## ONLINE VOTING SYSTEM USING TCP

A COURSE PROJECT REPORT

By

**MODEM UPENDRA [Reg No: RA2011031010071]**

**VARUN REDDY [Reg No: RA2011031010093]**

**IBRAHIM MUFEEZ [Reg No: RA2011031010100]**

Under the guidance of

**MS.M.SAFA**

*In partial fulfilment for the Course*

Of

18CSC302J - COMPUTER NETWORKS

In

Department of Networking and Communications



#### FACULTY OF ENGINEERING AND TECHNOLOGY

#### SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

**Kattankulathur, Chengalpattu District**

NOVEMBER 2021

## SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

**(Under Section 3 of UGC Act, 1956)**

**BONAFIDE CERTIFICATE**

It is to certify that this project report “**ONLINE VOTING SYSTEM**

**USING TCP”** is the bonafide work of **MODEM UPENDRA [RegNo:RA2011031010071] VARUN REDDY [RegNo:RA2011031010093] IBRAHIM MUFEEZ [Reg No: RA2011031010100]** who carried out the project work under my

supervision.

**SIGNATURE SIGNATURE**

**Dr.M.SAFA DR. ANNAPURANI PANAIYAPPAN .K**

**Assistant Proffessor HEAD OF THE DEPARTMENT Networking and Communications**

Networking and Communications

SRM Institute of Science and Technology

Potheri, SRM Nagar, Kattankulathur, SRM Institute of Science and Technology

Tamil Nadu 603203 Potheri, SRM Nagar, Kattankulathur,

Tamil Nadu 603203

ACKNOWLEDGEMENT

We express our heartfelt thanks to our honorable **Vice Chancellor Dr. C. MUTHAMIZHCHELVAN**, for being the beacon in all our endeavors.

We would like to express my warmth of gratitude to our **Registrar Dr.S. Ponnusamy**, for his encouragement. We express our profound gratitude to our **Dean (College of Engineering and Technology) Dr. T. V.Gopal**, for bringing out novelty in all executions.

We would like to express my heartfelt thanks to the Chairperson, School of Computing **Dr. Revathi Venkataraman**, for imparting confidence to complete my course project

We wish to express my sincere thanks to **Course Audit Professor HEAD OF THE DEPARTMENT**

**Networking and Communications** for their constant encouragement and support.

We are highly thankful to our Course project Internal guide **Dr. P. BALAMURUGAN, Assistant Professor, Department of Networking and Communications,** for hisassistance, timely suggestion and guidance throughout the duration of this course project.

We extend our gratitude to **Dr.Annapurani Panaiyappan.K, Professor and Head of Department of Networking and Communications School of Computing** and my Departmental colleagues for their Support.

Finally, we thank our parents and friends near and dear ones who directly and indirectly contributed to the successful completion of our project. Above all, I thank the almighty for showering his blessings on me to complete my Course project.

**CHAPTER 1**

**ABSTRACT**

In present day world in this pandemic situation, where cases are continuously rising and maintaining social distance is vital, voting is a big concern. In manual voting people need to gather together in large numbers which leads to more social contact and hence increase in the number of cases. A huge amount of money is spent by the election commission to conduct elections. Also transporting EVMs to remote areas is not easy, costly and has security concerns. Manual voting is prone to election manipulation and voter fraud. Handling the database of all the voters is a cumbersome task.

Therefore, we intend to create an online voting system which is cost-effective and secure, saves time in counting votes, is easy to handle, is available at the leisure of voters’ home with decent network connectivity and is completely online and automated and hence is less prone to electoral frauds.

Online voting systems protect the integrity of your vote by preventing voters from being able to vote multiple times. As a digital platform, they eliminate the need to gather in-person, cast votes using paper, or by any other means.

You may hear an online voting system being referred to as an online election system, an online e voting system, or electronic voting. These all make reference to the same thing: a secure voting tool that allows your group to collect input from your group and closely scrutinize the results in real time.

