Chapter 1

INTRODUCTION

1.1 Introduction about online voting

The introduction of an online voting system aims to provide a more convenient and efficient way for citizens to participate in elections. With paper-based voting systems, it can be difficult to locate specific candidates and ensure voter eligibility. It also made hectic and rush for voters to visit the Centre and vote the candidate. An online voting system addresses these issues by providing secure authentication and verification mechanisms, making the voting process more automated and streamlined. It made easy for authorized person to login in from its own device and vote. Furthermore, online voting systems can also increase transparency and provide faster results. While there are concerns regarding security and privacy, the benefits of an online voting system cannot be denied. In this context, the purpose and scope of the system are to ensure that every citizen can participate in the democratic process in a secure and hassle-free manner

The purpose of the online voting system is to provide a convenient platform for voters to exercise their democratic right without hassles. The system seeks to eliminate the need for standing in queues and using paper ballots, EVM machines that may be challenging to locate a specific candidate. The scope of the system is vast, as it can be used for various elections, ranging from local/state government to national assembly polls. Additionally, the benefits of the system include an increase in voter turnout and enhanced accessibility for all. The features of the system include a secure authentication and verification process using username and PIN However, the potential challenges and concerns with the system include the possibility of hacking or tampering with the votes, which can lead to false results. Implementing the system would require collaboration between various government agencies, and there is a need to create awareness among voters about the system's benefits. Overall, the purpose and scope of the online voting system can revolutionize the way elections are conducted in India, and it is essential to work towards its implementation in a secure and transparent manner.

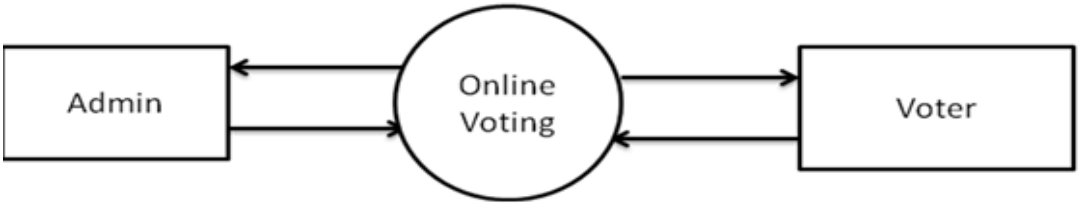

Fig 3.1:Level-0-DFD

Level-1

FIG:1.1- DATA FLOW DIAGRAM

1.2 Overview

A software package entitled "E-vote" will be the ultimate product of the development team. This software will allow organizations to design and run elections securely and privately on a server connected to a distributed network. It will be possible to customize the ballots for each election as well as to set certain parameters for each election, such as whether or not voters enhances online presence through SEO techniques and mobile-friendly design.

may change their votes once they have been submitted. The E-vote package will also include election as well as to set certain parameters for each election, such as whether or not voters may change their votes once they have been submitted. The E-vote package will also include the software needed by voters to register their votes using computers connected to a distributed network. The server will then track votes and calculate statistics on the results of the election. The longterm goal of this project is to greatly reduce the cost and complexity of running elections by removing the direct involvement of humans in the mechanics of gathering and counting votes.All aspects of the software will use a graphical user interface. The system will be furnished with a full on-line help system, as well as installation software.

1.3 Scope of the E-vote Project

This E-vote software should only be regarded as a pilot project, meant to examine the feasibility of voting technology and to explore its potential. It is not intended for elections on a national scale, at least at this stage. However, the software will be designed to be scalable to full-scale elections given greater time, manpower and testing resources.

Through the utilization of state-of-the-art encryption techniques, multi-factor authentication protocols, and stringent security measures, the proposed OVS aims to safeguard the integrity

and confidentiality of the voting process. Furthermore, by incorporating features such as user-friendly interfaces, multi-language support, and compatibility with assistive technologies, the system strives to ensure the accessibility and inclusivity of the electoral process for all citizens. In this introduction, the key components and features of the proposed Online Voting System, as well as discuss the significance and potential impact of implementing such a system in modern democracies. By exploring the challenges and opportunities associated with online voting, we aim to underscore the importance of advancing technological solutions that can enhance the democratic experience while upholding the fundamental principles of transparency, security, and fairness in elections.

This project seeks to address these challenges by proposing the design and implementation of a robust and secure Online Voting System (OVS). The primary objective of this system is to provide a reliable platform for conducting elections in a manner that is both technologically advanced and democratically sound. Through the utilization of state-of-the-art encryption techniques, multi-factor authentication protocols, and stringent security measures, the proposed OVS aims to safeguard the integrity and confidentiality of the voting process. Furthermore, by incorporating features such as user-friendly interfaces, multi-language support, and compatibility with assistive technologies, the system strives to ensure the accessibility and inclusivity of the electoral process for all citizens.

In this introduction, the key components and features of the proposed Online Voting System, as well as discuss the significance and potential impact of implementing such a system in modern democracies. By exploring the challenges and opportunities associated with online voting, we aim to underscore the importance of advancing technological solutions that can enhance the democratic experience while upholding the fundamental principles of transparency, security, and fairness in elections.

Chapter 2

GENERAL DESCRIPTION

2.1 Product Functions

The E-vote software package will be made up of four basic components:

2.1.1 E-vote Elections Server

The Server will be responsible for storing the settings of each election, generating passwords for authorized Voters, receiving and authenticating votes, storing votes on the E-vote Voting Database, generating statistics at the end of each election and maintaining and verifying security and voter privacy. The Server will also potentially contact all authorized voters by e-mail to give them their username information , passwords, Server address, election code, instructions to obtain the Voting Client and contact information of the Elections Officer.

2.1.2 E-vote Election Editor

The Election Editor will be a piece of software that enables Elections Officers to design custom ballots and define configuration settings for each election. It will also allow the Elections Officer to suspend an election. The Elections Officer will also use the Election editor to enter the enumeration list. The enumeration will be a list of all voters that are authorized to vote in an election. This list could include the usernames and e-mail addresses of each authorized voter. It will be possible to configure elections so as not to require voter e-mail addresses, if this is desired.

2.1.3 E-vote Voting Database

The Voting Database will hold the enumeration list for each election as well as all vote registered for each election. It will be encrypted and will not be directly accessible by anybody. The votes stored from each election will be deleted from the database at a preset time after the termination of each election.

2.1.4 E-vote Voting Client

The E-vote Voting Client will be a simple piece of software that voters with only a minimal background in computers will be able to install on their own computers .They will

use it to vote by establishing a network connection to the Server and sending their encrypted votes. They will enter the Server address and their username , password and election code before establishing the connection. Once they have connected, the Server will send them the ballot, which they will then fill out using the Client and send back to the Server. Voters will also use the Client to change their votes if this option has been enabled in a particular election.

2.2 **User Characteristics**

2.2.1 Customers

The customers are the people or organizations who purchase the E-vote Elections Software. They will be authorized to host elections on the Server. The Voting Client will be available free of charge, and any purchasers of the server software will be authorized to distribute it to their voters. Each customer will be responsible for providing a System Administrator to overlook the installation and operation of the Server. The customers will also be responsible for providing a host for the Server.

2.2.2 System Administrator

A System Administrator will be required to oversee the installation and operation of each E-vote Elections Server. The System Administrator will not have control over or access to any particular elections once they are activated on the server, but he or she will be the only one able to authorize Elections Officers to start new elections. The System Administrator is assumed to have at least a college level background in computers and networking. This person is necessary to ensure that the overall system is working and that its security integrity is not breached.

2.2.3 Elections Officer

The Elections Officer is an impartial individual who is given responsibility for overseeing individual elections by the organization holding each election. Each election housed on a Server may have one or more Elections Officers and it is possible for an Elections Officer to be responsible for more than one election. This person is needed to monitor each election in a way that is independent of the general overall maintenance provided by the Systems Administrator.

Once an Elections Officer is authorized to set up an election by the System Administrator, he or she is the only one with the ability to design ballots, configure election options, enter an enumeration list and suspend the election. He or she is also responsible for answering questions over e-mail that voters or others may have during

the election.

The Elections Officer is expected to be well versed in the elections protocol of the organization whose election he or she is supervising and is expected to be comfortable using GUI-based computer applications. It is assumed that the Election Officer speaks and writes English, since the implementation team will not have the time or the knowledge to write multiple versions of the Elections Editor software to accommodate other languages. This could be done in a future release of the software.

2.2.4 Voters

Voters are those people who are authorized by the Elections Officer to vote in each election using the E-vote Voting Client. They are expected to have access to a fully networked computer and to be comfortable using GUI-based computer applications. They must also have a secure access to a private e-mail address.

2.3 Operating Environment

2.3.1 **E-vote Elections Server**

The Elections Server will be written in Java, so the computer hosting it must be capable of running Java bytecode. The Server will run under Linux. The computer hosting the server must be accessible by other machines on a network. Testing will be done using the computers in Reynolds 008 as a basis for the minimum hardware requirements to run the E-vote Election Server. Professor Stacey stated in class that she did not expect the software would run the US election, but that it would be reasonable to assume that the class could go down to Reynolds 008 to vote. From this statement we infer that the E-vote Election Server must be able to run on one of the computers in the lab. It must be able to host an election where Voters can use the Voter Client software on all of the other computers in Reynolds 008 to participate in an election. There are approximately forty machines in the Reynolds lab. Thus, we will insure that at least forty voters can simultaneously vote on any one election in the Reynolds lab.

2.3.2 E-vote Election Editor

The Election editor will be written in Java, so the computer hosting it must be capable of running Java bytecode. Since we are using the lab in Reynolds as a basis for the minimum requirement to host the E-vote Election Server, the lab computers will also serve as the basic hardware requirement for the E-vote Election editor.

2.3.3 E-vote Voting Database

The Voting Database will be required to hold voter identities, as stated in the project handout. The computer hosting the database must have either Postgres or MySQL installed.

2.3.4 E-vote Voting Client

The Voting Client will be written in Java, so the computers that it is installed on must be capable of running Java bytecode. It will run under Linux or Windows 95 and above. It is expected that all computers it will be installed on will have high speed Internet access

such as cable modem or DSL, as was stated in class. The computers that will run the client should be able to run GUI-based Java computer programs.

2.4 User Problem Statement

As it stands, almost no elections are held electronically. This means that there is a great cost associated with collecting and counting votes, since many people must be hired to perform and check these tasks. Manual elections take a long time to set up, and occurrences such as recounts can greatly delay the reporting of results. All of these time delays can be at least partially eliminated by having computers run elections .There are also many problems relating to the accuracy of manual elections. The intentional inaccuracies introduced by the corruption of election officials can be eliminated by having the election handled by an entirely impartial computer.

