**A**

***PROJECT REPORT***

*on*

**FINGERPRINT VOTING SYSTEM**

*Submitted in partial fulfilment of the requirements for the degree of*

**BACHELOR OF TECHNOLOGY**

****

Session: -Jan-June 2023

Submitted by

Aniket Kothari (19ETCCS004)

Dipesh Vyas(19ETCCS016)

Tarun Paliwal (19ETCCS072)

VIII semester, CSE

Under Guidance of

Mr. Aaditya Maheshwari

Head of industry project

CSE & Techno India NJR Institute of Technology

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**TECHNO INDIA NJR INSTITUTE OF TECHNOLOGY, UDAIPUR-313001**

**MAY - 2023**

**A**

***PROJECT REPORT***

*on*

**FINGERPRINT VOTING SYSTEM**

*Submitted in partial fulfilment of the requirements for the degree of*

**BACHELOR OF TECHNOLOGY**



**Session: - Jan-June 2023**

Under Guidance of

Mr. Aaditya Maheshwari

Head of Industry Project

CSE & Techno India NJR Institute of Technology

Submitted by

Aniket Kothari (19ETCCS004)

Dipesh Vyas (19ETCCS016)

Tarun Paliwal (19ETCCS072)

VIII semester , CSE

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**TECHNO INDIA NJR INSTITUTE OF TECHNOLOGY, UDAIPUR-313001**

**MAY - 2023**



Department of Computer Science and Engineering

Techno India NJR Institute of Technology, Udaipur-313001

**Certificate**

This is to certify that project work titled “FINGER PRINT VOTING SYSTEM” by **Aniket Kothari ,Dipesh Vyas & Tarun Paliwal** was successfully carried out in the Department of Computer Science and Engineering, TINJRIT and the report is approved for submission in the partial fulfillment of the requirements for award of degree of Bachelor of Technology in Computer Science and Engineering.

Mr. Aaditya Maheshwari Dr. Rimpy Bishnoi

Head of Industry Project Head of Department

CSE & Techno India NJR Dept. of CSE TINJRIT, Udaipur

Date...................... Date......................



Department of Computer Science and Engineering

Techno India NJR Institute of Technology, Udaipur-313001

**Examiner Certificate**

This is to certify that the following student

**Aniket Kothari, Dipesh Vyas, Tarun Paliwal**

of final year B.Tech. (Computer Science and Engineering), was examined for the project work titled

***“Fingerprint Voting System”***

during the academic year 2022 – 2023 at Techno India NJR Institute of Technology, Udaipur

**Remarks:**

**Date:**

Signature Signature

(**Internal Examiner**) (**External Examiner**)

Name :- ……………………… Name :- ………………………

Designation:- ……………….. Designation:- ………………..

Department: - ………………. Department: - ……………….

Organization:- ……………… Organization:- ………………

**PREFACE**

The most severe and frequently happening situations while conducting elections is rigging (One person give multiple votes). Right now, we are using an electronic voting machine and to identify people who already gave their vote, people will be given an ink mark on the finger. But today because of rapid development in technology, that can be erased and there is a chance of rigging. Through this project, you are going to create a unique Fingerprint based Voting system in order to avoid the above-mentioned problem. Biometric Finger print devices are used in the Electronic Voting machine for voter verification. We have designed a finger print based voting machine where there is no need for the user to carry his ID which contains his required details. The person at the polling booth needs only to place his Finger on the device, thus allowing the acquisition of an on-spot fingerprint from the voter which serves as an identification. We have designed this project as a prototype only in C#.net and not used any real hardware. We have taken the scanned images for the working of the Fingerprint based Voting System.

**ACKNOWLEDGMENT**

We take this opportunity to record our sincere thanks to all who helped us to successfully complete this work. Firstly, We are grateful to our **supervisor Mr.Aaditya Maheshwari** for his invaluable guidance and constant encouragement, support and most importantly for giving us the opportunity to carry out this work.

We would like to express our deepest sense of gratitude and humble regards to our

**Head of Department Dr. Rimpy Bishnoi** for giving invariable encouragement in our endeavors and providing necessary facility for the same. Also a sincere thanks to all faculty members of CSE, TINJRIT for their help in the project directly or indirectly.

Finally, We would like to thank my friends for their support and discussions that have proved very valuable for us. We are indebted to our parents for providing constant support, love and encouragement. We thank them for the sacrifices they made so that we could grow up in a learning environment. They have always stood by us in everything we have done, providing constant support, encouragement and love

**Aniket Kothari & 19ETCCS004**

**Dipesh Vyas & 19ETCCS016**

**Tarun Paliwal & 19ETCCS072**

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

TECHNO INDIA NJR INSTITUTE OF TECHNOLOGY, UDAIPUR-313001

**CONTENTS**

**Abstract ……………………………………………………………….**

**Contents ……………………………………………………………….**

**List of Figures ……………………………………………………............**

**List of Tables …………………………………………………………….**

**List of Abbreviations Used……………………………………………….**

**Chapter 1** INTRODUCTION

1.1 Basics

1.2 Purpose

1.3 System Analysis

1.4 Modules and their Description

1.5 Existing system

1.6 Proposed system

**Chapter 2** IMPLEMENTATION

2.1Modules

2.2 Modules description

**Chapter 3** INPUT AND OUTPUT DESIGNS

**Chapter 4** SOFTWARE ENVIROMENT

4.1 Features of .NET

4.2 Features of SQL server

**Chapter 5** SYSTEM INFORMATION

5.1 System design

5.2 System specification

5.3 System Study

5.4 System Testing

5.5 System Analysis

**Chapter 6** SCREEN SHOTS

**Chapter 7** LITERATURE SURVEY

**Chapter 8** CONCLUTION

**Chapter 9** BIBILIOGRAPHY

**LIST OF FIGURES**

1. Fig1- Existing System
2. Fig2- .NETFrame work
3. Fig3- Data flow diagram
4. Fig4- UML diagrams
5. Fig5- Class diagram
6. Fig6- Sequence diagram
7. Fig7- Activity diagram
8. SCREENSHOTS

**INTRODUCTION**

**1.1 Basics**

Biometrics is the statistical analysis for measuring biological data. It refers to technologies that calculate and find human body characteristics, such as DNA, fingerprint, retina, irises, voice or any type of noisy patterns, facial patterns and hand gesture measure, for authentication and verification purposes.

In our research, we have used fingerprint for the purpose of voter identification and authentication. As each and every individual has unique fingerprint patterns, it helps to attain maximum accuracy.

