Online Voting System Using TCP In Python

A COURSE PROJECT REPORT

By

MANIVEER REDDY(RA2011031010127)
ADITYA BALAJI YALAVARTHY(RA2011031010120)
SHRIDHAR SURADA(RA2011031010132)
HADI (RA2011031010116)

Under the guidance of

Dr. THANGAREVATHI

In partial fulfilment for the Course

of

18CSC302J - COMPUTER NETWORKS

in Networking and Communication

FACULTY OF ENGINEERING AND TECHNOLOGY

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

Kattankulathur, Chengalpattu District

NOVEMBER 2022

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

(Under Section 3 of UGC Act, 1956)

BONAFIDE CERTIFICATE

Certified that this mini project report "**Online Voting System Using TCP In Python**" is the bonafide work of **Aditya Balaji Yalavarthy, Maniveer Reddy, Shridhar Surada, Hadi** who carried out the project work under my supervision.

SIGNATURE

THANGAREVATHI
Assistant Professor
NWC
SRM Institute of Science and Technology

Table of Contents

Project Analysis

The topic allocated to us for the socket programming project is "e-Voting System". Here, the client establishes a connection with the server, this implies that the TCP protocol is being used. The Server should allocate a new thread for every new incoming Client, to accomplish this feature we took care of concipient thread, that is , when the number of connections are made with the server, that time each thread doesn't interfere with one another. Therefore, we synchronized the threads.

❖ **Design and Implementation:**

1. A secure server that only allows clients with authentic names and passwords bcast votes.
2. Server checks for authenticity of the client & also checks if client has already voted. It retains a message to the client according to the security check.
3. Votes are registered by admin and the votes list is stored in a csv file.
4. Server can take the client name and password and match it with the txt file.
5. If details match, then the votes is redirected to the secured Voting page.
6. The votes will then cast the vote by mentioning the poll symbol of the candidate from the candidate list provided by the server.
7. The system (server) can handle multiple clients and creates a new thread for each of them.
8. One client can cast a vote once and only once.

Requirements

Python Libraires Required :

- ➔ Pandas
- ➔ Tkinter
- ➔ Socket
- ➔ Subprocess

Tools Used

- Programming: Python
- Connection: Socket Programming
- Protocols: TCP
- User Interface: python-Tkinter
- Data Storage: Using CSV files
- Data Updates: python-pandas
- OS Calls: python-subprocess

How to Run

1. Open terminal/command prompt on your PC.
2. Navigate to 'Voting' folder
3. Run command:

   python homePage.py
4. A new home page window should open. If this doesn't happen, check your installations.
5. Login into Admin using given details in 'How to Login'.
6. Click on the 'Run Server' Button.
7. Use the best of the Buttons as you need.

How to Login

- **Admin Login:**
  - Admin ID: Admin
  - Passwords: admin

- **Voter Login:**

  - Server should be running for votes to be able to login.

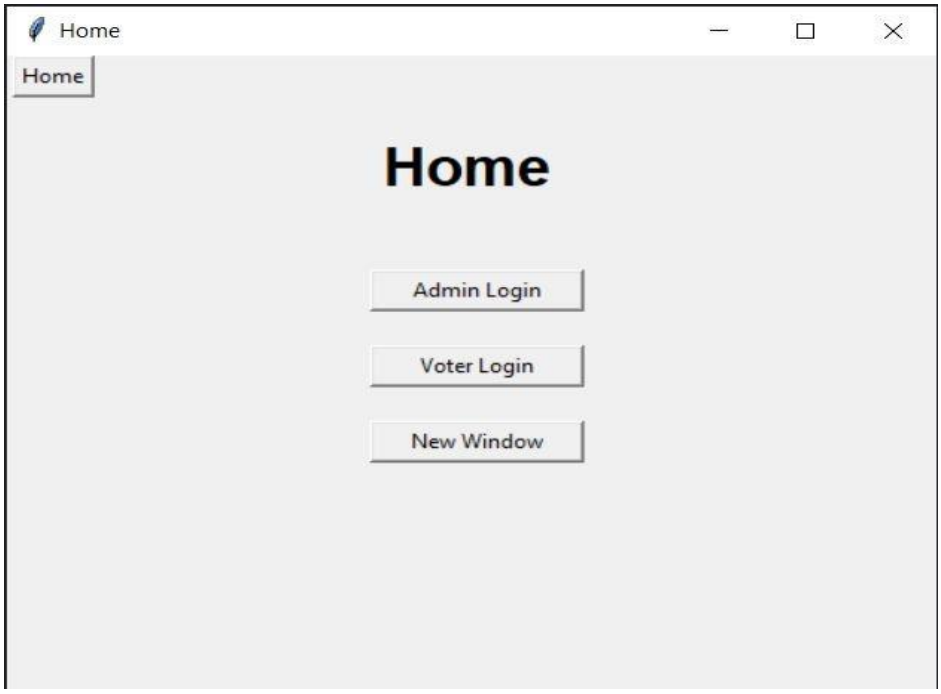
  - ➔ Already registered votes I.Ds : 10001 to 10005
  - ➔ Passwords (for already registered votes) : Abcd

