

**INTEGRATED PROJECT REPORT**  
On  
**जनमत**

Submitted in partial fulfillment of the requirement for the  
Course Integrated Project (CS 203) of

**COMPUTER SCIENCE AND ENGINEERING**  
**B.E. Batch-2020**  
**in**

**JUNE-2023**



**Under the Guidance of**  
**Dr. Parul Goyal**

**Submitted By**

**Sampada Sharma**  
**Roll. No. 2010992077**

**Sanchit Arora**  
**Roll. No. 2010992081**

**Shreshtha Puri**  
**Roll. No. 2010992096**

**Shruti**  
**Roll. No. 2010992097**

**CHITKARA UNIVERSITY**

## **(Annexure –C)**

### **CERTIFICATE**

This is to be certified that the project entitled “**जनमत**” has been submitted for the Bachelor of Computer Science Engineering at Chitkara University, Punjab during the academic semester January 2023- May-2023 is a bona fide piece of project work carried out by “Samapada sharma , Sanchit arora , Shreshtha Puri , Shruti Bansal roll no.: 2010992077, 2010992081,2010992096, 2010992097 respectively ” towards the partial fulfillment for the award of the course Integrated Project (CS 203) under the guidance of “Dr. Parul ” and supervision.

**Sign. of Project Guide**  
:  
**Dr. PARUL GOYAL**

Assistant Professor

(Department of Computer Science and Engineering)

## **(Annexure –D)**

### **CANDIDATE'S DECLARATION**

We, Sampada sharma , Sanchit arora , Shreshtha Puri , Shruti Bansal roll no.: 2010992077, 2010992081,2010992096, 2010992097 respectively, B.E.-2020 of the Chitkara University, Punjab hereby declare that the Integrated Project Report entitled “जनमत” is an original work and data provided in the study is authentic to the best of our knowledge. This report has not been submitted to any other Institute for the award of any other course.

<b>Sign. of Student 1</b>	<b>Sign. of Student 2</b>	<b>Sign. of Student 3</b>	<b>Sign. of Student 4</b>
Sampada Sharma	Sanchit Arora	Shreshtha Puri	Shruti
ID No : 2010992077.	ID No: 2010992081	ID No: 2010992096	ID No: 2010992097

## **(Annexure -E)**

### **ACKNOWLEDGEMENT**

It is our pleasure to be indebted to various people, who directly or indirectly contributed in the development of this work and who influenced my thinking, behavior and acts during the course of study.

We express our sincere gratitude to all for providing me an opportunity to undergo Integrated Project as the part of the curriculum.

We are thankful to “Mrs. Parul ” for his support, cooperation, and motivation provided to us during the training for constant inspiration, presence and blessings.

We also extend our sincere appreciation to “**Dr. Parul Goyal**” who provided his valuable suggestions and precious time in accomplishing our Integrated project report.

Lastly, We would like to thank the almighty and our parents for their moral support and friends with whom we shared our day-to-day experience and received lots of suggestions that improve our quality of work.

Sampada Sharma

ID No : 2010992077.

Sanchit Arora

ID No: 2010992081

Shreshtha Puri

ID No: 2010992096

Shruti

ID No: 2010992097

## ABSTRACT AND KEYWORDS

This project seeks to tackle the challenges that arise during elections in India, with the overarching goal of improving the fairness and integrity of the electoral process. The project acknowledges the presence of issues such as insecurity at polling stations, corruption leading to vote-selling, and accessibility constraints, all of which undermine the democratic principles and disenfranchise citizens.

To address these challenges, the project proposes a comprehensive set of solutions aimed at different aspects of the electoral process. These solutions are designed to enhance security, combat corruption, improve accessibility, promote voter education and awareness, and foster transparent electoral processes. By implementing these measures, the project aims to strengthen the democratic fabric of the nation and ensure a more inclusive and representative electoral system.

Elaboration:

**1. Enhanced Security Protocols:**

Recognizing the issue of insecurity at polling stations, the project suggests the implementation of stringent security protocols. This includes deploying adequate security personnel, utilizing advanced surveillance systems, and employing biometric authentication methods to ensure a safe and secure environment for voters. By enhancing security measures, the project aims to eliminate instances of polling place hijacking and protect the integrity of the voting process.

**2. Anti-Corruption Initiatives:**

To address corruption and the practice of vote-selling, the project advocates for comprehensive anti-corruption initiatives. These initiatives involve raising awareness about the detrimental effects of vote-selling through voter education campaigns. Additionally, the project aims to strengthen anti-corruption agencies, empowering them to investigate and prosecute electoral malpractices effectively. By addressing corruption, the project aims to restore the authenticity and credibility of the electoral process.

**3. Improved Accessibility:**

The project acknowledges the challenges faced by citizens in reaching polling stations, particularly in remote or rural areas. To overcome these accessibility constraints, the project proposes alternative voting methods. This can involve setting up mobile voting units or temporary polling booths in remote areas, facilitating postal voting for those unable to travel, or exploring the feasibility of online voting where appropriate. By enhancing accessibility, the project aims to ensure that all citizens have an equal opportunity to exercise their voting rights.

**4. Voter Education and Awareness Campaigns:**

Recognizing the importance of voter education, the project advocates for comprehensive awareness campaigns. These campaigns would inform citizens about their rights, responsibilities, and the significance of their vote. Additionally, they would provide unbiased information about the electoral process, candidates, and parties. By empowering voters with knowledge, the project aims to promote informed decision-making and encourage active participation in the democratic process.

**5. Transparent Electoral Processes:**

To instill confidence in the electoral system, the project emphasizes the need for transparent processes. This involves ensuring transparency during the counting of votes and regulating campaign financing to prevent manipulation. Furthermore, the project suggests engaging independent election observers to monitor and report on the fairness of elections, thus fostering accountability and impartiality. By promoting transparency, the project aims to uphold the principles of fairness and integrity in the electoral process.

In conclusion, this project proposes a holistic approach to address the challenges faced during elections in India. By implementing measures to enhance security, combat corruption, improve accessibility, promote voter education, and foster transparency, the project aims to strengthen the electoral system and ensure a more inclusive and representative democracy in the country.

Keywords:

Elections, electoral process, fairness, integrity, insecurity, polling stations, corruption, vote-selling, accessibility, security protocols, anti-corruption initiatives, alternative voting methods, voter education, awareness campaigns, transparent processes, democratic principles, representative system, India.

## **Table of Contents:**

1. Introduction
1.1 Background
1.2 Problem Statement
2. Literature Review
3. Scope
4. Software and Hardware Requirement Specification
4.1 Method
4.2 Programming/Working Environment
4.3 Requirements to Run the Application
5. System Design
5.1 Database Analysis and Design
6. Program's Structure Analysis
6.1 GUI Construction
7. Code Implementation
7.1 Database Connection
7.2 Code Implementation
8. System Testing
8.1 Testing Strategy
8.2 Test Cases
8.3 Error Handling
8.4 Performance Evaluation
9. Limitations
10. Future Scope
11. Conclusion
12. References

# CHAPTER 1

## INTRODUCTION

### 1.1 Background

India is a vibrant democracy with people electing their representatives at several levels. People in India elect their representatives with the help of the voting system.

Voting may be a method for a gaggle, like a gathering or an electorate, so as to form a collective decision or express opinion usually following discussions, debates or campaigns.

Voting often takes place at a polling station. But is the practice of selecting a particular candidate or candidates or parties being conducted fairly?

No, is the response to this query.

The Indian constitution gave its citizens the freedom to choose their own government through political channels.

However, going to the polls is still not regarded as a secure option in many places these days because of the hijacking of polling places or the corruption that compels people/voters to switch their choice of representatives. When elections are conducted, many people find it difficult to travel, and some are compelled to sell their ballots in exchange for cash. Don't these difficulties with voting require a solution?

Therefore, "जनमत" is the solution to some extent for the problems faced by the people during elections. Enhancing

### Electoral Processes in India: Addressing Challenges and Introducing "जनमत"

India, known for its vibrant democracy, values the participation and voices of its citizens in the electoral process. The selection of representatives through the voting system is a fundamental aspect of democratic governance. However, the practice of selecting candidates or parties is not always conducted fairly, posing significant challenges to the integrity and inclusiveness of the electoral process.

India, with its rich history and diverse population, is widely recognized as the world's largest democracy. The country's political landscape is characterized by multiple levels of governance and a robust system of electing representatives through the voting system. The Indian constitution grants its citizens the freedom to choose their own government through political channels. However, despite these democratic principles, the electoral process in India faces several challenges that undermine its fairness and integrity.

This project aims to address the difficulties encountered during elections in India and propose solutions to enhance the fairness and integrity of the electoral system. These challenges include insecurity at polling stations, corruption leading to vote-selling, and accessibility constraints that impede citizens' ability to exercise their voting rights.

In recent times, concerns have emerged regarding the security of polling places, with instances of hijacking that compromise the trust and fairness of elections. Furthermore, corruption has undermined the democratic principles as voters are coerced or influenced to sell their ballots in exchange for monetary gains. Additionally, citizens face obstacles in traveling to polling stations, particularly those residing in remote areas, resulting in reduced voter turnout and disenfranchisement.



## 1.2 Problem Statement:

The practice of selecting candidates or parties through the voting system in India is marred by various issues that impede the democratic ideals enshrined in the constitution. This project aims to address the following challenges:

### 1. Insecurity at Polling Stations:

Instances of insecurity at polling stations raise concerns about the fairness and integrity of the electoral process. The hijacking of polling places undermines the trust citizens place in the system and creates an environment where voters may feel coerced or influenced to change their choice of representatives.

### 2. Corruption and Vote-Selling:

The prevalence of corruption in certain regions of India has led to a disheartening practice of vote-selling. Citizens are coerced or enticed to sell their ballots in exchange for monetary gains, compromising the fundamental principles of democracy and eroding the power of individual choices.

### 3. Accessibility Constraints:

Many citizens face difficulties in participating in the electoral process due to accessibility constraints. This is particularly pronounced in remote or rural areas, where the lack of adequate infrastructure and long distances to polling stations pose significant challenges. As a result, voter turnout is affected, and some citizens are effectively disenfranchised.

These challenges highlight the need to address the existing shortcomings in the electoral process and ensure a fair, inclusive, and representative democracy in India. By recognizing and tackling these issues, this project aims to propose solutions that will enhance the integrity and effectiveness of the electoral system, empowering citizens to exercise their voting rights without fear or undue influence.

To tackle these challenges, the project proposes a comprehensive set of solutions. These solutions encompass enhanced security protocols to safeguard polling stations, anti-corruption initiatives to raise awareness and deter electoral malpractices, improved accessibility through alternative voting methods, comprehensive voter education and awareness campaigns, and the promotion of transparent electoral processes.

