JAVA PROJECT REPORT

(Project Term Jan-May 2023)

COURSE CODE:- CSE310

"School of Computer Science and Engineering"

Title:- Revolutionizing Democracy: The Power of Online Voting.

SUBMITTED BY :-

NAME	REGISTRATION NO.	ROLL NO.
AMIT GUPTA	12112447	RK21WYB48

SUBMITTED TO:- Dr.A.Ranjith Kumar

DECLARATION

We declare that minor project entitled **“ONLINE VOTING SYSTEM”** is our own work conducted under the supervision of **“”**, Department of Information Technology, **Lovely Professional University.** We further declare that, to the best of our knowledge the project does not contain any work which has been submitted for the award of the degree either in the University or in any other University/Deemed University without proper citations.

Name of Student 1: - Anirudh harsh
Registration Number: - 12111207

Name of Student 2: - Vishal Kumar Singh
Registration Number: - 12104228

Name of Student 3: - Amit Gupta
Registration Number: 12112447

(Anirudh Harsh)
Date:26-04-23

(Vishal Kumar Singh)
Date: 26-04-23

(Amit Gupta)
Date: 26-04-23

TABLE OF CONTENTS

1. ABSTRACT

The Online Voting System is a web-based application. The system has a centralized database to keep records of all the Voters and Candidates and Final Results. This Online Voting System is based on SMS sending to voters, to confirmation of Vote. This web-based system is time saving, workload reduced information available at time and it provide security for the data. During the election, the election commission of India has introduced a new method of polling by online voting system (OVS). The election commission will maintain this website. This is a simple, safe, and secure method that takes minimum of time. The word VOTE means to choose from a list, to elect or to determine. The main goal of voting (in a scenario involving the citizens of a given country) is to come up with leaders of the people's choice. Most countries, India not an exception have problems when it comes to voting. Some of the problems involved include ridging votes during election, insecure or inaccessible polling stations, inadequate polling materials and inexperienced personnel.

2. INTRODUCTION

India has democratic government. As now all Indian citizens become a part of the growing digital India .They have a digital ID that is Aadhar card. Voting schemes have evolved from counting hands in early days to systems that include paper, punch card, electronic voting machine. An electronic voting system which is used nowadays provide some characteristic different from the traditional voting technique, and it provides improved features of voting system over traditional voting system such as accuracy, convenience, flexibility, privacy, verifiability, and mobility. But Electronic voting systems suffers from various drawbacks such as time consuming, consumes large volume of paperwork, no direct role for the higher officials, damage of machines due to lack of attention, mass update doesn't allow users to update and edit many items simultaneously etc. These drawbacks can overcome by Online Voting System. This is a voting system by which any voter can use his/her voting rights from anywhere in the country. Voter can cast their votes from anywhere in the country without visiting to voting booths, in highly secured way. That makes voting a fearless of violence and that increases the percentage of voting.

3. ABOUT PROJECT

The objective of the system is a replacement of the traditional system that is in existence. This smart system reduces the time for voting and the system is reliable, and faster. In this system the voter username and password will be sent through SMS. The voter cast their vote enter the confirmation OTP sent their mobile number . Database maintained by this system usually contains the Voters information, Candidate information, The result of total votes.

4. SCOPE OF PROJECT

Online Voting System has a good scope in future due to following reasons:

- Voter can Vote from anywhere for his/her Constituency.
- Vote count will make easy and fast.
- Invalid Vote will be rejected.
- It Maintains all The Information of all the Candidates and Votes.
- It checks Voter have Voted or Not.
- You can observe All Information Related to any Voting System Online.
- It Increase the Voting Percentage.
- Finally, it makes Easy Voting by Avoiding problems like Security, Booth capturing. The actual purpose of going for this system is to make the organizational process to get speed up.

3. MODULES

3.1 ADMINISTRATIVE MODULE

Online Voting is a voting system by which any Voter can use his\her voting rights from anywhere in India. Online voting for association contains-:

- Voter's information in database.
- Voter's Names with ID.
- Voter's vote in a database.
- Calculation of total number of votes

Various operational works that are done in the system are:-

- Recording information of the Voter in Voter database.
- Checking of information filled by voter.
- Discard the false information.
- Each information is maintained by admin.

