SECURE ELECTRONIC VOTING

By

NC Anam Saud (GrpLdr)

CSUO Zeeshan Mubarak

GC Najeeb Ur Rehman

Supervisor:

LecWaseemIqbal

Co-Supervisor:

Asst Prof Ahmad RazaCheema

Submitted to the Faculty of Computer Science

National University of Sciences and Technology, Rawalpindi in partial fulfillment for the requirements of a B.E Degree in Computer Software Engineering

July 2013

CERTIFICATE

Certified that the contents and form of project report entitled “Secure Electronic Voting” submitted by 1) Anam Saud, 2) Zeeshan Mubarak, and 3) Najeeb Ur Rehman have been found satisfactory for the requirement of the degree.

Supervisor: _____________________

LecWaseemIqbal

ABSTRACT

Voting is indeed one of the most important features of a free democratic society. The right to exercise free will through voting is being practice for a long time now. Elections are a critical component of any democracy. Elections decide the fate of countries and their citizens, so while the introduction of e-voting may seem like a natural step in the modern world, it is one that should be taken with caution. Electronic voting thus referred to as e-voting is gaining more and more public interest.

Voting is done, not only for deciding a countries leadership but it is a method of voicing ones opinion hence it is practiced in many organizations including universities and other multinational companies to make various decisions.

E-voting in general refers to any kind of voting in electronic form. Thus e-voting includes voting by telephone as well as voting machines in voting booths. This project however deals with e-voting in the sense of voting with the use of an ordinary computer via internet.

It is argued that the ease with which voting can be performed will increase participation. However, security cannot be bargained with conveniences. Hence this Electronic system aims to provide a secure mechanism for electronic voting which not only increases the ease of voters but also will keep the system secure. This system will address the issues of integrity, anonymity, registration and authentication in particular. The need to reconcile identification and anonymity, on one hand, and verifiability and anonymity on the other hand will also be addressed.

As a result this thesis will discuss an end product that is a web based application for secure e-voting that is applicable to society elections in MCS. The project can take further enhancement in adding new platform e.g. android application.

DECLARATION

No portion of the work presented in this dissertation has been submitted in support of another award or qualification either at this institution or elsewhere.

DEDICATION

In the name of Allah, the Most Merciful, the Most Beneficent

Dedicated to our parents and mentors who have been a constant source of encouragement for us and to our teachers who have given us inspiration throughout our degree

ACKNOWLEDGEMENT

All praises for ALLAH who gave us the strength and determination and enlightened us with the requisite knowledge on portion of this subject to complete this project. We gratefully acknowledge the continuous guidance and motivation provided to us by our project advisor LecWaseemIqbal and Co-adviser Lec Ahmad RazaCheema without his personal supervision advice and help timely help completion would not have been possible

We deeply treasure the unparalleled support and forbearance that we received from our friends for their critical reviews and useful suggestions that helped us in completion of our degree project. We are deeply indebted to our family for their never ending patience and support for us and to our parents for the strength they gave us through their prayers

Table of Contents

List of Figures

Chapter 1: Introduction

1.1 BACKGROUND

Voting is an integral part of any society. The method mostly used for the voting purpose is a manual system. Where people have to physically go to a polling booth and cast their vote. This method is definitely slow and not very efficient. Human error is greatly involved in it and hence counting is greatly affected. Ever since we trace back the history of voting it has been manually conducted. However recently there are few countries which have started the use of casting a vote electronically either using a biometric system or other means of electronic voting.

Electronic voting thus referred to as e-voting is gaining more and more public interest. Voting is done, not only for deciding a countries leadership but it is a method of voicing ones opinion hence it is practiced in many organizations including universities and other multinational companies to make various decisions.

Electronic voting (also known as e-voting) is a term encompassing several different types of voting, embracing both electronic means of casting a vote and electronic means of counting votes.

Electronic voting technology can include punched cards, optical scan voting systems and specialized voting kiosks (including self-contained direct-recording electronic voting systems, or DRE). It can also involve transmission of ballots and votes via telephones, private computer networks, or the Internet.

In general, two main types of e-Voting can be identified:

E-voting which is physically supervised by representatives of governmental or independent electoral authorities (e.g. electronic voting machines located at polling stations);

Remote e-Voting where voting is performed within the voter's sole influence, and is not physically supervised by representatives of governmental authorities (e.g. voting from one's personal computer, mobile phone, television via the internet

1.1.1 Introduction

The current voting system in Pakistan is based on the cumbersome process that begins from voter’s registration to the casting of vote at some specific designated polling-booth. The key problem in this context is the hectic process of casting vote at only a specific place with the readily available information for this voter at both ends. According to a study a very large number of literate populations of Pakistan do not show up for voting due to this cumbersome process.

In this project we will develop a comprehensive scientific solution for electronic voting with the capability to cast vote from any location using internet. The scope of the project is a web application that is applicable to Software Society Elections in Military College of Signals (MCS).

In this project we have a website which is the client side and a server end that consists of multiple servers performing various tasks.

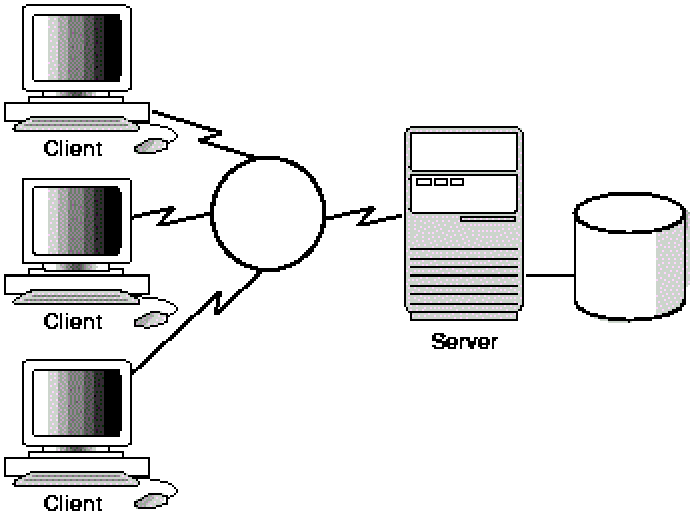


Fig:1a

1.2 PROBLEM DOMAIN

The unsolved problems of e-voting include the ability of malicious actors to intercept Internet communications, log in as someone else, and hack into servers to rewrite or corrupt code. While these are also big problems in e-commerce, if a hacker steals money, the theft can soon be discovered. A bank or store can decide whether any losses are an acceptable cost of doing business.

Voting is a different and harder problem. Lost votes aren't acceptable. And a voting system is supposed to protect the anonymity of a person's vote—quite unlike a banking or e-commerce transaction—while at the same time validating that it was cast accurately, in a manner that maintains records that a losing candidate will accept as valid and verified.

1.3 GOALS AND OBJECTIVES

1.3.1 Overall goals

To evaluate objectively the issues/constraints and capture requirements for deployment of electronic voting system.To streamline constraints/requirements and develop a customized infrastructure of electronic voting. Also evaluate/test the security, authenticity and efficiency of the proposed solution.

1.3.2 Objectives

The major objectives include that, only eligible voters are able to vote. No voter is permitted to vote more than once. No one should be able to determine the value of anyone else's vote. No one can duplicate a vote. No one can alter another person's vote without being detected.

1.4 DELIVERABLES

1.4.1 Software Deliverables

A Client side Web Application for electronic devices with internet connection which will allow voters to register their votes and vote on the voting day and see the results. A client program is required to perform a number of functions on behalf of the voter. These functions include exchanging messages with the servers, processing user input and performing the necessary cryptographic functions. Since it is a web Application there is no requirement for the voter to install an application.

A configured server which will perform the following tasks:

- Manage the database of the voters.
- Provide identification and authentication mechanisms.
- Ensure anonymity that is to prevent linking a specific voter to the vote that was cast.
- Collection of votes.
- Counting of votes.
- Making sure that the person who has cast the vote once cannot vote again.

1.4.2 Hardware Deliverables

N/A

Chapter 2: Literature Review

2.1INTRODUCTION

Electronic voting systems have been in use since the 1960swhen punched card systems were first used. Their first widespread use was in the USA. The newer optical scan voting systems allow a computer to count a voter's mark on a ballot. DRE voting machines which collect and tabulate votes in a single machine are used by all voters in all elections in Brazil and India, and also the United States. Internet voting systems have gained popularity and have been used for government elections and referendums in the United Kingdom, Estonia and Switzerland as well as in elections in Canada and party primary elections in the United States and France.

Nowadays Electronic vote has become more popular around the world. Some of the countries which uses electronic and vote on line are: United States, Brazil, Australia, Canada, Belgium, Germany, Romania, France, Venezuela, Philippines, The European Union, Switzerland, Italy, Norway, Romania and United Kingdom.

The first mechanized voting device was patented in the United States in 1892, and for nearly a century the United States was the only country using automated voting equipment. Since the 1980s, Brazil, India, the Nether- lands, the Philippines, Russia, and Venezuela have introduced e-voting systems. E-voting is not a panacea, but when properly implemented, it can be a useful tool for democratic elections.

Paper-based electronic voting system

Sometimes called a "document ballot voting system", paper-based voting systems originated as a system where votes are cast and counted by hand, using paper ballots but counted electronically.

Most recently, these systems can include an Electronic Ballot Marker (EBM), that allows voters to make their selections using an electronic input device, usually a touch screen system similar to a DRE.

Direct-recording electronic (DRE) voting system

The most recent configuration in the evolution of voting systems is known as direct recording electronic, or DRE's. They are an electronic implementation of the old mechanical lever systems. As with the lever machines, there is no ballot; the possible choices are visible to the voter on the front of the machine. The voter directly enters choices into electronic storage with the use of a touch-screen, push-buttons, or similar device. An alphabetic keyboard is often provided with the entry device to allow for the possibility of write-in votes. The voter's choices are stored in these machines via a memory cartridge, diskette or smart-card and added to the choices of all other voters.

In 1996, 7.7% of the registered voters in the United States used some type of direct recording electronic voting system.

Public network DRE voting system

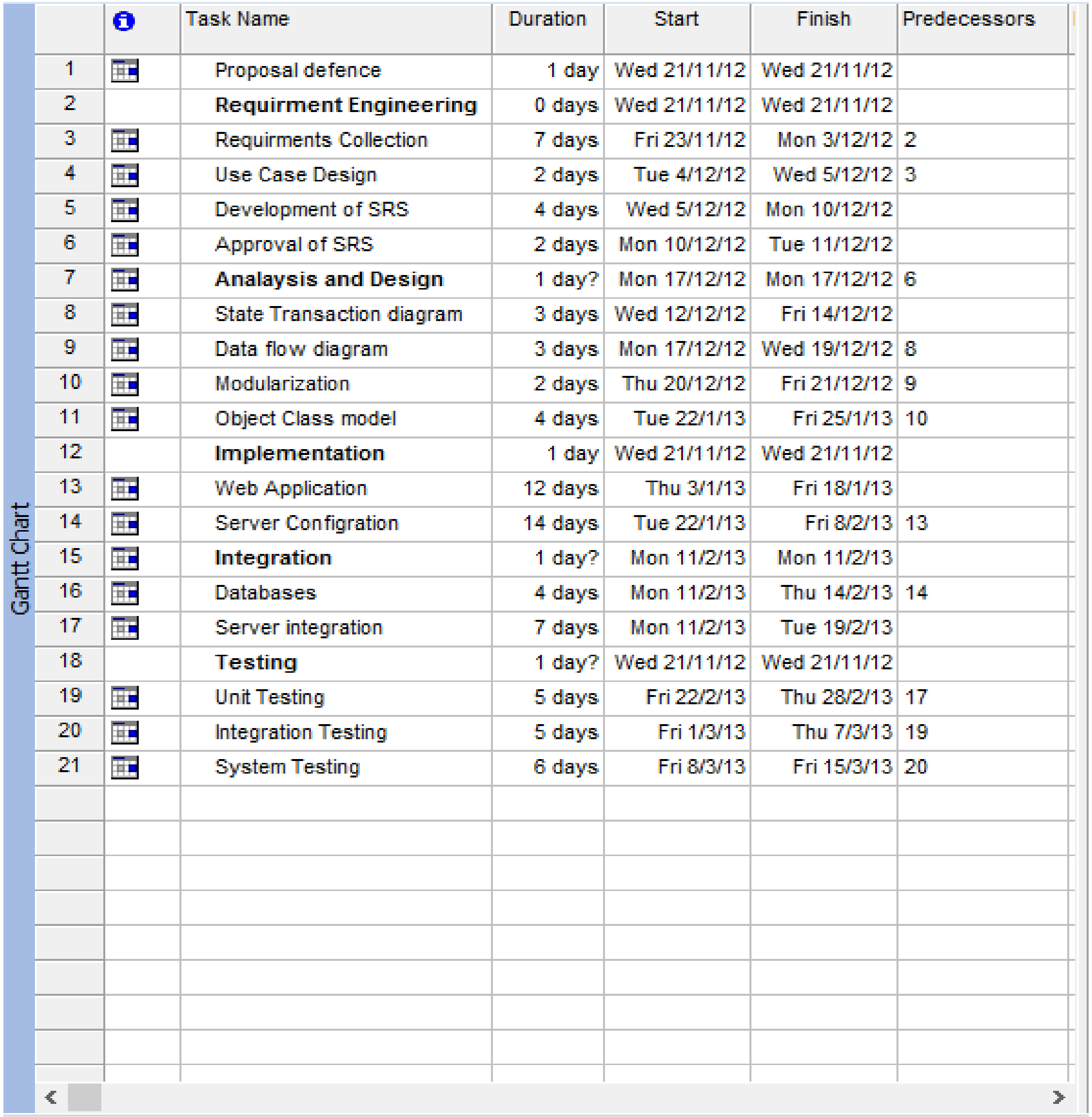
A Public Network Direct Recording Electronic (DRE) Voting System is an election system that uses electronic ballots and transmits vote data from the polling place to another location over a public network. Vote data may be transmitted as individual ballots as they are cast, periodically as batches of ballots throughout the Election Day, or as one batch at the close of voting. For purposes of the Guidelines, Public Network DRE Voting Systems are considered a form of DRE Voting System and are subject to the standards applicable to DRE Voting Systems.

Electronic voting systems may offer advantages disadvantages exist as well including the potential for flaws or weakness in any electronic component. Compared to other voting techniques. An electronic voting system can be involved in any one of a number of steps in the setup, distributing, voting, collecting, and counting of ballots, and thus may or may not introduce advantages into any of these steps.

