Software Design and Analysis

Document Report

For

E-Voting System

# I21-0433 Nida Azam

# I21-1520 Arsal Miraj

# I20-0461 Basit Rauf

**Startex**

26 November,2024

Table of Content

1. **Introduction****3**

1.1 Purpose 3

1.2 Product scope 3

1.3 Objective 3

1.4 Problem Statement 3

1. **Requirements 4**

2.1 Functional Requirements 4

2.2 Non-Functional Requirements 4

1. **System Design Models 5**

3.1 Use Cases 5

3.2 Domain Model 6

3.3 Class Diagram 7

3.4 System Sequence Diagram 8

# Introduction

## 1.1 Purpose

The purpose of the E-Voting system is to make the voting process more robust, secure, and hassle-free. Arranging physical voting systems can be challenging due to logistical complexities, high costs, and time-consuming procedures. Additionally, traditional voting methods are prone to issues such as vote stealing, tampering, and inaccurate results. This e-voting system addresses these challenges by providing a streamlined, digital solution that ensures security, accuracy, and convenience for both voters and election administrators.

## 1.2 Product scope

The system is not designed to handle nationwide elections but is well-suited for conducting votes at an inter-regional level. It provides essential functionalities such as voter registration, vote casting, withdrawal, and management of voters and candidates. Additionally, the system includes robust features for tallying votes accurately and generating comprehensive reports, ensuring transparency and reliability. These capabilities make it a practical and secure solution for small-scale elections.

## 1.3 Objective

The objective of the E-Voting System is to provide a secure and reliable platform for inter-regional elections, ensuring the integrity of votes and preventing tampering or inaccuracies. It aims to streamline election processes through efficient voter registration, vote casting, candidate management, accurate vote tallying, and comprehensive report generation, making elections more transparent, accessible, and hassle-free.

## 1.4 Problem Statement

Traditional voting methods face challenges such as logistical complexities, high costs, vote tampering, and inaccurate results. These issues highlight the need for a secure and efficient solution that simplifies the voting process, ensures accuracy, and enhances trust in election outcomes.

# Requirements

## 2.1 Functional Requirements

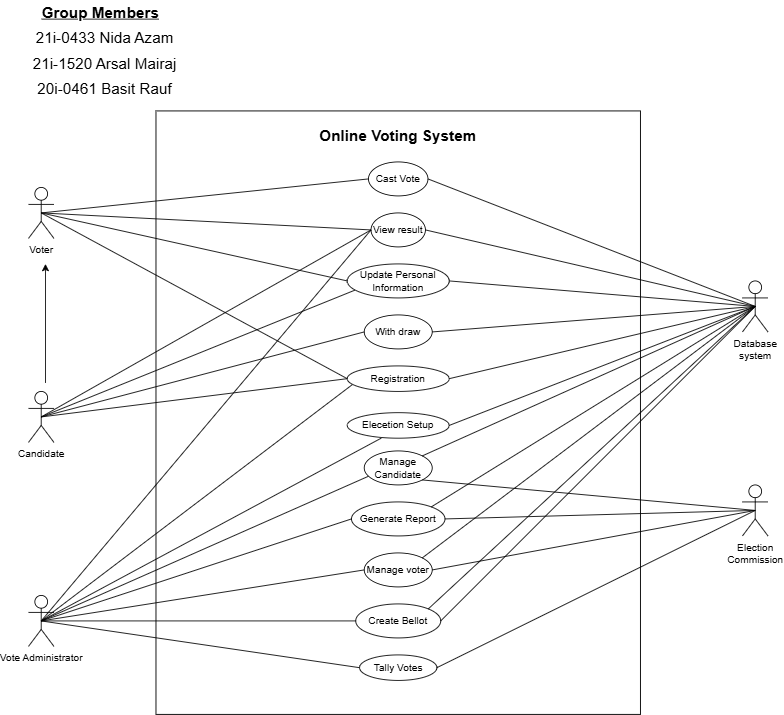
The E-Voting System includes key functional requirements such as **User Management** for registering and managing voters and candidates, and **Election Setup** for configuring election details. It allows users to **cast votes**, **withdraw** votes if necessary, and ensures **vote tallying** is accurate and transparent. Additionally, the system generates **reports** to provide clear insights into election results, ensuring accountability and ease of access to important data.

