PROJECT

ELECTRONIC VOTING SYSTEM

Student Minor Research Project

ELECTRONIC VOTING MACHINE DEMO PROGRAM IN C++

Under RUSA 2.0 Scheme

(Through Ch.S.D.St.Theresa's College for Women (Autonomous), Eluru, AP)

Submitted by

Ms Y S Neeraja, III B.Sc. MECs (Reg.No.11705004)
Ms B Likitha, III B.Sc. MECs (Reg.NO.11705005)

Under the guidance of

Smt.Ch.Lakshmi
Assistant Professor & Project Advisor

DEPARTMENT OF COMPUTER SCIENCE

SRI Y N COLLEGE
(AUTONOMOUS)
Thrice Accredited by NAAC at 'A' Grade
Recognized by UGC as "College with Potential for Excellence"
Narsapur-534275, AP, India
December -2019

DEPARTMENT OF COMPUTER SCIENCE

SRI Y N COLLEGE

(AUTONOMOUS)

Thrice Accredited by NAAC at ‘A’ Grade

Recognized by UGC as “College with Potential for Excellence”

Narsapur-534275, AP, India

CERTIFICATE

This is to certify that the project work entitled ***“Electronic Voting Machine Demo Program in C++”*** *is bonafide work carried out by Ms Y S Neeraja (Reg.No: 11705004), Ms B.Likitha (Reg.No: 11705005), submitted in Third Year of the degree B.Sc. in Computer Science during the year 2019-20 is an authentic work under my supervision and guidance.*

To the best of my knowledge, the matter embodied in the project work has not been submitted to any other College/Institute.

Date: 29-12-2019

Smt.Ch.Lakshmi
Project Advisor
Dept.of Computer Science

ACKNOWLEDGEMENT

We place on record and warmly acknowledge the continuous encouragement, invaluable supervision, timely suggestions and inspired guidance offered by our Project advisor, ***Smt.Ch.Lakshmi,*** *Assistant Professor, Dept.of Computer Science,* ***Sri Y N College (Autonomous), Narsapur*** *in bringing this report to a successful completion.*

We are grateful to ***Ms. K R Sravanthi****, Asst.Professor, Dept. of Computer Science for permitting us to make use of the facilities available in the department to carry out the project successfully. Last but not the least we express our sincere thanks to all of our friends who have patiently extended all sorts of help for accomplishing this undertaking.*

Finally we extend our gratefulness to one and all who are directly or indirectly involved in the successful completion of this project work.

Ms. Y S Neeraja	**Ms. B Likitha**
III B.Sc MECs	III B.Sc MECs
Reg. No 11705004	Reg. No.11705005

DECLARATION

We, the undersigned, declare that the project entitled “Electronic Voting Machine Demo Program in C++”, being submitted in Third Year of Bachelor of Science in Computer Science, Sri Y N College (Autonomous), is the work carried out by us.

Ms. Y S Neeraja	**Ms. B Likitha**
III B.Sc MECs	III B.Sc MECs
Reg. No 11705004	Reg. No.11705005

ABSTRACT

Electronic Voting Machine (EVM) is a simple electronic device used to record votes in place of ballot papers and boxes which were used earlier in conventional voting system. Fundamental right to vote or simply voting in elections forms the basis of democracy. All earlier elections be it state elections or centre elections a voter used to cast his/her favorite candidate by putting the stamp against his/her name and then folding the ballot paper as per a prescribed method before putting it in the Ballot Box. This is a long, time-consuming process and very much prone to errors.

This situation continued till election scene was completely changed by electronic voting machine. No more ballot paper, ballot boxes, stamping, etc. all this condensed into a simple box called ballot unit of the electronic voting machine. Because biometric identifiers cannot be easily misplaced, forged, or shared, they are considered more reliable for person recognition than traditional token or knowledge based methods.

So the Electronic voting system has to be improved based on the current technologies viz., biometric system. This project discusses the Electronic Voting Machine by using the software program in C++.

INDEX

1. INTRODUCTION

The function of compiling, editing and processing data makes the voting system complicated and tedious process. Electronic voting system is an effort in the direction to use computer for the purpose of voting system. It is both user friendly as well as time saving.

This is a foundation level. Emphasis has been given in this application to replicate all the processes required in a traditional system. EVM system is a self-sufficient with all the information required about the voters, candidate and votes. Once the information is fed to the system it can identify individual voters, their votes and the candidates. Tedious function such as counting the votes has been simplified in this system. The basic function of the polling has been reduced to single operation. A single polling manager can manage large number of voters.

The program code has been Written in 'C++' language has been chosen because of its procedural nature and the flexibility it provides in interacting with hardware and operating system efficiently.

NEED TO COMPUTARISE THE SYSTEM

Information is needed in the organization for planning, staffing and controlling purposes. Regardless of the nature of the information required, the information should process the characteristics of accuracy, timeliness and relevancy. In the recent years need for information improvement by reports lacking one or more of these characteristics and by increased paper work volume, rising casts and pressures from outside changes.

Fortunately computers thrive on repetitive large volume processing tasks, are fast and accurate. The processing capability in many organizations has been stained by:

1. Growth in size and complexity of the organization.
2. The increased requirements for data from external sources.
3. The demand of administrators for more information.

More than a million new pages of data are generated each minute of the day in offices . Compare to other processing method, the use of computers may make it possible for certain administrative costs to be reduced while the level of processing activity remains stable. The increased cost and clerical labor materials and other expenses associated with the data processing operation require eventual managerial attention.

We all agree that meaningful information is timely information. But with an increase in volume and size of an organization, there is only a reduction in the speed of processing. Rapid changes are taking place in the world socially, economically and technically. Such changes have a significant impact on the environment which organization must operate on the planning that managers must do and on the information that must have. Many educational administrators are responsible for supervising the activities of a large number of schools scattered over a district. They must have accurate information if they are to control such an effort properly. But if a data processing operation is stained to or beyond the capacity for which it was originally planned, inaccuracies will begin to appear. Inadequate control will permit inadequate performance. Thus the administrator will logically demand better quality in the information he receives. It is due to these reassures (increased paper work volume, cost, pressure from outside changes demand for timeliness and demand for quality) that most if the organization today is opting for computers to do data processing for them.

Computers are most efficient used in processing operation that includes:-

Large volume of input:- Greater the volume of data that must be processed to produce the needed information, the more economical computer processing becomes relative to other possible methods.

Repetition of projects: - Because of the expenses involved in preparing a task for computer processing, it is frequently more economical to use the computer for repetitive purpose.

Desired and necessary greater speed in processing:- Greater the need for timely information, greater will be quite accurate if the task to be performed has been properly prepared.

THE VOTING SYSTEM:

The voting system can be described as “A process in which the input (voter) votes in favors of a given decision (candidate)”. Votes are procured, counted and a statically result is provided.

Traditionally there are two basic types of voting system:-

Open voting system: - In this system the voter votes in favors of a decision without hiding his identity, example of system is rising of hands. This is in itself a very elementary system and useful when the level of decision to be taken is comparatively less effective to the voter. The number of voters should be less and countable and discipline and honestly is maintained by the voters. The voting system is people oriented.

Second voting system: - In this system identity of the voter is hidden and once the vote has been given it is not possible identifies a given vote with the voter. In this system paper work is usually done, as the identity of the voter has to be concealed. A very large number of voters can take part in such system. The whole process in itself system oriented and have security secrecy and counting becomes easier.

Computer Voting System

Manual voting system at large scale becomes a tiring and costly affair in terms of money, manpower and time. The use of computer can easily reduce the effort and make voting an easy as well as voter and candidate friendly affairs. It can reduce the amount of money spent, manpower and time to a very large extent.

In the present system of voting the individual votes are registered in form of ballot papers. The names of candidates is written or printed in the ballot papers in serial orders, which can be easily recognized by the voters. A separate list of voters is made so that the voters can be allowed to sequentially and chances of non-voters or revolting of an individual can be controlled. The whole system is done under the control of a residing officer. Each voter is given a ballot paper on which he stamps his individual choice. The ballot paper is then folded in a particular way and then insert in the ballot box. As soon as an individual voter casts his vote, his name is deleted from the voter list.

At the end of polling the ballot boxes are sealed and sent for counting. The counting of the ballot papers is done in a redefined method. The results are declared after getting convinced that the counting has been done properly. The process of counting takes time since it is done manually.

Problems in Existing System:

The present system has been marked by many problems some of which have been enumerated below: -

Expensive: - The present form of voting is an expensive affair. If needs a lot of preparation in terms of time and money. The cost of paper work and stationary is one of the most expensive parts. Expenses are incurred in processing ballot boxes and ballot papers. If in case ballot papers are printed wrong re-printing has to be done.

Time consuming - A lot of time is consumed in counting of votes. The manual counting is in itself a tedious job and hence requires time.

Invalid votes: - Manual voting has a regular problem of invalid votes. A number of votes become invalid due to a number of unintentional reasons.

Manipulation work: - Voting process at times is hindered by presence of external elements to favor a given candidate. Examples are booth capturing.

Manpower required: - The present system of voting involves the use of lots of manpower at each and every stage of voting process.

User identification: - This poses one of the serious problems of current voting system. During voting process many people cast there vote more than one times. This causes problems as wrong people are selected.

Manipulation in counting: - As counting process is done manually it may happen that votes are wrongly counted or they may be manipulated in some way in favors of some candidates.

Portability factor: - After polling is over ballot boxes are usually taken to some other places for counting. This is risk as the ballot boxes may be stolen or there may be some mishap reroute to counting system.

Software Program of the EVM:

Despite design features that make the election software complex to extract the information from the control unit processor, a criminal has a variety of options to steal the information from it. These include decapsulating the chip and examining it under a microscope. Here we did not attempt to extract the software using these methods; once the software is extracted, it is straightforward to reverse- engineer it using standard disassembly tools.

The software (evm.cpp) developed here for the demo is tested on Turbo C++ Version 3.0 IDE. When the user runs the program, it displays a real-time voting machine on the screen with the list of the candidates and corresponding buttons to vote for them (refer Fig. 2). The program requires ‘C’ graphics files such as egavga.bgi, goth.chr, lcom.chr and trip.chr for proper display.

In order to vote for a party, the user has to click the respective button using mouse. A beep sounds to confirm successful submission of vote.

At any point of time if you want to check the status of the votes being polled, just click ‘result’ button in the control panel (not shown here). The next screen will display the percentage of votes received by different candidates in the form of a bar chart. To exit the program, click ‘exit’ button.

2. SYSTEM REQUIREMENT SPECIFICATION(SRS)

SYSTEM REQUIREMENTS:

The aim of the system to be developed ------- Electronic-voting system is to develop software, which should automate the process of voting of an electoral system. The system is supposed to be used as a subsystem in a layer electoral system, which could be a manual system or a computerized one. Therefore the proposed system must be able to function under both circumstances.

The basic functionality that the system must provide is

1. The system must provide for manipulation of candidate data.

The manipulation of candidate data consists of: -

Adding a new voter record in the existing voter list. The system must provide enough mechanisms to check the data validity and integrity.

Deletion of existing voters from the voter list. The system must be able to notice and notify the deletion request for a non-existent record. The system should have a mechanism of asserting record deletion to avoid accidental deletion of record.

Modification of voter records in the voter list. The system must be able to modify details of a voter for any specific data field, or for all the fields. The system must provide for validation in this module also. The system should have a mechanism of asserting record modification to avoid accidental modification of records.

2. The system must provide for manipulation of candidate data. The manipulation of candidate data consists of: -

Adding a new candidate record in the existing list. The system must provide enough mechanisms to check the data for validity and integrity.

Deletion of existing candidates from the candidate list. The system must be able to notice and notify the deletion request for anon-existent record. The system should have a mechanism of asserting record deletion to avoid accidental deletion of records.

Modification of candidate records in the candidate list. The system must be able to modify details of a candidate for any specific data field, or for all the fields. The system must provide for validation in this module also. The system should have a mechanism of asserting records modification to avoid accidental modification of records.

3. The system must have a rich set of utilities for listing the various details of the candidates as well as voters. The listing should be both a generalized listing for all candidates or voters. And detailed listing of an individual candidate or voter.

4. The system must provide a method and interference for the voting process. In which a voter can select a candidate from the list of candidates. There should further be a method to display the details of the winner of a particular voting process. The details of the margin of the victory should also be displayed.

5. The system must provide interference to print the details of candidates and voters. In both a summarized format and a detailed individual report.

6. The system must provide help whenever necessary. And should also provide tips, validity criteria etc… while data manipulation.

7. The system must have easy way of switching in between different modules.

NON-FUNCTIONAL REQUIREMENTS:

Nonfunctional requirements are just as important to your business analysis as the functional requirements when it comes to defining the look and feel of the solution. Nonfunctional requirements are a challenge because different people interpret them differently from organization to organization (or even from department to department in the organization). You need to understand a lot about the people using the solution and make sure your nonfunctionals document its performance.

You create the nonfunctional requirements based on your elicitations from the users, who they are, and what their expectations of the system performance are. When you create nonfunctional requirements, you need to think about things like the following:

Performance: How well does the system perform? To understand the performance requirements, ask stakeholders questions such as "What are the number of concurrent users?", "What are the system or query response times?", and "What is the system's capacity in terms of memory, disk space, and data volumes?"

Security: Who has access to the system, and how much access do they have? To understand the security requirements, ask questions such as "Which users are authorized to perform which functions?", "What is the privacy of the information being captured and stored?", and "What features need to be in place to log user access and authenticate users?"

Reliability: *Reliability* is how the system operates based on the expectation of the end-user. Think about buying a car. You probably *purchase* a car because of the *functionality* (0–60 mph in 8 seconds, A/C, satellite radio, and so on), but you probably think about going to shop for that new car because of the *reliability* of the car. Similarly, you want to make sure you find out how consistently the business wants the solution to perform and what maintenance and support you need to make sure it stays that way. To elicit the reliability requirements, ask questions such as "When is the system expected to be available?", "What downtime does the system have for the administrators to perform maintenance, and when is the best time to schedule downtime?", and

“What notification do the users need when the system is going down for maintenance? How much advance notice should they receive?”

Compatibility: *Compatibility* refers to the extent to which the solution plays nice with other applications. To elicit compatibility nonfunctional requirements, ask questions such as “What common standards, common technology, and protocols exist on the workstation?”; “How well does the solution work with the common build?”; “What kinds of data exchange do you envision?”; and “What information (data) must be exchanged with other systems?”

Maintainability: *Maintainability* deals with how easy the system is to maintain and repair. To elicit the nonfunctionals for maintainability, ask questions such as “What is the ability to change one component without affecting others?”, “What effects do the maintenance activities have on customers, users, and employees?”, and “Who performs system upgrades? Who is responsible for interfaces?”

Transferability: *Transferability* refers to the ease with which a system can be transferred to a different hardware or software environment. Some of these concerns are lessening now that many companies are creating browser-based applications, yet these concerns have expanded with the mobile apps (like those you see on your smartphone) and the different versions and standards for e-readers.

Usability: *Usability* concerns the ways by which the user is able to learn, operate, and interpret the system results. This category includes ease of entry, learning, and handling, as well as the system’s intuitiveness.

Metrics and measurements: With any nonfunctional requirement, you must understand what measurement criteria you’ll use to determine whether the requirement is successful and met. You’re defining how well the solution meets the requirements. To elicit the metric, ask questions like “What are some aspects surrounding that requirement that you can measure?” and “What are the acceptable measurement time frames that are acceptable for the stakeholder?”

SYSTEM SPECIFICATION:

MINIMUM HARDWATRE

REQUIREMENTS PROCESSOR:

DUAL CORE OR

ABOVE RAM: 2GB OR ABOVE

HARD DISK: 256GB OR ABOVE

PRINTER: LINE_PRINTER (DOT

MATRIX/INKJET) KEYBOARD:NORMAL OR ABOVE

APPLICATION SOFTWARE

C++ S/W BORLAND C++/ TURBO

C++ OPERATING SYSTEM:

WINDOWS 98 S/W

LANGUAGE: C++

3. SYSTEM DESIGN

DESIGN SPECIFICATION:

To implement the requirements as specified in the system requirement specification document of Electronic Voting System. The proposed system is to be developed using “c++” language. The system has a main module, which will have several other modules as depicted in the structure diagram of the system.

The functionally requirements will be fulfill as: -To provide for manipulation of voter data

A new voter record is added in the existing voter list, by using add-vote () Module. This module will use the voter structure to input voter details. Each field in the voter structure will be checked for the integrity constraints. As specified in SRS, the details are added into existing voter detail file voter.dat. This file ensures the secondary storage of data.

Deletion of existing voters from the voter list is done by the module delete-voter (). The system checks for the existing of the record to be delete in the file voter.bat. For nonexistent data it notifies the user .The module also record deletions by flashing a message.

Modification of voter records in the voter list is handled by a module modify-voter (). This module reads the voter No from the user searches has its existence in the file voter.dat. If the record is found the details are displayed where typing new values can modify them. The module also assents record modification by flashing a message.

2. To provide for manipulation of candidate data.

A new candidate record is added in the existing candidate list, by using add-candidate () module. This module will use the candidate structure to input candidate details. Each field in the candidate structure will be checked for the integrity constraints as specified in SRS. The details are added into exiting voter-details file voter.dat. The file is ensures the secondary storage of the data.

Deletion of existing candidates from the voter list is done by the module del ete- candidate (). The system checks for the existence of the record to be deleted in the file candidate.dat. For non-existent data it notifies the user. The module also asserts record deletion by flashing a message.

Modification of voter records in the voter list is handled by a module modify-candidate (). These module reeds the voter numbers from the user searches its existence in the file candidate.dat. If the record is found the details are displayed where typing new values can modify them. The module also asserts record modification by flashing a message.

4. The listing of the various details of the candidates as well as voters. Handled by four different module

dislay-code (),
showlist-voter
(), display-
candidate(),
showlist-candidate () modules give detailed listing of an individual candidate or voter showlist-voter ().
Showlist-candidate () gives generalized listing for all candidates of voter.

5. The voting process is handled by the module vote (), fmain (), find (), showlist (). A voter can select a candidate from the list of candidates, this functionality is provided by module vote (). The method to display the details of the winner of a particular voting process is handled by find () and showlist (). The detail of the margin of the victory is dislayed by showlist().

6. The system provides interface to print the details of candidates and voters by using module print (), which is called from different modules with different values to print different values to print different data as per the specification of SRS.

7. The system provides help by a module help (). And provides tips, validity criteria etc, by displaying message at the screen using a module statusbar().

DATA DICTIONARY

FUNCTION	DESCRIPTION
1. ADD_VOTER()	This function adds voters name to the voter list
2. VOTER_INFORMATION ()	This function gives information about the voters, name, address etc by putting its voter number.
3. VOTER_LIST ()	This function shows list of the voters
4. ISSUE-VOTING_CARD ()	This function ISSUES voting cards to the voters for validating the election.
5. RESET_STATUS()	This function resets the date to get it filled again f illed.
(A) FOR CANDIDATE	To reset votes received by candidate to zero.
(B) FOR VOTER	To reset all the status in the voter file.
6. DELETE_VOTER () .	To give the voter name to delete voter record.
7. ELECTION ()	This function is used to give votes by the voters.
8. VOTE_CALCULATION ()	To calculate votes given by the voters.
9. ADD_CANDIDATE()	This function adds the candidate record to the Candidate file.
10 CANDIDATE_INFORMATION ()	This function gives the information of the candidates.
11 DELETE_CANDIDATE ()	To give candidate serial number to delete recor
12. VOTING_LIST ()	To display of the votes received by the candidates.
13. ADD_VOTES	To add votes in the favors of the given candidate.
14. VOTING_DATE ()	To display the date of voting.
15. SET_DATE ()	To set the date for the new election.
16. VALID_DATE ()	To return the validity of today’s date.
17. RETURN_DATE ()	To return the election date.
18. LINE_HOR ()	To draw horizontal line.

Data Base Tables:-

This project uses many tables:

- Admin
- Voter
- Candidate

Admin Table:-

Field Name	Data Type	Description
Username	Varchar	Login id for Admin.(Primary key)
Password	Varchar	Password for Login

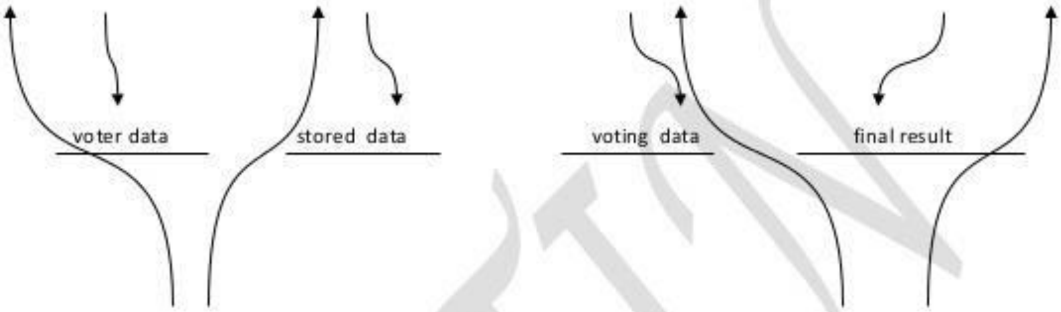
Voter Table:-

Field Name	Data Type	Description
VoterId	Integer	Login id for Voter(Primary key)
Name	Varchar	Name of the voter
Sex	Varchar	Sex of voter
Age	Integer	Age of voter
City	Varchar	City of voter

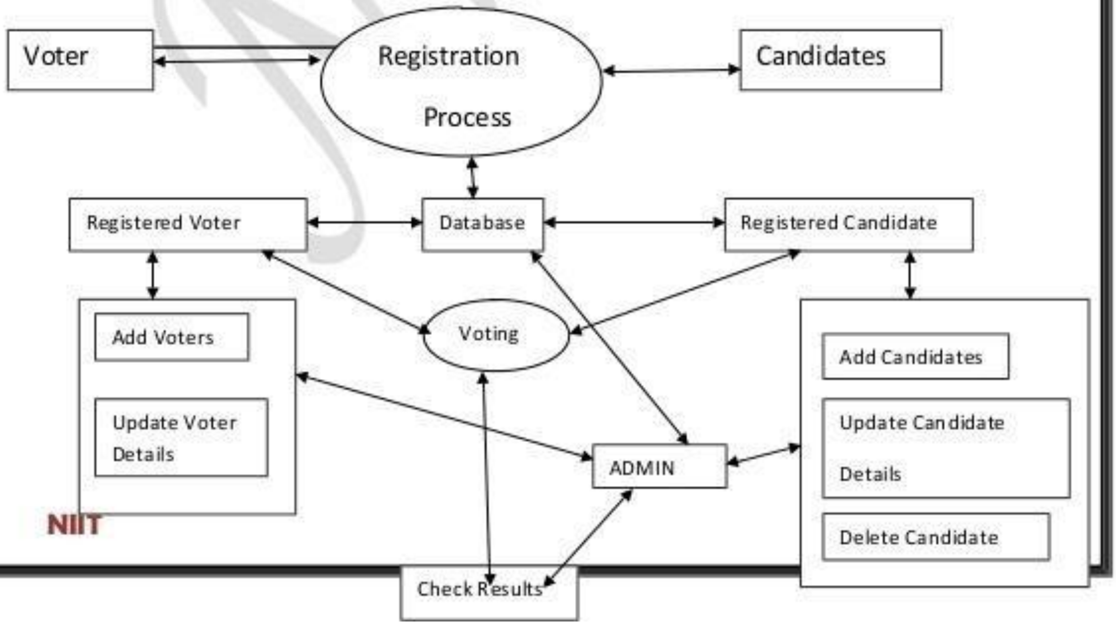
Security	Varchar	Security Question
Status	Boolean	Status of voter(he/she can vote or not)

