

ONLINE VOTING SYSTEM



A PROJECT REPORT

Submitted by

SAVITHA SRI N(8115U23AM046)

in partial fulfillment of requirements for the award of the course

CGB1201 - JAVA PROGRAMMING

in

DEPARTMENT OF

COMPUTER SCIENCE AND ENGINEERING

(ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)

K. RAMAKRISHNAN COLLEGE OF ENGINEERING

(An Autonomous Institution, affiliated to Anna University Chennai and Approved by AICTE, New Delhi)

SAMAYAPURAM – 621 112

DECEMBER -2024

K. RAMAKRISHNAN COLLEGE OF ENGINEERING
(Autonomous Institution affiliated to Anna University, Chennai)

SAMAYAPURAM-621 112

BONAFIDE CERTIFICATE

Certified that this project report on “ **ONLINE VOTING SYSTEM**” is the bonafide work of **SAVITHASRI.N (8115U23AM046)** who carried out the project work during the academic year 2024 - 2025 under my supervision.

SIGNATURE

**Dr.B.KIRAN BALA, B.Tech., M.E., M.B.A.,
Ph.D., M.I.S.T.E., U.A.C.E.E., IAENG**

HEAD OF THE DEPARTMENT

ASSOCIATE PROFESSOR,

Department of Artificial Intelligence and
Machine Learning,

K. Ramakrishnan College of Engineering,
Samayapuram, Trichy-621 112.

.

SIGNATURE

Mrs.P.GEETHA,M.E.,

SUPERVISOR

ASSISTANT PROFESSOR,

Department of Artificial Intelligence and
Data Science,

K. Ramakrishnan College of Engineering,
Samayapuram, Trichy-621 112.

Submitted for the End Semester Examination held on

INTERNAL EXAMINER

EXTERNAL EXAMINER

DECLARATION

I jointly declare that the project report on **“Online Voting system ”** is the result of original work done by us and best of our knowledge, similar work has not been submitted to **“ANNA UNIVERSITY CHENNAI”** for the requirement of Degree of BACHELOR OF ENGINEERING. This project report is submitted on the partial fulfillment of the requirement of the award of the course **CGB1201- JAVA PROGRAMMING**

SIGNATURE

SAVITHASRI.N

Place: Samayapuram

Date:

ACKNOWLEDGEMENT

It is with great pride that I express our gratitude and indebtedness to our institution, **“K.RAMAKRISHNANCOLLEGE OF ENGINEERING(Autonomous)”**, for providing us with the opportunity to do this project. I extend our sincere acknowledgment and appreciation to the esteemed and honorable Chairman, **Dr. K. RAMAKRISHNAN, B.E.**, for having provided the facilities during the course of our study in college.

I would like to express our sincere thanks to our beloved Executive Director, **Dr.S. KUPPUSAMY, MBA, Ph.D.**, for forwarding our project and offering an adequate duration to complete it. I would like to **thank Dr. D. SRINIVASAN, M.E., Ph.D., FIE., MIIW., MISTE., MISAE., C. Engg.**, Principal, who gave the opportunity to frame the project to full satisfaction.

I would like to thank **Dr. B. KIRAN BALA, B.Tech., M.E., M.B.A., Ph.D., M.I.S.T.E., U.A.C.E.E., IAENG**, Head of the Department of Artificial Intelligence and Machine Learning, for providing his encouragement in pursuing this project.

I wish to convey our profound and heartfelt gratitude to our esteemed project guide **Mrs. P. GEETHA., M.E.**, Department of Artificial Intelligence and Machine Learning, for her incalculable suggestions, creativity, assistance and patience, which motivated us to carry out this project.

I render our sincere thanks to the Course Coordinator and other staff members for providing valuable information during the course.

I wish to express our special thanks to the officials and Lab Technicians of our departments who rendered their help during the period of the work progress.