3.2 USER/VOTER MODULE

The user after their registration only can login for voting. The user will view nominee details with their image before they can vote. After knowing the nominee details the user can login for voting. They should vote for board of director and the manager in the association. The count will take for each voting. After voting the person/user cannot logon to vote again.

4. DESIGN

4.1 Data Flow Diagram

The data flow diagram(DFD) is a graphical tool used for expressing system requirements in a graphical form. The DFD also known as the "bubble chart" as the purpose of clarification system requirements and identification major transformation that will become program in

system design. Thus, DFD can be stated as the starting point of the design phase that functionality decomposes the requirements specification down to the lowest level of details. The DFD consists of series of bubble joined by lines. The bubble represents data transformation, and the lines represents the data flows in the system. A DFD describes what data flow is does not to construct a Data Flow Diagram, we use.

•Arrow: An arrow identifies the data flow in motion. It is a pipeline through which information is flow like the rectangle in the flowchart.

•Circle: A circle stands for process that converts data into information.

• Open End Box: An open-ended box represents a data store, data at rest or a temporary repository of data.

• Squares: A square defines a source or destination of system.

Level 0

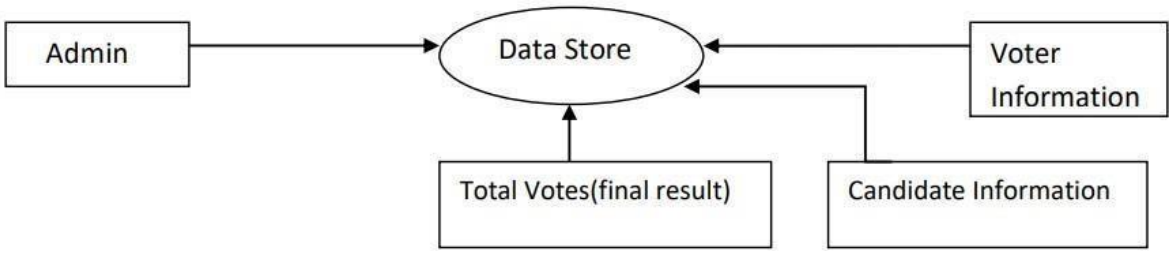


Fig-1.1

5. CONCLUSION

This online Voting system will manage the Voter's information by which voter can login and use his voting rights. The system will incorporate all features of voting system. It provides the tools for maintaining voter's vote to every party and it count total no. of every party. There is a DATABASE which is maintained by the ELECTION COMMISION OF INDIA in which all the names of voter with complete information are stored. In this user who is above 18years's register his/her information on the database and when he/she want to vote he/she has to login by his id and password and can vote to any party only single time. Voting detail store in database and the result is displayed by calculation. By online voting system percentage of voting is increases. It decreases the cost and time of voting process. It is very easy to use, and it is very less time consuming. It is very easy to debug. The traditional method of manual voting system has few drawbacks. This method is obviously not efficient as it wastes the voter's energy and quite slow in term of completion. This smart system involves the voters can cast their vote easily and can be implemented to the entire India.

