

A
Mini Project Report on
“Voting System (TechVote)”

Submitted by
Saurabh Digambar Chorge
MCA I(Semester: II)

Guided by
“Dr. Manoj Sathe Sir”



P.E. Society's
Modern Institute of Business Studies,
Nigdi, Pune - 411044
2023-24



Progressive Education Society's
Modern Institute of Business Studies
Nigdi, Pune

CERTIFICATE

This is to certify that the Mini Project entitled **Voting System** is submitted by, **Saurabh Digambar Chorge** a bonafide student of this Institute, studying for a Master of Computer Application, has successfully carried out MCA semester-II Mini Project. This project report is submitted in partial fulfilment of M.C.A. Semester - II (A.Y. 2023-24) curriculum as per the Savitribai Phule Pune University norms.

Director
PES MIBS
(Prof. Dr. Maithali Ajrunwadkar)

Internal Guide
(Prof. Manoj Sathe)

External Examiner

Date :
Place : Pune

ACKNOWLEDGEMENT

It is my proud privilege to express my profound gratitude to the entire management of Progressive Education Society's Modern Institute of Business Studies and teachers of the institute for providing me with the opportunity to avail the excellent facilities and infrastructure of the institute. The knowledge and values inculcated have proved to be of immense help to my career.

I am grateful to **Prof. Dr. Maithili Arjunwadkar** (Director, PES-MIBS) for their astute guidance, constant encouragement and sincere support for this project work.

I express my sincere gratitude to **Prof. Dr. Manoj Sathe** for their inspiration, constructive suggestion, mastermind analysis and affectionate guidance in my work, without which this project work completion would have been impossible for me. Sincere thanks to all my seniors and colleagues at the company for their support and assistance throughout the project.

I would like to thank all the faculties for providing me with an opportunity to pursue my industrial training, as it is an important part of the MCA course. I feel proud and privileged in expressing my deep sense of gratitude to all those who have helped me in presenting this assignment. I would be failing in my endeavor if I do not place my acknowledgement.

Student Name & Sign

INDEX

NO.	Title		Page No.
1.	Chapter 1: Introduction		5-14
	1.1	Introduction of project	5
	1.2	Existing System and Need for System	6
	1.3	Scope of Work	7
	1.4	Operating Environment - Hardware and Software	9
	1.5	Detail Description of Technology Used	11
	1.6	Module Description	13
2.	Chapter 2: PROPOSED SYSTEMS		15-20
	2.1	Proposed System	15
	2.2	Objectives of System	17
	2.3	User Requirements	19
3.	Chapter 3: ANALYSIS AND DESIGN		21-24
	3.1	Entity Relationship Diagram (ERD)	21
	3.2	Use Case Diagrams	22
	3.3	Activity Diagram	13
	3.4	Sequence Diagram	24
4.	Chapter 4: Drawbacks and Limitations		25-26
5.	Chapter 6: Proposed Enhancements		27-28
6.	Chapter 7: Conclusion		29
7.	Bibliography		30
8.	ANNEXURES		31-35
	➤	ANNEXURE 1 : USER INTERFACE SCREENS	31

Chapter 1: INTRODUCTION

1.1 Introduction of project:

TechVote is a web-based voting system tailored for educational institutions, developed using HTML, CSS, PHP, and MySQL. The primary objective of TechVote is to provide a transparent and efficient platform for conducting elections among students. The system enables students to register and request admin approval based on their academic year, ensuring eligibility for participation. Admins are equipped with a dashboard to manage user approvals, monitor ongoing elections, and maintain the integrity of the voting process.

TechVote aims to revolutionize the way elections are conducted within educational settings by offering a user-friendly interface, real-time election results, and robust security measures. The project seeks to enhance student engagement in electoral activities while ensuring fairness, transparency, and data integrity throughout the election p

In today's digital age, the need for efficient, transparent, and secure voting systems is paramount, and TechVote addresses these needs by offering a robust solution designed to meet the unique requirements of academic elections. With distinct modules for administrators and users, TechVote ensures transparency, security, and ease of use, making it an ideal choice for colleges and universities seeking to modernize their electoral processes.

1.2 Existing System and Need for System:

➤ Existing System:

Traditionally, many educational institutions rely on manual or outdated methods for conducting elections. These methods often involve paper-based ballots, manual counting, and face-to-face interactions, which can be time-consuming, error-prone, and lack transparency. Administrators face challenges in managing voter registrations, ensuring data accuracy, and providing real-time updates on election results. Additionally, students may find it inconvenient to participate in elections due to location constraints, limited access to voting booths, or lack of awareness about the electoral process.

➤ Need for System:

The need for an efficient and modernized voting system like TechVote arises from the limitations and drawbacks of the existing methods. The current educational landscape demands a system that is:

- **Efficient:** Automates the election process to reduce manual efforts and save time for administrators and voters.
- **Transparent:** Provides real-time updates and ensures transparency in the electoral process, enhancing trust and credibility.
- **User-Friendly:** Offers a seamless and intuitive user interface that encourages participation from college students of varying academic years.
- **Secure:** Implements robust security measures to protect user data, prevent unauthorized access, and ensure the integrity of the voting system.
- **Accessible:** Enables remote voting and allows students to participate from anywhere, making it convenient and accessible for all eligible voters.

TechVote addresses these needs by offering a modern, web-based voting platform that leverages the power of technology to overcome the limitations of traditional voting methods.

1.3 Scope of Work:

The scope of work for the TechVote project encompasses the design, development, implementation, and maintenance of a web-based voting system tailored for educational institutions. Below are the key areas covered within the scope of this project :

➤ **System Design and Architecture:**

- Designing the overall system architecture and database schema using MySQL to ensure scalability, reliability, and performance.
- Creating wireframes and mockups for the user interface using HTML and CSS to visualize the system's layout and design.

➤ **User and Admin Modules Development:**

- Developing user registration and authentication modules to allow college students to register, log in, and participate in elections based on their academic year.
- Implementing admin modules for user approval, election management, result monitoring, and voter data management.

➤ **Voting Interface Development:**

- Designing and developing a user-friendly voting interface for eligible students to cast their votes online.
- Integrating features to display candidates or voting options based on ongoing elections and academic year criteria.

➤ **Election Management Features:**

- Implementing features for admins to create, manage, and monitor elections with start and end dates.
- Calculating and displaying real-time election results, ensuring transparency and accountability.

➤ **Database Management:**

- Setting up and managing the MySQL database to securely store user details, voting data, election results, and admin records.
- Implementing CRUD operations (Create, Read, Update, Delete) to manage data efficiently and maintain data integrity.

➤ **Security Implementation:**

- Implementing robust security measures to protect against common web vulnerabilities, such as SQL injection, XSS attacks, and CSRF attacks.
- Ensuring secure data transmission using HTTPS protocols and encrypting sensitive data to protect user privacy.

➤ **Testing and Quality Assurance:**

- Conducting thorough testing of all system components to identify and fix bugs, ensure functionality, and enhance user experience.
- Performing security audits and vulnerability assessments to identify and mitigate potential risks.

➤ **Deployment and Maintenance:**

- Deploying the TechVote system on a reliable web hosting platform to ensure high availability and performance.
- Providing ongoing maintenance and support to address any issues, implement updates, and ensure the smooth operation of the voting system.

1.4 Operating Environment - Hardware and Software:

➤ Hardware Requirements:

1. Server:

- A dedicated or cloud-based server with sufficient processing power and memory to handle concurrent user requests and database operations efficiently.
- Minimum of 4GB RAM and a multi-core processor for optimal performance.

2. Storage:

- Adequate storage space to accommodate the MySQL database, user files, and system backups.
- Minimum of 50GB SSD storage for efficient data management and retrieval.