By implementing these solutions, the project seeks to strengthen the democratic foundation of India's electoral system. The project recognizes the importance of upholding the principles of fairness, inclusivity, and transparency to ensure that every citizen can exercise their right to vote without coercion, fear, or undue influence. Through these endeavors, the aim is to foster a more vibrant, representative, and participatory democracy that reflects the true will of the people.

In the subsequent sections, this report will elaborate on each proposed solution, outlining their key features and potential impact on addressing the challenges faced during elections in India. By analyzing and implementing these solutions, we can take significant strides toward fortifying the electoral process and upholding the democratic ideals that form the cornerstone of India's governance.

## CHAPTER 2

### LITERATURE

The literature review section provides an overview of relevant research, studies, and publications related to the "जनमत" project, focusing on the key themes of digital voting, electoral systems, and technology-driven democracy. The literature review serves to establish the project's theoretical and conceptual foundations and to identify existing gaps and areas of potential contribution. The following is a summary of the literature review for the project:

#### 1. Digital Voting Systems:

- The literature on digital voting systems highlights the potential benefits of technology in enhancing the voting process, such as increased accessibility, efficiency, and accuracy.
- Research studies have examined various types of digital voting systems, including internet-based voting, mobile voting, and electronic voting machines, assessing their strengths, limitations, and security concerns.
- Existing literature emphasizes the importance of robust security measures, data integrity, and transparency in digital voting systems to maintain trust and confidence among voters and stakeholders.

#### 2. Electoral Systems and Democratic Processes:

- Literature on electoral systems explores different models of democracy, electoral laws, and practices around the world.
- Studies analyze the impact of electoral systems on representation, inclusivity, and political outcomes, highlighting the strengths and weaknesses of various electoral frameworks such as proportional representation and plurality systems.
- Scholars also examine challenges related to voter turnout, voter education, and participation in electoral processes, seeking innovative solutions to enhance citizen engagement and strengthen democratic institutions.

#### 3. Technology-Driven Democracy:

- The literature on technology-driven democracy investigates the role of technology in shaping democratic processes and citizen participation.
- Research focuses on the use of digital tools and platforms for civic engagement, political activism, and information dissemination.
- Studies explore the potential of technology to bridge the digital divide, amplify marginalized voices, and foster transparency and accountability in governance.
- Additionally, literature addresses the ethical and legal implications of utilizing technology in democratic practices, highlighting concerns such as privacy, data protection, and the risk of digital exclusion.

#### 4. Case Studies and Innovations:

- Case studies and innovative projects related to digital voting and technology-driven democracy provide valuable insights and practical examples.
- Researchers and practitioners share their experiences in implementing digital voting systems, highlighting success stories, challenges faced, and lessons learned.
- These case studies and innovations help inform the design and development of the "जनमत" project by drawing on real-world experiences and best practices.

By synthesizing and analyzing the existing literature, the project team gains a comprehensive understanding of the field and can identify opportunities for innovation and improvement in the "जनमत" project. The literature review also informs the project's methodology, design decisions, and the overall approach to addressing the challenges and limitations of the current electoral system.

## CHAPTER 3

### SCOPE FOR “जनमत”

#### Scope of the "जनमत" Project:

The scope of the "जनमत" project encompasses the design, development, and implementation of a digital voting platform aimed at enhancing the voting process and addressing the challenges faced during elections. The project focuses on the following key aspects:

##### 1. User Registration and Authentication:

- The project includes a user registration system that allows eligible voters to create accounts and authenticate their identities securely.
- The registration process may involve verification procedures to ensure the accuracy and integrity of user information.

##### 2. Candidate Management:

- The system provides functionalities for managing candidate profiles, including candidate registration, verification, and display of relevant information such as party affiliation, qualifications, and campaign materials.

##### 3. Voting Processes:

- The project aims to streamline the voting process by providing a user-friendly interface for casting votes.
- The system should support the selection of preferred candidates or parties and provide mechanisms to prevent multiple voting or fraudulent activities.

##### 4. Result Calculation and Reporting:

- The project includes algorithms and mechanisms for accurately calculating election results based on the votes cast.
- The system generates reports summarizing the election results, which can be made available to users and relevant authorities.

##### 5. Database Management:

- The project requires the establishment of a secure and reliable database to store and manage user data, candidate information, voting records, and election results.
- The database should be scalable, efficient, and designed to ensure data integrity and confidentiality.

##### 6. User Interface and Experience:

- The system's user interface (UI) should be intuitive, visually appealing, and responsive to provide a seamless user experience.
- The project may involve the construction of graphical user interfaces (GUIs) for users to interact with the application.

##### 7. Security and Privacy:

- The project prioritizes the implementation of robust security measures to protect user data, prevent unauthorized access, and ensure the integrity of the voting process.
- Consideration should be given to data encryption, secure authentication, and protection against cyber threats.

It is important to note that the scope of the "जनमत" project may vary based on specific requirements, resources, and constraints. Additional functionalities and features can be incorporated to meet the needs of different electoral systems and jurisdictions. The project team should define the scope clearly during the initial stages and continuously evaluate and adjust it as necessary throughout the project's lifecycle to ensure a successful and impactful implementation.

# CHAPTER 4

## SOFTWARE AND HARDWARE SPECIFICATIONS

### **4.1 Methods:**

The "जनमत" project aims to address the challenges faced during elections in India and provide a solution to improve the fairness and transparency of the electoral process. As a software engineer, we can utilize the following methods to successfully undertake the "जनमत" project:

#### **1. Requirement Analysis:**

Start by conducting a thorough requirement analysis to understand the specific needs and objectives of the project. Collaborate with stakeholders, election authorities, and potential users to gather detailed requirements. Identify the key functionalities, features, and constraints of the system.

#### **2. System Design:**

Based on the gathered requirements, design the overall system architecture and flow. Define the modules, components, and their interactions. Determine the user interface design, backend systems, database structure, and integration points. Consider scalability, security, and performance in the design.

#### **3. Technology Selection:**

Choose appropriate technologies, frameworks, and tools for the project. Consider factors such as scalability, security, ease of development, and compatibility. Select programming languages, frameworks, database management systems, and other tools that align with the project's objectives.

#### **4. Development:**

Implement the functionalities and features of the "जनमत" project. Divide the development tasks into smaller modules or user stories to facilitate incremental development. Follow coding best practices, maintain code quality, and ensure proper documentation. Implement security measures, such as authentication, access control, and data encryption.

#### **5. User Interface Design:**

Design an intuitive and user-friendly interface for the "जनमत" application. Consider the user experience, accessibility, and ease of use. Create visually appealing screens, layout designs, and interactive elements. Conduct usability testing and iterate on the design based on user feedback.

#### **6. Database Design and Implementation:**

Analyze the data requirements and design the database schema accordingly. Identify the entities, attributes, and relationships. Normalize the tables to minimize redundancy and ensure data integrity. Implement the database schema and perform necessary optimizations for efficient data storage and retrieval.

#### **7. Testing and Quality Assurance:**

Thoroughly test the application to ensure its functionality, reliability, and security. Conduct unit testing, integration testing, and system testing to identify and fix any bugs or issues. Perform performance testing to ensure the application can handle the expected user load. Implement automated testing frameworks to streamline the testing process.

#### **8. Deployment and Integration:**

Prepare the application for deployment by setting up the necessary infrastructure, servers, and hosting environment. Deploy the application on a reliable and scalable platform. Integrate with external systems or APIs for data exchange or third-party services, if required. Implement proper monitoring and logging mechanisms for real-time tracking of the application's performance.

#### **9. User Training and Documentation:**

Provide training and documentation to educate end-users, election authorities, and administrators on how to use the "जनमत" application effectively. Create user manuals, guides, and FAQs to assist users in navigating the system. Document the system architecture, modules, APIs, and dependencies for future maintenance and enhancements.

#### **10. Maintenance and Support:**

Ensure continuous maintenance and support for the "जनमत" application. Regularly monitor and address any issues or bugs that arise. Provide ongoing updates, bug fixes, and enhancements based on user feedback and changing requirements. Regularly backup data and ensure the application's security and stability.

Throughout the project, follow agile methodologies such as Scrum or Kanban to facilitate iterative development and collaboration. Utilize project management tools, version control systems, and communication platforms to ensure effective project tracking and team coordination.

By employing these methods, we can effectively plan, design, develop, and implement the "जनमत" project, contributing to improving the electoral process and empowering citizens in India to participate in a fair and transparent democracy.

## **4.2 Programming/Working Environment:**

The application can be developed using a combination of programming languages, frameworks, and tools, depending on the specific requirements. Some commonly used technologies for developing such applications include:

- Programming Languages: Java, Python, C#, PHP, JavaScript
- Frameworks: Spring, Django, .NET, Laravel, React, Angular
- Databases: MySQL, PostgreSQL, MongoDB
- Development Tools: Integrated Development Environments (IDEs) such as Eclipse, Visual Studio, or IntelliJ IDEA, version control systems (e.g., Git), project management tools, and testing frameworks.

## **4.3 Requirements to Run the Application:**

### **1. Software Requirements:**

- Operating System: Windows, macOS, Linux
- Web Browsers: Chrome, Firefox, Safari, Edge
- Java Runtime Environment (JRE) or Python interpreter (depending on the chosen programming language)
- Database Management System (DBMS) software (e.g., MySQL, PostgreSQL) if a database is used.

### **2. Hardware Requirements:**

- Processor: Intel Core i5 or equivalent
- RAM: Minimum 4 GB (8 GB or higher recommended for optimal performance)
- Storage: Sufficient disk space to store the application and associated data
- Display: Monitor or screen with a resolution of 1280x800 or higher
- Internet connectivity for accessing online platforms and databases.

It is important to note that the specific software and hardware requirements may vary depending on the scope and complexity of the application. The mentioned requirements provide a general outline for the development and execution of the proposed solutions in the project.

## CHAPTER 5

# DATABASE AND IMPLEMENTATION

In the context of the project addressing challenges during elections in India and proposing solutions, the database plays a vital role in storing and managing various data related to voters, candidates, polling stations, election results, and other relevant information. An effective database system is crucial for ensuring the integrity, efficiency, and transparency of the electoral process. Here is an overview of the database analyzing, design, and implementation for the project:

### 1. Database Analysis:

Database analysis involves understanding the requirements and objectives of the project and identifying the data entities and their relationships. It includes:

- Identifying the key entities: Voters, candidates, polling stations, election campaigns, election results, etc.
- Determining the attributes and properties of each entity: Name, age, address, voter ID, party affiliation, vote count, election date, etc.
- Analyzing the relationships between entities: One-to-one, one-to-many, or many-to-many relationships to establish data associations and dependencies.