Workflow Description

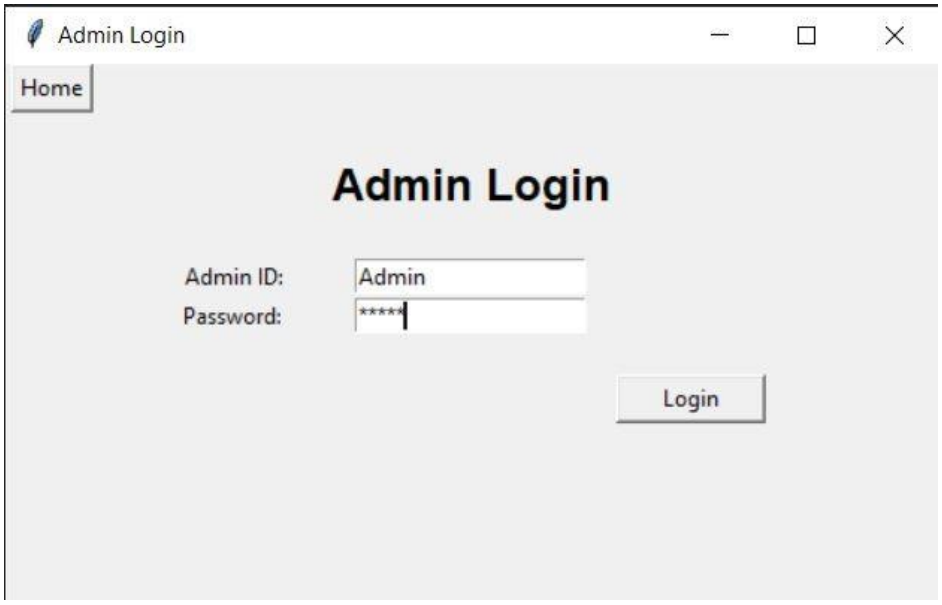
- Inside Description to run & test this project :

  1. Open terminal & run python homePage.py to open Home Page Window.
  2. Log into Admin and pies 'Run Server'. This will run the Server in a new console window.
  3. Now that the Server is running, fetuin to the admin home page window.
  4. Pies 'Register Vote and enter details to register a new votes. Remember oi note down the 'Vote ID' that you will receive on successful registration.
  5. Pies 'Home' to retain to the Home. Now, pies 'Voter Login' to open the voter login page.
  6. Enter the login details and you are redirected to the Voting Page. You will receive an message if the voter is invalid oi has already cast a vote.
  7. Cast a Vote. Now on receiving a success message, press home to return to home.
  8. Login into Admin again. Press 'Show Votes' to check the votes that all parties have received so far.
  9. Return to Home. You can press 'New Window' to open multiple pages and cast a vote coincidently from multiple votes.