A database is developed that stores fingerprint of every individual in the constituency.So,It checks the illegal and the repetition of votes. Hence the elections would be conducted fair and free from any type of rigging.

“**Fingerprint Election System**” is the system we designed that helps the user to vote smoothly and effectively.

* The Voting system will have server services which are each connected to a remote link database for storing persistent data.
* Admin will maintain all information regarding voter and responsible for fair conduction of the elections.

**1.2. Purpose**

* The Purpose of this project is to conduct an election easily and effectively.Hence,the election will be fair and free from any bad practice.
* This research intends to make an election dynamic. So, any number of candidates can apply for the election.
* This research on “Fingerprint Election System” which is being developed as required for an academic course.
* This research is done to provide a detailed specification of the requirements for the developers.

**1.3 System Analysis**

**Study of the System**

System Analysis is the primary stage according to System Development Life Cycle model. This primary begins with the analyst.

The hardware requirement of this project is an external fingerprint sensor i.e. Hamster Pro H20.

The software implementation includes sublime text 3(Version-3207),Microsoft Visual Studio 2019,MySql Server 2008 R2,X-Code(For Testing),Windows Platform.

**1.4 Modules and their Description**

The system has 1 major module and its submodules:

1. **Admin Login**
2. **AddCandidate:**  
   **-> A**dmin can add as manynumbers of candidate dynamically who wanna contest the election.
3. **Add Election:**  
   **-> S**ystem allows admin to add election.
4. **View Election:**  
   **->**After creating an election, the admin would be able to see the details of election expect who cast votes to whom.
5. **Add Voter:**  
   **->**System allows admin to register voters by scanning their thumb impressions scanned and storing their details into the remote database.
6. **View Result:**  
   **->**Post elections admin can view results and can check who won the election.

**1.5Existing System**

* **Problem with the current scenario**

Elections are the strength of democracy, but all too frequently, We never care about the mechanism of the election. Electronic Voting Machines (“EVM") was commissioned by the Chief Election Commissioner in 1977. The EVMs were designed and manufactured by Election Commission of India in collaboration with Bharat Electronics Limited (BEL) and Electronics Corporation of India Limited (ECIL).

EVM consists of two major units, i) Control Unit, ii) Balloting Unit. The two units are interfaced by a five-meter cable. The Control Unit is with the concerned officer assigned and the Balloting Unit will be kept inside the voting compartment.



***Figure 1.1***

There are 2 types of problems with EVM which is currently in use:

1. **Security Problems:** Individual can change the program installed in the EVM and can tamper the results after the polling for the favor of the candidate.
2. **Illegal Voting (Rigging):** Rigging is the most common problem faced by the election commission of India.In which an individual cast more than one vote in a different constituency
3. **Manual adding of candidature:** India is a democratic country,Each and Every individual has right to stand at the election,If a large number of candidates contest the election it will create difficulty to the election commission to conduct elections.Since our project is dynamic, we can add as many candidates we want.

**1.6Proposed System:**

We develop our system in such a manner so that we can eliminate the drawbacks of the existing system and enhance the authentication and verification process of the election.

A pattern recognition system works by the acquired data and comparing the features set against the data stored in the remotely and safe database.So,It checks the illegal and the repetition of votes. Hence the elections would be conducted fair and free from any type of rigging. It will fully automate the process of casting the election and then it will be much easier for the people to cast the votes and elect their favorite candidate and hence it will add power to the running democracy.

**IMPLEMENTATION**

**MODULES:**

* Login
* Registration
* Fingerprint Verficiation
* New Candidate
* Result

**MODULES DESCRIPTION:**

**Login: -**

This module enables the user and admin to login to the system by entering id and password.

**Registration: -**

In this module the admin will verify the user and register the user who will vote.

**Fingerprint Verification: -**

The authenticated user can vote for the candidate for one time for a particular election.

**New Candidate:-**

Admin will add the number of candidates nominated forElection whenever new election is announced.

**Result: -**

Admin and user can view the election result by using the election id once the election results are out.

**INPUT DESIGN AND OUTPUT DESIGN**

**INPUT DESIGN**

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy. Input Design considered the following things:

* What data should be given as input?
* How the data should be arranged or coded?
* The dialog to guide the operating personnel in providing input.
* Methods for preparing input validations and steps to follow when error occur.

**OBJECTIVES**

1. Input Design is the process of converting a user-oriented description of the input into a computer-based system. This design is important to avoid errors in the data input process and show the correct direction to the management for getting correct information from the computerized system.

2. It is achieved by creating user-friendly screens for the data entry to handle large volume of data. The goal of designing input is to make data entry easier and to be free from errors. The data entry screen is designed in such a way that all the data manipulates can be performed. It also provides record viewing facilities.

3. When the data is entered it will check for its validity. Data can be entered with the help of screens. Appropriate messages are provided as when needed so that the user will not be in maize of instant. Thus the objective of input design is to create an input layout that is easy to follow

**OUTPUT DESIGN**

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and intelligent output design improves the system’s relationship to help user decision-making.

1. Designing computer output should proceed in an organized, well thought out manner; the right output must be developed while ensuring that each output element is designed so that people will find the system can use easily and effectively. When analysis design computer output, they should Identify the specific output that is needed to meet the requirements.

2. Select methods for presenting information.

3. Create document, report, or other formats that contain information produced by the system.

The output form of an information system should accomplish one or more of the following objectives.

* Convey information about past activities, current status or projections of the
* Future.
* Signal important events, opportunities, problems, or warnings.
* Trigger an action.
* Confirm an action.

**Software Environment**

4.1 Features OF. Net:

Microsoft .NET is a set of Microsoft software technologies for rapidly building and integrating XML Web services, Microsoft Windows-based applications, and Web solutions. The .NET Framework is a language-neutral platform for writing programs that can easily and securely interoperate. There’s no language barrier with .NET: there are numerous languages available to the developer including Managed C++, C#, Visual Basic and Java Script. The .NET framework provides the foundation for components to interact seamlessly, whether locally or remotely on different platforms. It standardizes common data types and communications protocols so that components created in different languages can easily interoperate.

“.NET” is also the collective name given to various software components built upon the .NET platform. These will be both products (Visual Studio.NET and Windows.NET Server, for instance) and services (like Passport, .NET My Services, and so on).