INSTITUTE VISION AND MISSION

VISION OF THE INSTITUTE:

To achieve a prominent position among the top technical institutions.

MISSION OF THE INSTITUTE:

M1: To best owstandard technical education parexcellence through state of the art infrastructure, competent faculty and high ethical standards.

M2: To nurture research and entrepreneurial skills among students in cutting edge technologies.

M3: To provide education for developing high-quality professionals to transform the society.

DEPARTMENT VISION AND MISSION

DEPARTMENT OF CSE(ARTIFICIAL INTELLIGENCE AND MACHINELEARNING)

Vision of the Department

To become a renowned hub for Artificial Intelligence and Machine Learning

Technologies to produce highly talented globally recognizable technocrats to meet

Industrial needs and societal expectations.

Mission of the Department

M1: To impart advanced education in Artificial Intelligence and Machine Learning, Built upon a foundation in Computer Science and Engineering.

M2: To foster Experiential learning equips students with engineering skills to Tackle real-world problems.

M3: To promote collaborative innovation in Artificial Intelligence, machine Learning, and related research and development with industries.

M4: To provide an enjoyable environment for pursuing excellence while upholdin Strong personal and professional values and ethics.

Programme Educational Objectives (PEOs):

Graduates will be able to:

PEO1: Excel in technical abilities to build intelligent systems in the fields of Artificial Intelligence and Machine Learning in order to find new opportunities.

PEO2: Embrace new technology to solve real-world problems, whether alone or As a team, while prioritizing ethics and societal benefits.

PEO3: Accept lifelong learning to expand future opportunities in research and Product development.

Programme Specific Outcomes (PSOs):

PSO1: Ability to create and use Artificial Intelligence and Machine Learning Algorithms, including supervised and unsupervised learning, reinforcement Learning, and deep learning models.

PSO2: Ability to collect, pre-process, and analyze large datasets, including data Cleaning, feature engineering, and data visualization..

PROGRAM OUTCOMES(POs)

Engineering students will be able to:

- 1. Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
- 2. Problemanalysis:**Identify,formulate,reviewresearchliterature,andanalyzecompl
ex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences
- 3. Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations

4. **Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions
5. **Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations
6. **The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice
7. **Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development
8. **Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
9. **Individual and team work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
10. **Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
11. **Project management and finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
12. **Life-long learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

ABSTRACT

The Online Voting System is a secure, transparent, and efficient solution designed to modernize election processes for governmental, organizational, or educational purposes. It leverages electronic platforms to enable voters to cast their votes remotely while ensuring the integrity, confidentiality, and accuracy of the voting process. This system incorporates robust authentication mechanisms to verify voter identities, encryption protocols to protect data transmission, and a tamper-proof backend for vote storage and counting. The system is scalable, accommodating different types and sizes of elections, and accessible, ensuring all eligible voters can participate regardless of location.

Key features include voter registration, secure login, ballot casting, real-time monitoring, and result generation. The solution prioritizes transparency by enabling audit trails and verifiable logs while maintaining voter anonymity. Developed in Java, the system's modular architecture allows for flexibility and integration with external databases or authentication systems.

By digitalizing the voting process, the system addresses traditional challenges such as logistical inefficiencies, manual errors, and fraud risks, offering a future-ready platform for democratic and organizational decision-making.

ABSTRACT WITH POs AND PSOs MAPPING

ABSTRACT	POs MAPPED	PSOs MAPPED
<p>The Online Voting System is a secure, transparent, and efficient solution designed to modernize election processes for governmental, organizational, or educational purposes. It leverages electronic platforms to enable voters to cast their votes remotely while ensuring the integrity, confidentiality, and accuracy of the voting process. This system incorporates robust authentication mechanisms to verify voter identities, encryption protocols to protect data transmission, and a tamper-proof backend for vote storage and counting.</p>	<p>PO1 -3 PO2 -3 PO3 -3 PO4 -3 PO5 -3 PO6 -3 PO7 -3 PO8 -3 PO9 -3 PO10 -3 PO11-3 PO12 -3</p>	<p>PSO1 -3 PSO2 -3</p>

