Online Voting System

Software Requirements Specification

Ishrat Jahan; Md. Shakib Patoary; Md. Ramjan Ali

Department of Software Engineering
Gazipur Digital University

August 2025

Contents

0.1 Objective(s)

- To develop a secure, reliable, and user-friendly online voting system that ensures transparency and accuracy in the election process.

0.2 Problem Analysis

- Traditional voting systems take a lot of time, can have mistakes, and are not very secure, so we need a safe and efficient online voting system.

0.3 Methodology

- Requirement Specification Procedure:

1 Introduction

This document specifies the Software Requirements Specification (SRS) for the Online Voting System. It defines the functional and non-functional requirements of the system and provides a reference for all stakeholders involved in the development process.The online voting system is a system that enables all citizens to cast their vote online.The purpose is to increase the voting percentage across the country, as in the present system people who live their hometown are not able to cast vote during the elections. So due to this the voting percentage across the country is very less. Through this software those people who live out of their hometown will also be able to cast their votes as this system is online.

1.1 Intended Audience

This document is intended for:

- Developers: For implementation and technical design.
- Testers: For validation and verification.
- Project Managers: For tracking progress and scope.
- Stakeholders: For understanding system capabilities and constraints.

1.2 Document Organization

- Section 2: Overall Description – Product perspective, features, users, and environment.
- Section 3: System Features – Detailed functional requirements.
- Section 4: External Interface Requirements – UI, hardware, software, and communication interfaces.
- Section 5: Non-Functional Requirements – Performance, security, usability, etc.
- Section 6: Other Requirements – Legal, database, and additional constraints.

1.3 Project Scope

Product Description: The Online Voting System is a secure, web-based platform designed to conduct transparent and efficient election. It automates key tasks such as voter registration, authentication, ballot casting, result tabulation, and administrative oversight to minimize errors and improve efficiency.

Objectives and Goals:

- Ensure secure and tamper-proof voting.
- Provide a simple, user-friendly interface.
- Enable real-time aggregation and display of results.
- Support effective administrative control and monitoring.
- Enhance transparency and trust by minimizing manual intervention.

Relation to Business Goals: The system aligns with organizational objectives by encouraging democratic participation, ensuring fair election practices, and improving operational efficiency. It directly contributes to higher voter engagement, reliable reporting, and streamlined election management.

Scope Limitations: This SRS covers the core online voting platform. Large-scale governmental elections or integration with external identity verification systems are beyond the scope of this version and may be considered in future releases.

2 Overall Description

2.1 Product Perspective

The software product is a standalone system and not a part of a larger system. The system is made up of two parts: one running visibly for the end users, in this case the voters, through web pages, and the other running on the server. The two users of the system, namely the voters and the administrator, interact with the system in different ways. The administrator configures and manages the system on the server according to requirements, while the voters cast their votes using the provided web interface. These votes are then accepted and processed by the system on the server.

2.2 Product Features

The Online Voting System provides several high-level features designed to meet both user and organizational needs. These features are organized to make them understandable to any reader of the SRS.

Voter Features

- **Register / Login:** Voters can securely register and log in to the system.
- **Verify Eligibility:** Ensures only eligible voters can participate.
- **Cast Vote:** Allows voters to cast their vote using the web interface.

- **View Voting Confirmation:** Voters can confirm that their vote has been recorded successfully.
- **View Election Status:** Enables voters to check the ongoing status of the election.
- **Logout:** Securely log out of the system.
- **Report Issue / Contact Support:** Voters can report problems or contact support for assistance.

Admin Features

- **Create Election:** Administrators can set up new elections.
- **Manage Candidates List:** Add, update, or remove candidates for elections.
- **View / Generate Results:** Access and generate election results.
- **Send Notifications to Voters:** Notify voters about upcoming elections or important updates.
- **Resolve Voter Issues / Support:** Handle voter queries and issues.
- **Close Election:** Officially end an election after completion.

2.3 User Classes and Characteristics

The Online Voting System has two main user classes, each with distinct characteristics and roles.

