

VOTING SYSTEM

Summer Training Project

**BACHELOR OF TECHNOLOGY
COMPUTER SCIENCE AND ENGINEERING
ARTIFICIAL INTELLIGENCE AND MACHING LEARNING**

Submitted

To



**Dr. A.P.J. ABDUL KALAM TECHNICAL UNIVERSITY
LUCKNOW**

SUBMITTED BY

NABEELA HAQ	(2200101530064)
NIMISHA JAISWAL	(2200101530066)
ATUL MAURYA	(2202841530012)
UTKARSH SINGH	(2202840100240)

**UNDER THE SUPERVISION OF
Mr. GAURAV NARAIN SINGH
Assistant Professor**



UNITED INSTITUTE OF TECHNOLOGY, PRAYAGRAJ

Table of Contents

- Introduction
- Literature
- Proposed Project
- Snapshot
- Conclusion

Introduction

In the democratic process, counting votes accurately is a fundamental task that ensures the integrity of elections. With the advent of technology, the manual process of vote counting has evolved into automated systems to expedite and enhance accuracy. This project, "Vote Counting System in Python," aims to create a robust and user-friendly program for tallying votes in various elections, from student council elections to larger-scale political contests.

Project Objective:

The primary objective of this project is to develop a Python-based vote counting system that can handle the following key functionalities:

Vote Collection: The system should allow users to input and store individual votes, ensuring the secrecy and security of each vote.

Vote Tallying: The system will tally the votes, providing real-time updates on the vote counts for each candidate or option.

Result Reporting: After the vote counting process, the system should generate comprehensive reports displaying the final results, including the percentage of votes received by each candidate or option.

User-Friendly Interface: A user-friendly command-line or graphical interface will be designed to make the system accessible to election administrators and users.

Security Measures: Implement security measures to prevent tampering with the vote data and ensure the integrity of the election process.

Benefits:

Efficiency: Automation reduces the time and effort required for manual counting, allowing for quicker election results.

Accuracy: Minimizing human error in the vote tallying process ensures more precise results.

Transparency: The system's reporting feature promotes transparency by providing clear and understandable results to stakeholders.

Scalability: The project can be adapted for various election scales, making it versatile for different scenarios.

Customization: Users can customize the system to fit specific election rules and requirements.

This project will utilize Python's versatility and ease of use to create a reliable and efficient vote counting system that can be employed in a wide range of election scenarios. Through this endeavor, we aim to contribute to the promotion of fair and transparent elections.

Literature

In Indian constitution every individual who is a citizen of India, gets the voting power once he becomes 18years old. There is no maximum age for the elderly people to vote for the country and hence anybody who is 18years of age and is an Indian citizen can vote in their respective booth places whenever the elections are conducted. So, the person who is a candidate who stands for election will appeal for vote and there are many instances while people go to vote the candidates will be present near the voting booths. We have also seen many instances where there are a lot of fights which happen during the election. So, we have to ensure that the election process goes on smoothly without any issues. Being one of the largest democracies in the world, we have been still following the same voting system that is been used since ages. Even now the development is not much seen in terms of technology as earlier we were using the Ballot paper voting and in recent years the country has introduced the EVM machines. And in the recent times and in particular in the last two years where the pandemic has left an everlasting impact on every individual lives, we have to find a solution to ensure that we also have an alternative in times of the pandemic.

In 2016 P. Vidyasree, S. Raju G. Madhavi, “Desisting the Fraud in India's Voting Process through Multi Modal biometrics” This paper provides the conceptual solution through multimodal biometrics which helps in enhancing the security, eradicating the fraud and provides the high level authentication by linking with the Aadhaar card database. High accuracy will be achieved by fusion of face and finger print recognition systems compared to present EVM system. And in Canada’s Ontario there was a election which was held online in recent years.

On the day of elections each and every individual who belongs to that particular district or state returns to their voting booth place or city, and gives vote to the particular party that he wants them to come in power. Many candidates lure the voters with different personal benefits so that they get the votes from them, and every party will have a manifesto, agenda and different schemes announced that those all schemes will be put into action when their party comes into power. So ultimately, when the voting days approach many people who stays in different in search of better jobs/lifestyle, return to their native just to cast their vote taking a pay cut from their employers for a good 3-4 days. Even though voting is a fundamental right of every individual, they still have to travel on their own to the place and the employers doesn’t have any special leave allocation for the purpose to be served.