6. SAMPLE CODE

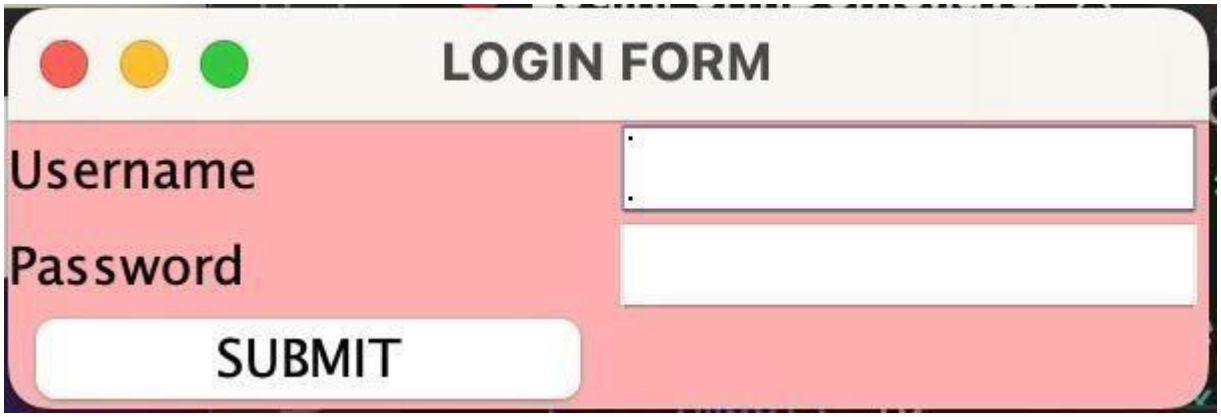
```
import javax.swing.*;
import javax.swing.border.Border;
import javax.swing.plaf.BorderUIResource;
```

```
import java.awt.*;
import java.awt.event.*;
import java.lang.Exception;

//create CreateLoginForm class to create login form
//class extends JFrame to create a window where our component add
//class implements ActionListener to perform an action on button click
class CreateLoginForm extends JFrame implements ActionListener
{
   //initialize button, panel, label, and text field
   JButton b1;
   JPanel newPanel;
   JLabel userLabel, passLabel;
   final JTextField  textField1, textField2;
   //Border border2 = BoxLayout
   //calling constructor
   CreateLoginForm()
   {
      //create label for username
      userLabel = new JLabel();
      userLabel.setText("Username");      //set label value for textField1

      //create text field to get username from the user
      textField1 = new JTextField(15);    //set length of the text

      //create label for password
      passLabel = new JLabel();
      passLabel.setText("Password");      //set label value for textField2

      //create text field to get password from the user
      textField2 = new JPasswordField(15);    //set length for the password

      //create submit button
      b1 = new JButton("SUBMIT"); //set label to button
                                  "); //set label to button

      //create panel to put form elements
      newPanel = new JPanel(new GridLayout(3, 1));
      //newPanel.setBounds(400, 300, 300, 300);
      newPanel.setBackground(Color.PINK);
      newPanel.add(userLabel);    //set username label to panel
      newPanel.add(textField1);   //set text field to panel
```

```
        newPanel.add(passLabel);    //set password label to panel
        newPanel.add(textField2);   //set text field to panel
        newPanel.add(b1);           //set button to panel

        //set border to panel
        add(newPanel, BorderLayout.CENTER);

        //perform action on button click
        b1.addActionListener(this);     //add action listener to button
        setTitle("LOGIN FORM");  //set title to the login form

    }

    //define abstract method actionPerformed() which will be called on button click
    public void actionPerformed(ActionEvent ae)     //pass action listener as a parameter
    {
        String userValue = textField1.getText();        //get user entered username from the
textField1
        String passValue = textField2.getText();        //get user entered pasword from the
textField2

        //check whether the credentials are authentic or not
        if (userValue.equals("test1@gmail.com") && passValue.equals("test")) { //if authentic,
navigate user to a new page

            //create instance of the NewPage
            VotingManagementSystem votingSystem1 = new VotingManagementSystem();
            //make page visible to the user
            votingSystem1.setVisible(true);

            //create a welcome label and set it to the new page
            JLabel wel_label = new JLabel("Welcome: "+userValue);
            votingSystem1.getContentPane().add(wel_label);
        }
        else{
            //show error message
            System.out.println("Please enter valid username and password");
        }
    }
}
//create the main class
```

```
class LoginForm
{
   //main() method start
   public static void main(String arg[])
   {
      try
      {
         //create instance of the CreateLoginForm
         CreateLoginForm form = new CreateLoginForm();
         form.setSize(500,300);  //set size of the frame

         //setLocationRelativeTo(null);
         //form.setBounds(3000, 3000, 1000, 1000);
         form.setVisible(true);  //make form visible to the user
      }
      catch(Exception e)
      {
         //handle exception
         JOptionPane.showMessageDialog(null, e.getMessage());
      }
   }
}
```

LOGIN OUTPUT




```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
import java.awt.Image;
```