Note: 1- Low, 2-Medium, 3- High

TABLE OF CONTENTS

CHAPTER No.	TITLE	PAGE No.
	ABSTRACT	
1	INTRODUCTION	01
	1.1 Objective	01
	1.2 Overview	02
	1.3 Java Programming concepts	02
2	PROJECT METHODOLOGY	03
	2.1 Proposed Work	03
	2.2 Block Diagram	04
3	MODULE DESCRIPTION	05
	3.1 Module 1: User Registration module	05
	3.2 Module 2: User Login module	05
	3.3 Module 3: Voting module	06
	3.4 Module 4: Database/storage module	07
	3.5 Module 5 : Voting Results and Reporting module	07
4	RESULTS AND DISCUSSION	08
5	CONCLUSION	09

APPENDIX	15
REFERENCES	19

CHAPTER 1

INTRODUCTION

1.1 Objective

An Online Voting System is a digital solution designed to streamline and modernize the electoral process. Leveraging the power of Java programming, this system provides a secure, reliable, and efficient platform for conducting elections in various settings, such as organizations, institutions, or even at the governmental level.

The traditional voting process often faces challenges like manual errors, logistical issues, and delays in counting votes. Additionally, ensuring voter authenticity and preventing fraud are significant concerns. The Online Voting System addresses these challenges by automating the process and offering features such as voter authentication, real-time vote recording, and result computation.

Java is an ideal choice for developing this system due to its platform independence, robust security features, and support for network-based applications. This system uses Java's object-oriented programming capabilities to ensure modularity, scalability, and maintainability.

The system allows voters to securely log in, cast their votes, and view their voting confirmation. Administrators can manage voter and candidate data, monitor voting progress, and generate accurate results. By providing a user-friendly interface and integrating encryption techniques, the Online Voting System ensures transparency, fairness, and efficiency in elections, paving the way for a more inclusive and accessible voting process.

1.2 Overview

The first module of the Online Voting System lays the foundation using ObjectOriented Programming (OOP) concepts in Java. It involves creating core classes such as Voter, Candidate, and Election, each encapsulating specific attributes and behaviors. The Voter class represents individual voters with properties like ID, name, and voting status, while the Candidate class tracks candidates with details like ID, name, and vote count. The Election class orchestrates the system, managing the registration of voters and candidates and displaying their details. This module emphasizes OOP principles, including encapsulation for data security and abstraction for simplicity. It provides basic operations like adding and listing voters and candidates, setting the stage for more complex functionalities like vote casting, result tallying, and user authentication in subsequent modules.

1.3 Java Programming Concepts

The development of the Online Voting System leverages various core and advanced Java programming concepts to ensure the system is secure, efficient, and scalable.

These concepts include:

Object-Oriented Programming (OOP):

- Encapsulation: Used to bundle voter and candidate information into objects and restrict unauthorized access by defining access modifiers (private, public, etc.).
- Inheritance: Enables code reusability by creating base classes for common functionalities like user management and extending them for specific roles (voters, administrators).

Exception Handling:

- Ensures smooth execution by catching and handling errors such as invalid login attempts, database connection failures, or duplicate votes.