### 2. Database Design:

Database design translates the requirements identified during the analysis phase into a logical and efficient structure. It involves:

- Conceptual Design: Creating an Entity-Relationship Diagram (ERD) that represents the entities, relationships, and attributes. The ERD provides a high-level view of the database structure and helps in understanding the data flow.
- Logical Design: Transforming the conceptual model into a detailed logical model. This includes defining tables, primary and foreign keys, constraints, and normalization techniques to minimize redundancy and ensure data integrity.
- Physical Design: Implementing logical design in a specific database management system (DBMS). This includes decisions on indexing, partitioning, and storage allocation for optimal performance.

### 3. Database Implementation:

Database implementation involves creating the database schema, tables, and relationships in the chosen DBMS. It includes:

- Creating the tables: Defining the tables based on the logical design, including the data types, lengths, and constraints of each attribute.
- Defining relationships: Establishing relationships between tables using primary and foreign keys to enforce referential integrity.
- Writing SQL queries: Implementing SQL statements for creating tables, inserting data, updating records, and retrieving information from the database.
- Security measures: Implementing appropriate security measures such as user authentication, access control, and data encryption to protect the confidentiality and integrity of the data.

It is important to regularly maintain the database by performing backups, optimizing performance, and ensuring data consistency. Additionally, data archival processes can be implemented to store historical election data for future reference and analysis.

By analyzing, designing, and implementing a well-structured and secure database system, the project aims to provide a reliable foundation for storing and managing electoral data. This database will support the proposed solutions and facilitate transparent, accurate, and efficient elections, ultimately strengthening the democratic process in India.

# CHAPTER 6

## Program's Structure Analyzing and GUI Constructing

### 1. Program's Structure Analysis:

Before constructing the GUI (Graphical User Interface) for the "जनमत" project, it is essential to analyze and define the program's structure. This involves breaking down the functionality into logical components and determining how they interact with each other. Here are the key steps involved:

- a. Identify Modules: Identify the major functional modules or components of the "जनमत" project. For example, we might have modules for user registration, candidate management, voting process, result calculation, and reporting.
- b. Define Module Interactions: Determine how these modules interact with each other. Identify the flow of data and actions between different modules. Consider the dependencies and the sequence in which modules need to be executed.
- c. Design Data Structures: Analyze the data requirements of each module and design the appropriate data structures to store and manipulate the data. This includes defining the necessary classes, objects, and variables to represent different entities such as voters, candidates, and election results.
- d. Identify Functionality: Determine the specific functionality and operations that each module should perform. Define the input and output requirements for each module. Consider error handling and validation to ensure data integrity and reliability.
- e. Define APIs/Interfaces: If the project involves integration with external systems or APIs, define the interfaces or APIs required to interact with those systems. Specify the input and output parameters and the expected behavior of these interfaces.

### 2. GUI Construction:

Once the program's structure is analyzed, the next step is to construct the Graphical User Interface (GUI) for the "जनमत" project. The GUI provides an interactive and user-friendly interface for users to interact with the system. Here's how we can construct the GUI effectively:

- a. Identify User Requirements: Understand the needs and preferences of the end-users. Identify the key functionalities and actions they need to perform through the GUI. Consider the user experience, accessibility, and ease of use in designing the GUI.
- b. Wireframing and Design: Create wireframes or mockups of the GUI to visualize the structure. Consider the placement of different elements such as buttons, forms, tables, and navigation menus. Design the GUI with a clean and intuitive interface, adhering to established design principles.
- c. Implement GUI Components: Use appropriate programming languages, frameworks, and libraries to implement the GUI components. Utilize libraries such as Swing or JavaFX for Java-based applications or frameworks like React or Angular for web-based applications. Implement the necessary forms, buttons, dropdowns, tables, and other interactive elements.
- d. Implement Event Handling: Associate appropriate event handlers with GUI components to capture user actions. Define the behavior of the GUI components when the user interacts with them, such as button clicks, form submissions, or menu selections.
- e. Validate User Input: Implement validation mechanisms to ensure the accuracy and integrity of user input. Validate and sanitize user inputs to prevent data corruption or security vulnerabilities.
- f. Error Handling and Feedback: Implement error handling mechanisms to display meaningful error messages to users when something goes wrong. Provide appropriate feedback to users for successful operations, such as displaying success messages or progress indicators.
- g. Testing and Iteration: Perform thorough testing of the GUI to ensure its functionality, usability, and responsiveness. Seek feedback from users and stakeholders to make iterative improvements to the GUI design and behavior.

# CHAPTER 7

## CODE IMPLEMENTATION AND DATABASE CONNECTION

### CODE IMPLEMENTATION:

To implement the "जनमत" project, we have written code to handle the various functionalities and interactions within the system. Here are some key areas where code implementation is required:

#### 1. User Registration:

- Implement code to capture user registration details such as name, email, and password.
- Validate the input and store the user information securely in the database.
- Implement appropriate error handling and feedback messages for registration success or failure.

#### 2. Candidate Management:

- Develop code to add, update, and delete candidate information.
- Implement functionality to retrieve candidate details from the database.
- Handle operations such as candidate nomination, withdrawal, and verification.

#### 3. Voting Process:

- Design and implement the code to facilitate the voting process.
- Verify the eligibility of voters and candidates before allowing the vote.
- Update the vote count and track the voter's choice in the database.
- Implement measures to prevent multiple voting or fraudulent activities.

#### 4. Result Calculation:

- Write code to calculate and determine the election results.
- Implement algorithms or rules for vote counting, considering factors such as majority, proportional representation, or any specific electoral system.
- Generate reports or visual representations of the election results.

#### 5. Reporting and Analytics:

- Develop code to generate various reports and analytics related to the election process.
- Implement functionality to extract data from the database and present it in meaningful formats, such as charts or graphs.
- Enable users to access and analyze election data based on different parameters or criteria.

### DATABASE CONNECTION:

To connect our application with a database, we established a connection and performed database operations using appropriate APIs or libraries. Here's an outline of the database connection process:

#### 1. Choose a Database Management System (DBMS):

- Select a suitable DBMS that aligns with the requirements of the project. Popular options include MySQL, PostgreSQL, MongoDB, or SQLite.

#### 2. Install and Configure the DBMS:

- Install the chosen DBMS and configure it according to our development environment.
- Set up the necessary user accounts, privileges, and database schemas.

#### 3. Choose a Database Connectivity Library:

- Depending on the programming language we are using, select a library or framework that provides database connectivity.
- Examples include JDBC for Java, SQLAlchemy for Python, or Sequelize for Node.js.

#### 4. Establish a Connection:

- Write code to establish a connection to the database using the appropriate driver or connector for our chosen DBMS.
- Provide the necessary credentials, such as username, password, host, and port, to establish the connection.

## 5. Perform Database Operations:

- Once the connection is established, we can execute various database operations such as inserting, updating, retrieving, or deleting data.
- Write code to create SQL queries or use an ORM (Object-Relational Mapping) framework for database operations.
- Ensure proper error handling and transaction management to maintain data integrity.

## 6. Close the Connection:

- After completing the necessary database operations, close the connection to release resources and ensure a clean exit.

By implementing the required code and establishing a connection to the database, we have ensured the proper functioning and data management of the "जनमत" project.

Here are the snapshots of the following codes database connections

## Home page and its code :

Front end:

```
background : url('log01.svg') 50% 50% no-repeat ;
opacity: 0.2;
top : 0px;
left : 0px;
height : 100%;
width : 100%;
cursor : wait;
}
</style>
</head>
<body>
<h1>
    हर घर जनमत ✅
</h1>
<div id= "buttons">
    <div class="zoomin">
<a href='#info'>
    <button class="home_page1">
        News Updates
    </button>
</a>
</div>
<br>
<div class="zoomin">
<a href="#vote">
    <button class="home_page2">
        Vote now
    </button>
</a>
</div>
</div>

<!-- LOADER -->

<div class="loader">
    <script type="text/javascript" src="http://code.jquery.com/jquery-latest.js"></script>
```

# हर घर जनमत ✅



News Updates

Vote now

## LOGIN PAGE AND ITS CODE

Front end - HTML

```
// Establish the connection
connection.connect((err) => {
  if (err) {
    console.error('Error connecting to the database:', err);
    return;
  }
  console.log('Connected to the database');
});

// Create Express app
const app = express();

// Use body-parser middleware
app.use(bodyParser.urlencoded({ extended: true }));

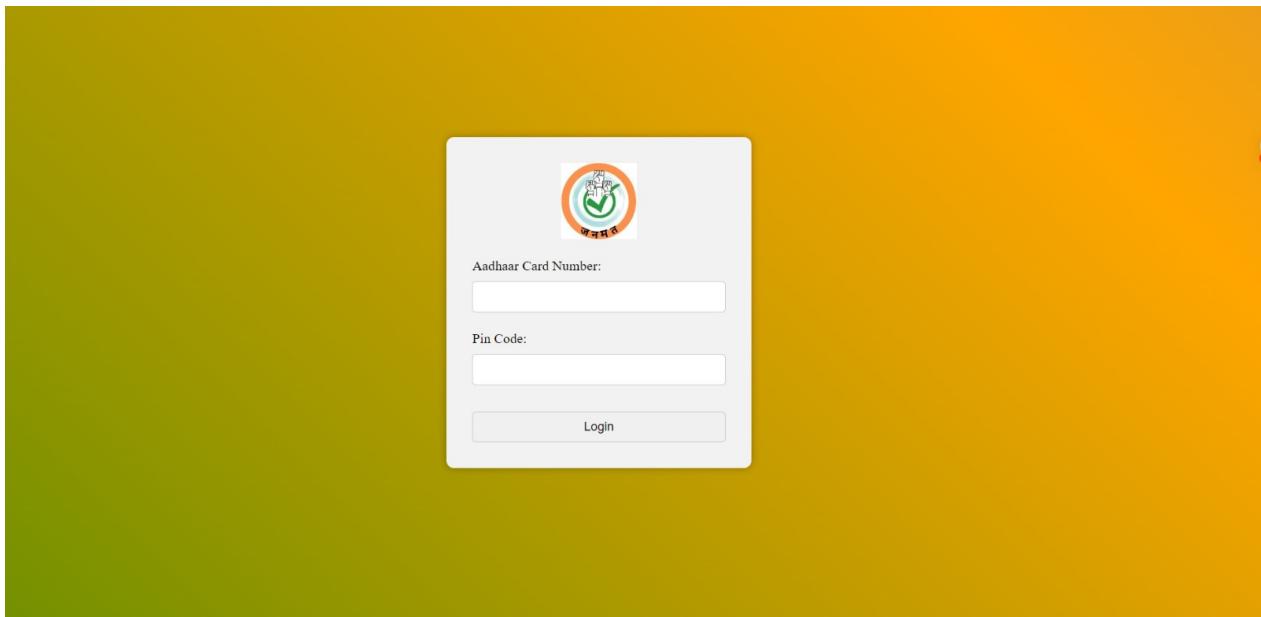
// Serve login page
app.get('/', (req, res) => {
  res.send(`|
    <form action="/login" method="POST">
      <input type="text" name="aadhar" placeholder="Aadhar" required />
      <input type="password" name="pin" placeholder="PIN" required />
      <button type="submit">Login</button>
    </form>
`);
```