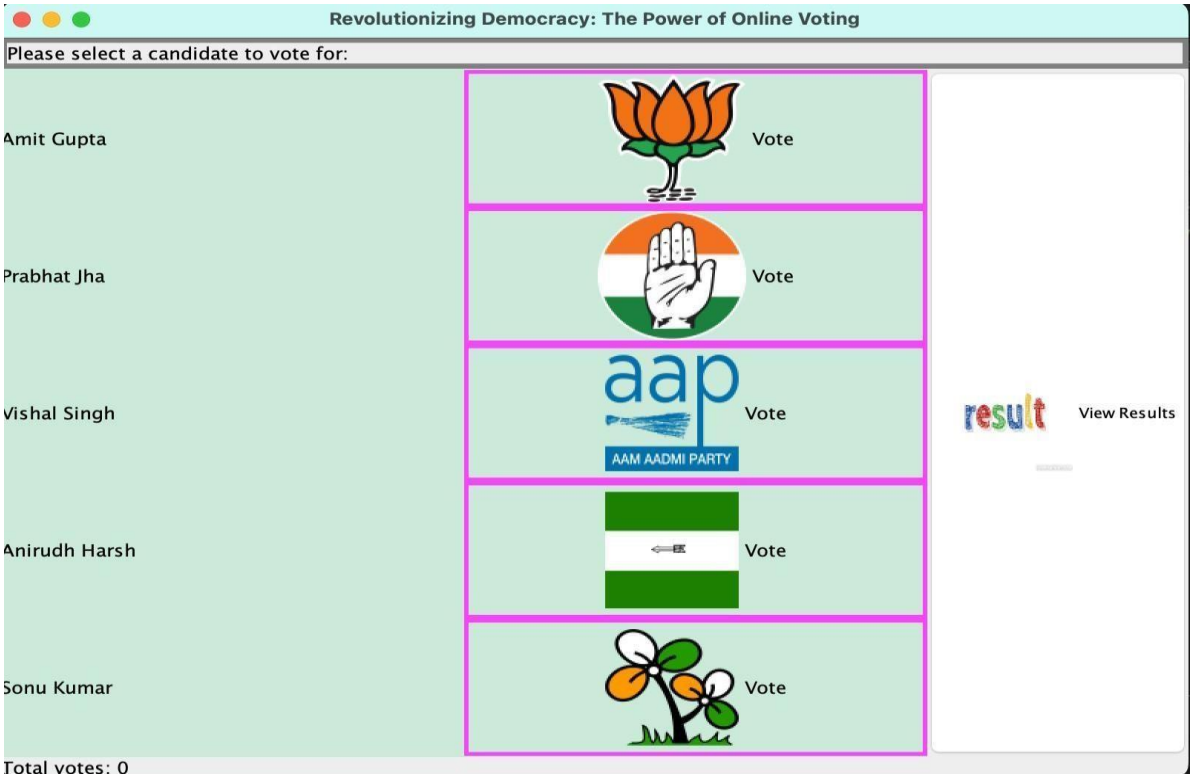
```
import javax.swing.ImageIcon;
import  javax.swing.JFrame;
import javax.swing.border.Border;

public class VotingManagementSystem extends JFrame {


    // This array stores the number of votes for each candidate
    private int[] candidateVotes;

    // This array stores the names of the candidates
    private String[] candidateNames;

    // This variable stores the total number of votes
    private int totalVotes;

    // These are the GUI components
    private JLabel titleLabel;
    private JPanel candidatePanel;
    private JLabel[] candidateLabels;
    private JButton[] voteButtons;
    private JLabel totalVotesLabel;
    private JButton resultsButton;

    // This constructor initializes the arrays and variables

    public VotingManagementSystem(String[] names) {
        candidateNames = names;
        candidateVotes = new int[names.length];
        totalVotes = 0;
        // JFrame Frame = new JFrame();
        // Frame.getContentPane().setBackground(new Color(255,123,0));
        // Set up the GUI
        setTitle("Voting Management System");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setLayout(new BorderLayout());

        setBackground(Color.CYAN);
        setBounds(600, 500, 500, 500);
        setPreferredSize(new Dimension(800, 600));
        Border border = BorderFactory.createLineBorder(Color.gray,4);
        Border border1 = BorderFactory.createLineBorder(Color.MAGENTA,3);

        titleLabel = new JLabel("Please select a candidate to vote for:");
        titleLabel.setBackground(Color.PINK);
        titleLabel.setBorder(border);
        add(titleLabel, BorderLayout.NORTH);

        //JFrame Frame = new JFrame();
        candidatePanel = new JPanel(new GridLayout(names.length, 2));
        candidateLabels = new JLabel[names.length];
        voteButtons = new JButton[names.length];
```

```
        //voteButtons.setBackground(new Color(244,211,011));
        ImageIcon j=new ImageIcon("b.png");
         Image image = j.getImage(); // transform it
         Image newimg = image.getScaledInstance(100, 100,
java.awt.Image.SCALE_SMOOTH); // scale it the smooth way
         j = new ImageIcon(newimg);

         ImageIcon z=new ImageIcon("c.png");
         Image image1 = z.getImage(); // transform it
         Image newimg1 = image1.getScaledInstance(100, 100,
java.awt.Image.SCALE_SMOOTH); // scale it the smooth way
         z = new ImageIcon(newimg1);

         ImageIcon z1=new ImageIcon("a.png");
         Image image2 = z1.getImage(); // transform it
         Image newimg2 = image2.getScaledInstance(90, 90,
java.awt.Image.SCALE_SMOOTH); // scale it the smooth way
         z1 = new ImageIcon(newimg2);

         ImageIcon z2=new ImageIcon("j.png");
         Image image3 = z2.getImage(); // transform it
         Image newimg3 = image3.getScaledInstance(90, 90,
java.awt.Image.SCALE_SMOOTH); // scale it the smooth way
         z2 = new ImageIcon(newimg3);

         ImageIcon z3=new ImageIcon("t.png");
         Image image4 = z3.getImage(); // transform it
         Image newimg4 = image4.getScaledInstance(90, 90,