The unintentional inaccuracies of manual elections, such as improperly printed or filled out ballots, can also be eliminated by electronic elections which use clear and consistent interfaces.

It should be noted that although electronic elections have a great number of advantages, they are largely untested and thus even the best systems may be prone to problems, at least initially. Although they can certainly be designed with full security and privacy features,

corruption of the results can be very difficult to detect if someone does manage to break through the security of the software. The privacy feature of the software means that what the software is actually doing during an election cannot be transparent to election officials. One must also be very careful that security holes are not built into the elections software by the developers and that the

officials who run and maintain the elections do not have the power to corrupt results.

2.5 User Objectives

Any organization running an electronic election will want software that is easy to install and run. Ballots must be easy to design and they must be flexible as to the number and types of questions. The Election Server must run efficiently and securely. It must be impossible for

anyone to break into the system and corrupt the results, prematurely know the results of the election, vote when they are not authorized to do so or vote more than once. The results of the election must be clearly presented upon the completion of the election.

Voters need the ballots to be clear and easy to fill out and answer. It is important that it be impossible for anyone to associate a voter's name with his or her vote.

2.6 General Constraints

The development team must design, develop and test this software within the space of three months. They also have important limitations placed on their time due to many other projects that they must work on. They also suffer from severe lack of funding. Due to these constraints, as well as the limited number of people working on the project, it may be necessary to prioritize certain aspects of the project over others. Functionality and security will be the first priorities.

The developing and testing environment is limited to the University of Guelph computer labs. This means that the developers do not have access to the full commercial system that is necessary to fully test this system under realistic working conditions.

Chapter 3

LITERATURE SURVEY

[1] Bhushan M. Pawar et.al The aim of this project is to make the democratic process simple for the students at the college level. Presently in our college, vote casting is performed by utilizing paper and counting is done manually so it expends students as well as educators valuable time, also there can be a possibility of error while tallying the cast votes. All this makes the vote casting process very dreary so in our project, the vote capturing and tallying is done on the web. It saves processing time, avoids human errors and there won't be any invalid votes It has a basic user interface of application which attracts users. As this application is planned for students, so verification happens on the basis of unique ID code which is students registered ID, to cast their votes remotely from any place. This is a combo box application so it additionally comprises university question papers, syllabus, and college fundamental data or different activities of the college.

[2] Sridharan Srivatsan et.al This paper aims at creation of a voting system by providing a cost-effective solution to the government along with ensuring non-traceability and integrity of the votes cast while providing great convenience to voters. This system is developed robustly to ensure that all eligible voters having a Universal Identification Number of their country (For Example the Smart Card in USA) is allowed to cast their respective vote. The voters, who cast multiple votes during the process of voting is ensured to be prevented. Also, to ensure the maintenance of authenticity, any biometric identification of the voters could be used for accessing the terminal to cast their vote and restricting them to cast again. The process of online voting could be deployed with three phases - the voter registration online vote capturing and the instant online counting and result declaration. A Secret Voting Password provided to voter during registration acts as an authentication mechanism which enables the voters to securely cast their vote along with their captured biometric identification. A Simulation result of implementation of the same is described in this paper by describing the robustness of this system.

Chapter 4

SYSTEM ANALYSIS

4.1 proposed System:

A proposed system of online voting could incorporate the following features and technologies to ensure a secure, accessible, and transparent process:

1. Blockchain-based architecture: Utilize blockchain technology to ensure the integrity and transparency of the voting process.

2. End-to-end verifiability: Allow voters to verify their votes were recorded and counted correctly.

3. Two-factor authentication: Require voters to provide additional verification beyond just a password.

4. Open-source code: Make the voting software's source code publicly available for scrutiny and review.

5. Independent audits: Conduct regular security audits and penetration testing to identify vulnerabilities.

6. Accessible voting interface: Ensure the voting platform is accessible for voters with disabilities.

7. Multi-factor voter verification: Implement robust voter identification and verification processes.

8. Real-time results reporting: Provide live updates on vote counts and election results.

9. Paper ballot backups: Offer a paper ballot option for voters who prefer it or in case of a system failure.

10. Voter education and support: Provide clear instructions, tutorials, and support resources for voters.

11. Secure voter registration: Implement a secure and automated voter registration process.

12. Post-election audits and analysis: Conduct thorough audits and analysis after the election to ensure accuracy and identify areas for improvement.

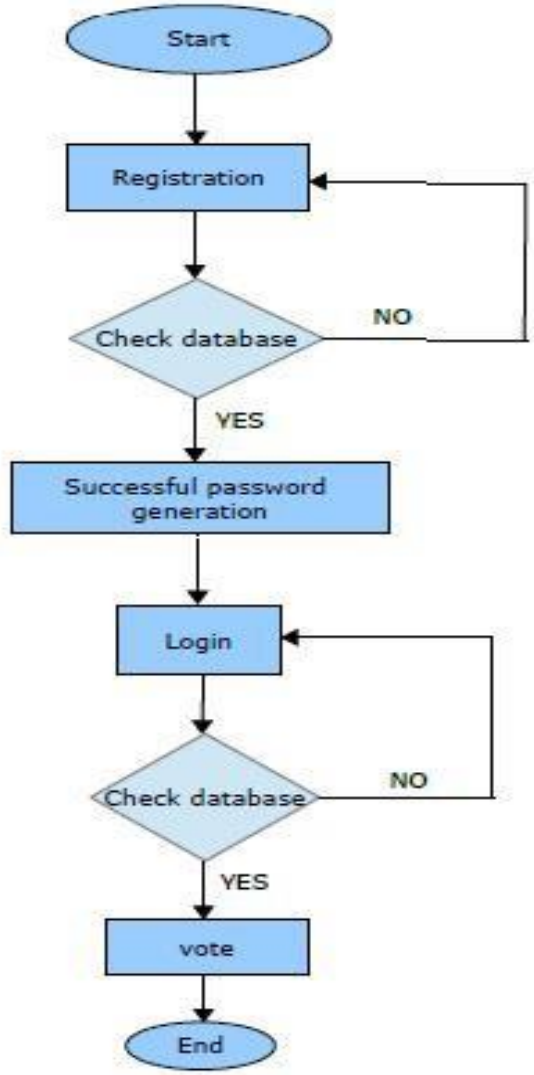

FIG:4.1:FLOW CHART

4.1.1 Disadventages of Proposed System:

1. Security risks: Online voting systems are vulnerable to cyber attacks, hacking, and data breaches, which could compromise the integrity of the election.

2. Voter authentication: Ensuring the authenticity of voters and preventing impersonation can be challenging in online voting systems.

3. Privacy concerns: Online voting systems may not fully protect voter privacy, potentially allowing others to see how individuals voted.

4. Technical issues: Technical problems like server crashes, connectivity issues, or software glitches could disrupt the voting process.

5. Voter verification: Ensuring that only eligible voters can cast ballots can be difficult in online voting systems.

6. Lack of transparency: Online voting systems may make it difficult to verify the accuracy of vote counts and ensure transparency in the electoral process.

7. Dependence on technology: Online voting systems require reliable internet access and devices, which could exclude some voters.

8. Vote manipulation: Online voting systems may be vulnerable to manipulation by malicious actors.

9. Auditability: Online voting systems can make it challenging to conduct accurate audits and recounts.

10. Public trust: Some voters may be hesitant to trust online voting systems, potentially leading to decreased voter turnout.

4.2 Existing System:

here existing systems of online voting vary depending on the country, state, or organization implementing it. Some common features and technologies used in online voting systems include:

1. Electronic Ballot Marking: Digital ballots that can be marked electronically.

2. Online Polling Places: Websites or portals where voters can cast their ballots.

3. Electronic Voting Machines: Specialized devices used in polling stations.

4. Mobile Voting Apps: Applications allowing voters to cast ballots via smartphones.

5. Email or Fax Voting: Systems where ballots are sent and returned via email or fax.

6. Blockchain-based Systems: Utilizing blockchain technology for secure and transparent voting.

7. End-to-End Verifiability: Systems that enable voters to verify their votes were recorded and counted correctly.

8. Two-Factor Authentication: Security measures requiring voters to provide additional verification beyond just a password.

9. Audit Trails: Records of all system activities to ensure transparency and detect potential issues.

10. Accessibility Features: Options for voters with disabilities to ensure equal access.

A Use Case Diagram is a vital tool in system design, it provides a visual representation of how users interact with a system. It serves as a blueprint for understanding the functional requirements of a system from a user's perspective, aiding in the communication between stakeholders and guiding the development process.

4.3 FEATURES

In the online voting system, features play a significant role in ensuring a smooth and secure voting process. The system should offer candidate registration and document verification, with an auto-generated user ID for each voter.

It should be easy to understand and use, with an online interface that makes it accessible to all eligible voters. Automation of the voting process is a crucial feature that streamlines the entire process and saves time. The system should be able to display results in real-time, ensuring transparency in the voting process. To ensure secure authentication and verification of voters, the system should use AADHAAR ID-based online election. While there may be concerns and challenges, such as the need to prevent hacking and ensure data security, implementing an online voting system has numerous benefits and is the way forward in modern democracy. for voters, who no longer have to travel to polling booths or wait in long queues to cast their vote. leadership votes and student government elections. These benefits contribute to a more democratic process and promote greater participation among citizens.