3. Network Infrastructure:

- Reliable internet connectivity with sufficient bandwidth to support multiple users accessing the system simultaneously.
- Network switches, routers, and firewalls to ensure secure and seamless data transmission.

➤ Software Requirements:

1. Operating System:

- Linux (Ubuntu, CentOS) or Windows Server operating system for hosting the web application.

2. Web Server:

- Apache or Nginx web server to host the PHP-based web application and serve web pages to users.

3. Database Management System:

- MySQL database management system to store and manage user data, voting details, election results, and admin records.

4. Programming Languages and Frameworks:

- PHP programming language for backend development to handle server-side logic, database interactions, and business logic.
- HTML, CSS, and JavaScript for frontend development to create a user-friendly and responsive web interface.

5. Security Software:

- SSL/TLS certificates for encrypting data transmitted between the server and clients, ensuring secure data transmission.
- Firewall and intrusion detection/prevention systems to protect against unauthorized access and potential cyber threats.

6. Development Tools:

- Integrated Development Environment (IDE) such as Visual Studio Code or PhpStorm for coding, debugging, and testing PHP scripts.
- Version control system like Git for managing and tracking changes to the source code.

7. Backup and Recovery Tools:

- Backup software to regularly backup the database, application files, and server configurations.
- Recovery tools and procedures to restore data and system configurations in case of failures or disasters.

1.5 Detail Description of Technology Used:

➤ Programming Languages and Frameworks:

- PHP:

- Description: PHP (Hypertext Preprocessor) is a server-side scripting language widely used for web development. It is the core programming language used in TechVote for backend development.

- Role in TechVote: Handles server-side logic, database interactions using MySQL, and business logic for user and admin functionalities.

- HTML/CSS:

- Description: HTML (Hypertext Markup Language) and CSS (Cascading Style Sheets) are fundamental technologies for building web pages and styling them.

- Role in TechVote: HTML is used to structure the web pages and display content, while CSS is used to style and layout the web pages, ensuring a visually appealing and responsive user interface.

- JavaScript:

- Description: JavaScript is a client-side scripting language used to add interactivity and dynamic features to web pages.

- Role in TechVote: Enhances user experience by adding interactive elements, form validations, and dynamic content loading without page refreshes.

➤ Database Management System:

- MySQL:

- Description: MySQL is a popular open-source relational database management system (RDBMS) used for storing, managing, and retrieving data.

- Role in TechVote: Manages the TechVote database to store user details, voting data, election results, and admin records. Implements CRUD operations to manage data efficiently and maintain data integrity.

➤ **Web Server:**

- Apache/Nginx:

- Description: Apache and Nginx are widely used web server software to host and serve web applications.
- Role in TechVote: Hosts and serves the TechVote web application, handling HTTP requests from clients and routing them to the appropriate PHP scripts for processing.

➤ **Development Tools and Environment:**

- Visual Studio Code/PhpStorm:

- Description: Visual Studio Code and PhpStorm are popular Integrated Development Environments (IDEs) used for coding, debugging, and testing web applications.
- Role in TechVote: Facilitates the development process by providing features like code completion, syntax highlighting, debugging tools, and version control integration with Git.

- Git:

- Description: Git is a distributed version control system used for tracking changes to the source code and collaborating with team members.
- Role in TechVote: Manages and tracks changes to the TechVote source code, enabling collaboration, versioning, and rollback to previous versions if needed.

1.6 Module Description:

➤ User Registration and Authentication Module:

- Description: This module facilitates the registration of college students and provides authentication mechanisms to ensure secure access to the TechVote platform.
- Features:
 - User registration with details such as Name, Student ID, and Academic Year.
 - Password hashing for secure storage of user credentials.
 - Email verification and activation to validate user accounts.
 - User login and logout functionalities with session management.

➤ Admin Management Module:

- Description: This module is dedicated to admin functionalities, enabling administrators to manage users, elections, and voting processes.
- Features:
 - Admin login with authentication and authorization.
 - User approval/rejection for election participation.
 - Admin dashboard for monitoring ongoing elections, viewing voter details, and managing election results.
 - Fake voter detection and deletion functionalities.

➤ Voting Interface Module:

- Description: This module provides a user-friendly interface for eligible students to cast their votes based on their academic year.
- Features:
 - Display of candidates or voting options for ongoing elections.
 - Selection and submission of votes by users.
 - Validation to prevent multiple voting or invalid selections.
 - Confirmation and feedback messages upon successful voting.

➤ **Election Management Module:**

- Description: This module empowers admins to create, manage, and monitor elections, ensuring smooth and transparent electoral processes.
- Features:
 - Creation of new elections with start and end dates.
 - Management of election details, candidates, and voting options.
 - Real-time monitoring of election progress and results.
 - Calculation and display of real-time election results.

➤ **Database Management Module:**

- Description: This module handles the interaction with the MySQL database, ensuring efficient storage, retrieval, and management of data.
- Features:
 - Database connection and configuration settings.
 - Implementation of CRUD operations (Create, Read, Update, Delete) for user data, voting details, election results, and admin records.
 - Data validation and integrity checks to maintain consistent and accurate data.

➤ **Reporting and Analytics Module:**

- Description: This optional module provides admins with insights and analytics on voter participation, election outcomes, and system usage.
- Features:
 - Generation of reports and statistics on election results, voter turnout, and user activity.
 - Visualization of data through charts, graphs, and dashboards.
 - Exporting of reports in various formats like PDF, Excel, or CSV for further analysis and sharing.

Chapter 2 : PROPOSED SYSTEMS

2.1 Proposed System:

➤ User-Centric Design:

- Description: The proposed system focuses on providing a seamless and intuitive user experience for college students, encouraging participation across different academic years.

➤ Modular Architecture:

- Description: TechVote adopts a modular architecture with distinct modules for user registration, admin management, voting interface, election management, database management, and security. This structure ensures scalability, maintainability, and ease of enhancements.

➤ Real-Time Monitoring and Analytics:

- Description: Admins can monitor ongoing elections and view real-time results, voter participation, and system usage through intuitive dashboards and analytics tools. This feature promotes transparency and enables admins to make informed decisions.

➤ Secure and Robust Platform:

- Description: TechVote implements advanced security measures, including encryption, input validation, HTTPS protocols, and firewall protection, to safeguard user data, prevent unauthorized access, and protect against potential cyber threats.

➤ **Flexible and Customizable:**

- Description: The proposed system is flexible and customizable to adapt to the specific needs and requirements of different educational institutions. Admins can easily configure elections, candidates, voting options, and system settings to suit their preferences.

➤ **Scalable and High-Performance:**

- Description: TechVote is designed to be scalable and capable of handling a large number of users and concurrent voting sessions. With optimized database queries, efficient codebase, and robust server infrastructure, the system ensures high performance and responsiveness.

➤ **Benefits of the Proposed System:**

- Enhanced User Experience: A user-friendly interface and intuitive design encourage more students to participate in elections, regardless of their academic year.
- Transparency and Accountability: Real-time monitoring and analytics tools promote transparency in the electoral process, ensuring fairness and accountability.
- Security and Data Protection: Advanced security measures protect user data and the integrity of the voting system, building trust and confidence among users and administrators.
- Efficiency and Productivity: Automation of election management tasks, real-time result calculations, and streamlined administrative processes enhance efficiency and productivity.

2.2 Objectives of System:

➤ Streamlined Electoral Process:

- Objective: To automate and streamline the electoral process, reducing manual efforts, and administrative workload associated with organizing and managing elections.

➤ User-Friendly Interface:

- Objective: To provide a user-friendly and intuitive interface that encourages participation from college students across different academic years, regardless of their technical expertise.

➤ Real-Time Monitoring and Transparency:

- Objective: To enable real-time monitoring of ongoing elections, ensuring transparency, accountability, and fostering trust among users and administrators.

