

# **Online portal for e-voting and opinion polls**

**N. D. Dahanayake  
2023**



# **Online portal for e-voting and opinion polls**

**A thesis submitted for the Degree of Master of  
Information Technology**

**N. D. Dahanayake  
University of Colombo School of Computing  
2023**



# Declaration

The thesis is my original work and has not been submitted previously for a degree at this or any other university/institute.

To the best of my knowledge, it does not contain any material published or written by another person, except as acknowledged in the text.

Student Name: Nimesh Dilhan Dahanayake

Registration Number: 2019/MIT/008

Index Number: 19550081



Signature: \_\_\_\_\_

Date: 23/02/2023

This is to certify that this thesis is based on the work of

Mr. Nimesh Dilhan Dahanayake

under my supervision. The thesis has been prepared according to the format stipulated and is of an acceptable standard.

Certified by:

Supervisor Name: Prof. K P Hewagamage



Signature: \_\_\_\_\_

Date

# Abstract

Many public voting and referendum administration systems are still operated by hand, which is an issue nowadays. The vast majority of tasks must be carried out by hand, which increases the likelihood of human mistakes. Partners want outcomes much more quickly now than they did in the past. As a result, they want full openness throughout the whole voting process, not simply the count of the votes. The poll management systems will automate all of these functions.

The point of this study is to plan an electronic voting framework based on the discretionary procedure embraced in Sri Lanka and to create a secure authentication system for voting. This system would be fulfilling a prudent requirement in the current Sri Lankan society. The voters of the system are allowed to vote with the use of their own NIC number, Email, mobile number, and OTP. Then, those votes are recorded in the database. Then the internet acts as the medium interconnecting all the voters in different locations and the administrators. The administrator is provided with the privilege to log on to the system from his account created and to analyze the votes cast by the voters. The system is a new development to the current market software since the security, privacy (can't trace voters), and validation features are achieved interestingly and solidly which allows the users to be more confident in using the system

This system has been designed as a web-based application and MVC architecture is used. Cutting-edge technologies have been selected to develop the entire system. The system will be run with PHP as the backend scripting language, MySQL as the database server, and Apache as the web server. HTML and CSS were chosen to design the user interface of the system and also selected Bootstrap for an attractive look and responsive interface design for all applicable devices. JSON and JavaScript libraries such as JQuery were chosen for dynamic web content. Codeigniter 3, Adobe Photoshop CS6, and Visual studio code are selected as the development tool.

After project implementation, various levels of testing were conducted including user acceptance testing. During project testing, the system was put into use in a live environment. The system was then evaluated using a user evaluation process to obtain user feedback and make any necessary modifications. The majority of users were content with the system, and there is no negative feedback.

# Acknowledgment

First and foremost, I would like to express my sincere gratitude to my project supervisor, Prof. K P Hewagamage for spending his valuable time advising, guiding, encouraging, and supporting me to accomplish this project successfully. Also, I would like to take this opportunity again to thank Prof. K P Hewagamage for imparting his theoretical and technical knowledge.

I would also like to extend my eternal gratitude to all other lectures of UCSC for helping me to complete my final year individual project by providing a better environment.

Moreover, I express my appreciation to my friends and parents for encouraging and helping me during the project implementations. Finally, I highly appreciate everybody who supports me in various types to complete my final year individual project.

# List of Acronyms

NIC - National Identity Card

EVM - Electric Voting Machine

DRE - Direct Recording Electronics

UML - Unified Modeling Language

UI - User Interface

API - Application Programming Interface

CSS - Cascading Style Sheets

AJAX - Asynchronous JavaScript and XML

SDM - System Development Method

MVC - Model-view-controller

ER - Entity Relationship

# Table of Contents

## Contents

Declaration .....	i
Abstract .....	ii
Acknowledgment .....	iii
List of Acronyms .....	iv
Table of Contents .....	v
Table of Figures .....	vii
List of Tables .....	viii
Chapter 1 - Introduction .....	1
1.1 Project Overview .....	1
1.2 Objectives .....	3
1.3 Background of the study .....	4
1.4 Scope of the study .....	5
1.5 Structure of the Dissertation .....	6
Chapter 2 - Background .....	7
2.1 Introduction .....	7
2.2 Requirement Analysis .....	8
2.2.1 Functional Requirements .....	8
2.2.2 Non-Functional Requirements .....	9
2.3 Review of Similar Systems .....	10
2.4 Related Technologies .....	14
Chapter 3 - Design .....	15
3.1 Introduction .....	15
3.2 Related design strategies .....	16
3.2.1 Object-Oriented Analysis and Designing .....	16
3.2.2 MVC architecture .....	16
3.3 System Architecture .....	17
3.4 UML diagrams .....	18
3.4.1 Use Case Diagram .....	18
3.4.2 Activity Diagram .....	21
3.4.3 Sequence diagram .....	23
3.5 Database Designing .....	25
3.6 User Interface Designing .....	26
Chapter 4 - Implementation .....	29