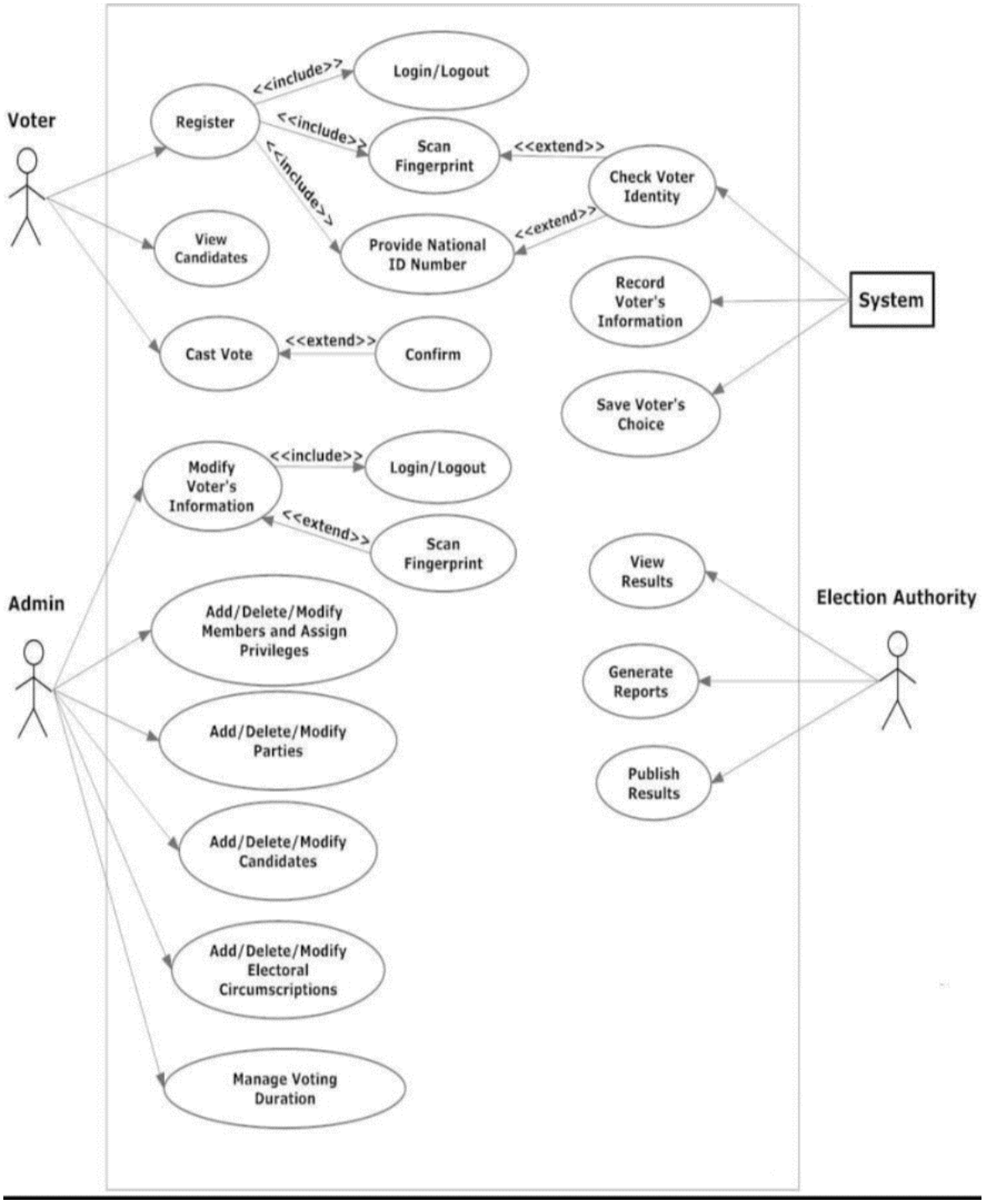

FIG:4.2 -USE CASE DIAGRAM

4.4 SECURE AUTHENTICATION AND VERIFICATION

Online voting systems rely on secure authentication and verification techniques to ensure only eligible voters can cast their vote. In India, voters can use their voter identification number to cast their vote through the online module of the project.

The online voting app created by the Election Commission Authority only allows verified voters to vote. The Estonian I-voting system provides an even more secure authentication method by using the PIN of the national ID card for authentication. Any electronic voting system must ensure that a voter's identity is securely verified before allowing them to cast their ballot. This helps to prevent voter fraud and ensures that the election results are fair and accurate.

4.5 BENEFITS

Online voting offers numerous benefits over traditional voting methods. Firstly, it provides accessibility and ease use of online voting system. It eliminates the need for manual counting of votes, thereby reducing the chances of human error and ensuring a smooth and efficient election process. With the online voting system, voters can cast their vote conveniently from their mobile devices or computers, making it easily accessible to a wider audience. Additionally, the system can eliminate fake ballots, rigging of votes, and other forms of electoral fraud. The automation of the voting process provides a credible and transparent voting process that gives voters the confidence that their vote will be counted fairly. By adopting an online voting system, election authorities can streamline and modernize their processes while improving the accuracy and reliability of their voting system.

4.6 RESULT AND TRANSPERANCY

The online voting system offers a high level of transparency and efficiency in the electoral process. It automates the voting process, reduces the chances of errors or manipulation, and ensures fair and accurate results. The system is designed with secure authentication and verification features that help to eliminate fraudulent activities, such as vote selling or intimidation. The real-time results make it easier for voters to follow the progress of the election and have trust in the outcome. However, there are still concerns

related to security and privacy that must be addressed by implementing appropriate effective online voting system could greatly benefit the democratic process.

4.7 AUTOMATION OF THE VOTING PROCESS

Automation of the voting process is one of the key advantages of having a measure, such as the use of VVAT (Voter-Verifiable Audit Trail) that can help to enhance transparency and accountability. Overall, the online voting system has the potential revolutionize the electoral process, and with careful implementation and monitoring, it can ensure greater transparency and trust in democratic elections.

4.8 CHALLENGES AND CONCERNS

While online voting has the potential to increase voter turnout and reduce costs, there are several challenges and concerns that must be addressed. One of the biggest concerns is the security and privacy of the voting process. It is crucial that the authentication and verification methods are secure and so that no outside interference can corrupt the results. Additionally, there is a concern that online voting could disenfranchise certain groups who may not have easy access to the internet or who may not be tech-savvy. Another challenge is the potential for technical difficulties or glitches during the voting process. It is important that the system is reliable and can handle a high volume of users without crashing Finally, there is a concern that online voting could lead to a lack of transparency in the election process.

Chapter 5

METHODOLOGY

This software is being developed for convenient use of voters. it can be used for various elections, ranging from local/state government to national assembly polls User just have to register, login and vote his/her favorable candidate.

Online voting system contains:

a) Voter's information in database.

b) Voter's Names with ID and password.

c) Voter's vote in a database.

d) Calculation of total number of votes

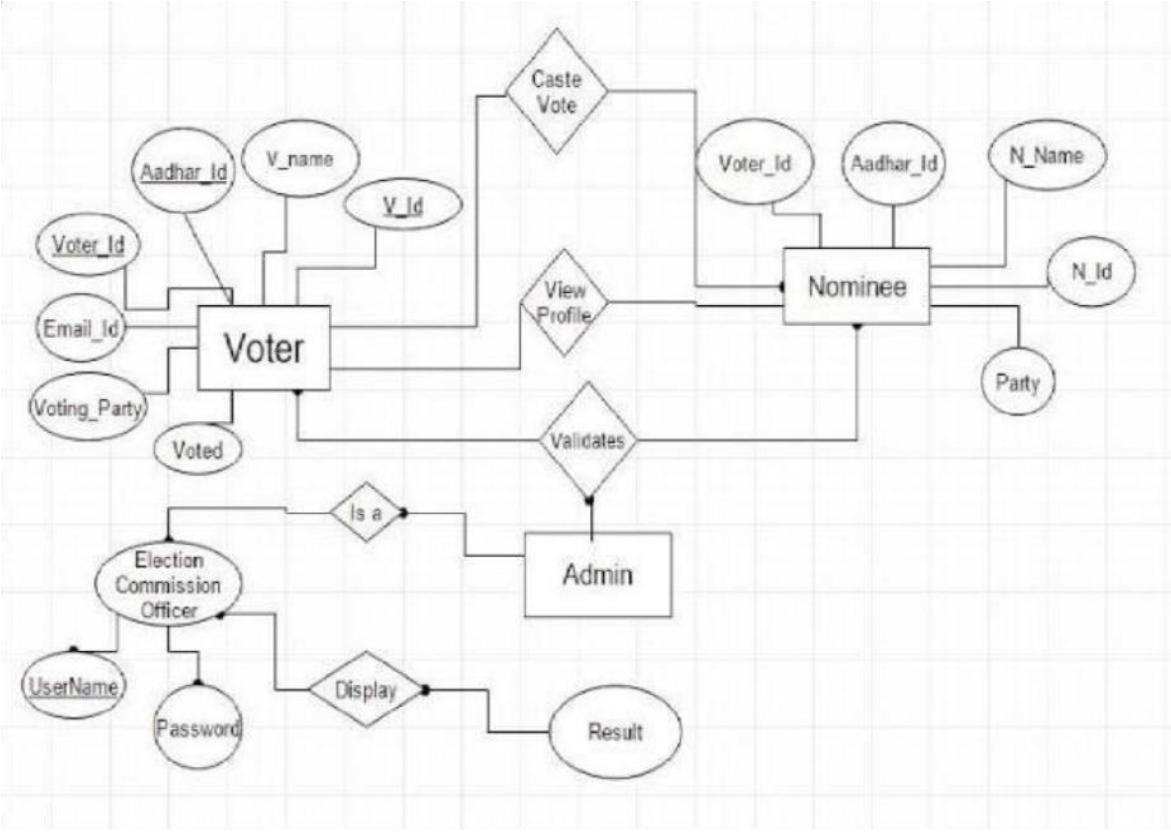

FIG:5-E.R DIAGRAM OF ONLINE VOTING SYSTEM

This online voting system can be explained using following phases:

1) Registration phase.
2) Authentication phase.
3) Voting phase.
4) Counting phase.
5) Result

1) Registration phase:

The registration phase in an online voting system is a critical step that ensures the security and integrity of the voting process

2) Authentication phase:

The authentication phase in an online voting system is a critical step that ensures the security and integrity of the voting process

3) Voting phase:

The voting phase in an online voting system is the critical component where voters cast their ballots.

4) Counting phase:

Election Counting is the most critical part of the ENCORE application.

ENCORE Application is an integrated system that includes all the major activities related to the conduct of the election

5) Result: In this phase the result will be generated

Our system has a server back-end which takes care of all the authentication and the admin panel will take care of each and every function going within such as, user registration, candidate registration, eligibility, casting vote by user, result, etc.

Chapter 6.

SYSTEM REQUIREMENTS

The primary priorities of this design are, in order of importance:

1) Functionality

2) Reliability

3) Maintainability

4) Security and Privacy

5) Scalability

6) Interfaces

6.1 Functional Requirements

The requirements that are essential are marked with a dash, while the ones which are less essential are marked with a bullet.

6.1.1 Election editor

 There will be the option of providing additional information about election issues on the ballot.

 The ballot will have an arbitrary number of questions.

 The text of each question must be specified.

 All questions will be multiple choice. The number of choices as well as the texts of the choices may be customized.

 The start date and end date of the voting must be specified.

 Allows the Elections Officer the option of letting voters change their votes before the election ends.

 Lets it be specified how long the information stored on the database will persist after the election terminates.

 Lets it be specified from a limited list of options what statistics are wanted when an election ends. This list will include the results for each option of each question (total number of votes and percentages) as well as voter patterns such as the time distribution of votes.