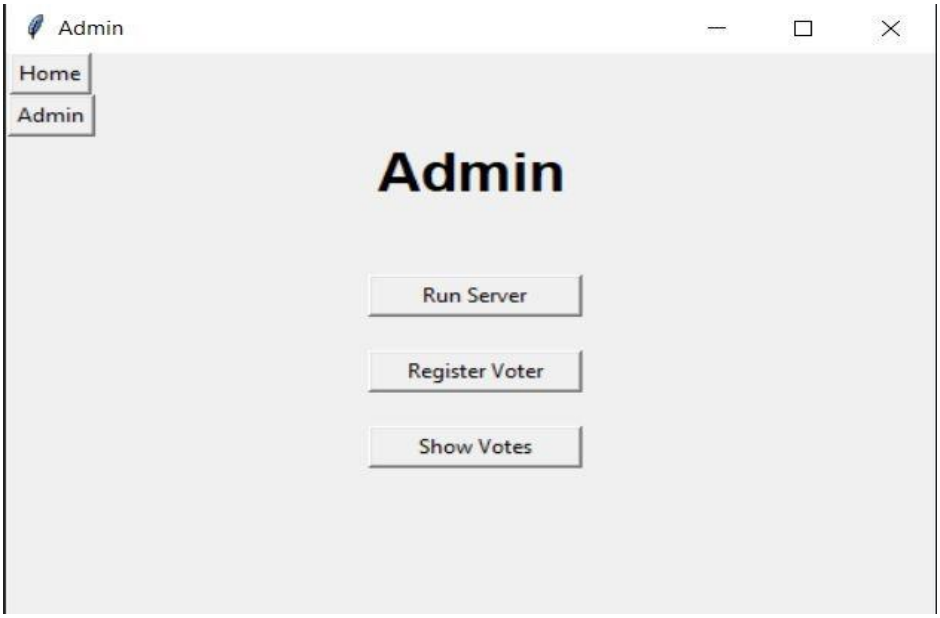
Stepwise Output / Test Cases



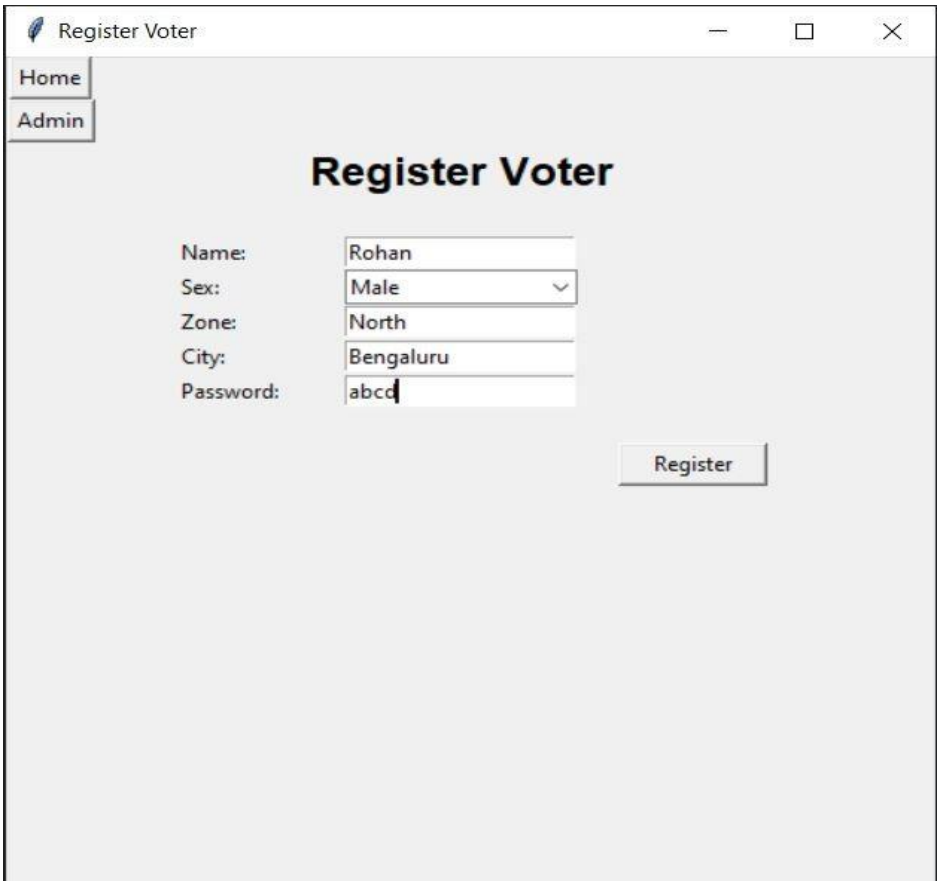
Home Page



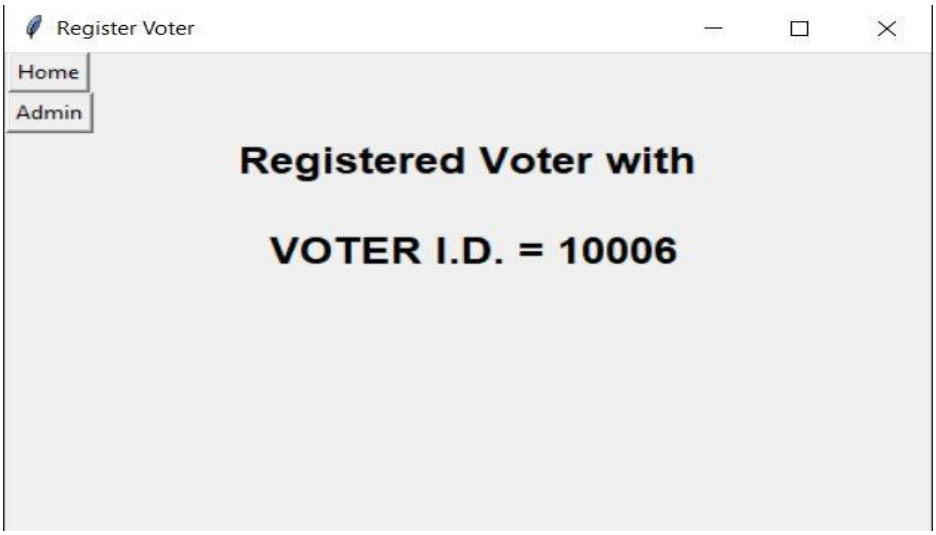
Admin Login



