**2021 Spring CPSC 5360 Software Engineering**

Project Report on

**“Simulating a voting system in the United States”**

**Instructor:**

Dr. Stefan Andrei

**Date: 05/03/2021**

**Submitted By:**

**Group Members:**

**Calvin Medley (L20121365)**

**Bilal Mahmood (L)**

**Matina Shrestha (L20523584)**

Table of Contents:

1. Introduction
2. Business model
3. Use Case Diagram
4. Sequence Diagram
5. Domain Model
6. Analysis Class Diagram
7. State Chart Diagram
8. State Chart Diagram of this Project
9. Design Pattern
10. Output
11. Discussion and Conclusion

Introduction:

In this project, we want to create a voting system where the official admin can add new voters, new candidates, double-check that no one is registered and did not cast their votes more than once, put all this information in a database and update it in timely manner. The system will also be able to display the voters list, the candidate list from the database, generate the final review of how many votes were casted and for which candidate, present the result to the public. The following are the basic models used to represent the project.

Business Model:

The business model is defined as the representation of the business that is performed in an organization. It comprises of significant entities and represent their work and interaction. It is also regarded as abstract representation of an organization depicted visually. Use case diagrams, class diagrams, flow charts etc. are necessary in a workable business model.Diagram

Description automatically generated

This figure is an entity diagram of the business model where important entities and their prime activities are shown.

Here in the diagram, the general public or a person inputs their name, their id, search for their desired candidates and cast their votes. The administrator checks if the voter is legitimate or not and that they have submitted their vote only once and validates the information and update the database which will also update the information on the website.

**Use Case Diagram**

Use case diagram represents the user’s interaction with the system that show the relationship between the user and different use cases in which the user is involved. This shows actions assigned/performed by main components of the system. Such main components are regarded as Actors.

Diagram

Description automatically generated

Diagram

Description automatically generated

The use case diagrams show the User role & System role. There are two Users namely the Admin and the Voter. The Admin does the activity of credential verification and can access important information and users cast their votes and check the candidates. While the system will register the new voter, new candidates and updates the database.

**Sequence Diagram**

Sequence diagram is an interaction diagram that shows how objects operate with each other and in what order. Basically it is ordering of the steps performed during the application cycle. It is the visual representation of the system’s work flow. It can also be termed as WorkFlow Architecture.

Diagram

Description automatically generated

The basic steps include 🡪

1. The admin adds new candidate and selects a position.
2. He also adds a new voter and calculate the voter ID to verify it is not a duplicate voter
3. Then the system records the vote and checks if it is a duplicate vote.
4. The voter database is updated.
5. The system also displays the candidate list.
6. The system creates report for result.
7. The general public can access vote information/results from the website.

Diagram

Description automatically generated

Fig: Domain Model

**Analysis Class Diagram**

Analysis Class Diagram depicts the relationship among various entities of the system. Such diagram is used to obtain a conceptual model of the entities used in the application. This diagram gives the complete details of the entities and their tasks.

**State Chart Diagram**

A State chart diagram describes a state machine. Now to clarify it state machine can be defined as a machine which defines different states of an object and these states are controlled by external or internal events.

State-chart defines the behavior of instances of a given class. It shows the possible states of an object, the events it can detect, and its response to those events. State diagrams are composed of a finite number of states.

**State Chart Diagram of this Project**

![Diagram

Description automatically generated]()

Output:

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Conclusion:

In conclusion, a complete voting system can be designed after passing various phases and steps and it can be deployed in real time situations too.