Candidate Table:-

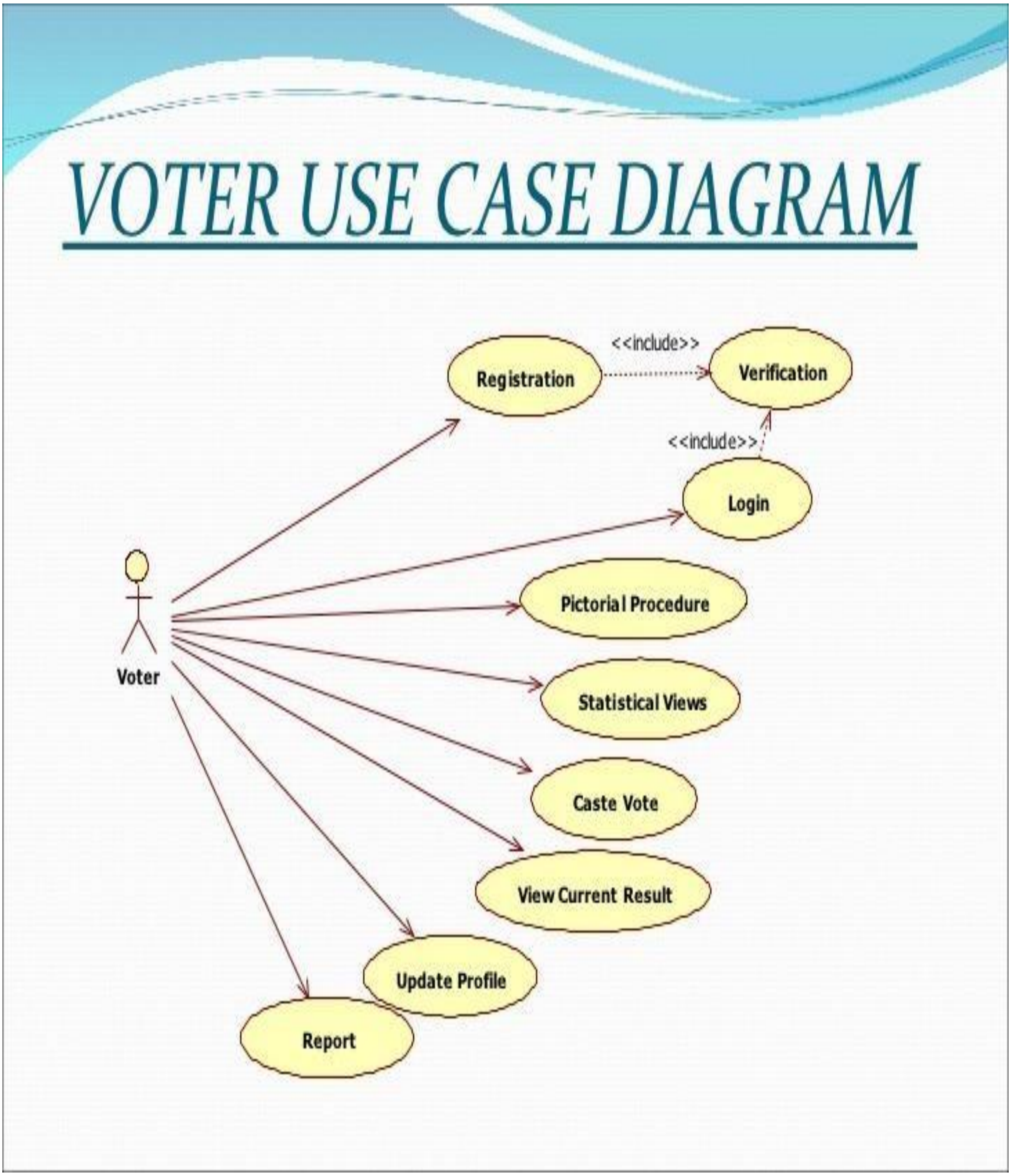
Field Name	Data Type	Description
Symbol	Varchar	Party Symbol (Primary key)
Name	Varchar	Name of the voter
Sex	Varchar	Sex of voter
Age	Integer	Age of voter
City	Varchar	City of voter
Count	Integer	Count the no of votes


voter data
stored data
voting data
final result

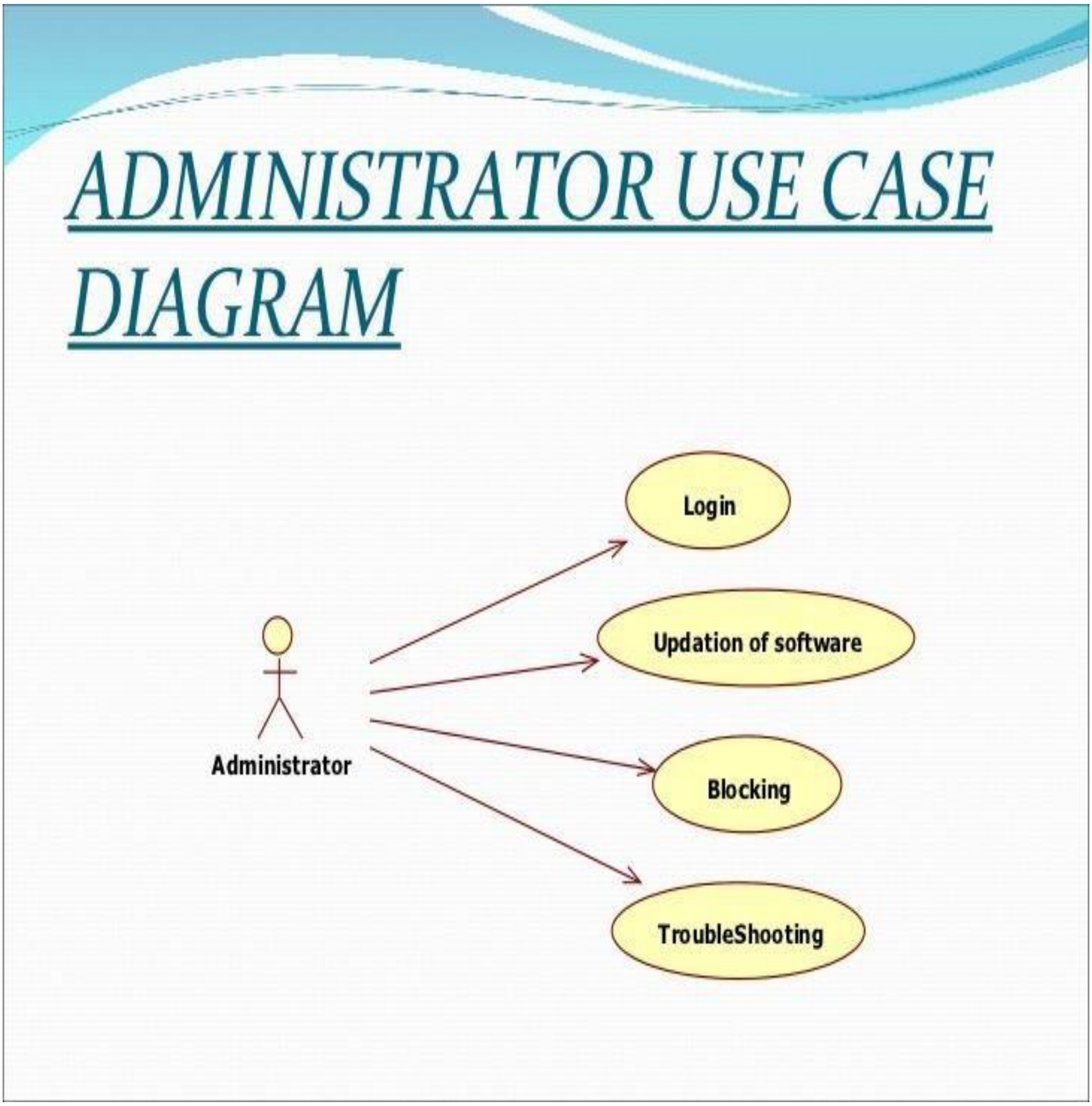
ER Diagram:


Voter
Registration Process
Candidates
Registered Voter
Database
Registered Candidate
Add Voters
Update Voter Details
Voting
Add Candidates
Update Candidate Details
ADMIN
Delete Candidate
Check Results

Use case Diagram

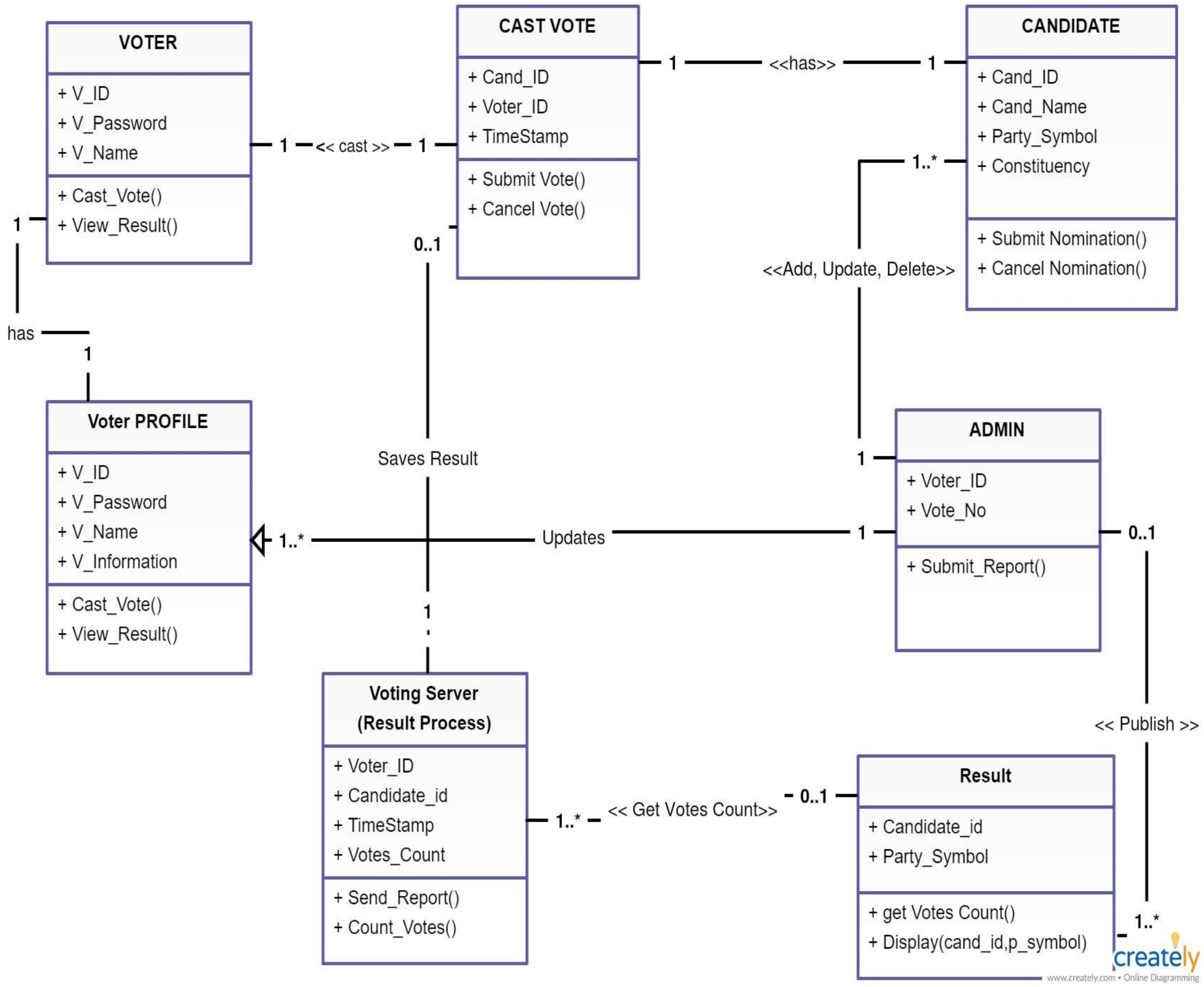


VOTER USE CASE DIAGRAM

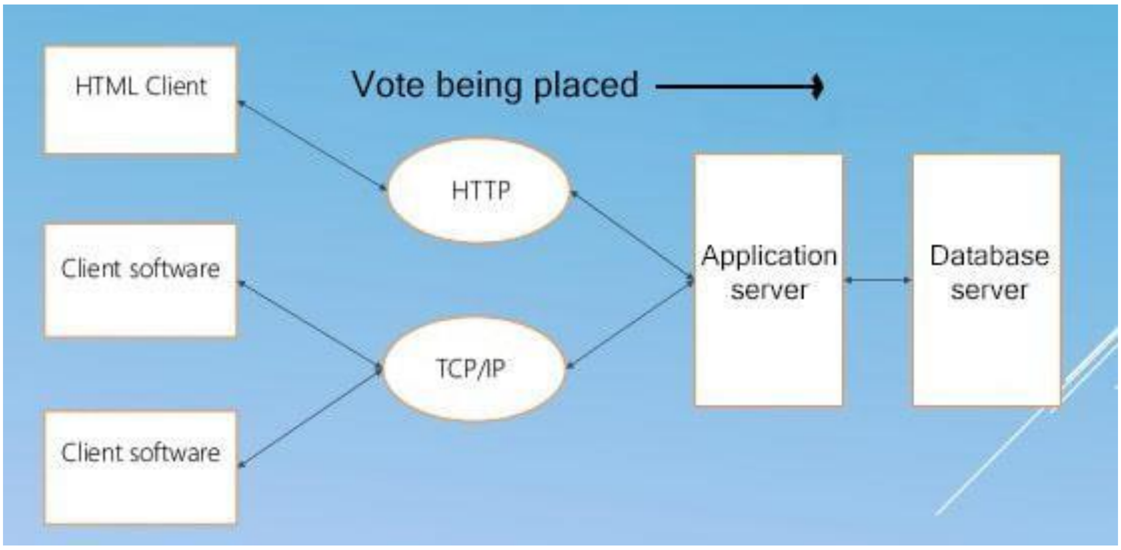


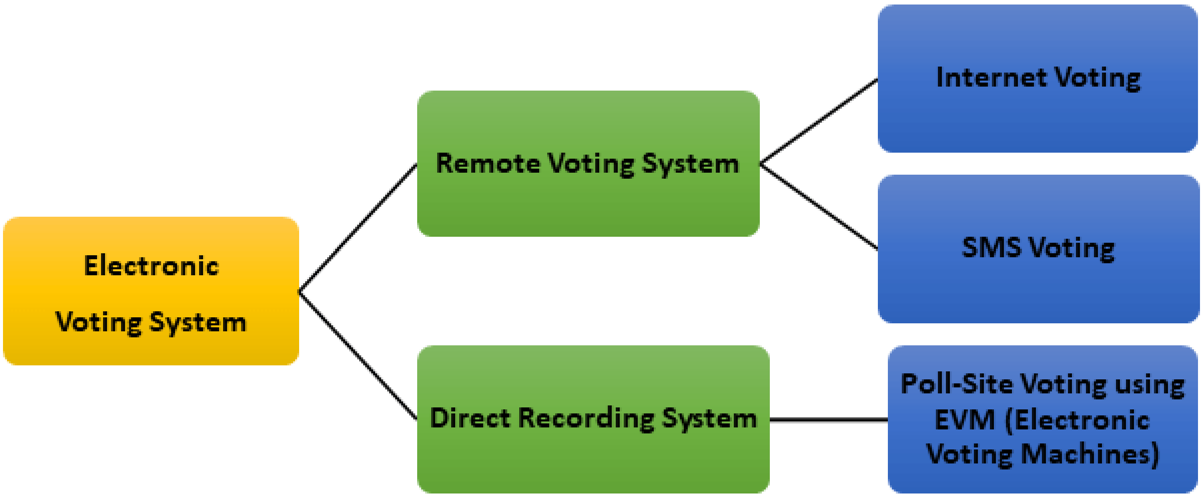
ADMINISTRATOR USE CASE DIAGRAM

E-Voting System Class Diagram



SYSTEM ARCHITECTURE





4. TECHNOLOGY DESCRIPTION

C++ is a high-level object-oriented programming language that helps programmers write fast, portable programs. C++ provides rich library support in the form of **Standard Template Library (STL)**.

C++ Language Features

Some of the interesting features of C++ are:

- ***Object-oriented***: C++ is an object-oriented programming language. This means that the focus is on "objects" and manipulations around these objects. Information about how these manipulations work is abstracted out from the consumer of the object.
- ***Rich library support***: Through C++ Standard Template Library (STL) many functions are available that help in quickly writing code. For instance, there are standard libraries for various containers like sets, maps, hash tables, etc.
- ***Speed***: C++ is the preferred choice when latency is a critical metric. The compilation, as well as the execution time of a C++ program, is much faster than most other general purpose programming languages.
- ***Compiled***: A C++ code has to be first compiled into low-level code and then executed, unlike interpreted programming languages where no compilation is needed.
- ***Pointer Support***: C++ also supports pointers which are widely used in programming and are often not available in several programming languages.

It is one of the most important **programming languages** because almost all the programs/systems that you use have some or the other part of the codebase that is written in C/C++. Be it Windows, be it the photo editing software, be it your favorite game, be it your web browser, C++ plays an integral role in almost all applications that we use.

Uses/Applications of C++ Language

After exploring C++ features, let's have look at some interesting areas where C++ is popularly used.

Operating Systems

Be it Microsoft Windows or Mac OSX or Linux - all of them are programmed in C++. C/C++ is the backbone of all the well-known operating systems owing to the fact that it is a strongly typed and a fast programming language which makes it an ideal choice for developing an operating system. Moreover, C is quite close to the assembly language which further helps in writing low- level operating system modules.

Browsers

The rendering engines of various web browsers are programmed in C++ simply because if the speed that it offers. The rendering engines require faster execution to make sure that users don't have to wait for the content to come up on the screen. As a result, such low-latency systems employ C++ as the programming language.

Libraries

Many high-level libraries use C++ as the core programming language. For instance, several Machine Learning libraries use C++ in the backend because of its speed. Tensorflow, one of the most widely used Machine Learning libraries uses C++ as the backend programming language. Such libraries required high-performance computations because they involve multiplications of huge matrices for the purpose of training Machine Learning models. As a result, performance becomes critical. C++ comes to the rescue in such libraries.

Graphics

All graphics applications require fast rendering and just like the case of web browsers, here also C++ helps in reducing the latency. Software that employ computer vision, digital image processing, high-end graphical processing - they all use C++ as the backend programming language. Even the popular games that are heavy on graphics use C++ as the primary programming language. The speed that C++ offers in such situations helps the developers in

expanding the target audience because an optimized application can run even on low-end devices that do not have high computation power available.

Banking Applications

One of the most popularly used core-banking system - Infosys Finacle uses C++ as one of the backend programming languages. Banking applications process millions of transactions on a daily basis and require high concurrency and low latency support. C++ automatically becomes the preferred choice in such applications owing to its speed and multithreading support that is made available through various Standard Template Libraries that come as a part of the C++ programming kit.

Cloud/Distributed Systems

Large organizations that develop cloud storage systems and other distributed systems also use C++ because it connects very well with the hardware and is compatible with a lot of machines. Cloud storage systems use scalable file-systems that work close to the hardware. C++ becomes a preferred choice in such situations because it is close to the hardware and also the multithreading libraries in C++ provide high concurrency and load tolerance which is very much needed in such scenarios.

Databases

Postgres and MySQL - two of the most widely used databases are written in C++ and C, the precursor to C++. These databases are used in almost all of the well-known applications that we all use in our day to day life - Quora, YouTube, etc.

Embedded Systems

Various embedded systems like medical machines, smartwatches, etc. use C++ as the primary programming language because of the fact that C++ is closer to the hardware level as compared to other high-level programming languages.

Telephone Switches

Because of the fact that it is one of the fastest programming languages, C++ is widely used in programming telephone switches, routers, and space probes.

Compilers

The compilers of various programming languages use C and C++ as the backend programming language. This is because of the fact that both C and C++ are relatively lower level languages and are closer to the hardware and therefore are the ideal choice for such compilation systems. These are a few uses and applications of C++ programming language. Now, let's know more about C++ advantages over other programming languages.

Advantages of C++ Language

C++ has the following 2 features that make it a preferred choice in most of the applications:

- ***Speed***: C++ is faster than most other programming languages and it provides excellent concurrency support. This makes it useful in those areas where performance is quite critical and the latency required is very low. Such requirements occur all the time in high- load servers such as web servers, application servers, database servers, etc. C++ plays a key role in such servers.
- ***Closer to hardware***: C++ is closer to hardware than most other programming languages like Python, etc. This makes it useful in those areas where the software is closely coupled with hardware and low-level support is required at the software level.

5. CODING

```
#include<stdio.h> // HEADER FILE FOR STANDARD I/0
#include<graphics.h> // HEADER FILE FOR GRAPHICS
MODE #include<dos.h>     // HEADER FILE FOR ENABLING
SOUND #include<conio.h> // HEADER FILE FOR CONSOLE
I/O #include<stdlib.h> // HEADER FILE FOR LIBRARY
FUNCTIONS union REGS i,o;
int initmouse();        // FUNCTION TO INITIALIZE MOUSE POINTER
void showmouseptr(); // FUNCTION TO SHOW POINTER
void restrictmouseptr(int,int,int,int); // FUNCTION TO RESTRICT
POINTER void getmousepos(int *,int *,int *); // TO GET POINTER
POSITION
void format();     // FUNCTION TO DRAW LAYOUT OF EVM
void graph();      // FUNCTION TO DISPLAY RESULT AS
GRAPH
void welcome();    // FUNCTION TO DISPLAY WELCOME MESSAGE
void boundry();
int vote1=0,vote2=0,vote3=0,vote4=0,vote5=0; // VARIABLES TO HOLD VOTES FOR
CANDIDATES
int
button,x,y;
void
main()
{      int gd=DETECT,gm;
       initgraph(&gd,&gm,"c:\\tc\\bgi"); // INITIALIZING GRAPHICS
       MODE randomize();
       boundry();
     welcome();               // CALLING WELCOME FUNCTION
     cleardevice();           // CLEARING THE SCREEN
     format();                // CALLING FORMAT
     FUNCTION
     showmouseptr();
     restrictmouseptr(0,0,675,435);    // RESTRICTING MOUSE POINTER WITHIN SCREEN
```

```
do
{
```

```
getmousepos(&button,&x,&y);
 if((button&1)==1&&x>475&&x<580&&y>250&&y<280)
{ break;}
else if((button&1)==1&&x>280&&x<380&&y>105&&y<125)
{ setcolor(RED);circle(270,115,5);
   sound(1200);
   delay(500);
   nosound();
   setcolor(BLACK);circle(270,115,5);
   vote1++; }
else if((button&1)==1&&x>280&&x<380&&y>155&&y<175)
{ setcolor(RED);circle(270,165,5);
   sound(1200);
   delay(500);
   nosound();
   setcolor(BLACK);circle(270,165,5);
   vote2++; }
else if((button&1)==1&&x>280&&x<380&&y>205&&y<225)
{ setcolor(RED);circle(270,215,5);
   sound(1200);
   delay(500);
   nosound();
   setcolor(BLACK);circle(270,215,5);
   vote3++; }
else if((button&1)==1&&x>280&&x<380&&y>255&&y<275)
{ setcolor(RED);circle(270,265,5);
   sound(1200);
   delay(500);
   nosound();
   setcolor(BLACK);circle(270,265,5);
   vote4++; }
```

```
else if((button&1)==1&&x>280&&x<380&&y>305&&y<325)
{ setcolor(RED);circle(270,315,5);
   sound(1200);
   delay(500);
   nosound();
   setcolor(BLACK);circle(270,315,5);
   vote5++; }
}               // END OF DO
while(1);

 cleardevice()
 ; initmouse();
 showmousep
 tr();
 boundry();
 graph();
 getch();
}                   // END OF MAIN FUNCTION
void boundry()
{
   setcolor(1+random(14));
   rectangle(0,0,635,475);
   setcolor(1+random(14));
   rectangle(3,3,632,472);
}
void welcome()
{ randomize();
  settextstyle(8,0,4);
  setcolor(1+random(1
  4));
  outtextxy(200,100,"WELCOM
  E"); delay(800);
  setcolor(1+random(14));
  outtextxy(250,160,"T
  O"); delay(800);
  setcolor(1+random(14
  ));
  outtextxy(50,220,"ELECTRONIC VOTING SYSTEM");
  delay(800);
  while(!kbhit())
  {  setcolor(1+random(14));
     outtextxy(50,400,"Press any key to continue.... ");
     delay(500);
     setcolor(BLACK);
     outtextxy(50,400,"Press any key to continue.... ");
     delay(500);
  }


}

void format()
{ setcolor(6);
    rectangle( 90,30,400,380);
    rectangle(
    87,27,403,383);
    settextstyle(0,0,5);
    outtextxy(140,40,"E V
    M"); line(90,80,400,80);
    settextstyle(8,0,2);
    outtextxy(100,100,"MEGHRA
    J");
    rectangle(95,100,250,130);
    arc(290,115,90,270,10);
    arc(370,115,270,90,
    10);
    line(290,105,370,10
    5);
```

```
line(290,125,370,12
5);
```

```
outtextxy(100,150,"DINESH");
rectangle(95,150,250,180);
arc(290,165,90,270,10);
arc(370,165,270,90,10);
line(290,155,370,155);
line(290,175,370,175);

outtextxy(100,200,"RAKESH");
rectangle(95,200,250,230);
arc(290,215,90,270,10);
arc(370,215,270,90,10);
line(290,205,370,205);
line(290,225,370,225);
outtextxy(100,250,"DEEPAK");
rectangle(95,250,250,280);
arc(290,265,90,270,10);
arc(370,265,270,90,10);
line(290,255,370,255);
line(290,275,370,275);

outtextxy(100,300,"ANUJAY");
rectangle(95,300,250,330);
arc(290,315,90,270,10);
arc(370,315,270,90,10);
line(290,305,370,305);
line(290,325,370,325);
```