(a) Voters / End Users

- **Frequency of use:** During each election period (occasional, per election)
- **Functions used:** Register/Login, Verify Eligibility, Cast Vote, View Voting Confirmation, View Election Status, Logout, Report Issue/Contact Support
- **Expertise:** Low to moderate technical knowledge (basic computer or smartphone skills)
- **Privileges:** Limited to personal voting actions
- **Importance:** Primary users, most important for system purpose

(b) Administrators / Election Managers

- **Frequency of use:** Regular (before, during, and after each election)
- **Functions used:** Create Election, Manage Candidates List, View/Generate Results, Send Notifications to Voters, Resolve Voter Issues/Support, Close Election
- **Expertise:** High technical knowledge (familiar with election processes and system administration)

- **Privileges:** Full administrative access to system controls and configurations
- **Importance:** Secondary users but vital for system operation, security, and reliability

2.4 Operating Environment

The Online Voting System will operate in a defined set of hardware and software environments to ensure smooth and secure functionality.

Hardware Platform

- Web server (Cloud-based or on-premise) for hosting the application and database
- Client devices: Desktop computers, laptops, tablets, and smartphones for voter access
- Network infrastructure: Internet connectivity for both server and client communication

Operating System

- Server: Linux-based (Ubuntu 20.04+, CentOS 8+) or Windows Server 2019+
- Client: Any modern OS (Windows 10+, macOS, Linux, Android 10+, iOS 13+)

Software Dependencies

- Web server: Apache / Nginx / IIS
- Database: MySQL
- Web technologies: Java
- Optional: SSL/TLS for secure connections, email service for notifications

Other Requirements

- Must function over standard internet connections, with appropriate security and encryption
- Compatible with modern web browsers (Chrome, Firefox, Edge, Safari)
- Should handle concurrent access by multiple users (voters and admin) efficiently
- Secure storage of voter information and voting records to maintain confidentiality

2.5 Design and Implementation Constraints

This section describes the constraints that may limit design choices and implementation options for the Online Voting System. These constraints ensure compliance with corporate policies, regulatory standards, hardware limitations, and integration requirements.

- **Regulatory Compliance:** The system must comply with national election laws and data protection regulations to ensure voter privacy and election integrity.
- **Hardware Limitations:** The system should operate efficiently on standard web servers and client devices without requiring high-end hardware.
- **Software Interfaces:** The system must integrate with notification services to send alerts and updates to voters.
- **Database Constraints:** The database must securely handle large volumes of voter data and election results, ensuring consistency and reliability.
- **Security Considerations:** End-to-end encryption must be implemented for vote transmission and storage to prevent tampering and unauthorized access.
- **Design Conventions:** The system should follow standard web development best practices and programming conventions to ensure maintainability and future updates.
- **Programming Language Constraint:** The entire system will be developed using **Java** as the primary programming language. Other languages or frameworks are not permitted.
- **Concurrent Operations:** The system must be able to handle multiple users (voters and admin) simultaneously without performance degradation.
- **Maintenance Responsibility:** The delivered software must be maintainable by the customer organization, so coding standards, documentation, and modular design must be followed.

3 System Features

3.1 Feature 1: User Registration / Login

Description and Priority

Provides secure access to the system by verifying the user's identity before allowing participation.
Priority: High
Priority Components: Benefit: 9, Penalty: 9, Cost: 4, Risk: 8

Stimulus/Response Sequences

User opens the system → enters credentials → system validates input → if valid, access granted; else, error displayed.

Functional Requirements

FR1. The system shall allow users to register with unique credentials.

FR2. The system shall allow users to log in securely.
FR3. The system shall lock accounts after three failed attempts.
FR4. The system shall log authentication attempts.
FR5. The system shall enforce session timeouts.

3.2 Feature 2: Verify Eligibility

Description and Priority
Ensures only eligible users can vote by validating voter identity and eligibility criteria.
Priority: High
Priority Components: Benefit: 9, Penalty: 8, Cost: 5, Risk: 7

Stimulus/Response Sequences
User requests to vote → system checks eligibility against voter list → if eligible, access granted; else, restricted.

Functional Requirements
FR1. The system shall verify voter details against the voter database.
FR2. The system shall prevent ineligible users from voting.
FR3. The system shall notify users if eligibility fails.

3.3 Feature 3: Cast Vote

Description and Priority
Allows eligible voters to cast their vote through the web interface securely.
Priority: High
Priority Components: Benefit: 9, Penalty: 9, Cost: 6, Risk: 8

Stimulus/Response Sequences
User selects candidate → system confirms choice → vote stored in database securely.