## 2.2 Non-Functional Requirements

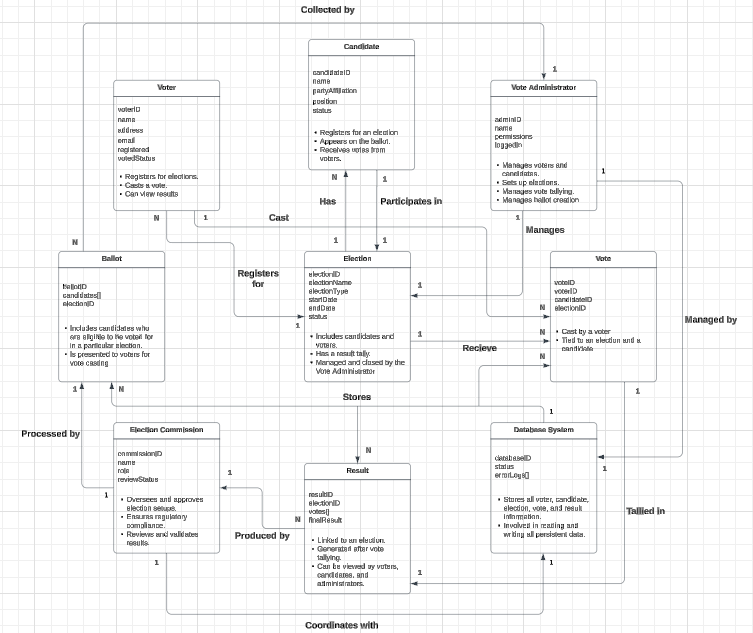
The E-Voting System must ensure **Performance** by providing fast, responsive interactions even under heavy loads. **Scalability** is essential, allowing the system to grow and handle increasing voter numbers or multiple elections without compromising efficiency. **Security** is a priority to safeguard voter data and prevent tampering, while **Availability** ensures the system is accessible at all times with minimal downtime. The system must also adhere to relevant **Compliance** standards and provide an intuitive, user-friendly interface for ease of use and better **Usability** for both voters and administrators.