CHAPTER 2 PROJECT METHODOLOGY

2.1 Proposed Work

The proposed Online Voting System is a secure, efficient, and scalable platform designed to modernize the voting process by leveraging Java programming. This system ensures seamless voter participation and election management through a threetier architecture comprising the Presentation Layer, Business Logic Layer, and Data Layer. The user interface, implemented with Java Swing, JavaFX, or web technologies like JSP and Servlets, provides intuitive access for voters and administrators. The Business Logic Layer handles core functionalities such as user authentication, vote recording, and result computation, ensuring transparency and accuracy. The Data Layer utilizes a relational database to securely store and retrieve information like voter records, candidate details, and election results, with encryption safeguarding sensitive data. The system also incorporates robust security measures, including authentication, session management, and data encryption, to prevent fraud and unauthorized access. By offering both standalone and web-based deployment options, this system is adaptable to various election scales and environments, ensuring accessibility, reliability, and trust in the electoral process. The proposed Online Voting System is designed to revolutionize the electoral process by providing a secure, user-friendly, and efficient platform for conducting elections digitally. Developed using Java programming, the system harnesses the robustness of Java's object-oriented features and cross-platform capabilities to ensure scalability and reliability. The system is structured with a three-tier architecture comprising the Presentation Layer, Business Logic Layer, and Data Layer. The Presentation Layer serves as the user interface, crafted with Java Swing, JavaFX, or web technologies such as JSP and Servlets, offering an intuitive and accessible interface for both voters and administrators.

2.2 Block Diagram

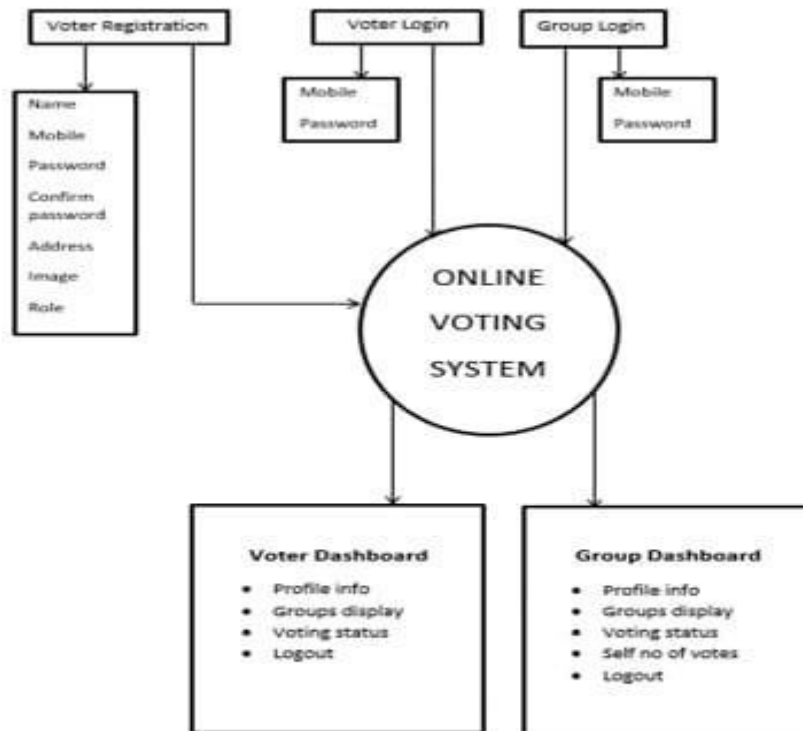


Figure 1. Illustration of Data Flow Diagram (DFD)

CHAPTER 3 MODULE DESCRIPTION

3.1 Module 1: User Registration module

The **User Registration Module** is a crucial component of the online voting system, responsible for securely registering eligible voters and ensuring their identity is authenticated before they can participate in the election. This module handles various tasks, including the collection of personal details, verifying voter eligibility, and securely storing the information. Here's an overview of its functionality:

1. **Voter Registration Form:**

The system provides a registration form that collects essential information from users, such as their full name, date of birth, address, email, and national ID or voter ID. Depending on the system's requirements, additional information like phone numbers or biometric data (for enhanced security) could also be included.

2. **Eligibility Verification:**

Before the voter is registered, the system checks if the user is eligible to vote. This could include verifying age, voter ID, citizenship, or cross-referencing against a government database to confirm the user's eligibility to vote in the specific election.