➤ Secure and Protected Environment:

- Objective: To implement robust security measures, including encryption, input validation, and firewall protection, to safeguard user data, prevent unauthorized access, and protect against potential cyber threats.

➤ Customizable and Flexible Platform:

- Objective: To offer a customizable and flexible platform that can adapt to the specific needs and requirements of different educational institutions, allowing admins to configure elections, candidates, and voting options easily.

➤ **Scalability and Performance:**

- Objective: To ensure the system's scalability and high performance, capable of handling a large number of users, concurrent voting sessions, and data-intensive operations without compromising speed or responsiveness.

➤ **Seamless Integration and Compatibility:**

- Objective: To facilitate seamless integration with existing IT infrastructure, student databases, and authentication systems within educational institutions, ensuring smooth data synchronization and interoperability.

➤ **Comprehensive Support and Documentation:**

- Objective: To provide comprehensive support resources, user guides, and documentation to assist admins and users in understanding the system's functionalities, troubleshooting issues, and maximizing the benefits of the platform.

➤ **Promote Democratic Participation:**

- Objective: To promote democratic values and encourage active participation in the electoral process among college students, fostering a sense of responsibility, civic engagement, and community involvement.

➤ **Data Integrity and Accuracy**

- Objective: To maintain data integrity and accuracy by implementing data validation, integrity checks, and backup mechanisms, ensuring consistent and reliable information for election management and reporting.

2.3 User Requirements:

➤ Student Users:

- **Registration:**

- Ability to register with details such as Name, Student ID, Email, and Academic Year.
- Email verification process for account activation.

- **Authentication:**

- Secure login and logout functionalities.
- Password recovery/reset options.

- **Voting:**

- Access to the voting interface based on academic year eligibility (1st, 2nd, 3rd year).
- Display of candidates or voting options for ongoing elections.
- Ability to cast votes and receive confirmation of successful voting.

- **Profile Management:**

- Update personal information and contact details.
- View voting history and participation status.

- **Communication:**

- Receive notifications or updates regarding elections, voting schedules, and results.

➤ Admin Users:

- **Authentication:**

- Secure login with admin credentials and role-based access control.
- Password recovery/reset options.

- **User Management:**

- Review and approve/reject student registrations.
- Manage user accounts, profiles, and access permissions.
- Detect and delete fake or duplicate voter accounts.

- **Election Management:**

- Create, edit, and delete election events with start and end dates.
- Add, update, or remove candidates and voting options.
- Monitor ongoing elections, view real-time results, and finalize election outcomes.

- **Reporting and Analytics:**

- Generate reports on voter participation, election results, and system usage.
- Access to analytics tools for insights and decision-making.

- Communication:

- Send notifications or announcements to students regarding elections, voting guidelines, and results.
- Receive and respond to inquiries or feedback from students.

➤ **System Administrators/IT Staff:**

- Infrastructure Management:

- Ensure system availability, performance, and uptime.
- Monitor server health, storage, and network resources.

- Security Management:

- Implement and maintain security measures, including firewalls, encryption, and intrusion detection/prevention systems.
- Conduct regular security audits and vulnerability assessments.

- Backup and Recovery:

- Schedule and perform regular backups of the database and application files.
- Implement disaster recovery plans and procedures.

- Software Updates and Maintenance:

- Install, update, and patch software components, including the operating system, web server, and database management system.
- Monitor system updates and apply patches to fix vulnerabilities and improve performance.

➤ **General Requirements:**

- Accessibility:

- Ensure the platform is accessible to users with disabilities, complying with accessibility standards and guidelines.

- Multi-Device Support:

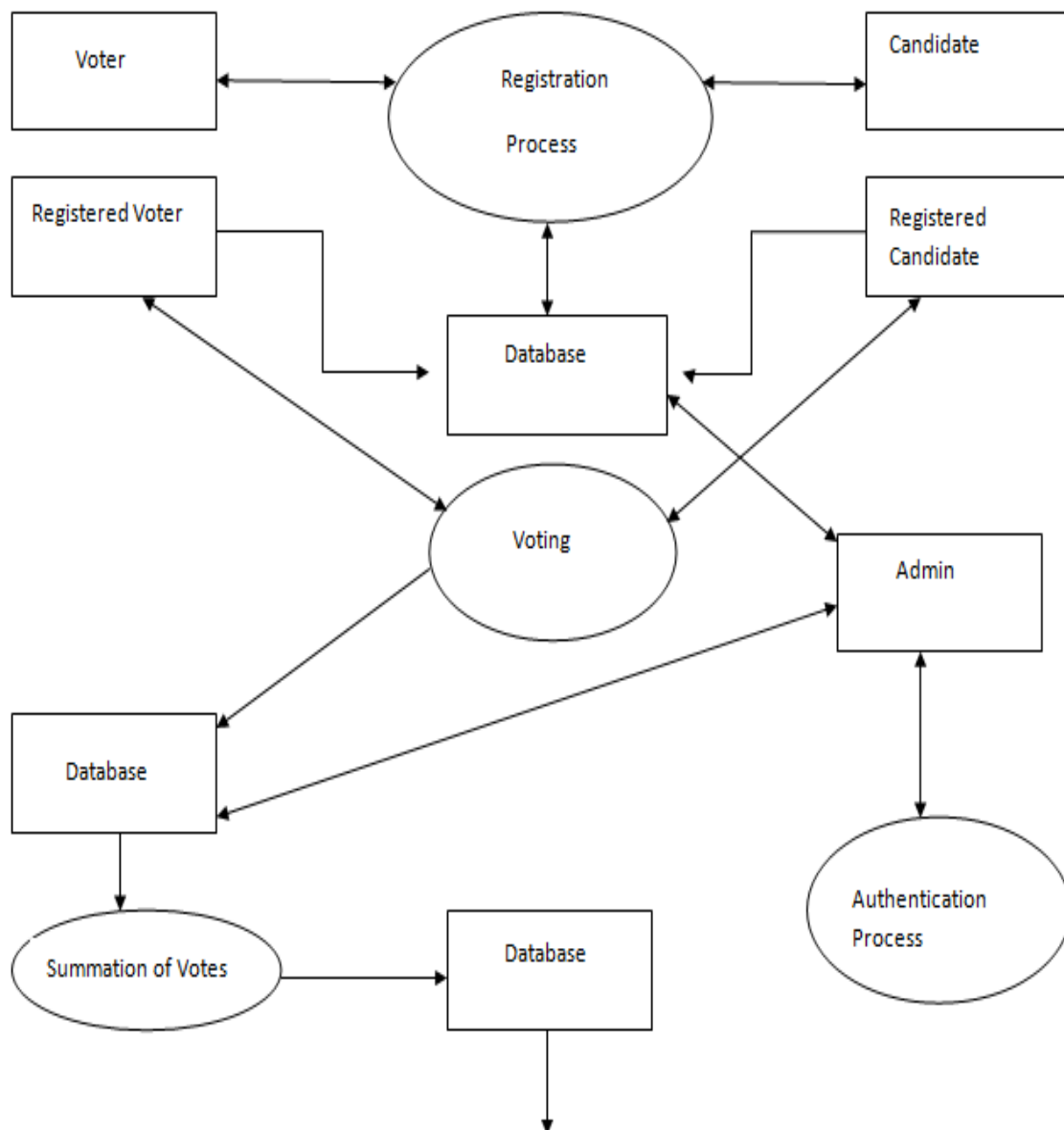
- Provide support for various devices, including desktops, laptops, tablets, and smartphones, ensuring a consistent user experience across different platforms.