## TABLE OF CONTENTS

|  |  |  |
| --- | --- | --- |
| **CHAPTERS** | **CONTENTS** | **PAGE NO.** |
| **1.** | **ABSTRACT** | 4 |
| **2.** | **INTRODUCTION** | 6 |
| **3.** | **REQUIREMENT ANALYSIS** | 7 |
| **4.** | **ARCHITECTURE & DESIGN** | 8 |
| **5.** | **IMPLEMENTATION** | 13 |
| **6.** | **EXPERIMENT RESULTS & ANALYSIS** | 21 |
| 6.1. RESULTS |  |
| 6.2. RESULT ANALYSIS |  |
| 6.3. CONCLUSION & FUTURE WORK |  |
| **7.** | **REFERENCES** | 27 |

## CHAPTER 2

## INTRODUCTION

TCP or Transmission Control Protocol is a transport layer protocol that facilitates the transmission of packets from source to destination. TCP is [connection-oriented](https://en.wikipedia.org/wiki/Connection-oriented_communication) protocol, and a connection between client and server is established before data can be sent between the computing devices in a network. This protocol is used with an [IP](https://www.javatpoint.com/ip-full-form) protocol, so together, they are referred to as a [TCP/IP](https://www.javatpoint.com/tcp-ip-full-form).

TCP is a reliable protocol as it follows the flow and error control mechanism. It also supports the acknowledgment mechanism, which checks the state and sound arrival of the data. In the acknowledgment mechanism, the receiver sends either positive or negative acknowledgment to the sender so that the sender can get to know whether the data packet has been received or needs to resend.

The Transmission Control Protocol provides a communication service at an intermediate level between an application program and the Internet Protocol. It provides host-to-host connectivity at the [transport layer](https://en.wikipedia.org/wiki/Transport_layer) of the [Internet model](https://en.wikipedia.org/wiki/Internet_model).

Our Online Voting System is cost-effective, secure, time saving, efficient and can handle large databases, and is available to voters online with decent network connectivity and less prone to electoral fraud since each voter has their own unique ID and password.

## CHAPTER 3

## REQUIREMENT ANALYSIS

### Hardware Requirements

Processor 2.4 / 5 GHz Clock Speed

RAM 2 GB

Hard Disk 100 MB (Minimum free space)

### Software Requirements

Operating System Windows 7 and above

Language Python

Local Platform Visual Studio Code

Connection Socket Programming

Protocol TCP

Data Storage CSV file-based Storage

Data Updates & Append Pandas

OS Calls python-subprocess

GUI Module Tkinter

# CHAPTER 4

# ARCHITECTURE AND DESIGN

When the client establishes a connection with the server, it implies that the TCP protocol is being used. The server allocates a new thread for every new incoming client(voter), using concurrent threading.

Concurrent threading means when number of connections are made with the server, that time each thread doesn’t interfere with one another hence, synchronizing the threads. Server checks for authenticity of the client & also checks if client has already voted. It returns a message to the client according to the security check.

Voters register themselves with their unique AADHAR number and the voter list is stored in a csv file. Server can take the client’s name and password and match it with the txt file. If details match, then the voter is redirected to the secured Voting page.

The voters will then cast the vote by mentioning the poll symbol of the candidate from the candidate list provided by the server.

The system (server) can handle multiple clients and creates a new thread for each of them. One client(voter) can cast a vote once and only once.

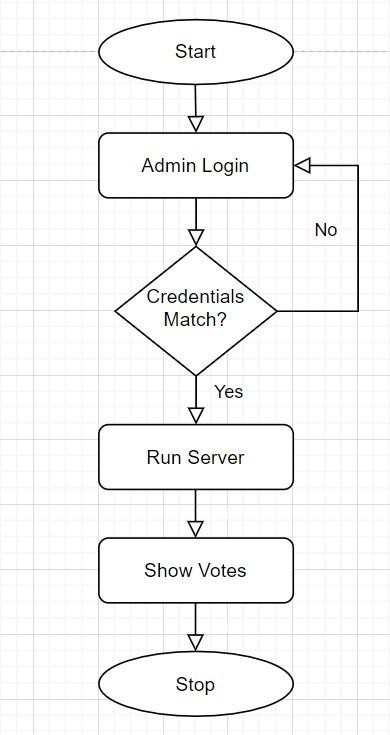
## FUNCTIONAL REQUIREMENTS:

* Registration and verification of users (voters).
* Each voter is given a unique voter-ID and password.
* Each user is able to vote once.
* Database Management.
* Viewability of vote count by admin.
* Error handling.