Chapter 3: Project Plan

3.1 PROJECT OVERVIEW

3.1.1. Project Organization



	ℹ	Task Name	Duration	Start	Finish	Predecessors
1	📅	Proposal defence	1 day	Wed 21/11/12	Wed 21/11/12	
2		**Requirment Engineering**	0 days	Wed 21/11/12	Wed 21/11/12	
3	📅	Requirments Collection	7 days	Fri 23/11/12	Mon 3/12/12	2
4	📅	Use Case Design	2 days	Tue 4/12/12	Wed 5/12/12	3
5	📅	Development of SRS	4 days	Wed 5/12/12	Mon 10/12/12	
6	📅	Approval of SRS	2 days	Mon 10/12/12	Tue 11/12/12	
7	📅	**Analaysis and Design**	1 day?	Mon 17/12/12	Mon 17/12/12	6
8	📅	State Transaction diagram	3 days	Wed 12/12/12	Fri 14/12/12	
9	📅	Data flow diagram	3 days	Mon 17/12/12	Wed 19/12/12	8
10	📅	Modularization	2 days	Thu 20/12/12	Fri 21/12/12	9
11	📅	Object Class model	4 days	Tue 22/1/13	Fri 25/1/13	10
12		**Implementation**	1 day	Wed 21/11/12	Wed 21/11/12	
13	📅	Web Application	12 days	Thu 3/1/13	Fri 18/1/13	
14	📅	Server Configration	14 days	Tue 22/1/13	Fri 8/2/13	13
15	📅	**Integration**	1 day?	Mon 11/2/13	Mon 11/2/13	
16	📅	Databases	4 days	Mon 11/2/13	Thu 14/2/13	14
17	📅	Server integration	7 days	Mon 11/2/13	Tue 19/2/13	
18		**Testing**	1 day?	Wed 21/11/12	Wed 21/11/12	
19	📅	Unit Testing	5 days	Fri 22/2/13	Thu 28/2/13	17
20	📅	Integration Testing	5 days	Fri 1/3/13	Thu 7/3/13	19
21	📅	System Testing	6 days	Fri 8/3/13	Fri 15/3/13	20

Fig: 3a

3.1.2 Work Breakdown Structure

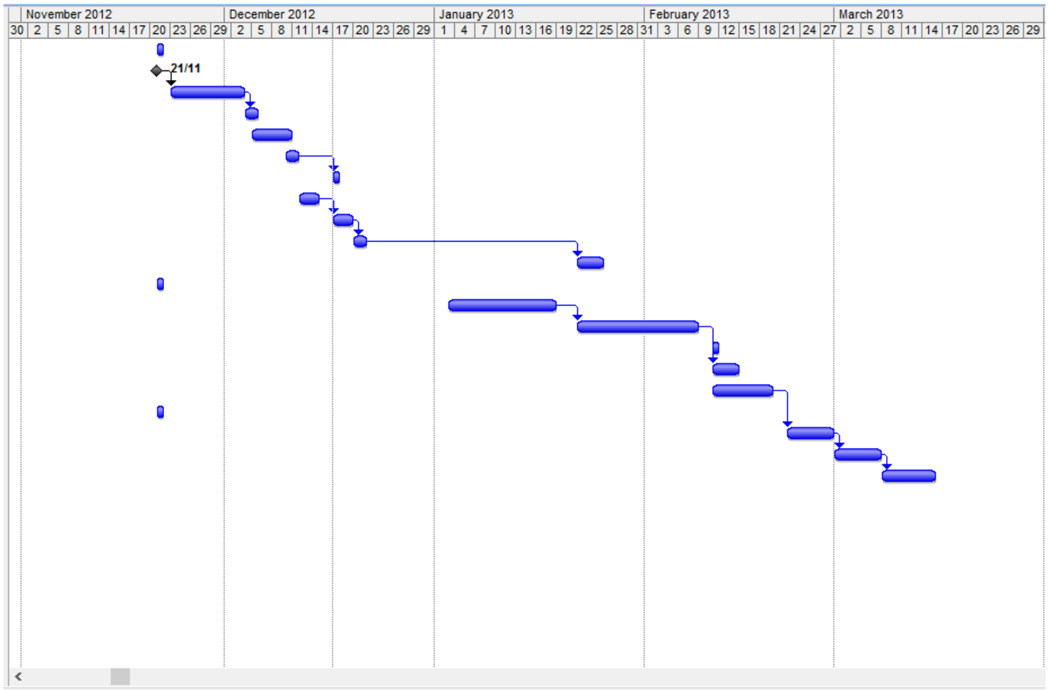
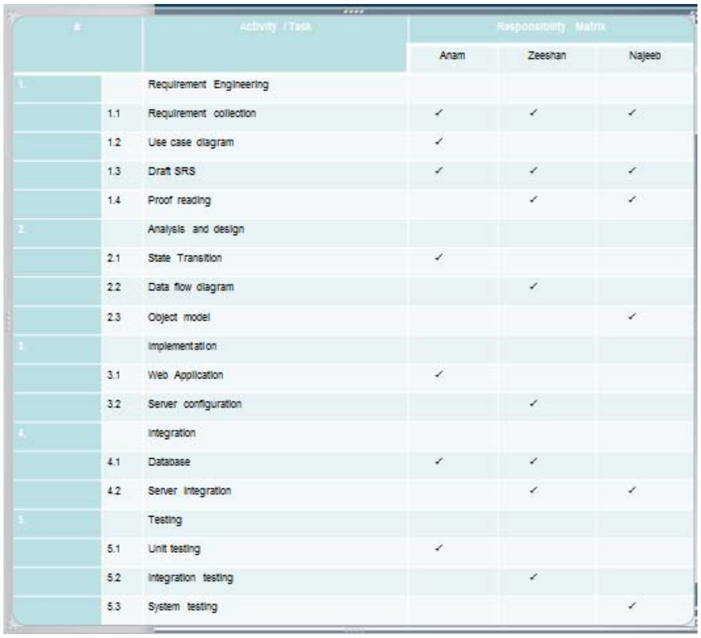


Fig: 3b

3.1.3 Roles and Responsibilities



#		Activity / Task	Responsibility Matrix		
			Anam	Zeeshan	Najeeb
1.		Requirement Engineering			
	1.1	Requirement collection	✓	✓	✓
	1.2	Use case diagram	✓		
	1.3	Draft SRS	✓	✓	✓
	1.4	Proof reading		✓	✓
2.		Analysis and design			
	2.1	State Transition	✓		
	2.2	Data flow diagram		✓	
	2.3	Object model			✓
3.		Implementation			
	3.1	Web Application	✓		
	3.2	Server configuration		✓	
4.		Integration			
	4.1	Database	✓	✓	
	4.2	Server integration		✓	✓
5.		Testing			
	5.1	Unit testing	✓		
	5.2	Integration testing		✓	
	5.3	System testing			✓

Fig: 3c

Chapter 4: System Requirements (SRS)

4.1 PURPOSE

Voting is indeed one of the most important features of a free democratic society. The right to exercise free will through voting is being practice for a long time now. Elections are a critical component of any democracy. Elections decide the fate of countries and their citizens, so while the introduction of e-voting may seem like a natural step in the modern world, it is one that should be taken with caution. Electronic voting thus referred to as e-voting is gaining more and more public interest. Voting is done, not only for deciding a countries leadership but it is a method of voicing ones opinion hence it is practiced in many organizations including universities and other multinational companies to make various decisions.

E-voting in general refers to any kind of voting in electronic form. Thus e-voting includes voting by telephone as well as voting machines in voting booths. This project however deals with e-voting in the sense of voting with the use of an ordinary computer via internet.

It is argued that the ease with which voting can be performed will increase participation. However, security cannot be bargained with conveniences. Hence this Electronic system aims to provide a secure mechanism for electronic voting which not only increases the ease of voters but also will keep the system secure. This system will address the issues of integrity, anonymity, registration and authentication in particular. The need to reconcile identification and anonymity, on one hand, and verifiability and anonymity on the other hand will also be addressed.

This document specifies the software requirements of *Secure Electronic System*. The scope of our end product is a web based application for secure e-voting that is applicable to society elections in MCS. The project can take further enhancement in adding new platform e.g. android application.

4.1.1 Document Conventions

Simple paragraphs Font Style: Calibri

	Font Size: 12 Alignment: Justified
Headings:	
Heading 1:	Font Style: Times (Heading 1) Font Size: 18 Alignment: Left Align Bold
Heading 2:	Font Size: 14
Heading 3:	Font Size: 12

4.2 AUDIENCE

Audience for this SRS template is:

- Supervisors
- Development Team
- Project Coordinator
- Faculty Members of respected evaluation panel

Above mentioned audience is required to have understanding of:

- The Voting Process
- Database Servers
- Client/Server Architecture
- Cryptography
- Network Security
- Shared Group Key

4.3 PRODUCT SCOPE

The current voting system in Pakistan is based on the cumbersome process that begins from voter's registration to the casting of vote at some specific designated polling-booth. The key problem in this context is the hectic process of casting vote at only a specific place with the readily available information for this voter at both ends. According to a study a very large number of literate populations of Pakistan do not show up for voting due to this cumbersome process.

In this project we will develop a comprehensive scientific solution for electronic voting with the capability to cast vote from any location using internet. The scope of the project is a web application that is applicable to Software Society Elections in Military College of Signals (MCS). Once it is successful only then future enhancements can be made and that is to address the need of electronic voting system in Pakistan, especially emphasizing the security aspects of such infrastructure based on the scalability of our project. One more enhancement can be made in the project that is to add another platform for voting i.e. android application.

Secure Electronic Voting (SEV) includes a tested, documented, and functional system which no longer depends on voters going to a polling booth to vote and paper based voting.

4.4 REFERENCES

- Project Synopsis
- Project Proposal

4.5 DESCRIPTION

4.5.1 Product Prospective

The current voting system is cumbersome and inefficient. It lacks transparency. Voters have to physically go to the polling booths for voting and many voters just avoid this physical exercise. Calculation of votes is also done manually.

However there are many disadvantages of a manual voting system:

Ballot Design

Though the use of a printed ballot remains the most popular voting method, issues regarding the location of candidate names along with voting booth mistakes such as miscast votes plague manual elections. Various ballot designs such as the punch card style add to voting booth issues due to close proximity of candidate names and difficulty in changing votes.

Ballot Count

The results of manual elections come into question due to several factors regarding human error or corrupt election practices. As seen in recent high-profile national elections, the miscount of cast votes along with illegal vote suppression practices cast doubt on the reliability of the manual election process. Additionally, questionable relationships between local voting authorities and major political parties continue to cloud manual elections.

Voter Error

The use of the punch card type manual ballot opens opportunities for voter error such as unintentional and unclear candidate selection due to confusing ballot organization. In the case of unintentional candidate selection, the public faces vote disqualification as they cannot make a second selection on a single election ballot. Additionally, voters also face vote nullification as the use of the write-in section on some ballot cards is prohibited for making ballot corrections.

Lack of Mobility

The voters have to physically go to the polling booth to vote. Many voters avoid this physical exercise. Hence there is less participation and low voters urn over.

In this project we propose to develop a scientific solution for electronic voting. With the capability to cast vote from any location, using electronic devices. It will provide authentication, Integrity and confidentiality of the votes. This is a web application to vote securely on internet from any location.

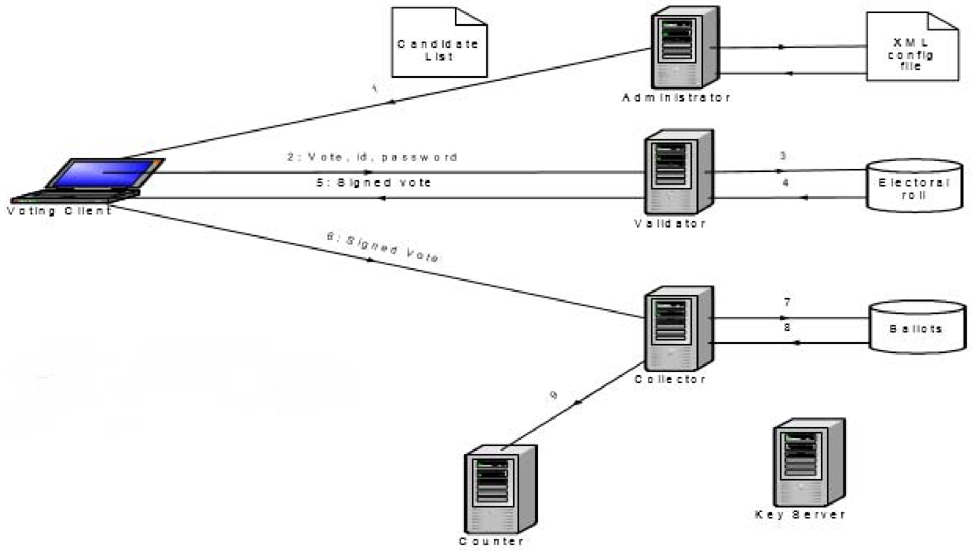


Fig: 4a

4.5.2 Product Functions

- A web interface
- Voter (user) registration in a database
- Voter identification
- Voter authentication
- Vote Casting
- Encrypting the vote
- Anonymize the votes
- Collection of vote
- Counting of the votes
- Display of results

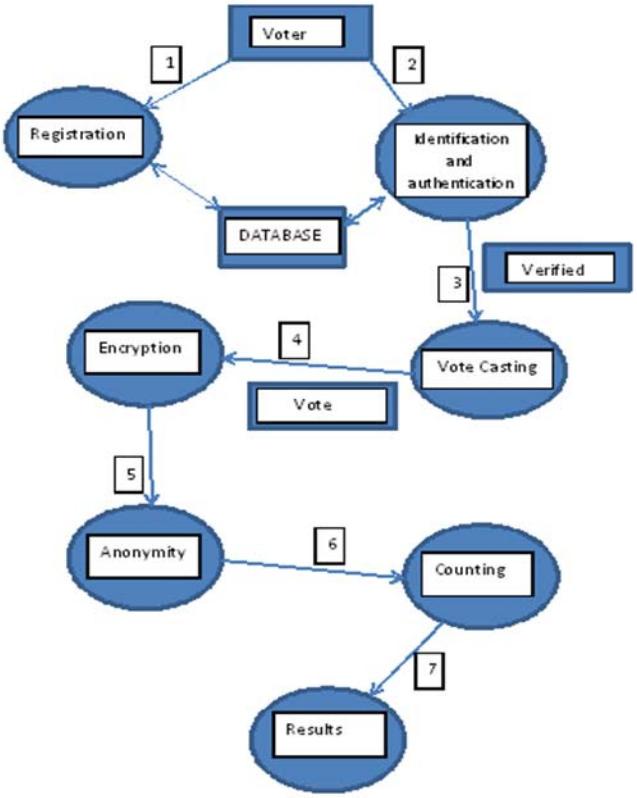


Fig: 4b

4.5.3 User Classes and Characteristics

Secure Electronic Voting (SEVS) is a product to facilitate the general public in the election process, by allowing them to cast their votes from any location where there is any electronic device with an internet connection. Apart from them, the body that is conducting elections can also get an ease in their responsibilities by using this project.

4.5.4 Operating Environment

Secure Electronic Voting (SEVS) provides its interface to the clients on a web browser on any computer and any operating system with internet available. On the server end, the web application will be using PHP to respond to the client request. The database will be configured using MySQL.

4.5.5 Design and Implementation Constraints

An internet connection and a computer are mandatory for the client.The server with the storage capacity enough to store user data in the database. The server must also be efficient enough to respond to multiple client requests at one time. Server should have the capability to run the latest versions of PHP and MySQL. Secure channels to ensure privacy and secrecy.
Anonymous channels to prevent linking a specific voter to the vote that was cast.

4.5.6 User Documentation

The user documentation consists of a user manual.

4.5.7 Assumptions and Dependencies

The voter completely trusts the voting system and mechanism. The voter has studied the user guide and seen the demonstration video thus making himself fully aware of the process of voting on *Secure Electronic Voting System*. The voter has an electronic device i.e. a desktop computer, laptop, tablet or an android phone. The electronic device has an internet connection available and a web browser installed. This internet connection is working properly.
The voter is well aware of the process of registration on a form on internet. The user has not shared his voting id and e-mail password with anyone. The voter is not under any political or external pressure. He has the complete right to practice free will.