**THE .NET FRAMEWORK**

The .NET Framework has two main parts:

1. The Common Language Runtime (CLR).

2. A hierarchical set of class libraries.

The CLR is described as the “execution engine” of .NET. It provides the environment within which programs run. The most important features are

* Conversion from a low-level assembler-style language, called Intermediate Language (IL), into code native to the platform being executed on.
* Memory management, notably including garbage collection.
* Checking and enforcing security restrictions on the running code.
* Loading and executing programs, with version control and other such features.
* The following features of the .NET framework are also worth description:

**Managed Code**

The code that targets .NET, and which contains certain extra

Information - “metadata” - to describe itself. Whilst both managed and unmanaged code can run in the runtime, only managed code contains the information that allows the CLR to guarantee, for instance, safe execution and interoperability.

**Managed Data**

With Managed Code comes Managed Data. CLR provides memory allocation and Deal location facilities, and garbage collection. Some .NET languages use Managed Data by default, such as C#, Visual Basic.NET and JScript.NET, whereas others, namely C++, do not. Targeting CLR can, depending on the language you’re using, impose certain constraints on the features available. As with managed and unmanaged code, one can have both managed and unmanaged data in .NET applications - data that doesn’t get garbage collected but instead is looked after by unmanaged code.

**Common Type System**

The CLR uses something called the Common Type System (CTS) to strictly enforce type-safety. This ensures that all classes are compatible with each other, by describing types in a common way. CTS define how types work within the runtime, which enables types in one language to interoperate with types in another language, including cross-language exception handling. As well as ensuring that types are only used in appropriate ways, the runtime also ensures that code doesn’t attempt to access memory that hasn’t been allocated to it.

**Common Language Specification**

The CLR provides built-in support for language interoperability. To ensure that you can develop managed code that can be fully used by developers using any programming language, a set of language features and rules for using them called the Common Language Specification (CLS) has been defined. Components that follow these rules and expose only CLS features are considered CLS-compliant.

**THE CLASS LIBRARY**

.NET provides a single-rooted hierarchy of classes, containing over 7000 types. The root of the namespace is called System; this contains basic types like Byte, Double, Boolean, and String, as well as Object. All objects derive from System. Object. As well as objects, there are value types. Value types can be allocated on the stack, which can provide useful flexibility. There are also efficient means of converting value types to object types if and when necessary.

The set of classes is pretty comprehensive, providing collections, file, screen, and network I/O, threading, and so on, as well as XML and database connectivity.

The class library is subdivided into a number of sets (or namespaces), each providing distinct areas of functionality, with dependencies between the namespaces kept to a minimum.

**LANGUAGES SUPPORTED BY .NET**

The multi-language capability of the .NET Framework and Visual Studio .NET enables developers to use their existing programming skills to build all types of applications and XML Web services. The .NET framework supports new versions of Microsoft’s old favorites Visual Basic and C++ (as VB.NET and Managed C++), but there are also a number of new additions to the family.

Visual Basic .NET has been updated to include many new and improved language features that make it a powerful object-oriented programming language. These features include inheritance, interfaces, and overloading, among others. Visual Basic also now supports structured exception handling, custom attributes and also supports multi-threading.

Visual Basic .NET is also CLS compliant, which means that any CLS-compliant language can use the classes, objects, and components you create in Visual Basic .NET.

Managed Extensions for C++ and attributed programming are just some of the enhancements made to the C++ language. Managed Extensions simplify the task of migrating existing C++ applications to the new .NET Framework.

C# is Microsoft’s new language. It’s a C-style language that is essentially “C++ for Rapid Application Development”. Unlike other languages, its specification is just the grammar of the language. It has no standard library of its own, and instead has been designed with the intention of using the .NET libraries as its own.

Microsoft Visual J# .NET provides the easiest transition for Java-language developers into the world of XML Web Services and dramatically improves the interoperability of Java-language programs with existing software written in a variety of other programming languages.

Active State has created Visual Perl and Visual Python, which enable .NET-aware applications to be built in either Perl or Python. Both products can be integrated into the Visual Studio .NET environment. Visual Perl includes support for Active State’s Perl Dev Kit.

Other languages for which .NET compilers are available include

* FORTRAN
* COBOL
* Eiffel

Fig1 **.**Net Framework

|  |  |
| --- | --- |
| ASP.NET  XML WEB SERVICES | Windows Forms |
| Base Class Libraries | |
| Common Language Runtime | |
| Operating System | |

C#.NET is also compliant with CLS (Common Language Specification) and supports structured exception handling. CLS is set of rules and constructs that are supported by the CLR (Common Language Runtime). CLR is the runtime environment provided by the .NET Framework; it manages the execution of the code and also makes the development process easier by providing services.

C#.NET is a CLS-compliant language. Any objects, classes, or components that created in C#.NET can be used in any other CLS-compliant language. In addition, we can use objects, classes, and components created in other CLS-compliant languages in C#.NET .The use of CLS ensures complete interoperability among applications, regardless of the languages used to create the application.

**CONSTRUCTORS AND DESTRUCTORS:**

Constructors are used to initialize objects, whereas destructors are used to destroy them. In other words, destructors are used to release the resources allocated to the object. In C#.NET the sub finalize procedure is available. The sub finalize procedure is used to complete the tasks that must be performed when an object is destroyed. The sub finalize procedure is called automatically when an object is destroyed. In addition, the sub finalize procedure can be called only from the class it belongs to or from derived classes.

**GARBAGE COLLECTION**

Garbage Collection is another new feature in C#.NET. The .NET Framework monitors allocated resources, such as objects and variables. In addition, the .NET Framework automatically releases memory for reuse by destroying objects that are no longer in use.

In C#.NET, the garbage collector checks for the objects that are not currently in use by applications. When the garbage collector comes across an object that is marked for garbage collection, it releases the memory occupied by the object.

**OVERLOADING**

Overloading is another feature in C#. Overloading enables us to define multiple procedures with the same name, where each procedure has a different set of arguments. Besides using overloading for procedures, we can use it for constructors and properties in a class.

**MULTITHREADING:**

C#.NET also supports multithreading. An application that supports multithreading can handle multiple tasks simultaneously, we can use multithreading to decrease the time taken by an application to respond to user interaction.

**STRUCTURED EXCEPTION HANDLING**

C#.NET supports structured handling, which enables us to detect and remove errors at runtime. In C#.NET, we need to use Try…Catch…Finally statements to create exception handlers. Using Try…Catch…Finally statements, we can create robust and effective exception handlers to improve the performance of our application.

**THE .NET FRAMEWORK**

The .NET Framework is a new computing platform that simplifies application development in the highly distributed environment of the Internet.

**OBJECTIVES OF. NET FRAMEWORK**

1. To provide a consistent object-oriented programming environment whether object codes is stored and executed locally on Internet-distributed, or executed remotely.