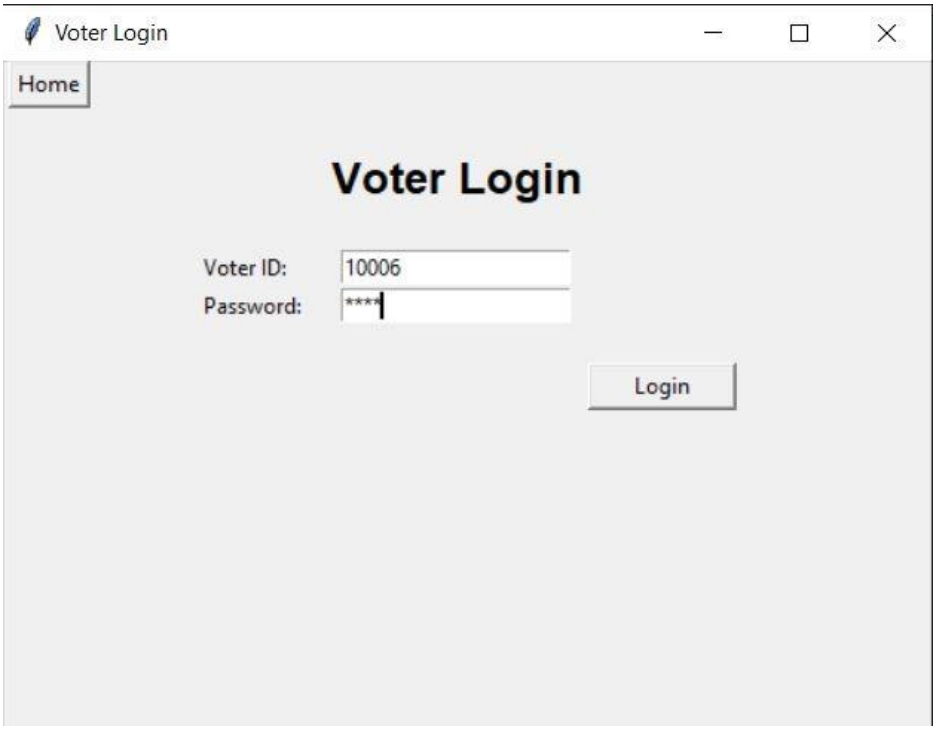
Admin Home



Registei Votei

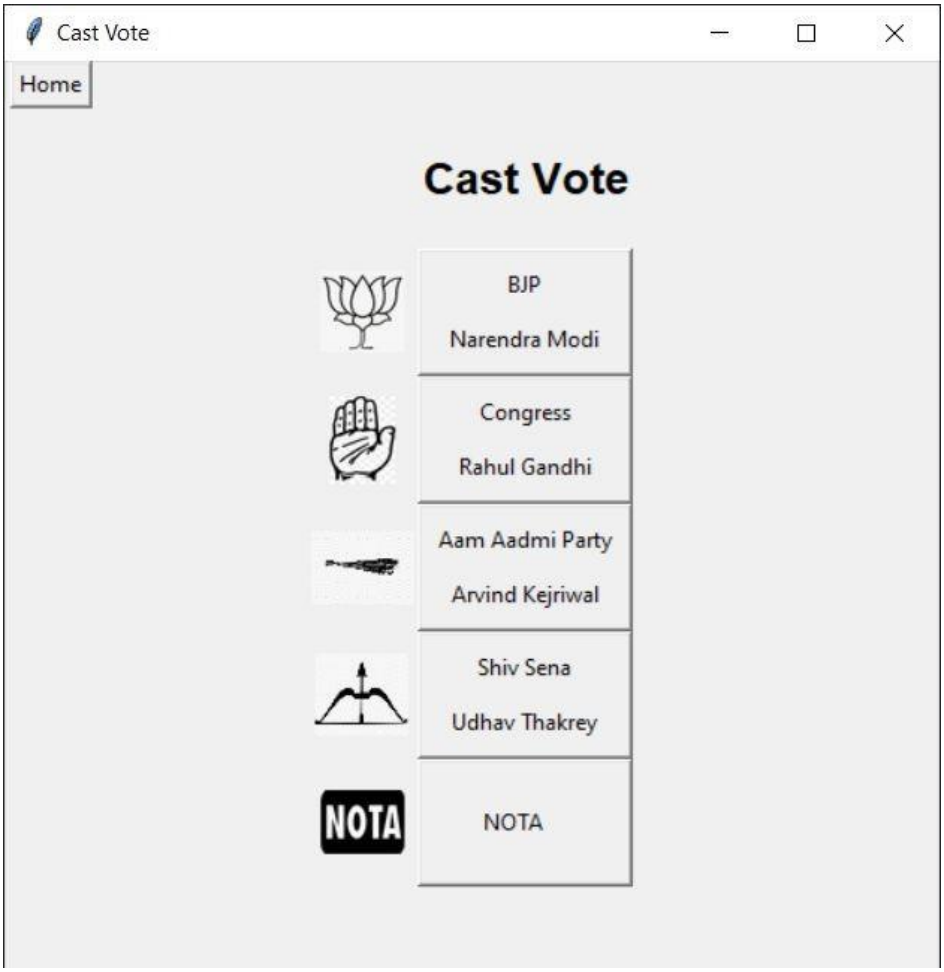


Register Success Message

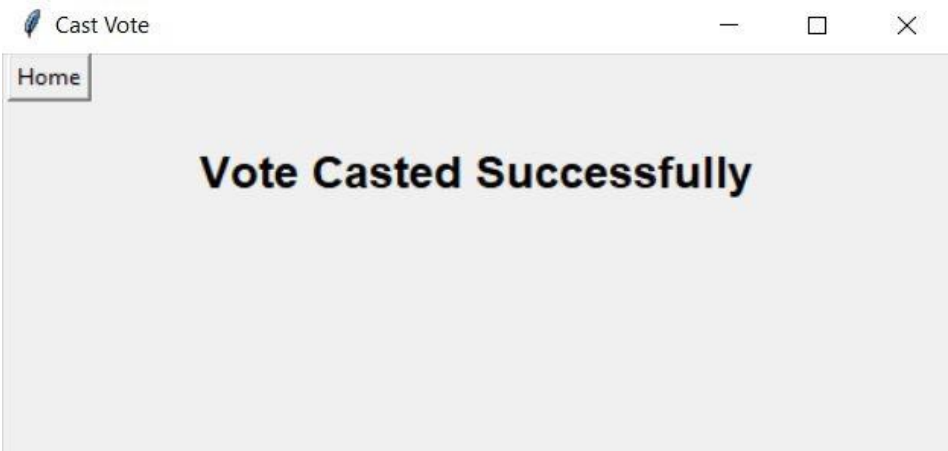