4.6 EXTERNAL INTERFACE REQUIREMENT

4.6.1 User Interfaces

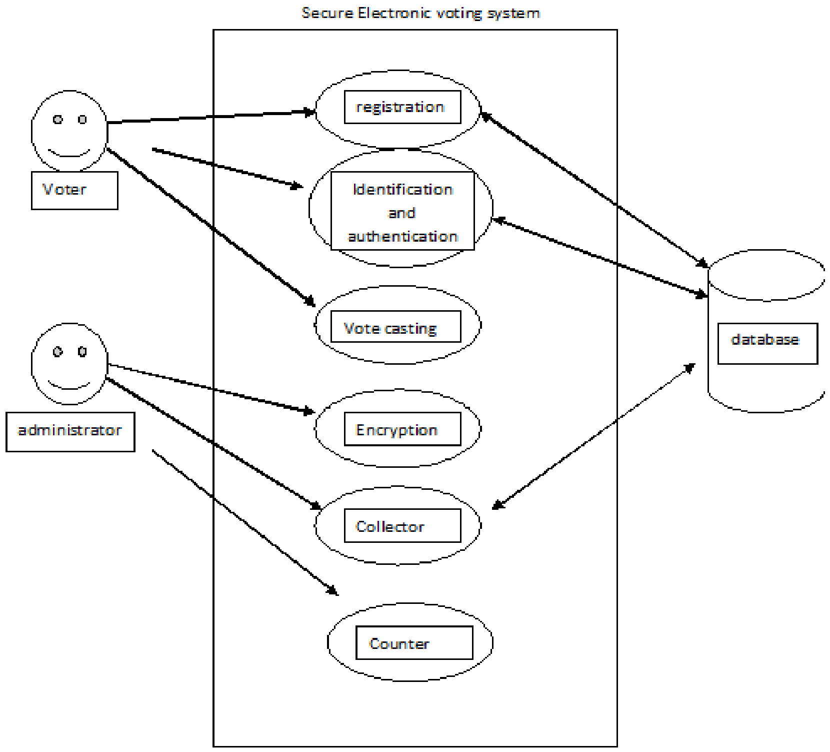


Fig: 4c

- Home page
- Help in every page and step
- Registration form
- Authentication page
- Voting ballot
- Result display page
- Messages

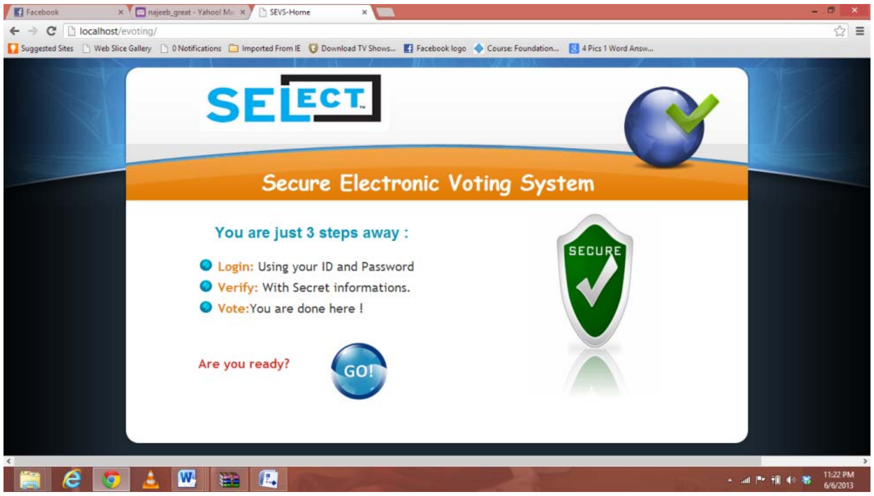
  Success message i.e. vote counted successfully, registered successfully

  Error message i.e. Invalid voter already voted.

Confirmation message i.e. are you sure you want to vote 'xyz'? yes|no;

- Home screen can be accessed by anyone.
- Authentication page will be available to only registered users.
- Ballot page will be available to only authenticated user.
- Results can be viewed by everyone.

- Administrator page can only be seen by the administrator.



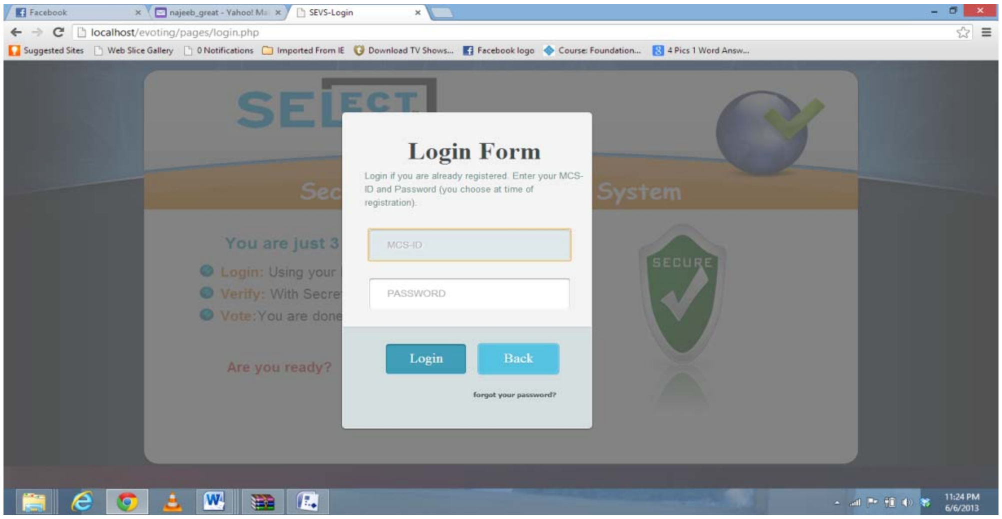
Secure Electronic Voting System

You are just 3 steps away :

- Login: Using your ID and Password
- Verify: With Secret informations.
- Vote:You are done here !

Are you ready?

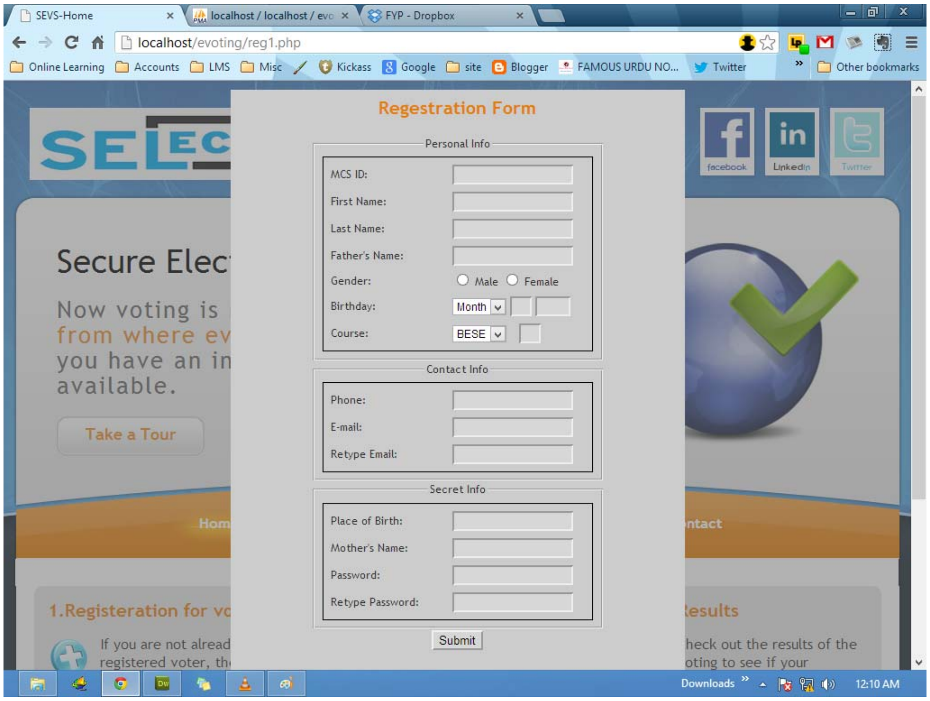
Fig: 4d



Login Form

Login if you are already registered. Enter your MCS-ID and Password (you choose at time of registration).

Fig: 4e



Regestration Form

Personal Info

MCS ID:	
First Name:	
Last Name:	
Father's Name:	
Gender:	○ Male ○ Female
Birthday:	Month
Course:	BESE

Contact Info

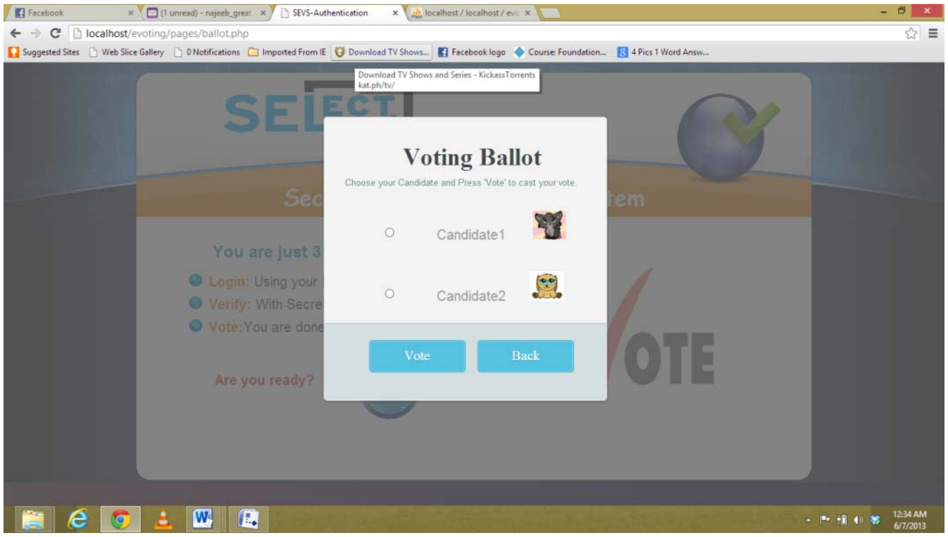
Phone:	
E-mail:	
Retype Email:	

Secret Info

Place of Birth:	
Mother's Name:	
Password:	
Retype Password:	

Submit

Fig: 4f



Voting Ballot

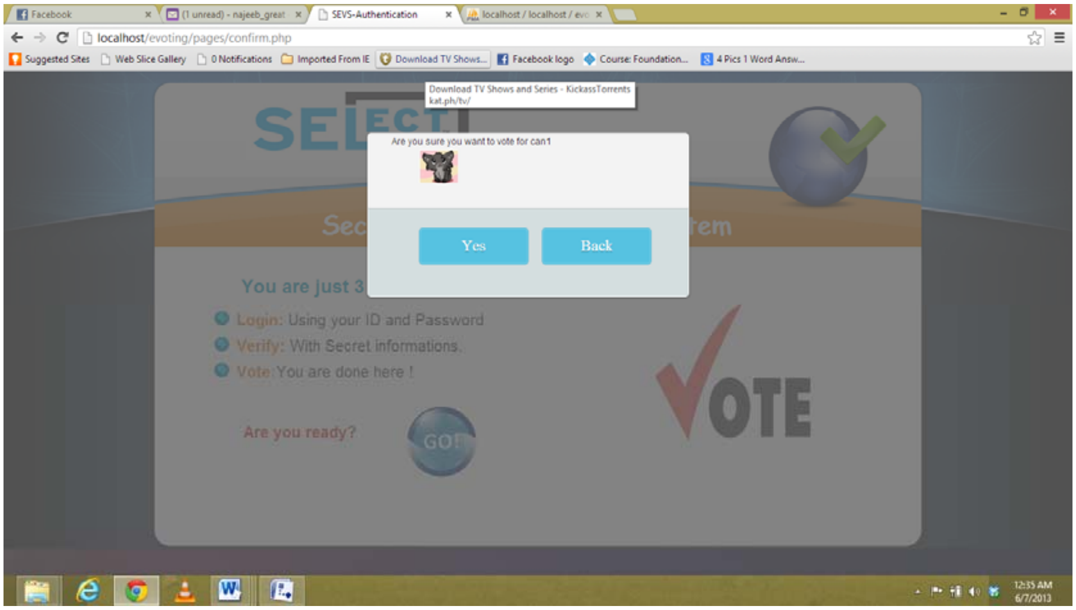
Choose your Candidate and Press 'Vote' to cast your vote.

○ Candidate1

○ Candidate2

Vote Back

Fig: 4g



localhost/evoting/pages/confirm.php

Are you sure you want to vote for can1

You are just 3

Login: Using your ID and Password

Verify: With Secret informations.

Vote: You are done here !

Are you ready?

Fig: 4h

4.6.2 Hardware Interfaces

Client can vote using any computer device. However at server end there will be a powerful server needed. The server machine with following specifications is suggested to be used:

1x Processor (Intel core i3 above)

1 x 4 GB RAM

1 x 500GB hard disk

4.6.3 Software Interfaces

Apache server 2.2 or later

PHP 5.3 or later

MySQL 5.5 or later

HTTP request/ respond communication b/w client and server

User registration data in database

Data retrieval from the database

Authentication credentials to the database

Communications Interfaces

A web browser with java script enabled.

Registration form to register users in database.

Http and PHP requests to the server.

Encrypted communication for sending votes.

Modification of IP headers to provide anonymity.

4.7 SYSTEM FEATURES

4.7.1 Transparency

Our system will be more open to public scrutiny without compromising the security.

4.7.1.1 Description and Priority

Secure electronic voting will be a system which will provide the facility to analyze and monitor the whole process without compromising the basic security requirements.
It is one of the high priorities of our system.

4.7.1.2 Stimulus/Response Sequences

If any of the stake holder wishes to view the system he will be shown the counters which are maintained throughout the process thus no extra or illegal votes can be added to the count as the number of total legal voter is also known and displayed over the system.

4.7.1.3 Functional Requirements

In order to provide the transparency the system must be able to maintain the counters from registration phase to counting phase and they must verify each other.

4.7.2 Anonymity

4.7.2.1Description and Priority

Secure electronic voting will be a system which will provide the anonymity to the voter so that no one will be able to find that which voter have voted to which candidate. It is one of the basic requirements of any voting system therefore it is high priority feature of our system

4.7.2.2 Stimulus/Response Sequences

Once the voter will cast its vote then the vote will be encrypted and sent to the server where a special function will be applied to randomize the vote so that no one can find out that whose vote is this.

4.7.2.3 Functional Requirements

In order to provide the anonymity the system must have a randomization function at the server end which must be applied before the counting or decrypting of the vote.

4.7.3 Mobility

4.7.3.1 Description and Priority

Secure electronic voting will be a system which will provide the mobility to the voter so that a registered voter can vote for the elections from anywhere with a computer having an internet connection. It is one of the basic requirement of our voting system therefore it is high priority feature of our system

4.7.3.2 Stimulus/Response Sequences

If the voter wishes to cast its vote he will use his id on the web application to vote for the candidate he wants.

4.7.3.3Functional Requirements

In order to provide the mobility the system must have a web application which is accessible over the internet.

4.8 OTHER NONFUNCTIONL REQUIREMENT

4.8.1 Performance Requirements

As the project will be running on real time basis, so system should be fast enough that there is no delay in process. Connection between clients and the server and data transfer as well as synchronization try not to irritate the users with delays.

4.8.2 Safety Requirements

- Backup database
- UPS and generators
- Prevent any physical access to the server by any unauthorized user.

4.8.3 Security Requirements

Secure channels to ensure privacy and secrecy. In e-voting a communication channel is secure if it ensures secrecy through the use of symmetric or asymmetric key encryption;

and ensures data integrity by means of digital signatures, message Digests or message authentication codes (MACs).

Hence this system will provide these security mechanisms.

4.8.4 Software Quality Attributes

- **Accuracy:** Can't alter a vote, or discount a validated vote from the final tally or include an invalid vote in the final count.
- **Privacy (un-traceability):** Prevents any agency from linking a specific voter with the ballot he cast, and does not allow voters to prove the way they voted.
- **Soundness or robustness:** The system should ensure that the election process is not affected by illegal behavior or faulty procedures.
- **Mobility:** Voters should be able to cast their votes from anywhere without geographical constraints. The system should also be available and accessible during the polling phase.
- **Integrity:** System is tamper proof, and data integrity ensures that data has not been modified during transit.
- **Convenience:** a system should allow users to cast their vote easily, quickly and with minimal instruction. Convenience is translated into usability and reliability requirements. The challenge however is to balance convenience and security.
- **Fairness:** Early results should displayed, so that voters are not be influenced by intermediate results.

4.9 OTHER REQUIREMENTS

Anonymous channels

The TCP/IP protocol suite an Internet packet carries the IP addresses of the source and the destination machines. This information is particularly important in reliable communication especially at the transport level. The TCP protocol establishes a reliable channel between processes running on different machines, and therefore between client and server. In e-voting systems the awareness by the destination machine of the IP address of the source machine, may compromise the anonymity requirement. Even if a client sends a vote without any identifying information the identity of the voter can be extracted from the IP address. Voting protocols seek to overcome this problem by implementing an anonymous channel whereby a server can reliably and securely receive messages but cannot determine the identity of the sending machine.

Interfacing

Some researchers consider that the most important issue in e-voting is interfacing. Avoiding bias in voting procedures has also become one of the prime issues in e- voting implementation. Hence a comprehensive and easy to use user interface is also an important requirement of this system. So that voters of different backgrounds and languages can use the system without getting confused and stuck.

Chapter 5: System Design

5.1 INTRODUCTION

Voting is indeed one of the most important features of a free democratic society. The right to exercise free will through voting is being practice for a long time now. Elections are a critical component of any democracy. Elections decide the fate of countries and their citizens, so while the introduction of e-voting may seem like a natural step in the modern world, it is one that should be taken with caution. Electronic voting thus referred to as e-voting is gaining more and more public interest.

