



UNITEDWORLD SCHOOL OF COMPUTATIONAL INTELLIGENCE (USCI)

**Summative Assessment (SA)
Online Voting System**

Submitted BY
Viken Patel
(Enrl. No.: 20220701033)

Course Code and Title: 21BSCS24C05-Professional Training

**B.Sc. (Hons.) Computer Science / Data Science /
AIML**

IV Semester – Dec 2023 – April 2024

Dec/Apr 2024

Index

1. Introduction	3
2. Explanation	4
3. Project Code	6
4. Output	9
5. Conclusion	11
6. Bibliography	13

Introduction

Ghost voters are people who do not really exist but whose votes are influenced by powerful parties in an attempt to influence election outcomes in their favor. This is a problem that is often observed in India. By offering a personalized and safe voting experience, the suggested software solution seeks to preserve the election process's integrity. Every qualified voter will be given the chance to cast their ballots within this project during a predetermined window of time, free from outside interference. The poll will only be open to Indian nationals who are at least eighteen years old, and each voter will need to authenticate themselves with a unique user ID and password. The capacity for voters to choose their particular assembly districts and voting locations, guaranteeing a legal and private casting of their ballots in favor of their preferred leaders.

The project discusses India's requirement to switch from a manual to an online voting system, utilizing cutting-edge technology to increase efficiency, speed, and accuracy. This thesis proposes an online approach in response to issues that the manual system has caused for voters and the government. Voters would have the option to use government-provided resources or their own devices for voter registration, voting, result computation, and result declaration under this proposed system. The study also highlights how such a change might help the democratic process in the long run by lowering the possibility of corruption connected to the existing manual procedure.

In order to emphasize a fundamental need of an online voting system or tech-based system that would allow transparency and security, a Java-based script has been built. However, my proposal is a learning model (beginning) that will take a lot of effort to put into practice. It is limited to Java, but there are other systems that are fast, secure, and have high-end interfaces that make it possible to utilize programming languages and create apps.

Explanation

"Vote" refers to the process of choosing, choosing representatives, and choosing results. Like in many other countries, the main goal of voting in India is to democratically choose leaders who represent the people's desire. However, much as in other nations, voting in India is fraught with difficulties, such as election fraud, problems with polling place accessibility and security, a lack of necessary voting supplies, and a staffing shortfall. The Smart Voting System is made to successfully address these issues. This novel voting mechanism is designed to provide Indian voters enough time to cast their ballots. In order to guarantee that voters are ready for online voting prior to the election, the system also provides user training.

A simple Java programming system for online voting is implemented in the code below. The code consists of a single function named "main," which serves as the program's entry point, and a single class named "OnlineVotingSystem."

The code generates three integer variables, "Name, Phone, and Party or team" and "SubmitYourVote and CheckResult," and initializes them to see the result at the end of the main procedure. The total number of votes cast for each contender will be stored in these variables.

The user is then prompted to enter their name and age by the code using a Scanner object to read input from them. The user's age is then verified to see if they are older than eighteen (18) and so entitled to vote. The code asks the user to choose a candidate to vote for after displaying a list of candidates if they are qualified to vote. The variable "userSelection" contains the selection made by the user.

The code then checks the value of "userSelection" and increases the vote total for the associated candidate using an if-else expression. After then, the user receives a thank-you message from the computer along with the total number of votes cast for each contender. When a user is not qualified to vote, the application notifies them of this by displaying a message. In order to free up system resources, the code finally terminates the Scanner object.

Remember that this code is only an illustration of a simple online voting system; in actuality, scalability and security are significant considerations that must be considered when putting an online voting system into place. Furthermore, this code uses the console to receive inputs. In a real-world setting, the system ought to have an easy-to-use interface, store voter data in a database, encrypt sensitive data, employ secure protocols, and have adequate session management.