## Front-end -CSS

```
* {
    margin: 0;
    padding: 0;
    box-sizing: border-box;
}

.container{
    height: 100vh;
    width: 100%;
    background: linear-gradient(45deg, #008000, #FFA500, #4169E1);
    background-size: 300% 300%;
    animation: color 10s ease-in-out infinite;
}

@keyframes color {
    0%{
        background-position: 0 50%;
    }
    50%{
        background-position: 100% 50%;
    }
    100%{
        background-position: 0 50%;
    }
}
```



## Backend -javascript

```
// Add event listener to Verify button
function myFunction() {
    const aadharValue = aadharInput.value.trim();
    const pinValue = pinInput.value.trim();

    // Check if both fields are valid
    if (/^\d{12}$/.test(aadharValue) && /^\d{6}$/.test(pinValue)) {
        aadharInput.parentNode.classList.add('valid');
        aadharInput.parentNode.querySelector('.icon').innerHTML = '\u2708';
        aadharInput.parentNode.querySelector('.status').innerHTML = 'Valid';

        // pinInput.parentNode.classList.remove("wrong");
        pinInput.parentNode.classList.add('valid');
        pinInput.parentNode.querySelector('.icon').innerHTML = '\u2708';
        pinInput.parentNode.querySelector('.status').innerHTML = 'Valid';
    }
}
```

```
<form action="http://localhost:3002/login" method="POST">

    <!-- <h1>Verify</h1> -->
    <div class="box">
        
        <div class="input">
            <label for="aadhar">Aadhaar Card Number:</label>
            <input type="text" id="aadhar" name="aadhar"/>
            <span class="icon"></span>
            <span class="status"></span>
        </div>
        <div class="input">
            <label for="pin">Pin Code:</label>
            <input type="text" id="pin" name="pin" />
            <span class="icon"></span>
            <span class="status"></span>
        </div>
        <p class="validation-text"></p>
        <!-- <a href="facial.html"> <button onclick="myFunction()" i
        <!-- <button onclick="myFunction()" id="verify">Verify</button>
        <input type="submit" value="Login">
    </div>
```

## Database

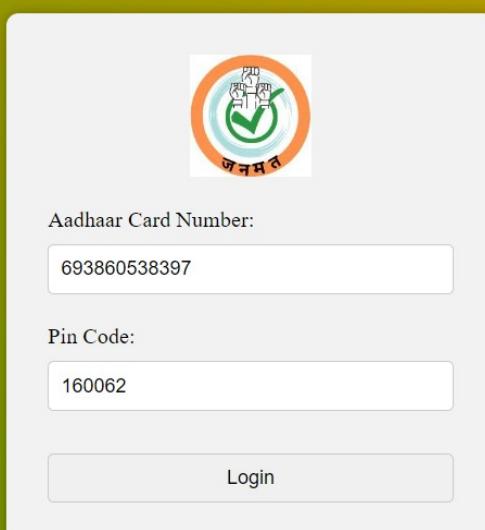
```
sql>
sql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| sys |
+-----+
4 rows in set (0.00 sec)

mysql> create database user_verification;
Query OK, 1 row affected (0.01 sec)

mysql> use user_verification;
Database changed
mysql> CREATE TABLE users (
    ->     id INT AUTO_INCREMENT PRIMARY KEY,
    ->     aadhar VARCHAR(12) NOT NULL,
    ->     pincode VARCHAR(6) NOT NULL,
    ->     name VARCHAR(255) NOT NULL,
    ->     dob DATE NOT NULL
    -> );
Query OK, 0 rows affected (0.03 sec)

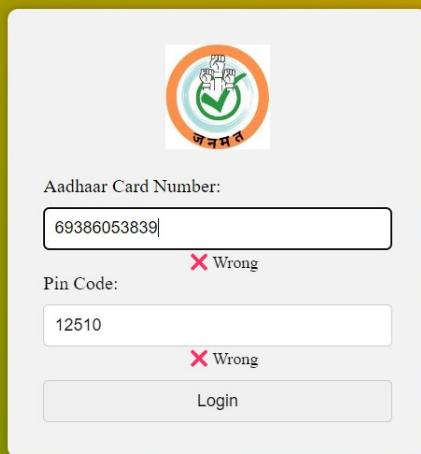
x  mysql> |
```

When user enter the correct details



Login successful

When the user enter the wrong details :



Invalid login credentials

## Facial recognition page

Front end :

```
</head>
<body>
    <div id="main">
        <h1>PLEASE BE STEADY..</h1>
        <div id="instructions">
            
            <h2>Instructions</h2>
            <p>1. Look into the camera</p>
            <p>2. Enter your name in
                the box below</p>
            <p>3. Click the
                Recognize button</p>
        </div>
        <div id="camera">
            <video id="video" width="600" height="400" margin-top="0px"></video>
            <br>
            <div id="inputs">
                <input type="text" id="name" placeholder="Name">
                <input type="button" id="start" value="Start">
                <input type="button" id="recognize" value="Recognize">
            </div>
        </div>
        <div id="results">
            <h2>Results</h2>
            <p>Name: <span id="output"></span></p>
        </div>
    </div>
</body>
```

# PLEASE BE STEADY..

**Results**

Name:

Start



**Instructions**

1. Look into the camera
2. Enter your name in the box below
3. Click the Recognize button

Javascript :

Code for starting of video stream

```
const start = document.getElementById('start');
const recognize = document.getElementById('recognize');
const name = document.getElementById('name');
const output = document.getElementById('output');

// Start video stream
start.addEventListener('click', () => {
    navigator.mediaDevices
        .getUserMedia({ video: true, audio: false })
        .then(stream => {
            video.srcObject = stream;
            video.play();
        });
});
```

## Code for facial recognition

```
// Recognize face
recognize.addEventListener('click', () => {
  if (name.value == '') {
    output.innerHTML = 'Please enter your name';
  } else {
    output.innerHTML = 'Recognizing face...';
    setTimeout(() => {
      output.innerHTML = `Welcome, ${name.value}!`;
    }, 3000);
  }
});
recognize.addEventListener('click', () => {
if (name.value == '') {
  output.innerHTML = 'Please enter your name';
} else {
  output.innerHTML = 'Recognizing face...';
  setTimeout(() => {
    const recognizedName = name.value;
    if (recognizedName === 'Shreshtha' || recognizedName === 'Sanchit') {
      window.location.href = 'finger.html'; // Replace 'fingerprint.html' with the actual URL of the next p
    } else {
      alert('Access denied!');
      window.location.href = 'index.html'; // Replace 'index.html' with the actual URL of the main page
    }
  }, 3000);
}
})
```

Java script code for verification and recognitions :

```
const video = document.getElementById("video");

Promise.all([
  faceapi.nets.ssdMobileNetv1.loadFromUri("/models"),
  faceapi.nets.faceRecognitionNet.loadFromUri("/models"),
  faceapi.nets.faceLandmark68Net.loadFromUri("/models"),
])
  .then(startWebcam)
  .then(faceRecognition);

function startWebcam() {
  navigator.mediaDevices
    .getUserMedia({
      video: true,
      audio: false,
    })
    .then((stream) => {
      video.srcObject = stream;
    })
    .catch((error) => {
      console.error(error);
    });
}
```

```

function getLabeledFaceDescriptions() {
  const labels = ["sanchit","shreshtha"];
  return Promise.all(
    labels.map(async (label) => {
      const descriptions = [];
      for (let i = 1; i <= 2; i++) {
        const img = await faceapi.fetchImage(`./labels/${label}/${i}.jpeg`);
        const detections = await faceapi
          .detectSingleFace(img)
          .withFaceLandmarks()
          .withFaceDescriptor();
        descriptions.push(detections.descriptor);
      }
      return new faceapi.LabeledFaceDescriptors(label, descriptions);
    })
  );
}

async function faceRecognition() {
  const labeledFaceDescriptors = await getLabeledFaceDescriptions();
  const faceMatcher = new faceapi.FaceMatcher(labeledFaceDescriptors);

  video.addEventListener("playing", () => {
    location.reload();
  });
}

```

```

async function faceRecognition() {
  const labeledFaceDescriptors = await getLabeledFaceDescriptions();
  const faceMatcher = new faceapi.FaceMatcher(labeledFaceDescriptors);

  video.addEventListener("playing", () => {
    location.reload();
  });

  const canvas = faceapi.createCanvasFromMedia(video);
  document.body.append(canvas);

  const displaySize = { width: video.width, height: video.height };
  faceapi.matchDimensions(canvas, displaySize);

  setInterval(async () => {
    const detections = await faceapi
      .detectAllFaces(video)
      .withFaceLandmarks()
      .withFaceDescriptors();

    const resizedDetections = faceapi.resizeResults(detections, displaySize);

    canvas.getContext("2d").clearRect(0, 0, canvas.width, canvas.height);

    const results = resizedDetections.map((d) => {
      return faceMatcher.findBestMatch(d.descriptor);
    });
  });
}