4.1 Introduction .....	29
4.2 Implementation Environment .....	29
4.3 System Development Tools and Technologies .....	30
4.4 Major Code Segments.....	32
4.4.1 Database Connection .....	32
4.4.2 Login Handling Page .....	34
4.4.3 Email validation.....	38
4.4.4 Vote page validation .....	38
4.4.5 Invitation sending process .....	41
4.4.6 Result sending process.....	43
4.4.7 Report Generation.....	43
4.5 Reusable Components .....	44
<b>Chapter 5 – Testing and Evaluation.....</b>	<b>45</b>
5.1 Related Testing Types .....	45
5.1.1 Unit Testing .....	45
5.1.2 Regression Testing.....	45
5.1.3 Integrated Testing .....	46
5.1.4 User Acceptance Test .....	46
5.2 Test Cases .....	47
5.3 User Evaluation .....	51
5.4 Results of the Testing .....	52
<b>Chapter 6 - Conclusion .....</b>	<b>53</b>
6.1 Introduction .....	53
6.2 Lesson Learnt .....	54
6.3 Critical Assessment of the project .....	54
6.4 System Weaknesses .....	55
6.5 Future Enhancement .....	55
<b>List of References .....</b>	<b>57</b>
<b>Appendix A – User Documentation.....</b>	<b>59</b>
<b>Appendix B – MIS Reports.....</b>	<b>70</b>
Description of MIS Report Module .....	70
Decision Making.....	70
Tracking records .....	70
Evidence of the reports .....	71
User wise reports .....	71
Poll wise reports .....	72
Decision wise reports.....	73



# Table of Figures

Figure 1 - Systems Development Method .....	14
Figure 2 - System Architecture Diagram .....	17
Figure 3 - Use Case Diagram.....	19
Figure 4 - Activity Diagram 1 .....	21
Figure 5 - Activity diagram 2 .....	22
Figure 6 - Sequence Diagram 1 .....	23
Figure 7 - Sequence Diagram 2 .....	24
Figure 8 - ER Diagram .....	25
Figure 9 - Sign up and Login Prototype .....	26
Figure 10 - Update User Prototype .....	27
Figure 11 - Poll Creation Prototype.....	27
Figure 12 - Add Answers Prototype .....	28
Figure 13 - Voting Page Prototype .....	28
Figure 14 - Database Connection Code .....	33
Figure 15 - OTP Sending Code 1.....	34
Figure 16 - OTP Sending Code 2.....	35
Figure 17 - OTP Sending Code 3.....	35
Figure 18 - OTP Generate Code .....	36
Figure 19 - OTP Verification Code .....	36
Figure 20 - User Registration Code .....	37
Figure 21 – Death Status Validation Code .....	38
Figure 22 - Vote Page Validation Code 1 .....	39
Figure 23 - Vote Page Validation Code 2.....	39
Figure 24 - Vote Page Validation Code 3.....	40
Figure 25 - Invitation Sending Code 1.....	41
Figure 26 - Invitation Sending Code 2.....	42
Figure 27- Invitation Sending Code 3.....	42
Figure 28 – Poll Report Generation Code .....	43
Figure 29 - User Report Generation Code .....	44
Figure 30 - Pie chart of the evaluation summary .....	52
Figure 31 - Sign Up Page .....	59
Figure 32 - Login Page .....	59
Figure 33 - OTP Entering Page .....	60
Figure 34 - Dashboard Page 1 .....	60
Figure 35 - Dashboard Page 2 .....	61
Figure 36 - Poll Selecting Page .....	61
Figure 37 - Voting Page.....	62
Figure 38 - user Management Page .....	63
Figure 39 - Dropdown Page.....	63
Figure 40 - User Data Update Page .....	64
Figure 41 - User Additional Data Update Page .....	65
Figure 42 - Poll Management Page.....	66
Figure 43 - Invitation Sending Page .....	67
Figure 44 - Poll Creation Button .....	68
Figure 45 - Poll Creation Page.....	68
Figure 46 - Question Creation Page.....	69
Figure 47 - Answer Adding Page .....	69
Figure 48 – User-wise Report Page .....	71
Figure 49 - User-wise Graph Page.....	71
Figure 50 - Poll Result Page .....	72
Figure 51 - Poll Report Page .....	72
Figure 52 - File Generation Buttons .....	73

# List of Tables

Table 1 - Use case scenarios.....	20
Table 2 - Implementation Requirements .....	29
Table 3 - Important Test Cases .....	50
Table 4 - summary of the user evaluation result.....	51
Table 5 - Test execution status .....	52

# Chapter 1 - Introduction

## 1.1 Project Overview

Digital transformation has an impact on the process of registering to vote. Even while it's garnered very little public discussion, the issue is starting to get some traction. Voting registration is the better monitoring procedure that ensures that only individuals who are entitled to vote can cast their ballots at the polls. It's not uncommon for the final steps in the voter information process to take place on Election Day. Before Election Day, residents register to vote and their voting credentials are verified at the polls on Election Day.

Decentralization and millions of voters make voter registration a complicated endeavor. Voting registration has always been a municipal and election authority responsibility. State governments must now adopt obligations formerly handled by campaigning authorities under the Help Sri Lanka Vote Act (Parliamentary Elections Act | Volume VI, 2021). Specifically, Committee mandates that the states adopt a "single, consistent, and standardized" voting IT system for Sri Lanka elections.