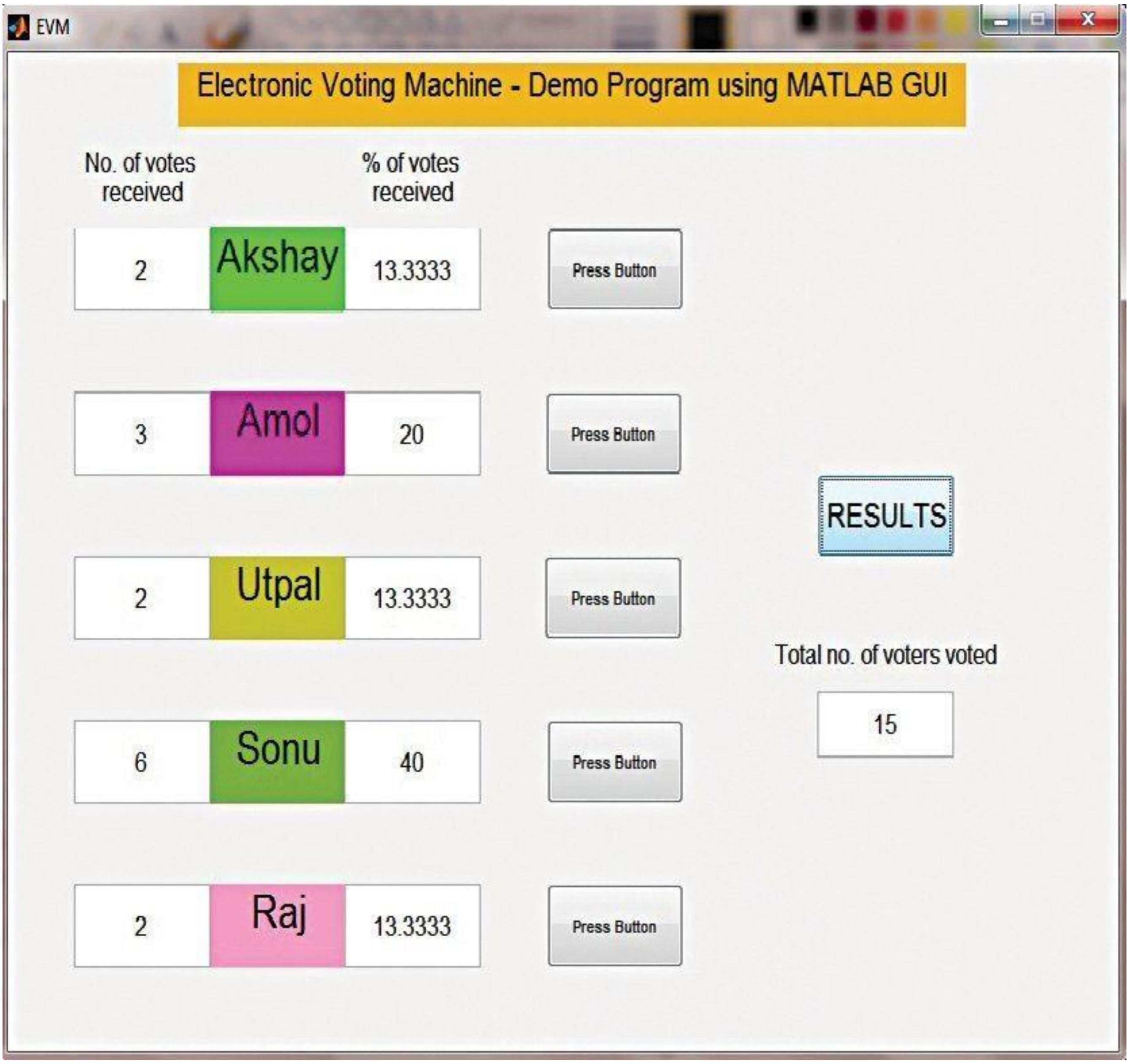
```
rectangle(475,250,580,280);
outtextxy(480,250,"RESULTS");
```

```
    outtextxy(50,400,"Presented By:- EFY Enterprises Pvt Ltd");
}
void showmouseptr()
{
i.x.ax=1;
int86(0x33,&i,
&o);
}
void restrictmouseptr(int x1, int y1, int x2, int y2)
{    i.x.ax=7;
     i.x.cx=x1;
     i.x.dx=x2;
     int86(0x33,&i,
     &o); i.x.ax=8;
     i.x.cx=y1;
     i.x.dx=y2;
     int86(0x33,&i,
     &o);
}
void getmousepos(int *button, int *x, int *y)
{ i.x.ax=3;
    int86(0x33,&i,&o);
    *button=o.x.bx;
    *x=o.x.cx;
    *y=o.x.dx;
}
void graph()
{ outtextxy(200,100,"RESULTS(in % votes)");
    int
    candidate1=((vote1*100)/(vote1+vote2+vote3+vote4+vote5))
    ;                                                        int
    candidate2=((vote2*100)/(vote1+vote2+vote3+vote4+vote5))
    ;                                                        int
    candidate3=((vote3*100)/(vote1+vote2+vote3+vote4+vote5))
    ;                                                        int
```

```
candidate4=((vote4*100)/(vote1+vote2+vote3+vote4+vote5))
;                                                       int
candidate5=((vote5*100)/(vote1+vote2+vote3+vote4+vote5))
;
```

```
   setcolor(2);
   rectangle(100,300,130,300-
   candidate1);outtextxy(100,300,"ME");
   rectangle(200,300,230,300-
   candidate2);outtextxy(200,300,"DI");
   rectangle(300,300,330,300-
   candidate3);outtextxy(300,300,"RA");
   rectangle(400,300,430,300-
   candidate4);outtextxy(400,300,"DE");
   rectangle(500,300,530,300-
   candidate5);outtextxy(500,300,"AN");

   setcolor(1+random(14));
   rectangle(545,400,600,4
   30);
   outtextxy(550,400,"EXI
   T");

   do
{
getmousepos(&button,&x,&y);
 if((button&1)==1&&x>545&&x<600&&y>400&&y<430)
{ break;}
}            // END OF DO
while(1);



}
initmouse()
{
         ax=0;
         int86(0x
         33,&i,&
         o);
         return(o
        .x.ax);
}
```

6. SCREENSHOTS

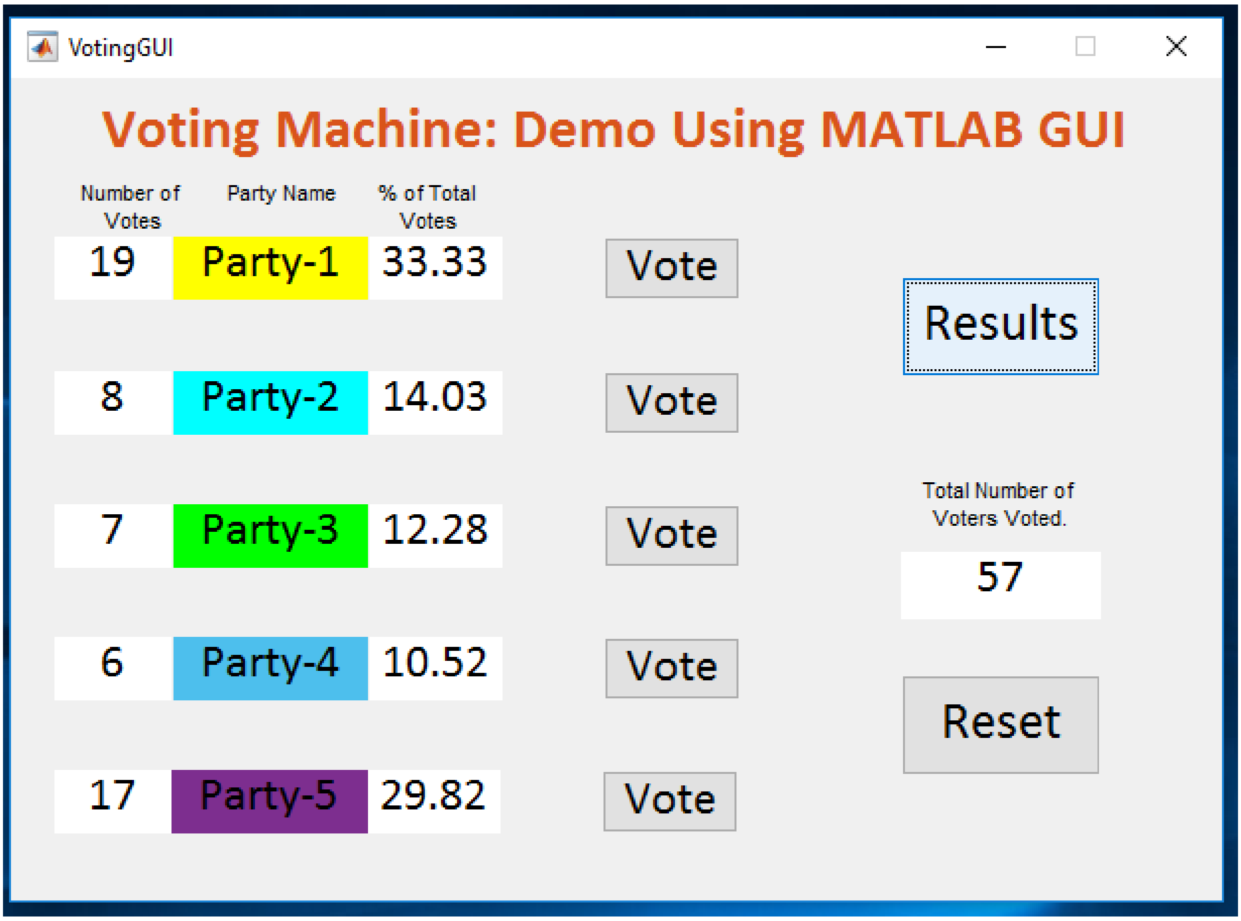


Electronic Voting Machine - Demo Program using MATLAB GUI

No. of votes received		% of votes received
2	Akshay	13.3333
3	Amol	20
2	Utpal	13.3333
6	Sonu	40
2	Raj	13.3333

Total no. of voters voted

15

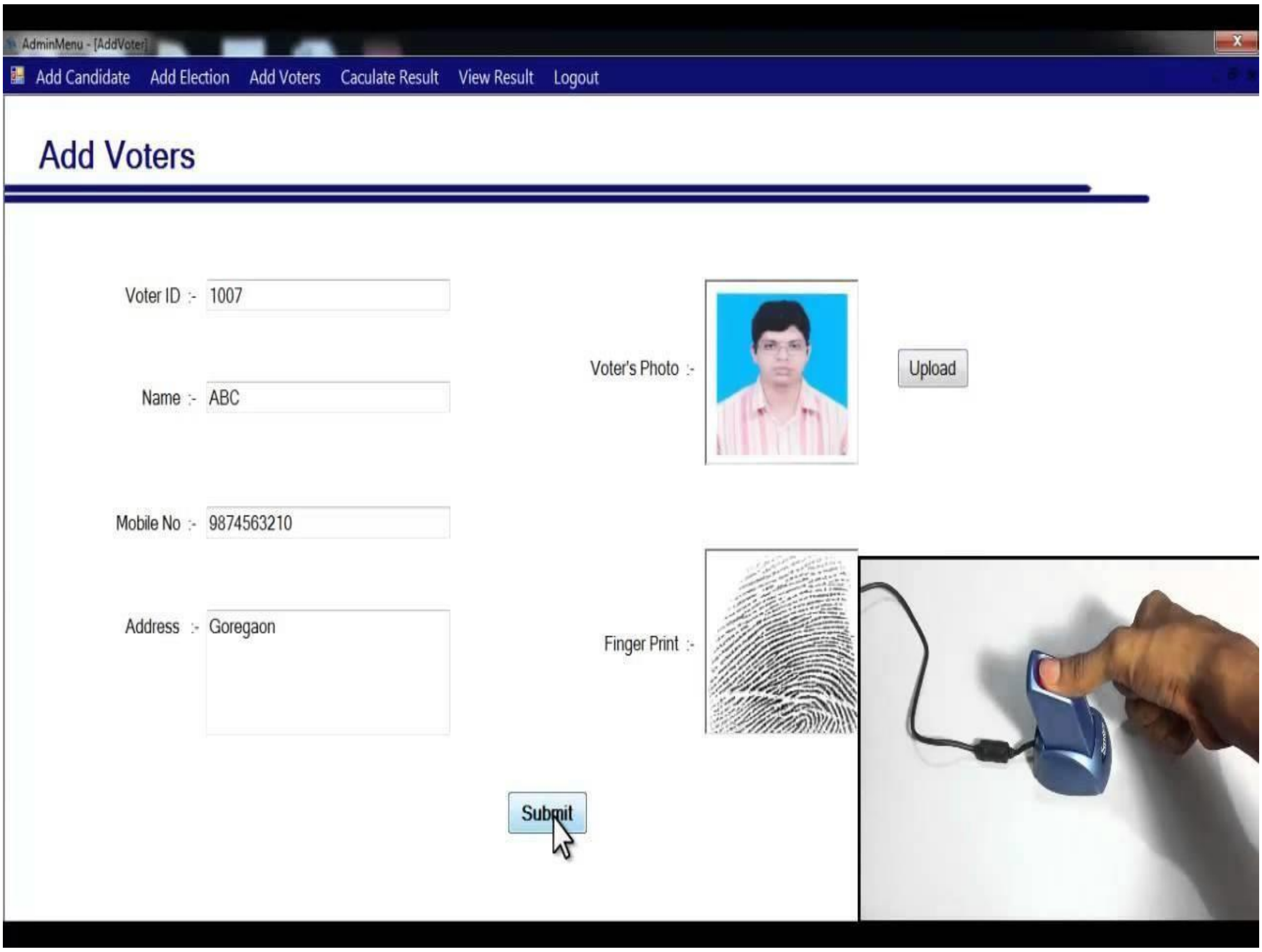


Voting Machine: Demo Using MATLAB GUI

Number of Votes	Party Name	% of Total Votes
19	Party-1	33.33
8	Party-2	14.03
7	Party-3	12.28
6	Party-4	10.52
17	Party-5	29.82

Total Number of Voters Voted.

57



Add Voters

Voter ID :- 1007

Name :- ABC

Mobile No :- 9874563210

Address :- Goregaon

Voter's Photo :-

Finger Print :-

7. CONCLUSION

The function of compiling, editing and processing data makes the voting system complicated and tedious process. Electronic voting system is an effort in the direction to use computer for the purpose of voting system. It is both user friendly as well as time saving. This is a foundation level. Emphasis has been given in this application to replicate all the process required in a traditional system. EVM system is self-sufficient with all the information required about the voters, candidate and the voters. Once the information is fed to the system, it can identify individual voters, their votes and then can press any key to return such as counting the votes has been simplified in this system. The basic function of the polling has been reduced to single key operation. A single polling manager can manage large number of voters.

LIMITATIONS:

An effort had been done to develop the system with the wide scope in view. Regardless of the number of application the software has its own limitation. The following are some of the notable limitation of the electronic voting system.

1. Security: - The system administrators can hacks the system at his will. He can change data output and hence can affect the result.
2. Voters education level: - Voters should be at least that much educated to enter the vote, i.e. press the required button. A totally computer illiterate has a very high chance of polling a wrong vote i.e. in favors of an undesirable candidate.
3. Specific voting system: - Since the system has been developed to handle the specific type of vote, i.e. only one candidate has to be chosen from the given list by one vote. This system cannot be used for other specific type of voting for e.g. electoral college voting (Indian resident Election) and conditional voting.
4. Basic system: - The EVM system is itself a basic and elementary form of software and hence cannot support other input devices as separate keyboard etc.

5. Confidentiality: - If the requirement of the voting is to keep the confidentiality of the votes, it is not possible since the system administrator would always be in a position to overview the votes of each and every individual at the time of voting.
6. The screen: - The screen is not too good in clarity and there are chances of mistakes during voting.

POSSIBLE IMPROVEMENTS:

1. In case of large data storage, the storing of data e.g. individual voter details would take a long time. Hence improvement applications can minimize the time.

2. Specific input devices (e.g. voting ads) could be an added advantage to the system.

3. Candidate name and number are only possible options in this application. Individual candidate terms can be added to the application.

4. Improvement can be done to make the application workable on net so that voting can be possible.

SCOPE:

1. Consumer survey
2. Large scale voting
3. Distance voting
4. Highly secured and error free result

BIBLIOGRAPHY

Text Books:

1. TURBO C++ by Robert Lafore
2. C++ Programming by E. BALAGURUSAMI
3. C++ by YASWANT KANITKAR

References:

www.joblagao.com

www.electronicsforu.com

1. INTRODUCTION

The function of compiling, editing and processing data makes the voting system complicated and tedious process. Electronic voting system is an effort in the direction to use computer for the purpose of voting system. It is both user friendly as well as time saving.

This is a foundation level. Emphasis has been given in this application to replicate all the processes required in a traditional system. EVM system is a self-sufficient with all the information required about the voters, candidate and votes. Once the information is fed to the system it can identify individual voters, their votes and the candidates. Tedious function such as counting the votes has been simplified in this system. The basic function of the polling has been reduced to single operation. A single polling manager can manage large number of voters.

The program code has been Written in ‘C++’ language has been chosen because of its procedural nature and the flexibility it provides in interacting with hardware and operating system efficiently.

NEED TO COMPUTARISE THE SYSTEM

Information is needed in the organization for planning, staffing and controlling purposes. Regardless of the nature of the information required, the information should process the characteristics of accuracy, timeliness and relevancy. In the recent years need for information improvement by reports lacking one or more of these characteristics and by increased paper work volume, rising casts and pressures from outside changes.

Fortunately computers thrive on repetitive large volume processing tasks, are fast and accurate. The processing capability in many organizations has been stained by:

4. Growth in size and complexity of the organization.
5. The increased requirements for data from external sources.
6. The demand of administrators for more information.

More than a million new pages of data are generated each minute of the day in offices . Compare to other processing method, the use of computers may make it possible for certain administrative costs to be reduced while the level of processing activity remains stable. The increased cost and clerical labor materials and other expenses associated with the data processing operation require eventual managerial attention.

We all agree that meaningful information is timely information. But with an increase in volume and size of an organization, there is only a reduction in the speed of processing. Rapid changes are taking place in the world socially, economically and technically. Such changes have a significant impact on the environment which organization must operate on the planning that managers must do and on the information that must have. Many educational administrators are responsible for supervising the activities of a large number of schools scattered over a district. They must have accurate information if they are to control such an effort properly. But if a data processing operation is stained to or beyond the capacity for which it was originally planned, inaccuracies will begin to appear. Inadequate control will permit inadequate performance. Thus the administrator will logically demand better quality in the information he receives. It is due to these reassures (increased paper work volume, cost, pressure from outside changes demand for timeliness and demand for quality) that most if the organization today is opting for computers to do data processing for them.

Computers are most efficient used in processing operation that includes:-

Large volume of input:- Greater the volume of data that must be processed to produce the needed information, the more economical computer processing becomes relative to other possible methods.

Repetition of projects: - Because of the expenses involved in preparing a task for computer processing, it is frequently more economical to use the computer for repetitive purpose.

Desired and necessary greater speed in processing:- Greater the need for timely information, greater will be quite accurate if the task to be performed has been properly prepared.

THE VOTING SYSTEM:

The voting system can be described as "A process in which the input (voter) votes in favors of a given decision (candidate)". Votes are procured, counted and a statically result is provided.

Traditionally there are two basic types of voting system:-

Open voting system: - In this system the voter votes in favors of a decision without hiding his identity, example of system is rising of hands. This is in itself a very elementary system and useful when the level of decision to be taken is comparatively less effective to the voter. The number of voters should be less and countable and discipline and honestly is maintained by the voters. The voting system is people oriented.

Second voting system: - In this system identity of the voter is hidden and once the vote has been given it is not possible identifies a given vote with the voter. In this system paper work is usually done, as the identity of the voter has to be concealed. A very large number of voters can take part in such system. The whole process in itself system oriented and have security secrecy and counting becomes easier.

Computer Voting System

Manual voting system at large scale becomes a tiring and costly affair in terms of money, manpower and time. The use of computer can easily reduce the effort and make voting an easy as well as voter and candidate friendly affairs. It can reduce the amount of money spent, manpower and time to a very large extent.

In the present system of voting the individual votes are registered in form of ballot papers. The names of candidates is written or printed in the ballot papers in serial orders, which can be easily recognized by the voters. A separate list of voters is made so that the voters can be allowed to sequentially and chances of non-voters or revolting of an individual can be controlled. The whole system is done under the control of a residing officer. Each voter is given a ballot paper on which he stamps his individual choice. The ballot paper is then folded in a particular way and then insert in the ballot box. As soon as an individual voter casts his vote, his name is deleted from the voter list.

At the end of polling the ballot boxes are sealed and sent for counting. The counting of the ballot papers is done in a redefined method. The results are declared after getting convinced that the counting has been done properly. The process of counting takes time since it is done manually.

Problems in Existing System:

The present system has been marked by many problems some of which have been enumerated below: -

Expensive: - The present form of voting is an expensive affair. If needs a lot of preparation in terms of time and money. The cost of paper work and stationary is one of the most expensive parts. Expenses are incurred in processing ballot boxes and ballot papers. If in case ballot papers are printed wrong re-printing has to be done.

Time consuming - A lot of time is consumed in counting of votes. The manual counting is in itself a tedious job and hence requires time.

Invalid votes: - Manual voting has a regular problem of invalid votes. A number of votes become invalid due to a number of unintentional reasons.

Manipulation work: - Voting process at times is hindered by presence of external elements to favor a given candidate. Examples are booth capturing.

Manpower required: - The present system of voting involves the use of lots of manpower at each and every stage of voting process.

User identification: - This poses one of the serious problems of current voting system. During voting process many people cast there vote more than one times. This causes problems as wrong people are selected.

Manipulation in counting: - As counting process is done manually it may happen that votes are wrongly counted or they may be manipulated in some way in favors of some candidates.

Portability factor: - After polling is over ballot boxes are usually taken to some other places for counting. This is risk as the ballot boxes may be stolen or there may be some mishap reroute to counting system.

Software Program of the EVM:

Despite design features that make the election software complex to extract the information from the control unit processor, a criminal has a variety of options to steal the information from it. These include decapsulating the chip and examining it under a microscope. Here we did not attempt to extract the software using these methods; once the software is extracted, it is straightforward to reverse- engineer it using standard disassembly tools.

The software (evm.cpp) developed here for the demo is tested on Turbo C++ Version 3.0 IDE. When the user runs the program, it displays a real-time voting machine on the screen with the list of the candidates and corresponding buttons to vote for them (refer Fig. 2). The program requires ‘C’ graphics files such as egavga.bgi, goth.chr, lcom.chr and trip.chr for proper display.

In order to vote for a party, the user has to click the respective button using mouse. A beep sounds to confirm successful submission of vote.

At any point of time if you want to check the status of the votes being polled, just click ‘result’ button in the control panel (not shown here). The next screen will display the percentage of votes received by different candidates in the form of a bar chart. To exit the program, click ‘exit’ button.

2. SYSTEM REQUIREMENT SPECIFICATION(SRS)

SYSTEM REQUIREMENTS:

The aim of the system to be developed ------- Electronic-voting system is to develop software, which should automate the process of voting of an electoral system. The system is supposed to be used as a subsystem in a layer electoral system, which could be a manual system or a computerized one. Therefore the proposed system must be able to function under both circumstances.

The basic functionality that the system must provide is

8. The system must provide for manipulation of candidate data.

The manipulation of candidate data consists of: -

Adding a new voter record in the existing voter list. The system must provide enough mechanisms to check the data validity and integrity.

Deletion of existing voters from the voter list. The system must be able to notice and notify the deletion request for a non-existent record. The system should have a mechanism of asserting record deletion to avoid accidental deletion of record.

Modification of voter records in the voter list. The system must be able to modify details of a voter for any specific data field, or for all the fields. The system must provide for validation in this module also. The system should have a mechanism of asserting record modification to avoid accidental modification of records.

9. The system must provide for manipulation of candidate data. The manipulation of candidate data consists of: -

Adding a new candidate record in the existing list. The system must provide enough mechanisms to check the data for validity and integrity.

Deletion of existing candidates from the candidate list. The system must be able to notice and notify the deletion request for anon-existent record. The system should have a mechanism of asserting record deletion to avoid accidental deletion of records.

Modification of candidate records in the candidate list. The system must be able to modify details of a candidate for any specific data field, or for all the fields. The system must provide for validation in this module also. The system should have a mechanism of asserting records modification to avoid accidental modification of records.

10. The system must have a rich set of utilities for listing the various details of the candidates as well as voters. The listing should be both a generalized listing for all candidates or voters. And detailed listing of an individual candidate or voter.