## NON - FUNCTIONAL REQUIREMENTS:

* All the pages load on time.
* Server upgradation whenever required.
* Security of Voter details.
* Should work perfectly for at least 5 different clients(voters) at the same time.
* Confidentiality of user data.

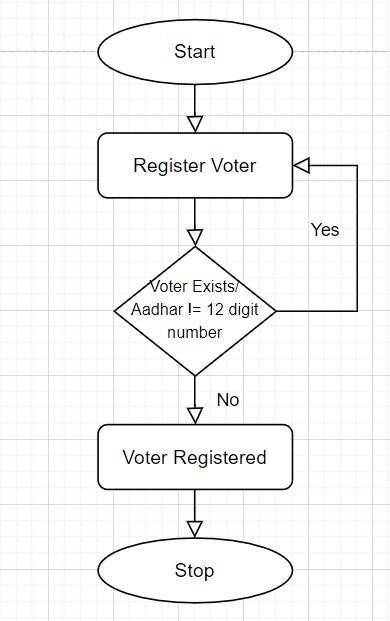
### Architecture Diagram



**Fig. 4.1 (Admin)**

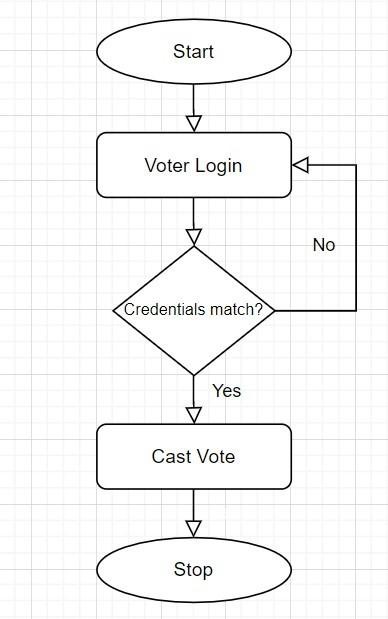
According to the above diagram for Admin login, firstly login screen is displayed. The credentials are asked from the admin. If the credentials do not match, it links to the login page again and if the credentials are matched then it proceeds to the next page which includes “Run Server” and “Show Votes”. The admin has to first run the server

and only then he/she can see votes.



**Fig. 4.2 (Register Voter)**

According to the above diagram for Register Voter, if the voter already exists or the AADHAR card number is not equal to 12 digits then it pops a message and then links back to the login screen. If the conditions are satisfied then the voter gets registered successfully.



**Fig. 4.3 (Voter login and Casting vote)**

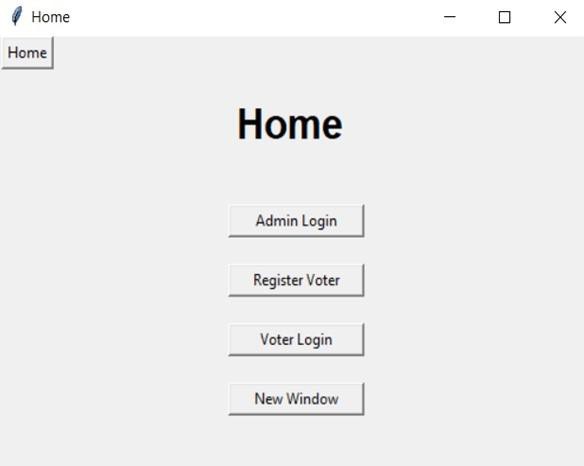
According to the above diagram for Voter login and for Casting vote, firstly voter login screen is displayed. The credentials are asked from the voter. If the credentials do not match, it links to the login page again and if the credentials are matched then it proceeds to the next page which includes “Casting Vote”. The

voter can cast his/her vote and the vote gets stored.