If they have a digital presence, anybody can push the "Like" button on just about everything that has a presence online. It was estimated in January 2021 that there was 50.8 percent of Sri Lankans had internet access, and there was 141.7 percent of Sri Lankans had access to a smartphone (Digital in Sri Lanka: All the Statistics You Need in 2021 — DataReportal – Global Digital Insights, 2021). The election strategic system in Sri Lanka is still an expensive and time-consuming manual procedure, even though more and more services are migrating online. Many public voting and referendum administration systems are still operated by hand, which is an issue nowadays. The vast majority of tasks must be carried out by hand, which increases the likelihood of human mistakes. Partners want outcomes much more quickly now than they did in the past. As a result, they want full openness throughout the whole voting process, not simply the count of the ballots. The referendum management systems will automate all of these functions. Realistically, we will have to transition to the voting process to execute the referendum because of how quickly services are moving online and technical advancement.

It is important for applicants to keep eligible people off the register lists and to ensure that those who are on the lists are not removed or re-registered. In addition to registering new voters, a third

activity occurs regularly when people reach voting age or relocate to a new neighborhood and desire to participate in the democratic process. On the other hand, the process of registering swing voters is done on a "retail" basis, whilst the purging function is done "wholesale."

Voter purging is necessary because those who have been recognized as legal voters may lose their status due to various circumstances. Because of changes in domicile or redistricting, a registered voter may no longer be eligible to vote in some electoral operations. Individuals registered to vote in many jurisdictions can only cast ballots in the jurisdiction where they can do so lawfully.

The cleansing of a voter information list should be at a minimum somewhat automated because there are millions of registered voters. As a result, the use of computers is designed to verify a great deal of information obtained from other reliable information against the will of the database of people eligible to vote to determine whether a person possesses to be qualified for voting.

## 1.2 Objectives

It is a national requirement to have a system where Sri Lankan citizens could register for online voting for public elections, referendums, and opinion polls of specified matters. In this project, a prototype system will be developed allowing Sri Lankan citizens to register for voting using their public, private and confidential details. This Project will demonstrate the sample opinion polls.

- O1** Users will be able to register with a variety of verification options in the public voting and election monitoring platform to secure and validate their participation in government-related activities.
  - The telephone number will be validated by sending OTP to the voter.
  - The voter's location will be validated using the current location of the voter by obtaining election division from a simple government database.
  - The voter's National identity card number and death status will be validated using the sample government database.
- O2** Provide a low-cost, secured and centralized data store to store data and other files of the opinion polls.
- O3** Reduce human errors, and processing time and increase security, and accuracy by automating the vote-counting process.
- O4** Let users know the real-time status of the opinion polls to get ideas about their polls.
- O5** Reduce corruption and assaults on the voting process by automating the whole voting process.
- O6** Enhance the administrator's decision-making process by generating highly reliable MIS reports.

### **1.3 Background of the study**

We can mainly identify three types of online voting systems. The first one is ‘polling site internet voting’, here voter can cast their votes through the internet but client devices are physically stored in authorized polling places. All the software and hardware are controlled by the election authority and voter authentication is done generally. The second one is ‘Kiosk Internet Voting’, where the voter can cast ballots through client machines, and software are controlled by election officials but machines are stored in public places. Physical surroundings and voter verification are not directly under the election authority. The last one is ‘Remote Internet Voting’, in which the proposed system is done using this method. In this method, both the client machine and the physical environment are under the client. Voters can authenticate on their own and cast votes at their convenience. Compared to the first two methods, the last one is more vulnerable to security issues. But for the opinion polls, the last one is best because it is very convenient and easy to access the voters. (Internet Voting —, 2021)

The IEEE online voting system can be utilized for casting votes during the election period which is held in schools, etc. proposed system uses the voter’s NIC number to log into the system but this IEEE online voting system will provide a secret voter ID that voters can use to login into the system. Similar to the proposed system this IEEE system uses email validation and Mobile phone number validation using OTP (ONLINE VOTING SYSTEM - The IEEE Maker Project, 2021).

## **1.4 Scope of the study**

The following features of an ‘Online portal for e-voting and opinion polls’ are being considered for implementation in a PHP-based web environment.

### **User management**

This module will allow the creation of user accounts and will take care of all operations related to the users

- Create a safe and authenticating facility for users to verify their NIC number and password when logging into the system.
- To guarantee that the registration details of the users are genuine, use validation mechanisms (mobile, email, location validation).
- Set up a database to keep track of all of the many ways people can participate in the system and their personal and professional information.

### **Administration management**

The main purpose of the module is to handle administrative data and administrative activities more conveniently.

- For administrators to successfully administer the system, develop a backend administration area.
- Manage administrator’s personal information.
- User-related operations
- Poll-related operation

### **Poll management**

The main purpose of the module is to handle opinion polls data more conveniently.

- Manage opinion poll information.
- Opinion poll tracking operations.
- Poll validating operations.

### **Reports**

This report module will generate all the reports that cover the data and operations related to the users and opinion polls to help future management.

- User-related report generation.
- Opinion poll-related report generation.
- Decision-related report generation.

## 1.5 Structure of the Dissertation

The dissertation mainly includes several chapters and those chapters mostly contain the stages of the system development life cycle.

➤ **Introduction**

By reading this chapter reader should be able to identify the scope and the objectives of the project and what is the need for this project. Also, explain the subject matter and the background of the proposed project.

➤ **Background**

Describe the functionality of the existing system by using appropriate UML diagrams. Provide non-functional and functional requirements of the system. Also compare and contrast similar systems.