 An enumeration list must be entered which could include the usernames, passwords and e-mail addresses of each authorized voter.

 Will allow the Elections Officer to change the election settings, provided that the election has not started yet.

 Provides Elections Officer with an election code to uniquely identify each new election that is created.

 The look and feel of the ballot will be customizable.

 Allows the Elections Officer to specify a time window to accommodate differences in time settings on individual computers.

 Allows the Elections Officer to decide what criteria will be used by voters to identify themselves from a limited numbers of options.

 What qualifies as a win must be customized from a limited number of options (i.e. is there a quorum for the election and/or each question, is there a minimum percentage of votes in favor needed for a choice in each question to win and do the results of certain questions depend on the results of other questions).

 There will be the option of having weighted votes to accommodate situations such as shareholders' elections.

 Specifies whether voters can refuse or spoil individual questions or their entire ballots.

 Must only allow creations of elections when it is authorized by the System Administrator.

 Will indicate to the Elections officer if there are not enough systems resources to accommodate a new election.

6.1.2 Election Server

 Stores the settings for each election after they are generated by the Election editor.

 Stores and generates passwords for all authorized voters on enumeration lists.

 Checks all Voting Clients when they attempt to log on to see if they are authentic (valid election code and a username and password that correspond to an authorized voter in that election that has not voted yet).

 Sends a copy of the appropriate ballots to Voting Clients.

 Sends each voter who has successfully voted a verification number to prove that they have voted.

 Encrypts all communication with the Voting Client.

 Stores all votes on the Election Database.

 Generates voting statistics at the end of each election. Statistics must only be available after the election has ended, not while it is still being carried out.

 Generates a complete list of statistics and a partial list of statistics.

 E-mails these statistics to the Elections Officers and another source.

 Must be able to accommodate multiple elections simultaneously.

 Contacts all authorized voters on enumeration lists by e-mail to give them their username information, passwords, Server address, election code, instructions to obtain the Voting Client and contact information of the Elections Officer

 Reports any attempted breaches of security to the Elections Officer.

 Allows the Elections Officer to suspend the election.

6.1.3 Election Database

Stores the enumeration list for each election.

 Stores votes as they are authenticated by the Server.

 Encrypts all stored data.

 Only allows the Server to read stored data.

 Deletes the data stored from each election at a preset time after the completion of each election.

6.1.4 Voting Client

 Can access an Election Server using a distributed network.

 Allows voters to log on to a Server by entering its address, their username, password and election code.

 Receives ballots from the Server after successfully logging on.

 Allows voters to enter their votes and send them to the Server.

 Encrypts all communications with the Server.

 Allows voters to change their votes after having submitted them if this is allowed in a particular election.

6.2 System Attributes

6.2.1 Communications Security

All communications between the Voting Client and the Elections Server must be encrypted to ensure the privacy of votes and voter information. Encryption of communications will also ensure that anyone packet sniffing over the network will be unable to extract any usable information from the data that is sent between the Client and Server. The server will time out after three minutes if no message is received from the Client.

All data sent to the server will conform to a pre-defined format to enable the software to detect any tampering of data in transit. If detected, the data will be discarded and the voter prompted to resubmit his or her vote.

Voters must be sure that nobody else has access to their e-mail addresses, as anyone reading their e-mail would have access to their voter identification information. The Elections Server

will limit the number of login attempts to prevent automated attacks to gain or prevent access, however the voters must ensure that their password is secure as the system would

not be able to detect misrepresentation of the voter. No encryption is foolproof, although many are highly reliable. It must be understood that developments of new algorithms could potentially break any encryption, as could sustained efforts with high-powered computers. Elections should thus be run over limited periods of time, such as a single day, in order to minimize the chances of

security breaches.

6.2.2 Storage Security

All voters must be able to record their votes anonymously without anybody being able to determine how they voted or change their votes. If voters are going to be able to change their own votes, the system must store their identification information along with their votes. This means that the file storing their votes must be encrypted so that nobody can read it directly at any stage and should be deleted soon after the election ends, but not so soon as to make a recount impossible. It must therefore be as close to impossible as possible for anyone to break into the database for at least the length of the election plus the amount of time the election data is stored afterwards.

6.2.2 Maintainability

The software will be well documented and it will be designed to be modular. The use of object oriented programming will also help to increase maintainability. This will make it easier for future developers to make changes and updates to the software with a minimal amount of effort.

6.2.3 Scalability

Both the E-vote software and this document are meant to be easily scalable to increase the scope and size of elections. All efforts will thus be made to use a software design that does not have built in size limitations.

6.2.4 Reliability

All efforts will be made to write software that is entirely reliable. However, the viability of electronic voting rests, in part, on the ability of systems administrators and elections officials to incorporate redundancy into any deployed voting system and to develop contingency plans for possible failures.

6.2.5 Interface

All aspects of the E-vote system will have a simple point and click interface using menus, text fields, buttons and all of the other components of systems with graphical user interfaces. This interface will be designed to be consistent. The interface will be designed to help accommodate people with disabilities such as colour blindness. The system will also be have a full on-line help system. Voting results will be posted on web servers in HTML format.

Some more additional attributes of Online Voting System are:

1. Security: Ensuring the integrity and confidentiality of the voting process.
2. Accuracy: Guaranteeing accurate vote counting and recording.
3. Accessibility: Allowing equal access for voters with disabilities.
4. Anonymity: Protecting voter privacy and anonymity.
5. Auditability: Providing transparent and verifiable audit trails.
6. Availability: Ensuring the system is always accessible during voting periods.
7. Usability: Offering an intuitive and user-friendly interface.
8. Scalability: Handling large numbers of voters and votes.
9. Reliability: Minimizing system downtime and errors.
10. Verifiability: Allowing voters to verify their votes were recorded correctly.
11. Transparency: Providing clear information about the voting process.
12. Accountability: Tracing all system activities and changes.
13. Compliance: Adhering to relevant laws and regulations.
14. Flexibility: Accommodating different voting methods and systems.
15. Interoperability: Enabling seamless integration with existing infrastructure.

Chapter 7

IMPLEMENTATION

7.1 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 concurrent 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.

7.1.1 Design and Implementation :

1. A secure server that only allows clients with authentic names and passwords to cast votes.

2. 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.

3. Voters are registered by admin and the voter 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 voter is redirected to the secured Voting page.

6. The voters 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.

7.1.2 Requirements Python Libraries Required :

➔ Pandas

➔ Tkinter

➔ Socket

➔ Subprocess

7.1.3 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’ part.

6. Click on the ‘Run Server’ Button.

7. Use the rest of the Buttons as per your need.

7.1.4 How to Login

Admin Login :

➔ Admin ID : Admin

➔ Password : admin 5

Voter Login:

❑ Server should be running for voters to be able to login.

➔ Already registered voter I.Ds : 10001 to 10005

➔ Password (for already registered voters) : abcd

7.1.5 Workflow Description

In order Description to run & test this project :

1. Open terminal & run python homePage.py to open Home Page Window.

2. Log into Admin and press ‘Run Server’. This will run the Server in a new console window.

3. Now that the server is running, return to the admin home page window.

4. Press ‘Register Voter’ and enter details to register a new voter. Remember or note down the ‘Voter ID’ that you will receive on successful registration.

5. Press ‘Home’ to return to the Home. Now, press ‘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 error message if the Voter is invalid or 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.

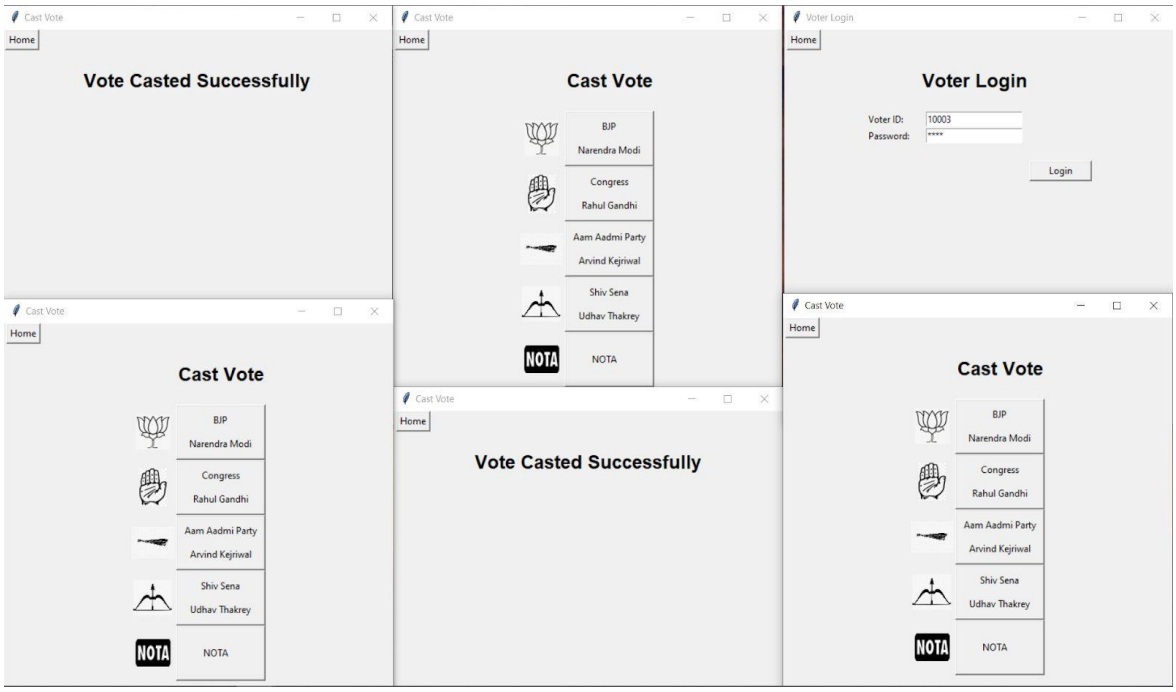

FIG:7.1-6 VOTERS

7.2 Code:

Code for home page

```
import subprocess as sb_p
import tkinter as tk
from tkinter import *
from Admin import AdmLogin
from voter import voterLogin

def Home(root, frame1, frame2):

    for frame in root.winfo_children():
        for widget in frame.winfo_children():
            widget.destroy()

        Button(frame2, text="Home", command = lambda: Home(root, frame1, frame2)).grid(row=0,column=0)
    Label(frame2, text="                                                    ").grid(row = 0,column = 1)
    Label(frame2, text="                                                    ").grid(row = 0,column = 2)
    Label(frame2, text="        ").grid(row = 1,column = 1)
    frame2.pack(side=TOP)

    root.title("Home")

     Label(frame1, text="Home", font=('Helvetica', 25, 'bold')).grid(row = 0, column = 1, rowspan=1)
    Label(frame1, text="").grid(row = 1,column = 0)
    #Admin Login
       admin = Button(frame1, text="Admin Login", width=15, command = lambda: AdmLogin(root, frame1))

    #Voter Login
       voter = Button(frame1, text="Voter Login", width=15, command = lambda: voterLogin(root, frame1))

    #New Tab
      newTab = Button(frame1, text="New Window", width=15, command = lambda: sb_p.call('start python homePage.py', shell=True))

    Label(frame1, text="").grid(row = 2,column = 0)
    Label(frame1, text="").grid(row = 4,column = 0)
    Label(frame1, text="").grid(row = 6,column = 0)
    admin.grid(row = 3, column = 1, columnspan = 2)
    voter.grid(row = 5, column = 1, columnspan = 2)
    newTab.grid(row = 7, column = 1, columnspan = 2)

    frame1.pack()
    root.mainloop()
```

```
def new_home():
    root = Tk()
    root.geometry('500x500')
    frame1 = Frame(root)
    frame2 = Frame(root)
    Home(root, frame1, frame2)

if __name__ == "__main__":
    new_home()
```