voter Login

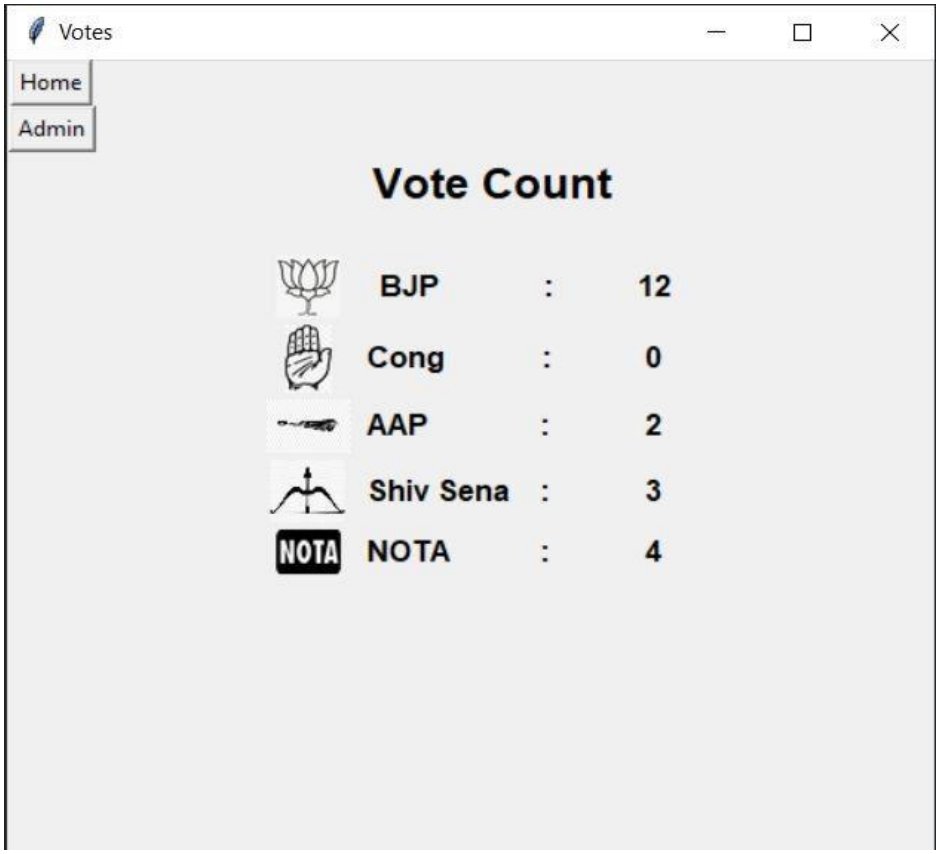
Test Case 1 : If detail matches, then it welcomes the voter and displays the name and poll symbol of the candidates.