11. The system must provide a method and interference for the voting process. In which a voter can select a candidate from the list of candidates. There should further be a method to display the details of the winner of a particular voting process. The details of the margin of the victory should also be displayed.

12. The system must provide interference to print the details of candidates and voters. In both a summarized format and a detailed individual report.

13. The system must provide help whenever necessary. And should also provide tips, validity criteria etc… while data manipulation.

14. The system must have easy way of switching in between different modules.

NON-FUNCTIONAL REQUIREMENTS:

Nonfunctional requirements are just as important to your business analysis as the functional requirements when it comes to defining the look and feel of the solution. Nonfunctional requirements are a challenge because different people interpret them differently from organization to organization (or even from department to department in the organization). You need to understand a lot about the people using the solution and make sure your nonfunctionals document its performance.

You create the nonfunctional requirements based on your elicitations from the users, who they are, and what their expectations of the system performance are. When you create nonfunctional requirements, you need to think about things like the following:

Performance: How well does the system perform? To understand the performance requirements, ask stakeholders questions such as “What are the number of concurrent users?”, “What are the system or query response times?”, and “What is the system’s capacity in terms of memory, disk space, and data volumes?”

Security: Who has access to the system, and how much access do they have? To understand the security requirements, ask questions such as “Which users are authorized to perform which functions?”, “What is the privacy of the information being captured and stored?”, and “What features need to be in place to log user access and authenticate users?”

Reliability: *Reliability* is how the system operates based on the expectation of the end-user. Think about buying a car. You probably *purchase* a car because of the *functionality* (0–60 mph in 8 seconds, A/C, satellite radio, and so on), but you probably think about going to shop for that new car because of the *reliability* of the car. Similarly, you want to make sure you find out how consistently the business wants the solution to perform and what maintenance and support you need to make sure it stays that way. To elicit the reliability requirements, ask questions such as “When is the system expected to be available?”, “What downtime does the system have for the administrators to perform maintenance, and when is the best time to schedule downtime?”, and

“What notification do the users need when the system is going down for maintenance? How much advance notice should they receive?”

Compatibility: *Compatibility* refers to the extent to which the solution plays nice with other applications. To elicit compatibility nonfunctional requirements, ask questions such as “What common standards, common technology, and protocols exist on the workstation?”; “How well does the solution work with the common build?”; “What kinds of data exchange do you envision?”; and “What information (data) must be exchanged with other systems?”

Maintainability: *Maintainability* deals with how easy the system is to maintain and repair. To elicit the nonfunctionals for maintainability, ask questions such as “What is the ability to change one component without affecting others?”, “What effects do the maintenance activities have on customers, users, and employees?”, and “Who performs system upgrades? Who is responsible for interfaces?”

Transferability: *Transferability* refers to the ease with which a system can be transferred to a different hardware or software environment. Some of these concerns are lessening now that many companies are creating browser-based applications, yet these concerns have expanded with the mobile apps (like those you see on your smartphone) and the different versions and standards for e-readers.

Usability: *Usability* concerns the ways by which the user is able to learn, operate, and interpret the system results. This category includes ease of entry, learning, and handling, as well as the system’s intuitiveness.

Metrics and measurements: With any nonfunctional requirement, you must understand what measurement criteria you’ll use to determine whether the requirement is successful and met. You’re defining how well the solution meets the requirements. To elicit the metric, ask questions like “What are some aspects surrounding that requirement that you can measure?” and “What are the acceptable measurement time frames that are acceptable for the stakeholder?”

SYSTEM SPECIFICATION:

MINIMUM HARDWATRE

REQUIREMENTS PROCESSOR:

DUAL CORE OR

ABOVE RAM: 2GB OR ABOVE

HARD DISK: 256GB OR ABOVE

PRINTER: LINE_PRINTER (DOT

MATRIX/INKJET) KEYBOARD:NORMAL OR ABOVE

APPLICATION SOFTWARE

C++ S/W BORLAND C++/ TURBO

C++ OPERATING SYSTEM:

WINDOWS 98 S/W

LANGUAGE: C++

3. SYSTEM DESIGN

DESIGN SPECIFICATION:

To implement the requirements as specified in the system requirement specification document of Electronic Voting System. The proposed system is to be developed using "c++" language. The system has a main module, which will have several other modules as depicted in the structure diagram of the system.

The functionally requirements will be fulfill as: -To provide for manipulation of voter data

A new voter record is added in the existing voter list, by using add-vote () Module. This module will use the voter structure to input voter details. Each field in the voter structure will be checked for the integrity constraints. As specified in SRS, the details are added into existing voter detail file voter.dat. This file ensures the secondary storage of data.

Deletion of existing voters from the voter list is done by the module delete-voter (). The system checks for the existing of the record to be delete in the file voter.bat. For nonexistent data it notifies the user .The module also record deletions by flashing a message.

Modification of voter records in the voter list is handled by a module modify-voter (). This module reads the voter No from the user searches has its existence in the file voter.dat. If the record is found the details are displayed where typing new values can modify them. The module also assents record modification by flashing a message.

2. To provide for manipulation of candidate data.

A new candidate record is added in the existing candidate list, by using add-candidate () module. This module will use the candidate structure to input candidate details. Each field in the candidate structure will be checked for the integrity constraints as specified in SRS. The details are added into exiting voter-details file voter.dat. The file is ensures the secondary storage of the data.

Deletion of existing candidates from the voter list is done by the module del

ete- candidate (). The system checks for the existence of the record to be deleted in the file candidate.dat. For non-existent data it notifies the user. The module also asserts record deletion by flashing a message.

Modification of voter records in the voter list is handled by a module modify-candidate (). These module reeds the voter numbers from the user searches its existence in the file candidate.dat. If the record is found the details are displayed where typing new values can modify them. The module also asserts record modification by flashing a message.

8. The listing of the various details of the candidates as well as voters. Handled by four different module

dislay-code (),

showlist-voter

(), display-

candidate(),

showlist-candidate () modules give detailed listing of an individual candidate or voter showlist-voter ().

Showlist-candidate () gives generalized listing for all candidates of voter.

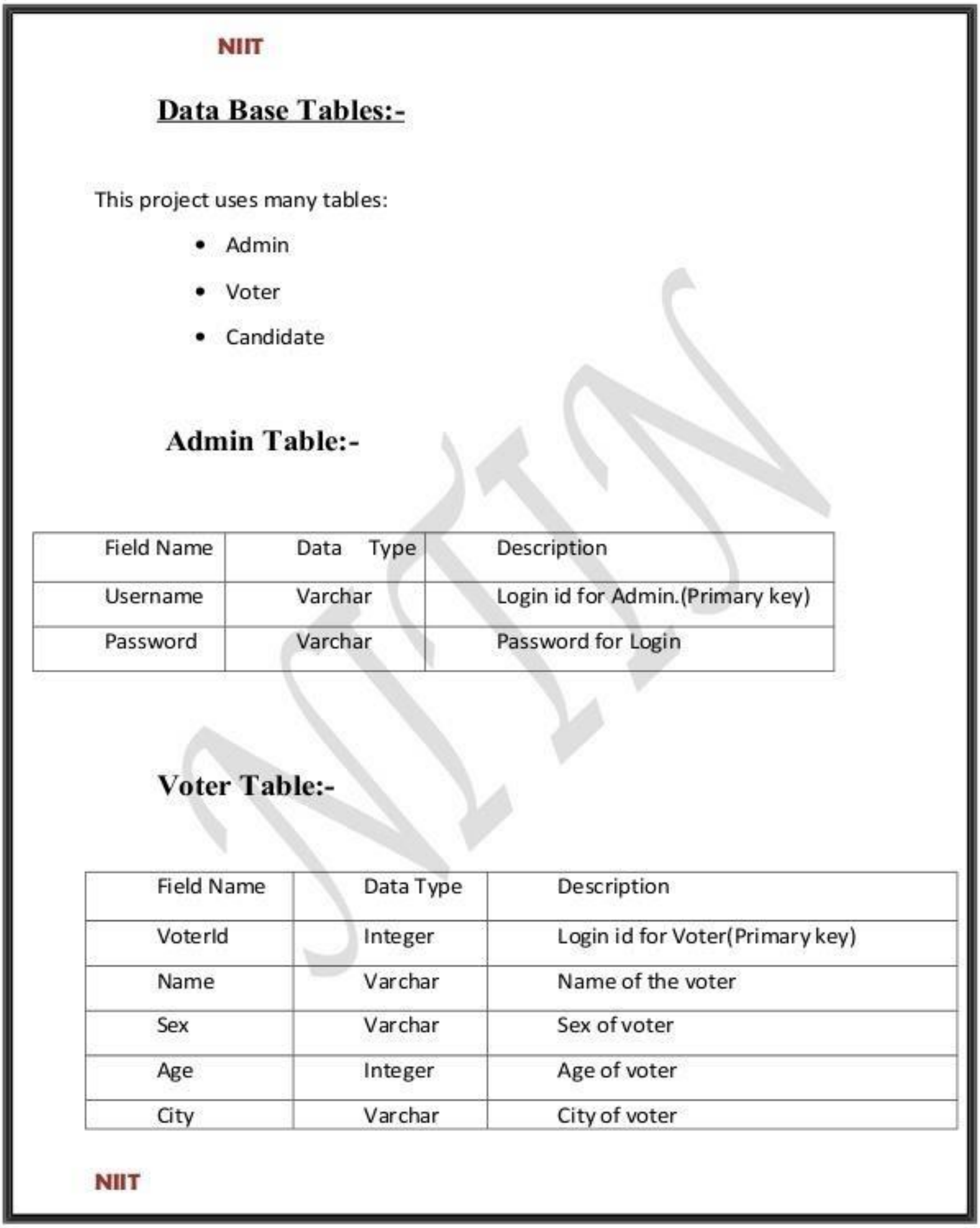
9. The voting process is handled by the module vote (), fmain (), find (), showlist (). A voter can select a candidate from the list of candidates, this functionality is provided by module vote (). The method to display the details of the winner of a particular voting process is handled by find () and showlist (). The detail of the margin of the victory is dislayed by showlist().

10. The system provides interface to print the details of candidates and voters by using module print (), which is called from different modules with different values to print different values to print different data as per the specification of SRS.

11. The system provides help by a module help (). And provides tips, validity criteria etc, by displaying message at the screen using a module statusbar().

DATA DICTIONARY

FUNCTION	DESCRIPTION
10. ADD_VOTER()	This function adds voters name to the voter list
11. VOTER_INFORMATION ()	This function gives information about the voters, name, address etc by putting its voter number.
12. VOTER_LIST ()	This function shows list of the voters
13. ISSUE-VOTING_CARD ()	This function ISSUES voting cards to the voters for validating the election.
14. RESET_STATUS()	This function resets the date to get it filled again f illed.
(A) FOR CANDIDATE	To reset votes received by candidate to zero.
(B) FOR VOTER	To reset all the status in the voter file.
15. DELETE_VOTER () .	To give the voter name to delete voter record.
16. ELECTION ()	This function is used to give votes by the voters.
17. VOTE_CALCULATION ()	To calculate votes given by the voters.
18. ADD_CANDIDATE()	This function adds the candidate record to the Candidate file.
12 CANDIDATE_INFORMATION ()	This function gives the information of the candidates.
13 DELETE_CANDIDATE ()	To give candidate serial number to delete recor
19. VOTING_LIST ()	To display of the votes received by the candidates.
20. ADD_VOTES	To add votes in the favors of the given candidate.
21. VOTING_DATE ()	To display the date of voting.
22. SET_DATE ()	To set the date for the new election.
23. VALID_DATE ()	To return the validity of today’s date.
24. RETURN_DATE ()	To return the election date.
25. LINE_HOR ()	To draw horizontal line.



Data Base Tables:-

This project uses many tables:

- Admin
- Voter
- Candidate

Admin Table:-

Field Name	Data Type	Description
Username	Varchar	Login id for Admin.(Primary key)
Password	Varchar	Password for Login

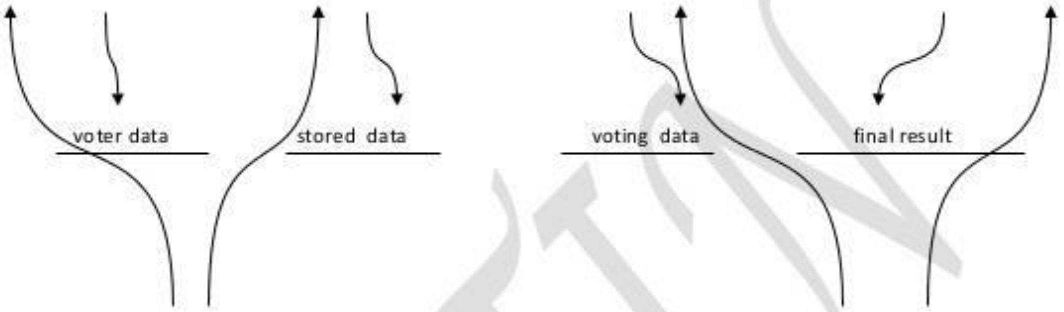
Voter Table:-

Field Name	Data Type	Description
VoterId	Integer	Login id for Voter(Primary key)
Name	Varchar	Name of the voter
Sex	Varchar	Sex of voter
Age	Integer	Age of voter
City	Varchar	City of voter

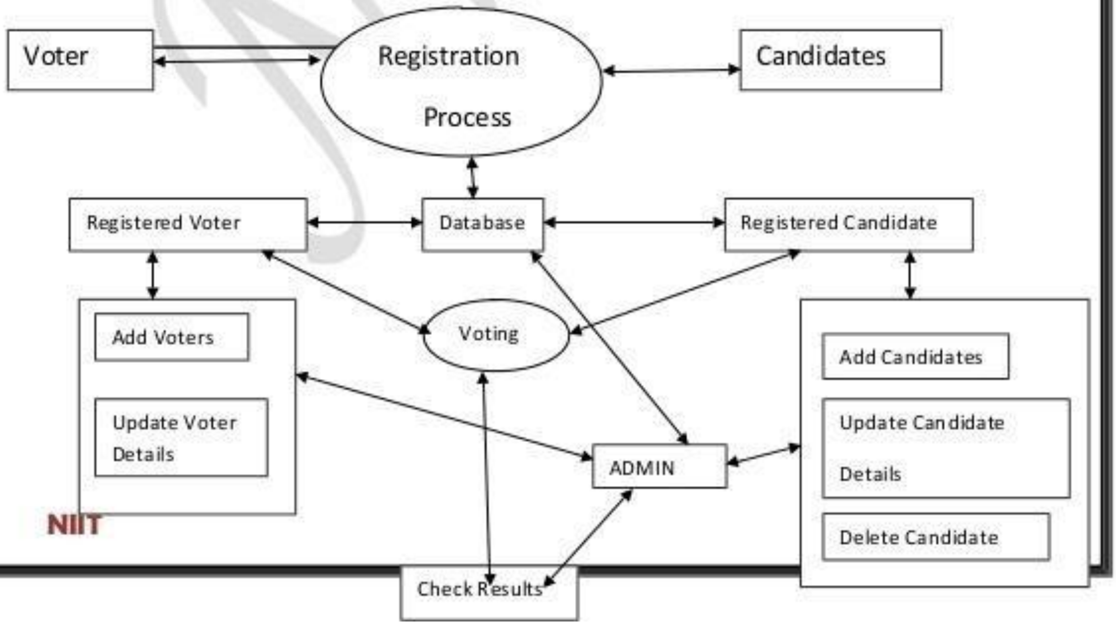
Security	Varchar	Security Question
Status	Boolean	Status of voter(he/she can vote or not)

Candidate Table:-

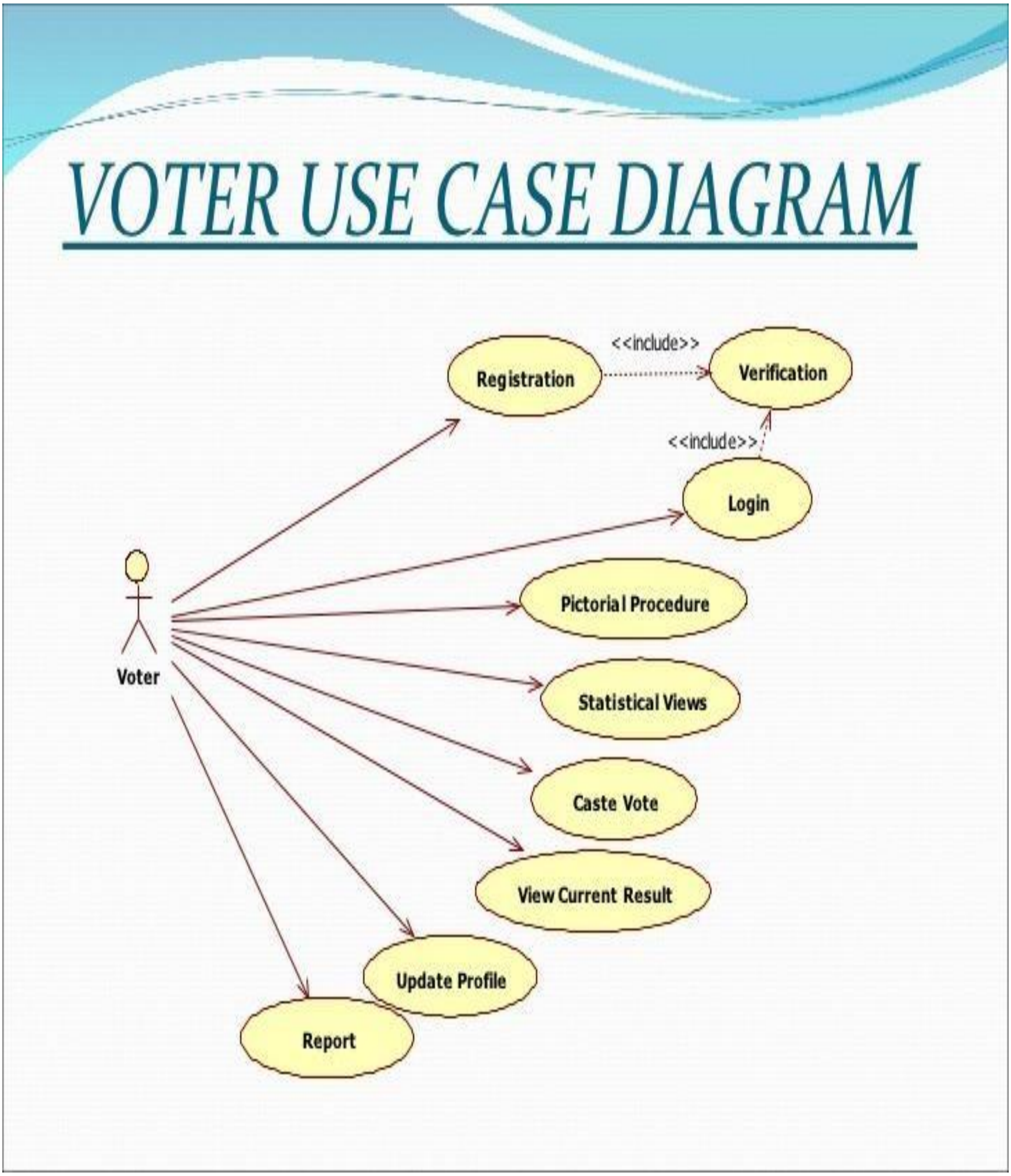
Field Name	Data Type	Description
Symbol	Varchar	Party Symbol (Primary key)
Name	Varchar	Name of the voter
Sex	Varchar	Sex of voter
Age	Integer	Age of voter
City	Varchar	City of voter
Count	Integer	Count the no of votes

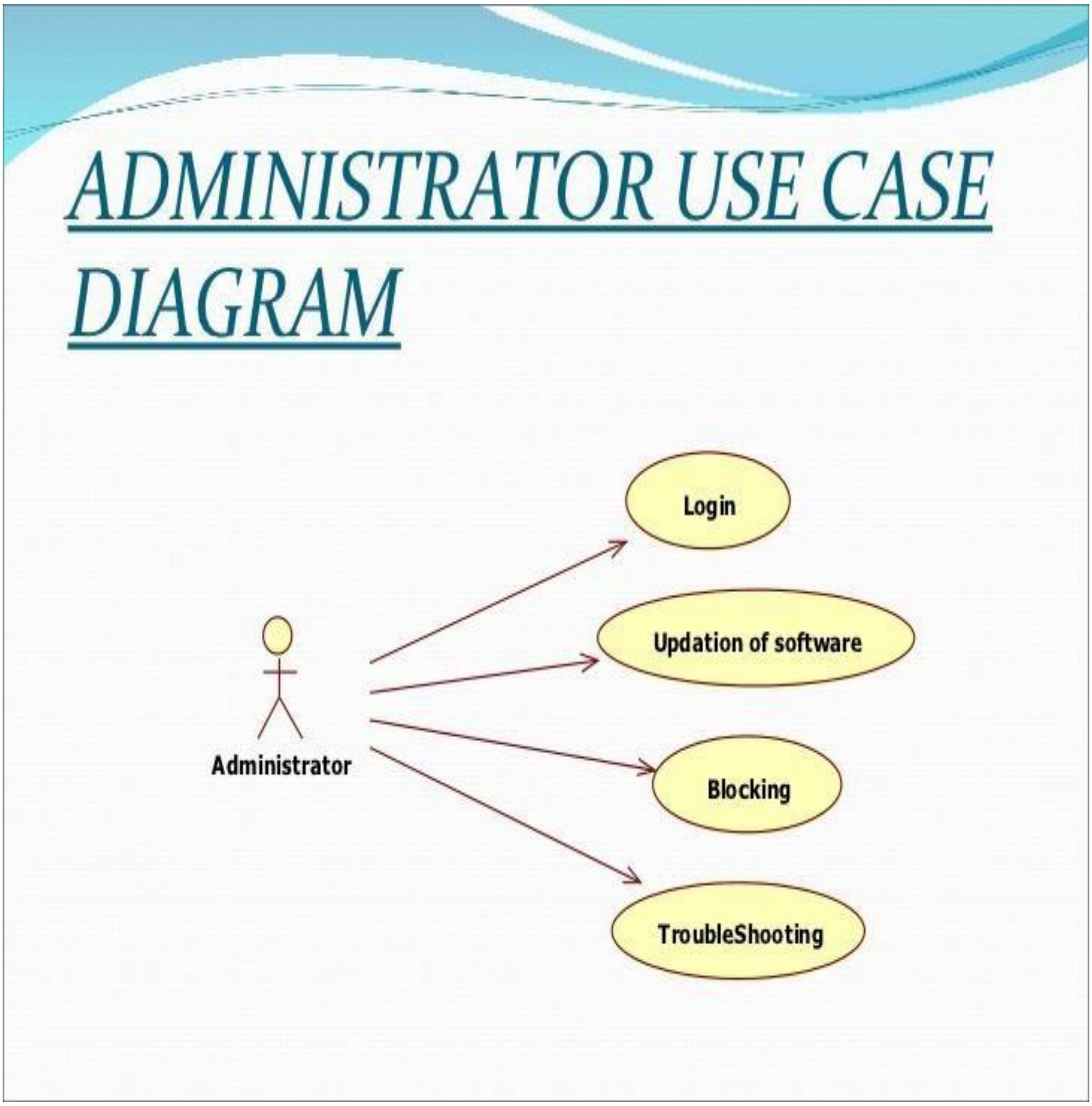

voter data
stored data
voting data
final result

ER Diagram:


Voter
Registration Process
Candidates
Registered Voter
Database
Registered Candidate
Voting
Add Voters
Update Voter Details
Add Candidates
Update Candidate Details
Delete Candidate
ADMIN
Check Results