```

## Modules and library for facial recognition

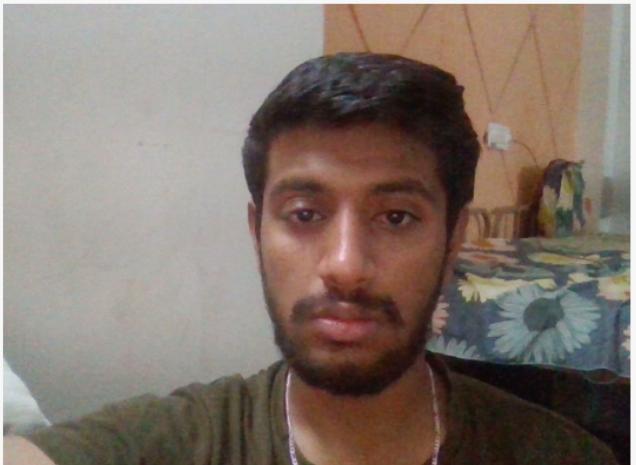
```
!function(t,e){"object"==typeof exports&&"undefined"!=typeof module?e(exports):"function"==typeof define&&define.amd?define(["exports"],e):e((t=t||self).faceapi=t.faceapi||{})(this,function(c){"use strict";var r=function(t,e){return(r=Object.setPrototypeOf||{__proto__:[]})instanceof Array&&function(t,e){t.__proto__=e}||function(t,e){for(var n in e)e.hasOwnProperty(n)&&(t[n]=e[n])})(t,e);function t(t,e){function n(){(this.constructor=t)r(t,e),t.prototype=null==e?Object.create(e):(n.prototype=e.prototype,new n)}}function y(i,a,s,u){return new(s=||Promise)(function(t,e){function n(t){try{o=u.next(t))}catch(t){e(t)}}function r(t){try{o=u.throw(t))}catch(t){e(t)}}function o(e){e.done?t.e.value):new s(function(t){t(e.value)}).then(n,r)}o((u=u.apply(i,a||[])).next())}}function R(n,r){var o,i,a,t,s={label:0, sent:function(){if(1&a[0])throw a[1];return a[1]},trys:[],ops:[],t={next:e(0),throw:e(1),return:e(2)}, "function"==typeof Symbol&&(t[Symbol.iterator]=function(){return this}),t;function e(e){return function(t){return function(e){if(o)throw new TypeError("Generator is already executing.");for(;s;)try{if(o=1,i&&(a=2&e[0]?i.return:e[0])?i.throw||(a=i.return)&&a.call(i,0):i.next))&&!(a=a.call(i,e[1])).done)return a;switch(i=0,a&&(e=[2&e[0],a.value]),e[0]) {case 0:case 1:a=e;break;case 4:return s.label++,{value:e[1],done:!1};case 5:s.label++,i=e[1],e=[0];continue;case 7:e=s.ops.pop(),s.trys.pop();continue;default:if(!(a=0<(a=s.trys).length&&a[a.length-1])&&(6==e[0]||2==e[0])){s=0;continue} if(3==e[0]&&(!a||e[1]>a[0]&&e[1]<a[3])){s.label=e[1];break}if(6==e[0]&&s.label<a[1]){s.label=a[1],a=e;break}if(a&&s.label<a[2]){s.label=a[2],s.ops.push(e);break}a[2]&&s.ops.pop(),s.trys.pop();continue}e=r.call(n,s) catch(t){e=[6,t],i=0} finally{o=a=0}if(5&e[0])throw e[1];return{value:e[0]?e[1]:void 0,done:!0}}([e,t])}}var o=(e.prototype.setPlatform=function(t,e){null!=this.platform&&console.warn("Platform "+this.platformName+" has already been set. Overwriting the platform with "+e+"."),this.platformName=t,this.platform=e},e.prototype.registerFlag=function(t,e,n){if(this.flagRegistry[t]={evaluationFn:e,setHook:n},null!=this.urlFlags[t])var r=this.urlFlags[t];console.warn("Setting feature override from URL "+t+": "+r+"."),this.set(t,r)},e.prototype.get=function(t){return t in this.flags||(this.flags[t]=this.evaluateFlag(t)),this.flags[t]},e.prototype.getNumber=function(t){return this.get(t)},e.prototype.getBool=function(t){return this.get(t)},e.prototype.getFlags=function(){return this.flags},Object.defineProperty(e.prototype,"features",{get:function(){return this.flags},enumerable:!0,configurable:!0}),e.prototype.set=function(t,e){if(null==this.flagRegistry[t])throw new Error("Cannot set flag "+t+" as it has not been registered.");this.flags[t]=e, null!=this.flagRegistry[t].setHook&&this.flagRegistry[t].setHook(e)},e.prototype.evaluateFlag=function(t){if(null==this.flagRegistry[t])throw new Error("Cannot evaluate flag "+t+": no evaluation function found.");return this.flagRegistry[t].evaluationFn()},e.prototype.setFlags=function(t){this.flags=Object.assign({},t)},e.prototype.reset=function(){this.
```

Video streaming :

## PLEASE BE STEADY..

### Results

Name:



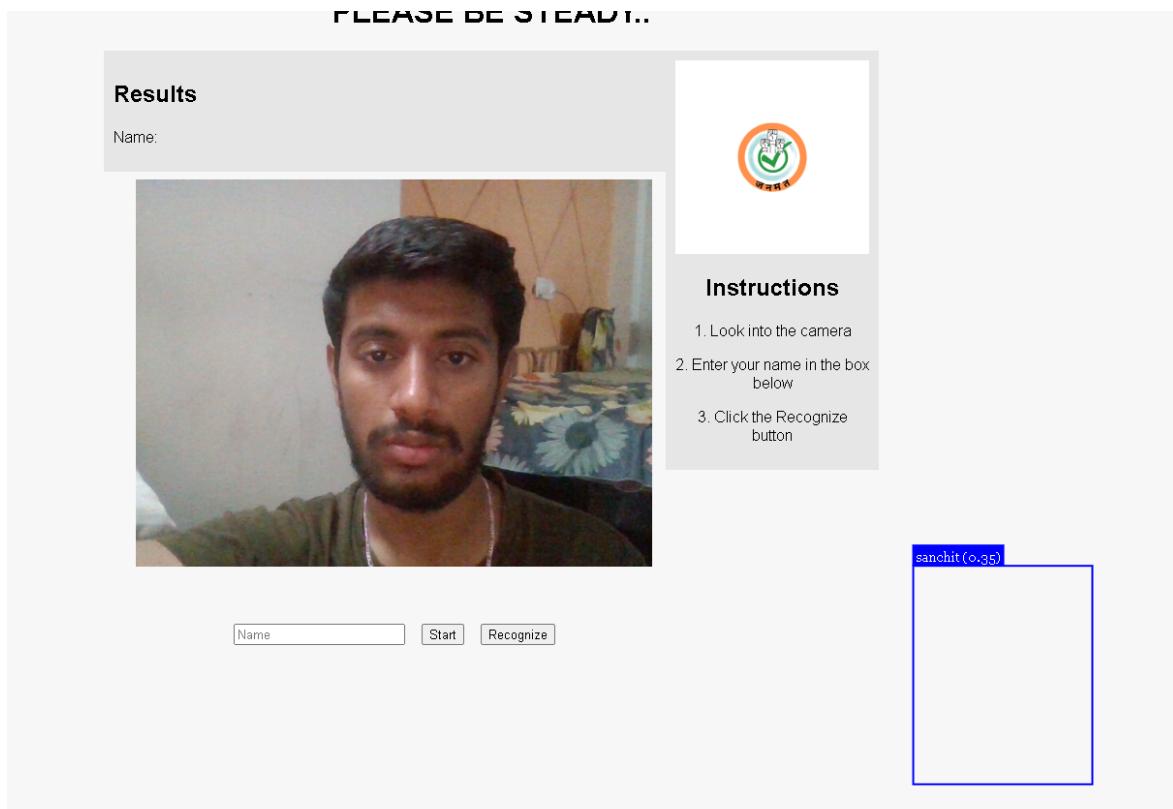


### Instructions

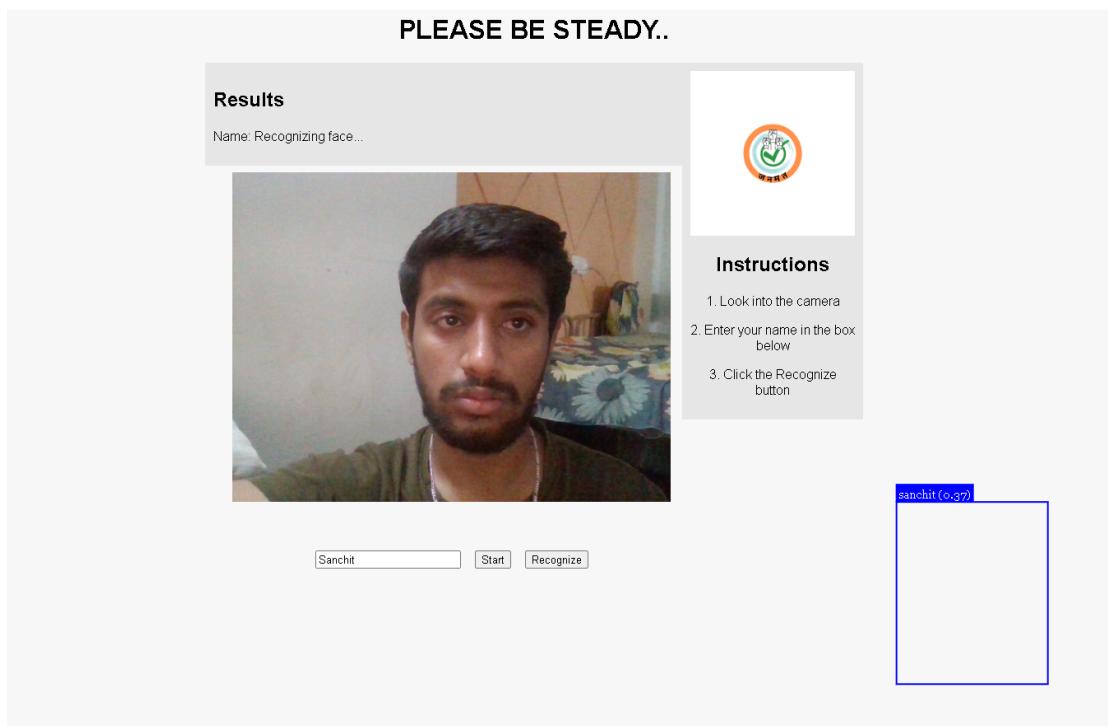
1. Look into the camera
2. Enter your name in the box below
3. Click the Recognize button