Voting Page



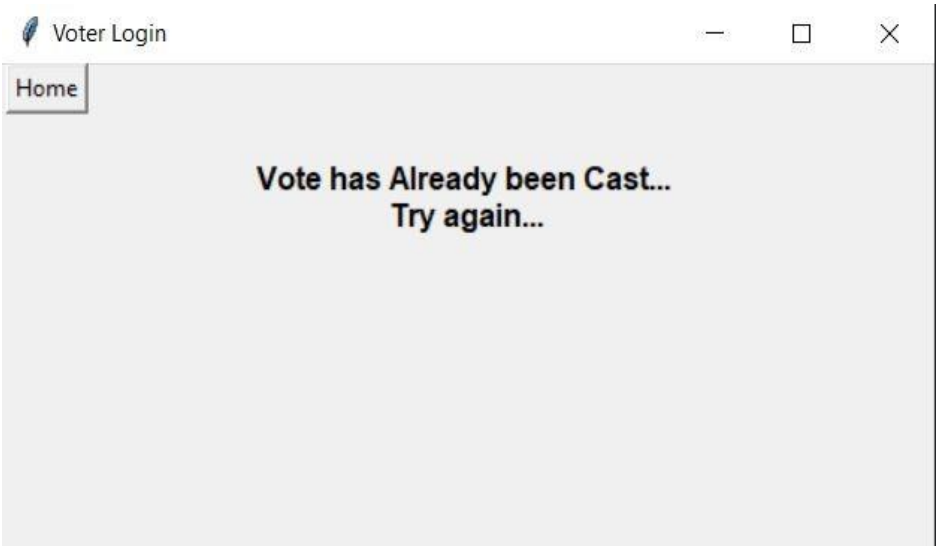
Vote Casted Successfully Message



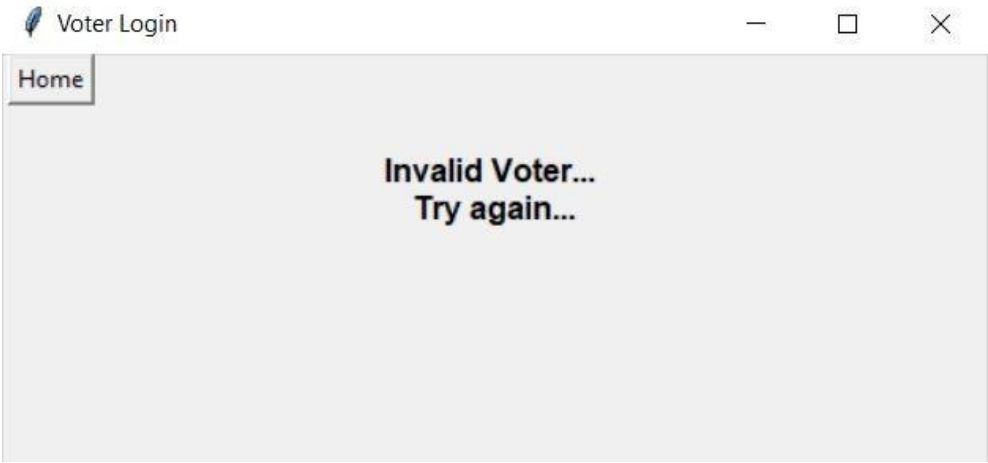
Show Votes

❖ Error Handling :

Test Case 2 : One client can cast a vote ONCE AND ONLY ONCE.