Code

```
J main_page.java
1  import javax.swing.*;
2  public class main_page extends javax.swing.JFrame {
3      public static int p1 = 0;
4      public static int p2 = 0;
5      public static int p3 = 0;
6      public main_page() {
7          initComponents();
8      }
9
10     private void initComponents() {
11         jButton2 = new javax.swing.JButton();
12         jLabel1 = new javax.swing.JLabel();
13         jLabel2 = new javax.swing.JLabel();
14         jTextField1 = new javax.swing.JTextField();
15         jLabel3 = new javax.swing.JLabel();
16         jTextField2 = new javax.swing.JTextField();
17         jButton1 = new javax.swing.JButton();
18         jButton2 = new javax.swing.JButton();
19         jButton3 = new javax.swing.JButton();
20         jLabel4 = new javax.swing.JLabel();
21         jButton1 = new javax.swing.JButton();
22         jButton4 = new javax.swing.JButton();
23         jButton2.setText(text:"RESEND");
24         jButton2.addActionListener(new java.awt.event.ActionListener() {
25             public void actionPerformed(java.awt.event.ActionEvent evt) {
26                 jButton2ActionPerformed(evt);
27             }
28         });
29         setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
30         jLabel1.setFont(new java.awt.Font(name:"Tahoma", style:1, size:24)); // NOI18N
31         jLabel1.setText(text:"ONLINE VOTING SYSTEM");
32         jLabel2.setText(text:"ENTER NAME");
33         jLabel3.setText(text:"ENTER PHONE");
34         jButton1.setText(text:"PARTY A");
35         jButton1.addActionListener(new java.awt.event.ActionListener() {
36             public void actionPerformed(java.awt.event.ActionEvent evt) {
37                 jButton1ActionPerformed(evt);
38             }
39         });
40
41         public static void main(String args[]) {
42             java.awt.EventQueue.invokeLater(new Runnable() {
43                 public void run() {
44                     new main_page().setVisible(b:true);
45                 }
46             });
47         }
48
49         // Variables declaration
50         private javax.swing.JButton jButton1;
51         private javax.swing.JButton jButton2;
52         private javax.swing.JButton jButton4;
53         private javax.swing.JLabel jLabel1;
54         private javax.swing.JLabel jLabel2;
55         private javax.swing.JLabel jLabel3;
56         private javax.swing.JLabel jLabel4;
57         private javax.swing.JRadioButton jButton1;
58         private javax.swing.JRadioButton jButton2;
59         private javax.swing.JRadioButton jButton3;
60         private javax.swing.JTextField jTextField1;
61         private javax.swing.JTextField jTextField2;
62
63         // End of variables declaration
64     }
65 }
```

```

145 private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {
146     if (jRadioButton3.isSelected()){
147         jRadioButton2.setSelected(b:false);
148         jRadioButton1.setSelected(b:false);
149     }
150 }
151 private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
152     if(jRadioButton1.isSelected()){
153         p1++;
154     }
155     else if(jRadioButton2.isSelected()){
156         p2++;
157     }
158     else if(jRadioButton3.isSelected()){
159         p3++;
160     }
161     else{
162         JOptionPane.showMessageDialog(rootPane, message:"Select a Party");
163     }
164     jTextField1.setText(t:"");
165     jTextField2.setText(t:"");
166     jRadioButton1.setSelected(b:false);
167     jRadioButton2.setSelected(b:false);
168     jRadioButton3.setSelected(b:false);
169 }
170 private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {
171     if(p1>p2 && p1>p3){
172         JOptionPane.showMessageDialog(rootPane, "Party A "+p1+"\nParty B "+p2+"\nParty C "+p3+"\n\nPARTY A has a lead");
173     }
174     else if(p2>p1&&p2>p3){
175         JOptionPane.showMessageDialog(rootPane, "Party A "+p1+"\nParty B "+p2+"\nParty C "+p3+"\n\nPARTY B has a lead");
176     }
177     else if(p3>p1&&p3>p2){
178         JOptionPane.showMessageDialog(rootPane, "Party A "+p1+"\nParty B "+p2+"\nParty C "+p3+"\n\nPARTY C has a lead");
179     }
180 }