Functional Requirements
FR1. The system shall allow users to cast a single vote.
FR2. The system shall record votes securely in the database.
FR3. The system shall prevent multiple votes from the same user.

3.4 Feature 4: View Voting Confirmation

Description and Priority
Provides confirmation after a user casts their vote successfully.
Priority: High
Priority Components: Benefit: 8, Penalty: 7, Cost: 3, Risk: 5

Stimulus/Response Sequences
User submits vote → system generates confirmation message.

Functional Requirements

FR1. The system shall provide confirmation of successful voting.
FR2. The system shall generate a unique confirmation ID.

3.5 Feature 5: View Election Status

Description and Priority
Allows voters to view the status of the election process without revealing results prematurely.
Priority: Medium
Priority Components: Benefit: 7, Penalty: 5, Cost: 4, Risk: 4

Stimulus/Response Sequences

User selects "Election Status" option → system displays real-time participation status.

Functional Requirements

FR1. The system shall display total number of voters who have voted.
FR2. The system shall not reveal actual results before election closure.

3.6 Feature 6: Report Issue / Contact Support

Description and Priority
Provides voters a way to report technical issues or request help.
Priority: Medium
Priority Components: Benefit: 7, Penalty: 6, Cost: 3, Risk: 4

Stimulus/Response Sequences

User clicks "Report Issue" → system provides form → user submits complaint → system logs and forwards to admin.

Functional Requirements

FR1. The system shall allow users to report issues.
FR2. The system shall notify admin of reported issues.

3.7 Feature 7: Logout

Description and Priority
Provides users the ability to safely exit the system.
Priority: High
Priority Components: Benefit: 8, Penalty: 6, Cost: 2, Risk: 4

Stimulus/Response Sequences

User clicks "Logout" → system ends session → returns to login page.

Functional Requirements

FR1. The system shall log out users after manual request.
FR2. The system shall clear session data on logout.

3.8 Feature 8: Create Election (Admin)

Description and Priority

Admin sets up and configures a new election.
Priority: High
Priority Components: Benefit: 9, Penalty: 8, Cost: 4, Risk: 6

Stimulus/Response Sequences

Admin selects "Create Election" → provides details → system initializes election.

Functional Requirements

FR1. The system shall allow admin to create elections.
FR2. The system shall define election details (name, date, type).

3.9 Feature 9: Manage Candidate List (Admin)

Description and Priority

Admin adds, edits, or removes candidates for the election.
Priority: High
Priority Components: Benefit: 8, Penalty: 7, Cost: 4, Risk: 5

Stimulus/Response Sequences

Admin accesses candidate management → system updates candidate database.

Functional Requirements

FR1. The system shall allow admin to add/remove/update candidates.

3.10 Feature 10: View / Generate Results (Admin)

Description and Priority

Provides the election results after voting ends.
Priority: High
Priority Components: Benefit: 9, Penalty: 9, Cost: 6, Risk: 8

Stimulus/Response Sequences

Admin requests results → system compiles votes → displays final results.

Functional Requirements

FR1. The system shall calculate results after election closes.
FR2. The system shall generate a results report.

3.11 Feature 11: Send Notifications to Voters (Admin)

Description and Priority
Allows admin to send reminders or announcements to voters.
Priority: Medium
Priority Components: Benefit: 7, Penalty: 5, Cost: 3, Risk: 4

Stimulus/Response Sequences

Admin writes message → system delivers to registered voters.

Functional Requirements

FR1. The system shall allow admin to send notifications to all voters.

3.12 Feature 12: Resolve Voter Issues / Support (Admin)

Description and Priority
Admin provides support by addressing voter-reported issues.
Priority: Medium
Priority Components: Benefit: 7, Penalty: 6, Cost: 3, Risk: 5

Stimulus/Response Sequences

Voter reports issue → admin reviews → system updates status to resolved.

Functional Requirements

FR1. The system shall allow admin to track and resolve voter issues.

3.13 Feature 13: Close Election (Admin)

Description and Priority
Allows admin to officially close election and stop voting process.
Priority: High
Priority Components: Benefit: 9, Penalty: 8, Cost: 3, Risk: 6

Stimulus/Response Sequences

Admin selects “Close Election” → system disables voting → election status updated.