If the vote already been casted

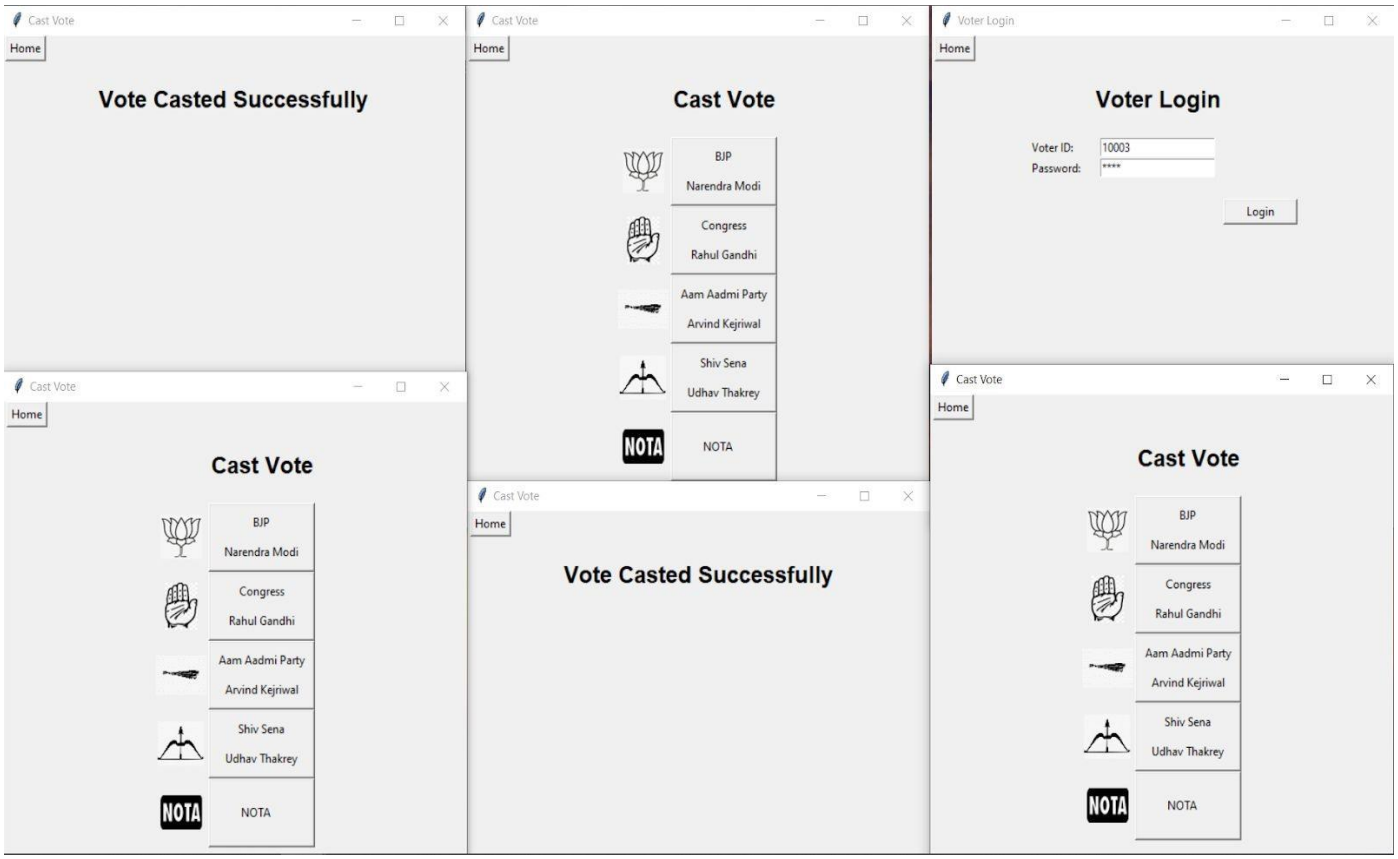


If a votei is not iegisteied/invalid votei

Error while casting vote

❖ Vote casting vote coincidently:

<u>Test Case 3</u>: This system should work perfectly for at least 5 different clients at the same time.



<u>6 voters</u>

Server Output

Select C:\ProgramData\Anaconda3\python.exe

```
Waiting for the connection
Listening on □ □□:4001
Connected to : ('192.168.0.113', 56631)
Voter Logged in...
ID:10006
Vote Received from ID: 10006  Processing...
Vote Casted Sucessfully by voter ID = 10006
Connected to : ('192.168.0.113', 56635)
Vote Already Cast by ID:10006
Vote Received from ID: 10006  Processing...
Vote Update Failed by voter ID = 10006
Connected to : ('192.168.0.113', 56636)
Invalid Voter
Vote Received from ID: 10006  Processing...
Vote Update Failed by voter ID = 10006
Connected to : ('192.168.0.113', 56663)
Connected to : ('192.168.0.113', 56664)
Connected to : ('192.168.0.113', 56665)
Connected to : ('192.168.0.113', 56666)
Connected to : ('192.168.0.113', 56667)
Voter Logged in...
ID:10001
Voter Logged in...
ID:10002
Connected to : ('192.168.0.113', 56668)
Voter Logged in...
ID:10006
Vote Received from ID: 10001  Processing...
Vote Casted Sucessfully by voter ID = 10001
Voter Logged in...
ID:10005
Vote Received from ID: 10005  Processing...
Vote Casted Sucessfully by voter ID = 10005
Voter Logged in...
ID:10004
Vote Received from ID: 10004  Processing...
Vote Casted Sucessfully by voter ID = 10004
Connected to : ('192.168.0.113', 56686)
Vote Already Cast by ID:10004
Vote Received from ID: 10004  Processing...
Vote Update Failed by voter ID = 10004
Connected to : ('192.168.0.113', 56687)
Voter Logged in...
ID:10005
Vote Received from ID: 10005  Processing...
Vote Casted Sucessfully by voter ID = 10005
Voter Logged in...
ID:10002
Vote Received from ID: 10002  Processing...
Vote Casted Sucessfully by voter ID = 10002
```

Database

	voter_id	Name	Gender	Zone	City	Passw	hasVoted
0	10001	Deep	Male	West	Gandhinag	abcd	0
1	10002	Prachi	Female	South	Surat	abcd	0
2	10003	Het	Male	East	Surat	abcd	0
3	10004	Shivanshi	Female	East	Gandhinag	abcd	0
4	10005	Rohan	Male	North	Bengaluru	abcd	0