E-voting in general refers to any kind of voting in electronic form. Thus e-voting includes Voting by telephone as well as voting machines in voting booths. This project however deals with e voting in the sense of voting with the use of an ordinary computer via internet.

Electronic system aims to provide a secure mechanism for electronic voting which not only increases the ease of voters but also will keep the system secure. This system will address the issues of integrity, anonymity, registration and authentication in particular. The need to reconcile identification and anonymity, on one hand, and verifiability and anonymity on the other hand will also be addressed.

5.1.2 Purpose

The purpose of this document is to accurately depict all the necessary information for the successful development of SEVS. It is the first version of the document; hence there is no revision or release number. The document can be altered to adjust any change in requirements or constraints. In case of any change, a new version of the SRS shall be produced. This process shall continue until a document is made which satisfies the conditions set by the supervisors and the faculty panel.

The SEVS shall provide the voter facility to register himself for online voting by the use of internet from anywhere and the caste his/her vote and view the results.

This SRS covers the SEVS including both software and hardware requirements as well as the basic features of the system. This document describes the overall system functionality describing the basic characteristics of the system, the scope of the project and the main features of the system. It provides an explanation as to what are the objectives of the system, along with certain assumptions and limitations regarding the implementation of the product.

5.1.3 Project Scope

The current voting system in Pakistan is based on the cumbersome process that begins from voter's registration to the casting of vote at some specific designated polling-booth. The key problem in this context is the hectic process of casting vote at only a specific place with the readily available information for this voter at both ends. According to a study a very large number ofliterate populations of Pakistan do not show up for voting due to this cumbersome process. In this project we will develop a comprehensive scientific solution for electronic voting with the capability to cast vote from any location using internet. The scope of the project is a web application that is applicable to Software Society Elections in Military College of Signals (MCS).

Once it is successful only then future enhancements can be made and that is to address the need of electronic voting system in Pakistan, especially emphasizing the security aspects of such

infrastructure based on the scalability of our project. One more enhancement can be made in the

project that is to add another platform for voting i.e. android application.

Secure Electronic Voting (SEV) includes a tested, documented, and functional system which no longer depends on voters going to a polling booth to vote and paper based voting.

5.2 DESIGN CONSIDERATION

5.2.1 Assumptions and Dependencies

An internet connection and a computer are mandatory for the client. The server which with the storage capacity enough to store user data in the database. The server must also be efficient enough to respond to multiple client requests at one time. Server should have the capability to run the latest versions of PHP and MySQL. Secure channels to ensure privacy and secrecy. Anonymous channels to prevent linking a specific voter to the vote that was cast.

5.2.2System Overview of SEVS

This is the first version of this system. The SRS describes the specifications of the entire project rather than a particular component. The following diagram describes the main components of SEVS. Namely,

- Central Management Server Application
- Web Interface

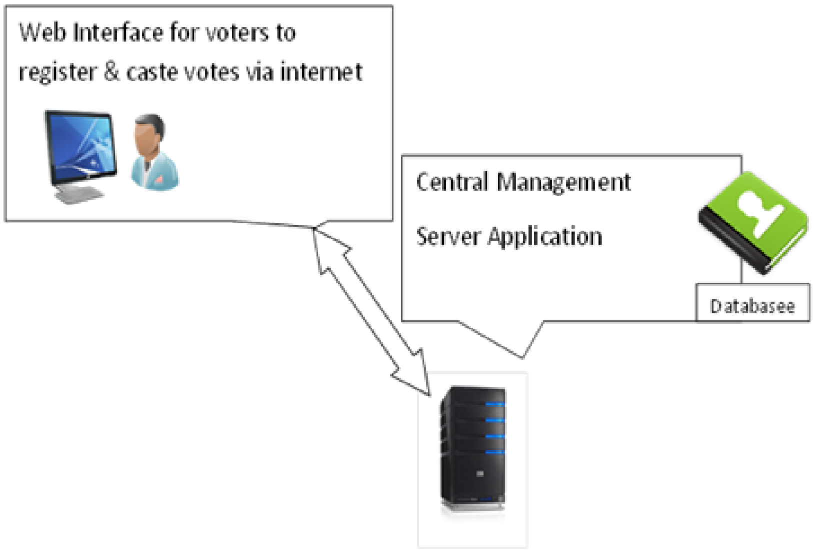

Web Interface for voters to register & caste votes via internet
Central Management Server Application
Databasee

Fig: 5a

5.3 SOFTWARE ARCHITECTURE DESIGN

5.3.1 SEVS System Architecture

SEVS will have a web interface which will allow the voter to register, vote and view the results via internet at any place. A candidate list shall be shown to the registered voter from which he will choose his option. the vote will be encrypted and sent to the server which shall anonymize as well as shuffles the sequence and send it to the counter which shall count the votes and update the database from where results can be retrieved.

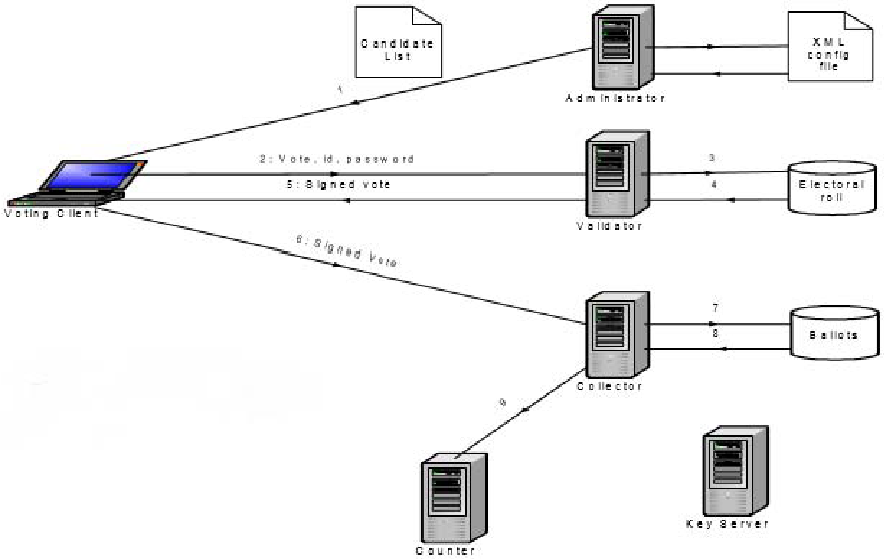


Fig: 5b

5.3.2 Architectural Strategies

The architecture is simple and flexible which allows adding more algorithms on the server end for enhanced security.Separation of tasks in layers allows for flexibility in implementation and allows the use of iterative approach. It allows building product in modules with clear and defined boundaries and defined interfaces/data to share.

The architectural pattern selected for this project is Model view Controller (MVC). MVC is simply an implementation of separation of Concerns.MVCis a design pattern. A reusable "recipe" for constructing your application. Generally, you don't want your user interface code and data access code to be mixed together, it makes changing either one more difficult. By placing data access code into a "Model" object and user interface code into a "View" object, you can use a "Controller" object to act as a go-between, sending messages/calling methods on the view object when the data changes and vice versa.

Benefits

The benefits of using MVC are it reduces code complexity. The code can be reuse. It also increases flexibility.

5.3.4 Data Flow diagram

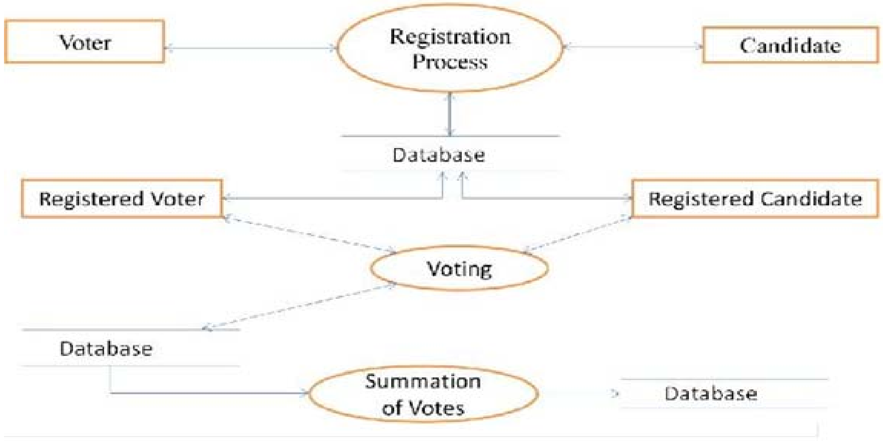


Fig:5c

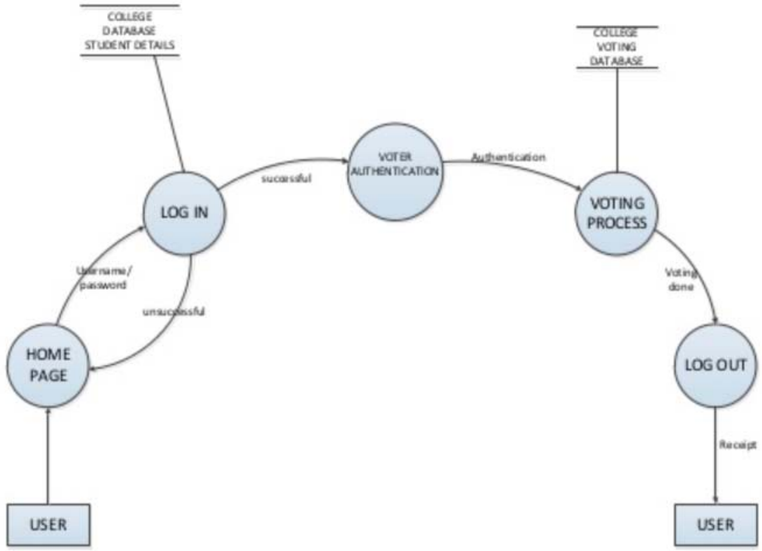


Fig: 5d

5.4 DETAIL SYSTEM DESIGN

5.4.1 Logical View

Logical view contains class diagram and use case diagram. It describes the static behavior of system.

Use-case Diagram

Figure 2 below shows the interaction of user with the system.

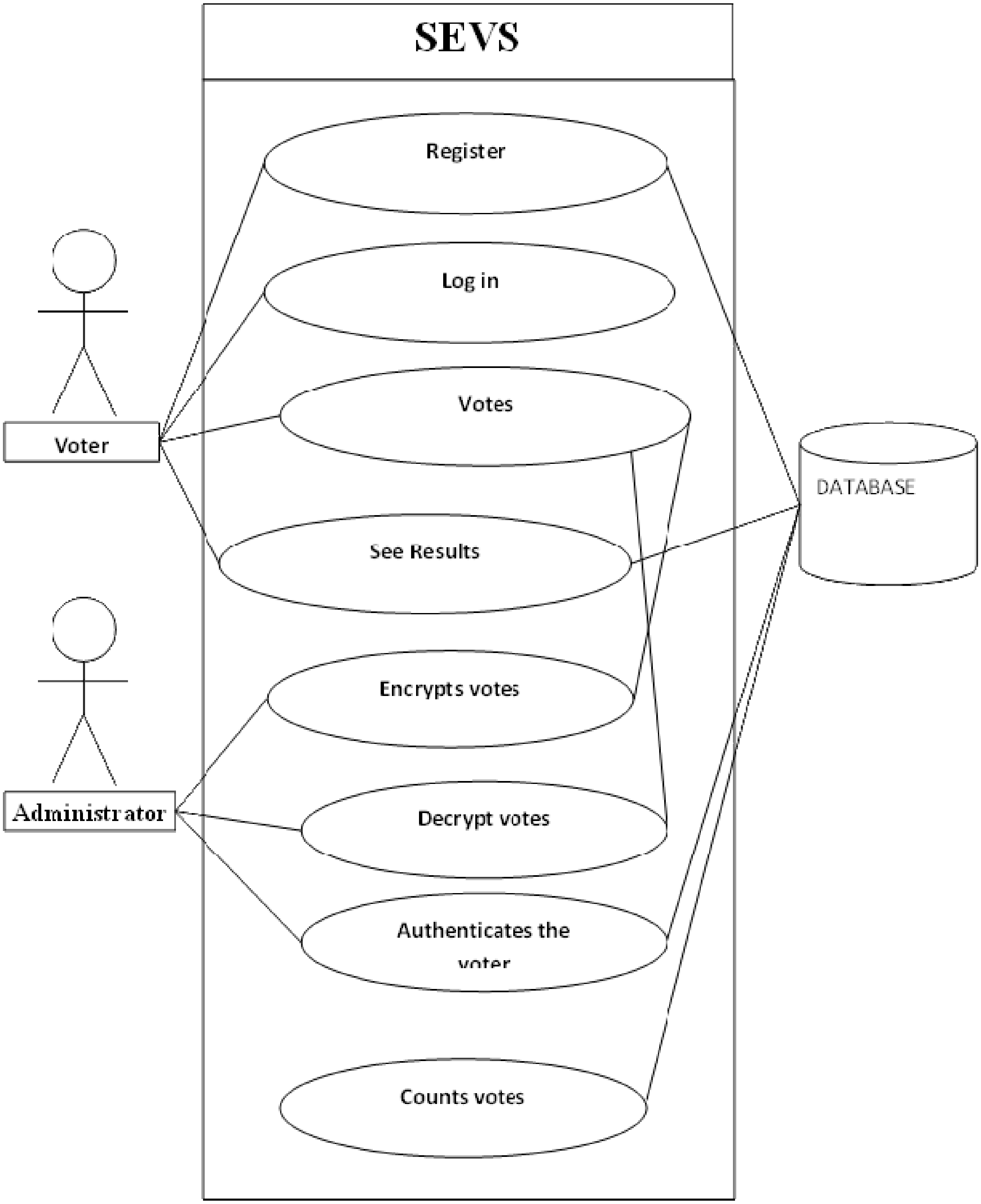

SEVS
Register
Log in
Votes
See Results
Encrypts votes
Decrypt votes
Authenticates the voter
Counts votes
Voter
Administrator
DATABASE

Fig: 5e

UC-1

Name: Register

Identifier: UC-1

Actors: Voter

Goals: The goal is to register the voter

Preconditions: Voter is not currently registered

Trigger: User clicks on the button

Related use cases: UC-2

Steps: user clicks on the button to register.

A form will be displayed.
Voter will fill in the form.
Voter will submit the form.
Form will be validated.
Database is updated.

Post conditions: voter is registered

UC-2

Name: Vote casting

Identifier: UC-2

Actors: Voter

Goals: The goal is to cast the vote

Preconditions:	UC-1 has been successfully completed User has logged on to the website
Trigger:	User selects vote button on website
Related use cases:	UC-3
Steps:	voter clicks on vote button He will be provided with list of the candidates. Voter will choose his desired option. He will submit the vote.
Post conditions:	vote is encrypted and sent to server.

UC-3

Name:	view results
Identifier:	UC-3
Actors:	voter
Goals:	the goal is to see the results of election.
Preconditions:	UC-2 has been successfully completed
Trigger:	user selects to view results
Related use cases:	UC-2

Steps: voter will log in to the system

Selects the results.