➤ **Design**

Further, illustrate the system by providing enhanced diagrams to show the static view and the dynamic view of the system. Need to design the UI required to develop the project.

➤ **Implementation**

All-important code segments and modules should be explained in this chapter, Module interaction should be mentioned by using appropriate techniques.

➤ **Testing and Evaluation**

This chapter describes all types of testing such as unit testing, integrated testing, and system testing. etc. All the strategies and the evaluation process should be described here.

➤ **Conclusion**

Explain how you overcome the problems explained at the beginning of the project. Describe how you achieve the objectives and explain future works



# Chapter 2 - Background

## 2.1 Introduction

Considering the implementation of any kind of system for a requirement, the background study plays a huge role since it identifies the current progress in the field and the necessities to be improved. Considering the current progress in E-voting systems plenty of research that has been carried out can be identified. Various nations use electronic voting systems, including those that allow voters to cast votes online and in physical voting booths. However, the security of these current e-voting systems has been a major concern for the authorities and users of the system. Malicious behavior and programming errors in the systems were not detected and compensated to make the system more accurate and trustworthy (Software Systems Safety, 2022). The problems regarding the current e-voting systems from various countries have been raised. Due to these reasons, the Trustworthy nature of the e-voting system has unfortunately decreased allowing users to rely always on traditional voting methods.

Therefore, in the past few years, current e-voting systems have been designed to achieve a set of fundamentals while ensuring security requirements at the same time. For the analysis of basic security requirements of the systems, the usage of cryptographic models has played a major role. Even more abstract so-called Dolev-Yao models also were used at times for the implementation of the system (Software Systems Safety, 2022). The outcome of all these efforts was to ensure the security of the system creating a risk-free environment for users of the e-voting systems and the relevant authorities who are in charge of the voting system in the country.

## 2.2 Requirement Analysis

Analysis has to do a major role in the software development life cycle. Before converting a manually driven process into a computerized system it is very important to get a vivid idea about the problem domain and the user requirements.

### 2.2.1 Functional Requirements

➤ **User management module.**

- User registration – Users can register in the e-voting system.
- Change password – Users can change their login password.
- Change user profile – Change and update the profile details of the users.
- Approve and block users – Registered users will approve or block by admins after reviewing the user profile.

➤ **Administration management module.**

- Admin creation – Super admin can create admins for the e-voting system.
- Change password – Admin can change their login password.
- Change admin profile – Change and update the profile details of the admin.

➤ **Poll management module.**

- Create polls – Create an opinion poll according to the requirements. Users or admins can create polls.
- Approve polls – Created polls can be approved by admins after reviewing the poll details.
- Validate and track polls – Validate and track poll details by admins after creating a poll.
- Cast vote – Users can cast votes for enabled polls.
- Terminate polls – Can terminate ongoing polls when the criteria are met.
- View poll progress – Can view reports of the polls by admins and users.

➤ **Reports generating module.**

- Generate user-related reports.
- Generate opinion poll-related reports.
- Generate decision-related reports.

## 2.2.2 Non-Functional Requirements

- **Effectiveness** – The system should exactly affect the current mechanism of the Online portal for e-voting and opinion polls.
- **Performance** – The system should not take a long time to cast votes, create polls, voter registration, etc. Also, the website should load very smoothly and quickly, if not the users get annoyed.
- **Reliability** – The system's information must be accurate. Typically expressed in a percentage.
- **User Friendliness** – Users should have less trouble using the system and the interface of the system should not confuse any user.
- **Availability** – The website needs to be accessible whenever a user wants it.
- **Quality** – The system should maintain the quality of the services at all phases.

## 2.3 Review of Similar Systems

Security Analysis of the Estonian Internet Voting System can be considered to identify the necessities that should be fulfilled in such an e-voting system that provides users with a safer platform. Estonia was the first nation on the planet to utilize Internet voting broadly, and today more than 30% of its votes are cast on the web. The security of the Estonian E-voting framework is given a blend of individual race perception, code audit, and antagonistic testing. Embracing a risk model that considers the progressed dangers confronted by a national decision framework including deceptive insiders and state-supported assaults, it is found that the E-voting framework has genuine compositional impediments and procedural crevices that possibly imperil the trustworthiness of decisions. Exploratory assaults on a generation of the framework exhibit how such aggressors could target the decision servers or voters' customers to change race results and then again undermine the authenticity of the framework. Discoveries represent the down-to-earth impediments to Internet voting in the present-day world, and they convey lessons for Estonia, for other nations considering receiving such frameworks, and for security research (Research Paper | Independent Report on E-voting in Estonia, 2022).

Australia is at present utilizing an open-source voting system. On the other hand, security models, for example, the voter-confirmed review trail take into account electronic voting frameworks that deliver a paper trail that can be seen and confirmed by a voter. In such a framework, the rightness trouble on the voting terminal's code is essentially less as should be obvious and check a physical item that portrays their vote. Regardless of the possibility that, for reasons unknown, the machines can't name the victor of a decision, then the paper tickets can be related, either mechanically or physically, to pick up logically more exact decision results. Voter-certain review trials are required in some U.S. states, and major DRE (Direct-Recording Electronic voting machines) merchants have made open proclamations that they would bolster such elements if their clients required it. The EVM (Electronic Voting Machines) project<sup>11</sup> is an aspiring endeavor to make an open-source voting framework with a voter-evident review trail a commendable objective. The model where singular sellers compose exclusive code to run our decisions seems, by all accounts, to be questionable; what's more, on the off chance that doesn't change the procedure of outlining the voting frameworks, People will have no certainty that the race results will mirror the will of the

electorate. It is deserved for the future to have powerful, all-around outlined decision frameworks to protect the bedrock of the vote-based system (Ethical Issues in E-Voting, 2022).

