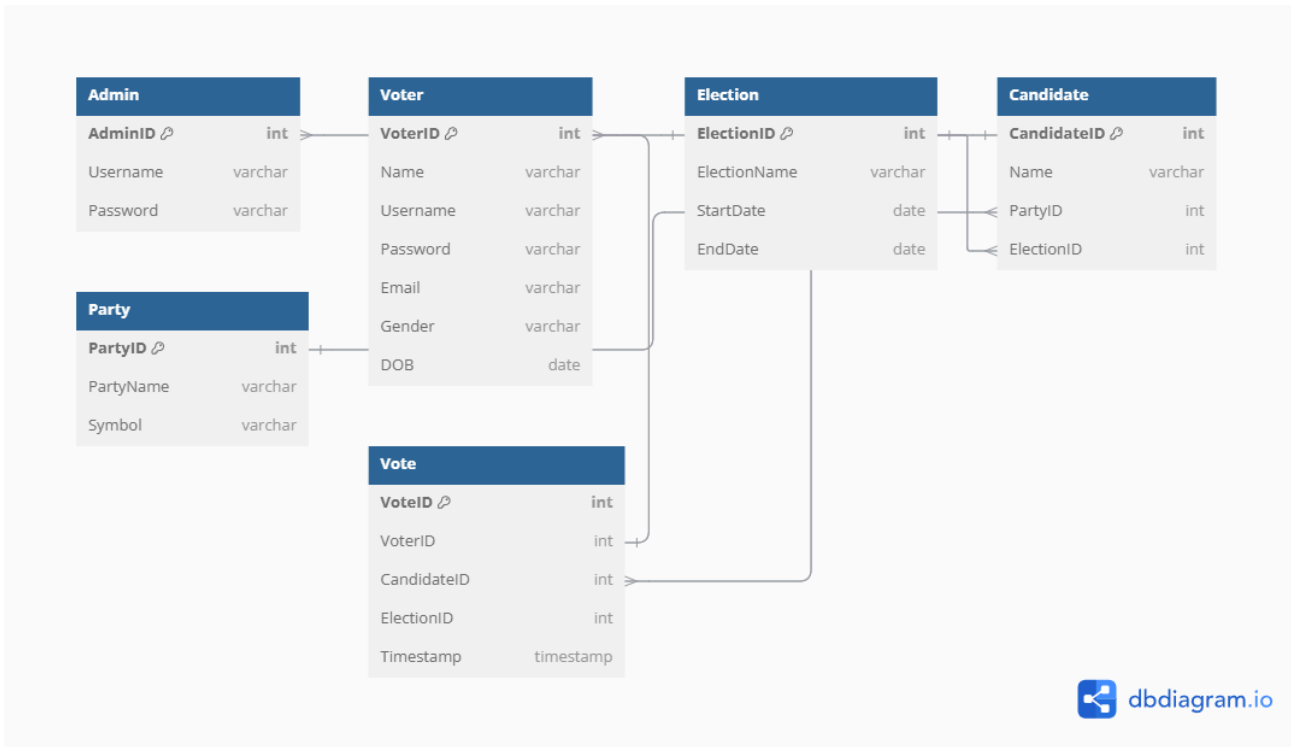


Fig 3.3: Class Diagram

3.2.4 Flowcharts or UML

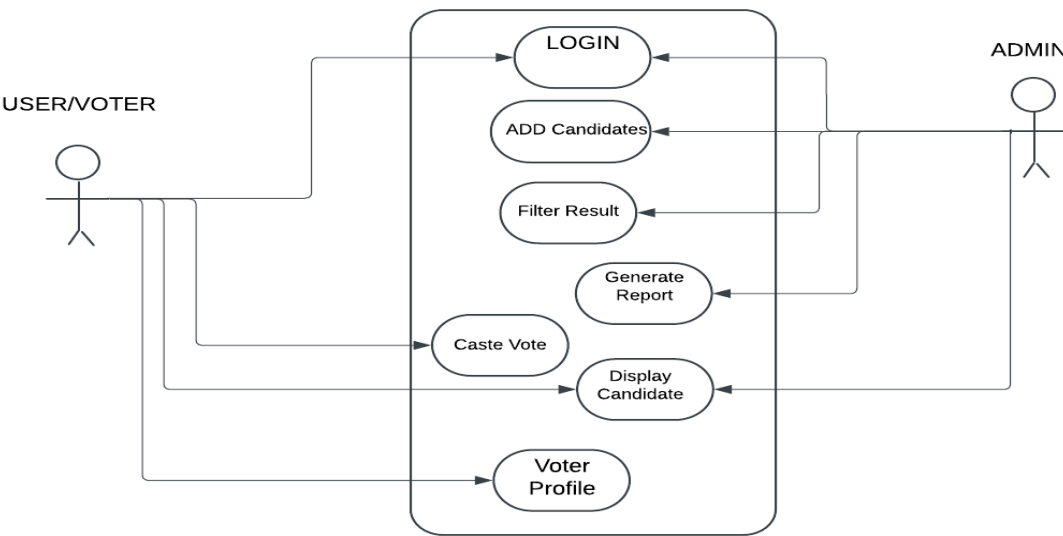


Fig 3.4: Use Case Diagram

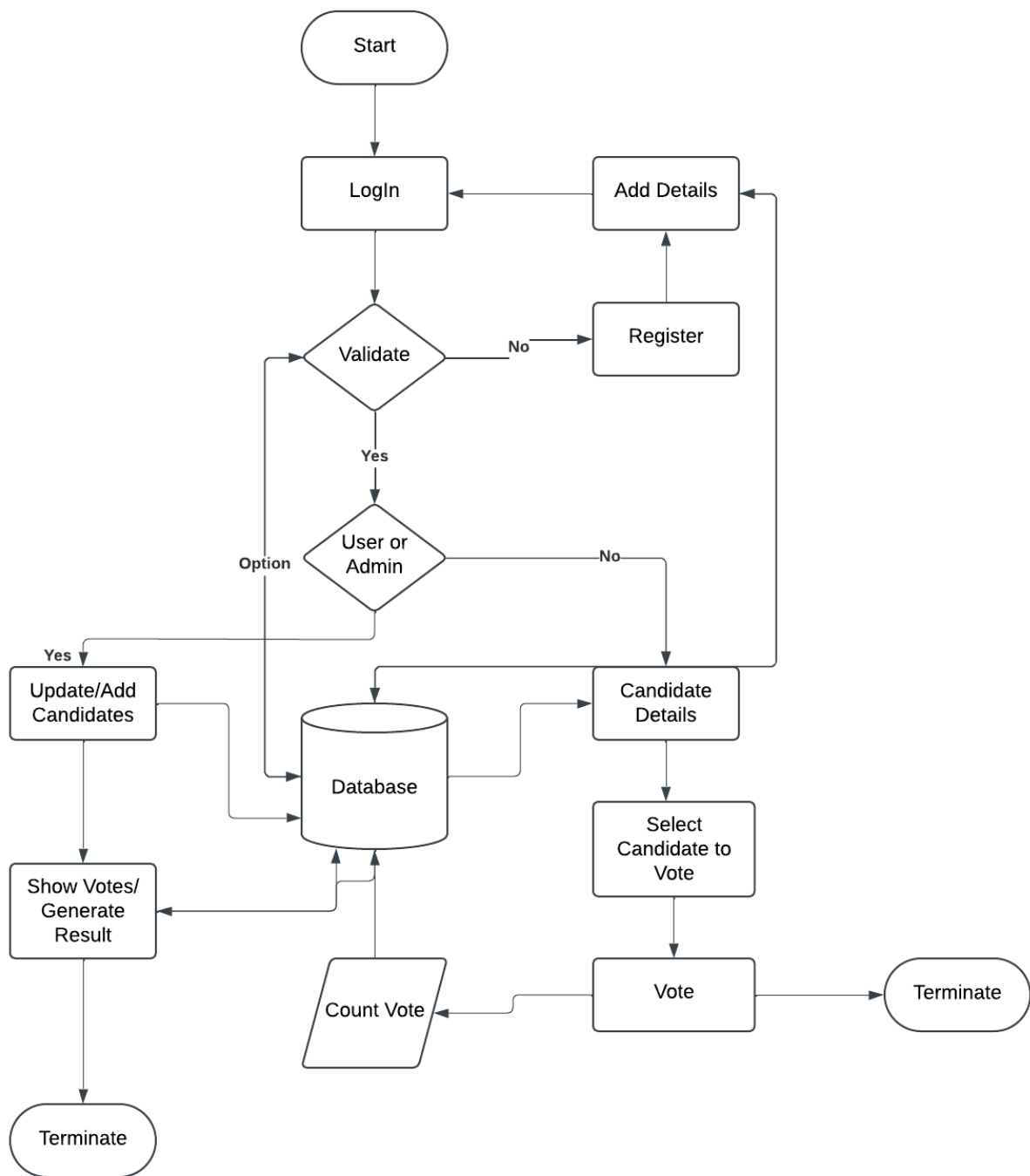


Fig 3.5: Flowchart

3.3 ER Diagram

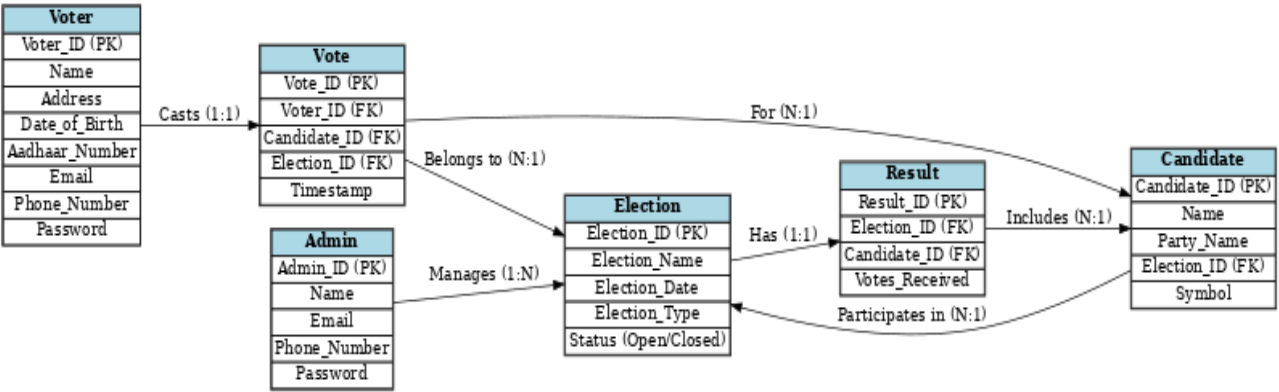


Fig 3.6: ER Database Model

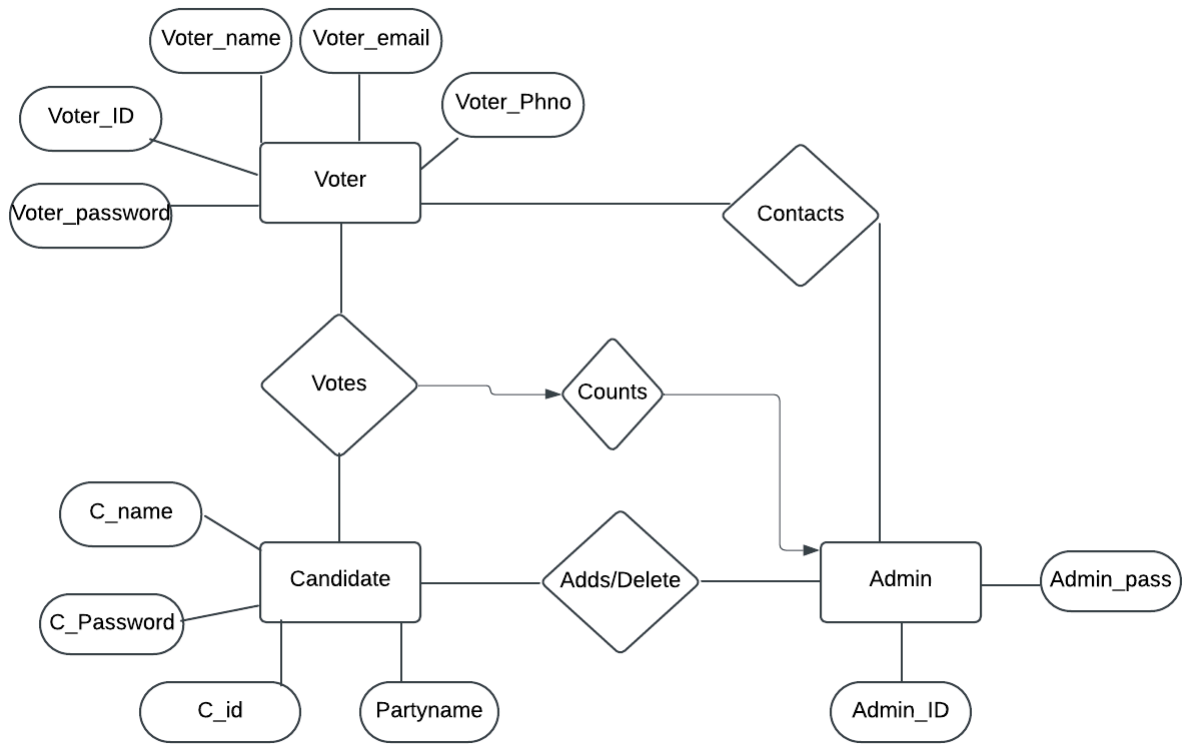


Fig 3.7: ER Diagram

CHAPTER 4

Implementation, Testing, and Maintenance

Soon To Be Completed.....

Chapter 5

Result and Discussion

Soon To Be Completed.....

Chapter 6

Conclusion & Future Scope

Conclusion

To Be Completed after CHAPTER 4 & 5.

Future Scope

Future Scopes of Online Voting System in India

1. Enhanced Accessibility for Remote Areas

Online voting can significantly improve voter participation by providing a convenient option for citizens living in remote or geographically inaccessible regions where physical polling booths are challenging to establish.

2. Increased Voter Turnout

