INTERNSHIP PROJECT-7 DOCUMENT

ON “VOTING SYSTEM (JAVA)”

Submitted by:	Submitted To:-
Navpreet Singh (INTERN)	Kanduri Abhinay (FOUNDER)
	RITHIN VERMA (CTO)

INDEX

INTRODUCTION:-

The Online Voting System is a digital platform designed to facilitate secure and convenient voting processes over the internet. It ensures voter authentication and prevents multiple voting while allowing quick vote casting and tallying. This system aims to simplify traditional election processes by providing a transparent and accessible way for voters to participate remotely.

SOFTWARE REQUIREMENTS:-

- **Programming Language**: Java (JDK 8 or higher)
- **IDE**: VS Code, IntelliJ IDEA, or any Java-compatible editor
- **Operating System**: Platform-independent (Windows, Linux, Mac)
- **Other Requirements**: Database like MySQL for persistent storage (if advanced)
- Command line or storage access.

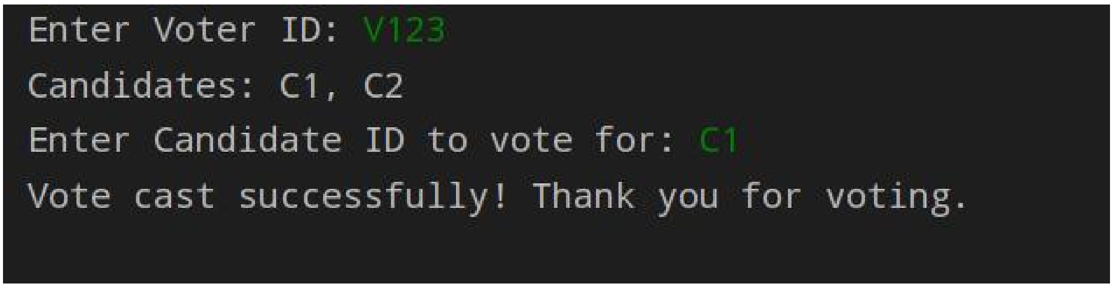
DESIGN:-

The system uses a simple client-side terminal interface to input voter ID and cast votes. Data is stored in memory using Java collections (HashMap) to keep track of registered voters, candidates, and votes. Voter authentication checks if a voter ID exists and if the voter has not already voted. Votes are recorded by incrementing candidate counters. The design supports simplicity and basic security mechanisms.**I**

NPUT:-

- **Voter ID:** A unique identifier for each eligible voter.
- **Candidate ID:** The identifier for the candidate of choice.

The system prompts the user to enter these inputs sequentially to verify identity and record the vote.



```
Enter Voter ID: V123
Candidates: C1, C2
Enter Candidate ID to vote for: C1
Vote cast successfully! Thank you for voting.
```

IMPLEMENTATION:-

The program is implemented using core Java features:

- HashMaps maintain voter statuses and candidate vote counts.
- Scanner class is used for reading console input.
- Conditional checks validate voter eligibility and candidate existence.
- Vote counting is implemented by incrementing candidate counters.
- Simple console feedback provides confirmation or error messages.

TESTING:-

Testing involves scenarios such as:

- Valid voter casting a vote successfully.
- Invalid voter ID rejected.
- Preventing voters from voting multiple times.
- Invalid candidate ID is handled gracefully.
  Manual testing via console input ensures correct system behavior.

ADVANTAGES:-

- Efficient and fast voting process.
- Easy to use with console-based interaction.
- Prevents multiple voting and unauthorized access.
- Reduces cost and time compared to traditional paper-based elections.
- Portable and easy to update or expand.

CONCLUSION:-

The Online Voting System provides a basic yet functional framework for conducting secure online elections. While designed for simplicity, it highlights the key features of voter authentication and vote recording. This project can be expanded with features such as encryption, user interfaces, and real database storage to serve real-world election needs.

REFERENCES:-

- **Java Official Documentation -** https://docs.oracle.com/en/java/
- Online Voting System Tutorials and Code Examples
- Cryptography and Security Basics in Java
- Various online project sources and GitHub repositories for voting systems