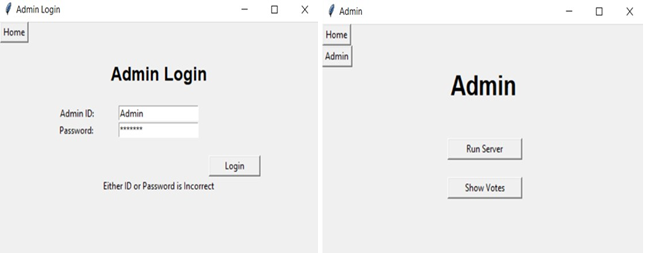
# CHAPTER 5

# IMPLEMENTATION

The program starts displaying the home page with 4 buttons –

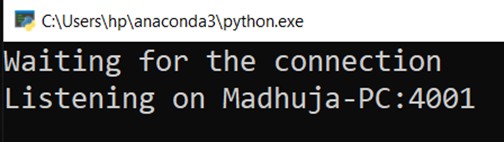
* Admin Login
* Register Voter
* Voter Login
* New Window

The Admin login page inputs Admin’s ID and password and verifies the login credentials.

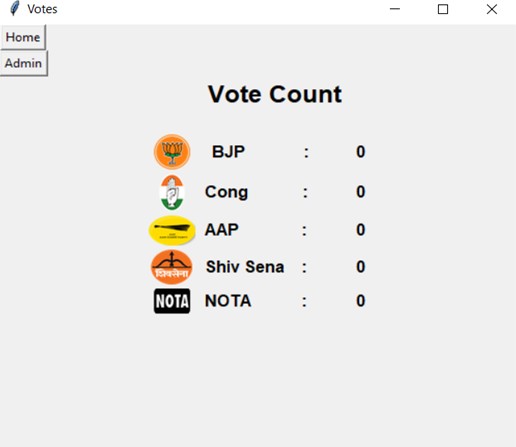


On providing the correct credentials, the program displays Admin page with 2 buttons –

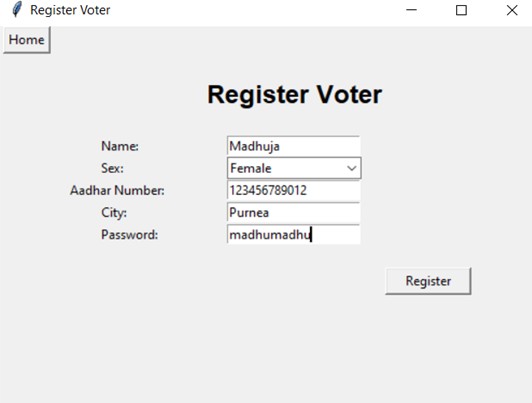
* Run Server
* Show Votes

Clicking Run Server, opens the server in a new window and the client establishes a connection with the server, this implies that the TCP protocol is being used. The server allocates a new thread for every new incoming client(voter), using concurrent threading, that is, when the number of connections are made with the server, that time each thread doesn’t interfere with one another hence, synchronizing the threads.

Clicking Show Votes displays the total vote count of all the parties. Initially, all the vote counts are zero.

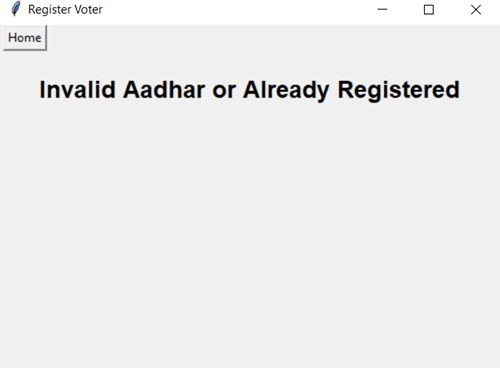


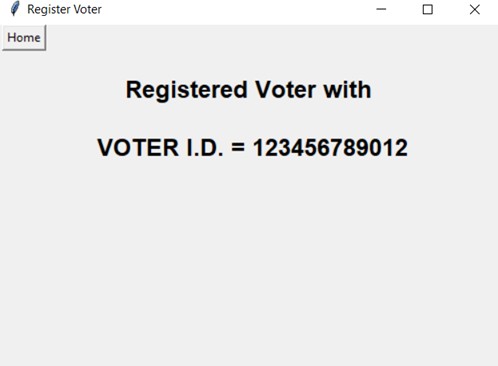
The voters register themselves by entering their details. The voters provide their Aadhar Number and set their own unique password.



If the Aadhar Number entered by the voter is invalid (not a 12-digit number) or already exists, the program gives an error message.