By enabling people to vote from anywhere using secure devices, online voting can encourage greater participation, especially among urban migrants, the youth, and NRIs (Non-Resident Indians).

3. Integration with Digital India Initiative

An online voting system aligns with the government's Digital India vision, promoting technology adoption across governance frameworks and making elections more efficient and cost-effective.

4. Environmental Sustainability

Reducing reliance on paper ballots and physical polling infrastructure minimizes environmental impact, contributing to sustainable electoral practices.

5. Real-Time Voting Analytics

Online systems can provide real-time data analytics to monitor voting trends, ensuring transparency and faster decision-making without compromising voter confidentiality.

6. Blockchain Integration for Enhanced Security

The incorporation of blockchain technology can ensure the integrity and immutability of votes, addressing concerns about tampering or fraud.

7. Inclusion of Differently-Abled Voters

Online systems can offer accessible interfaces with assistive technologies, making the voting process more inclusive for people with disabilities.

8. Cost and Resource Efficiency

Implementing online voting can significantly reduce costs associated with physical polling stations, manpower, and logistics, freeing resources for other developmental activities.

9. Increased Transparency and Trust

Online systems, if developed with robust security protocols and audit mechanisms, can enhance trust in the electoral process by reducing manual errors and enabling verifiable results.

10. Global Voting for NRIs

NRIs often face challenges in participating in elections due to physical distance. An online voting system can offer them a reliable and secure way to cast their votes, strengthening democratic inclusion.

11. AI-Driven Fraud Detection

Advanced AI algorithms can detect unusual voting patterns or suspicious activities in real time, helping to maintain the integrity of elections.

12. Piloting New Democratic Practices

Online systems could facilitate experiments with alternative democratic processes like ranked-choice voting or participatory budgeting, paving the way for innovative governance models.

13. Emergency Voting Mechanism

In scenarios like pandemics, natural disasters, or other emergencies, online voting ensures continuity in the democratic process without endangering public health or safety.

14. Interoperability with Aadhaar and Digital IDs

Integration with Aadhaar or other digital identity platforms can enhance authentication, ensuring only eligible voters participate while maintaining privacy.

15. Public-Private Partnerships for Innovation

Collaborations between government agencies and technology companies can foster innovation, developing scalable and secure online voting platforms tailored for India's diverse population.

Chapter 7

References

Reference For Codes -

- Github
- GeeksforGeeks
- Git
- ChatGPT
- Typescriptlang
- Devdocs

Reference For Diagrams -

- Lucidchart
- Ddraw.io
- DbDiagram.io
- ChatGPT
- Geeksforgeeks