2. To provide a code-execution environment to minimizes software deployment and guarantees safe execution of code.

3. Eliminates the performance problems.

There are different types of application, such as Windows-based applications and Web-based applications.

4.3 Features of SQL-SERVER

The OLAP Services feature available in SQL Server version 7.0 is now called SQL Server 2000 Analysis Services. The term OLAP Services has been replaced with the term Analysis Services. Analysis Services also includes a new data mining component. The Repository component available in SQL Server version 7.0 is now called Microsoft SQL Server 2000 Meta Data Services. References to the component now use the term Meta Data Services. The term repository is used only in reference to the repository engine within Meta Data Services

SQL-SERVER database consist of six type of objects,

They are,

1. TABLE

2. QUERY

3. FORM

4. REPORT

5. MACRO

**TABLE:**

A database is a collection of data about a specific topic.

**VIEWS OF TABLE:**

We can work with a table in two types,

1. Design View

2. Datasheet View

**Design View**

To build or modify the structure of a table we work in the table design view. We can specify what kind of data will be hold.

**Datasheet View**

To add, edit or analyses the data itself we work in tables datasheet view mode.

**QUERY:**

A query is a question that has to be asked the data. Access gathers data that answers the question from one or more table. The data that make up the answer is either dynaset (if you edit it) or a snapshot (it cannot be edited).Each time we run query, we get latest information in the dynaset. Access either displays the dynaset or snapshot for us to view or perform an action on it, such as deleting or updating.

**SYSTEM DESIGN**

**DATA FLOW DIAGRAM:**

1. The DFD is also called as bubble chart. It is a simple graphical formalism that can be used to represent a system in terms of input data to the system, various processing carried out on this data, and the output data is generated by this system.
2. The data flow diagram (DFD) is one of the most important modeling tools. It is used to model the system components. These components are the system process, the data used by the process, an external entity that interacts with the system and the information flows in the system.
3. DFD shows how the information moves through the system and how it is modified by a series of transformations. It is a graphical technique that depicts information flow and the transformations that are applied as data moves from input to output.
4. DFD is also known as bubble chart. A DFD may be used to represent a system at any level of abstraction. DFD may be partitioned into levels that represent increasing information flow and functional detail.

Add Election

Add Candidates

Add Voters

Voters cast Vote

Calculate Result

Announce Result

View Results

**UML DIAGRAMS**

UML stands for Unified Modeling Language. UML is a standardized general-purpose modeling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating models of object oriented computer software. In its current form UML is comprised of two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML.

The Unified Modeling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business modeling and other non-software systems.

The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems.

The UML is a very important part of developing objects oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

**GOALS:**

The Primary goals in the design of the UML are as follows:

1. Provide users a ready-to-use, expressive visual modeling Language so that they can develop and exchange meaningful models.
2. Provide extendibility and specialization mechanisms to extend the core concepts.
3. Be independent of particular programming languages and development process.
4. Provide a formal basis for understanding the modeling language.
5. Encourage the growth of OO tools market.
6. Support higher level development concepts such as collaborations, frameworks, patterns and components.
7. Integrate best practices.

**USE CASE DIAGRAM:**

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.

Admin

**CLASS DIAGRAM:**

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.

Admin

Login

Create Elections() Add Candidates () Add Voters () Check voters Biometric () Voters cast vote() Calculate Result () Announce Result () View Results ()

**SEQUENCE DIAGRAM:**

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.

Admin

Database

Add Elections

Add Candidates

Add Voters

Check Voters Biometric

Voters Cast Vote

Calculate Result

Announce Result

View Election Results

**ACTIVITY DIAGRAM:**

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control

Not Matched

Matched

Start

Add Elections

Add Candidates

Add Voters

Voters Cast Votes

Check Biometric

Calculate Result

Not Allowed

Announce Result

View Results

**SYSTEM REQUIREMENTS:**

**HARDWARE REQUIREMENTS:**

* System : Pentium IV 2.4 GHz.
* Hard Disk : 40 GB.
* Floppy Drive : 1.44 Mb.
* Monitor : 15 VGA Colour.
* Mouse : Logitech.
* Ram : 1GB

**SOFTWARE REQUIREMENTS:**

* Operating system : - Windows XP/7.
* Coding Language : C#.NET
* Data Base : MS SQL SERVER 2005

**SYSTEM STUDY**

**FEASIBILITY STUDY:**

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are

* ECONOMICAL FEASIBILITY
* TECHNICAL FEASIBILITY
* SOCIAL FEASIBILITY

**ECONOMICAL FEASIBILITY**

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

### TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

**SOCIAL FEASIBILITY**

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

### SYSTEM TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the

Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

**TYPES OF TESTS:**

**Unit testing**

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

**Integration testing**

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

**Functional test**

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted.

Invalid Input : identified classes of invalid input must be rejected.

Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised.

Systems/Procedures: interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

**System Test**

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

**White Box Testing**

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

**Black Box Testing**

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

**6.1 Unit Testing:**

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

**Test strategy and approach**

Field testing will be performed manually and functional tests will be written in detail.

**Test objectives**

* All field entries must work properly.
* Pages must be activated from the identified link.
* The entry screen, messages and responses must not be delayed.

**Features to be tested**

* Verify that the entries are of the correct format
* No duplicate entries should be allowed

All links should take the user to the correct page.

# 6.2 Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

**Test Results:** All the test cases mentioned above passed successfully. No defects encountered.

**6.3 Acceptance Testing**

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

**Test Results:** All the test cases mentioned above passed successfully. No defects encountered.

**SYSTEM ANALYSIS**

**EXISTING SYSTEM:**

The voter enters the machine and pulls a lever to close the curtain to unlocking the voting levers. The pollers then make his or her selection from a list of switches denoting the appropriate candidates or measures. The machine is configured to prevent over votes by locking out other candidates when one candidate's switch is flipped once the voter is finished and the lever is pulled which opens the curtain and increments the appropriate counters for each candidate and measure also the results are then hand written by the precinct officer at the conclusion of voting.

**DISADVANTAGES OF EXISTING SYSTEM:**

* Complex voting procedure
* More time to vote
* Recount of voting is not possible
* Expensive to test, complete tests are extremely rare
* Expensive to move and store
* Difficult to test
* Complex to maintain
* Far from secure against vote fraud.