StartRecognize

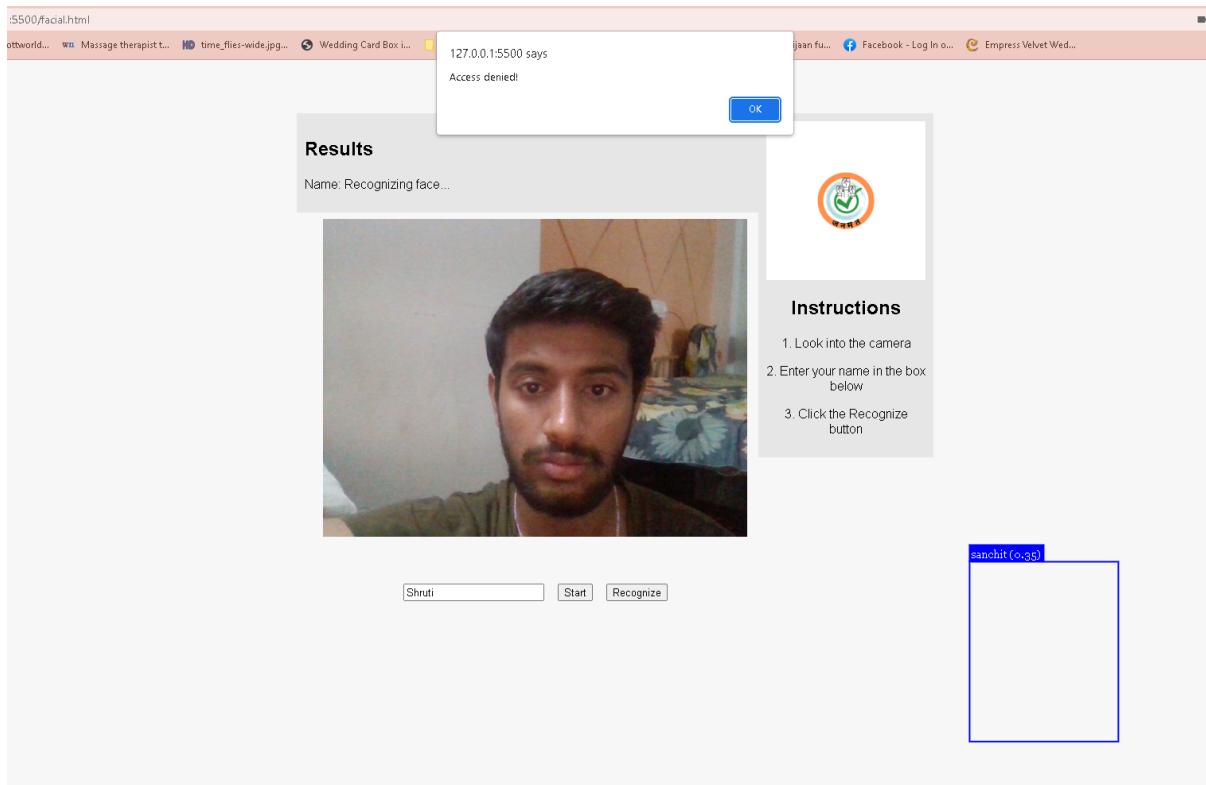
Facial recognition along with the name stored in the database



Snippet while reckoning of the face :  
If the face matches with the database it takes on to next page which is the fingerprint access



If the name and the person are different then the access is denied as shown in the snippet below:



## Fingerprint recognition page

Front end :

```
<body>

    <div id="con">
        
        <H1 id="text1">
            Finger Print verification
        </H1>

        <div id="container">

            <div id="fingerprint-container"></div>
            

            <div id="instructions-container">
                <h2>Instructions</h2>
                <ol>
                    <li>Place your finger on the scanner.</li>
                    <li>Wait for the scanner to read your fingerprint.</li>
                    <li>Click the Verify button to submit your fingerprint for verification.</li>
                </ol>
                <button id="verify-btn">Verify</button>
            </div>
        </div>
    </div>
<script>
```

## Java script for the fingerprint recognition

```
const fingerprintContainer = document.querySelector('#fingerprint-container');
const verifyBtn = document.querySelector('#verify-btn');

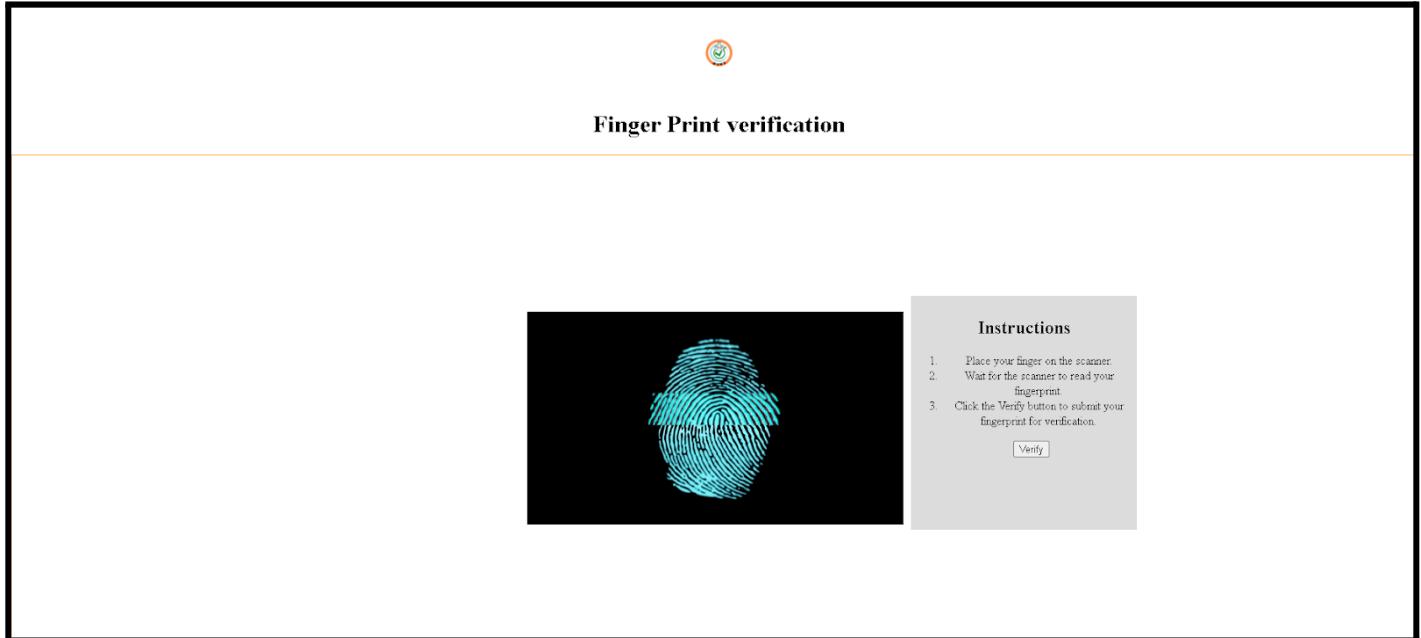
// Initialize fingerprint scanner
const scanner = new FingerprintScanner();

// Render fingerprint scanner in container
scanner.render(fingerprintContainer);

// Add event listener to verify button
verifyBtn.addEventListener('click', () => {
  // Get fingerprint data from scanner
});

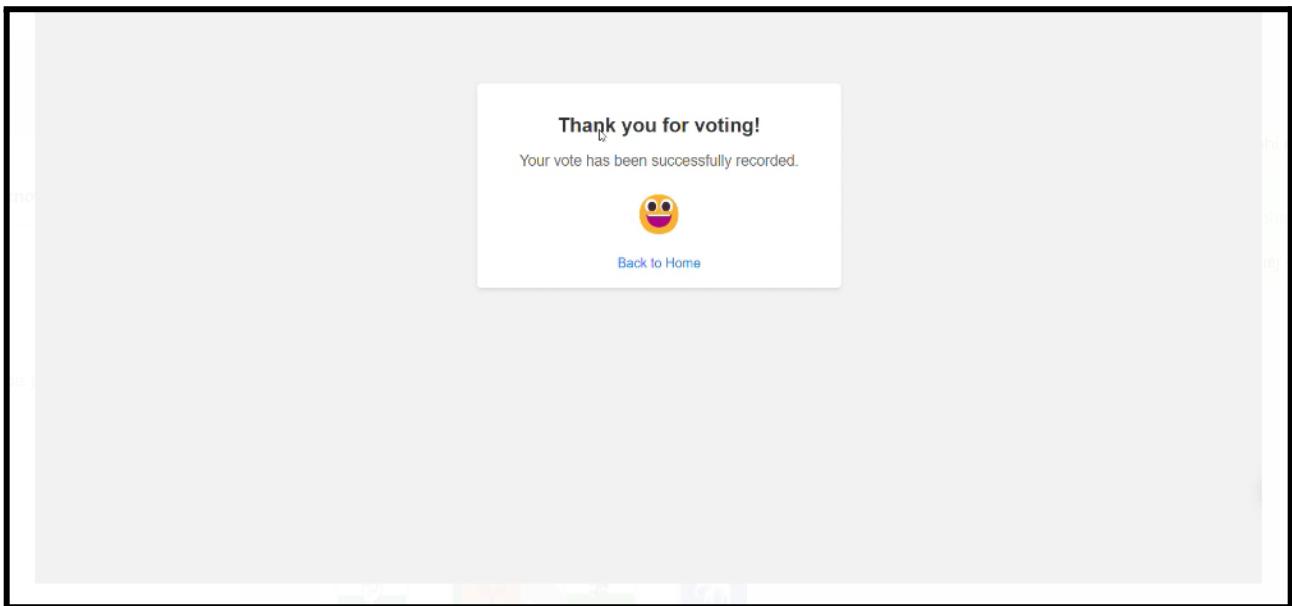
verifyBtn.addEventListener('click', () => {
  // Code to verify fingerprint here...

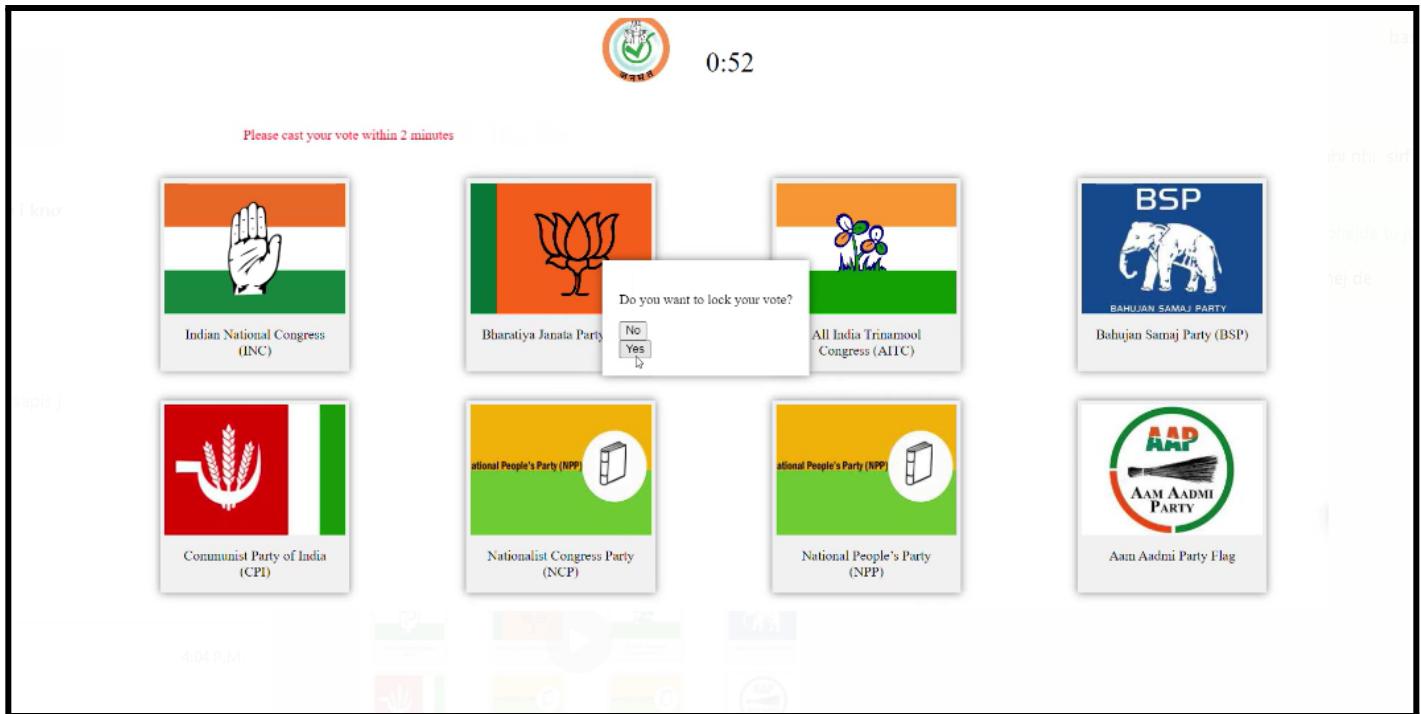
  // Display dialog box with verification message
  const dialogBox = document.createElement('div');
  dialogBox.classList.add('dialog-box');
  dialogBox.textContent = 'You have been verified!';
  document.body.appendChild(dialogBox);
```



# Voting page

## Front end :





acharuk\_012@gmail.com

localhost:1000/result

Result of Voting

INC : 0
BJP : 3
AITC : 1
BSP : 0
CPI : 0
NCP : 0
NPP : 0
AAP : 0

This HTML code represents a page that displays the results of the voting process.