Results will be retrieved from the system

5.4.2 Class Diagram

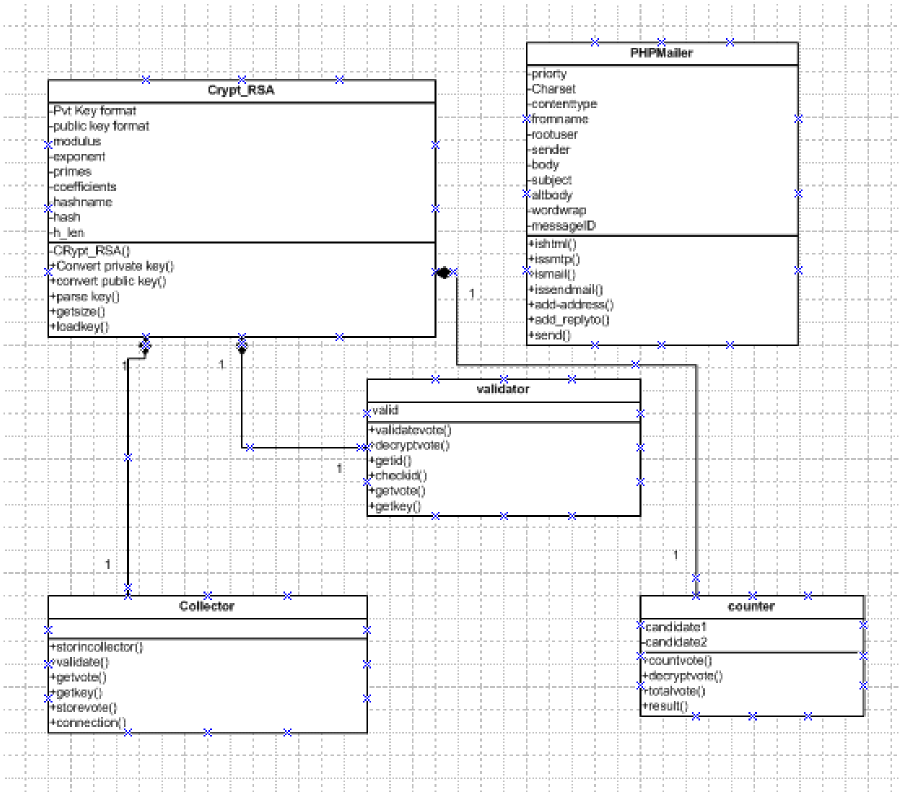


Fig: 5f

5.4.3 Use case diagrams

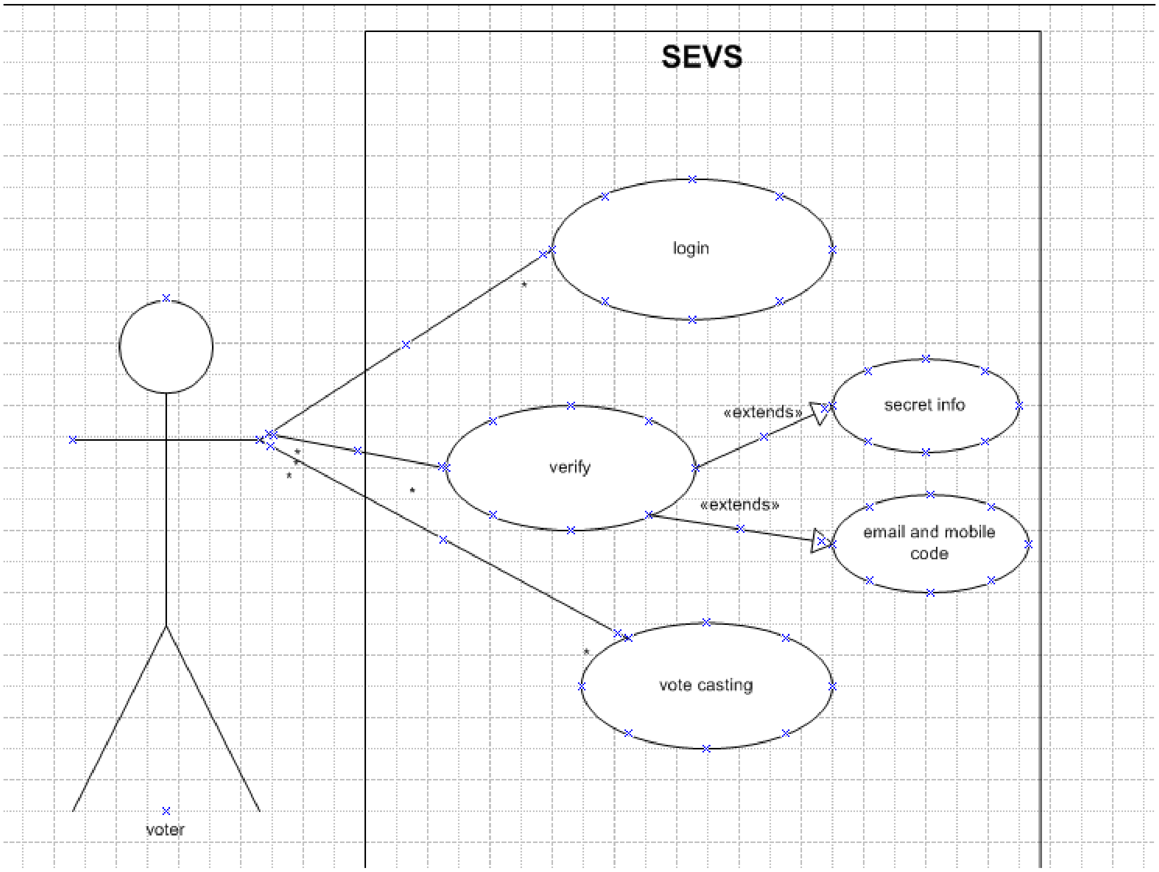

SEVS
login
verify
secret info
email and mobile code
vote casting
«extends»
«extends»
*
*
*
*
*
voter

Fig: 5g

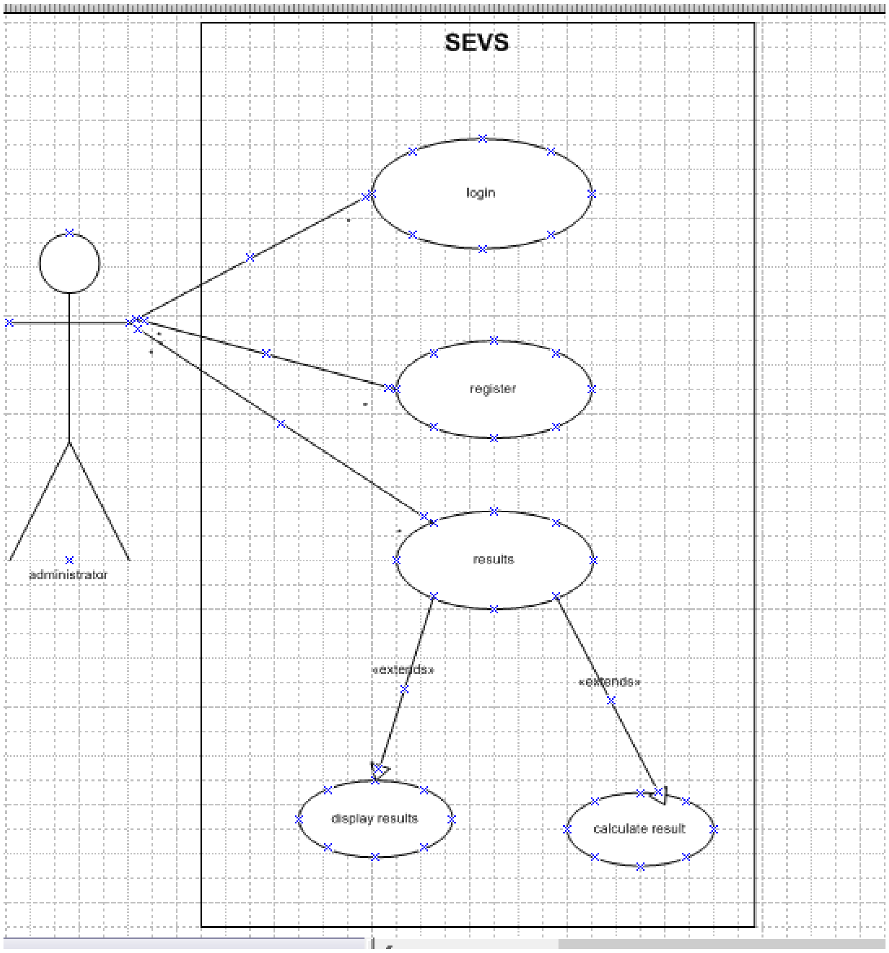


Fig: 5h

5.5 Dynamic View

Dynamic view comprises of Interaction diagrams. Following interaction diagrams included:

System Sequence Diagram

System sequence diagram represent interaction between user and system. System is considered as a black box. Inner functionality of system is not shown.

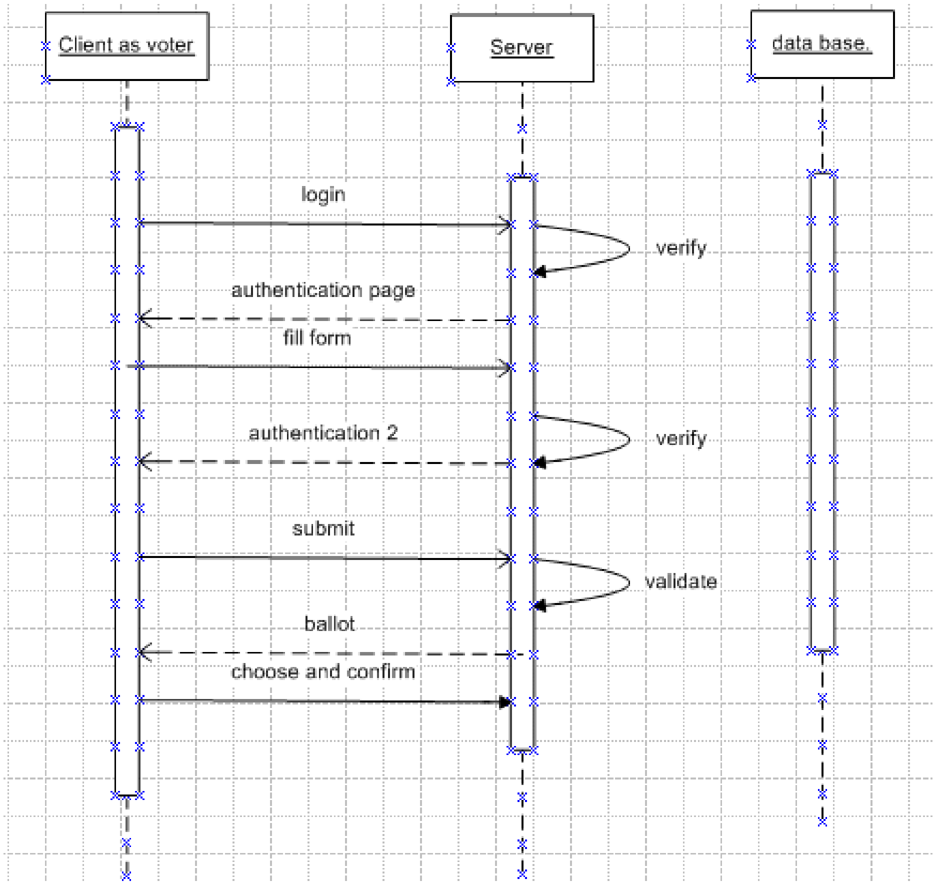

Client as voter
Server
data base.
login
verify
authentication page
fill form
authentication 2
verify
submit
validate
ballot
choose and confirm

Fig: 5i

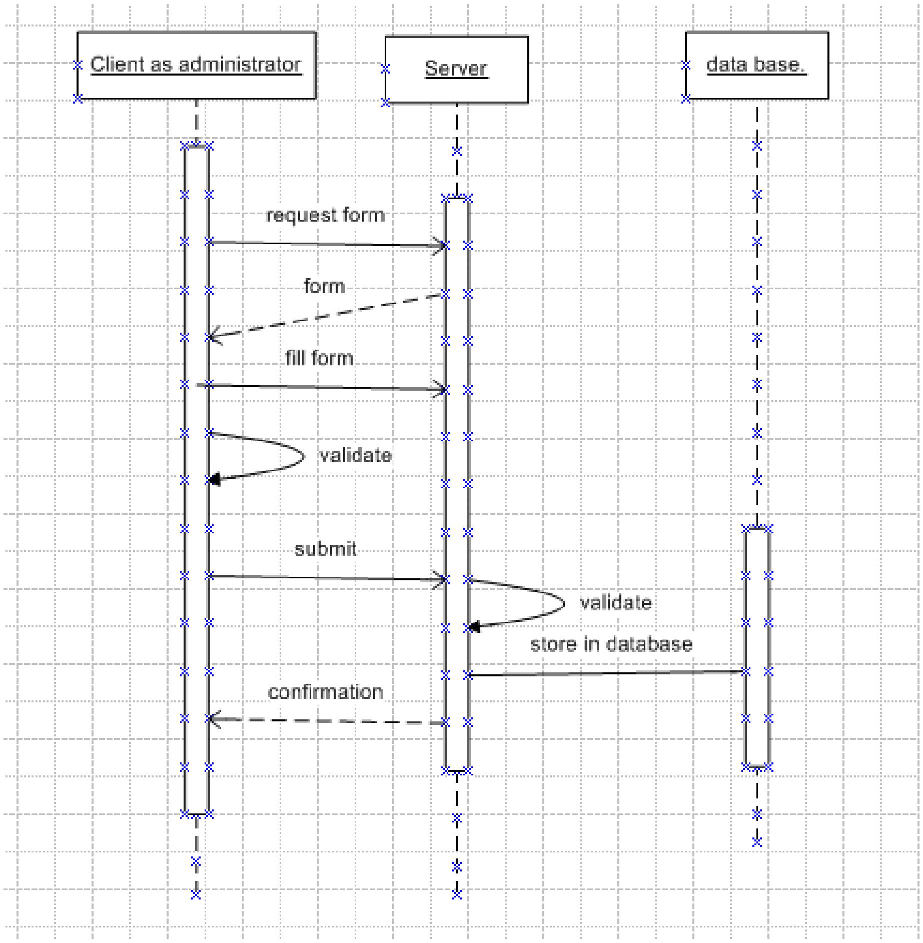

Client as administrator
Server
data base.
request form
form
fill form
validate
submit
validate
store in database
confirmation

Fig: 5j

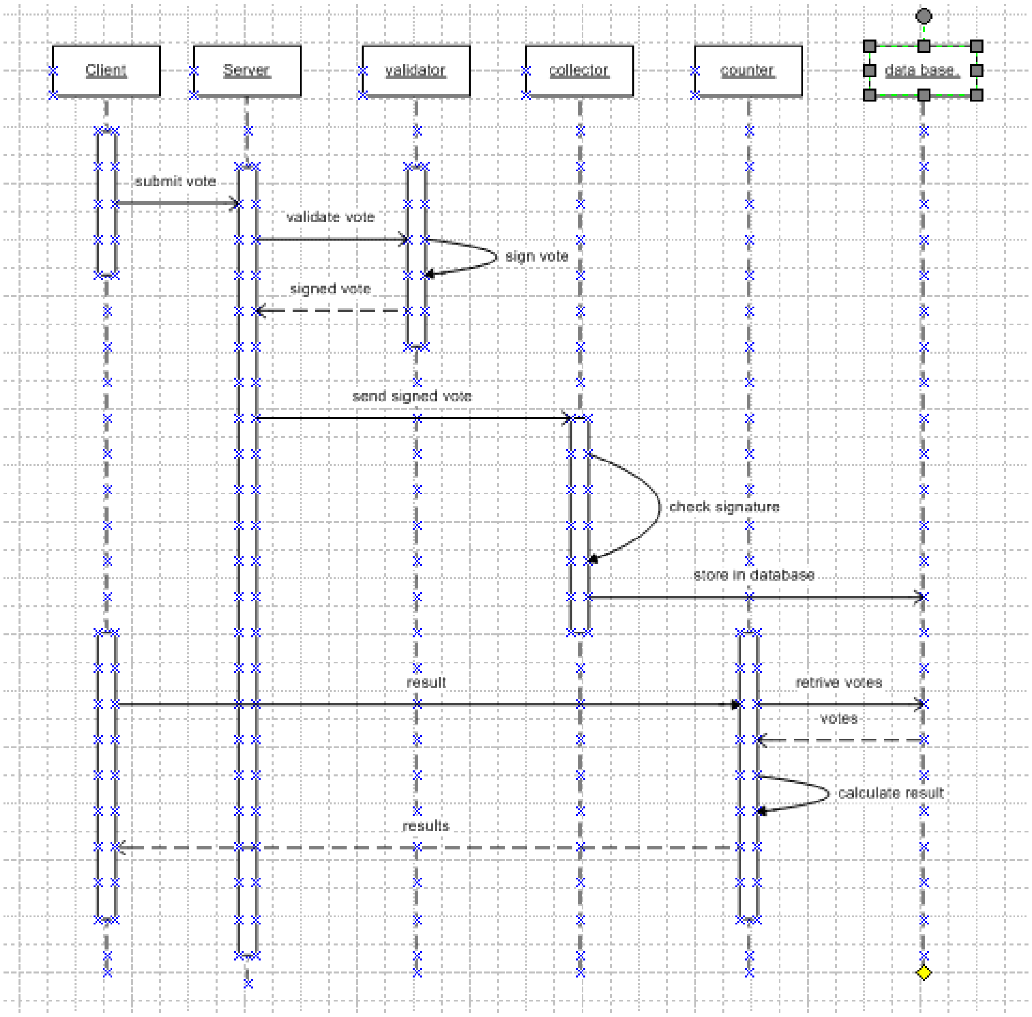

Client
Server
validator
collector
counter
data base
submit vote
validate vote
sign vote
signed vote
send signed vote
check signature
store in database
result
retrive votes
votes
calculate result
results

Fig: 5k

Chapter 6: System Implementation

6.1 SYSTEM IMPLEMENTATION

Secureelectronic voting system has been developed in the form of a web application which will be hosted on a web server. However the server side is segregated in to different servers on basis of separation of concerns as described in the protocol.