**PROPOSED SYSTEM:**

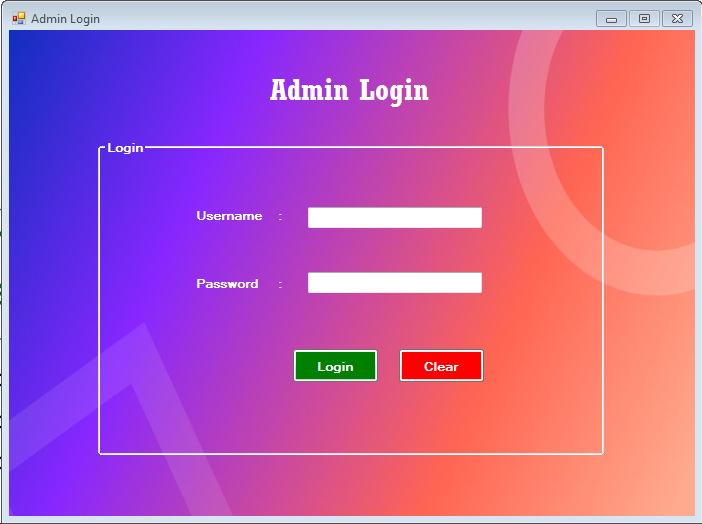
With the aim of conducting democratic election, we proposed the system to Endeavour to improve the easy usage of the voting machine with authentication and an acknowledgement slip will provide for every polling which occurs. And a touch screen is used to give input so it is so highly easy to overcome the button problem.

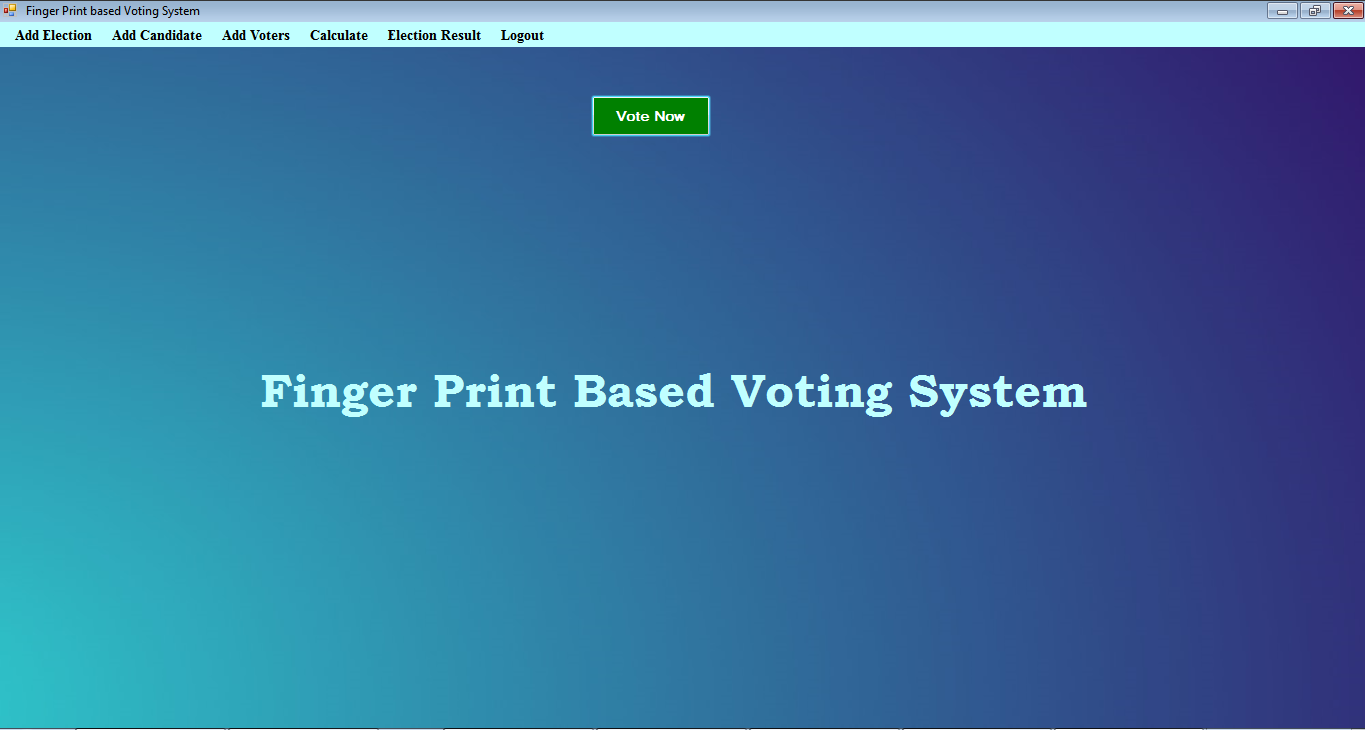
The system is free from intentional tamper. It is not possible to hack the system. Though this factor depends on the personnel integrity, attempts should be made to make the model as secure as possible. In this machine every user uses his/her finger print. The votes will be successful only after successful verification of their finger print.