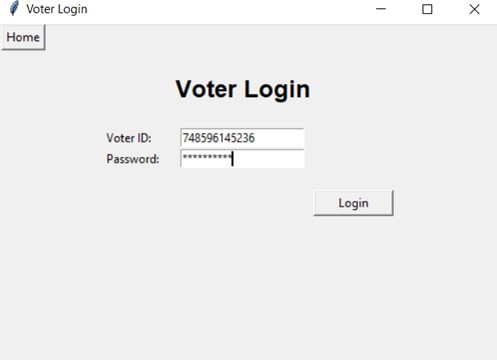
Else, a Voter is successfully registered with Voter ID as their Aadhar Number and voters list (database) is updated.

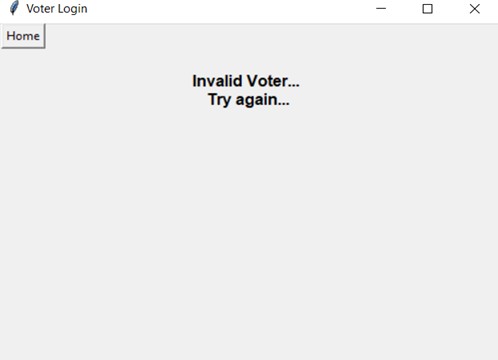




The Voter login page inputs Voter’s ID and password and verifies the login credentials.

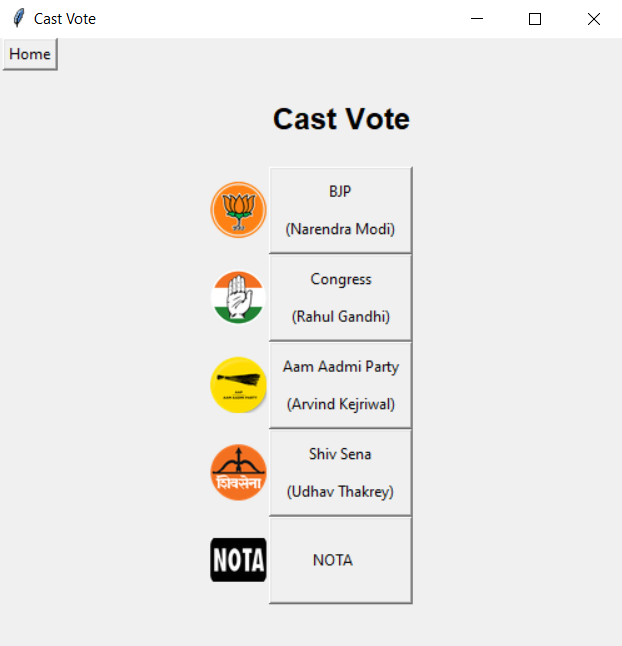
If the credentials don’t match, it gives an error message.

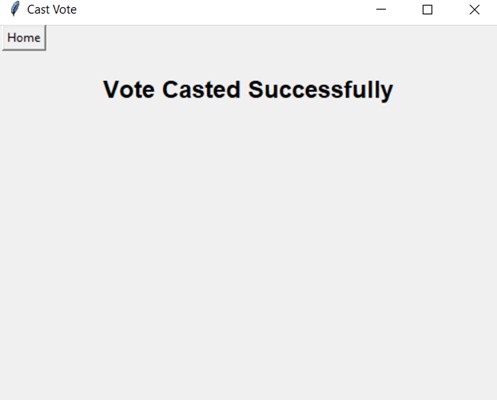




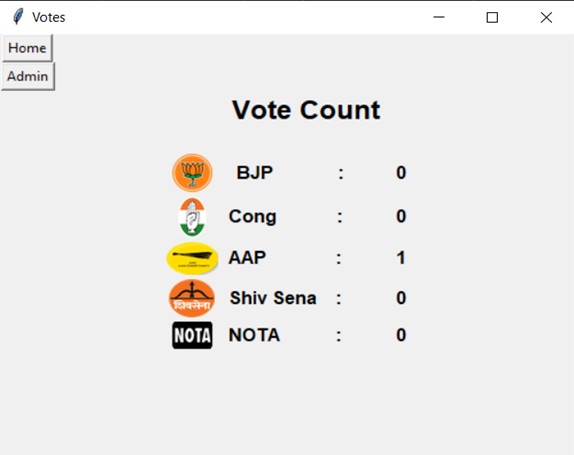
If the credentials match, the cast vote page is displayed and the voter can cast vote of their choice.