Proposed Project

In a country with a billion people who vote, it is definitely a time consuming and a challenging task for the government as well to conduct the election. A lot of funding and money is required to get the election procedure completed. Right from voting booths, to the registration kiosks and the people who perform the election duties, every aspect is time consuming and involves money. Once the election is finished the counting process starts and in few cases the counting goes on for entire day, so we are proposing a very efficient, simple and easy to use online voting portal, where an individual can vote from the comfort of their homes, by logging into their portal, and one time registration of the voters can be done for voting of different elections, in all levels, i.e. Village, Panchayat, district, state, and Central elections. Properly implemented, e-voting solutions can eliminate certain common avenues of fraud, speed up the processing of results, increase accessibility and make voting more convenient for citizens—in some cases, when used over a series of electoral events, possibly even reducing the cost of elections or referendums in the long term. So we propose the “ONLINE VOTING SYSTEM”. Where we have made two different portals one for the admins to start and stop and count the voting of people and the second portal is to register the voters during the initial registration process and also, they can login to cast the votes so that the counting can be done and the casting of the vote can be done from anywhere across the world. As it is completely based on internet they can cast vote from anywhere.

Explanation of the Project:

This code is a Python program that uses the Tkinter library to create a simple online voting system graphical user interface (GUI). The program allows users to vote for one of three candidates and then displays the voting results in a separate window when the "Submit Votes" button is pressed.

Here's a detailed explanation of the code:

Importing the Tkinter Library:

```
*import tkinter as tk*
```

This line imports the Tkinter library, which provides tools for creating graphical user interfaces.

Initializing Candidate Votes:

```
*candidate_votes = {"Candidate 1": 0, "Candidate 2": 0, "Candidate 3": 0}*
```

This dictionary candidate_votes is used to store the number of votes each candidate receives. Initially, all candidates have zero votes.

Defining the vote Function:

```
*def vote(candidate):*  
    *candidate_votes[candidate] += 1*  
    *print(f"You voted for {candidate}")*
```

This function is called when a user clicks on one of the "Vote" buttons. It takes the selected candidate as an argument and increments their vote count in the candidate_votes dictionary. It also prints a message to the console to indicate which candidate the user voted for.

Defining the submit_votes Function:

```
*def submit_votes():*  
    *result_window = tk.Toplevel(root)*  
    *result_window.title("Results")*  
  
    for candidate, votes in *candidate_votes.items():*  
        *label = tk.Label(result_window, text=f"{candidate}: {votes} votes")*  
        *label.pack()*
```

This function is called when the "Submit Votes" button is clicked. It creates a new top-level window (result_window) to display the voting results. It iterates through the candidate_votes dictionary and creates a label for each candidate, displaying their name and the number of votes they received in the result_window.

Creating the Main Window:

```
*root = tk.Tk()*  
*root.title("Online Voting System")*  
*This code initializes the main application window and sets its title to "Online Voting System."*
```

Creating Labels and Buttons for Candidates:

The code creates labels and "Vote" buttons for each candidate using similar patterns. For example:

```
*label1 = tk.Label(root, text="Candidate 1")*  
*label1.pack()*  
*button1 = tk.Button(root, text="Vote", command=lambda: vote("Candidate 1"))*  
*button1.pack()*
```

This creates a label displaying the candidate's name and a button with the text "Vote" that, when clicked, calls the vote function with the corresponding candidate's name.

Creating a "Submit Votes" Button:

```
*submit_button = tk.Button(root, text="Submit Votes", command=submit_votes)*
```

```
*submit_button.pack()*
```

This button triggers the submit_votes function when clicked, which displays the voting results.

Starting the GUI Event Loop:

```
*root.mainloop()*
```

Finally, this line starts the main event loop for the GUI application, allowing user interaction with the created interface.

When you run this code, it will open a window with the names of three candidates and "Vote" buttons next to each. Users can click the "Vote" button for their preferred candidate, and when they click "Submit Votes," a new window will appear displaying the voting results for each candidate.

Conclusion

The ONLINE VOTING SYSTEM shall reduce the time spend making long queues at the polling stations during voting. It shall also enable the voters to vote from any part of the globe as explained since this is an online application available on the internet.

Cases of vote miscounts shall also be solved. Since the voting process shall be open as early as possible, the voters shall have ample time to decide when and whom to vote for.