Because of the configuration standards and necessity, a model of the framework for the E-voting System has been produced utilizing PHP. The framework has a few points of interest that had been accomplished. The points of interest of the framework are as per the following. It gives trust in the voting framework, just the real voter is permitted to access voting. The framework is easy to understand, in the sense that the client can without much of a stretch comprehend the framework even though the client is a first-time client. This is because the outline is straightforward, and alluring, and doesn't have an excess of graphical things (Ethical Issues in E-Voting, 2022).

In the 2006 U.S. decision, it was evaluated that more than 66 million individuals would be voting on direct-recording electronic (DRE) frameworks in 34% of the country's provinces. Although these PC-based voting frameworks have been generally embraced, they have not been observationally turned out to be more useful than their antecedents. The arrangement of studies reported here thinks about the ease of use of information from a DRE with those from more customary voting innovations (paper polls, punch cards, and lever machines). Results show that there were few contrasts between the DRE furthermore, these more established techniques in proficiency or viability. Be that as it may, as far as client fulfillment, the DRE was essentially superior to the more established strategies. Paper tallies additionally perform well, however, members were significantly more fulfilled by their encounters voting on the DRE. The detachment between subjective and objective usability has potential policy ramifications (Analysis of an Electronic Voting System, 2022).

The objective of these studies was to address the ease of use of DREs contrasted with more established, more customary voting systems and also, to start to investigate the space of what is conceivable with DREs. It has been expected by strategy producers and others that DREs would be superior to anything the paper votes, lever machines, and punch cards have been supplanting. Our outcomes demonstrate that execution on DREs as far as productivity and viability are not superior to additional conventional systems, and because of the high rate of post-completion blunders, it might be outstandingly more terrible. This discovery recreated overall analyses and unequivocally demonstrates that DREs don't as a matter of course prompt better voting execution as had been accepted. This was a powerful impact what's more, even though there were a few

impacts of age and training on ticket fulfillment times, all in all, just PC aptitude influenced voters' levels of proficiency on the DRE. No distinctions in blunder rates were seen between the "standard" DRE and the more established voting techniques, yet the high recurrence of tickets containing mistakes is the reason for concern. The high per-vote mistake rates seen here are predictable with past work on more seasoned voting systems. Since the results of decisions have such imperative and far-reaching effects, this is risky. Even though there were no distinctions in vote finish times or mistake rates between the DRE and the other three systems, members were most fulfilled by their voting experience in the electronic voting framework (Analysis of an Electronic Voting System, 2022).

As the past examination has reported, antiquated paper polls work great. Most voters can perform at sensible productivity also, viability levels with paper tickets and are fulfilled with the experience. Then again, paper polls are seen as blocked off for some gatherings of individuals. For instance, paper tallies don't give voters inabilities, for example, low vision or manual expertise debilitations with the capacity to vote in an autonomous mystery poll. Be that as it may, there are, indeed, approaches that can make paper tickets more open (e.g. Braille tickets, advances, for example, and there are reasons to question that present business DREs do an especially great job with openness concerns. Generally speaking, the ease of use discoveries from the present studies appear that the utilization of DREs does not prompt more productive or successful voting execution, even though voters are extremely fulfilled by these electronic frameworks. The high fulfillment members feel with the DREs implies that natives may be troubled about relinquishing the new voting frameworks despite security concerns. Even though their execution of the DREs is not any better, voters might fight to keep the systems that they so strongly prefer (Analysis of an Electronic Voting System, 2022)

Moreover, doubters of Internet voting may likewise contend that the client has a decision of whether to utilize mechanical or conventional means with money-related exchanges. The outcomes of disappointment or alteration are essentially a matter for the client and money-related establishment to determine and may have almost no impact on outsiders. In decisions, be that as it may, a decision is being made about who will practice government control over the whole group. The result of framework disappointment could influence everybody, including people who wished to enlist their vote just by conventional systems (NEW APPLIED E-VOTING SYSTEM, 2022).

With money-related administrations, the client likewise has a decision about which monetary foundation to utilize, a customer can stay away from those whose record of mechanical unwavering quality or security appears to be questionable, or whose framework appears to be unduly tedious or hard to utilize. The voter who is troubled with the innovative choices offered by Canada's appointive framework does not be able to change to another supplier of voting administrations.

The point can likewise be made that budgetary exchanges happen all the time, and the individual client is liable to take part in various exchanges in a given year. Subsequently, both the establishment and the client can distinguish issues, gain from them, and obtain impressive offices by utilizing innovations. By complexity, government general races are a rare occasion in Canada one each three to five years is common and there may be far less chance to incrementally watch both the target components of a framework and the office of customers and organizations in working on them. With a constrained window of utilization for arrangement and true testing, there may be space for vulnerabilities in the server or potential for intrigue or control by insiders who could endeavor to change votes by concealing system modifying code inside of a server (NEW APPLIED E-VOTING SYSTEM, 2022)

.