# System Design Model

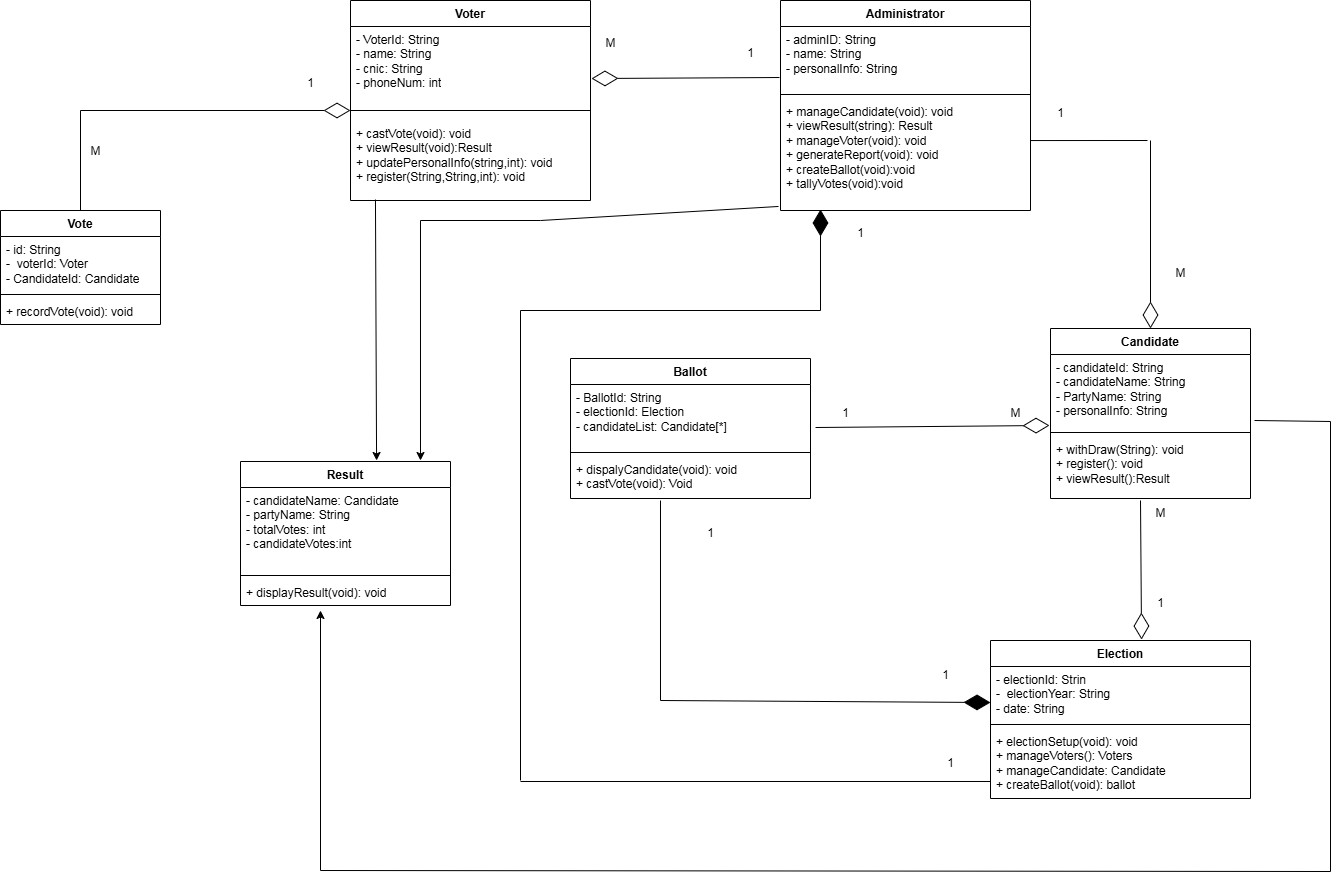
## 1.1 Use cases



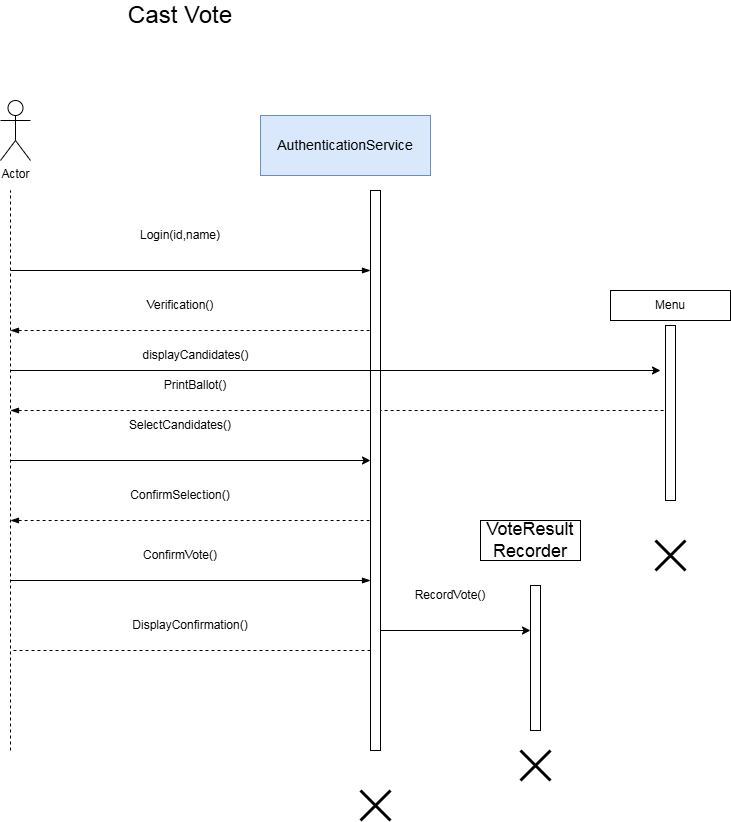
## 1.2 Domain Model

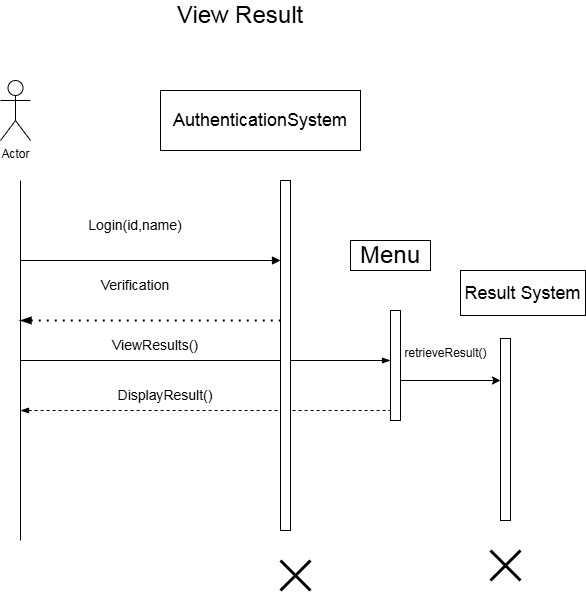


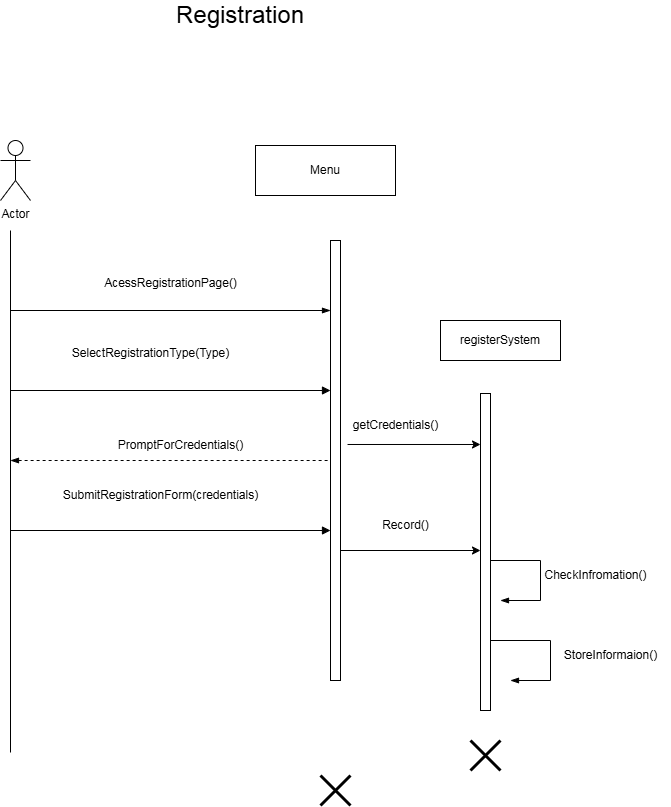