6.2 TECHNOLOGIES USED

The user interface of the website is developed using HTML 5.0 and CSS 3.0. The website coding is done in dream viewer using PHP as language. And the server is run on apache server. The database is made using MySQL.

6.2.1 Dream viewer

Adobe Dreamweaver is a proprietary web development application developed by Adobe Systems.

6.2.2 PHP

PHP is a server-side scripting language designed for web development.

PHP code is interpreted by a web server with a PHP processor module which generates the resulting web page: PHP commands can be embedded directly into an HTML source document rather than calling an external file to process data. It has also evolved to include a command-line interface capability and can be used in standalone graphical applications.

6.3 IMPLEMENTATION DETAILS

The website is developed using PHP as server side scripting language and HTML and CSS for interface designing. The pages are traversed using html anchor tag. The forms are made using forms tags

The home page consists of only 1 button.ie “Go” button. It refers to the login page.

The code is as follow for home page Go button.

```
<div id="go_botton"><a href="pages/login.php"><imgsrc="images/go_button.png"
alt="go to login" width="148" height="100" /></a></div>
```

The login page consists of two text boxes and two buttons. The code behind login button will check the entries of text boxes from database and redirect to either of menu page or authentication page on basis of category of user.

Code snippet is as follow

```
if(isset($_POST["login"]))

{

        $id=$_POST["mcs_id"];

        $pass=$_POST["password"];

        $c_id=stripslashes($id);

        $c_pass=stripslashes($pass);

        // To protect MySQL injection (more detail about MySQL injection)

        $sel="SELECT * FROM admin where id ='".$id."' LIMIT 1";

        $res=mysql_query($sel) or die(mysql_error());

        if(mysql_num_rows($res)==1)

        {

                $row=mysql_fetch_assoc($res);

                $p=$row["password"];

                if($p==$pass)
                {
                        header("Location: menu.php");
                        exit();
                        }
                }
        $select="SELECT * FROM voterdata where mcs_id ='".$id."' LIMIT 1";
        $result=mysql_query($select) or die(mysql_error());
if(mysql_num_rows($result)==1)
{
        $row=mysql_fetch_assoc($result);
        $p=$row["password"];
        if($p==$pass)
        {
        $_SESSION["login"]=1;
        $_SESSION["authentic"]=0;
        $_SESSION["userID"]=$row["mcs_id"];
        $username=$row["first_name"].' '.$row["last_name"];
        $_SESSION["username"]=$username;
        $_SESSION["email"]=$row["email"];
        header("Location: authentication.php");
        exit();

        }

        else

                $message="<h4 style='color:#F00'>Invalid password! Please try
again!</h4>";

}

else

        $message="<h4 style='color:#F00'>Invalid user name!  Please try again!</h4>";

}

?>
```

For administrator

Menu page has two main buttons i.e. register and results

Register

It will ask for input of voter data validate the inputs and store them in database using MySQL query.

Results

It will stop the voting process and count the votes from database after decrypting them.

```
<?php include("../inc/database.in.php"); ?>

<?php include("../counter.php"); ?>

<?php

        {
                $rsa = new Crypt_RSA();

                $counter=new counter();

                $counter->countVote($con,$rsa);

                $result=$counter->result();

        ?>
```

For Voter

Voter will be asked to provide secret info i.e. mother name and place of birth then it will be verified and an email and SMS will be sent to the voter and then he will have to provide email and mobile code to go on to the ballot paper. After selection of candidate it will be confirmed using dialog box and then further process is done.

```
$message="";

$success=false;

if(isset($_POST["submit"]))

{

$place=$_POST["place"];

$mother_name=$_POST["mother_name"];

$c_place=stripslashes($place);

$c_mother_name=stripslashes($mother_name);

        // To protect MySQL injection (more detail about MySQL injection)

        $select="SELECT * FROM  voterdata WHERE mcs_id ='".$_SESSION["userID"]."'
LIMIT 1";

        $result=mysql_query($select) or die(mysql_error());
```

```
if(mysql_num_rows($result)==1)

{



    $row=mysql_fetch_assoc($result);

    $p=$row["pob"];

    $m=$row["mother_name"];

if($place!=$p)

        echo "<h4 style='color:#F00'>&#8220 $place &#8221  is not the valid
place.</h4>";

elseif($mother_name!=$m)

        echo "<h4 style='color:#F00'>&#8220 $mother_name&#8221  is not the
valid Mother's name.</h4>";

else

        $code=createRandomCode();



</body>

";

        if (Send_Password_Recovery_Mail($_SESSION['email'],$messagecode))

        {

            $_SESSION["authentic"]=1;
```

```
         $_SESSION["ecode"]=$code;


        $success=true;

        header("Location: authentication2.php");

        exit();

         }

        else

         $message="<h4 style='color:#F00'>Connection Error! Please try again
!</h4>";

    }

else

$message="<h4 style='color:#F00'>Database Error! Please try again !</h4>";

}

?>

$message="";

if(isset($_POST["submit"]))

{

$code=$_POST["code"];

if($code==$_SESSION["ecode"])

{

    $_SESSION["authentic2"]=1;
```

```
        header("Location: ballot.php");

        exit();

        }

else

{

        $message="<h4 style='color:#F00'>Invalid E-mail Code.</h4>";

}

}

<?php

if(isset($_POST["submit"]) && $_POST['submit'] == 'Vote'

        && !empty($_POST['can']))

{

        if($_POST["can"]=="can1")

        {?>

<form action="../inc/vote.in.php" class="login-form" method="post">

<div class="content">

<div class="footer">

<input name="vote_value" type="hidden" value="can1" />

</div>

</form>
```

```
    <?php
    }
    elseif($_POST["can"]=="can2")
    {
        ?>
<form action="../inc/vote.in.php" class="login-form" method="post">
```

After the vote is submitted the value of the vote is encrypted using counter public key and the user id attached and again encrypted with validator public key and sent to validator.

Validator will decrypt the vote using its private key and strip the id validate the vote and attach md5 hash to it and send it back to the voter.

The voter will send the vote to collector who will match the md5 hash and store the vote in database along with its key.

Then counter will count the votes.

A PHP mailer library is used to send the email and an API is used to send the SMS.

PHPseclibrary is used to encrypt and decrypt votes and generate RSA keys and hashing. CryptRSA class of this library is used.

Validator code.

```
<?php include("inc/database.in.php");?>
<?php
class validator{
private $valid=false;
```

```
public function validateVote($vote){

        $decryptedVote= $this->decryptVote($vote);

        $id=$this->get_id($decryptedVote);



        if ($this->check_id($id)==false){

                $this->valid=false;

                }

        else

                $this->valid=true;

        $getVote=$this->getVote($decryptedVote);

        $key=$this->getkey();

        return array ('vote'=>$getVote,'key'=>$key,'valid'=>$this->valid);

        }

        private function decryptVote($value){

        $rsaValidator=$value['rsaValidator'];

        $ciphertext=$value['vote'];

        $rsaValidator->loadKey($_SESSION["V_pvk"]);

        $plaintext =$rsaValidator->decrypt($ciphertext);

        return $plaintext;

        }
```

```
private function get_id($value){
        $id = substr($value, -8);
        return $id;
        }
private function check_id($id){
        $select="SELECT * FROM voterdata where mcs_id ='".$id."' LIMIT 1";
        $result=mysql_query($select) or die(mysql_error());
        if(mysql_num_rows($result)==1){
                $row=mysql_fetch_assoc($result);
                $status=$row["vote_status"];
                if($status==0){
                        return true;
                        }
                else
                        return false;
                }
        }
private function getVote($decryptedVote){
        $len=strlen($decryptedVote);
        $vote=substr($decryptedVote,0,$len-8);
        return $vote;

        }

private function getkey(){

        $pvk=$_SESSION["V_pvk"];

        $pvk_d=md5($pvk);

        return $pvk_d;

        }

}

?>
```

Collector code

```
<?php include("inc/database.in.php");?>

<?php

class collector{

public function storeInCollector($signedVote,$rsaCounter){

                $valid=$this->validate($signedVote);

        if($valid){

                $key=$this->getKey();

                $vote=$this->getVote($signedVote);

                $store=$this->storeVote($key,$vote,$rsaCounter);

                if($store)
        return true;

        else

        return false;

        }

    }

private function validate($signedVote){

    $key=$signedVote['key'];

    $id=$_SESSION["userID"];

    $select="SELECT * FROM voterdata where mcs_id ='".$id."' LIMIT 1";

    $result=mysql_query($select) or die(mysql_error());

    if(mysql_num_rows($result)==1){

        $row=mysql_fetch_assoc($result);

        $db_key=$row["pvk"];

        if($db_key==$key){

            return true;

            }

        else

            return false;

        }

    }
    private function getVote($signedVote){

        $vote=$signedVote['vote'];

        return $vote;

        }

    private function getKey(){

        $key=$_SESSION["C_pvk"];

        return $key;

        }

    private function storeVote($k,$v,$c){

        $connection=$this->connection();

        $vote=base64_encode($v);

        $rsaCounter=base64_encode(serialize ($c));

        $insert_query="INSERT INTO vote(pvk,vote_value,rsacounter) VALUES (
'".$k."','".$vote."','".$rsaCounter."')";

        //echo $insert_query;

        $insert_data=mysql_query($insert_query,$connection);

        if(!$insert_data)

        {

        die(mysql_error() );

            return false;

        }
        else
        return true;
        }
    private function connection(){
        $connection = mysql_connect ("localhost","root","");
        if(!$connection)
        {
            die('Connection failed: '.mysql_error() );
            }
        $votes_db=mysql_select_db('evoting',$connection);
        if(!$votes_db)
        {
        die('Evoting database unavailable: '.mysql_error() );
            }
        return $connection;
        }
}
?>
```

Counter code

```
<?php include('inc/phpseclib/Crypt/RSA.php');?>
```

```
<?php

class counter{

          private $candidate1=0;

          private $candidate2=0;

                    public function countVote($connection,$rsa){

                    $total=$this->totalVote($connection);

                    $select="SELECT * FROM `vote`";

                    $result = mysql_query($select);

                    if (!$result){

                              die(mysql_error());

                    }



                    else {

                    while ($row=mysql_fetch_array($result)){

          // Append all results onto an array

          $rowset[] = $row;

                                        }

                              }

                    foreach ($rowset as $row) {

                              $key=$row['pvk'];
```

```
        $voteValue=$row['vote_value'];
        $r=$row['rsacounter'];
        $vote=base64_decode($voteValue);
        $rsa=unserialize (base64_decode($r));
        $rsaCounter=$rsa;
        $value=$this->decryptVote($vote,$key,$rsaCounter);
        if($value=='can1'){
            $this->candidate1++;
            }
        else{
            $this->candidate2++;
            }
    }
    }
private function totalVote($connection){
    $query="SELECT * FROM `vote`";
    $count=mysql_query($query) or die(mysql_error());
    $total= mysql_num_rows($count);
    return $total;
    }
```

```
          private function decryptVote($vote,$key,$rsa){

                    //$rsa=$_SESSION["rsacounter"];

                    $rsa->loadKey($key);

                    $plaintext =$rsa->decrypt($vote);

                    return $plaintext;

                    }

          public function result(){

                    return array(

                    'candidate1'=>$this->candidate1,

                    'candidate2'=>$this->candidate2);

                    }

}

?>
```

Chapter 7: Testing and Result Analysis

7.1 WEB TESTING

For testing a web application you need to do the functionality, performance and usability testing. You need to test the server side site interface and its compatibility with the client side.

You also need to test the security features of the web site.

7.2 Testing Functionality

In functionality testing of the web sites the following was done.

7.2.1 Links testing

Test the outgoing links from all the pages of the web site. Test all of the internal links. Test links leading to the same pages. Also testing, for the presence of orphan pages and check for broken links.

7.2.2 Forms testing

Forms are used to get information from users. In testing forms you should test the validation checks etc.

- Check all the validations on fields.
- Check for the default values of fields.
- Effect of wrong inputs to the fields.
- Testing to see if an invalid voter is allowed access or not.

7.2.3 Database testing

Test if all the database queries are executing correctly, data is retrieved correctly and also updated correctly

7.2.4 Cookies testing

Cookies are deleted so that no session information is saved.

7.2.5 Test for navigations

Navigation the web pages, different controls such as buttons, boxes etc.

7.2.6 Usability testing includes

Web site should be easy to use and user friendly. Instructions should be provided clearly. It should be consistent.

7.2.7 Content checking

Content should be logical and related. Easy to understand. Check for wrong spellings. Do not use of dark colors. You should follow a standard. These are common accepted standards like as I mentioned above about annoying colors, fonts, frames etc.

Content should be meaningful. All the anchor text should be working properly.

These are some basic standards that should be followed in web development. Your task is to validate all for UI testing

7.2.8 Interface Testing:

The main interfaces are:

Web server and application server interface

Application server and Database server interface.

Check if all the interactions between these servers are executed correctly.

7.2.9 Compatibility Testing:

Compatibility of your web site is very important testing aspect. Check browser compatibility and operating system compatibility.

7.2.10Performance testing:

Web application should sustain to heavy load. Web performance testing should include:

Web Load Testing

Web Stress Testing

Test performance of the application on different internet connection speed.

In **web load testing** test how many users can access the same page.

In **Stress testing** is given on input fields, login and sign up areas.

In web performance testing web site functionality on different operating systems, different hardware platforms are checked for software, hardware memory leakage errors.

7.2.11 Security Testing:

Following are some test cases for web security testing:

- Test by pasting internal URL directly into browser address bar without login. Internal pages should not open.
- Input invalid inputs in input fields like login username, password, and input text boxes. See the systems reaction on all invalid inputs.
- All transactions, error messages, security breach attempts should get logged in log files somewhere on web server.
- The reason for testing the security of a web is to recognize potential vulnerabilities and then overhaul them.
  - Network Scanning
  - Vulnerability Scanning
  - Password Cracking
  - Log Review
  - Integrity Checkers
  - Virus Detection

7.2.12 Usability

Usability testing is the process by which the human-computer interactions are checked and weaknesses are recognized. It should be easy to learn. Navigations are checked. Finally general appearance is checked.

7.2.13 Server Side Interface

In web testing the server side interface should be tested. This is done by confirming that communication is done correctly. Compatibility of server with software, hardware, network and database should be verified.

7.2.14 Client Side Compatibility

The client side compatibility is also tested in various platforms, using various browsers etc.