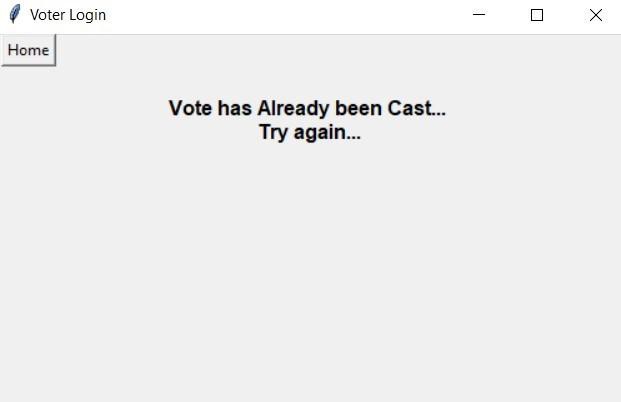
**Cast Vote**

****

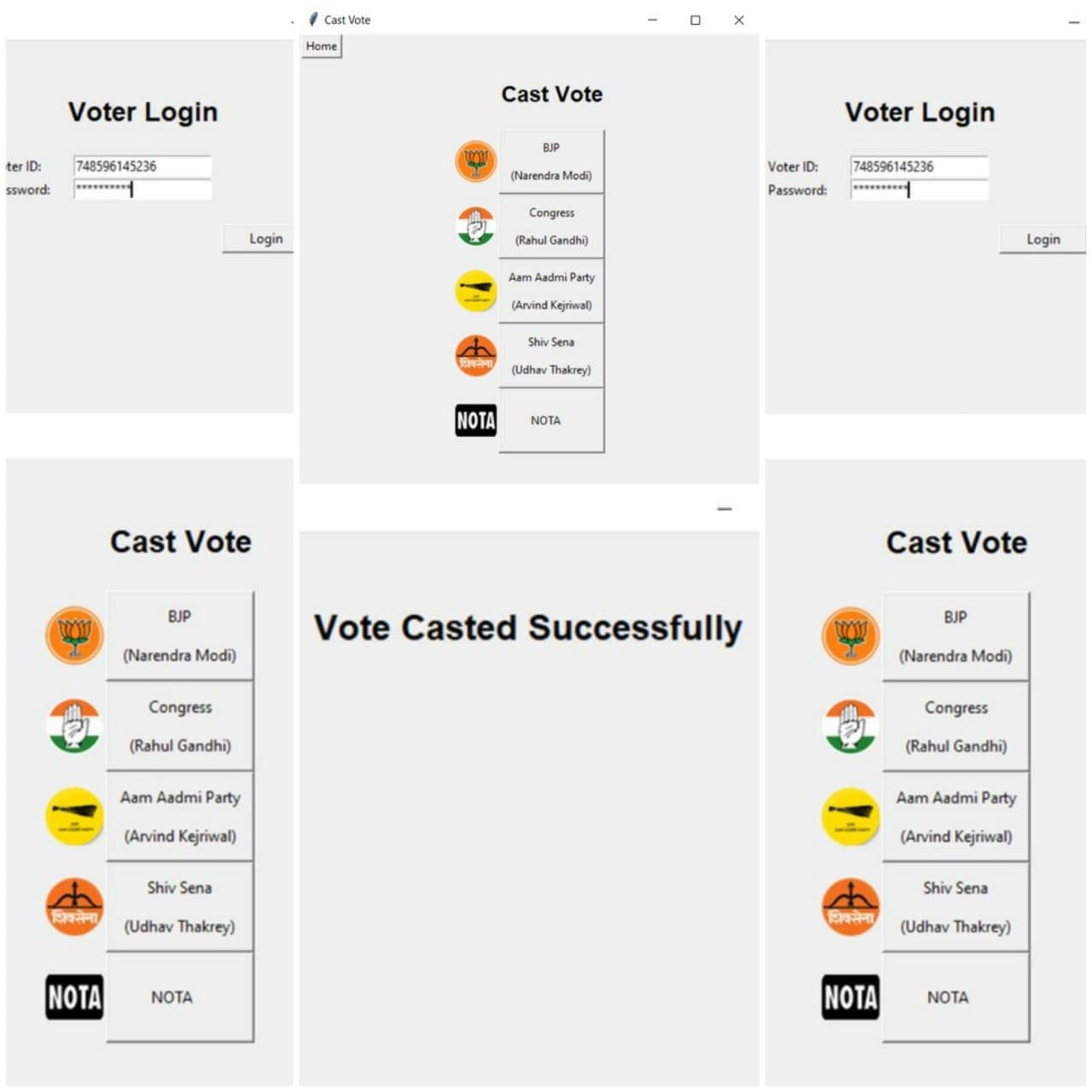


The casted vote is updated in the candidate list (database) and displayed in the show votes page.



One voter can cast a vote only once. If the voter tries to vote again the program displays an error message.

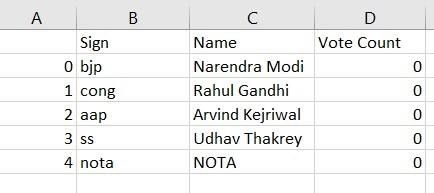
Clicking the new window button opens a new window which allows multiple voters to vote concurrently (should work for 5 different voters at a time).



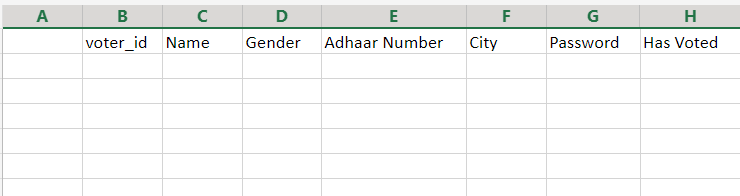
## CHAPTER 6

## EXPERIMENT RESULTS & ANALYSIS

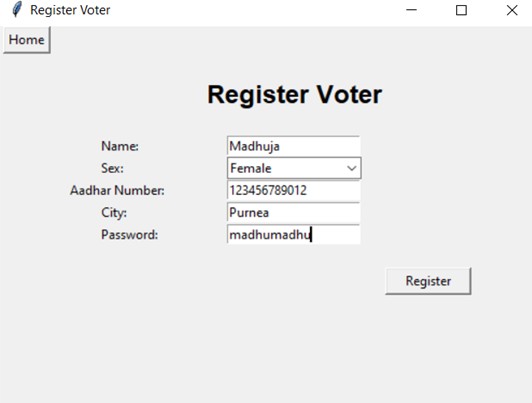