## 1.3 Class Diagram



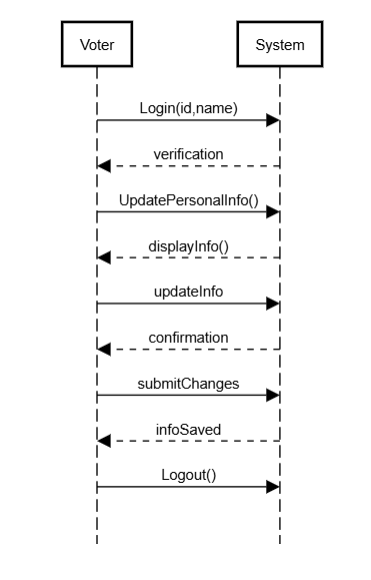
## 1.4 System Sequence Diagram



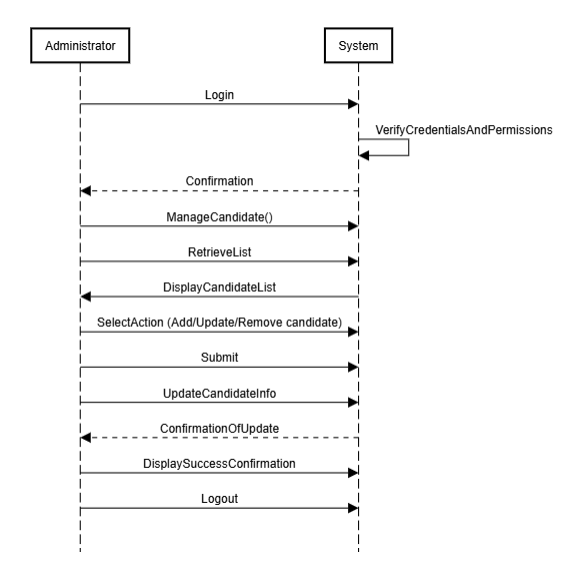




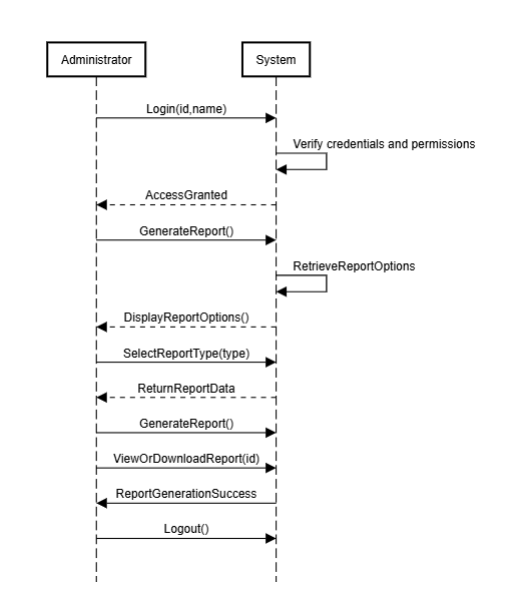
### Update Personal Information



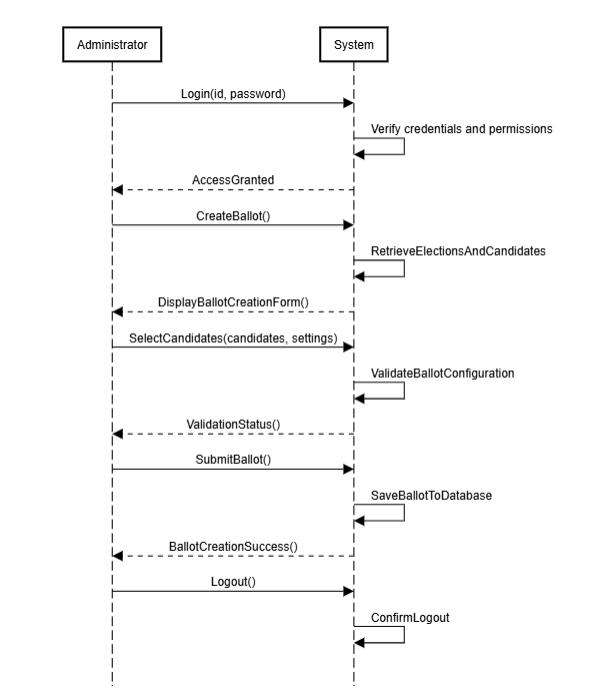