7.3 TEST CASES

Test case number	1
Description	Testing login screen
Preconditions	Login screen
Input	Click the Go button
Steps	Open up a web browser and point it to: https://<URL or IP address> and click on Go button
Expected output	Make sure Log In page should display
Results	Pass

Test case number	2
Description	Testing login screen
Preconditions	Login screen
Input	Anam, 4587458
Steps	Enter voters id and password and click on "login"
Expected output	Make sure ballot page should display
Results	Pass

Test case number	3
Description	Testing login/logout
Preconditions	Login screen
Input	Click the Sign out button
Steps	Click the "Sign Out" link

Expected output	Make sure Log In page should display
Results	Pass

Test case number	4
Description	Enter invalid voters id in the username field
Preconditions	Login screen
Input	Gguyy, 4545745
Steps	Enter invalid voters id in the username field
Expected output	Make sure error message should display as: ‘invalid user’
Results	Pass

Test case number	5
Description	Without providing voters id
Preconditions	
Input	Click the login button
Steps	
Expected output	Make sure error message should display as: ‘please enter username’
Results	Pass

Test case number	6
Description	Look at the top of the page
Preconditions	
Input	
Steps	
Expected output	Make sure that the Select banner should display
Results	Pass

Test case number	7
Description	Look at the bottom of the page
Preconditions	
Input	
Steps	
Expected output	Make sure that the copyright statement should display at the bottom of page.
Results	Pass

Test case number	8
Description	Click vote button, then click the candidate you want to vote then press submit button
Preconditions	Ballot screen
Input	Select the candidate and submit
Steps	
Expected output	A message saying, “your vote is casted appears”
Results	Pass

Chapter 9: Conclusion and future work

9.1 CONCLUSION

The project has been completed according to what was promised initially. All the requirements are being fulfilled as per the SRS. The scope was MCS society elections. Hence the project is providing all the features it had to for conduction elections in the university.

9.2 FUTURE WORK

Considering the current situation of the country and keeping I mind the latest elections held, this system can be very significant if future enhancements are done in it.
We can contact it with NADRA’s database and add death certificates of the deceased citizens to avoid the practice of other people casting votes for dead people. We can also add Google map for the easy of the voter.

Besides that we are planning to write a paper and send it to an IEEE conference.

Appendix A: References

[1] Voting, Parliamentary Office of Science and Technology, May 2001, http://www.parliament.uk/post/pn155.pdf.

[2] Schneier B., Applied Cryptography, John Wiley, 1996.

[3] Numi H., Salomaa A. and Santean L., Secret Ballot Elections in Computer Networks, Computers and Security 36(10), 1991, pp553-560.

[4] Delaune S., Kremer S and Ryan M., Verifying Properties of Electronic-Voting Protocols, ftp://ftp.cs.bham.ac.uk/pub/authors/M.D.Ryan/06-wote.pdf

[5] RachidAnane, Richard Freeland and GeorgiosTheodoropoulos , Computer and Network Systems, Coventry University, UK.

[6] Caltech/MIT Voting Technology Project, "What is what could be," http://web.mit.edu/voting/, July 2001; http://web.mit.edu/newsoffice/nr/2001/VTP_report _all.pdf.

[7] California Internet Voting Task Force, "A Report on the Feasibility of Internet Voting," Jan. 2000, www.ss.ca.gov/executive/ivote/final_report.htm

[8] J.Benaloh, M.Fischer. A Robust and Ver- ifiable Cryptographically Secure Election Scheme, Proceedings of 26th Symposium on Foundations of Computer Science. Port- land, OR. October 1985. IEEE 1985, pp. 372-382.

[9] Public-Key Cryptosystems

[10] Formal Specification and Analysis of an e-Voting System

Appendix B

User Manual

8.1SYSTEM SETTINGS

8.1.2 Software Requirements

Server Requirements Apache 2.4.2 server is required to be installed on the server system.Window7/8 operating system is required. Client can use any type of browser.

8.1.3 Hardware Requirements

A system with a processor of 2.4 GHz or more.1GB of RAM or more and Windows 7/8 operating system.

8.1.4 System Configuration

Following steps needs to be taken for configuring the software:

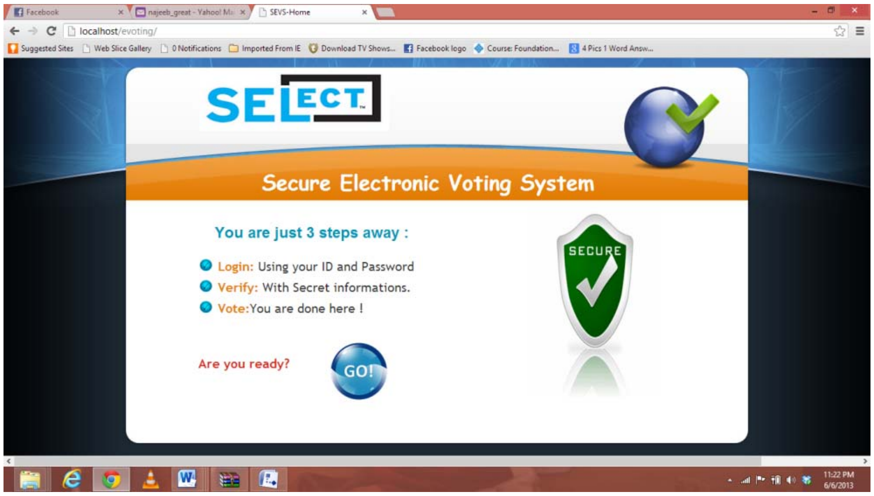
1.Install Wamp Server on the computer with specification mentioned above

2.Load the Registration project file from SEVS project Folder

8.2 User Web Interface

8.2. 1 Homepage

The homepage consists of following tabs whose details are described with them.



Secure Electronic Voting System

You are just 3 steps away :

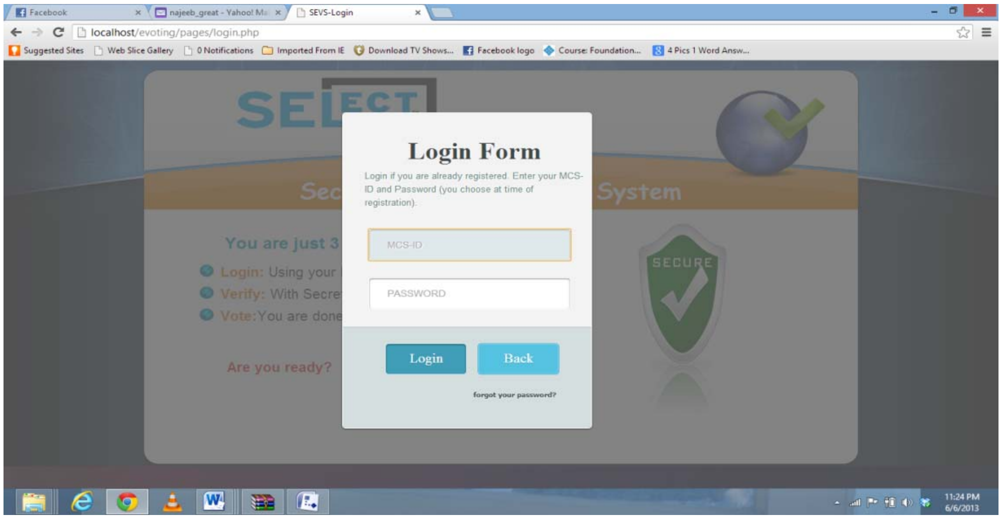
- Login: Using your ID and Password
- Verify: With Secret informations.
- Vote:You are done here !

Are you ready?

Fig: 8a

Go: It takes the user to the login page.

8.2.2 Login Page



Login Form

Login if you are already registered. Enter your MCS-ID and Password (you choose at time of registration).

Fig: 8b

It has 2 text boxes for MCS id and password.

The 2 buttons are login which takes to next page and back button.

There is also a link for "forget your password"

8.2.3 User Roles in SEVS

SEVS has defined two basic characters

Administrator

Registered User

Administrator

Administrator is required to enter username and password before the functionalities of the Server can be used.

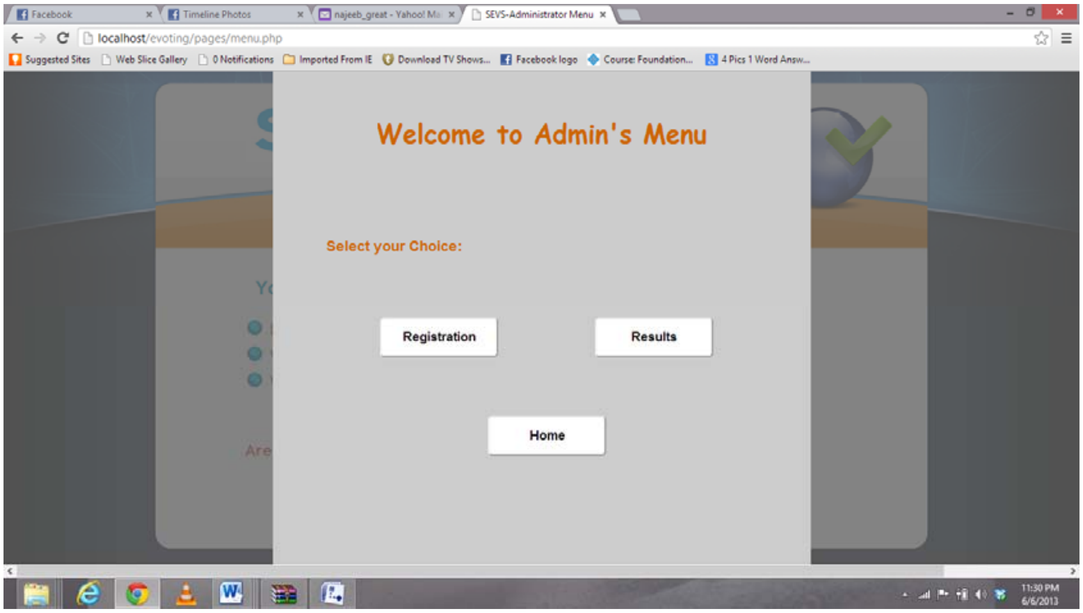
The Administrator will enter his username and password and click on login.

The credentials of the Administrator will be verified from the SEVS database and if valid the administrator will be logged into the server.

The Administrator can register new user or calculate and view results.

8.2.4 Registration Page

Click on register button from menu page

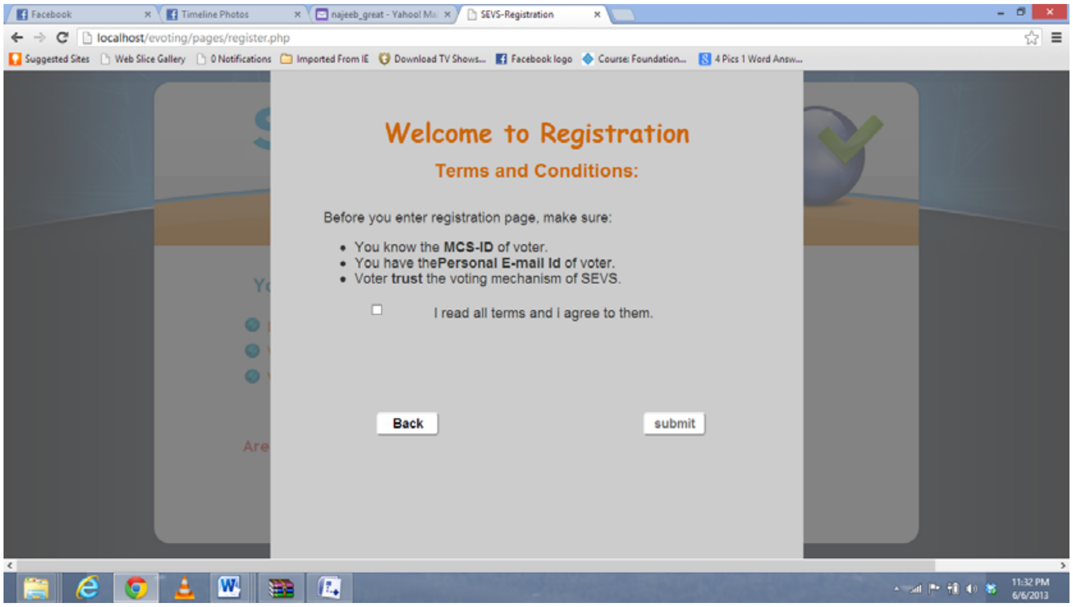


Welcome to Admin's Menu

Select your Choice:

Fig: 8c

Check the agreement radio box and click on submit



Welcome to Registration

Terms and Conditions:

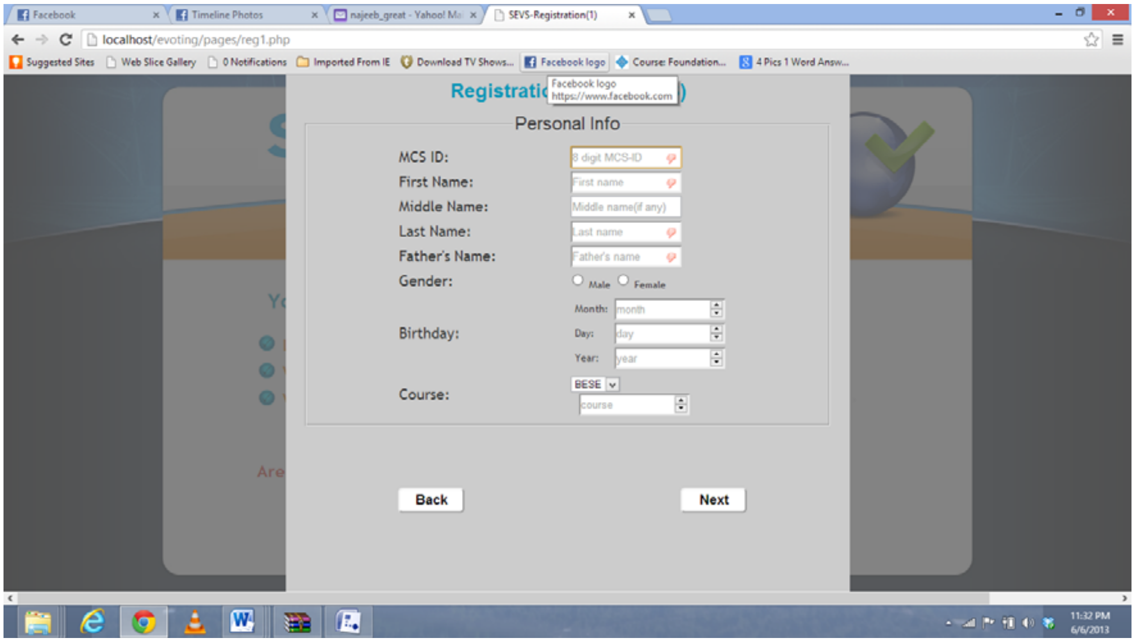
Before you enter registration page, make sure:

- You know the **MCS-ID** of voter.
- You have the**Personal E-mail Id** of voter.
- Voter **trust** the voting mechanism of SEVS.

□ I read all terms and I agree to them.