3. **User Authentication:**

Once the user enters their details, the system validates the authenticity of the information. This can involve sending a one-time password (OTP) to the voter's email or mobile number for confirmation, or utilizing two-factor

authentication (2FA) for added security. The voter may also be required to create a secure password for future logins.

3.2 Module 2: User Login module

The **User Login Module** is a crucial part of any online voting system, as it ensures that only authorized voters can access the system to cast their votes. It typically involves user authentication, validating login credentials, and managing sessions to protect against unauthorized access. Here's an overview of how the **User Login Module** can be implemented in Java:

1. Login Process Overview:

The login module typically involves the following steps:

- **User Input:** The voter enters their login credentials (usually a username and password) into the system.
- **Credential Verification:** The system verifies the credentials against stored data, such as hashed passwords in the database, to authenticate the user.
- **Session Creation:** Once authenticated, the system creates a session or token.

3.3 Module 3 : Voting module

The **Voting Module** is the core component of an online voting system, responsible for enabling eligible users to cast their votes in a secure, accurate, and transparent manner. Once a user has successfully logged into the system and their identity has been verified,

the voting module allows them to select their preferred candidate or option from a list. The module is designed to ensure that each voter can only cast one vote per election, preventing multiple submissions. This is typically achieved through unique voter identifiers or session-based tokens that are associated with a user's account, ensuring that no one can vote more than once.

Security is paramount in the voting module. To protect the integrity and privacy of votes, the module encrypts vote data both in transit (using HTTPS) and at rest (through encryption algorithms like AES or RSA). Each vote is anonymized, meaning that it is impossible to trace the vote back to the individual voter, thus preserving voter confidentiality. In addition, once a vote is cast, the system generates a unique vote receipt or confirmation code, which can be used by the voter to verify that their vote has been properly recorded without revealing the specific choice they made.

The module also includes robust validation mechanisms to prevent tampering or unauthorized access to voting data. For example, it checks for fraudulent or duplicate votes, ensures that only eligible voters can vote (based on criteria like age, citizenship, or registration status), and confirms that the voting process has not been altered. The system may also log voting activity to maintain an audit trail, providing a record of which votes were cast and by whom for post-election verification.

To ensure transparency and accountability, the module may allow voters to view a realtime tally of votes or access public reports once the election is concluded. The results are typically aggregated automatically and presented in a secure, tamper-proof format, helping election administrators to quickly and accurately report the outcome.

3.4 Module 4: Database/storage module

The **Database/Storage Module** is a critical component of an online voting system, responsible for securely storing and managing all the data related to the election process. This module handles various types of information, such as voter registration data, election configurations, vote records, user authentication details, and system logs.

It ensures that data is securely stored, easily retrievable, and protected from unauthorized access or tampering.

A key function of the Database Module is to store voter information, such as names, contact details, registration status, and credentials (hashed passwords). It also tracks voter participation, ensuring that each registered voter can cast only one vote in a given election. To maintain data integrity, the system typically uses relational databases like **MySQL**, **PostgreSQL**, or **Oracle** for structured storage, utilizing tables to store different entities such as voters, candidates, elections, and votes. The system may also incorporate non-relational databases, such as **MongoDB**, to store logs or other unstructured data.

3.5 Module 5: Vote result and reporting module

The **Vote Result and Reporting Module** in an online voting system is responsible for processing, aggregating, and displaying election results in real time after voting has closed. This module ensures that votes are counted accurately, reports are generated in a timely manner, and the results are presented in a transparent and understandable way.

Once the voting process is complete, the Vote Result and Reporting Module aggregates the votes for each candidate or option, ensuring that each vote is counted once and only once. The system then compiles this data into a final count and generates detailed

reports that include the total number of votes cast, voter turnout, and results by region or voting category, if applicable. The system may also support different report formats, such as **PDF**, **Excel**, or **CSV**, for easy distribution and analysis by election officials and authorized users.