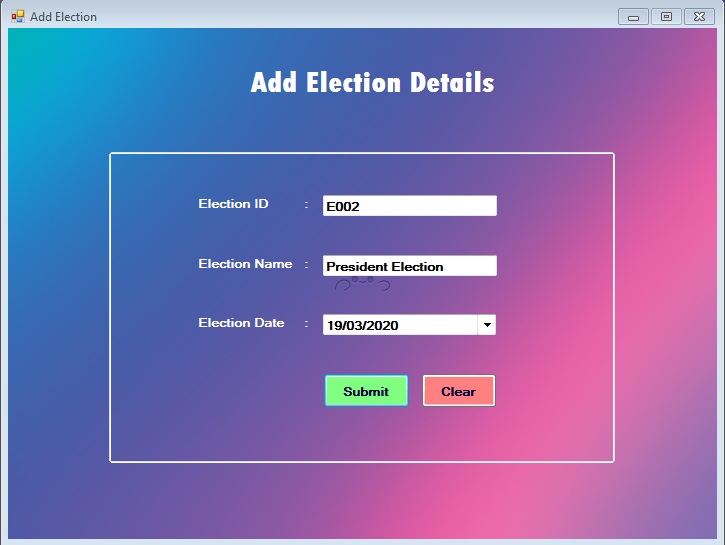
**ADVANTAGES OF PROPOSED SYSTEM:**

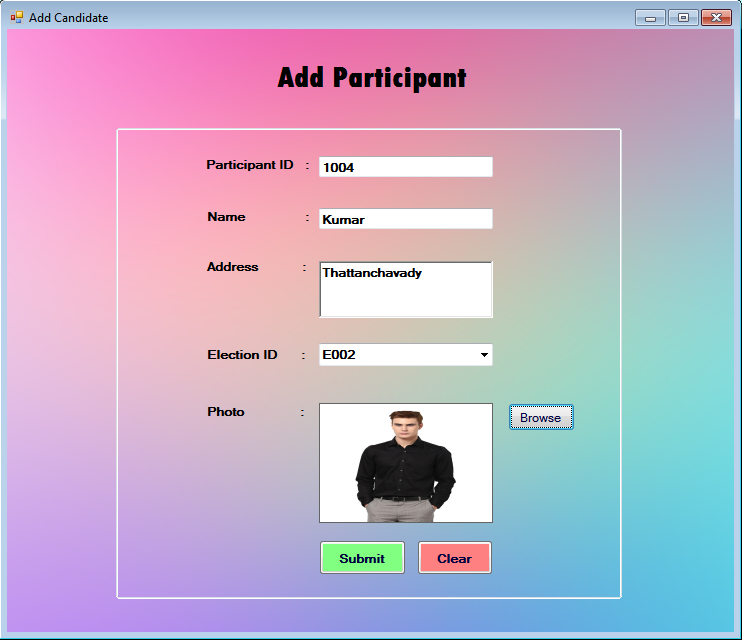
* From the existing system, the proposed system is more in the Security,Reliability,Scalability factors.
* The system will not allow the voter to vote two or more candidates.
* The system will allow the user to vote for one time for a particular election.
* The system will authenticate the user through his fingerprint so the user is uniquely identified.