code for register voter

```
import tkinter as tk
import dframe as df
import Admin as adm
from tkinter import ttk
from Admin import *
from tkinter import *
from dframe import *

def reg_server(root,frame1,name,sex,zone,city,passw):
    if(passw==" or passw==' '):
        msg = Message(frame1, text="Error: Missing Fileds", width=500)
        msg.grid(row = 10, column = 0, columnspan = 5)
        return -1

    vid = df.taking_data_voter(name, sex, zone, city, passw)
    for widget in frame1.winfo_children():
        widget.destroy()
    txt = "Registered Voter with\n\n VOTER I.D. = " + str(vid)
      Label(frame1, text=txt, font=('Helvetica', 18, 'bold')).grid(row = 2, column = 1,
columnspan=2)

def Register(root,frame1):

    root.title("Register Voter")
    for widget in frame1.winfo_children():
        widget.destroy()

    Label(frame1, text="Register Voter", font=('Helvetica', 18, 'bold')).grid(row = 0, column
= 2, rowspan=1)
    Label(frame1, text="").grid(row = 1,column = 0)
    #Label(frame1, text="Voter ID:      ", anchor="e", justify=LEFT).grid(row = 2,column =
0)
    Label(frame1, text="Name:         ", anchor="e", justify=LEFT).grid(row = 3,column = 0)
    Label(frame1, text="Sex:             ", anchor="e", justify=LEFT).grid(row = 4,column =
0)
    Label(frame1, text="Zone:         ", anchor="e", justify=LEFT).grid(row = 5,column = 0)
    Label(frame1, text="City:            ", anchor="e", justify=LEFT).grid(row = 6,column =
0)
```

```
    Label(frame1, text="Password:   ", anchor="e", justify=LEFT).grid(row = 7,column = 0)

    #voter_ID = tk.StringVar()
    name = tk.StringVar()
    sex = tk.StringVar()
    zone = tk.StringVar()
    city = tk.StringVar()
    password = tk.StringVar()

    #e1 = Entry(frame1, textvariable = voter_ID).grid(row = 2, column = 2)
    e2 = Entry(frame1, textvariable = name).grid(row = 3, column = 2)
    e5 = Entry(frame1, textvariable = zone).grid(row = 5, column = 2)
    e6 = Entry(frame1, textvariable = city).grid(row = 6, column = 2)
    e7 = Entry(frame1, textvariable = password).grid(row = 7, column = 2)

    e4 = ttk.Combobox(frame1, textvariable = sex, width=17)
    e4['values'] = ("Male","Female","Transgender")
    e4.grid(row = 4, column = 2)
    e4.current()

     reg = Button(frame1, text="Register", command = lambda: reg_server(root, frame1,
name.get(), sex.get(), zone.get(), city.get(), password.get()), width=10)
    Label(frame1, text="").grid(row = 8,column = 0)
    reg.grid(row = 9, column = 3, columnspan = 2)

    frame1.pack()
    root.mainloop()
```

Chapter 8

DESIGN CONSTRAINTS

8.1 Language Constraints

The software will only operate in English and will only allow ballots that use 7-bitASCII. This is because the first release of this project is only expected to be an exploration of voting technology, so it is reasonable to assume that it will be used primarily in North America. Both ASCII and English are used as standards in international computing. Future versions of this software could be produced in other language, but the current implementation team will not have the time or the linguistic expertise to do so.

8.2 Software and Hardware Constraints

8.2.1 Voting Client

The Voting Client software will only be tested to run on Linux or Microsoft Windows 95 or higher. The system with the Voting Client must be capable of running Java bytecode and must have access to the Internet with high-speed access such as cable or DSL. This was stated as a valid assumption in the class lab. The Voting Client must be written in Java and the description language for the ballot must be written in XML. The minimum hardware requirements for the Voting Client (which are the same as those for the JDK 1.1.2) are:

- Pentium 166-MHz or faster processor

- At least 32 Mbytes of physical RAM

- 65 Megabytes of free disks pace

8.2.2 Election Server

The Election Server will be written in Java with any configuration files written in XML. The Election Server will only be tested in the Linux environment. It may or may not run on other operating systems. The number of concurrent voters using the Election Server at any one time will not be limited, in order to allow for scalability. However, for the purposes of this project it will only be guaranteed that the system will function properly with less than forty concurrent voters. This number is based on the number of computers in the Reynolds 008 lab, since it was stated in the class lab by Professor Stacey that it would not be unreasonable

to expect the class to be able to go to the lab and all vote in the Reynolds 008 lab. For similar reasons, the minimum hardware requirements for the Election Server are the same as the computers in

8.2.3 Election Database

The computer hosting the Election Database must have either MySQL or Postgres installed.

8.3 Computer Language Constraints

All configuration files must be in XML. The statistical reports must be generated in HTML. The database used by the server must use MySQL. All software must be written in Java.

8.4 Encryption Constraints

The development team is limited in the type of encryption that can be used for building the system by what is either available in the Linux and the Windows Operating System, what can be found in Java libraries or by what they can write themselves. It is not in the budget to purchase third- party encryption software.

8.5 Illegal Voter Activity Constraints

There is a danger that outside of a public polling place, a voter could be coerced into voting for a particular candidate, or selling his or her vote. It will also be difficult to control vote solicitation at the time of voting. The E-vote software will have no provisions to prevent any of these problems.

8.6 Installation Constraints

The installation of the Election Server will already assume that MySQL has been installed and that the computer running the Election Server will be able to connect to the MySQL database. It is also required that there be network access to the computer running the Election Server, and that there is a capable Network Administrator and Database Administrator to carry through the installation.

The installation of the Voting Client will assume that the target computer will have network communication already set up.

Chapter 9

FUTURE SCOPE

After discussing the benefits and features of the online voting system, it's time to think about its implementation and future scope. The implementation of the online voting system would require a significant investment in hardware, software, and internet connectivity. However, the benefits of enhanced security, a faster and automated voting process, and increased transparency justify the investment. In the future, the system could be improved by incorporating blockchain technology for added security and auditability. A more comprehensive voter database that includes biometric information, such as fingerprints or facial recognition, could enhance authentication and verification measures.

While the online voting system offers many advantages over traditional paper-based voting, there are still some concerns and challenges to consider. Ensuring equal access and participation, protecting voter privacy and preventing fraud and hacking are some of the key issues that need to be addressed. In conclusion, the online voting system has enormous potential to modernize the election process in India.

It is important that the results are easily accessible and that the system is implementation and future development will require a multistakeholder approach that includes government, transparent to maintain trust in the electoral process. Despite these challenges and concerns, addressing these issues and implementing a civil society, and the private sector. With the right strategies and investments, the online voting system could become a reliable and secure tool for democratic participation and decision-making.

Chapter 10

CONCLUSION

In conclusion, Online Voting System is a highly innovative and technological solution to many of the challenges faced in traditional voting systems. It not only simplifies voting process but also saves time and resources. With secure authentication and verification measures in place, the system offers transparency and accountability in the electoral process. However, the implementation of the online voting system still raises concerns regarding security, privacy, and accessibility. It is important to address these issues before making the system available to the public. The online voting system has tremendous potential to revolutionize the electoral process and ensure greater participation and representation for all.

The way forward is to focus on developing a comprehensive security infrastructure and addressing concerns to make the system accessible to all. By doing so, we can create a more inclusive and democratic electoral process for the future.