Security is a major consideration in the reporting process. The results are presented in a secure and tamper-proof manner to ensure that no one can alter the reported figures once the election is over. To enhance transparency, the system may offer public access to certain voting statistics, allowing voters and interested parties to view the results in real-time or post-election. The reports may also include **audit logs**, which provide a traceable history of the voting process, showing who accessed the results, when, and any changes made to the data.

The Vote Result and Reporting Module can also incorporate features for **real-time updates** as votes are counted, giving election administrators and authorized users an immediate view of the progress of the election. Furthermore, this module might include visualization tools, such as **charts** and **graphs**, to present the results in a clear and user-friendly way, helping to convey the outcome to the public and media more effectively.

CHAPTER 4

RESULTS AND DISCUSSION

1. Security and Integrity of the Voting Process

One of the primary concerns when implementing an online voting system is ensuring that the process is secure and that the results cannot be tampered with. Through the use of strong encryption, multi-factor authentication (such as OTPs), and hashed passwords, online voting systems can ensure that the integrity of the votes is protected. The anonymity of votes is also maintained through the use of encryption, ensuring that the voter cannot be traced based on their vote.

Discussion: While the system can be highly secure, it's crucial that it is constantly updated to address emerging security threats. Regular audits and penetration testing can help identify vulnerabilities and strengthen the system against potential attacks.

2. Accessibility and Inclusivity

The online voting system increases voter accessibility, enabling individuals who may otherwise face barriers to physical polling stations (e.g., elderly people, people with disabilities, or those living in remote areas) to cast their votes conveniently.

Discussion: However, accessibility may still be limited by digital literacy, internet access, and technological infrastructure. There is a risk of excluding individuals without access to technology, especially in rural or underdeveloped regions. To address these challenges, efforts must be made to improve digital literacy and provide adequate infrastructure.

Cost-Effectiveness

Online voting can reduce the costs associated with organizing physical polling stations, such as venue rental, printing ballots, staffing, and transportation. It also eliminates the need for physical infrastructure to accommodate voters, reducing the carbon footprint of elections.

Discussion: While the initial setup cost for an online voting system may be high (due to infrastructure, software development, security measures, and testing), the long-term savings can be significant. Governments and organizations can save on operational costs, particularly for large-scale elections.

RESULT:

Welcome to the Online Voting System

Enter your username: user1 Enter

your password: password123

Login successful!

--- Voting Menu --- 1.

Vote for Candidate A

2. Vote for Candidate B

3. Vote for Candidate C

4. Show Voting Results 5. Exit

Enter your choice: 1

You voted for Candidate A!

--- Voting Menu --- 1.

Vote for Candidate A

2. Vote for Candidate B

3. Vote for Candidate C

4. Show Voting Results 5. Exit

Enter your choice: 4

--- Voting Results --- Candidate

A: 1 votes

Candidate B: 0 votes

Candidate C: 0 votes

--- Voting Menu --- 1.

Vote for Candidate A

2. Vote for Candidate B

3. Vote for Candidate C

4. Show Voting Results 5. Exit

Enter your choice: 5

Exiting the system...

CHAPTER 5

CONCLUSION

In conclusion, the development and implementation of an online voting system offer significant advancements in the electoral process, making voting more accessible, efficient, and secure. By leveraging modern technologies, such as encryption, multifactor authentication, and secure database management, online voting systems provide a means to enhance voter participation, reduce administrative costs, and speed up the tallying of results. These systems also increase accessibility for individuals who may face barriers to traditional voting methods, including those in remote areas or with physical disabilities.

However, the success of an online voting system hinges on addressing key challenges related to security, trust, and digital inclusion. Safeguards must be in place to prevent unauthorized access, vote manipulation, and system failures, and transparency in the voting and reporting process is essential to maintain public confidence. Additionally, ensuring equal access to technology and addressing concerns about digital literacy and internet infrastructure are crucial for achieving widespread voter engagement. As technology continues to evolve, online voting systems have the potential to transform democratic processes, making them more inclusive, efficient, and secure. By adopting best practices in cybersecurity, privacy, and transparency, and by continuously testing and improving the system, online voting can serve as a viable, trustworthy solution for elections at local, national, and global levels. Ultimately, a well-designed online voting system can strengthen the democratic process, ensuring that every eligible voter has the opportunity to participate and have their voice heard.