Fig: 8d

Fill the data and click next



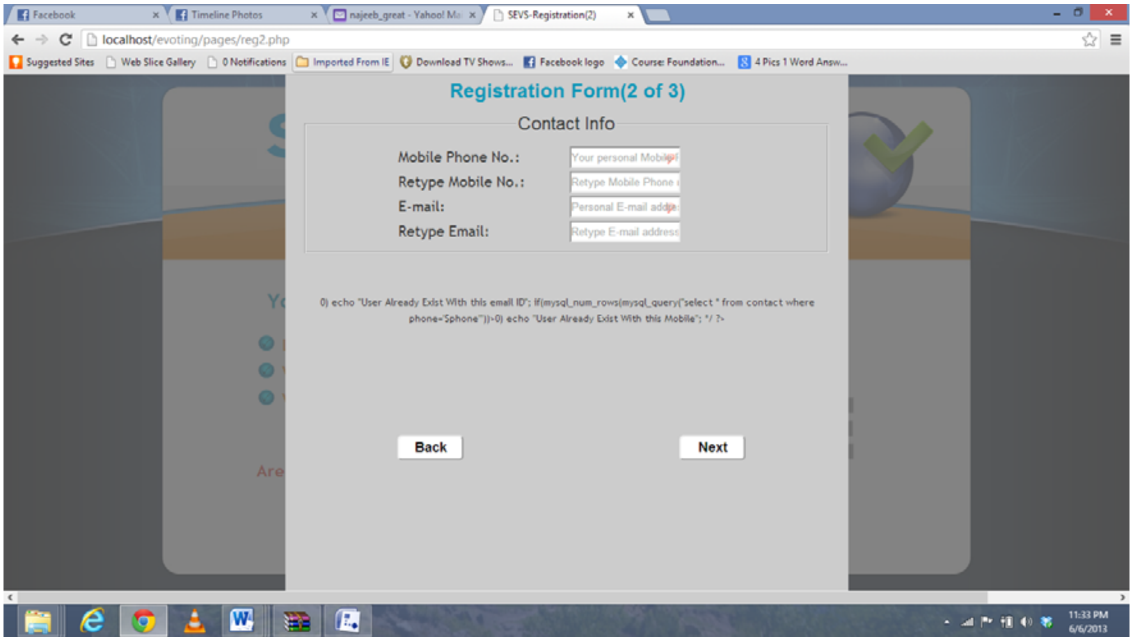
Registratio)

Personal Info

MCS ID: 8 digit MCS-ID
First Name: First name
Middle Name: Middle name(if any)
Last Name: Last name
Father's Name: Father's name
Gender: ○ Male ○ Female
Birthday: Month: month
Day: day
Year: year
Course: BESE
course

Fig: 8e

Fill the data and click next



Registration Form(2 of 3)

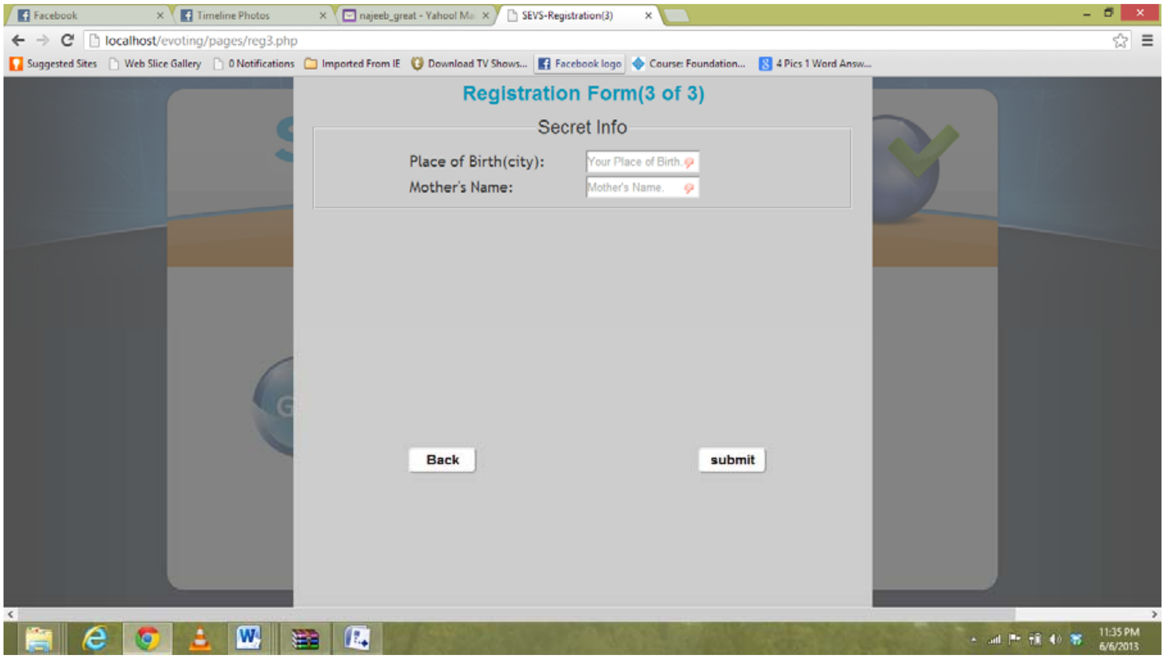
Contact Info

Mobile Phone No.:	Your personal Mobile
Retype Mobile No.:	Retype Mobile Phone
E-mail:	Personal E-mail addre
Retype Email:	Retype E-mail address

```
0) echo "User Already Exist With this email ID"; if(mysql_num_rows(mysql_query("select * from contact where
phone='$phone'"))>0) echo "User Already Exist With this Mobile"; */ ?>
```

Fig: 8f

Fill the data and click submit



Registration Form(3 of 3)

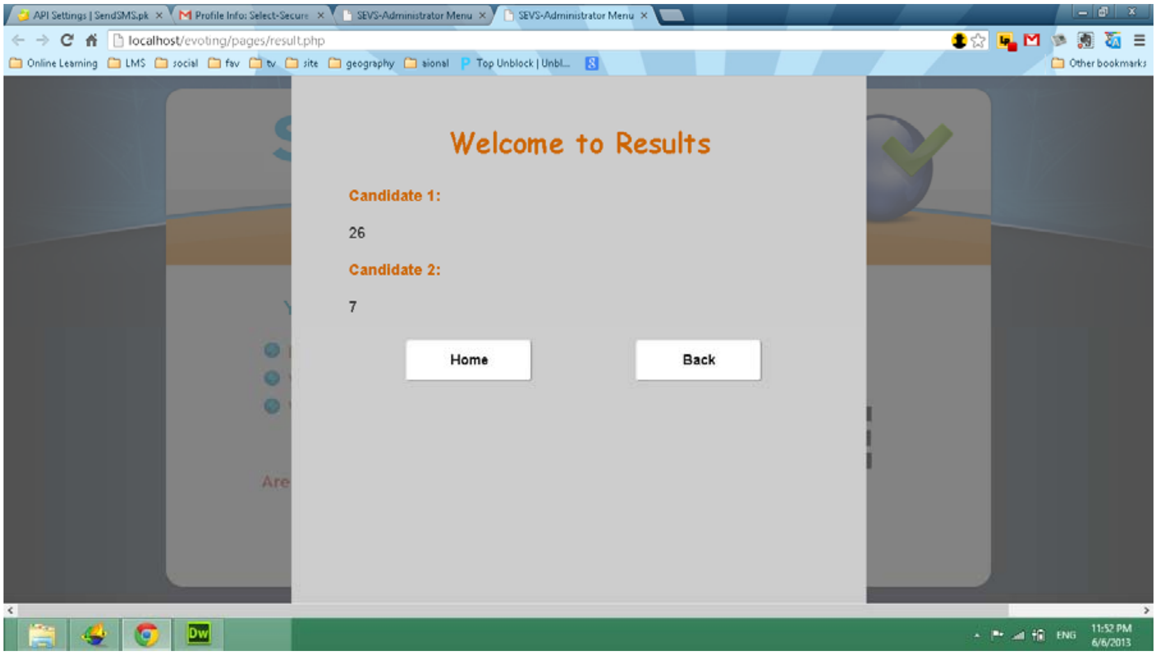
Secret Info

Place of Birth(city):	Your Place of Birth.
Mother's Name:	Mother's Name.

Fig: 8g

8.2.5 Result Page

Click on result button from menu page



Welcome to Results

Candidate 1:

26

Candidate 2:

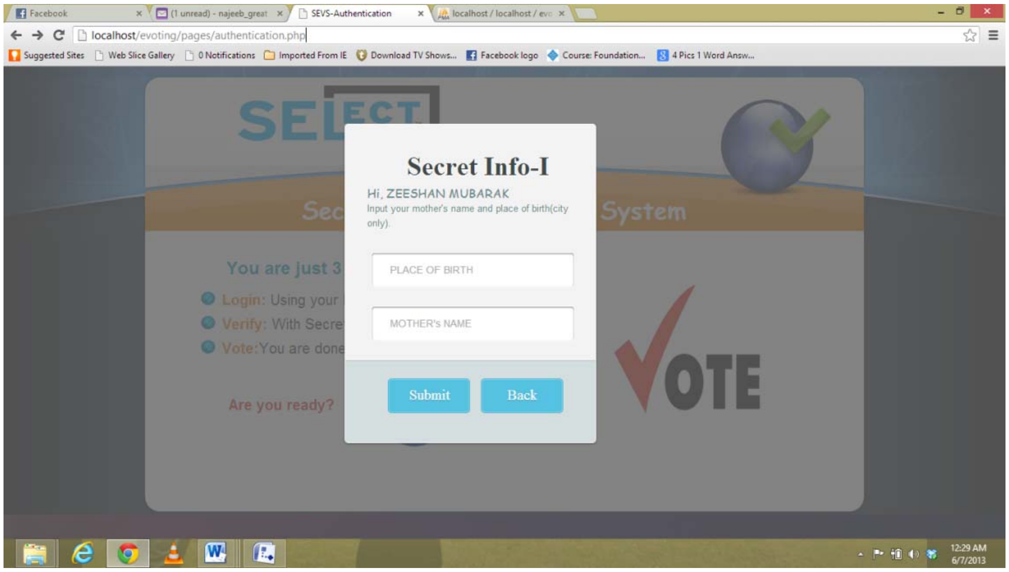
7

Fig: 8h

8.2.5 Registered Voter

Registered User is allowed cast the vote following these steps.

Provide the info and click submit

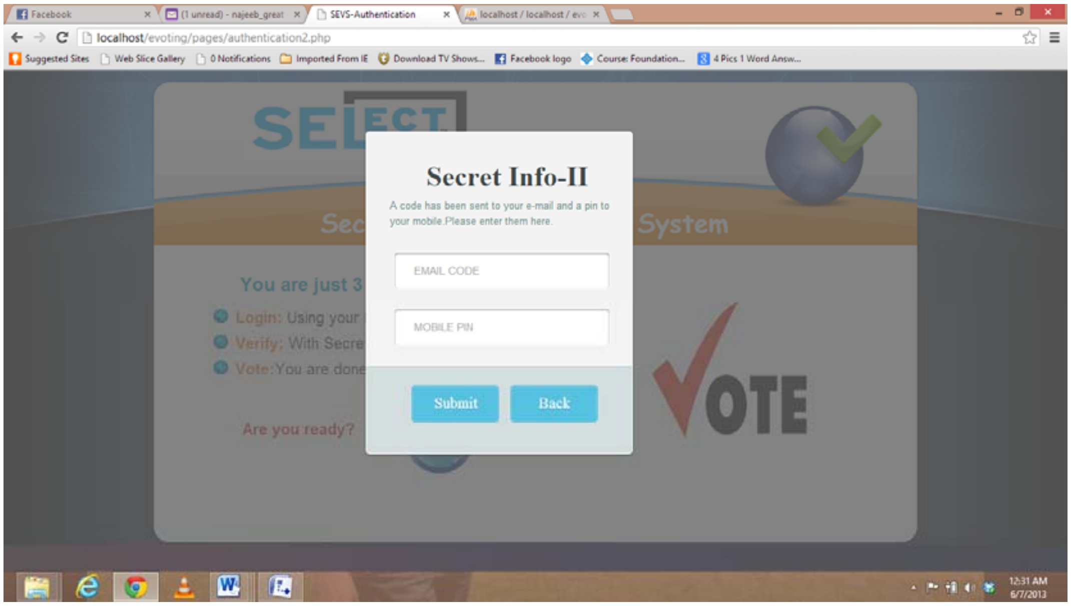


Secret Info-I

Hi, ZEESHAN MUBARAK

Input your mother's name and place of birth(city only).

Fig: 8i

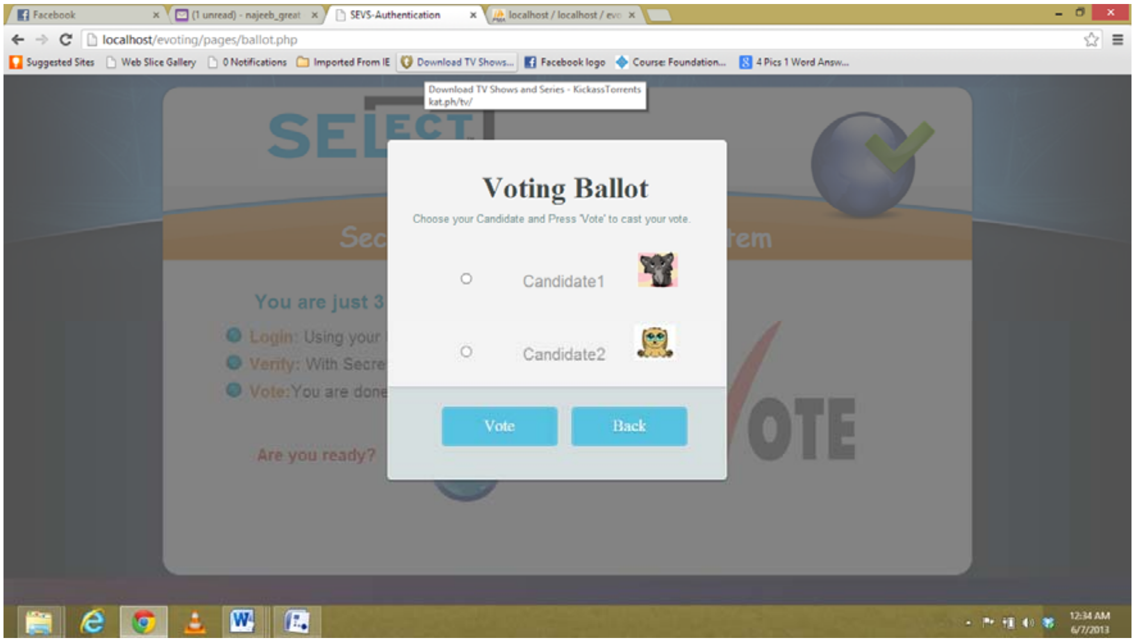


Secret Info-II

A code has been sent to your e-mail and a pin to your mobile.Please enter them here.

Fig: 8j

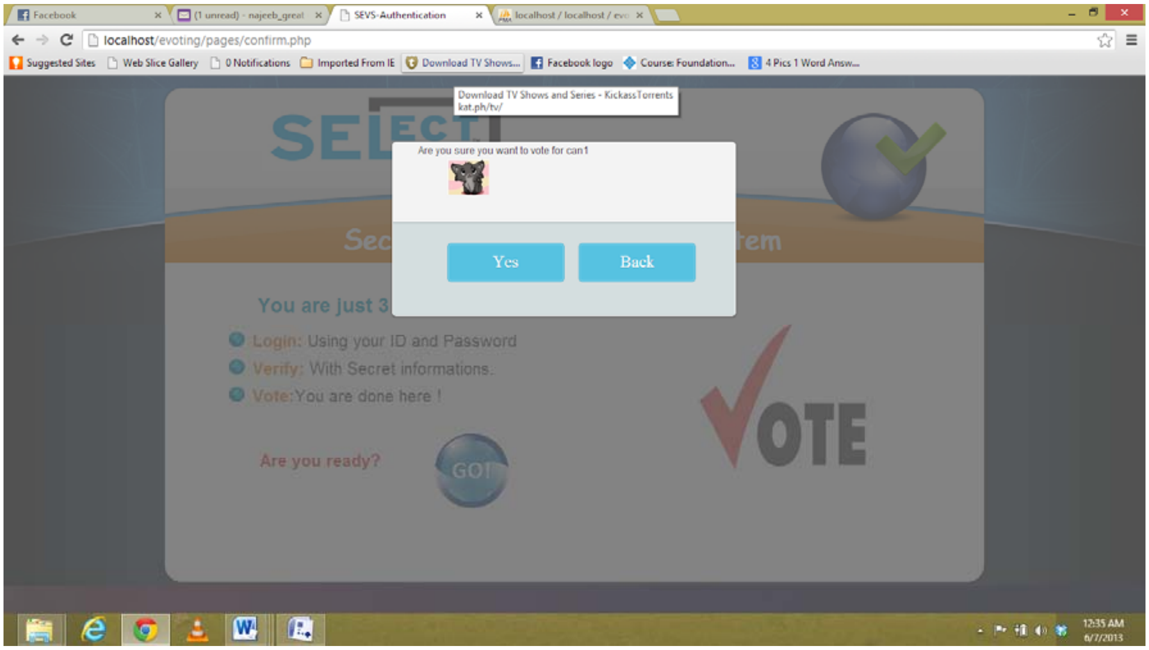
Choose your candidate and click vote



Choose your Candidate and Press 'Vote' to cast your vote.

Fig: 8k

Click yes for confirmation. The vote is casted.



Are you sure you want to vote for can1

You are just 3

- Login: Using your ID and Password
- Verify: With Secret informations.
- Vote: You are done here !

Are you ready?

Fig: 8l