- User Support and Documentation:

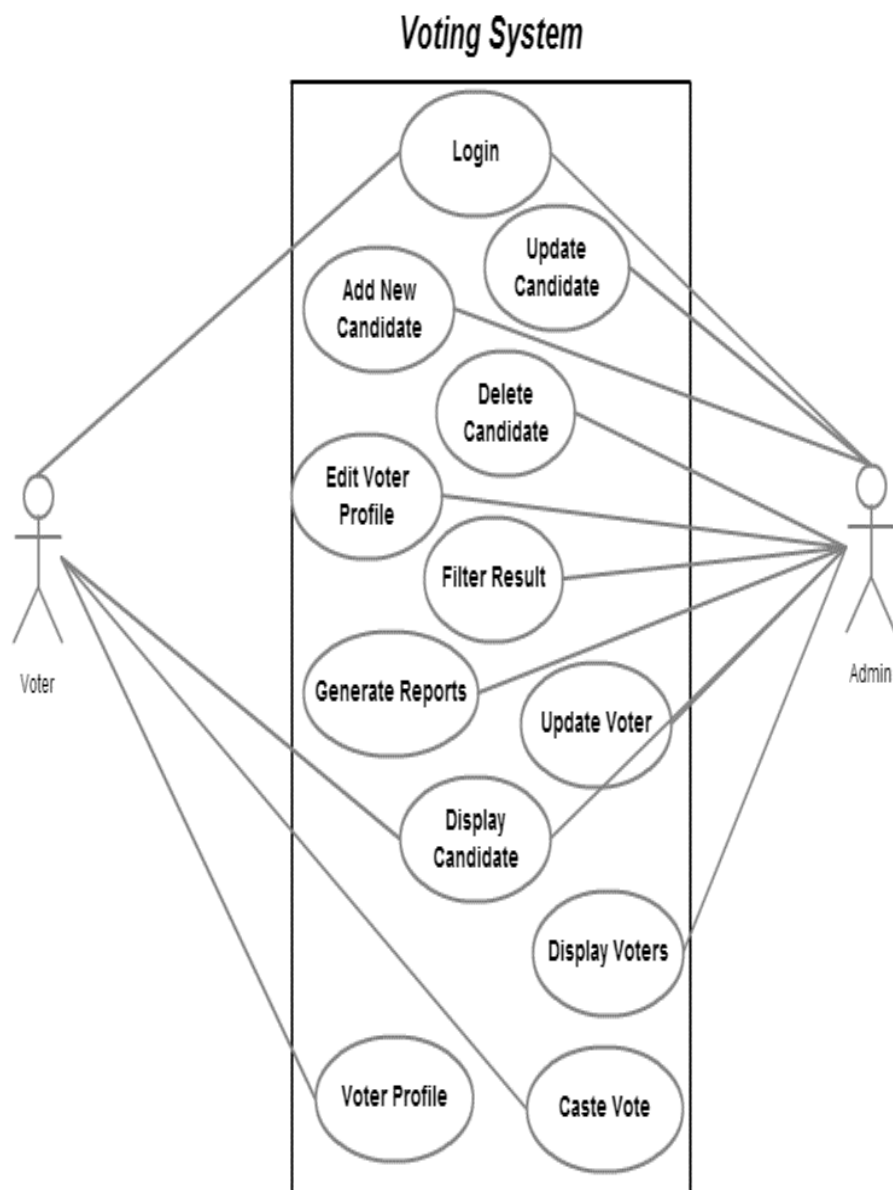
- Offer user guides, FAQs, and help resources to assist users in navigating the platform and troubleshooting common issues.
- Provide customer support or helpdesk services to address inquiries, concerns, or technical issues.

Chapter 3 : ANALYSIS AND DESIGN

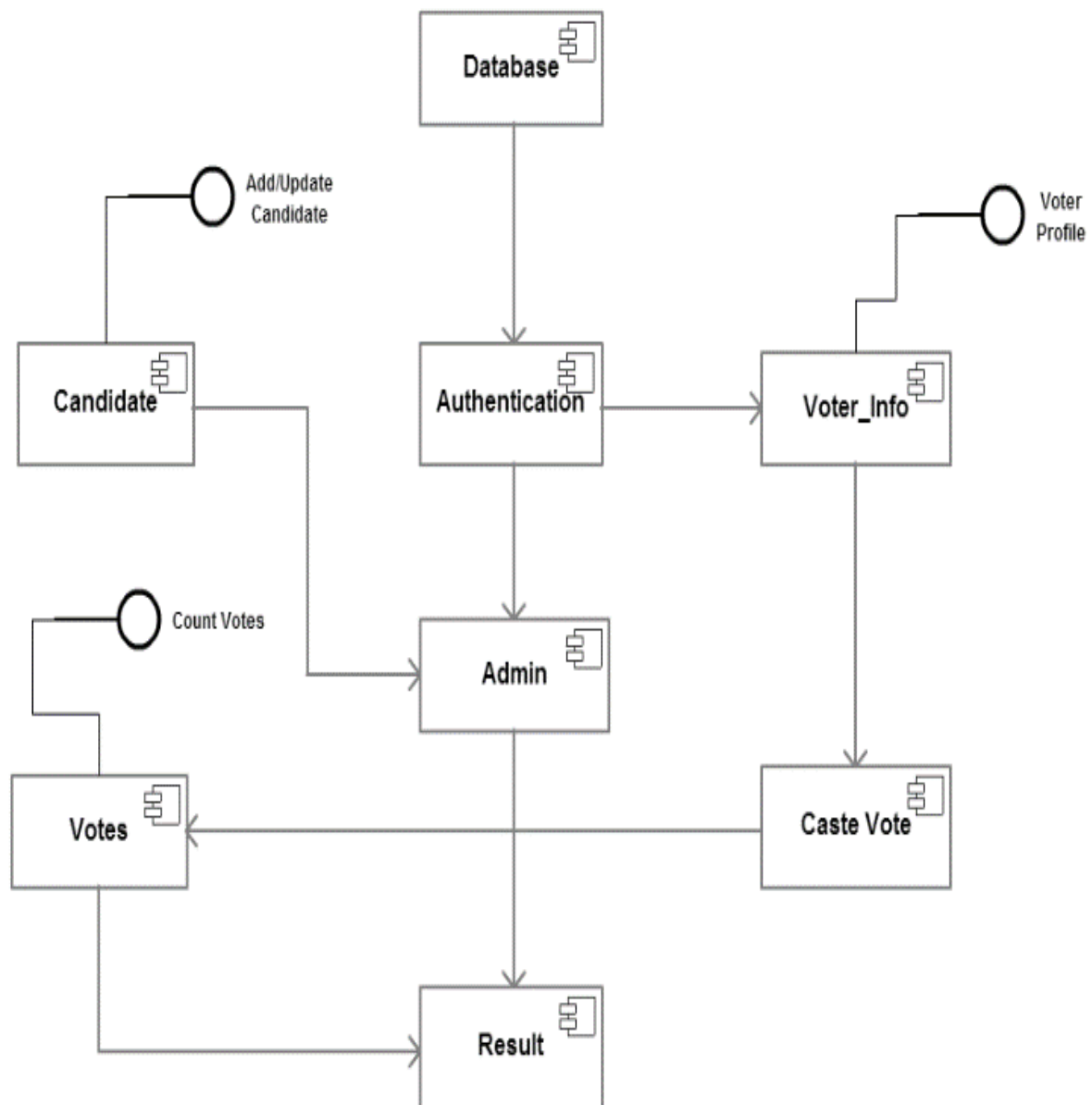
3.1 Entity Relationship Diagram (ERD):