* 1. **RESULTS**

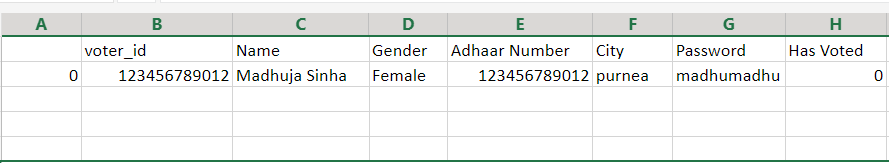
The candidate database is initialized with the entry of candidates participating in elections with their respective vote counts initialized to 0

Initially the voters list database is empty

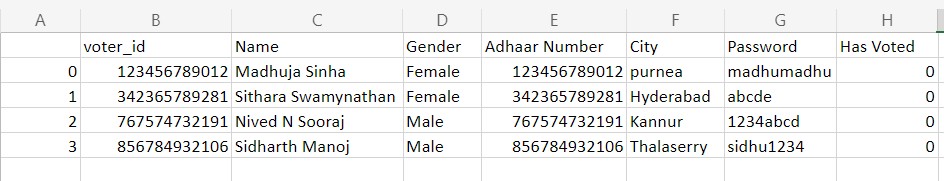


Register voter option is used to add/register an eligible voter to the voter list database

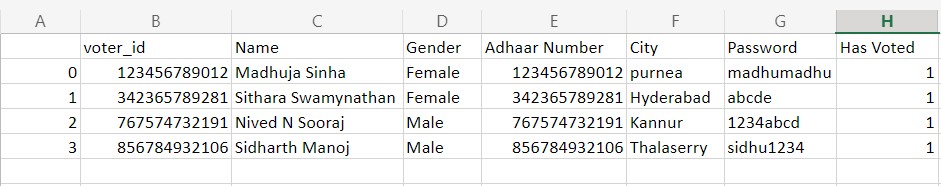


The registered voter is added to the voters list database with their details as well as voter id which is essential during the voting process.