```

```

109 .addComponent(jLabel3, javax.swing.GroupLayout.PREFERRED_SIZE, pref:28, javax.swing.GroupLayout.PREFERRED_SIZE)
110 .addComponent(jTextField2, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.
111 .addGroup(min:20, pref:20, max:20)
112 .addComponent(jLabel4)
113 .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
114 .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
115     .addGroup(layout.createSequentialGroup()
116         .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
117             .addComponent(jButton1)
118             .addComponent(jButton1))
119         .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
120         .addComponent(jRadioButton2)
121         .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
122         .addComponent(jRadioButton3))
123     .addGroup(layout.createSequentialGroup()
124         .addGap(min:41, pref:41, max:41)
125         .addComponent(jButton4)))
126     .addContainerGap(pref:33, Short.MAX_VALUE))
127 );
128 pack();
129 } // </editor-fold>
130 private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
131     // TODO add your handling code here:
132 }
133 private void jRadioButton1ActionPerformed(java.awt.event.ActionEvent evt) {
134     if (jRadioButton1.isSelected()){
135         jRadioButton2.setSelected(b:false);
136         jRadioButton3.setSelected(b:false);
137     }
138 }
139 private void jRadioButton2ActionPerformed(java.awt.event.ActionEvent evt) {
140     if (jRadioButton2.isSelected()){
141         jRadioButton1.setSelected(b:false);
142         jRadioButton3.setSelected(b:false);
143     }
144 }

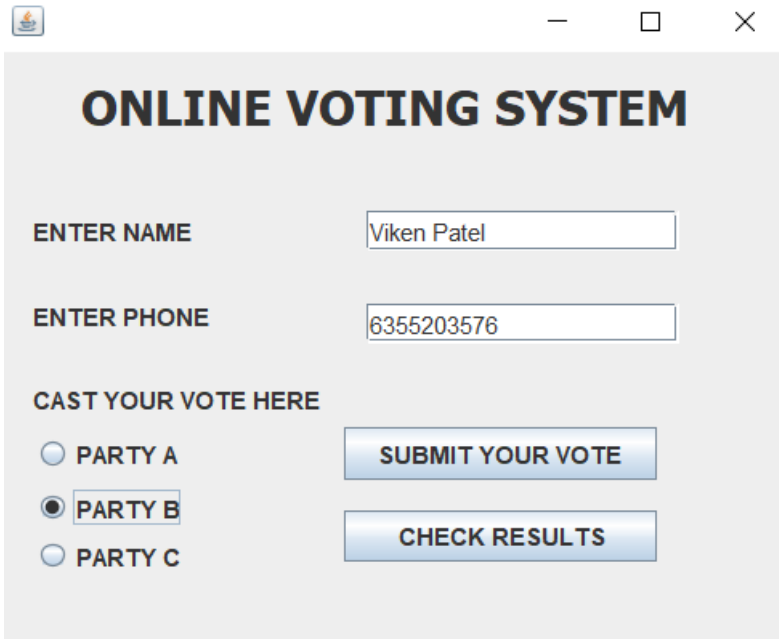
```

```

2 public class main_page extends javax.swing.JFrame {
10 private void initComponents() {
73 .addComponent(jLabel1)
74 .addGroup(layout.createParallelGroup().
75 .addGroup(layout.createSequentialGroup().
76 .addComponent(jLabel3, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
77 .addComponent(jLabel2, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
78 .addComponent(jTextField1)
79 .addComponent(jTextField2, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
80 .addComponent(jLabel4)
81 .addGroup(layout.createSequentialGroup().
82 .addComponent(jRadioButton1)
83 .addComponent(jRadioButton2)
84 .addComponent(jRadioButton3)
85 .addPreferredGap(javax.swing.GroupLayout.Alignment.LEADING, Short.MAX_VALUE)
86 .addComponent(jButton1, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
87 .addComponent(jButton4, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
88 .addGap(min:16, pref:16, max:16))))
89 .addContainerGap(pref:52, Short.MAX_VALUE)
90 );
91 layout.setVerticalGroup(
92 layout.createParallelGroup().
93 .addGroup(layout.createSequentialGroup().
94 .addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED_SIZE, Short.MAX_VALUE)
95 .addGap(min:31, pref:31, max:31)
96 .addGroup(layout.createParallelGroup().
97 .addComponent(jLabel2, javax.swing.GroupLayout.PREFERRED_SIZE, Short.MAX_VALUE)
98 .addComponent(jTextField1, javax.swing.GroupLayout.PREFERRED_SIZE, Short.MAX_VALUE)
99 .addComponent(jTextField2, javax.swing.GroupLayout.PREFERRED_SIZE, Short.MAX_VALUE)
100 .addComponent(jLabel4, javax.swing.GroupLayout.PREFERRED_SIZE, Short.MAX_VALUE)
101 .addGroup(layout.createSequentialGroup().
102 .addComponent(jRadioButton1)
103 .addComponent(jRadioButton2)
104 .addComponent(jRadioButton3)
105 .addPreferredGap(javax.swing.GroupLayout.Alignment.LEADING, Short.MAX_VALUE)
106 .addComponent(jButton1, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
107 .addComponent(jButton4, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
108 .addGap(min:16, pref:16, max:16))))
109 );
110
111 main_page.java
2 public class main_page extends javax.swing.JFrame {
10 private void initComponents() {
35 jRadioButton1.addActionListener(new java.awt.event.ActionListener() {
38 }
39 });
40 jRadioButton2.setText(text:"PARTY B");
41 jRadioButton2.addActionListener(new java.awt.event.ActionListener() {
42 public void actionPerformed(java.awt.event.ActionEvent evt) {
43 jRadioButton2ActionPerformed(evt);
44 }
45 });
46 jRadioButton3.setText(text:"PARTY C");
47 jRadioButton3.addActionListener(new java.awt.event.ActionListener() {
48 public void actionPerformed(java.awt.event.ActionEvent evt) {
49 jRadioButton3ActionPerformed(evt);
50 }
51 });
52 jLabel4.setText(text:"CAST YOUR VOTE HERE");
53 jButton1.setText(text:"SUBMIT YOUR VOTE");
54 jButton1.addActionListener(new java.awt.event.ActionListener() {
55 public void actionPerformed(java.awt.event.ActionEvent evt) {
56 jButton1ActionPerformed(evt);
57 }
58 });
59 jButton4.setText(text:"CHECK RESULTS");
60 jButton4.addActionListener(new java.awt.event.ActionListener() {
61 public void actionPerformed(java.awt.event.ActionEvent evt) {
62 jButton4ActionPerformed(evt);
63 }
64 });
65 javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
66 getContentPane().setLayout(layout);
67 layout.setHorizontalGroup(
68 layout.createParallelGroup().
69 .addGroup(layout.createSequentialGroup().
70 .addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED_SIZE, Short.MAX_VALUE)
71 .addGap(min:42, pref:42, max:42)
72 );