Use case Diagram





ADMINISTRATOR USE CASE DIAGRAM

Administrator

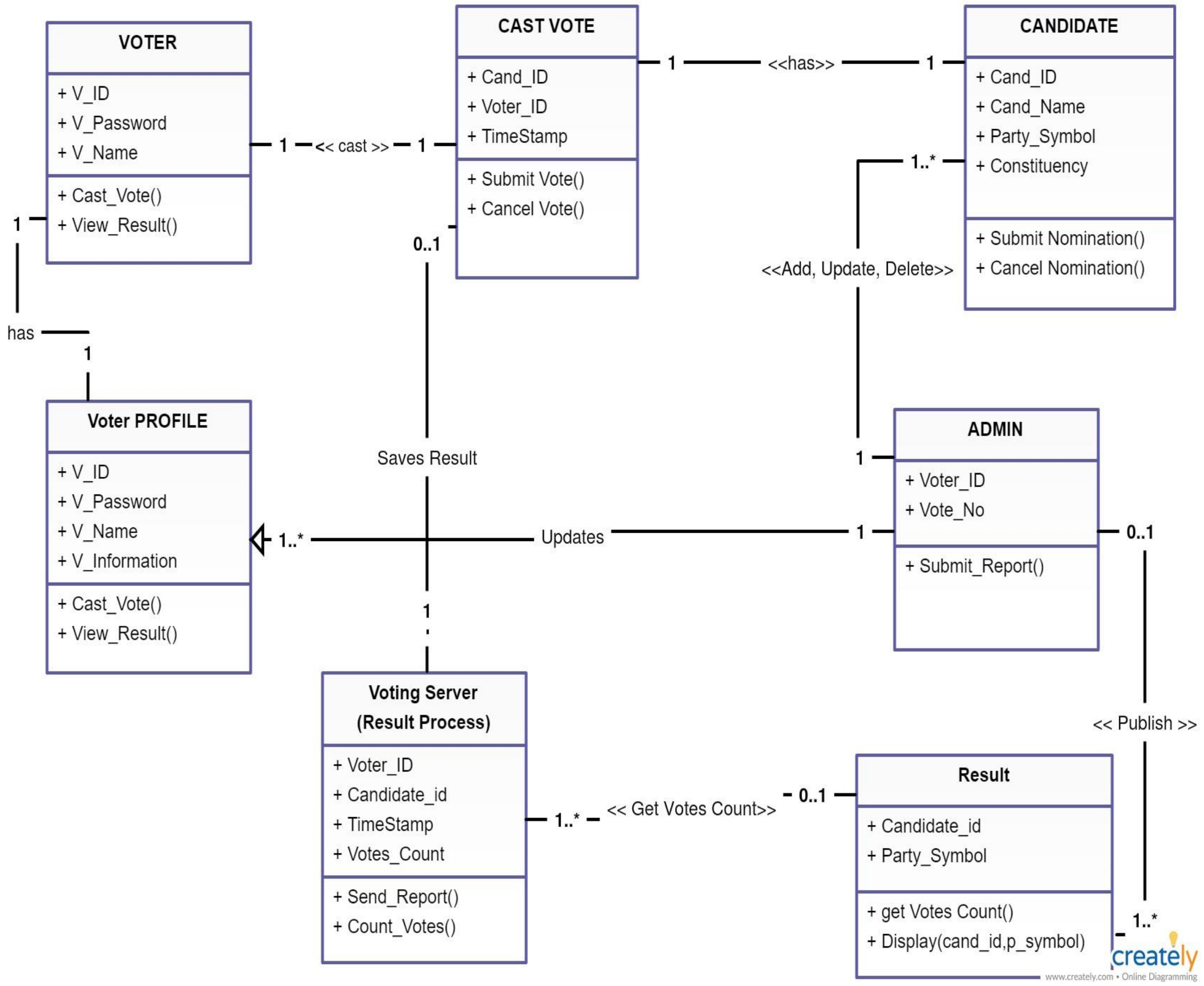
Login

Updation of software

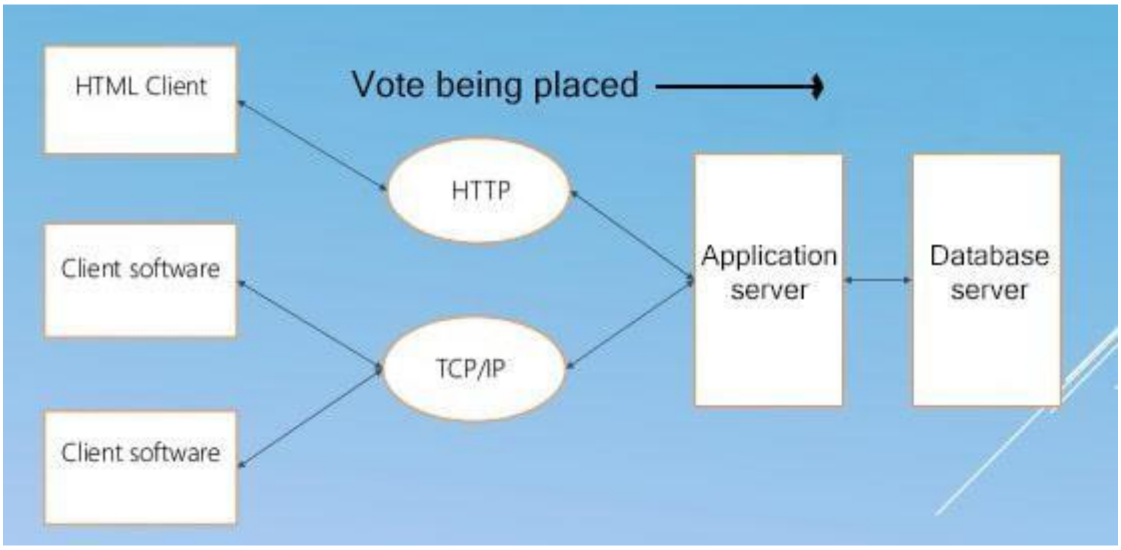
Blocking

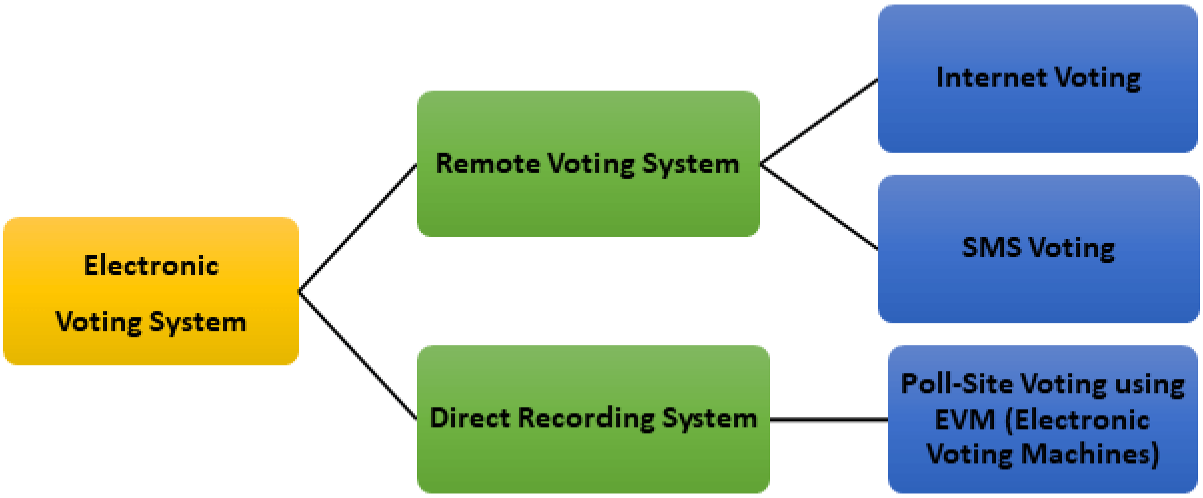
TroubleShooting

E-Voting System Class Diagram



SYSTEM ARCHITECTURE





4. TECHNOLOGY DESCRIPTION

C++ is a high-level object-oriented programming language that helps programmers write fast, portable programs. C++ provides rich library support in the form of **Standard Template Library (STL)**.

C++ Language Features

Some of the interesting features of C++ are:

- ***Object-oriented***: C++ is an object-oriented programming language. This means that the focus is on "objects" and manipulations around these objects. Information about how these manipulations work is abstracted out from the consumer of the object.
- ***Rich library support***: Through C++ Standard Template Library (STL) many functions are available that help in quickly writing code. For instance, there are standard libraries for various containers like sets, maps, hash tables, etc.
- ***Speed***: C++ is the preferred choice when latency is a critical metric. The compilation, as well as the execution time of a C++ program, is much faster than most other general purpose programming languages.
- ***Compiled***: A C++ code has to be first compiled into low-level code and then executed, unlike interpreted programming languages where no compilation is needed.
- ***Pointer Support***: C++ also supports pointers which are widely used in programming and are often not available in several programming languages.

It is one of the most important **programming languages** because almost all the programs/systems that you use have some or the other part of the codebase that is written in C/C++. Be it Windows, be it the photo editing software, be it your favorite game, be it your web browser, C++ plays an integral role in almost all applications that we use.

Uses/Applications of C++ Language

After exploring C++ features, let's have look at some interesting areas where C++ is popularly used.

Operating Systems

Be it Microsoft Windows or Mac OSX or Linux - all of them are programmed in C++. C/C++ is the backbone of all the well-known operating systems owing to the fact that it is a strongly typed and a fast programming language which makes it an ideal choice for developing an operating system. Moreover, C is quite close to the assembly language which further helps in writing low- level operating system modules.

Browsers

The rendering engines of various web browsers are programmed in C++ simply because if the speed that it offers. The rendering engines require faster execution to make sure that users don’t have to wait for the content to come up on the screen. As a result, such low-latency systems employ C++ as the programming language.

Libraries

Many high-level libraries use C++ as the core programming language. For instance, several Machine Learning libraries use C++ in the backend because of its speed. Tensorflow, one of the most widely used Machine Learning libraries uses C++ as the backend programming language. Such libraries required high-performance computations because they involve multiplications of huge matrices for the purpose of training Machine Learning models. As a result, performance becomes critical. C++ comes to the rescue in such libraries.

Graphics

All graphics applications require fast rendering and just like the case of web browsers, here also C++ helps in reducing the latency. Software that employ computer vision, digital image processing, high-end graphical processing - they all use C++ as the backend programming language. Even the popular games that are heavy on graphics use C++ as the primary programming language. The speed that C++ offers in such situations helps the developers in

expanding the target audience because an optimized application can run even on low-end devices that do not have high computation power available.

Banking Applications

One of the most popularly used core-banking system - Infosys Finacle uses C++ as one of the backend programming languages. Banking applications process millions of transactions on a daily basis and require high concurrency and low latency support. C++ automatically becomes the preferred choice in such applications owing to its speed and multithreading support that is made available through various Standard Template Libraries that come as a part of the C++ programming kit.

Cloud/Distributed Systems

Large organizations that develop cloud storage systems and other distributed systems also use C++ because it connects very well with the hardware and is compatible with a lot of machines. Cloud storage systems use scalable file-systems that work close to the hardware. C++ becomes a preferred choice in such situations because it is close to the hardware and also the multithreading libraries in C++ provide high concurrency and load tolerance which is very much needed in such scenarios.

Databases

Postgres and MySQL - two of the most widely used databases are written in C++ and C, the precursor to C++. These databases are used in almost all of the well-known applications that we all use in our day to day life - Quora, YouTube, etc.

Embedded Systems

Various embedded systems like medical machines, smartwatches, etc. use C++ as the primary programming language because of the fact that C++ is closer to the hardware level as compared to other high-level programming languages.

Telephone Switches

Because of the fact that it is one of the fastest programming languages, C++ is widely used in programming telephone switches, routers, and space probes.

Compilers

The compilers of various programming languages use C and C++ as the backend programming language. This is because of the fact that both C and C++ are relatively lower level languages and are closer to the hardware and therefore are the ideal choice for such compilation systems. These are a few uses and applications of C++ programming language. Now, let's know more about C++ advantages over other programming languages.

Advantages of C++ Language

C++ has the following 2 features that make it a preferred choice in most of the applications:

- ***Speed***: C++ is faster than most other programming languages and it provides excellent concurrency support. This makes it useful in those areas where performance is quite critical and the latency required is very low. Such requirements occur all the time in high- load servers such as web servers, application servers, database servers, etc. C++ plays a key role in such servers.
- ***Closer to hardware***: C++ is closer to hardware than most other programming languages like Python, etc. This makes it useful in those areas where the software is closely coupled with hardware and low-level support is required at the software level.

8. CODING

```
#include<stdio.h> // HEADER FILE FOR STANDARD I/0
#include<graphics.h> // HEADER FILE FOR GRAPHICS
MODE #include<dos.h>     // HEADER FILE FOR ENABLING
SOUND #include<conio.h> // HEADER FILE FOR CONSOLE
I/O #include<stdlib.h> // HEADER FILE FOR LIBRARY
FUNCTIONS union REGS i,o;
int initmouse();        // FUNCTION TO INITIALIZE MOUSE POINTER
void showmouseptr(); // FUNCTION TO SHOW POINTER
void restrictmouseptr(int,int,int,int); // FUNCTION TO RESTRICT
POINTER void getmousepos(int *,int *,int *); // TO GET POINTER
POSITION
void format();     // FUNCTION TO DRAW LAYOUT OF EVM
void graph();      // FUNCTION TO DISPLAY RESULT AS
GRAPH
void welcome();    // FUNCTION TO DISPLAY WELCOME MESSAGE
void boundry();
int vote1=0,vote2=0,vote3=0,vote4=0,vote5=0; // VARIABLES TO HOLD VOTES FOR
CANDIDATES
int
button,x,y;
void
main()
{     int gd=DETECT,gm;
      initgraph(&gd,&gm,"c:\\tc\\bgi"); // INITIALIZING GRAPHICS
      MODE randomize();
      boundry();
    welcome();               // CALLING WELCOME FUNCTION
    cleardevice();           // CLEARING THE SCREEN
    format();                // CALLING FORMAT
    FUNCTION
    showmouseptr();
    restrictmouseptr(0,0,675,435);    // RESTRICTING MOUSE POINTER WITHIN SCREEN
```

```
do
{
```

```
getmousepos(&button,&x,&y);
 if((button&1)==1&&x>475&&x<580&&y>250&&y<280)
{ break;}
else if((button&1)==1&&x>280&&x<380&&y>105&&y<125)
{ setcolor(RED);circle(270,115,5);
  sound(1200);
  delay(500);
  nosound();
  setcolor(BLACK);circle(270,115,5);
  vote1++; }
else if((button&1)==1&&x>280&&x<380&&y>155&&y<175)
{ setcolor(RED);circle(270,165,5);
  sound(1200);
  delay(500);
  nosound();
  setcolor(BLACK);circle(270,165,5);
  vote2++; }
else if((button&1)==1&&x>280&&x<380&&y>205&&y<225)
{ setcolor(RED);circle(270,215,5);
  sound(1200);
  delay(500);
  nosound();
  setcolor(BLACK);circle(270,215,5);
  vote3++; }
else if((button&1)==1&&x>280&&x<380&&y>255&&y<275)
{ setcolor(RED);circle(270,265,5);
  sound(1200);
  delay(500);
  nosound();
  setcolor(BLACK);circle(270,265,5);
  vote4++; }
```

```
else if((button&1)==1&&x>280&&x<380&&y>305&&y<325)
{ setcolor(RED);circle(270,315,5);
   sound(1200);
   delay(500);
   nosound();
   setcolor(BLACK);circle(270,315,5);
   vote5++; }
}              // END OF DO
while(1);

 cleardevice()
 ; initmouse();
 showmousep
 tr();
 boundry();
 graph();
 getch();
}                  // END OF MAIN FUNCTION
void boundry()
{
   setcolor(1+random(14));
   rectangle(0,0,635,475);
   setcolor(1+random(14));
   rectangle(3,3,632,472);
}
void welcome()
{ randomize();
  settextstyle(8,0,4);
  setcolor(1+random(1
  4));
  outtextxy(200,100,"WELCOM
  E"); delay(800);
  setcolor(1+random(14));
  outtextxy(250,160,"T
  O"); delay(800);
  setcolor(1+random(14
  ));
  outtextxy(50,220,"ELECTRONIC VOTING SYSTEM");
  delay(800);
  while(!kbhit())
  {  setcolor(1+random(14));
     outtextxy(50,400,"Press any key to continue .... ");
     delay(500);
     setcolor(BLACK);
     outtextxy(50,400,"Press any key to continue .... ");
     delay(500);
  }


}

void format()
{ setcolor(6);
    rectangle( 90,30,400,380);
    rectangle(
    87,27,403,383);
    settextstyle(0,0,5);
    outtextxy(140,40,"E V
    M"); line(90,80,400,80);
    settextstyle(8,0,2);
    outtextxy(100,100,"MEGHRA
    J");
    rectangle(95,100,250,130);
    arc(290,115,90,270,10);
    arc(370,115,270,90,
    10);
    line(290,105,370,10
    5);
```

```
line(290,125,370,12
5);
```

```
outtextxy(100,150,"DINESH");
rectangle(95,150,250,180);
arc(290,165,90,270,10);
arc(370,165,270,90,10);
line(290,155,370,155);
line(290,175,370,175);

outtextxy(100,200,"RAKESH");
rectangle(95,200,250,230);
arc(290,215,90,270,10);
arc(370,215,270,90,10);
line(290,205,370,205);
line(290,225,370,225);
outtextxy(100,250,"DEEPAK");
rectangle(95,250,250,280);
arc(290,265,90,270,10);
arc(370,265,270,90,
10);
line(290,255,370,255);
line(290,275,370,275);

outtextxy(100,300,"ANUJAY");
rectangle(95,300,250,330);
arc(290,315,90,270,10);
arc(370,315,270,90,10);
line(290,305,370,305);
line(290,325,370,325);
```

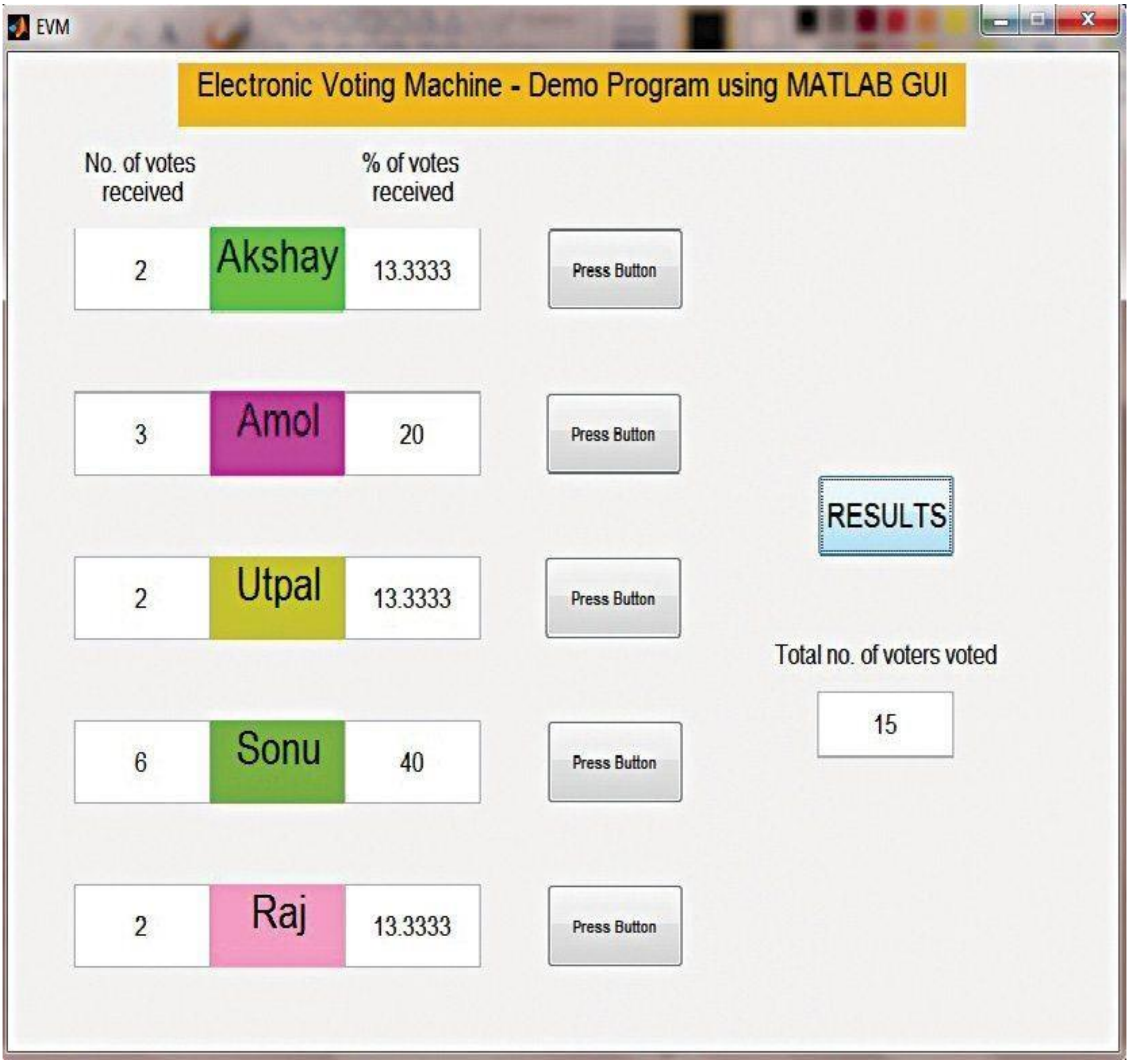
```
rectangle(475,250,580,280);
outtextxy(480,250,"RESULT
S");
```

```
   outtextxy(50,400,"Presented By:- EFY Enterprises Pvt Ltd");
}
void showmouseptr()
{
i.x.ax=1;
int86(0x33,&i,
&o);
}
void restrictmouseptr(int x1, int y1, int x2, int y2)
{    i.x.ax=7;
    i.x.cx=x1;
    i.x.dx=x2;
    int86(0x33,&i,
    &o); i.x.ax=8;
    i.x.cx=y1;
    i.x.dx=y2;
    int86(0x33,&i,
    &o);
}
void getmousepos(int *button, int *x, int *y)
{ i.x.ax=3;
   int86(0x33,&i,&o);
   *button=o.x.bx;
   *x=o.x.cx;
   *y=o.x.dx;
}
void graph()
{ outtextxy(200,100,"RESULTS(in % votes)");
   int
   candidate1=((vote1*100)/(vote1+vote2+vote3+vote4+vote5))
   ;                                                          int
   candidate2=((vote2*100)/(vote1+vote2+vote3+vote4+vote5))
   ;                                                          int
   candidate3=((vote3*100)/(vote1+vote2+vote3+vote4+vote5))
   ;                                                          int
```

```
candidate4=((vote4*100)/(vote1+vote2+vote3+vote4+vote5))
;                                                          int
candidate5=((vote5*100)/(vote1+vote2+vote3+vote4+vote5))
;
```

```
    setcolor(2);
    rectangle(100,300,130,300-
    candidate1);outtextxy(100,300,"ME");
    rectangle(200,300,230,300-
    candidate2);outtextxy(200,300,"DI");
    rectangle(300,300,330,300-
    candidate3);outtextxy(300,300,"RA");
    rectangle(400,300,430,300-
    candidate4);outtextxy(400,300,"DE");
    rectangle(500,300,530,300-
    candidate5);outtextxy(500,300,"AN");

    setcolor(1+random(14));
    rectangle(545,400,600,4
    30);
    outtextxy(550,400,"EXI
    T");

    do
{
getmousepos(&button,&x,&y);
 if((button&1)==1&&x>545&&x<600&&y>400&&y<430)
{ break;}
}               // END OF DO
while(1);



}
initmouse()
{
            ax=0;
            int86(0x
            33,&i,&
            o);
            return(o
        .x.ax);
}
```

9. SCREENSHOTS



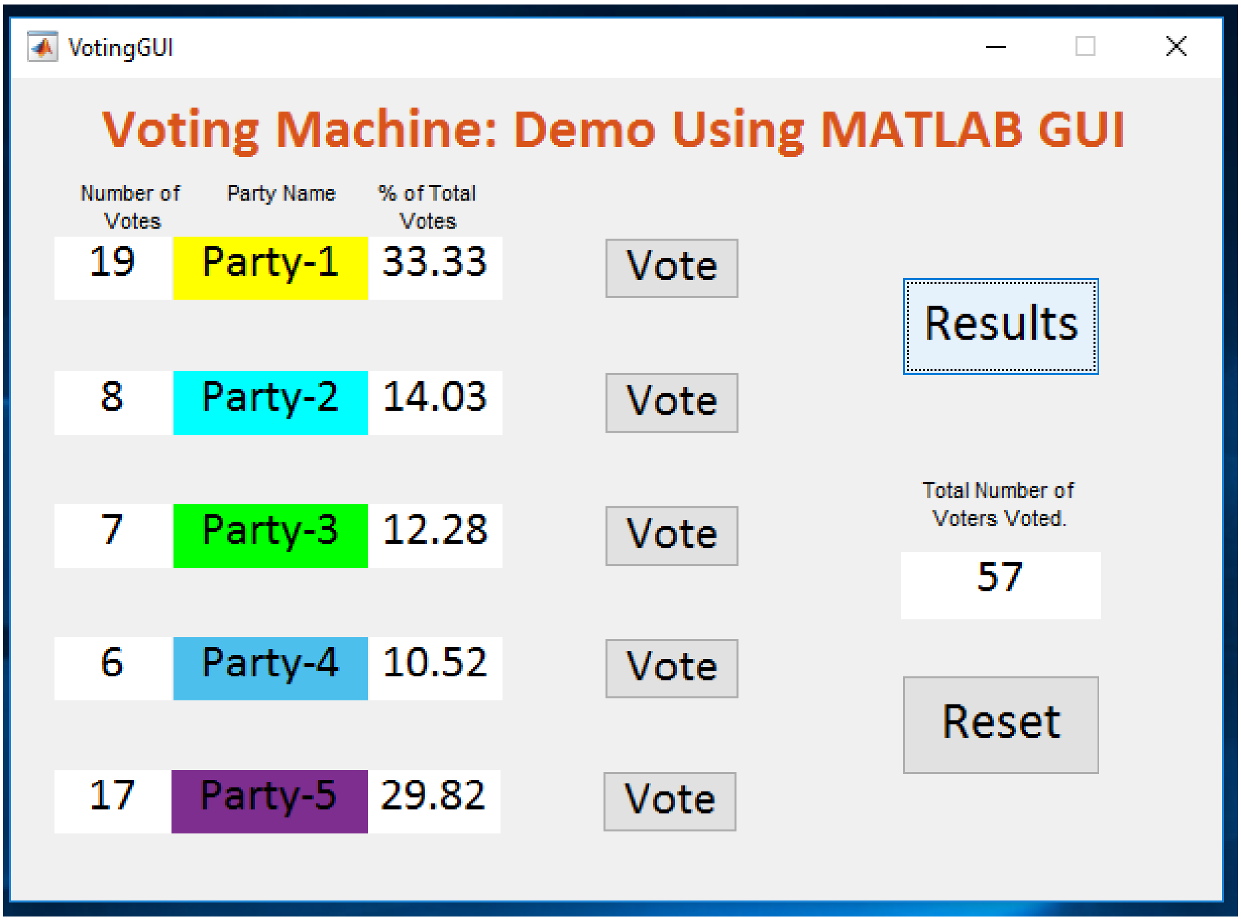
EVM

Electronic Voting Machine - Demo Program using MATLAB GUI

No. of votes received		% of votes received
2	Akshay	13.3333
3	Amol	20
2	Utpal	13.3333
6	Sonu	40
2	Raj	13.3333