Functional Requirements

FR1. The system shall allow admin to close elections.
FR2. The system shall prevent further votes after closure.

4 External Interface Requirements

4.1 User Interfaces

- **Logical Design:** The system provides a web-based interface with clear layout and navigation flow. Standard UI components like buttons, forms, tables, and dropdown menus are used.

- **GUI Standards:** Responsive design suitable for desktops, laptops, tablets, and smartphones. Intuitive navigation with a consistent color scheme.
- **Error Messages:** Clear and concise alerts, e.g., ”Invalid voter ID”, ”Session expired”, or ”Vote already cast”.
- **Components Requiring UIs:**
  - Login/Registration Page
  - Voter Dashboard
  - Vote Casting Page
  - Election Status Page
  - Voting Confirmation Page
  - Admin Dashboard
  - Candidate Management
  - Election Creation
  - Notification Center
  - Reporting & Results Pages
- **Accessibility Considerations:** Supports multiple languages, large-text mode, screen reader friendly, and keyboard navigation for better accessibility.

4.2 Hardware Interfaces

Not applicable.

4.3 Software Interfaces

- **Database:** MySQL database for storing voter information, election details, and votes securely.
- **Server OS:** Linux (Ubuntu/CentOS) or Windows Server.
- **Web Server Tools:** Apache, Nginx, or any compatible web server.
- **APIs/Libraries:** REST APIs for client-server communication; JSON format for data exchange.
- **Data Exchange Example:**

```
{
    "voter_id": "V123456",
    "candidate_id": "C987",
    "timestamp": "2025-08-21T10:30:00"
}
```

4.4 Communications Interfaces

- **Communication Methods:** Users access the system through the Internet using standard browsers.
- **Standards/Protocols:** HTTPS for secure communication; REST API for server-client interactions.
- **Security Considerations:** End-to-end encryption for all communications, token-based authentication, and session management to prevent unauthorized access.
- **Performance Expectations:** System should respond to user actions within 2-3 seconds under normal network conditions. Supports multiple concurrent users without noticeable lag.

4.5 Nonfunctional Requirements

4.5.1 Performance Requirements

- The system shall handle up to 10,000 concurrent users without performance degradation.
- Each vote casting operation shall be processed within 3 seconds.
- Election results shall be generated within 5 minutes after election closure.
- The system shall maintain response times under 2 seconds for navigation between web pages.

4.5.2 Safety Requirements

- The system shall ensure no loss of votes or voter information during normal operation and failures.
- Regular backups shall be performed hourly to prevent data loss.
- The system shall provide safeguards against accidental deletion or modification of election data.
- Compliance with local election regulations shall be enforced to ensure election integrity.

4.5.3 Security Requirements

- All communication shall be encrypted using SSL/TLS protocols.
- User authentication shall enforce unique credentials with optional two-factor authentication.
- The system shall prevent multiple voting by the same voter.
- Audit logs of all user and admin activities shall be maintained securely.
- Access control shall ensure only authorized personnel can modify election configurations or results.

4.5.4 Software Quality Attributes

- **Availability:** The system shall maintain 99.9% uptime during election periods.
- **Usability:** The system shall provide an intuitive interface, support multiple languages, and follow accessibility standards (WCAG).
- **Maintainability:** The system shall follow modular design principles and Java coding standards to simplify updates and maintenance.
- **Scalability:** The system shall support increasing numbers of voters by allowing horizontal scaling of the server infrastructure.
- **Portability:** The system shall be compatible with major web browsers and work on both desktop and mobile devices.
- **Reliability:** The system shall operate without critical failures during election periods and automatically recover from minor issues.