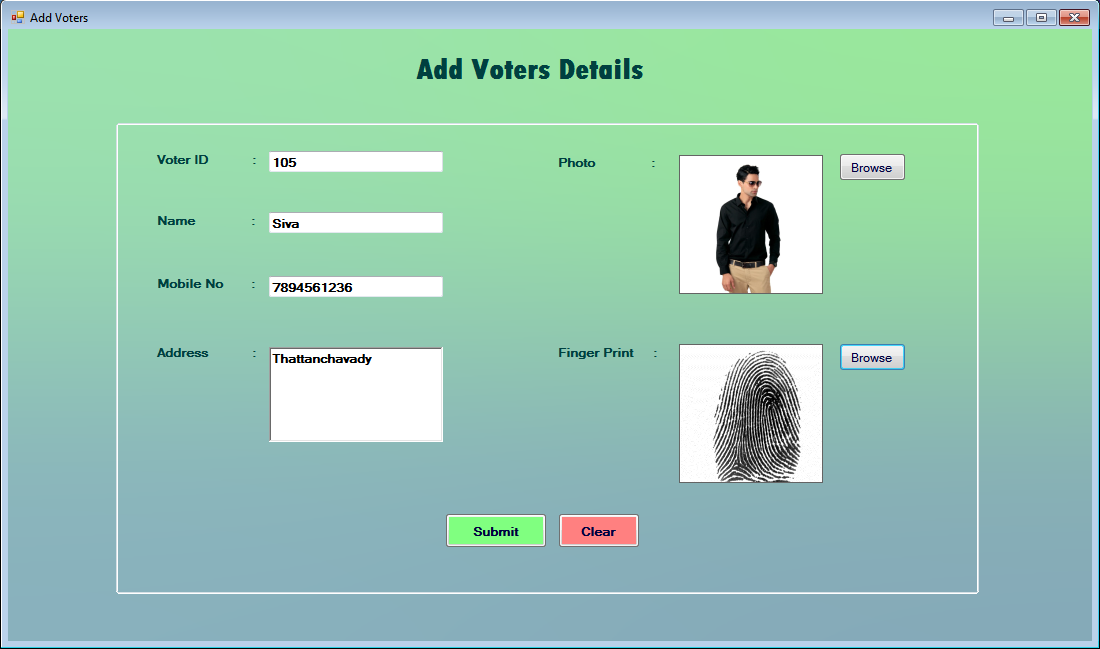
**SCREEN SHOTS**

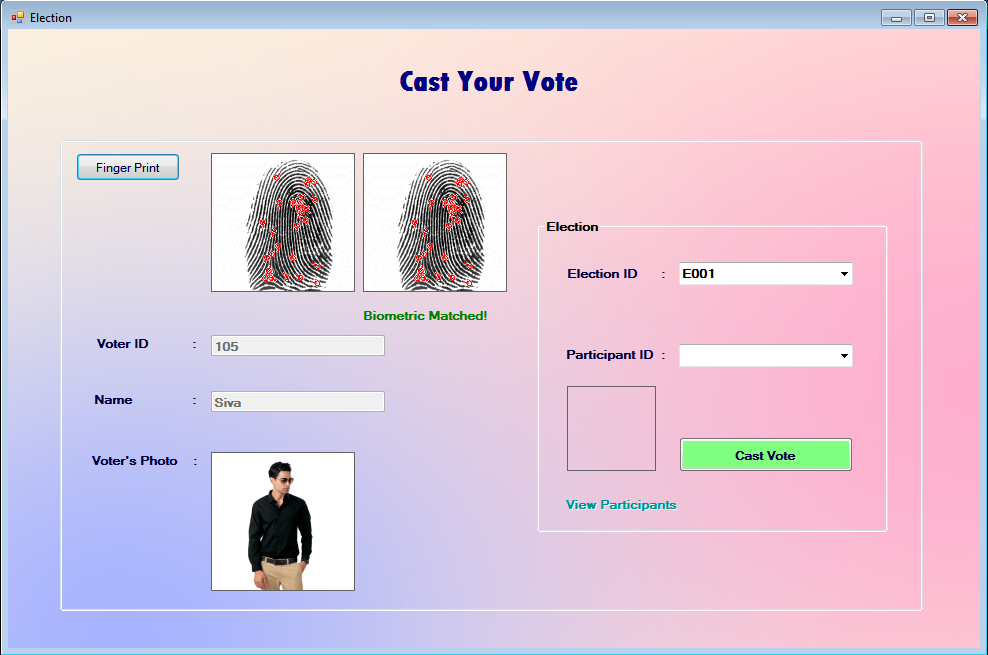


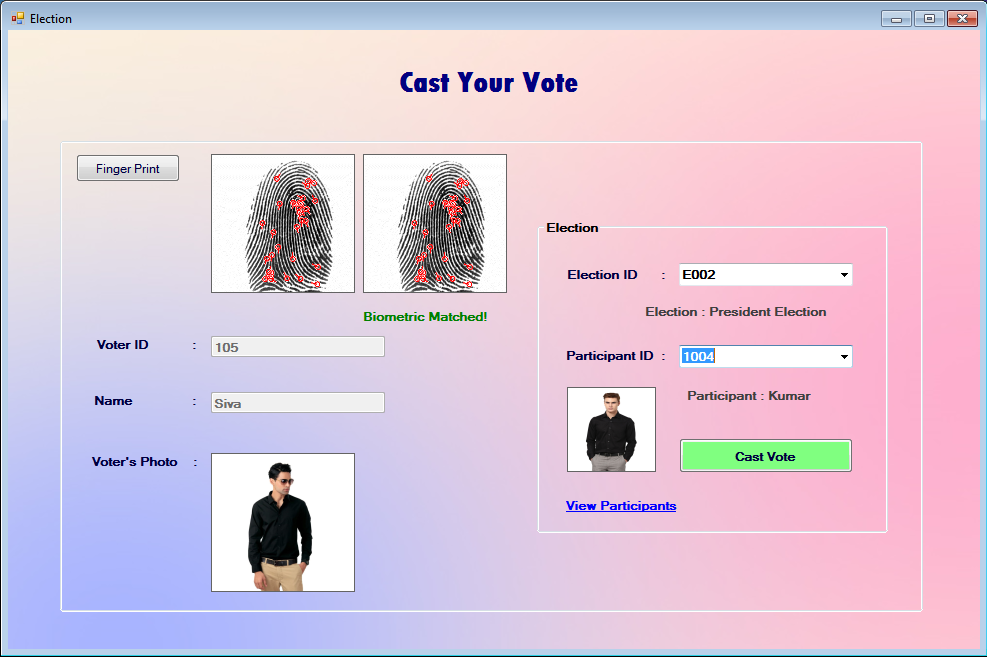


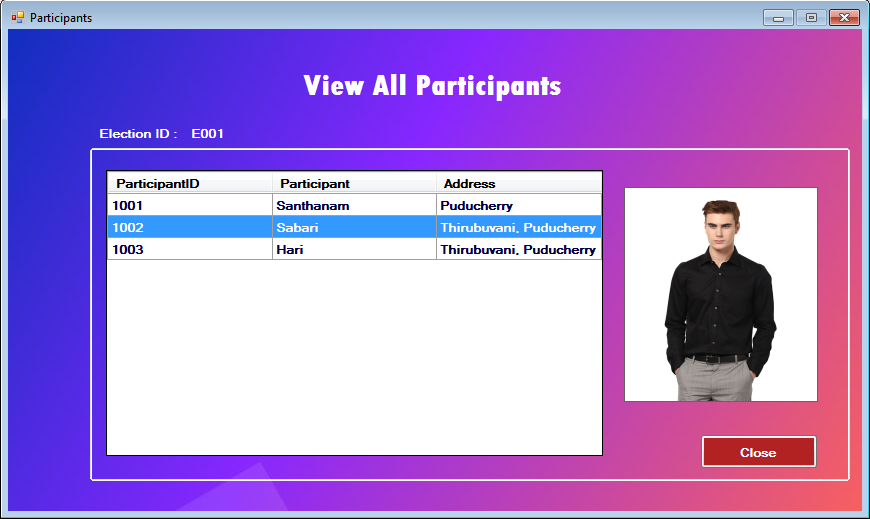


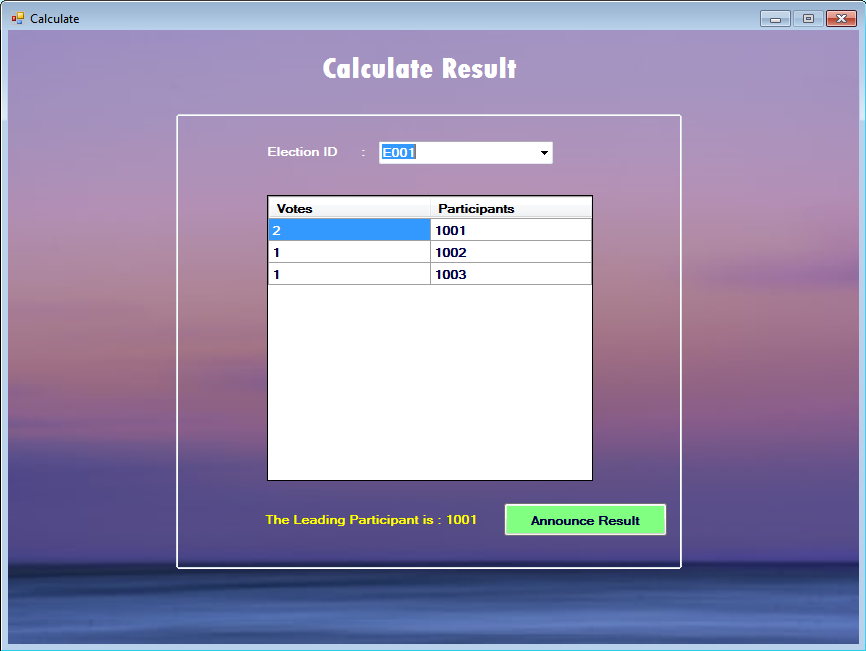


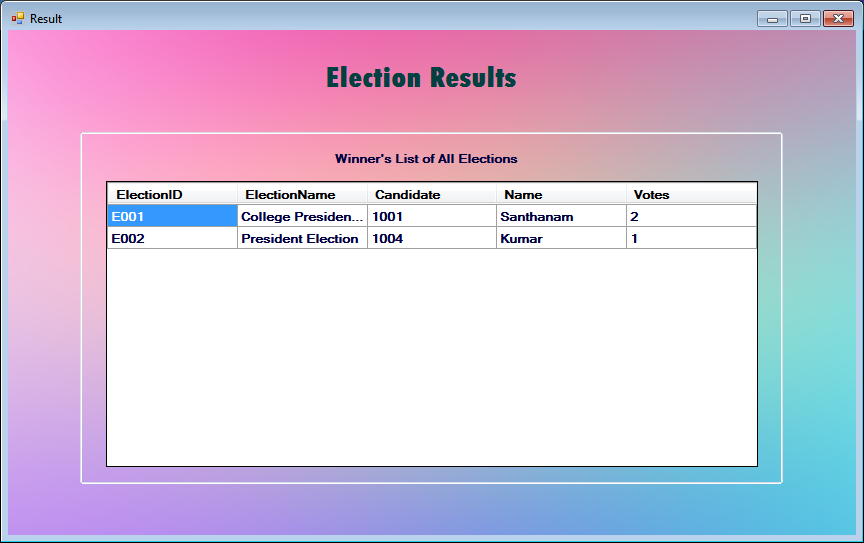












**LITERATURE SURVEY**

**1) A survey of cloud storage facilities**

**AUTHORS:**  H. Dewan and R. C. Hansdah

There are many applications such as software for processing customer records in telecom, patient records in hospitals, email processing software accessing a single email in a mailbox etc. which require to access a single record in a database consisting of millions of records. A basic feature of these applications is that they need to access data sets which are very large but simple. Cloud computing provides computing requirements for these kinds of new generation of applications involving very large data sets which cannot possibly be handled efficiently using traditional computing infrastructure. In this paper, we describe storage services provided by three well-known cloud service providers and give a comparison of their features with a view to characterize storage requirements of very large data sets as examples and we hope that it would act as a catalyst for the design of storage services for very large data set requirements in future. We also give a brief overview of other kinds of storage that have come up in the recent past for cloud computing.

**2) Security challenges for the public cloud**

**AUTHORS:** K. Ren, C. Wang, and Q. Wang

Cloud computing represents today's most exciting computing paradigm shift in information technology. However, security and privacy are perceived as primary obstacles to its wide adoption. Here, the authors outline several critical security challenges and motivate further investigation of security solutions for a trustworthy public cloud environment.

**3) Provable multicopy dynamic data possession in cloud computing systems**

**AUTHORS:** A. F. Barsoum and M. A. Hasan

Increasingly more and more organizations are opting for outsourcing data to remote cloud service providers (CSPs). Customers can rent the CSPs storage infrastructure to store and retrieve almost unlimited amount of data by paying fees metered in gigabyte/month. For an increased level of scalability, availability, and durability, some customers may want their data to be replicated on multiple servers across multiple data centers. The more copies the CSP is asked to store, the more fees the customers are charged. Therefore, customers need to have a strong guarantee that the CSP is storing all data copies that are agreed upon in the service contract, and all these copies are consistent with the most recent modifications issued by the customers. In this paper, we propose a map-based provable multicopy dynamic data possession (MB-PMDDP) scheme that has the following features: 1) it provides an evidence to the customers that the CSP is not cheating by storing fewer copies; 2) it supports outsourcing of dynamic data, i.e., it supports block-level operations, such as block modification, insertion, deletion, and append; and 3) it allows authorized users to seamlessly access the file copies stored by the CSP. We give a comparative analysis of the proposed MB-PMDDP scheme with a reference model obtained by extending existing provable possession of dynamic single-copy schemes. The theoretical analysis is validated through experimental results on a commercial cloud platform. In addition, we show the security against colluding servers, and discuss how to identify corrupted copies by slightly modifying the proposed scheme.