```


Output



A screenshot of a web application titled "ONLINE VOTING SYSTEM". The interface includes three input fields: "ENTER NAME" with the value "Viken Patel", "ENTER PHONE" with the value "6355203576", and "CAST YOUR VOTE HERE" with three radio buttons labeled "PARTY A", "PARTY B" (which is selected), and "PARTY C". To the right of the radio buttons are two buttons: "SUBMIT YOUR VOTE" and "CHECK RESULTS". The interface is displayed within a browser window with standard navigation icons.

ONLINE VOTING SYSTEM

ENTER NAME

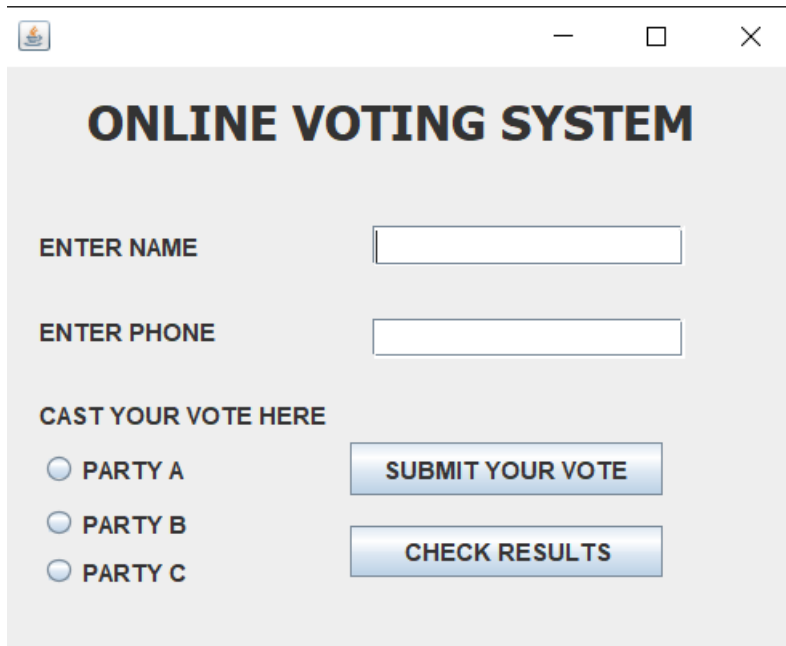
ENTER PHONE

CAST YOUR VOTE HERE

☐ PARTY A

☒ PARTY B

☐ PARTY C



A screenshot of the same "ONLINE VOTING SYSTEM" interface, but with the input fields empty. The "ENTER NAME" and "ENTER PHONE" fields are empty text boxes. The "CAST YOUR VOTE HERE" section shows three radio buttons labeled "PARTY A", "PARTY B", and "PARTY C", none of which are selected. The "SUBMIT YOUR VOTE" and "CHECK RESULTS" buttons are still present. The interface is displayed within a browser window with standard navigation icons.