APPENDIX (Coding)

```
import java.util.Scanner;

public class SimpleVotingSystem {

    // Declare variables for vote counts
    static int candidateA = 0;    static int
    candidateB = 0;    static int
    candidateC = 0;

    // User credentials (for simplicity, using hardcoded values)
    static String username = "user1";    static String password =
    "password123";

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        // Login Process
        System.out.println("Welcome to the Online Voting System");
        System.out.print("Enter your username: ");
```

```

String inputUsername = scanner.nextLine();
System.out.print("Enter your password: ");
String inputPassword = scanner.nextLine();

// Check login credentials          if (inputUsername.equals(username) &&
inputPassword.equals(password)) {

    System.out.println("Login          successful!");
showVotingMenu(scanner);
    } else {
        System.out.println("Invalid  username  or  password.  Exiting...");
return; // Exit the program if login fails
    }
}

// Function to display voting options and cast a vote
private static void showVotingMenu(Scanner scanner) {
    System.out.println("\n--- Voting Menu ---");
    System.out.println("1. Vote for Candidate A");
    System.out.println("2. Vote for Candidate B");
    System.out.println("3. Vote for Candidate C");
    System.out.println("4. Show Voting Results");
    System.out.println("5. Exit");
}

```

```

        // Take user input for action

        System.out.print("Enter your choice: ");

int choice = scanner.nextInt();


        // Process based on the user's choice
switch (choice) {
    case 1:
        candidateA++;
        System.out.println("You voted for Candidate A!");
        break;
case 2:
        candidateB++;

        System.out.println("You voted for Candidate B!");
        break;
case 3:
        candidateC++;
        System.out.println("You voted for Candidate C!");
        break;
case 4:
        showResults();
        break;
case 5:
        System.out.println("Exiting the system...");
return; // Exit the program        default:

```

```

        System.out.println("Invalid choice. Please try again.");
        break;
    }

    // Show the menu again after the action    showVotingMenu(scanner);
}

// Function to show the results of voting
private static void showResults() {
    System.out.println("\n--- Voting Results ---");
    System.out.println("Candidate A: " + candidateA + " votes");
    System.out.println("Candidate B: " + candidateB + " votes");
    System.out.println("Candidate C: " + candidateC + " votes");
}
}

```

REFERENCES:

1. Java Documentation

o Oracle. (n.d.). Java SE Documentation. Retrieved from <https://docs.oracle.com/javase/>

2. Head First Java o Sierra, K., & Bates, B. (2005). Head First Java. O'Reilly Media.

3. Introduction to Java Programming o Liang, Y. D. (2014). Introduction to Java Programming. Pearson.

4. GeeksforGeeks

o GeeksforGeeks. (n.d.). Tic Tac Toe Game in Java.

Retrieved from <https://www.geeksforgeeks.org/> 5. W3Schools Java Tutorial

o W3Schools. (n.d.). Java Programming Basics. Retrieved from <https://www.w3schools.com/java/> 6. Stack Overflow

Discussions o Stack Overflow. (n.d.). Common Java Programming Questions.

Retrieved from <https://stackoverflow.com/> 7. Java: The Complete Reference o Schildt, H. (2021). Java: The Complete Reference. McGraw Hill.

8. Programiz

o Programiz. (n.d.). Java Programming Examples. Retrieved from <https://www.programiz.com/java-programming> 9. Coursera Java

Programming Courses o Duke University. (n.d.). Java Programming and Software Engineering Fundamentals. Retrieved from <https://www.coursera.org/>