Votei Info Database

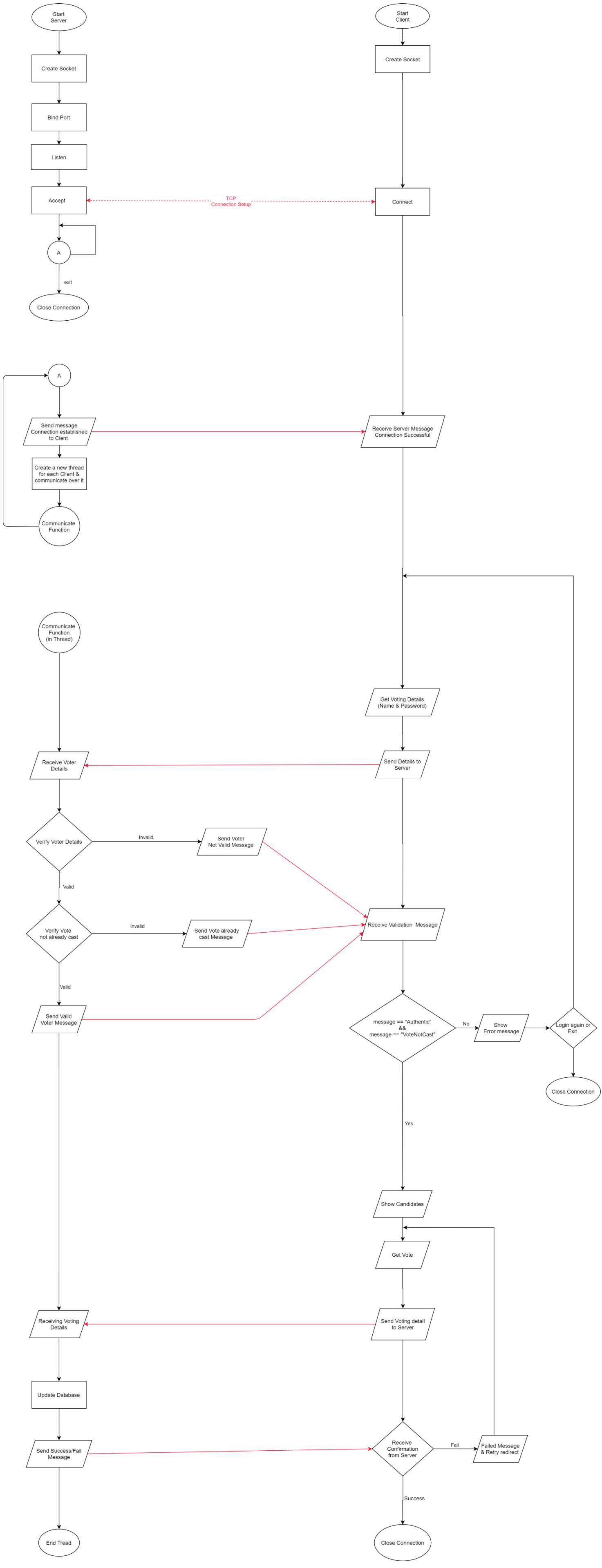
	Sign	Name	Vote Count
0	bjp	Narendra Modi	15
1	cong	Rahul Gandhi	0
2	aap	Arvind Kejriwal	3
3	ss	Udhav Thakrey	4
4	nota	NOTA	5

Candidate Info Database

Conclusion

For the 'E-voting system' project we learned how to implement TCP socket programming using Python. We also learned how to connect multiple clients with one server . As the requirement of the project was to allocate a new thread by server for new incoming Client, thus to accomplish this equipment we learned how to implement synchronized multithreading in python and implemented it in the code of socket programming.

Flow Chart


Start Server
Create Socket
Bind Port
Listen
Accept
A
exit
Close Connection
Start Client
Create Socket
Connect
TCP Connection Setup
A
Send message Connection established to Client
Create a new thread for each Client & communicate over it
Communicate Function
Receive Server Message Connection Successful
Communicate Function (in Thread)
Get Voting Details (Name & Password)
Receive Voter Details
Send Details to Server
Verify Voter Details
Invalid
Send Voter Not Valid Message
Valid
Verify Vote not already cast
Invalid
Send Vote already cast Message
Receive Validation Message
Valid
Send Valid Voter Message
message == "Authentic" && message == "VoteNotCast"
No
Show Error message
Login again or Exit
Close Connection
Yes
Show Candidates
Get Vote
Receiving Voting Details
Send Voting detail to Server
Update Database
Send Success/Fail Message
Receive Confirmation from Server
Fail
Failed Message & Retry redirect
Success
End Tread
Close Connection