The `<h1>` element displays the heading "Result of Voting" as the main title of the page.

Inside the `<div>` element with the class "container", there are several `<div>` elements with the class "result". These elements display the results for each political party.

The `<%= data.cX %>` syntax is typically used in server-side templating systems (e.g., Express.js with EJS) to insert dynamic content into the HTML page. Here, `data` is an object containing the voting results for each party, and `cX` represents the count for a particular party. For example, `<%= data.c1 %>` will display the vote count for the Indian National Congress (INC).

The `<a>` element represents a hyperlink and is used to create a link to the home page. The `href="/" href` attribute specifies the URL of the home page. The text "Back to Home" is the visible text for the link.

The class names ("container", "result", and "back-home") can be used to apply CSS styles or target these elements using JavaScript or other techniques.

Regarding the privacy of the voting result page, the code you provided is a static HTML code snippet. In its current form, this page can be accessed by anyone who has the URL. To restrict access and ensure privacy, you would need to implement appropriate server-side authentication and authorization mechanisms. These mechanisms would require users to authenticate (e.g., with a username and password) and ensure that only authorized individuals can access the voting result page.

```
16      <!-- Card grid section -->
17      <div class="card-grid">
18          <!-- Card 1 -->
19          <div class="card">
20              
21              <p>Indian National Congress (INC)</p>
22              <input type="hidden" value="1" id="card1" />
23          </div>
24          <!-- Card 2 -->
25          <div class="card" onclick="showPopup('Card 2')" id="2" name="b">
26              
27              <p>Bharatiya Janata Party (BJP)</p>
28              <input type="hidden" value="2" id="card2" />
29          </div>
30          <!-- Card 3 -->
31          <div class="card" onclick="showPopup('Card 3')" id="3" name="c">
32              
33              <p>All India Trinamool Congress (AITC)</p>
34              <input type="hidden" value="3" id="card3" />
35          </div>
36          <!-- Card 4 -->
37          <div class="card" onclick="showPopup('Card 4')" id="4" name="d">
38              
39              <p>Bahujan Samaj Party (BSP)</p>
40              <input type="hidden" value="4" id="card4" />
41          </div>
42          <!-- Card 5 -->
43          <div class="card" onclick="showPopup('Card 5')" id="5" name="e">
44              
45              <p>Communist Party of India (CPI)</p>
46              <input type="hidden" value="5" id="card5" />
47          </div>
48          <!-- Card 6 -->
49          <div class="card" onclick="showPopup('Card 6')" id="6" name="f">
50              
51              <p>Nationalist Congress Party (NCP)</p>
52              <input type="hidden" value="6" id="card6" />
53          </div>
54          <!-- Card 7 -->
55          <div class="card" onclick="showPopup('Card 7')" id="7" name="g">
56              
57              <p>National People's Party (NPP)</p>
58              <input type="hidden" value="7" id="card7" />
59          </div>
60          <!-- Card 8 -->
61          <div class="card" onclick="showPopup('Card 8')" id="8" name="h">
62              
63              <p>Aam Aadmi Party Flag</p>
64              <input type="hidden" value="8" id="card8" />
65          </div>
66      </div>
67  
```

```

79 <!-- JavaScript section -->
80 <script>
81     // Function to show the popup and store the selected option
82     function showPopup(option1) {
83         document.getElementById("popup").style.display = "block";
84         option = document.getElementById("myInput");
85         option.value = option1; // Stores the selected option value
86     }
87
88     // Function to hide the popup
89     function hidePopup() {
90         document.getElementById("popup").style.display = "none";
91     }
92
93     // Timer functionality
94     var timeLimit = 120;
95     var timer = document.getElementById("timer");
96     timer.innerHTML = formatTime(timeLimit);
97     var countdown = setInterval(function () {
98         timeLimit--;
99         timer.innerHTML = formatTime(timeLimit);
100        if (timeLimit <= 10) {
101            timer.style.color = "red";
102        }
103        if (timeLimit <= 0) {
104            clearInterval(countdown);
105            window.alert("Voting time is over!");
106        }
107    }, 1000);
108
109    // Format time in minutes and seconds
110    function formatTime(time) {
111        var minutes = Math.floor(time / 60);
112        var seconds = time % 60;
113        if (seconds < 10) {
114            seconds = "0" + seconds;
115        }
116        return minutes + ":" + seconds;
117    }
118 </script>
<body>
    <div class="container">
        <h1>Thank you for voting!</h1>
        <p>Your vote has been successfully recorded.</p>
        <div class="emoji">😊</div>
        <a href="/" class="back-home">Back to Home</a>
    </div>
</body>

```

This HTML code represents a page that displays the results of the voting process.

The `<h1>` element displays the heading "Result of Voting" as the main title of the page.

Inside the `<div>` element with the class "container", there are several `<div>` elements with the class "result". These elements display the results for each political party.

The `<%= data.cX %>` syntax is typically used in server-side templating systems (e.g., Express.js with EJS) to insert dynamic content into the HTML page. Here, `data` is an object containing the voting results for each party, and `cX` represents the count for a particular party. For example, `<%= data.c1 %>` will display the vote count for the Indian National Congress (INC).

#### Middleware and Configuration:

`app.use()` is used to set up middleware functions for the application.

`bodyParser.json()` parses JSON-formatted request bodies.

`app.set('view engine', 'ejs')` sets the view engine to EJS (Embedded JavaScript) for rendering dynamic HTML templates.

`express.static('public')` serves the static files (such as CSS and JavaScript) from the "public" directory.  
`express.urlencoded({ extended: true })` is used to parse URL-encoded request bodies.

#### routing:

This sets up a GET route for the home page ("").

When a user visits the home page, the server renders the "pollBooth" view.

```

<body>
  <div class="container">
    <h1>Result of Voting</h1>
    <!-- Display the results for each party -->
    <div class="result">INC : <%= data.c1 %></div>
    <div class="result">BJP : <%= data.c2 %></div>
    <div class="result">AITC : <%= data.c3 %></div>
    <div class="result">BSP : <%= data.c4 %></div>
    <div class="result">CPI : <%= data.c5 %></div>
    <div class="result">NCP : <%= data.c6 %></div>
    <div class="result">NPP : <%= data.c7 %></div>
    <div class="result">AAP : <%= data.c8 %></div>
    <!-- Link to navigate back to the home page -->
    <a href="/" class="back-home">Back to Home</a>
  </div>
</body>

```

```

app.listen(3000, () => {
  console.log('Server started on port 3000');
});

```

### Voting Result Page:

This sets up a GET route for the voting result page ("/result").

When a user requests the voting result page, the server prepares a data object containing the vote counts for each option.

The server renders the "result" view, passing the data object as a parameter to display the voting results.

```

app.get("/result", function(req, res) {
  const data = {
    c1: count1,
    c2: count2,
    c3: count3,
    c4: count4,
    c5: count5,
    c6: count6,
    c7: count7,
    c8: count8
  }

  res.render("result", {
    data: data
  });
}

```

```

app.post("/voted", function(req, res) {
  const option = req.body.cardNumber;

  if (option == "Card 1") {
    count1++;
  } else if (option == "Card 2") {
    count2++;
  } else if (option == "Card 3") {
    count3++;
  } else if (option == "Card 4") {
    count4++;
  } else if (option == "Card 5") {
    count5++;
  } else if (option == "Card 6") {
    count6++;
  } else if (option == "Card 7") {
    count7++;
  } else if (option == "Card 8") {
    count8++;
  }

  console.log(count1 + " " + count2 + " " + count3 + " " + count4 + " " + count5 + " " + count6 + " " + count7 + " " + count8 + " ");

  res.render("votedSuccess");
})

```

#### Vote Submission:

This sets up a POST route for vote submission ("/voted").

When a user submits their vote, the server retrieves the selected option from the request body.

Based on the selected option, the corresponding vote count variable is incremented.

The updated vote counts are logged to the console.

The server renders the "votedSuccess" view to display a success message after vote submission.

```

let count1 = 0;
let count2 = 0;
let count3 = 0;
let count4 = 0;
let count5 = 0;
let count6 = 0;
let count7 = 0;
let count8 = 0;

app.get("/", function(req, res) {
    res.render("pollBooth");
})

```

```

app.use(bodyParser.json());
app.set('view engine', 'ejs');
app.use(express.static('public'))
app.use(express.urlencoded({ extended: true }));
app.use(bodyParser.urlencoded({ extended: true }));

```

```

const express = require('express');
const app = express();
const bodyParser = require('body-parser');
const popup = require('node-popup');

```

```

69   <!-- Popup section -->
70   <div class="popup" id="popup">
71     <p>Do you want to lock your vote?</p>
72     <button onclick="hidePopup()">No</button>
73     <form action="/voted" method="post">
74       <input type="hidden" name="cardNumber" id="myInput"/>
75       <button type="submit">Yes</button>
76     </form>
77   </div>

```

Voting Result Page:

This sets up a GET route for the voting result page ("result").

When a user requests the voting result page, the server prepares a data object containing the vote counts for each option.

The server renders the "result" view, passing the data object as a parameter to display the voting results.

Server Startup:

This starts the server and makes it listen on port 3000.

Once the server starts successfully, a message is logged to the console.

Overall, this code sets up a basic web application using Express and EJS for a poll booth. It allows users to submit their votes, updates the vote counts, and displays the voting results on a separate page.