Chapter 11

APPENDIX

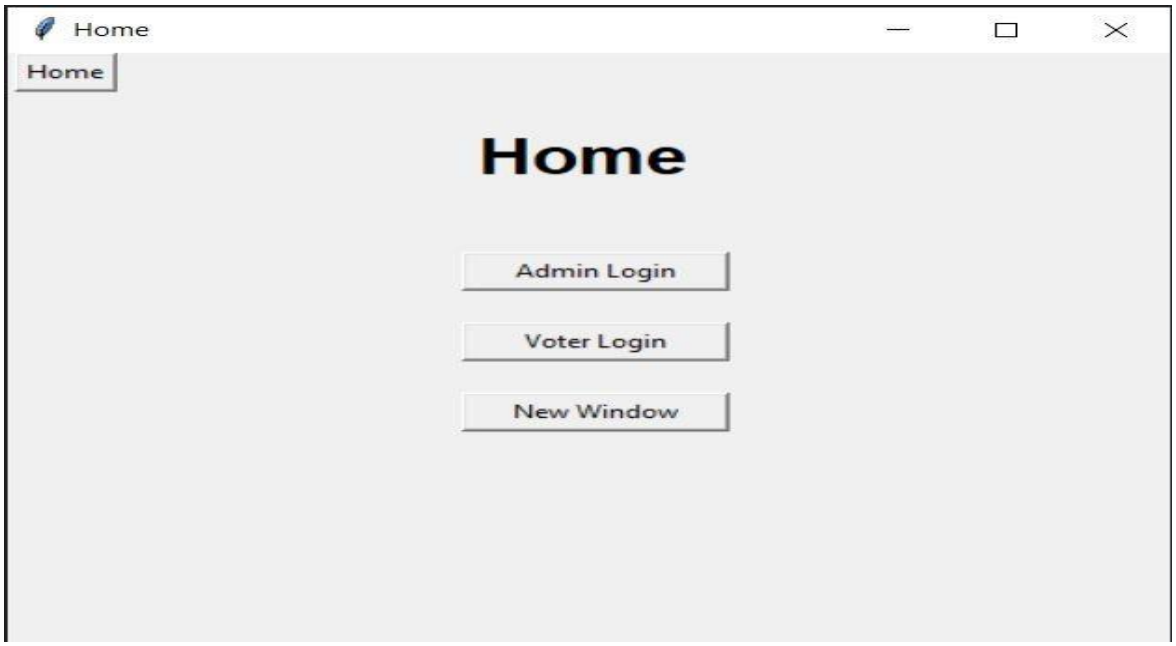

FIG:11.1-HOME PAGE

FIG:11.2-ADMIN LOGIN

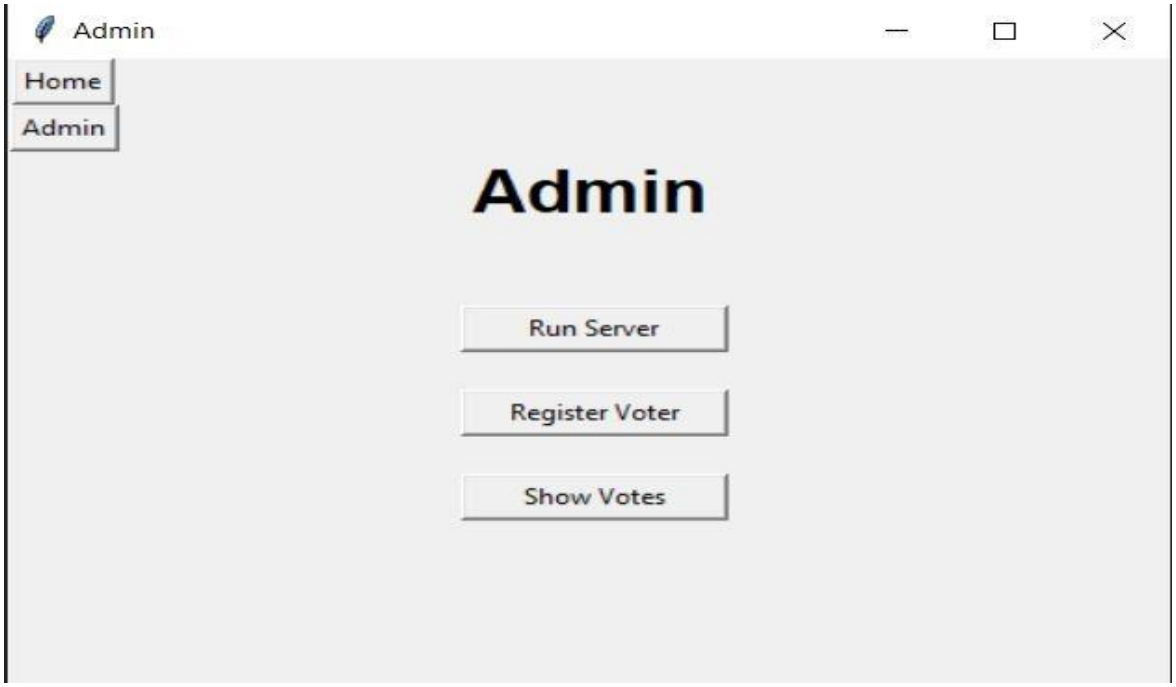

FIG:11.3-ADMIN PAGE

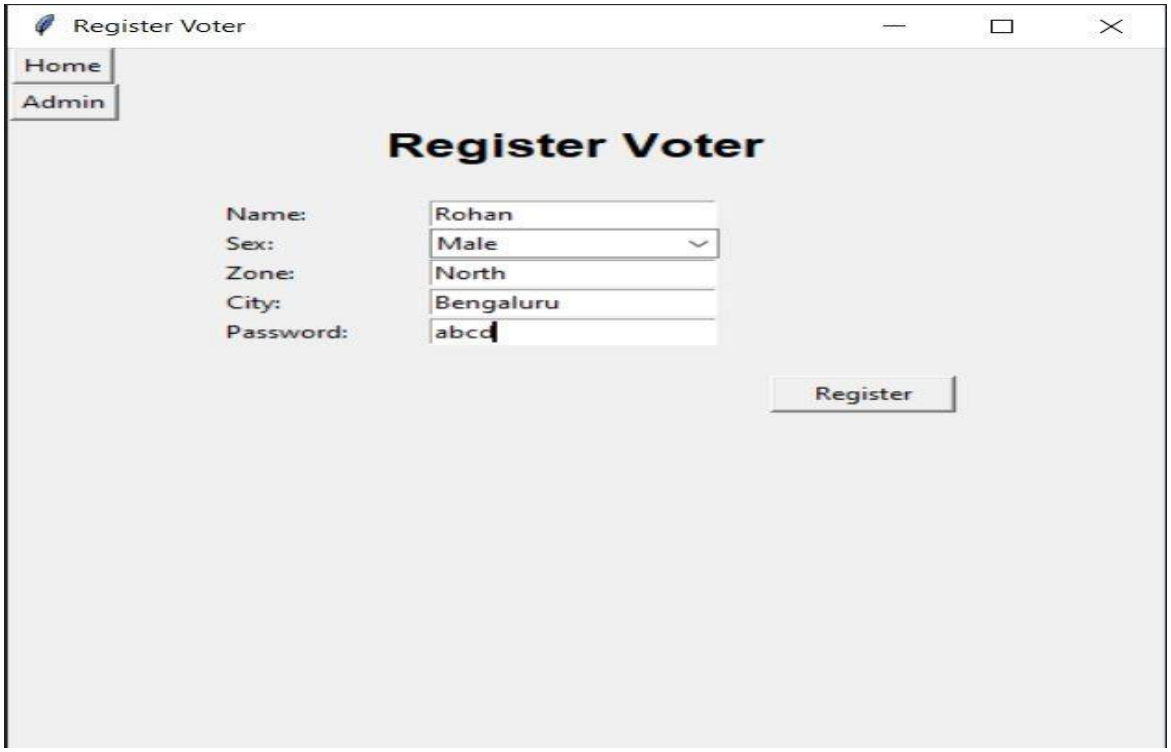

FIG:11.4-REGISTER VOTER

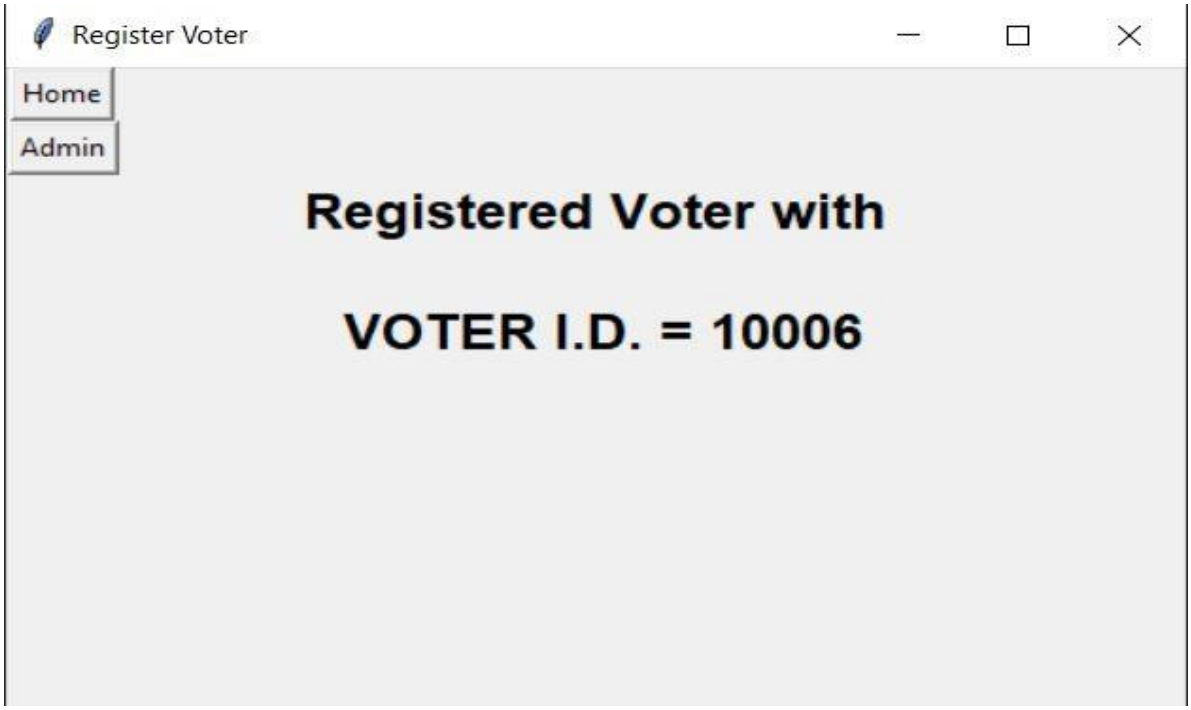

FIG:11.5-REGISTERED VOTER I.D

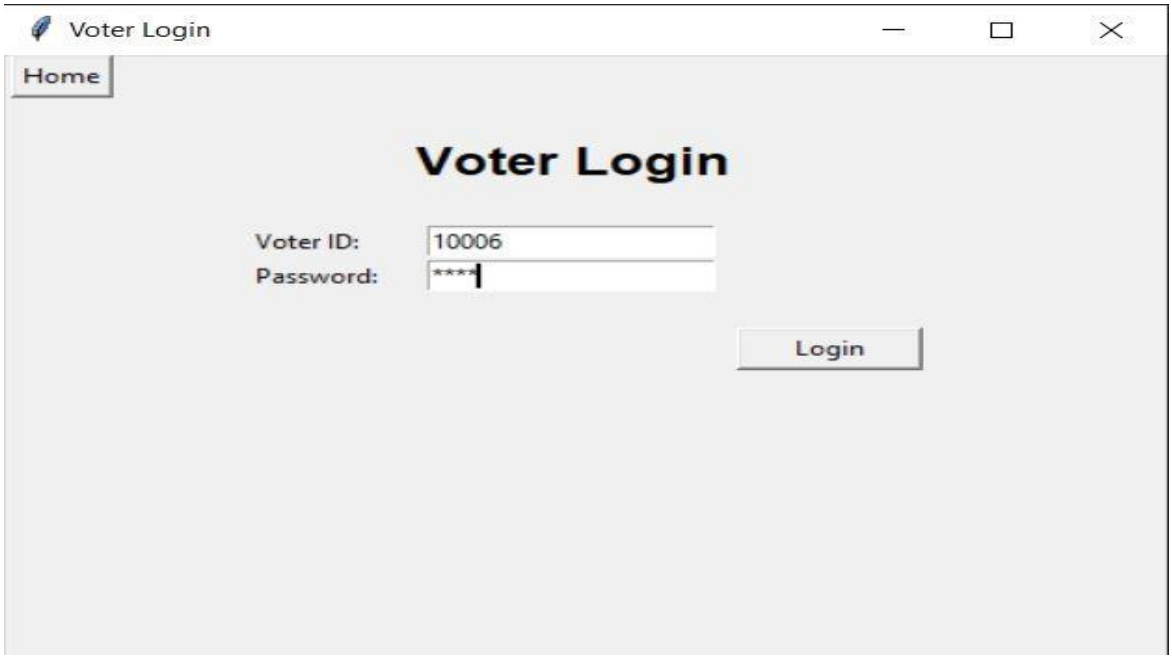