Total no. of voters voted

15

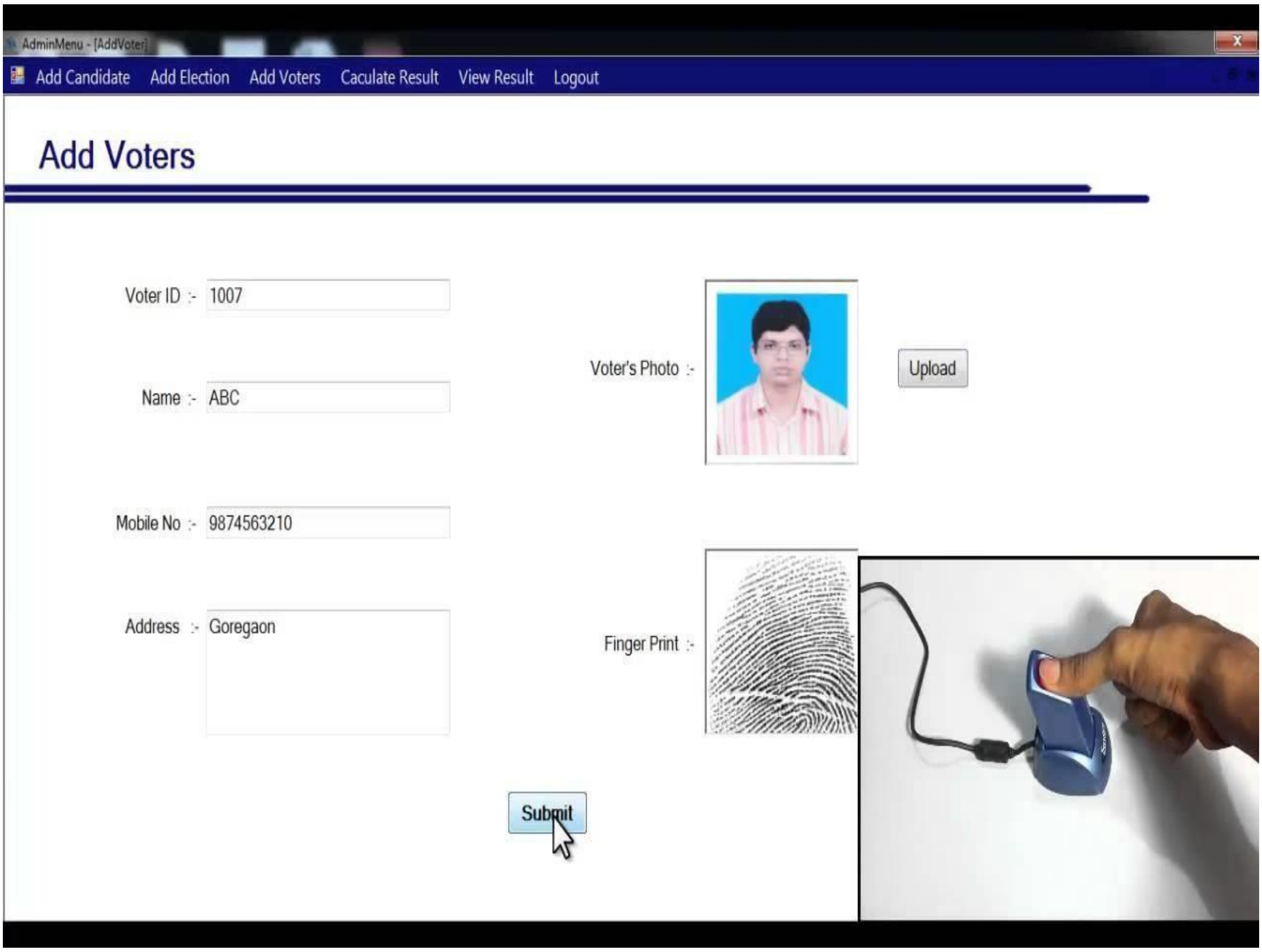


Voting Machine: Demo Using MATLAB GUI

Number of Votes	Party Name	% of Total Votes
19	Party-1	33.33
8	Party-2	14.03
7	Party-3	12.28
6	Party-4	10.52
17	Party-5	29.82

Total Number of Voters Voted.

57



Add Voters

Voter ID :- 1007

Name :- ABC

Mobile No :- 9874563210

Address :- Goregaon

Voter's Photo :-

Finger Print :-

10. CONCLUSION

The function of compiling, editing and processing data makes the voting system complicated and tedious process. Electronic voting system is an effort in the direction to use computer for the purpose of voting system. It is both user friendly as well as time saving. This is a foundation level. Emphasis has been given in this application to replicate all the process required in a traditional system. EVM system is self-sufficient with all the information required about the voters, candidate and the voters. Once the information is fed to the system, it can identify individual voters, their votes and then can press any key to return such as counting the votes has been simplified in this system. The basic function of the polling has been reduced to single key operation. A single polling manager can manage large number of voters.

LIMITATIONS:

An effort had been done to develop the system with the wide scope in view. Regardless of the number of application the software has its own limitation. The following are some of the notable limitation of the electronic voting system.

1. Security: - The system administrators can hacks the system at his will. He can change data output and hence can affect the result.
2. Voters education level: - Voters should be at least that much educated to enter the vote, i.e. press the required button. A totally computer illiterate has a very high chance of polling a wrong vote i.e. in favors of an undesirable candidate.
3. Specific voting system: - Since the system has been developed to handle the specific type of vote, i.e. only one candidate has to be chosen from the given list by one vote. This system cannot be used for other specific type of voting for e.g. electoral college voting (Indian resident Election) and conditional voting.
4. Basic system: - The EVM system is itself a basic and elementary form of software and hence cannot support other input devices as separate keyboard etc.

5. Confidentiality: - If the requirement of the voting is to keep the confidentiality of the votes, it is not possible since the system administrator would always be in a position to overview the votes of each and every individual at the time of voting.
6. The screen: - The screen is not too good in clarity and there are chances of mistakes during voting.

POSSIBLE IMPROVEMENTS:

5. In case of large data storage, the storing of data e.g. individual voter details would take a long time. Hence improvement applications can minimize the time.

6. Specific input devices (e.g. voting ads) could be an added advantage to the system.

7. Candidate name and number are only possible options in this application. Individual candidate terms can be added to the application.

8. Improvement can be done to make the application workable on net so that voting can be possible.

SCOPE:

5. Consumer survey
6. Large scale voting
7. Distance voting
8. Highly secured and error free result

BIBLIOGRAPHY

Text Books:

4. TURBO C++ by Robert Lafore
5. C++ Programming by E. BALAGURUSAMI
6. C++ by YASWANT KANITKAR

References:

www.joblagao.com

www.electronicsforu.com

1. INTRODUCTION

The function of compiling, editing and processing data makes the voting system complicated and tedious process. Electronic voting system is an effort in the direction to use computer for the purpose of voting system. It is both user friendly as well as time saving.

This is a foundation level. Emphasis has been given in this application to replicate all the processes required in a traditional system. EVM system is a self-sufficient with all the information required about the voters, candidate and votes. Once the information is fed to the system it can identify individual voters, their votes and the candidates. Tedious function such as counting the votes has been simplified in this system. The basic function of the polling has been reduced to single operation. A single polling manager can manage large number of voters.

The program code has been Written in 'C++' language has been chosen because of its procedural nature and the flexibility it provides in interacting with hardware and operating system efficiently.

NEED TO COMPUTARISE THE SYSTEM

Information is needed in the organization for planning, staffing and controlling purposes. Regardless of the nature of the information required, the information should process the characteristics of accuracy, timeliness and relevancy. In the recent years need for information improvement by reports lacking one or more of these characteristics and by increased paper work volume, rising casts and pressures from outside changes.

Fortunately computers thrive on repetitive large volume processing tasks, are fast and accurate. The processing capability in many organizations has been stained by:

7. Growth in size and complexity of the organization.
8. The increased requirements for data from external sources.
9. The demand of administrators for more information.

More than a million new pages of data are generated each minute of the day in offices . Compare to other processing method, the use of computers may make it possible for certain administrative costs to be reduced while the level of processing activity remains stable. The increased cost and clerical labor materials and other expenses associated with the data processing operation require eventual managerial attention.

We all agree that meaningful information is timely information. But with an increase in volume and size of an organization, there is only a reduction in the speed of processing. Rapid changes are taking place in the world socially, economically and technically. Such changes have a significant impact on the environment which organization must operate on the planning that managers must do and on the information that must have. Many educational administrators are responsible for supervising the activities of a large number of schools scattered over a district. They must have accurate information if they are to control such an effort properly. But if a data processing operation is stained to or beyond the capacity for which it was originally planned, inaccuracies will begin to appear. Inadequate control will permit inadequate performance. Thus the administrator will logically demand better quality in the information he receives. It is due to these reassures (increased paper work volume, cost, pressure from outside changes demand for timeliness and demand for quality) that most if the organization today is opting for computers to do data processing for them.

Computers are most efficient used in processing operation that includes:-

Large volume of input:- Greater the volume of data that must be processed to produce the needed information, the more economical computer processing becomes relative to other possible methods.

Repetition of projects: - Because of the expenses involved in preparing a task for computer processing, it is frequently more economical to use the computer for repetitive purpose.

Desired and necessary greater speed in processing:- Greater the need for timely information, greater will be quite accurate if the task to be performed has been properly prepared.

THE VOTING SYSTEM:

The voting system can be described as “A process in which the input (voter) votes in favors of a given decision (candidate)”. Votes are procured, counted and a statically result is provided.

Traditionally there are two basic types of voting system:-

Open voting system: - In this system the voter votes in favors of a decision without hiding his identity, example of system is rising of hands. This is in itself a very elementary system and useful when the level of decision to be taken is comparatively less effective to the voter. The number of voters should be less and countable and discipline and honestly is maintained by the voters. The voting system is people oriented.

Second voting system: - In this system identity of the voter is hidden and once the vote has been given it is not possible identifies a given vote with the voter. In this system paper work is usually done, as the identity of the voter has to be concealed. A very large number of voters can take part in such system. The whole process in itself system oriented and have security secrecy and counting becomes easier.

Computer Voting System

Manual voting system at large scale becomes a tiring and costly affair in terms of money, manpower and time. The use of computer can easily reduce the effort and make voting an easy as well as voter and candidate friendly affairs. It can reduce the amount of money spent, manpower and time to a very large extent.

In the present system of voting the individual votes are registered in form of ballot papers. The names of candidates is written or printed in the ballot papers in serial orders, which can be easily recognized by the voters. A separate list of voters is made so that the voters can be allowed to sequentially and chances of non-voters or revolting of an individual can be controlled. The whole system is done under the control of a residing officer. Each voter is given a ballot paper on which he stamps his individual choice. The ballot paper is then folded in a particular way and then insert in the ballot box. As soon as an individual voter casts his vote, his name is deleted from the voter list.

At the end of polling the ballot boxes are sealed and sent for counting. The counting of the ballot papers is done in a redefined method. The results are declared after getting convinced that the counting has been done properly. The process of counting takes time since it is done manually.

Problems in Existing System:

The present system has been marked by many problems some of which have been enumerated below: -

Expensive: - The present form of voting is an expensive affair. If needs a lot of preparation in terms of time and money. The cost of paper work and stationary is one of the most expensive parts. Expenses are incurred in processing ballot boxes and ballot papers. If in case ballot papers are printed wrong re-printing has to be done.

Time consuming - A lot of time is consumed in counting of votes. The manual counting is in itself a tedious job and hence requires time.

Invalid votes: - Manual voting has a regular problem of invalid votes. A number of votes become invalid due to a number of unintentional reasons.

Manipulation work: - Voting process at times is hindered by presence of external elements to favor a given candidate. Examples are booth capturing.

Manpower required: - The present system of voting involves the use of lots of manpower at each and every stage of voting process.

User identification: - This poses one of the serious problems of current voting system. During voting process many people cast there vote more than one times. This causes problems as wrong people are selected.

Manipulation in counting: - As counting process is done manually it may happen that votes are wrongly counted or they may be manipulated in some way in favors of some candidates.

Portability factor: - After polling is over ballot boxes are usually taken to some other places for counting. This is risk as the ballot boxes may be stolen or there may be some mishap reroute to counting system.

Software Program of the EVM:

Despite design features that make the election software complex to extract the information from the control unit processor, a criminal has a variety of options to steal the information from it. These include decapsulating the chip and examining it under a microscope. Here we did not attempt to extract the software using these methods; once the software is extracted, it is straightforward to reverse- engineer it using standard disassembly tools.

The software (evm.cpp) developed here for the demo is tested on Turbo C++ Version 3.0 IDE. When the user runs the program, it displays a real-time voting machine on the screen with the list of the candidates and corresponding buttons to vote for them (refer Fig. 2). The program requires ‘C’ graphics files such as egavga.bgi, goth.chr, lcom.chr and trip.chr for proper display.

In order to vote for a party, the user has to click the respective button using mouse. A beep sounds to confirm successful submission of vote.

At any point of time if you want to check the status of the votes being polled, just click ‘result’ button in the control panel (not shown here). The next screen will display the percentage of votes received by different candidates in the form of a bar chart. To exit the program, click ‘exit’ button.

2. SYSTEM REQUIREMENT SPECIFICATION(SRS)

SYSTEM REQUIREMENTS:

The aim of the system to be developed ------- Electronic-voting system is to develop software, which should automate the process of voting of an electoral system. The system is supposed to be used as a subsystem in a layer electoral system, which could be a manual system or a computerized one. Therefore the proposed system must be able to function under both circumstances.

The basic functionality that the system must provide is

15. The system must provide for manipulation of candidate data.

The manipulation of candidate data consists of: -

Adding a new voter record in the existing voter list. The system must provide enough mechanisms to check the data validity and integrity.

Deletion of existing voters from the voter list. The system must be able to notice and notify the deletion request for a non-existent record. The system should have a mechanism of asserting record deletion to avoid accidental deletion of record.

Modification of voter records in the voter list. The system must be able to modify details of a voter for any specific data field, or for all the fields. The system must provide for validation in this module also. The system should have a mechanism of asserting record modification to avoid accidental modification of records.

16. The system must provide for manipulation of candidate data. The manipulation of candidate data consists of: -

Adding a new candidate record in the existing list. The system must provide enough mechanisms to check the data for validity and integrity.

Deletion of existing candidates from the candidate list. The system must be able to notice and notify the deletion request for anon-existent record. The system should have a mechanism of asserting record deletion to avoid accidental deletion of records.

Modification of candidate records in the candidate list. The system must be able to modify details of a candidate for any specific data field, or for all the fields. The system must provide for validation in this module also. The system should have a mechanism of asserting records modification to avoid accidental modification of records.

17. The system must have a rich set of utilities for listing the various details of the candidates as well as voters. The listing should be both a generalized listing for all candidates or voters. And detailed listing of an individual candidate or voter.

18. The system must provide a method and interference for the voting process. In which a voter can select a candidate from the list of candidates. There should further be a method to display the details of the winner of a particular voting process. The details of the margin of the victory should also be displayed.

19. The system must provide interference to print the details of candidates and voters. In both a summarized format and a detailed individual report.

20. The system must provide help whenever necessary. And should also provide tips, validity criteria etc… while data manipulation.

21. The system must have easy way of switching in between different modules.

NON-FUNCTIONAL REQUIREMENTS:

Nonfunctional requirements are just as important to your business analysis as the functional requirements when it comes to defining the look and feel of the solution. Nonfunctional requirements are a challenge because different people interpret them differently from organization to organization (or even from department to department in the organization). You need to understand a lot about the people using the solution and make sure your nonfunctionals document its performance.

You create the nonfunctional requirements based on your elicitations from the users, who they are, and what their expectations of the system performance are. When you create nonfunctional requirements, you need to think about things like the following:

Performance: How well does the system perform? To understand the performance requirements, ask stakeholders questions such as "What are the number of concurrent users?", "What are the system or query response times?", and "What is the system's capacity in terms of memory, disk space, and data volumes?"

Security: Who has access to the system, and how much access do they have? To understand the security requirements, ask questions such as "Which users are authorized to perform which functions?", "What is the privacy of the information being captured and stored?", and "What features need to be in place to log user access and authenticate users?"

Reliability: *Reliability* is how the system operates based on the expectation of the end-user. Think about buying a car. You probably *purchase* a car because of the *functionality* (0–60 mph in 8 seconds, A/C, satellite radio, and so on), but you probably think about going to shop for that new car because of the *reliability* of the car. Similarly, you want to make sure you find out how consistently the business wants the solution to perform and what maintenance and support you need to make sure it stays that way. To elicit the reliability requirements, ask questions such as "When is the system expected to be available?", "What downtime does the system have for the administrators to perform maintenance, and when is the best time to schedule downtime?", and

“What notification do the users need when the system is going down for maintenance? How much advance notice should they receive?”

Compatibility: *Compatibility* refers to the extent to which the solution plays nice with other applications. To elicit compatibility nonfunctional requirements, ask questions such as “What common standards, common technology, and protocols exist on the workstation?”; “How well does the solution work with the common build?”; “What kinds of data exchange do you envision?”; and “What information (data) must be exchanged with other systems?”

Maintainability: *Maintainability* deals with how easy the system is to maintain and repair. To elicit the nonfunctionals for maintainability, ask questions such as “What is the ability to change one component without affecting others?”, “What effects do the maintenance activities have on customers, users, and employees?”, and “Who performs system upgrades? Who is responsible for interfaces?”

Transferability: *Transferability* refers to the ease with which a system can be transferred to a different hardware or software environment. Some of these concerns are lessening now that many companies are creating browser-based applications, yet these concerns have expanded with the mobile apps (like those you see on your smartphone) and the different versions and standards for e-readers.

Usability: *Usability* concerns the ways by which the user is able to learn, operate, and interpret the system results. This category includes ease of entry, learning, and handling, as well as the system’s intuitiveness.

Metrics and measurements: With any nonfunctional requirement, you must understand what measurement criteria you’ll use to determine whether the requirement is successful and met. You’re defining how well the solution meets the requirements. To elicit the metric, ask questions like “What are some aspects surrounding that requirement that you can measure?” and “What are the acceptable measurement time frames that are acceptable for the stakeholder?”

SYSTEM SPECIFICATION:

MINIMUM HARDWATRE

REQUIREMENTS PROCESSOR:

DUAL CORE OR

ABOVE RAM: 2GB OR ABOVE

HARD DISK: 256GB OR ABOVE

PRINTER: LINE_PRINTER (DOT

MATRIX/INKJET) KEYBOARD:NORMAL OR ABOVE

APPLICATION SOFTWARE

C++ S/W BORLAND C++/ TURBO

C++ OPERATING SYSTEM:

WINDOWS 98 S/W

LANGUAGE: C++

3. SYSTEM DESIGN

DESIGN SPECIFICATION:

To implement the requirements as specified in the system requirement specification document of Electronic Voting System. The proposed system is to be developed using “c++” language. The system has a main module, which will have several other modules as depicted in the structure diagram of the system.

The functionally requirements will be fulfill as: -To provide for manipulation of voter data

A new voter record is added in the existing voter list, by using add-vote () Module. This module will use the voter structure to input voter details. Each field in the voter structure will be checked for the integrity constraints. As specified in SRS, the details are added into existing voter detail file voter.dat. This file ensures the secondary storage of data.

Deletion of existing voters from the voter list is done by the module delete-voter (). The system checks for the existing of the record to be delete in the file voter.bat. For nonexistent data it notifies the user .The module also record deletions by flashing a message.

Modification of voter records in the voter list is handled by a module modify-voter (). This module reads the voter No from the user searches has its existence in the file voter.dat. If the record is found the details are displayed where typing new values can modify them. The module also assents record modification by flashing a message.

2. To provide for manipulation of candidate data.

A new candidate record is added in the existing candidate list, by using add-candidate () module. This module will use the candidate structure to input candidate details. Each field in the candidate structure will be checked for the integrity constraints as specified in SRS. The details are added into exiting voter-details file voter.dat. The file is ensures the secondary storage of the data.

Deletion of existing candidates from the voter list is done by the module delete- candidate (). The system checks for the existence of the record to be deleted in the file candidate.dat. For non-existent data it notifies the user. The module also asserts record deletion by flashing a message.

Modification of voter records in the voter list is handled by a module modify-candidate (). These module reeds the voter numbers from the user searches its existence in the file candidate.dat. If the record is found the details are displayed where typing new values can modify them. The module also asserts record modification by flashing a message.

12. The listing of the various details of the candidates as well as voters. Handled by four different module
dislay-code (),
showlist-voter
(), display-
candidate(),
showlist-candidate () modules give detailed listing of an individual candidate or voter showlist-voter ().
Showlist-candidate () gives generalized listing for all candidates of voter.

13. The voting process is handled by the module vote (), fmain (), find (), showlist (). A voter can select a candidate from the list of candidates, this functionality is provided by module vote (). The method to display the details of the winner of a particular voting process is handled by find () and showlist (). The detail of the margin of the victory is dislayed by showlist().

14. The system provides interface to print the details of candidates and voters by using module print (), which is called from different modules with different values to print different values to print different data as per the specification of SRS.

15. The system provides help by a module help (). And provides tips, validity criteria etc, by displaying message at the screen using a module statusbar().

DATA DICTIONARY

FUNCTION	DESCRIPTION
19. ADD_VOTER()	This function adds voters name to the voter list
20. VOTER_INFORMATION ()	This function gives information about the voters, name, address etc by putting its voter number.
21. VOTER_LIST ()	This function shows list of the voters
22. ISSUE-VOTING_CARD ()	This function ISSUES voting cards to the voters for validating the election.
23. RESET_STATUS()	This function resets the date to get it filled again f illed.
(A) FOR CANDIDATE	To reset votes received by candidate to zero.
(B) FOR VOTER	To reset all the status in the voter file.
24. DELETE_VOTER () .	To give the voter name to delete voter record.
25. ELECTION ()	This function is used to give votes by the voters.
26. VOTE_CALCULATION ()	To calculate votes given by the voters.
27. ADD_CANDIDATE()	This function adds the candidate record to the Candidate file.
14 CANDIDATE_INFORMATION ()	This function gives the information of the candidates.
15 DELETE_CANDIDATE ()	To give candidate serial number to delete recor
26. VOTING_LIST ()	To display of the votes received by the candidates.
27. ADD_VOTES	To add votes in the favors of the given candidate.
28. VOTING_DATE ()	To display the date of voting.
29. SET_DATE ()	To set the date for the new election.
30. VALID_DATE ()	To return the validity of today's date.
31. RETURN_DATE ()	To return the election date.
32. LINE_HOR ()	To draw horizontal line.