java.awt.Image.SCALE_SMOOTH); // scale it the smooth way
         z3 = new ImageIcon(newimg4);

         ImageIcon z4=new ImageIcon("r.png");
         Image image5 = z4.getImage(); // transform it
         Image newimg5 = image5.getScaledInstance(90, 90,
java.awt.Image.SCALE_SMOOTH); // scale it the smooth way
         z4 = new ImageIcon(newimg5);



        candidatePanel.setBackground(new Color(255,123,5));

        //Frame.setVisible(true);

        for ( int i=0; i < names.length; i++) {
            candidateLabels[i] = new JLabel(names[i]);
            candidatePanel.add(candidateLabels[i]);
            voteButtons[i] = new JButton("Vote");
            voteButtons[i].setBackground(new Color(255,011,101));
            voteButtons[i].setBorder(border1);
```

```
            int candidateIndex = i;
            voteButtons[i].addActionListener(new ActionListener() {
                public void actionPerformed(ActionEvent e) {
                    candidateVotes[candidateIndex]++;
                    totalVotes++;

                    JOptionPane.showMessageDialog(null, "Thank you for voting for " +
candidateNames[candidateIndex] );

                    updateTotalVotesLabel();
                }
            });
            candidatePanel.add(voteButtons[i]);

        }
        for(int a=0;a< names.length;a++)
            {
                voteButtons[0].setIcon(j);
                voteButtons[1].setIcon(z);
                voteButtons[2].setIcon(z1);
                voteButtons[3].setIcon(z2);
                voteButtons[4].setIcon(z3);
            }

        add(candidatePanel, BorderLayout.CENTER);

        totalVotesLabel = new JLabel("Total votes: " + totalVotes);
        add(totalVotesLabel, BorderLayout.SOUTH);

        resultsButton = new JButton("View Results");
        resultsButton.setIcon(z4);
        resultsButton.setBackground(Color.ORANGE);
        resultsButton.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                String results = "Voting Results:\n";
                for (int i = 0; i < candidateNames.length; i++) {
                    results += candidateNames[i] + ": " + candidateVotes[i] + " votes\n";
                }
                results += "Total votes: " + totalVotes;


                JOptionPane.showMessageDialog(null, results);


            }
        });
        add(resultsButton, BorderLayout.EAST);

        pack();
        setVisible(true);
    }
```

```
    // This method updates the total votes label
    private void updateTotalVotesLabel() {

        totalVotesLabel.setText("Total votes: " + totalVotes);
        totalVotesLabel.setBackground(Color.BLACK);
    }

    // This is the main method that creates and runs the voting management system
    public static void main(String[] args) {
        String[] candidateNames = {"Amit Gupta", "Vishal Singh", "Prabhat Jha ", "Anirudh
Harsh", "Sonuuuu"};
        VotingManagementSystem votingSystem = new
VotingManagementSystem(candidateNames);
    }

}
```

9. OUTCOME OF THE PROJECT




Message
Voting Results:
Amit Gupta: 3 votes
Prabhat Jha: 2 votes
Vishal Singh : 0 votes
Anirudh Harsh: 0 votes
Sonu Kumar: 0 votes
Total votes: 5
OK