# CHAPTER 8

## SYSTEM TESTING

System testing is an essential phase in the development of the "जनमत" project to ensure that the entire system functions as expected and meets the specified requirements. It involves testing the integrated components, modules, and functionalities of the system as a whole. Here's an overview of the system testing process for the "जनमत" project:

### 1. Test Planning:

- Define the testing objectives, scope, and test criteria.
- Identify the key functionalities, modules, and system components to be tested.
- Determine the testing approach, such as manual testing, automated testing, or a combination of both.
- Create a test plan that outlines the test strategy, test cases, and test data requirements.

### 2. Test Environment Setup:

- Set up a dedicated testing environment that closely resembles the production environment.
- Install the necessary software, libraries, and dependencies required for testing.
- Configure the testing environment to replicate real-world scenarios, including simulated user interactions and data.

### 3. Test Case Development:

- Develop comprehensive test cases based on the identified functionalities and requirements.
- Each test case should include the preconditions, steps to be executed, and expected results.
- Cover both positive and negative test scenarios to validate the system's behavior under different conditions.
- Consider edge cases, boundary values, and stress testing to assess the system's robustness and performance.

### 4. Test Execution:

- Execute the developed test cases systematically, following the test plan.
- Ensure that all test cases are executed and observed for expected results.
- Record the actual results and any deviations or issues encountered during testing.
- Perform regression testing to ensure that new changes or fixes do not affect previously tested functionalities.

### 5. Defect Tracking and Management:

- Use a defect tracking system or tool to log and manage identified defects or issues.
- Prioritize and assign the defects to the development team for resolution.
- Maintain clear communication channels between the testing and development teams to ensure efficient defect resolution.

### 6. Error Handling and Recovery Testing:

- Validate the system's behavior in error scenarios, such as network disruptions, database failures, or unexpected user inputs.
- Test the system's error handling mechanisms, including error messages, alerts, and graceful recovery.
- Ensure that the system maintains data integrity and stability even in error conditions.

### 7. Performance Testing:

- Conduct performance testing to assess the system's response time, scalability, and resource utilization under expected load conditions.
- Use load testing tools to simulate concurrent user activity and measure the system's performance metrics.
- Identify any bottlenecks or performance issues and optimize the system accordingly.

### 8. User Acceptance Testing (UAT):

- Involve end-users or representatives in the UAT phase to validate the system's functionality and usability.
- Collect feedback and observations from users to identify any gaps or improvements required.
- Address user feedback and make necessary adjustments to enhance the user experience.

### 9. Documentation:

- Document the test cases, test results, and any identified issues or defects.
- Provide comprehensive documentation on the testing approach, strategies, and test environment setup.

By conducting thorough system testing, we can identify and rectify any functional, performance, or usability issues in the "जनमत" project. It ensures that the system meets the desired quality standards and provides a reliable and user-friendly platform for the electoral process.

## CHAPTER 9 LIMITATIONS

While the "जनमत" project aims to address the challenges faced during elections and provide a solution to improve the voting process, it is important to acknowledge and consider the limitations that may exist. Here are some potential limitations to be aware of:

### 1. Accessibility and Digital Divide:

- The project assumes access to digital technology and internet connectivity, which may not be readily available to all segments of the population, particularly in rural or underprivileged areas.
- The digital divide could create disparities in participation and limit the reach and effectiveness of the "जनमत" platform.

### 2. Technological Infrastructure:

- The success of the project relies on the availability of reliable and robust technological infrastructure, including servers, databases, and network systems.
- Insufficient infrastructure or technical issues could impact the performance and availability of the platform, causing disruptions during the election process.

### 3. Data Security and Privacy:

- Ensuring the security and privacy of voter and candidate information is crucial in any election-related project.
- Adequate measures must be implemented to protect personal data from unauthorized access, hacking, or data breaches.
- Compliance with data protection laws and regulations, such as GDPR (General Data Protection Regulation), should be a priority.

### 4. Trust and Acceptance:

- Building trust and acceptance among the stakeholders, including voters, candidates, and electoral authorities, is essential for the success of the project.
- Overcoming skepticism or resistance to digital voting systems and ensuring transparency in the process may pose challenges.

### 5. Manipulation and Fraud:

- Despite efforts to enhance the security and integrity of the system, there is always a risk of manipulation or fraud, both from external actors and internal sources.
- Safeguards and mechanisms should be implemented to detect and prevent unauthorized access, tampering, or manipulation of data.

### 6. Legal and Regulatory Considerations:

- The project needs to comply with existing election laws, regulations, and guidelines of the specific country or region where it is implemented.
- Adhering to legal requirements related to voting, campaign finance, and candidate eligibility is crucial to ensure the project's legality and credibility.

### 7. User Training and Adoption:

- Proper training and education programs must be conducted to familiarize users with the "जनमत" platform and its features.
- Resistance to change or a lack of technological literacy among certain user groups could impact the adoption and utilization of the system.

### 8. Scalability:

- As the project aims to cater to a large number of voters and candidates, scalability becomes a critical factor.
- Ensuring that the system can handle an increasing number of users, concurrent transactions, and data volume is essential for a smooth election process.

It is important to address these limitations during the development and implementation of the "जनमत" project to mitigate risks and ensure its effectiveness. Regular evaluation and adaptation of the system based on user feedback and emerging technologies can help overcome some of these limitations over time.

## CHAPTER 10

### FUTURE SCOPE

The "जनमत" project lays a strong foundation for revolutionizing the voting process and addressing the challenges faced during elections. While the current implementation is focused on enhancing accessibility, transparency, and efficiency, there are several potential areas for future development and expansion. Here are some key aspects to consider for the future scope of the project:

#### 1. Mobile Application:

- Developing a mobile application for the "जनमत" project can significantly enhance accessibility and convenience for users.
- A mobile app would allow voters to participate in elections directly from their smartphones, increasing participation rates and reaching a wider audience.

#### 2. Advanced Security Measures:

- Continuously improving the security measures to ensure the integrity and confidentiality of voter data is essential.
- Exploring advanced techniques such as blockchain technology for secure and tamper-proof storage of voting records can further enhance the trustworthiness of the system.

#### 4. Expansion to Multiple Platforms and Devices:

- Extending the "जनमत" project to support various platforms and devices, including tablets, kiosks, and web browsers, can accommodate different user preferences and increase participation opportunities.

#### 5. Analytics and Reporting:

- Implementing advanced analytics and reporting features can provide valuable insights into voting patterns, demographic trends, and candidate performance.
- Such data can assist election authorities in making informed decisions and improving future electoral processes.

#### 6. Integration with Election Management Systems:

- Integrating the "जनमत" project with existing election management systems can streamline administrative processes, such as candidate registration, ballot preparation, and result consolidation.
- This integration can ensure seamless coordination between the digital voting platform and the overall election infrastructure.

#### 7. Expansion to International Elections:

- Expanding the project beyond national elections to support regional or international elections can provide a standardized and secure platform for democratic processes worldwide.
- Collaboration with international organizations and governments can facilitate the adoption of the "जनमत" system in diverse electoral contexts.

#### 8. Continuous User Feedback and Iterative Enhancements:

- Establishing channels for users to provide feedback and suggestions for improvement is crucial.
- Regularly updating and enhancing the system based on user input will ensure its relevance and effectiveness in meeting evolving needs.

#### 9. Collaboration with Election Authorities and NGOs:

- Collaborating with election authorities, non-governmental organizations (NGOs), and research institutions can facilitate knowledge sharing and expertise in election management and technology.
- This collaboration can help refine the "जनमत" project and adapt it to specific regional or cultural requirements.

#### 10. International Standards Compliance:

- Adhering to international standards and guidelines for secure and transparent elections, such as those outlined by the International Foundation for Electoral Systems (IFES) or the United Nations Electoral Assistance Division, can enhance the credibility and acceptance of the "जनमत" project at a global level.

In conclusion, the "जनमत" project has immense potential for future development and expansion. By embracing emerging technologies, collaborating with relevant stakeholders, and continuously iterating based on user feedback, the project can evolve into a comprehensive and widely adopted digital voting solution. The future scope of the project lies in its ability to adapt to changing needs and challenges, ultimately shaping the future of democratic processes worldwide.

## CHAPTER 11 CONCLUSION

The "জনমত" project has been developed with the goal of revolutionizing the voting process and addressing the challenges encountered during elections. By leveraging technology and designing user-friendly features, the project aims to enhance accessibility, transparency, and efficiency in the electoral system, ultimately strengthening democracy.

Throughout the project, significant progress has been made in various areas. The software has been meticulously designed and implemented, incorporating functionalities such as user registration, candidate management, voting processes, result calculation, and reporting. These components have been carefully structured to ensure a seamless user experience and efficient data management.

Furthermore, a robust database connection has been established, enabling secure storage and retrieval of crucial information related to users, candidates, and election results. By adhering to best practices in data security, the project ensures the integrity and availability of data, maintaining the trust of users and stakeholders.

System testing has played a vital role in ensuring the reliability and functionality of the "জনমত" project. Extensive testing, including the development and execution of comprehensive test cases, error handling, and performance evaluation, has been conducted. The project team has diligently addressed any identified issues or limitations, ensuring a stable and efficient system.

While the "জনমত" project shows great promise, it is important to acknowledge and address the limitations that may arise. Challenges such as accessibility and the digital divide, technological infrastructure requirements, data security and privacy concerns, trust and acceptance among stakeholders, manipulation and fraud risks, legal and regulatory considerations, user training, and scalability need to be proactively tackled to ensure the project's success.

In conclusion, the "জনমত" project marks a significant milestone in advancing the voting process and bolstering democracy. By harnessing the power of technology and adopting user-centric design principles, the project strives to enhance the electoral experience, promote inclusivity, and safeguard the democratic system. Continuous efforts to overcome limitations, coupled with ongoing monitoring, evaluation, and adaptation, will be crucial in achieving the project's objectives and making a lasting positive impact on the electoral landscape. The "জনমত" project stands as a testament to the power of innovation in shaping a more robust and participatory democracy for all.

## CHAPTER 12 REFERENCES

- [1]. <https://www.irjet.net/archives/V4/i12/IRJET-V4I12256.pdf>, Rajesh M. Ghadi, Priyanka S. Shelar, Online Voting System, International Research Journal of Engineering and Technology (IRJET), 2017
- [2]. <https://www.corteidh.or.cr/tabcas/28047.pdf>, Introducing Electronic Voting: Essential Considerations, International Institute for Democracy and Electoral Assistance, 2011
- [3]. <https://www.legalserviceindia.com/legal/article-7582-all-about-vote-and-voting-rights-in-india.html>, Kriti Gupta, All About Vote And Voting Rights In India, Legal Service India, E-journal
- [4]. <https://www.worldbank.org/en/about/leadership/votingpowers>
- [5]. <https://openai.com/blog/chatgpt>
- [6]. <https://ecisveep.nic.in/voters/how-to-vote/>