Data Base Tables:-

This project uses many tables:

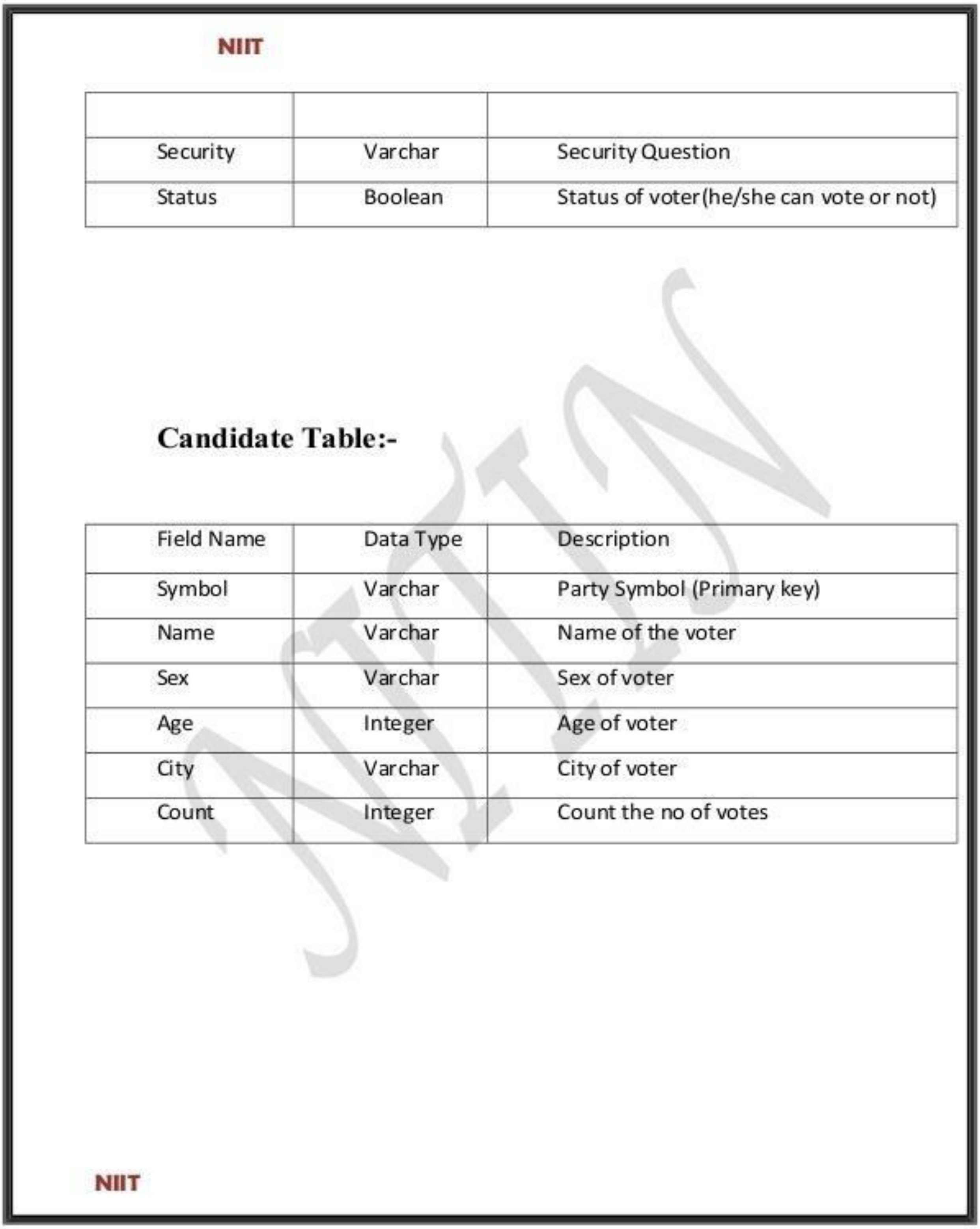
- Admin
- Voter
- Candidate

Admin Table:-

Field Name	Data Type	Description
Username	Varchar	Login id for Admin.(Primary key)
Password	Varchar	Password for Login

Voter Table:-

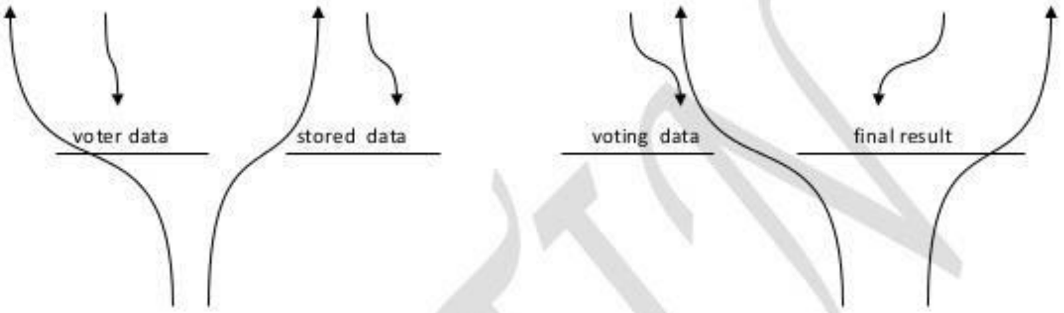
Field Name	Data Type	Description
VoterId	Integer	Login id for Voter(Primary key)
Name	Varchar	Name of the voter
Sex	Varchar	Sex of voter
Age	Integer	Age of voter
City	Varchar	City of voter



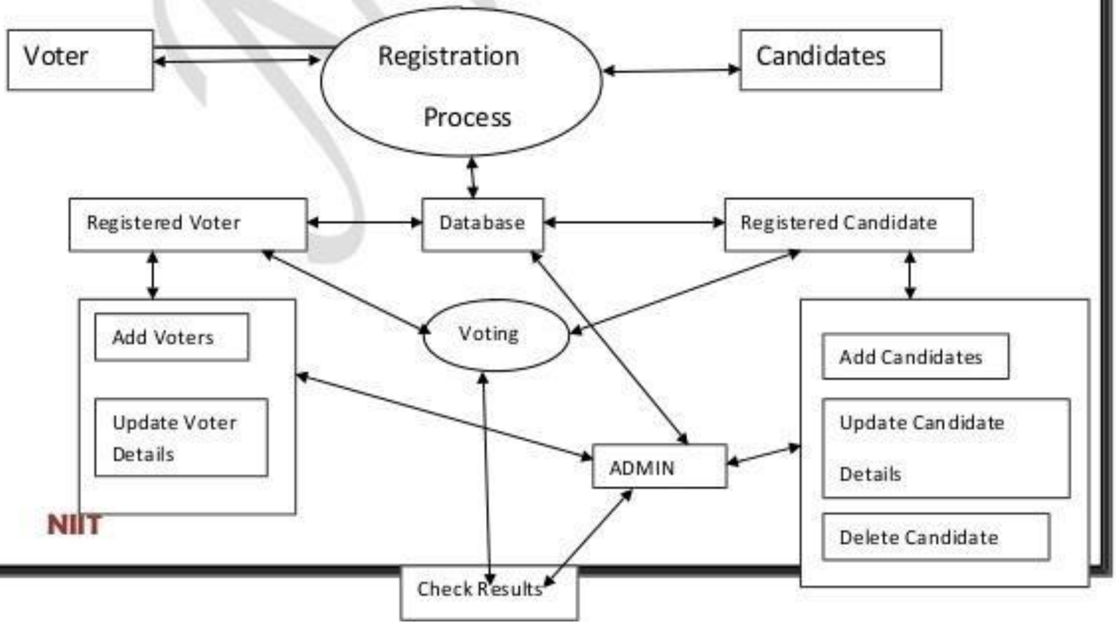
Security	Varchar	Security Question
Status	Boolean	Status of voter(he/she can vote or not)

Candidate Table:-

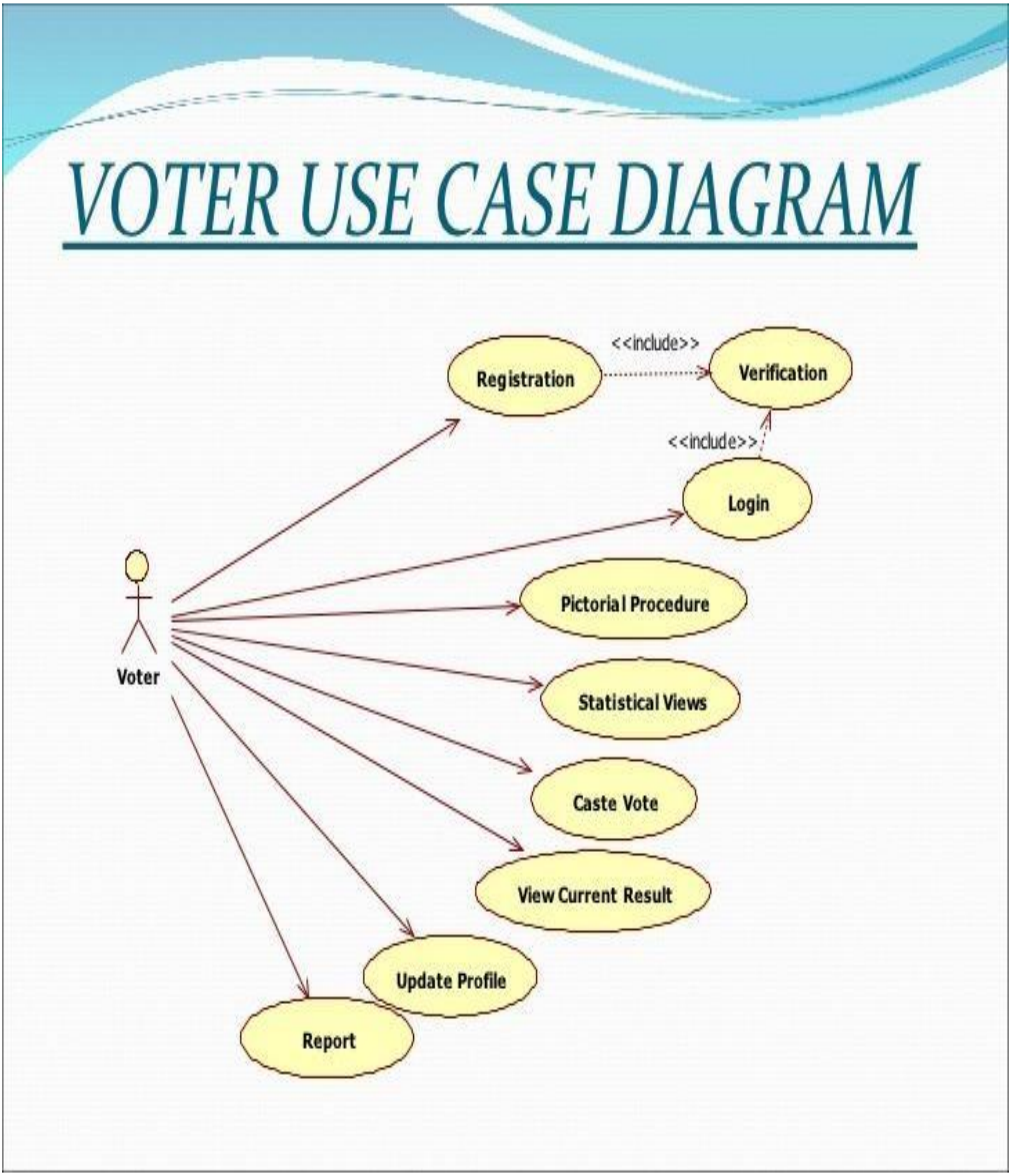
Field Name	Data Type	Description
Symbol	Varchar	Party Symbol (Primary key)
Name	Varchar	Name of the voter
Sex	Varchar	Sex of voter
Age	Integer	Age of voter
City	Varchar	City of voter
Count	Integer	Count the no of votes


voter data
stored data
voting data
final result

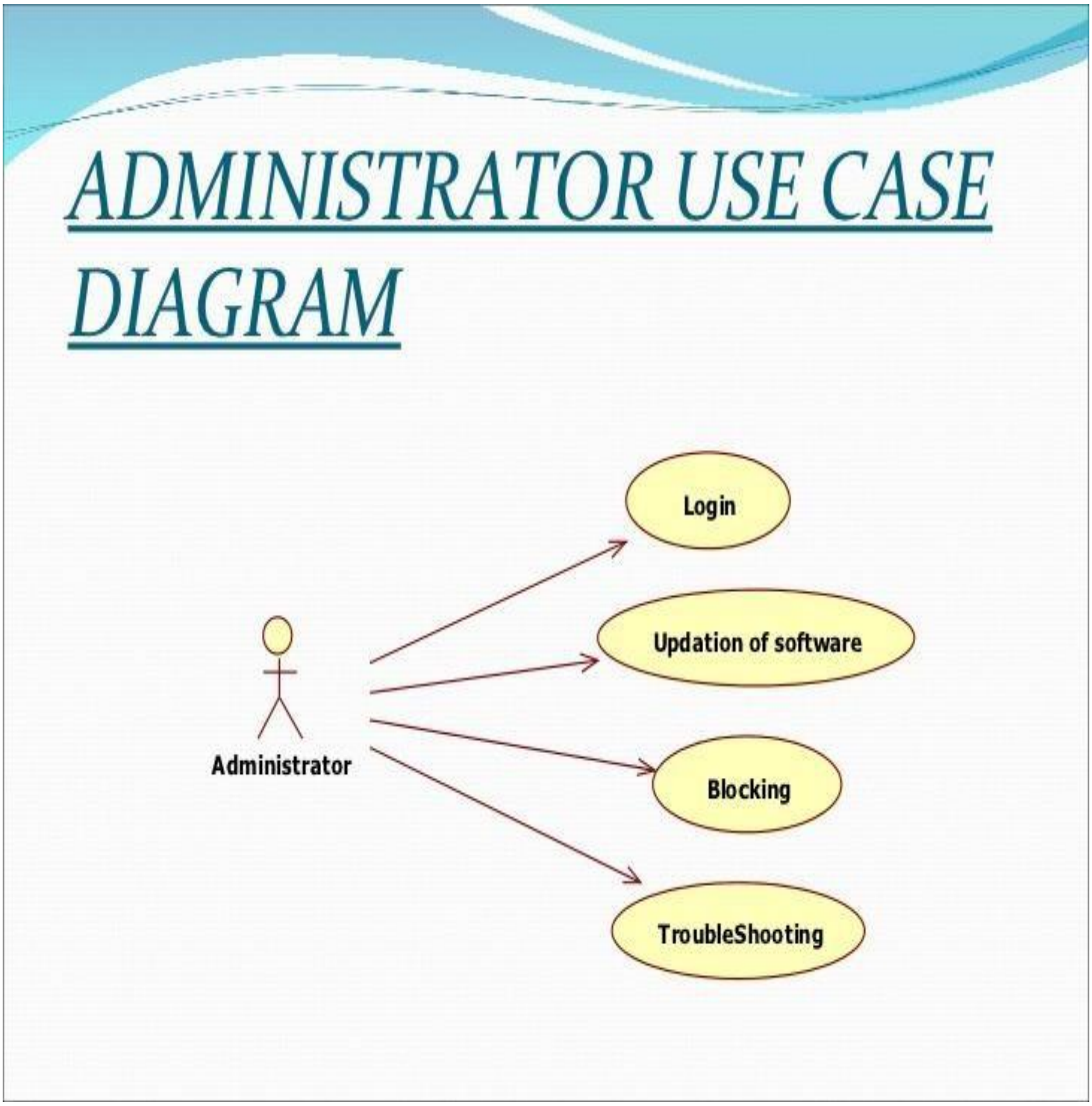
ER Diagram:


Voter
Registration Process
Candidates
Registered Voter
Database
Registered Candidate
Add Voters
Update Voter Details
Voting
Add Candidates
Update Candidate Details
ADMIN
Delete Candidate
Check Results

Use case Diagram

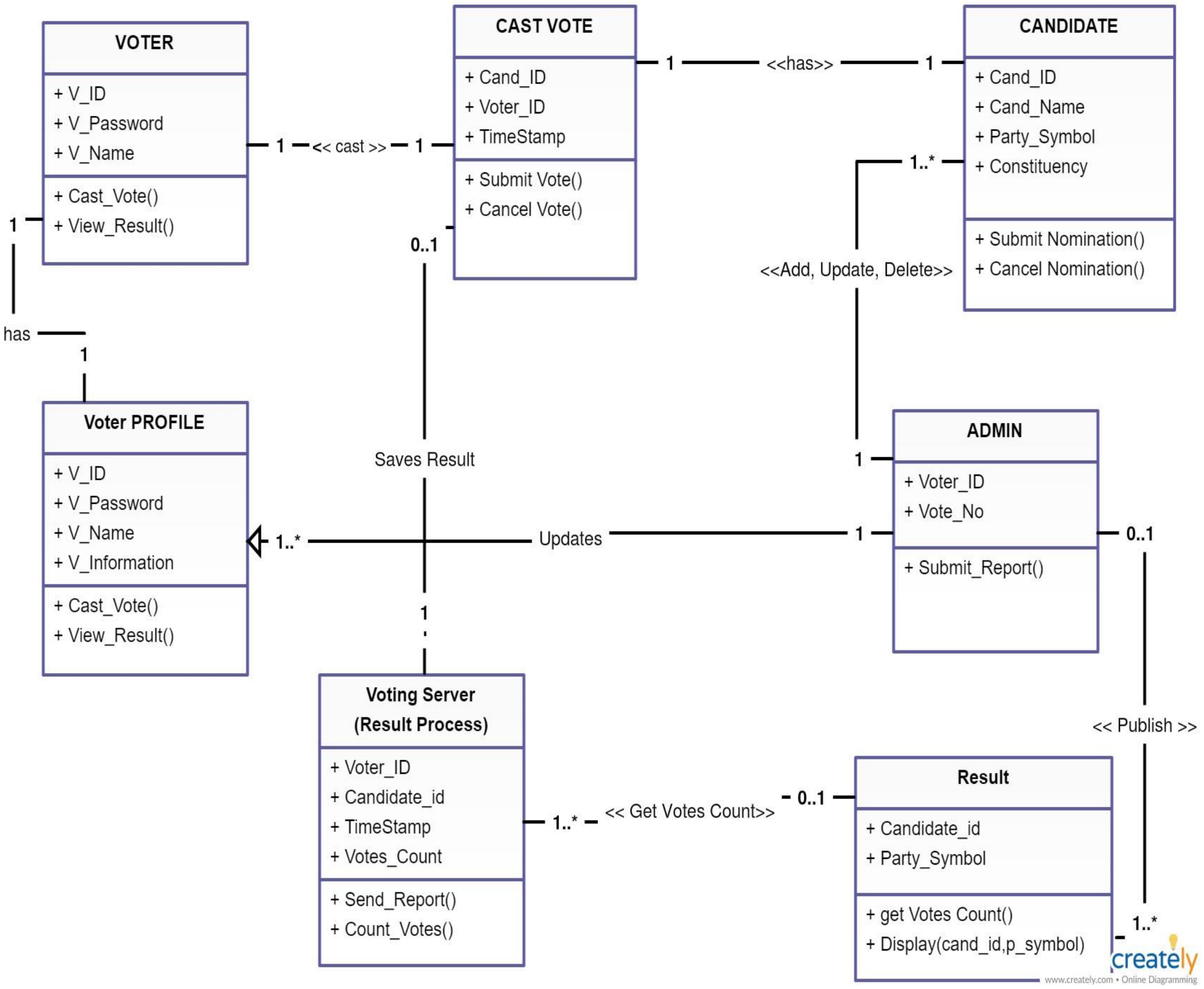


VOTER USE CASE DIAGRAM

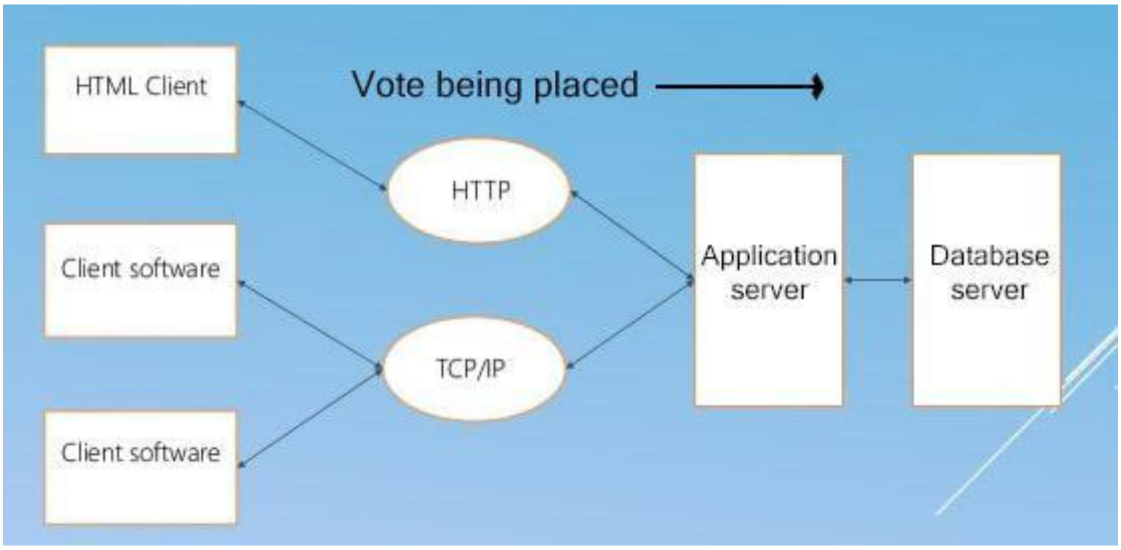


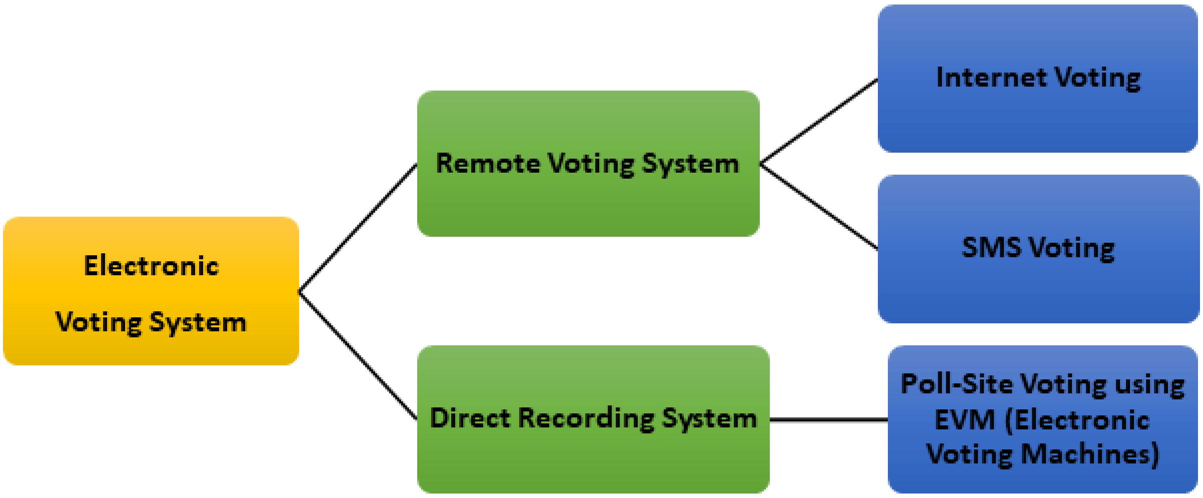
ADMINISTRATOR USE CASE DIAGRAM

E-Voting System Class Diagram



SYSTEM ARCHITECTURE





4. TECHNOLOGY DESCRIPTION

C++ is a high-level object-oriented programming language that helps programmers write fast, portable programs. C++ provides rich library support in the form of **Standard Template Library (STL)**.

C++ Language Features

Some of the interesting features of C++ are:

- ***Object-oriented***: C++ is an object-oriented programming language. This means that the focus is on "objects" and manipulations around these objects. Information about how these manipulations work is abstracted out from the consumer of the object.
- ***Rich library support***: Through C++ Standard Template Library (STL) many functions are available that help in quickly writing code. For instance, there are standard libraries for various containers like sets, maps, hash tables, etc.
- ***Speed***: C++ is the preferred choice when latency is a critical metric. The compilation, as well as the execution time of a C++ program, is much faster than most other general purpose programming languages.
- ***Compiled***: A C++ code has to be first compiled into low-level code and then executed, unlike interpreted programming languages where no compilation is needed.
- ***Pointer Support***: C++ also supports pointers which are widely used in programming and are often not available in several programming languages.

It is one of the most important **programming languages** because almost all the programs/systems that you use have some or the other part of the codebase that is written in C/C++. Be it Windows, be it the photo editing software, be it your favorite game, be it your web browser, C++ plays an integral role in almost all applications that we use.

Uses/Applications of C++ Language

After exploring C++ features, let's have look at some interesting areas where C++ is popularly used.

Operating Systems

Be it Microsoft Windows or Mac OSX or Linux - all of them are programmed in C++. C/C++ is the backbone of all the well-known operating systems owing to the fact that it is a strongly typed and a fast programming language which makes it an ideal choice for developing an operating system. Moreover, C is quite close to the assembly language which further helps in writing low- level operating system modules.