**4) Dynamic and public auditing with fair arbitration for cloud data**

**AUTHORS:** H. Jin, H. Jiang, and K. Zhou

Cloud users no longer physically possess their data, so how to ensure the integrity of their outsourced data becomes a challenging task. Recently proposed schemes such as “provable data possession” and “proofs of retrievability” are designed to address this problem, but they are designed to audit static archive data and therefore lack of data dynamics support. Moreover, threat models in these schemes usually assume an honest data owner and focus on detecting a dishonest cloud service provider despite the fact that clients may also misbehave. This paper proposes a public auditing scheme with data dynamics support and fairness arbitration of potential disputes. In particular, we design an index switcher to eliminate the limitation of index usage in tag computation in current schemes and achieve efficient handling of data dynamics. To address the fairness problem so that no party can misbehave without being detected, we further extend existing threat models and adopt signature exchange idea to design fair arbitration protocols, so that any possible dispute can be fairly settled. The security analysis shows our scheme is provably secure, and the performance evaluation demonstrates the overhead of data dynamics and dispute arbitration are reasonable.

**5) Privacy-preserving public auditing for shared cloud data supporting group dynamics**

**AUTHORS:** B. Wang, H. Li, and M. Li

In the cloud, data is often shared by a group of users. To ensure the long-term correctness of cloud shared data, a third-party public verifier can be introduced to audit data integrity. During the auditing, protecting the privacy of the contributors of shared data from the public auditor is a fundamental issue. However, this makes it challenging to simultaneously support group membership dynamics efficiently, due to the significant amount of computation needed to update the signatures on shared data. In this paper, we propose a novel privacy-preserving public auditing mechanism for shared cloud data. With our proposed mechanism, a public verifier is able to audit the integrity of shared data without retrieving the entire data from the cloud, and also without learning private identity information of the group members. Group dynamics (user join and user revocation) are efficiently handled by outsourcing signature updating operations to the cloud via a secure proxy re-signature scheme. Experimental results show that our mechanism is highly efficient for dynamic groups.

**CONCLUSION**

The package was designed in such a way that future modifications can be done easily. The following conclusions can be deduced from the development of the project.

Automation of the entire system improves the efficiency

It provides a friendly graphical user interface which proves to be better when compared to the existing system.

It gives appropriate access to the authorized users depending on their permissions.

It effectively overcomes the delay in communications.

Updating of information becomes so easier.

System security, data security and reliability are the striking features.

The System has adequate scope for modification in future if it is necessary.

**FUTURE ENHANCEMENTS:**

It is an easy way and an effective way for voting which allows the reduction of travelling,waiting in a long queues avoid unwanted votes and duplicated votes.And we can get the accurate number of votes.

Well I and my team member have worked hard in order to present an improved website better than the existing one’s regarding the information about the various activities. Still,we found out that the project can be done in a better way. Primarily, when we register we get an unique id by using that unique id we allowed to login and allowed to vote .Each and every user are allowed to vote only once nobody is allowed to vote more than once and only the registered persons only can vote.The candidates also gets registered by getting a unique id and he can compete from anywhere and can see the results.

The next enhancement that we can add different kinds of buttons for easy access and easy availability to each and every one in remote areas.These are the enhancements that we could think of at present.

**REFERENCES**

The following books were referred during the analysis and execution phase of the project

MICROSOFT .NET WITH C#

Microsoft .net series

ASP .NET 2.0 PROFESSIONAL

Wrox Publishers

ASP .NET WITH C# 2005

Apress Publications

C# COOK BOOK

O reilly Publications

PROGRAMMING MICROSOFT ASP .NET 2.0 APPLICATION

Wrox Professional Guide

BEGINNING ASP .NET 2.0 E-COMMERCE IN C# 2005

Novice to Professional.

WEBSITES:

www.google.com

www.microsoft.com