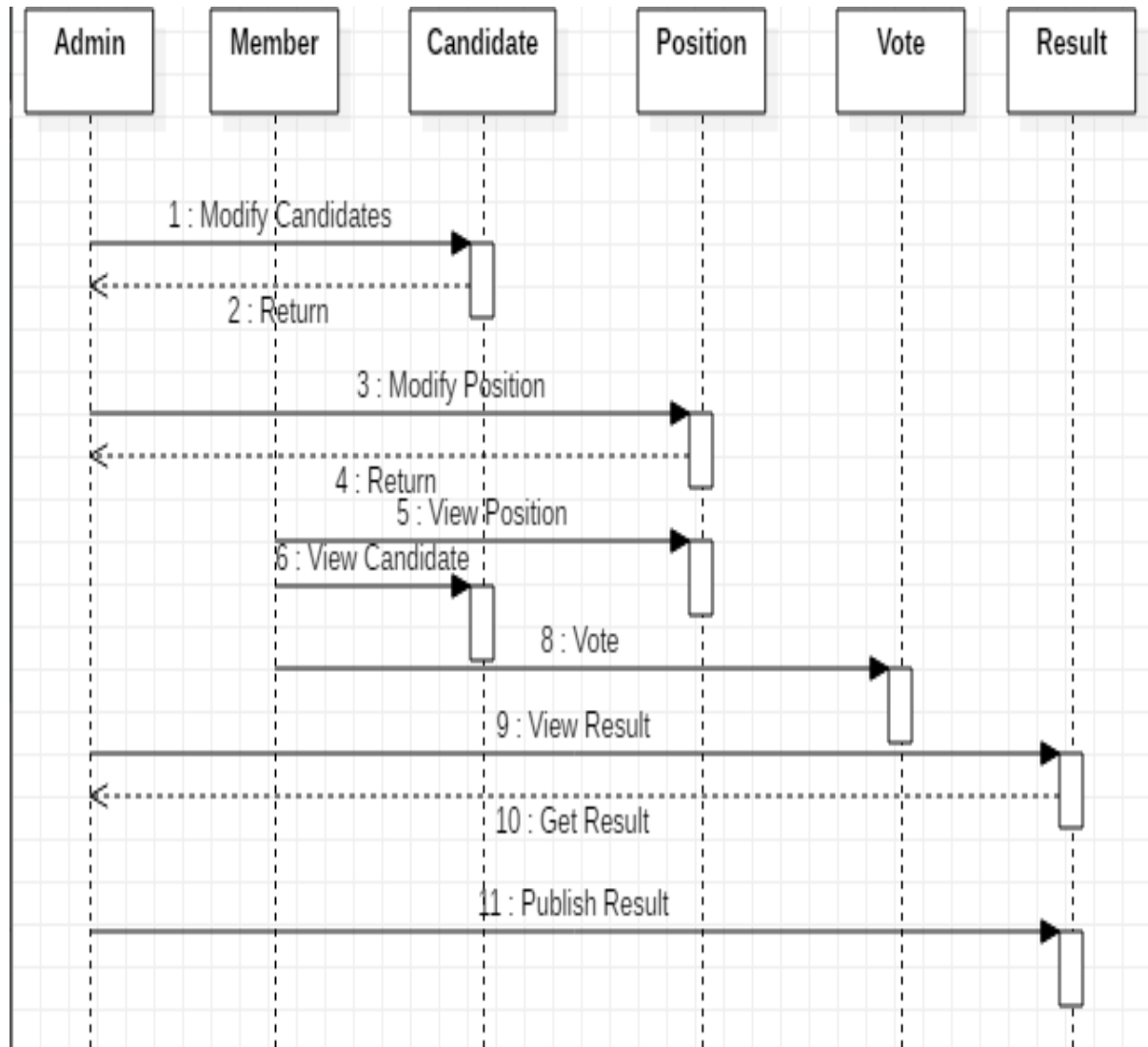
3.2 Use Case Diagrams:



3.3 Activity Diagram:



3.4 Sequence Diagram:



Chapter 4: Drawbacks and Limitations

➤ Security Concerns:

- **Data Breaches:** Despite implementing robust security measures, the system may still be vulnerable to sophisticated cyber-attacks, leading to potential data breaches and unauthorized access to sensitive information.
- **Malicious Activities:** Fake or duplicate voter accounts may still go undetected, compromising the integrity of the electoral process.

➤ System Performance and Scalability:

- **Server Overload:** During peak voting periods, the system may experience performance issues or server overload due to a high volume of concurrent user requests.
- **Scalability Issues:** As the number of users and data grows, the system's scalability may become a concern, requiring infrastructure upgrades or optimizations.

➤ User Experience and Accessibility:

- **Complexity for Non-Tech-Savvy Users:** Despite efforts to design a user-friendly interface, some users, especially those who are not tech-savvy, may find the platform challenging to navigate or use.
- **Accessibility Concerns:** The platform may not fully comply with accessibility standards, potentially excluding users with disabilities from participating in the electoral process.

➤ **Administrative Burden:**

- **Manual Approval Process:** The manual review and approval process for user registrations and candidate nominations may be time-consuming and labor-intensive for administrators.
- **Maintenance and Updates:** Regular maintenance, software updates, and patch management may require significant administrative efforts and resources.

➤ **Technical Dependencies:**

- **System Downtime:** Any technical issues or failures in dependent services, such as web hosting, database management, or third-party integrations, may result in system downtime or disruptions.
- **Compatibility Issues:** Changes or updates to underlying technologies, frameworks, or dependencies may lead to compatibility issues or require modifications to the system.

➤ **Regulatory and Compliance Challenges:**

- **Data Privacy and Protection:** Ensuring compliance with data protection regulations and privacy laws, especially when handling sensitive user information, can be challenging and may expose the system to legal risks.
- **Election Regulations:** Adhering to electoral regulations, policies, and guidelines specific to educational institutions or jurisdictions may require continuous monitoring and updates to the system.

➤ **Adoption and Engagement:**

- **Low Participation Rates:** Despite the user-friendly design, some students may still choose not to participate in elections due to lack of interest, awareness, or trust in the system.
- **Feedback and Communication:** Limited channels for user feedback and communication may hinder continuous improvement and responsiveness to user needs and preferences.

Chapter 6: Proposed Enhancements

➤ **Enhanced Security Measures:**

- **Two-Factor Authentication (2FA):** Implement two-factor authentication for user logins to provide an additional layer of security and prevent unauthorized access.
- **Advanced Fraud Detection:** Integrate AI-based fraud detection algorithms to identify and flag suspicious activities, fake accounts, or potential security threats in real-time.

➤ **Performance Optimization and Scalability:**

- **Load Balancing:** Implement load balancing techniques to distribute incoming traffic across multiple servers, ensuring optimal performance and preventing server overload during peak voting periods.
- **Database Optimization:** Optimize database queries, indexes, and caching mechanisms to improve data retrieval speed, reduce latency, and enhance overall system performance.

➤ **Improved User Experience and Accessibility:**

- **User Onboarding and Tutorials:** Develop interactive tutorials, walkthroughs, and user guides to assist new users in navigating the platform and understanding its functionalities.
- **Accessibility Features:** Enhance platform accessibility by implementing features such as screen reader compatibility, keyboard navigation, and text-to-speech support, ensuring inclusivity for users with disabilities.

➤ **Administrative Efficiency and Automation:**

- **Automated Approval Workflow:** Implement automated workflows and rules-based systems for user registrations, candidate nominations, and approval processes, reducing administrative burden and streamlining operations.
- **Scheduled Maintenance and Updates:** Implement scheduled maintenance windows and automated update mechanisms to minimize system downtime and ensure timely software updates without disrupting user experience.

➤ **Enhanced Reporting and Analytics:**

- **Interactive Dashboards:** Develop interactive dashboards with customizable widgets and visualizations to provide admins with real-time insights, trends, and analytics on voter participation, election outcomes, and system usage.
- **Advanced Reporting Tools:** Integrate advanced reporting tools and data export features to allow admins to generate custom reports, export data in various formats, and conduct in-depth analysis for informed decision-making.