We have added four entries to the voters database using the register voter option.



After each voter casts their vote, the has Voted column will change to 1 to indicate that their vote has been cast.



The vote casted to the respective candidate by each voter is updated live on the candidate database.

## RESULT ANALYSIS

* + 1. The program starts displaying the home page with 4 buttons –
       - Admin Login
       - Register Voter
       - Voter Login
       - New Window
    2. The Admin login page inputs Admin’s ID and password and verifies the login credentials.
    3. On providing the correct credentials, the program displays Admin page with 2 buttons –
       - Run Server
       - Show Votes
    4. Clicking Run Server, opens the server in a new window and the client establishes a connection with the server, this implies that the TCP protocol is being used. The server allocates a new thread for every new incoming client(voter), using concurrent threading, that is, when the number of connections are made with the server, that time each thread doesn’t interfere with one another hence, synchronizing the threads.
    5. Clicking Show Votes displays the total vote count of all the parties. Initially, all the vote counts are zero.
    6. The voters register themselves by entering their details. The voters provide their Aadhar Number and set their own unique password.
    7. If the Aadhar Number entered by the voter is invalid (not a 12-digit number) or already exists, the program gives an error message. Else, a Voter is successfully registered with Voter ID as their Aadhar Number and voters list (database) is updated.
    8. The Voter login page inputs Voter’s ID and password and verifies the login credentials. If the credentials don’t match, it gives an error message.
    9. The casted vote is updated in the candidate list (database) and displayed in the show votes page.
    10. One voter can cast a vote only once. If the voter tries to vote again the program displays an error message.
    11. Clicking the new window button opens a new window which allows multiple voters to vote concurrently (should work for 5 different voters at a time).

After analyzing the working and result produced by our online voting system, we have come to the conclusion that an online voting system has the following depicted advantages over a traditional voting system

|  |  |
| --- | --- |
| **MANUAL VOTING SYSTEM** | **ONLINE VOTING SYSTEM** |
| Requires high maintenance | Requires low maintenance |
| Expensive | Cost-Effective |
| Highly prone to electoral frauds | Less prone to electoral frauds |
| Difficult to manage Database | Easy to manage Database |
| Takes lot of time to count votes | Saves time in counting votes |
| Too much paperwork and errors during data entry | No paperwork required and is completely automated |

## 6.3. CONCLUSION & FUTURE WORK

In this project, we have implemented an online voting system using TCP protocol as well as multithreading. Hence, from this project we learned how to implement TCP socket programming using Python. We also learned how to connect multiple clients with one server. As per the design of the project, we allocated a new thread by server for every new incoming voter, thus, we learned how to implement synchronized multithreading in python and implemented it in the code of socket programming.

Such an online voting system if implemented on a large scale with a more capable server and more security features could potentially change the face of elections. Online voting systems are cost-effective and secure, saves time in counting votes, is easy to handle, is available at the leisure of voters’ home with decent network connectivity and is completely online and automated and hence is less prone to electoral frauds. Hence it is much better than the current traditional voting systems by a large number of factors.

**CHAPTER 7**

**REFERENCES**

1. https://[www.javatpoint.com/tcp](http://www.javatpoint.com/tcp)
2. <https://totalview.io/blog/multithreading-multithreaded-applications>
3. https://[www.fortinet.com/resources/cyberglossary/tcp-ip](http://www.fortinet.com/resources/cyberglossary/tcp-ip)
4. https://[www.tutorialspoint.com/python/python\_multithreading.htm](http://www.tutorialspoint.com/python/python_multithreading.htm)
5. https://[www.geeksforgeeks.org/socket-programming-cc/](http://www.geeksforgeeks.org/socket-programming-cc/)
6. https://pythongeeks.org/subprocess-in-python/
7. https://pandas.pydata.org/