.

## 2.4 Related Technologies

The prototyping model is a process for developing systems that involve creating a prototype, testing it, and then making any necessary adjustments until a working prototype is obtained from which the whole system or product may then be created. In situations where not all of the project requirements are known in detail earlier, this model performed better. Between the engineers and the users, it is an iterative, testing process.

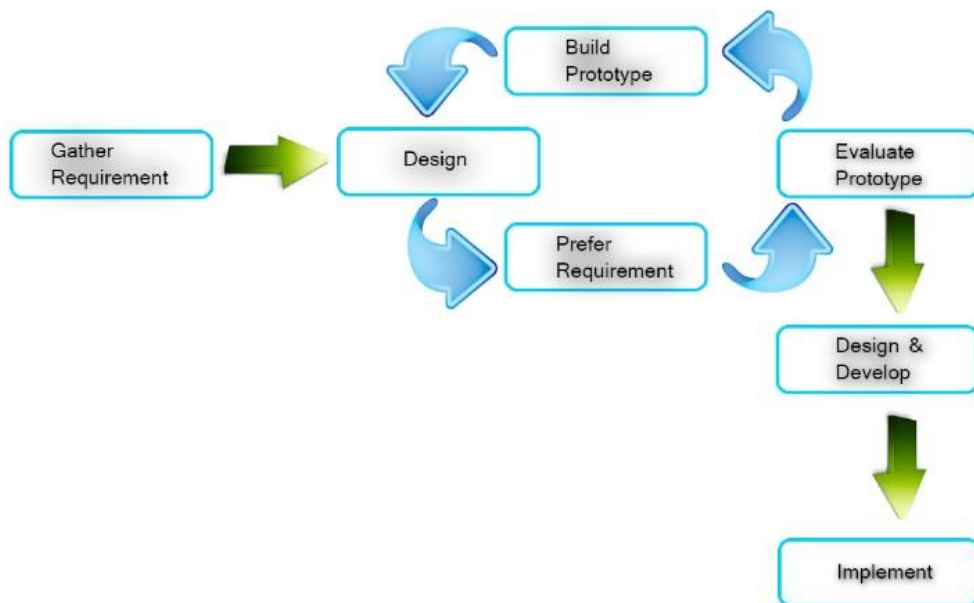


Figure 1 - Systems Development Method

The system should be web-based due to a lot of users interacting with the system and MVC architecture will be used.



# Chapter 3 - Design

## 3.1 Introduction

Software design is the procedure by which an individual develops a specification of a software product that is designed to achieve goals, using a collection of simple components and bound to limitations. Software design can be either "all the activities involved in conceptualizing, framing, implementing, commissioning, and ultimately modifying complex systems" or "the activity before programming and after requirements specification". An appropriate process model is selected to develop the system. A set of UML diagrams were used during the process.

The prototyping model is a system development approach in which a model is made, tested, and then revised as necessary until, at long last, a worthy model is built from which the entire system or object may now be developed. This methodology functions well in circumstances where not all of the project requirements are understood in detail at the beginning. Between the designers and the clients, there is an iterative, experimental process. Defining the fundamental requirements, developing a working prototype, testing the working prototype, and altering or extending the requirements are the fundamental processes in this methodology.

This can be considered a good approach to the system. First, the basic requirements are identified after a proper requirement analysis then the working prototype is created fulfilling those requirements. Next verification of the working prototype is performed changing and adapting for the requirements of the customers. The advantages of this system also play a big role in adopting this kind of system. System requirements early are not known in advance but still, we can effectively get the first step to building a requirement-fulfilling system. Creating fundamentally new software can be done as well as the developers lack confidence in software architecture and algorithms.

## **3.2 Related design strategies**

A web-based system is an information system which is used web technologies to deliver information and services to users. The proposed system was decided to be developed as a web-based solution. Following are the business advantages of web-based solutions.

- Access the system from anywhere and anytime.
- Customization is a very simple thing.
- Maintenance and installation are easy.
- More useful for users.
- Development is cost-effective.

### **3.2.1 Object-Oriented Analysis and Designing**

The object-oriented concept and visual modeling are applied across the life cycles of the development of a system, business, or application using object-oriented analysis and design, a popular technical method that promotes better product quality and stakeholder communication.

The popular software development process, Unified Process says that object-oriented analysis and design are best to be carried out iteratively and incrementally in present-day software engineering. The outcomes of the OOAD activities, the analysis models for OOA, and the design models for OOD, respectively, will be improved and evolve continuously, driven by important considerations like risks and commercial value. OOAD gives a perfect way for the development of the system. It enables the developer to design and create objects in real business. It also can be represented with a computerized system easily.

### **3.2.2 MVC architecture**

One of the most popular software development patterns is the Model-View-Controller (MVC) architecture. Separation of concerns is a design principle used in the logic of MVC architecture. The goal of this idea is to divide an application into distinct components, each of which deals with a distinct subject. CodeIgniter is a popular web development framework that supports Model-View-Controller architecture. Therefore, this framework is used to develop this system fast and easy way.