Browsers

The rendering engines of various web browsers are programmed in C++ simply because if the speed that it offers. The rendering engines require faster execution to make sure that users don’t have to wait for the content to come up on the screen. As a result, such low-latency systems employ C++ as the programming language.

Libraries

Many high-level libraries use C++ as the core programming language. For instance, several Machine Learning libraries use C++ in the backend because of its speed. Tensorflow, one of the most widely used Machine Learning libraries uses C++ as the backend programming language. Such libraries required high-performance computations because they involve multiplications of huge matrices for the purpose of training Machine Learning models. As a result, performance becomes critical. C++ comes to the rescue in such libraries.

Graphics

All graphics applications require fast rendering and just like the case of web browsers, here also C++ helps in reducing the latency. Software that employ computer vision, digital image processing, high-end graphical processing - they all use C++ as the backend programming language. Even the popular games that are heavy on graphics use C++ as the primary programming language. The speed that C++ offers in such situations helps the developers in

expanding the target audience because an optimized application can run even on low-end devices that do not have high computation power available.

Banking Applications

One of the most popularly used core-banking system - Infosys Finacle uses C++ as one of the backend programming languages. Banking applications process millions of transactions on a daily basis and require high concurrency and low latency support. C++ automatically becomes the preferred choice in such applications owing to its speed and multithreading support that is made available through various Standard Template Libraries that come as a part of the C++ programming kit.

Cloud/Distributed Systems

Large organizations that develop cloud storage systems and other distributed systems also use C++ because it connects very well with the hardware and is compatible with a lot of machines. Cloud storage systems use scalable file-systems that work close to the hardware. C++ becomes a preferred choice in such situations because it is close to the hardware and also the multithreading libraries in C++ provide high concurrency and load tolerance which is very much needed in such scenarios.

Databases

Postgres and MySQL - two of the most widely used databases are written in C++ and C, the precursor to C++. These databases are used in almost all of the well-known applications that we all use in our day to day life - Quora, YouTube, etc.

Embedded Systems

Various embedded systems like medical machines, smartwatches, etc. use C++ as the primary programming language because of the fact that C++ is closer to the hardware level as compared to other high-level programming languages.

Telephone Switches

Because of the fact that it is one of the fastest programming languages, C++ is widely used in programming telephone switches, routers, and space probes.

Compilers

The compilers of various programming languages use C and C++ as the backend programming language. This is because of the fact that both C and C++ are relatively lower level languages and are closer to the hardware and therefore are the ideal choice for such compilation systems. These are a few uses and applications of C++ programming language. Now, let's know more about C++ advantages over other programming languages.

Advantages of C++ Language

C++ has the following 2 features that make it a preferred choice in most of the applications:

- ***Speed***: C++ is faster than most other programming languages and it provides excellent concurrency support. This makes it useful in those areas where performance is quite critical and the latency required is very low. Such requirements occur all the time in high- load servers such as web servers, application servers, database servers, etc. C++ plays a key role in such servers.
- ***Closer to hardware***: C++ is closer to hardware than most other programming languages like Python, etc. This makes it useful in those areas where the software is closely coupled with hardware and low-level support is required at the software level.

11. CODING

```
#include<stdio.h> // HEADER FILE FOR STANDARD I/0
#include<graphics.h> // HEADER FILE FOR GRAPHICS
MODE #include<dos.h>     // HEADER FILE FOR ENABLING
SOUND #include<conio.h> // HEADER FILE FOR CONSOLE
I/O #include<stdlib.h> // HEADER FILE FOR LIBRARY
FUNCTIONS union REGS i,o;
int initmouse();        // FUNCTION TO INITIALIZE MOUSE POINTER
void showmouseptr(); // FUNCTION TO SHOW POINTER
void restrictmouseptr(int,int,int,int); // FUNCTION TO RESTRICT
POINTER void getmousepos(int *,int *,int *); // TO GET POINTER
POSITION
void format();     // FUNCTION TO DRAW LAYOUT OF EVM
void graph();      // FUNCTION TO DISPLAY RESULT AS
GRAPH
void welcome();    // FUNCTION TO DISPLAY WELCOME MESSAGE
void boundry();
int vote1=0,vote2=0,vote3=0,vote4=0,vote5=0; // VARIABLES TO HOLD VOTES FOR
CANDIDATES
int
button,x,y;
void
main()
{     int gd=DETECT,gm;
    initgraph(&gd,&gm,"c:\\tc\\bgi"); // INITIALIZING GRAPHICS
    MODE randomize();
    boundry();
   welcome();              // CALLING WELCOME FUNCTION
   cleardevice();          // CLEARING THE SCREEN
   format();               // CALLING FORMAT
   FUNCTION
   showmouseptr();
   restrictmouseptr(0,0,675,435);    // RESTRICTING MOUSE POINTER WITHIN SCREEN
```

```
do
{
```

```
getmousepos(&button,&x,&y);
 if((button&1)==1&&x>475&&x<580&&y>250&&y<280)
{ break;}
else if((button&1)==1&&x>280&&x<380&&y>105&&y<125)
{ setcolor(RED);circle(270,115,5);
  sound(1200);
  delay(500);
  nosound();
  setcolor(BLACK);circle(270,115,5);
  vote1++; }
else if((button&1)==1&&x>280&&x<380&&y>155&&y<175)
{ setcolor(RED);circle(270,165,5);
  sound(1200);
  delay(500);
  nosound();
  setcolor(BLACK);circle(270,165,5);
  vote2++; }
else if((button&1)==1&&x>280&&x<380&&y>205&&y<225)
{ setcolor(RED);circle(270,215,5);
  sound(1200);
  delay(500);
  nosound();
  setcolor(BLACK);circle(270,215,5);
  vote3++; }
else if((button&1)==1&&x>280&&x<380&&y>255&&y<275)
{ setcolor(RED);circle(270,265,5);
  sound(1200);
  delay(500);
  nosound();
  setcolor(BLACK);circle(270,265,5);
  vote4++; }
```

```
else if((button&1)==1&&x>280&&x<380&&y>305&&y<325)
{ setcolor(RED);circle(270,315,5);
   sound(1200);
   delay(500);
   nosound();
   setcolor(BLACK);circle(270,315,5);
   vote5++; }
}              // END OF DO
while(1);

 cleardevice()
 ; initmouse();
 showmousep
 tr();
 boundry();
 graph();
 getch();
}                  // END OF MAIN FUNCTION
void boundry()
{
   setcolor(1+random(14));
   rectangle(0,0,635,475);
   setcolor(1+random(14));
   rectangle(3,3,632,472);
}
void welcome()
{ randomize();
  settextstyle(8,0,4);
  setcolor(1+random(1
  4));
  outtextxy(200,100,"WELCOM
  E"); delay(800);
  setcolor(1+random(14));
  outtextxy(250,160,"T
  O"); delay(800);
  setcolor(1+random(14
  ));
  outtextxy(50,220,"ELECTRONIC VOTING SYSTEM");
  delay(800);
  while(!kbhit())
  {  setcolor(1+random(14));
     outtextxy(50,400,"Press any key to continue.... ");
     delay(500);
     setcolor(BLACK);
     outtextxy(50,400,"Press any key to continue.... ");
     delay(500);
  }


}

void format()
{ setcolor(6);
   rectangle( 90,30,400,380);
   rectangle(
   87,27,403,383);
   settextstyle(0,0,5);
   outtextxy(140,40,"E V
   M"); line(90,80,400,80);
   settextstyle(8,0,2);
   outtextxy(100,100,"MEGHRA
   J");
   rectangle(95,100,250,130);
   arc(290,115,90,270,10);
   arc(370,115,270,90,
   10);
   line(290,105,370,10
   5);
```

```
line(290,125,370,12
5);
```

```
outtextxy(100,150,"DINESH");
rectangle(95,150,250,180);
arc(290,165,90,270,10);
arc(370,165,270,90,10);
line(290,155,370,155);
line(290,175,370,175);

outtextxy(100,200,"RAKESH");
rectangle(95,200,250,230);
arc(290,215,90,270,10);
arc(370,215,270,90,10);
line(290,205,370,205);
line(290,225,370,225);
outtextxy(100,250,"DEEPAK");
rectangle(95,250,250,280);
arc(290,265,90,270,10);
arc(370,265,270,90,10);
line(290,255,370,255);
line(290,275,370,275);

outtextxy(100,300,"ANUJAY");
rectangle(95,300,250,330);
arc(290,315,90,270,10);
arc(370,315,270,90,10);
line(290,305,370,305);
line(290,325,370,325);
```

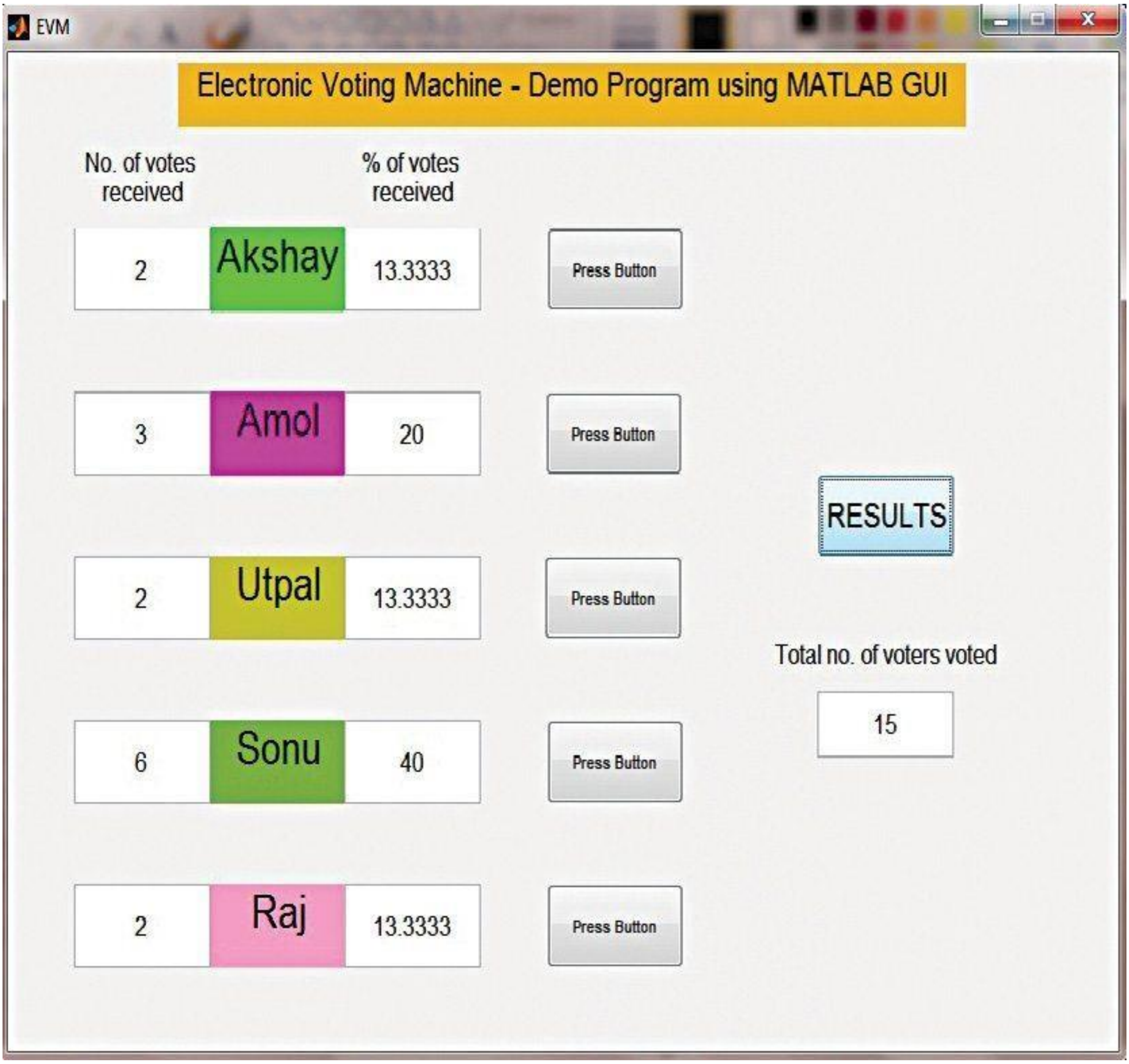
```
rectangle(475,250,580,280);
outtextxy(480,250,"RESULT
S");
```

```
   outtextxy(50,400,"Presented By:- EFY Enterprises Pvt Ltd");
}
void showmouseptr()
{
i.x.ax=1;
int86(0x33,&i,
&o);
}
void restrictmouseptr(int x1, int y1, int x2, int y2)
{    i.x.ax=7;
     i.x.cx=x1;
     i.x.dx=x2;
     int86(0x33,&i,
     &o); i.x.ax=8;
     i.x.cx=y1;
     i.x.dx=y2;
     int86(0x33,&i,
     &o);
}
void getmousepos(int *button, int *x, int *y)
{ i.x.ax=3;
    int86(0x33,&i,&o);
    *button=o.x.bx;
    *x=o.x.cx;
    *y=o.x.dx;
}
void graph()
{ outtextxy(200,100,"RESULTS(in % votes)");
    int
    candidate1=((vote1*100)/(vote1+vote2+vote3+vote4+vote5))
    ;                                                         int
    candidate2=((vote2*100)/(vote1+vote2+vote3+vote4+vote5))
    ;                                                         int
    candidate3=((vote3*100)/(vote1+vote2+vote3+vote4+vote5))
    ;                                                         int
```

```
candidate4=((vote4*100)/(vote1+vote2+vote3+vote4+vote5))
;                                                        int
candidate5=((vote5*100)/(vote1+vote2+vote3+vote4+vote5))
;
```

```
   setcolor(2);
   rectangle(100,300,130,300-
   candidate1);outtextxy(100,300,"ME");
   rectangle(200,300,230,300-
   candidate2);outtextxy(200,300,"DI");
   rectangle(300,300,330,300-
   candidate3);outtextxy(300,300,"RA");
   rectangle(400,300,430,300-
   candidate4);outtextxy(400,300,"DE");
   rectangle(500,300,530,300-
   candidate5);outtextxy(500,300,"AN");

   setcolor(1+random(14));
   rectangle(545,400,600,4
   30);
   outtextxy(550,400,"EXI
   T");

   do
{
getmousepos(&button,&x,&y);
 if((button&1)==1&&x>545&&x<600&&y>400&&y<430)
{ break;}
}             // END OF DO
while(1);



}
initmouse()
{
         ax=0;
         int86(0x
         33,&i,&
         o);
         return(o
        .x.ax);
}
```

12. SCREENSHOTS

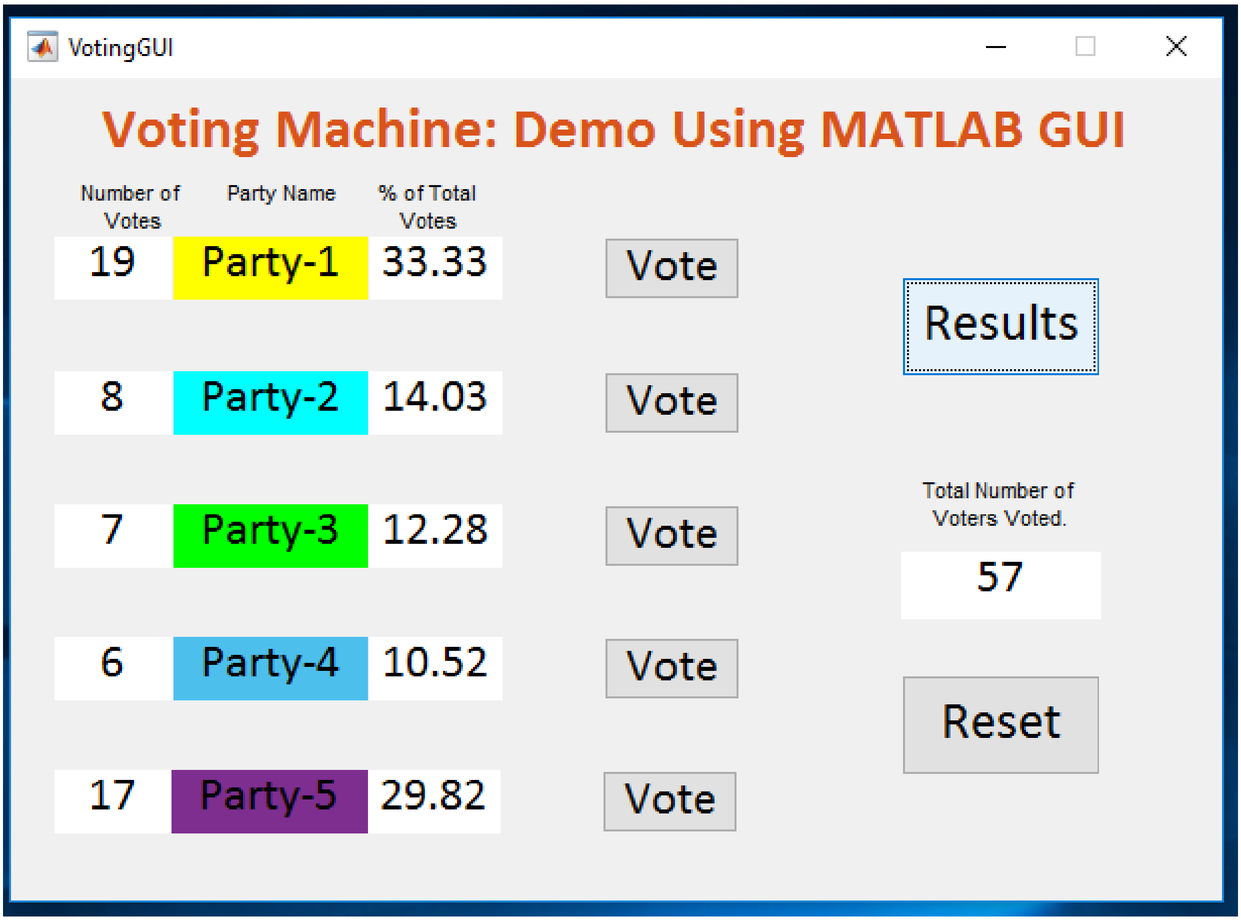


Electronic Voting Machine - Demo Program using MATLAB GUI

No. of votes received		% of votes received
2	Akshay	13.3333
3	Amol	20
2	Utpal	13.3333
6	Sonu	40
2	Raj	13.3333

Total no. of voters voted

15

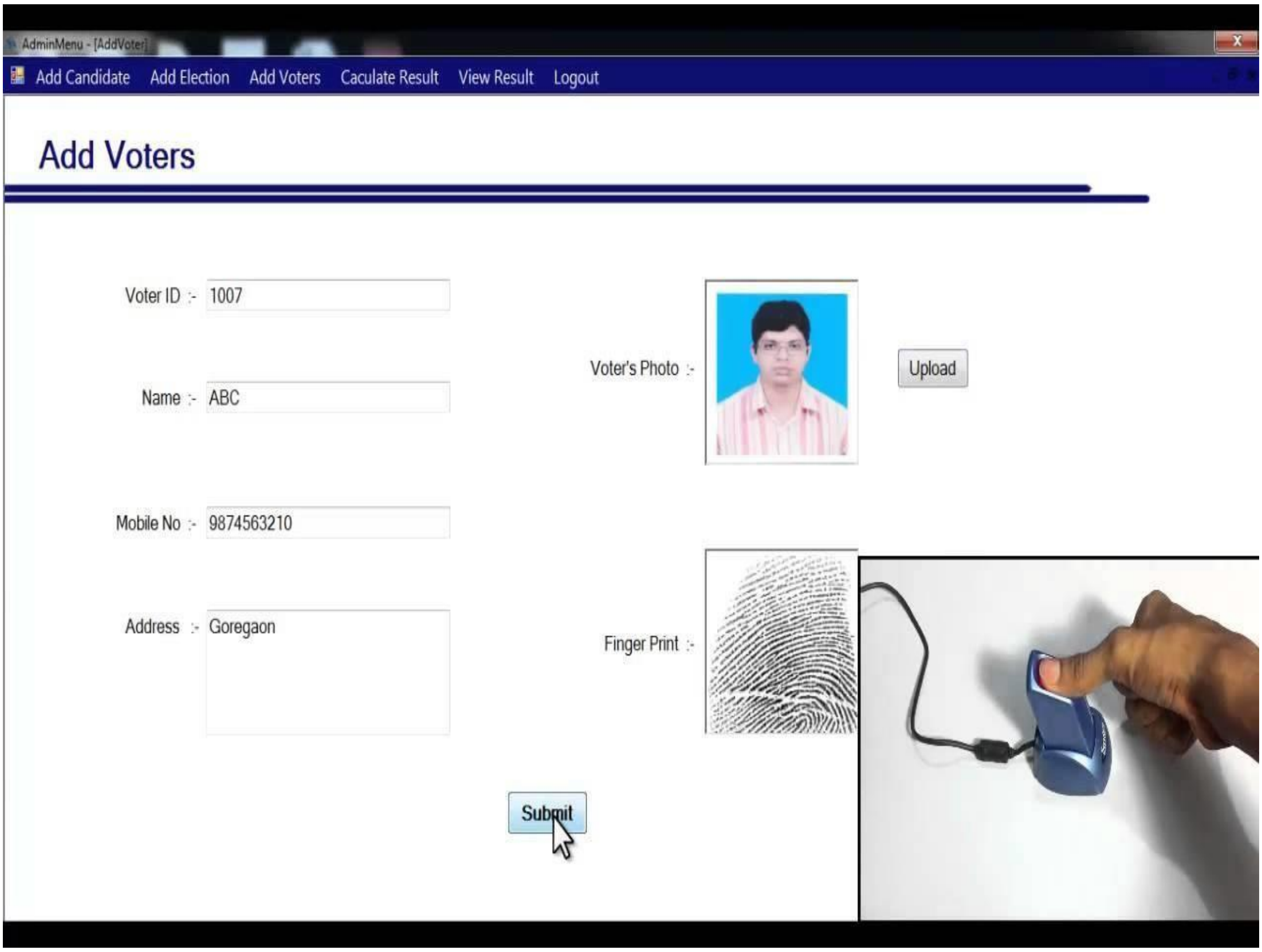


Voting Machine: Demo Using MATLAB GUI

Number of Votes	Party Name	% of Total Votes
19	Party-1	33.33
8	Party-2	14.03
7	Party-3	12.28
6	Party-4	10.52
17	Party-5	29.82

Total Number of Voters Voted.

57



Voter ID :- 1007

Name :- ABC

Mobile No :- 9874563210

Address :- Goregaon

13. CONCLUSION

The function of compiling, editing and processing data makes the voting system complicated and tedious process. Electronic voting system is an effort in the direction to use computer for the purpose of voting system. It is both user friendly as well as time saving. This is a foundation level. Emphasis has been given in this application to replicate all the process required in a traditional system. EVM system is self-sufficient with all the information required about the voters, candidate and the voters. Once the information is fed to the system, it can identify individual voters, their votes and then can press any key to return such as counting the votes has been simplified in this system. The basic function of the polling has been reduced to single key operation. A single polling manager can manage large number of voters.

LIMITATIONS:

An effort had been done to develop the system with the wide scope in view. Regardless of the number of application the software has its own limitation. The following are some of the notable limitation of the electronic voting system.

1. Security: - The system administrators can hacks the system at his will. He can change data output and hence can affect the result.
2. Voters education level: - Voters should be at least that much educated to enter the vote, i.e. press the required button. A totally computer illiterate has a very high chance of polling a wrong vote i.e. in favors of an undesirable candidate.
3. Specific voting system: - Since the system has been developed to handle the specific type of vote, i.e. only one candidate has to be chosen from the given list by one vote. This system cannot be used for other specific type of voting for e.g. electoral college voting (Indian resident Election) and conditional voting.
4. Basic system: - The EVM system is itself a basic and elementary form of software and hence cannot support other input devices as separate keyboard etc.

5. Confidentiality: - If the requirement of the voting is to keep the confidentiality of the votes, it is not possible since the system administrator would always be in a position to overview the votes of each and every individual at the time of voting.
6. The screen: - The screen is not too good in clarity and there are chances of mistakes during voting.

POSSIBLE IMPROVEMENTS:

9. In case of large data storage, the storing of data e.g. individual voter details would take a long time. Hence improvement applications can minimize the time.

10. Specific input devices (e.g. voting ads) could be an added advantage to the system.

11. Candidate name and number are only possible options in this application. Individual candidate terms can be added to the application.

12. Improvement can be done to make the application workable on net so that voting can be possible.

SCOPE:

9. Consumer survey
10. Large scale voting
11. Distance voting
12. Highly secured and error free result

BIBLIOGRAPHY

Text Books:

7. TURBO C++ by Robert Lafore
8. C++ Programming by E. BALAGURUSAMI
9. C++ by YASWANT KANITKAR

References:

www.joblagao.
com

www.electroni
csforu.com