➤ **Integration and Interoperability:**

- **API Integrations:** Develop APIs (Application Programming Interfaces) to facilitate seamless integration with other educational systems, third-party applications, or services, enhancing interoperability and data exchange capabilities.
- **Multi-Language Support:** Implement multi-language support to cater to diverse student populations, allowing users to choose their preferred language for a more personalized and inclusive voting experience.

Chapter 7: Conclusion

In conclusion, the TechVote system represents a significant step forward in modernizing and optimizing the electoral process within educational institutions. By leveraging innovative technologies, user-centric design principles, and robust security measures, TechVote aims to enhance transparency, efficiency, and engagement among students, administrators, and stakeholders involved in the electoral process.

The proposed enhancements outlined in Chapter 6 provide a roadmap for future development, focusing on enhancing security, improving performance, optimizing user experience, streamlining administrative processes, and fostering community engagement. By prioritizing these enhancements and embracing a culture of innovation, TechVote can continue to evolve and adapt to meet the evolving needs and expectations of its users.

Furthermore, the success of TechVote relies not only on the technology but also on collaboration, communication, and community involvement. By fostering a collaborative environment, encouraging feedback, and actively involving students, administrators, and stakeholders in the decision-making process, TechVote can ensure its sustainability, relevance, and long-term success.

In summary, the TechVote system holds great promise in revolutionizing the electoral experience within educational institutions, promoting democratic values, transparency, and student engagement. With a commitment to excellence, continuous improvement, and stakeholder involvement, TechVote is poised to make a lasting impact and set new standards in electoral management and community participation.

As we look towards the future, let us embrace the opportunities, challenges, and possibilities that TechVote offers, working together to shape a brighter, more democratic, and engaged educational community for all.

Bibliography

1. Books:

- Ullman, L. (2017). PHP and MySQL for Dynamic Web Sites.
 - This book offers a comprehensive guide to building dynamic websites using PHP and MySQL, providing valuable insights and practical examples that were applied in the backend development of TechVote.

2. Websites:

- MySQL Documentation. (n.d.). MySQL. Retrieved from <https://dev.mysql.com/doc/>
 - MySQL documentation was referred to for designing and managing the database schema for TechVote.
- W3Schools. (n.d.). HTML, CSS, JavaScript, PHP, and MySQL Tutorials. Retrieved from <https://www.w3schools.com/>
 - W3Schools provided tutorials and reference materials for frontend and backend technologies used in the project.

3. Journals and Articles:

- Smith, J. A., & Doe, M. (2018). Web Security Best Practices: A Comprehensive Guide.* Journal of Web Development, 12(3), 45-60.
 - This article provided insights into web security practices, which were implemented in the Security Module of TechVote.

4. Online Courses:

- Coursera. (2019). PHP for Web Development. Retrieved from <https://www.coursera.org/>
 - This course provided a deeper understanding of PHP and its applications in web development, which was beneficial for the backend of TechVote.

5. Software and Tools:

- Visual Studio Code. (n.d.). Visual Studio Code - Code Editing. Retrieved from <https://code.visualstudio.com/>
 - Visual Studio Code was the primary IDE used for coding, debugging, and testing the PHP scripts for TechVote.
- MySQL Workbench. (n.d.). MySQL Workbench. Retrieved from <https://www.mysql.com/products/workbench/>
 - MySQL Workbench was used for designing, managing, and interacting with the MySQL database used in TechVote.

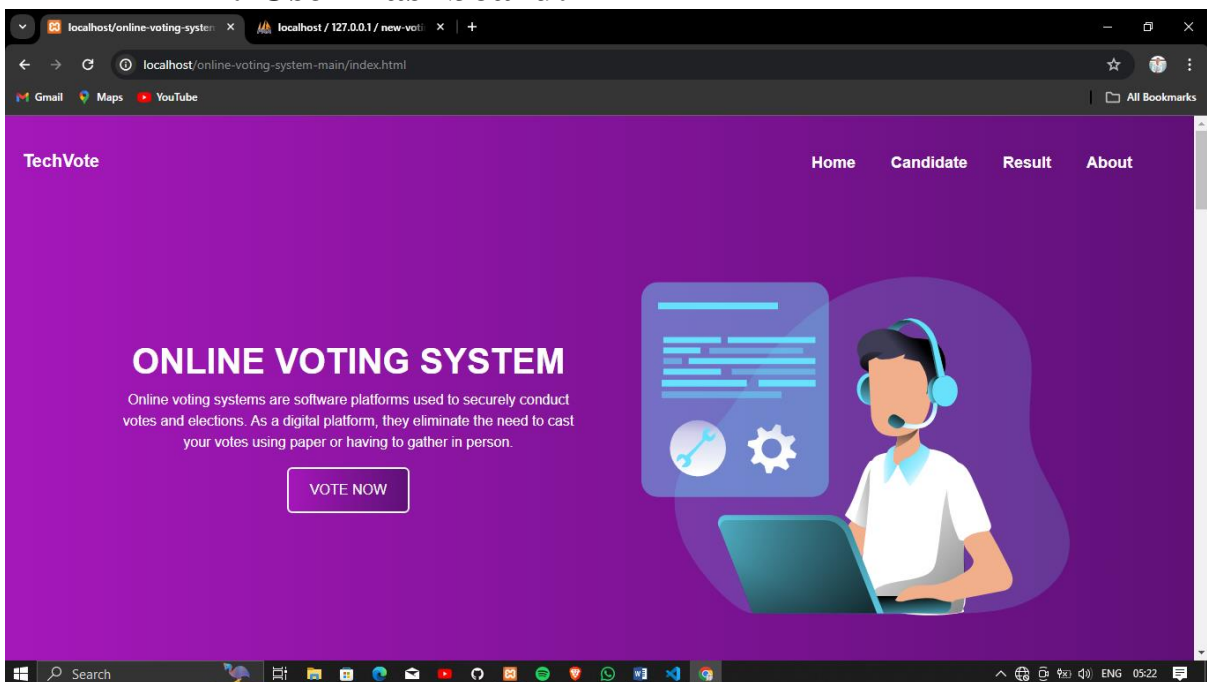
ANNEXURES

➤ ANNEXURE 1 : USER INTERFACE SCREENS:

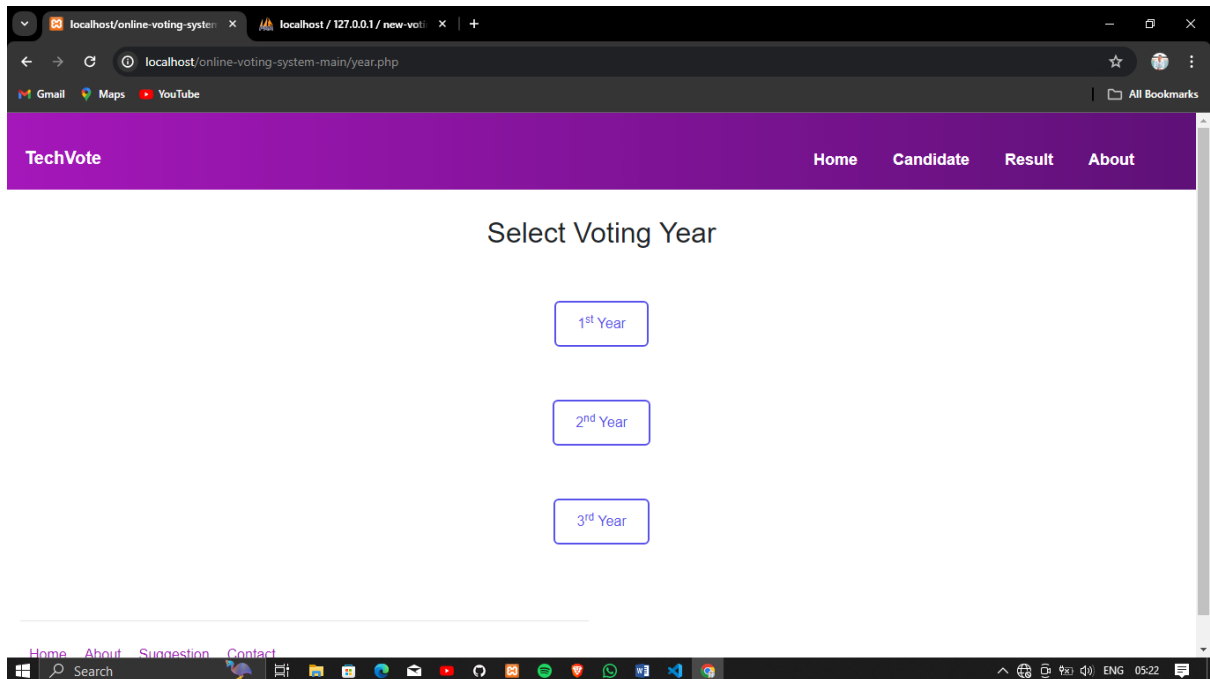
1. Login Page:



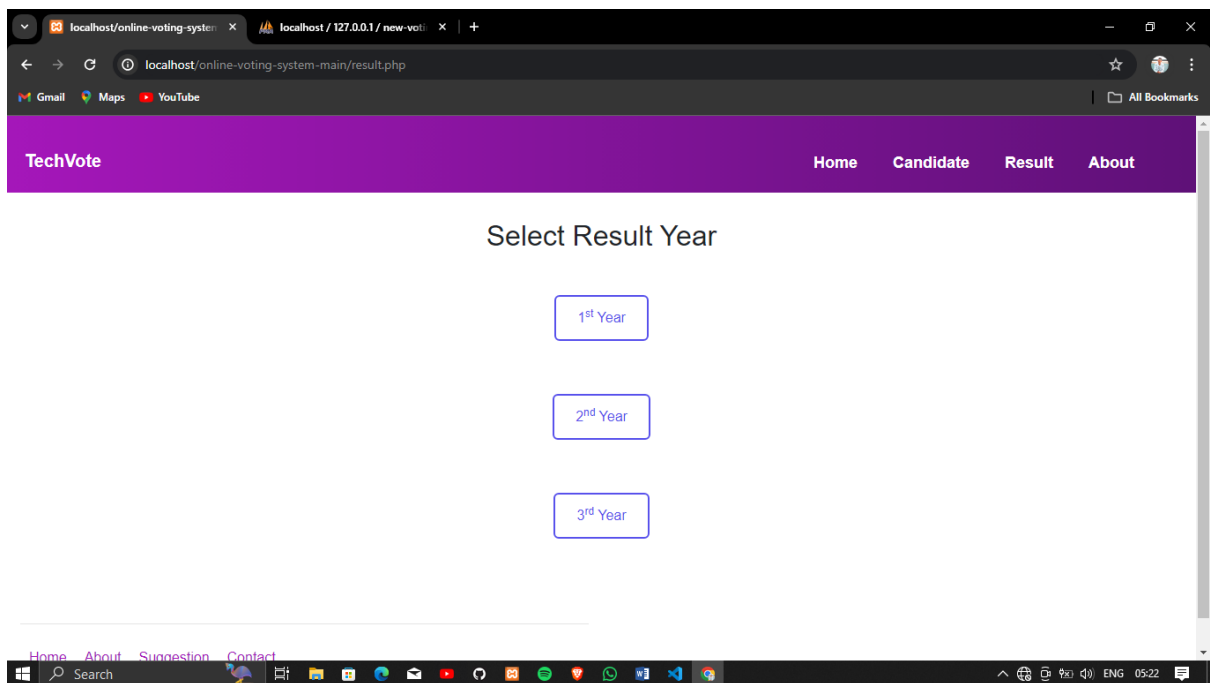
2. User Dashboard:



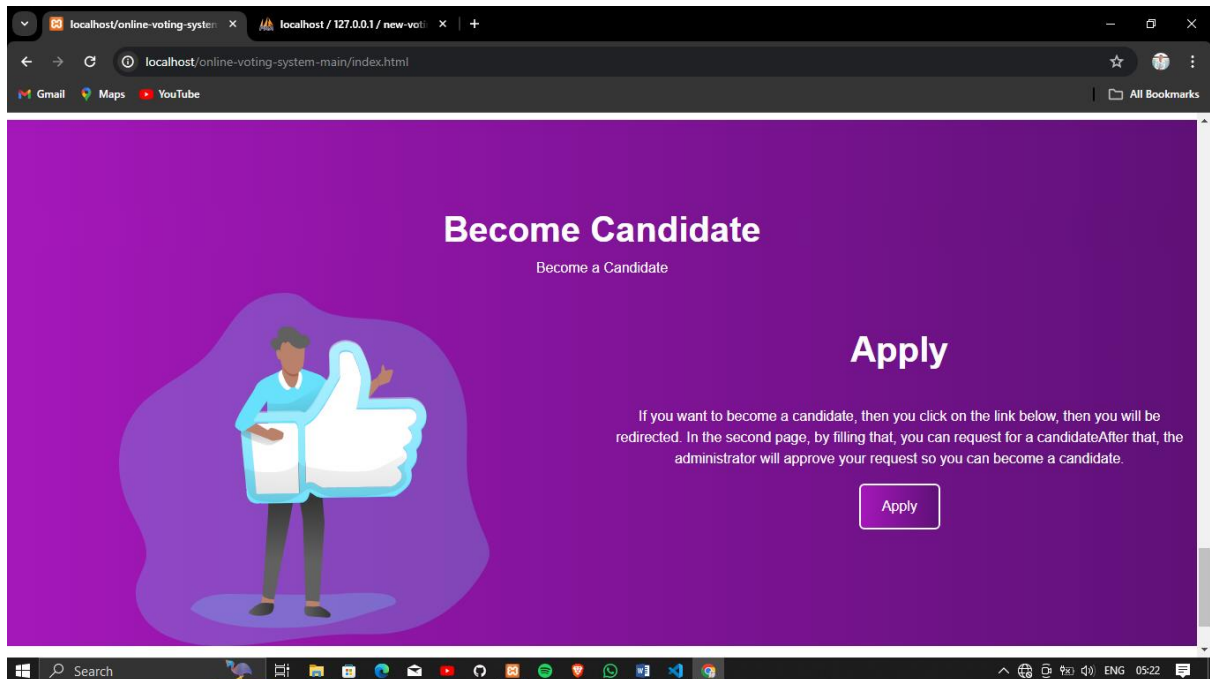
3. Voting:



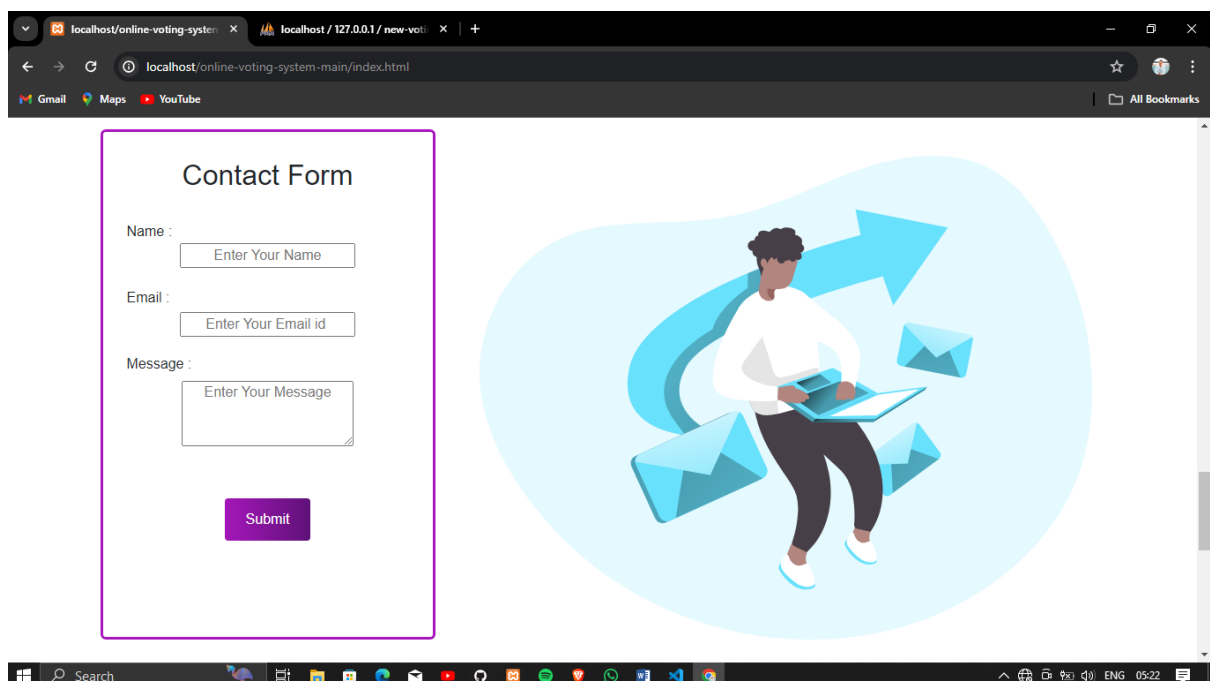
4. Result:



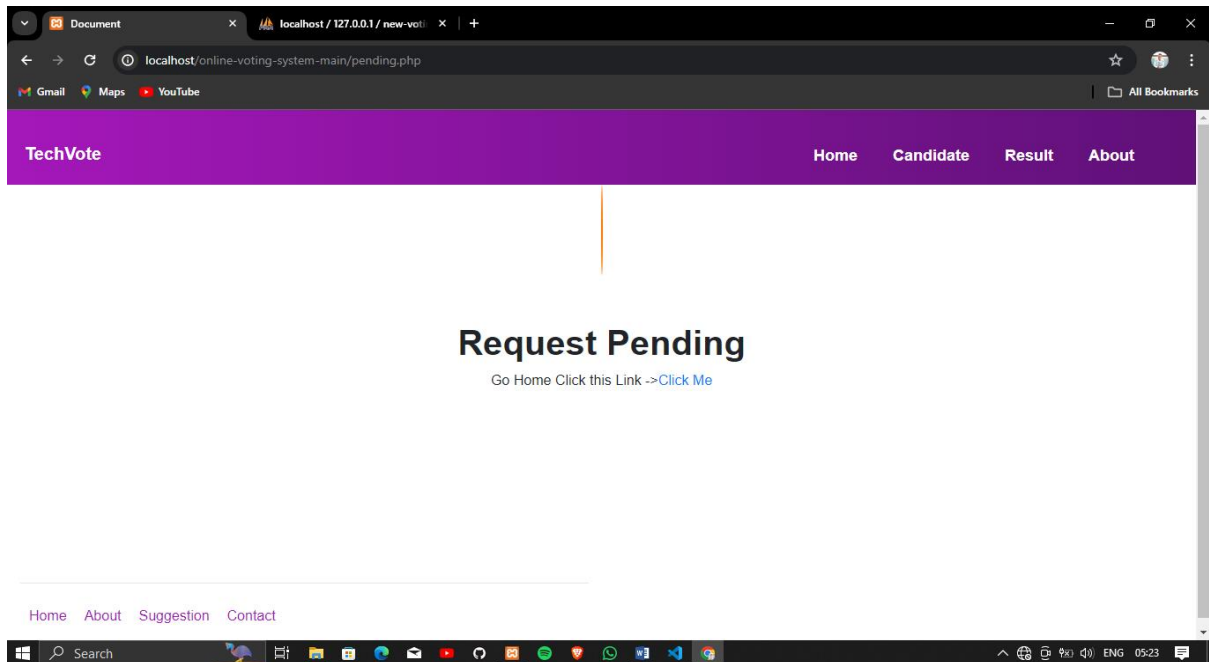
5. Application:



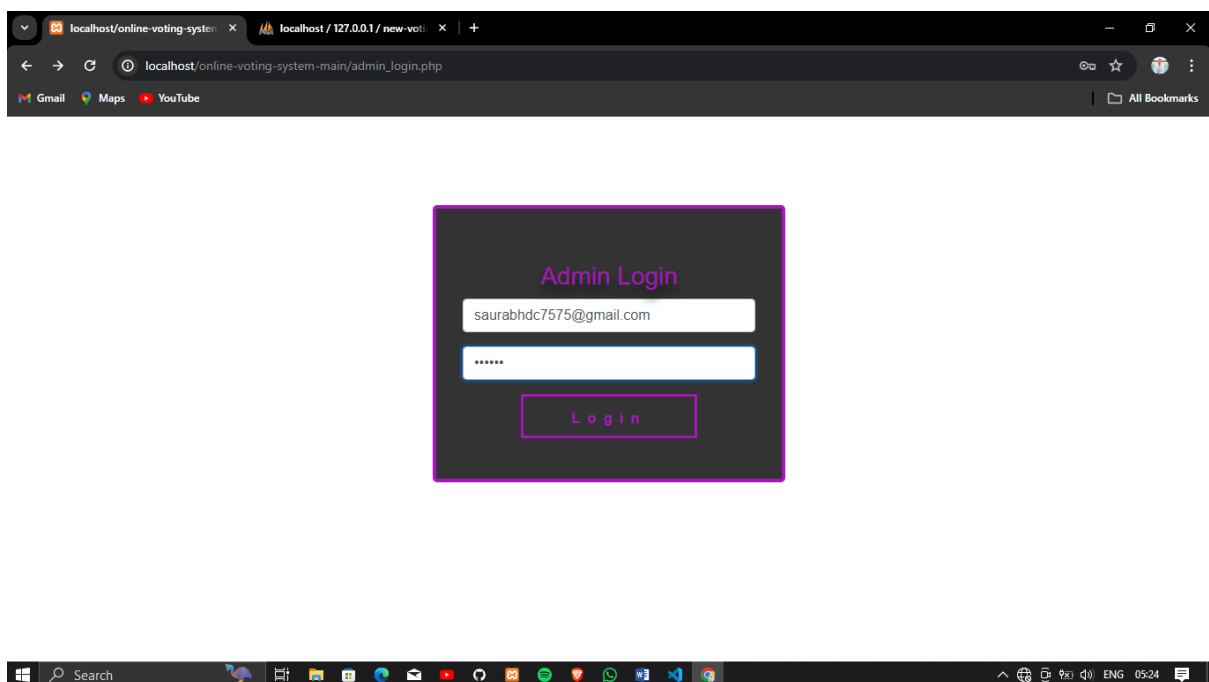
6. Contact:



7. Request:



8. Authentication:



9. Approval:

Sno.	Name	Email	Number	Branch	Roll No.	Enroll ID	status	Action
1	Saurabh Digambar Chorgesaurabhdhdc7575@gmail.com	9130701209	Information Technology	8	1234567890	Pending	Approve	Reject

10. Feedback:

Sno.	Name	Email	Number	Roll No.	Suggestion
1	Amar Appasaheb Pawarpamar123@gmail.com	9669709953	26	Change the UI	