ONLINE VOTING SYSTEM

ENTER NAME

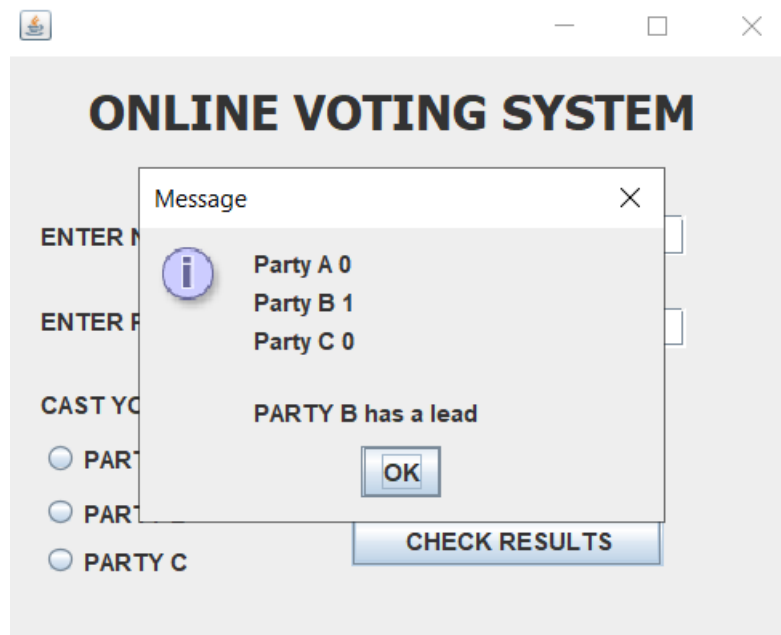
ENTER PHONE

CAST YOUR VOTE HERE

☐ PARTY A

☐ PARTY B

☐ PARTY C



The system is composed of three parties: Party A, Party B, and Party C. All three of the parties are against some `JRadioButtons`. The party vote totals are contained in variables `p1`, `p1`, and `p3`. The user has to provide their name and phone number, select their preferred party by clicking the `JRadioButton`, and then click the `JButton` labelled "Submit Your Vote." Users have to select one party each time they cast a vote. If no party is selected, a dialog pop-up box with the title "Select a Party" appears. The dialog windows are constructed using the `JOptionPane`. We have to make sure that a voter cannot use their vote to support more than one party. This is made feasible by the subsequent tactic:

Conclusion

This online voting system will handle the voter's information, and they may utilize it to log in and use their right to vote. The voting system will have all of its components integrated within it. It calculates the total number of votes cast for each party and provides people with the option to continue voting for each one. Every voter's name and all of their personal details are kept up to date in a database maintained by the Indian Election Commission.

A voter who is at least eighteen years old must first use their ID and password to register their information in the database before they may cast a ballot. After that, they may vote once for any party. A database holds the voting information, and a computation shows the result. The online voting process raises the percentage of votes cast. It reduces the cost and duration of the voting procedure. It's really easy to use and takes very little work. Debugging is really easy.

References

Patidar, Bhupendra. 2022. "Online Voting System Project in Java with Source Code and Project Report." *Codebun*. March 26.

<https://codebun.com/online-voting-system-project-in-java-with-source-code-and-project-report/>.

"PROJECT REPORT ONLINE VOTING SYSTEM." 2012. *SlideShare*. Slideshare. July 28.

<https://www.slideshare.net/wilsonnandasaba/project-report-online-voting-system>.

"Online Voting System Report - [DOCX Document]." 2024. *Vdocument.In*. Accessed April 1. <https://vdocument.in/online-voting-system-report.html>.

"Build Software Better, Together." 2024. *GitHub*. Accessed April 1. <https://github.com/topics/voting-system?l=java>.

"Project Report of Online Voting System." 2024. *Scribd*. Scribd. Accessed April 1. <https://www.scribd.com/document/550446048/Project-Report-of-Online-Voting-System>.

Plagiarism



PLAGIARISM SCAN REPORT

Date April 01, 2024

Exclude URL: NO



Unique Content 100

Plagiarized Content 0

Word Count 919

Records Found 0

CONTENT CHECKED FOR PLAGIARISM:

UNITEDWORLD SCHOOL OF COMPUTATIONAL INTELLIGENCE (USCI)

Summative Assessment (SA)

Submitted BY

Viken Patel

(Enrl. No.: 20220701033)

Course Code and Title: 21BSCS24C05-Professional Training

B.Sc. (Hons.) Computer Science / Data Science / AIML

IV Semester – Dec 2023 – April 2024

center24384000

Dec/Apr 2024

Index

Introduction 3

Explanation 4

Project Code 6

Output 8

Conclusion 9

Bibliography 10

Introduction