FIG:11.6-VOTER LOGIN

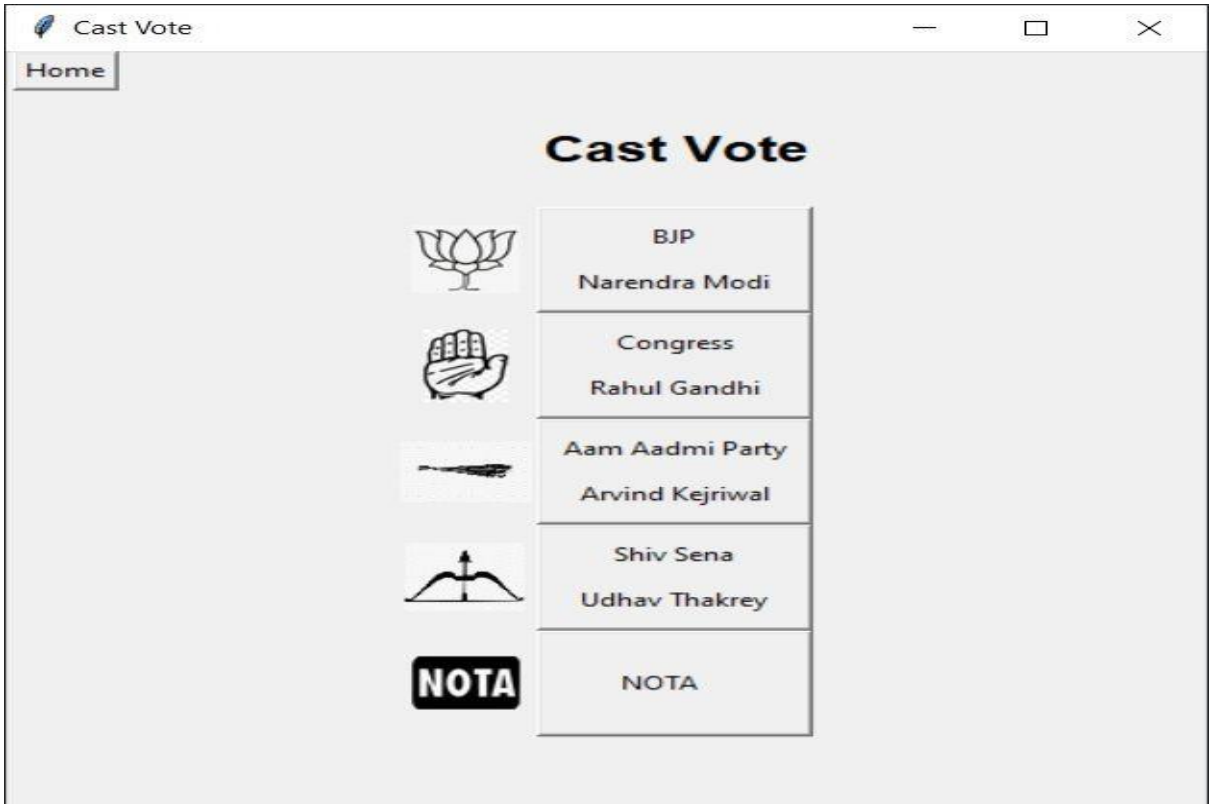

FIG:11.7-CAST VOTE

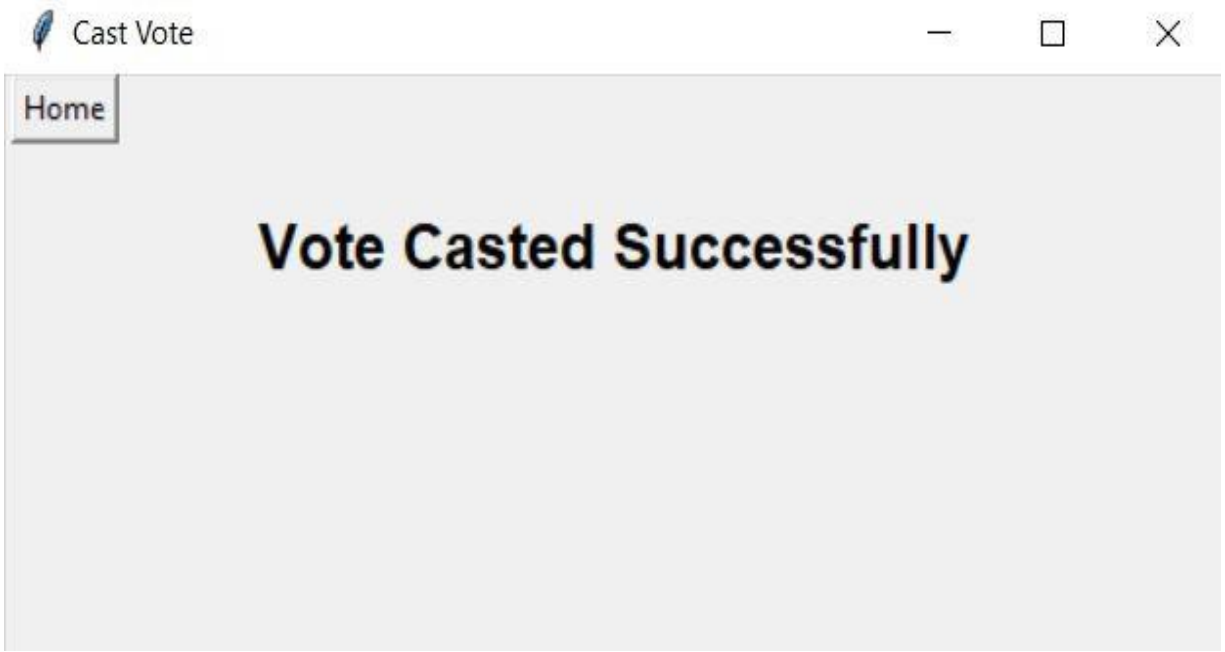

FIG:11.8-VOTE CASTED SUCCESSFULLY

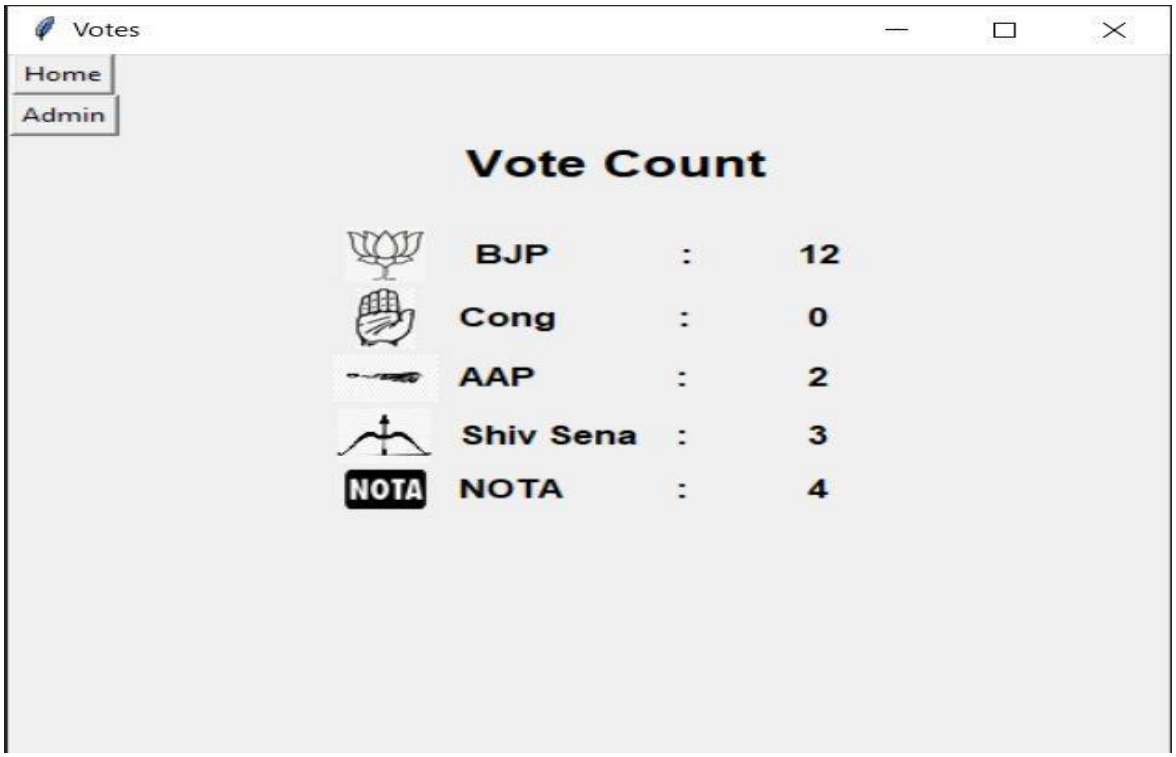

FIG:11.9-VOTE COUNT

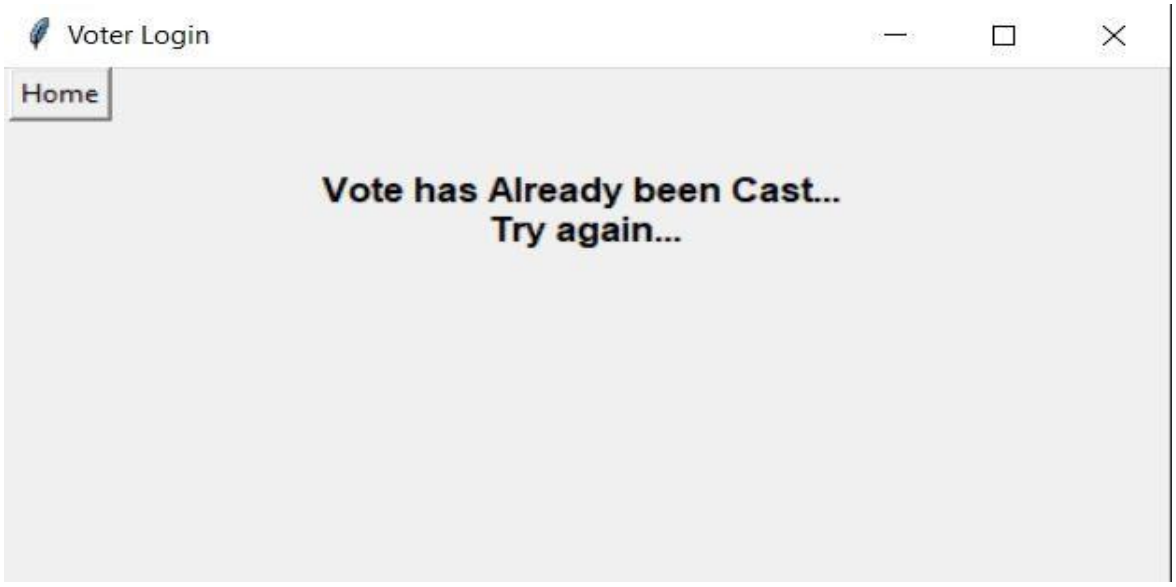

FIG:11.10- IF VOTE HAS BEEN ALREADY CASTED