### Manage Candidates



### Generate Report

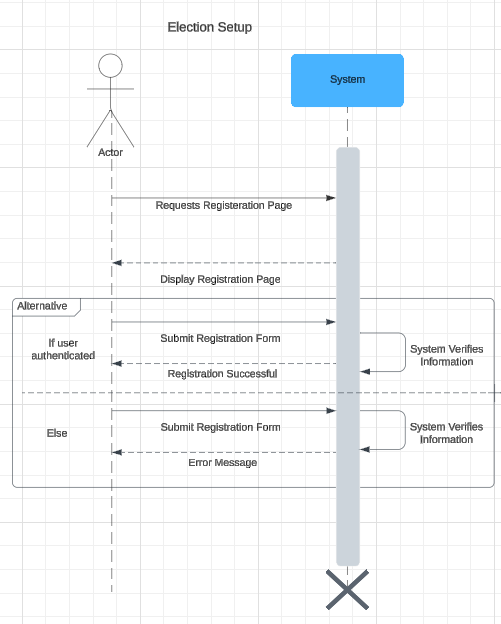


### Create Ballot



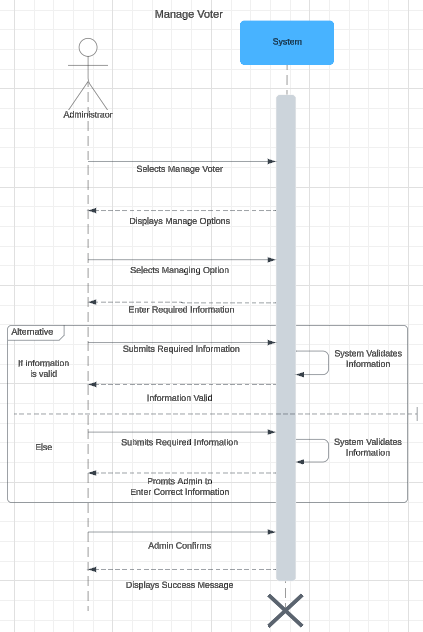
### Election Setup

### 



### 

### Manage Voter



### 

### 

### Tally Votes