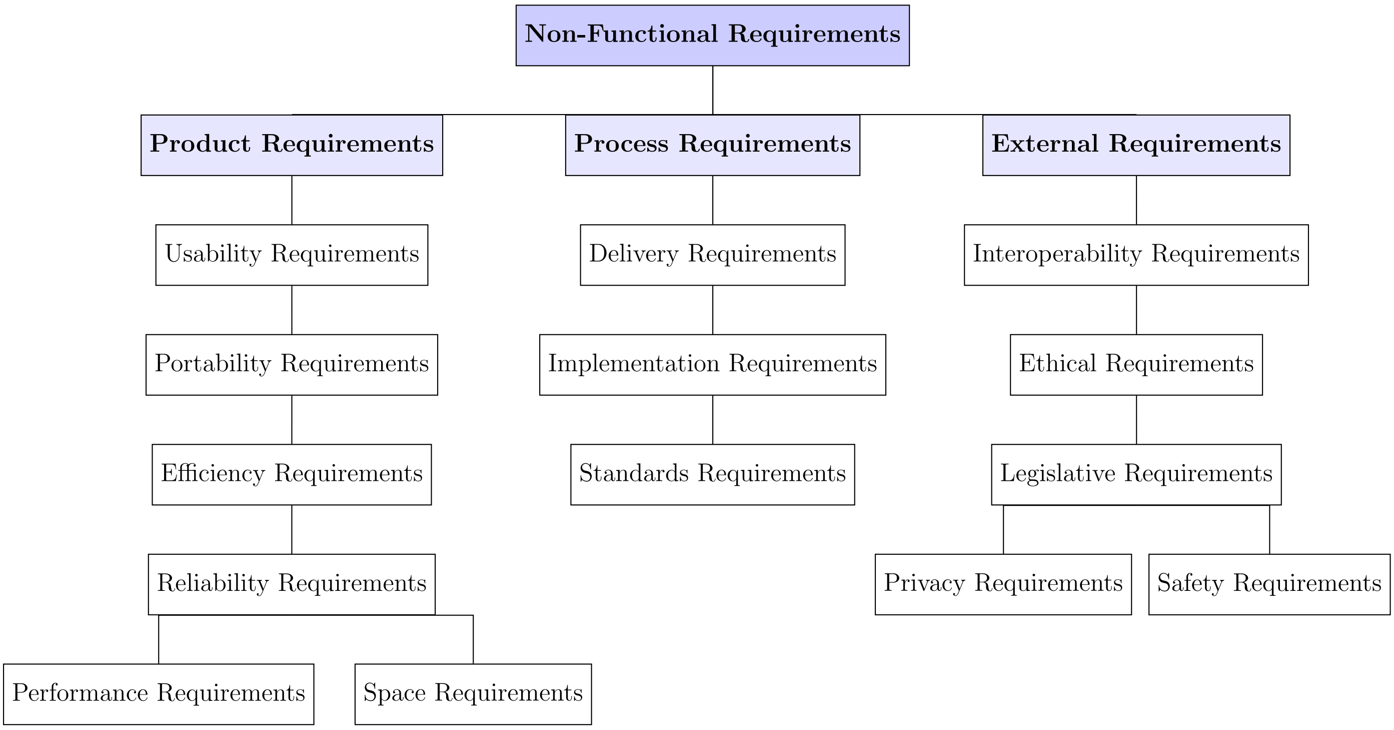

Figure 1: Non-Functional Requirements Online Voting System

5 Other Requirements

5.1 Database Requirements

- The system shall use a secure and reliable database to store voter, candidate, vote, and election information.

5.2 Internationalization Requirements

- The system shall support both English and Bengali languages.
- Future support for additional languages shall be possible through a multilingual framework.

5.3 Legal Requirements

- The system shall comply with national election laws and data protection regulations.
- Voter privacy shall be maintained, and all votes shall remain confidential.

5.4 Security Requirements

- The voting process shall include security measures to prevent vote tampering or fraud.

5.5 Reuse Objectives

- Modules such as login, notification, and result generation shall be designed to be reusable in future election or voting projects.

5.6 Performance and Scalability Requirements

- The system shall handle multiple voters logging in and voting simultaneously without performance degradation.
- Server scalability mechanisms shall be implemented to manage high traffic during election periods.

5.7 Support and Maintenance Requirements

- 24/7 helpdesk support shall be available for voter and admin issues.
- Regular maintenance and updates shall be performed to fix bugs and improve system performance.

Contribution Table:

Name	**Contribution**	**Percentage**
Md.Shakib Patoary	Introduction, Overall Description, System Features	34%
Ishrat Jahan	System Features, External Interface Requirements	33%
Ramjan Ali	System Features, Other Requirements	33%