### 3.3 System Architecture

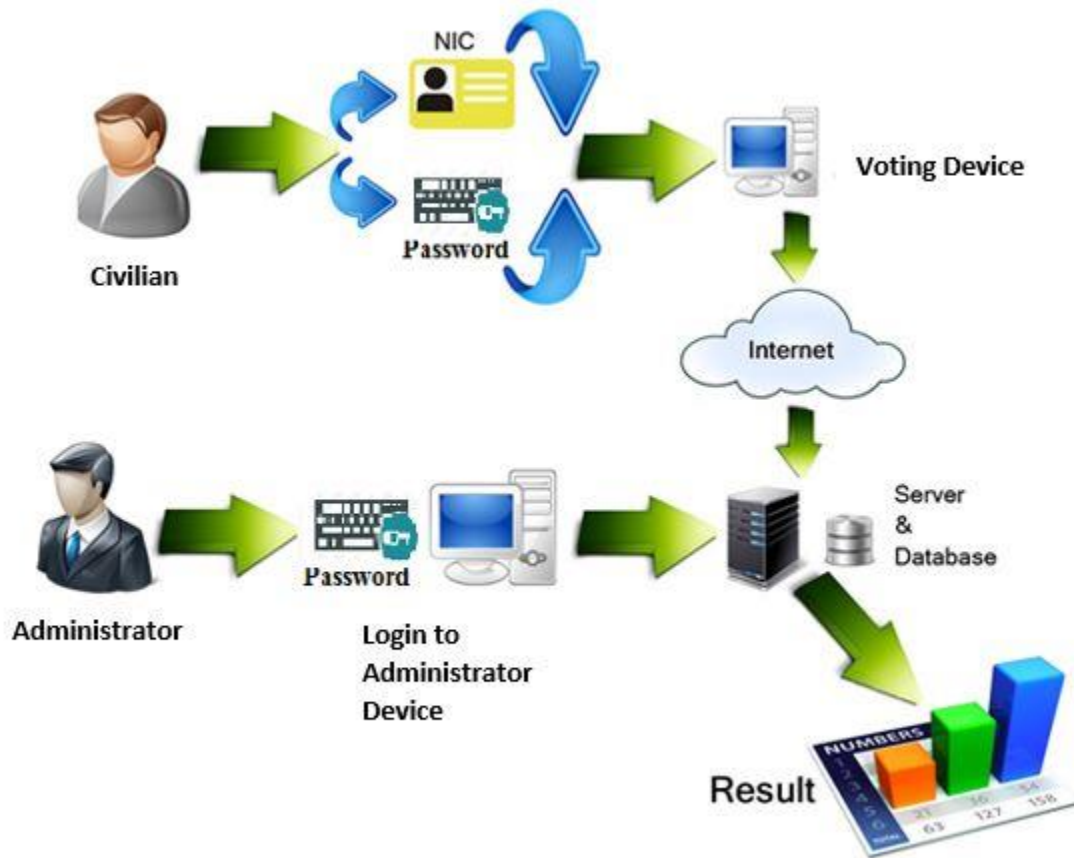


Figure 2 - System Architecture Diagram

The above diagram indicates the high-level system architecture diagram of the system. Software architecture is a collection of rules that define how software is developed and put together. The architecture specifies how the software system will be structured and organized. The levels of complexity, relationships between the various system components, and other aspects are also covered. The goals of a project can be described using architecture, and it can also be used to direct the planning and development of a new system.

## **3.4 UML diagrams**

The Unified Modeling Language (UML) is a standard visual demonstrating dialect proposed to be utilized for

- Demonstrating business and comparative procedures,
- Examination, outline, and usage of programming-based systems

UML is a typical language for business experts, programming developers and engineers used to depict, indicate, plan, and record existing or new business procedures, structure, and conduct of antiquities of programming frameworks.

UML can be connected to assorted application areas (e.g., keeping money, account, web, aviation, medicinal services, and so on.) It can be utilized with all significant article and segment programming advancement systems and for different usage stages (e.g., J2EE. NET).

UML is a standard displaying dialect, not a product improvement process. UML 1.4.2 Specification clarified that procedure: (UML 1.4 versus UML 2.0 as languages to describe software ... - research gate)

### **3.4.1 Use Case Diagram**

Use case diagram identifies the system requirements of a system by analyzing the user requirement. Use case diagrams to represent the system actors and how they are going to interact with the system and what is their core role in the system. Also, it represents how the external systems get involved with the proposed system and which way it is going to interact with the proposed system. These external systems can be either person, system components 0900, or the whole system.

A use case is a solitary unit of important work. It gives an abnormal state perspective of conduct discernible to somebody or something outside the system.