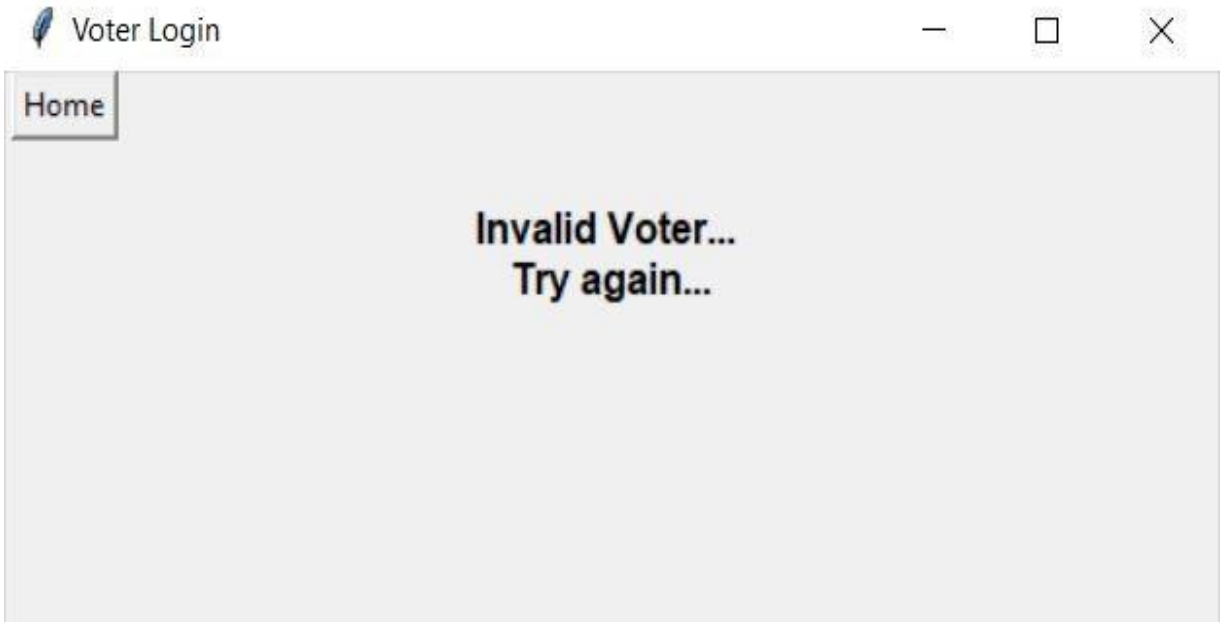

FIG:11.11- INVALID VOTER

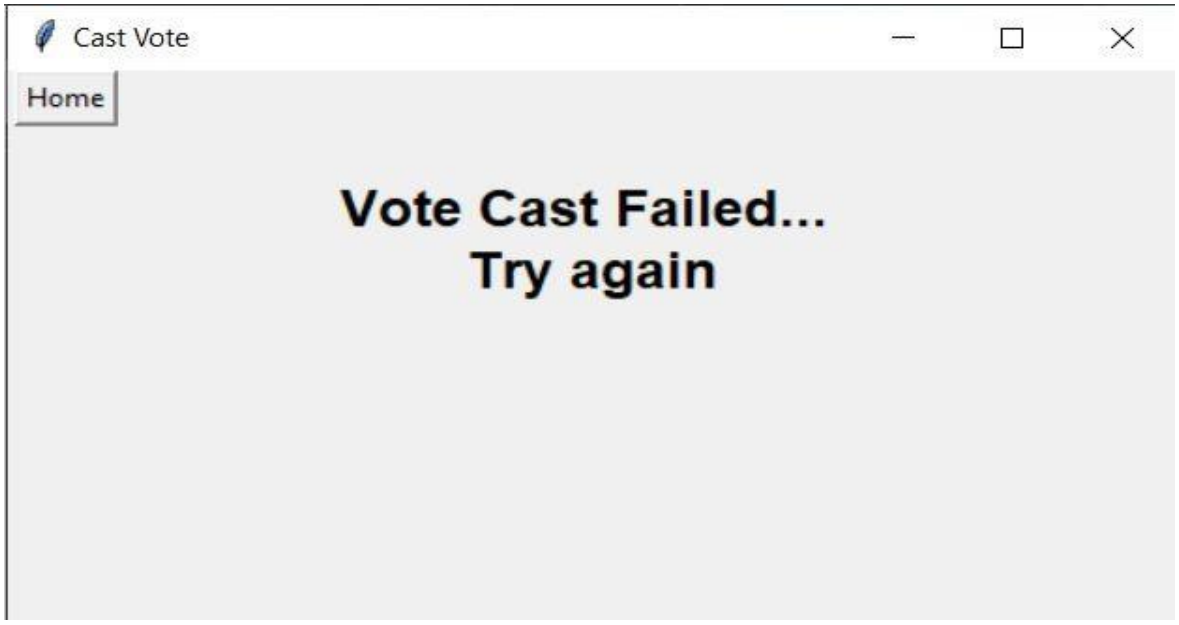

FIG:11.12-VOTE CAST FAILED

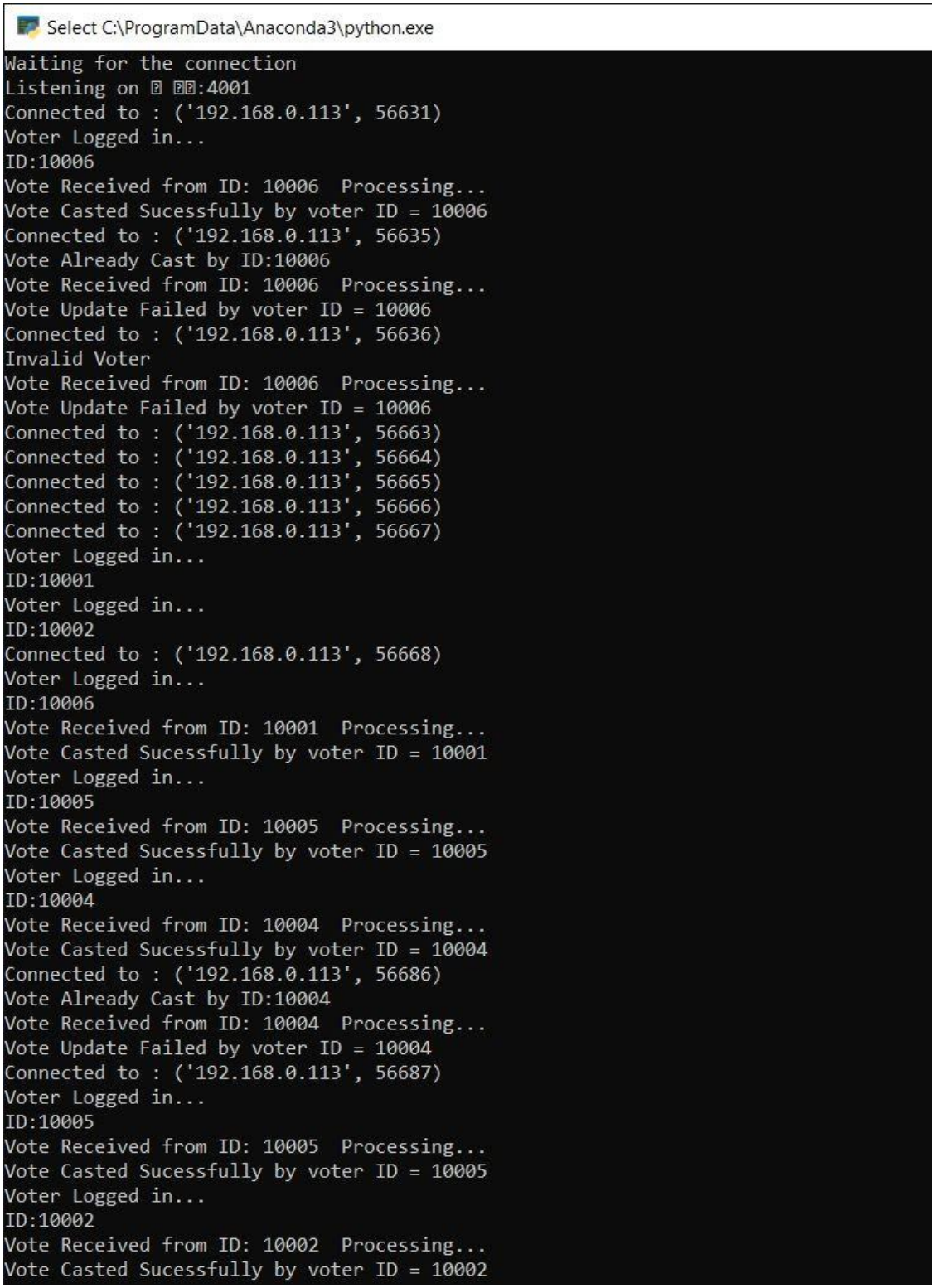

FIG:11.13-SERVER

	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

FIG:11.14- VOTER 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

FIG:11.15-CANDIDATE INFO DATABASE