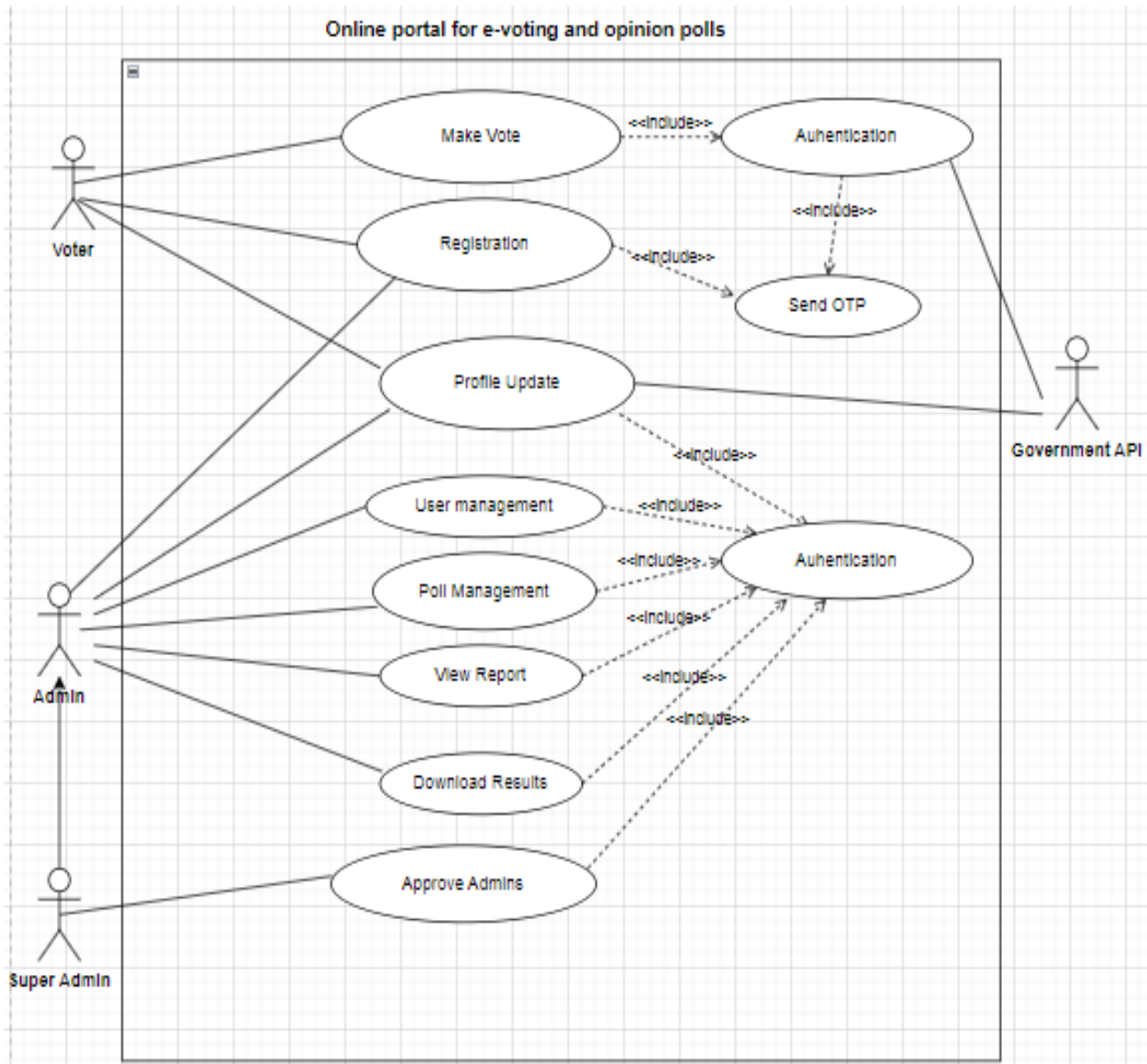


Figure 3 - Use Case Diagram

## Use case scenarios

Class of use case	Use case	Description
Authentication use case.	<ol style="list-style-type: none"> <li>1. Login as admin.</li> <li>2. Login as the user.</li> </ol>	<ol style="list-style-type: none"> <li>1. Admin logs into the system</li> <li>2. The User logs into the system.</li> </ol>
Registration use case.	<ol style="list-style-type: none"> <li>1. Register as admin.</li> <li>2. Register as a user.</li> </ol>	<ol style="list-style-type: none"> <li>1. Admin registers as a user and then supper admin will make that user an admin.</li> <li>2. Users will register via mobile number</li> </ol>
Profile update use case.	<ol style="list-style-type: none"> <li>1. Update user data</li> <li>2. Update user additional data</li> </ol>	<ol style="list-style-type: none"> <li>1. Update basic profile details</li> <li>2. Update additional profile details after the registration.</li> </ol>
Make vote use case	Select a relevant poll and vote.	Select poll and then select a poll question and finally vote for the answer.
User management	<ol style="list-style-type: none"> <li>1. View user details.</li> <li>2. Disable users</li> <li>3. Set as admins.</li> </ol>	<ol style="list-style-type: none"> <li>1. Admins can view user details.</li> <li>2. Admins can disable users and block voting for them.</li> <li>3. Admin can give authorization for users to act as an admin.</li> </ol>
Poll management	<ol style="list-style-type: none"> <li>1. Create poll</li> <li>2. Invite voters</li> <li>3. End poll</li> <li>4. Edit poll</li> <li>5. View results</li> </ol>	<ol style="list-style-type: none"> <li>1. Admin creates a poll by adding all the data.</li> <li>2. Invite voters to vote for the specific poll.</li> <li>3. End the poll on the poll end date.</li> <li>4. Edit poll according to requirements.</li> <li>5. View results and make decisions</li> </ol>
View reports	<ol style="list-style-type: none"> <li>1. User reports</li> <li>2. Poll report</li> </ol>	<ol style="list-style-type: none"> <li>1. View user reports and filter according to the requirements</li> <li>2. View poll reports for making decisions.</li> </ol>

Table 1 - Use case scenarios

### 3.4.2 Activity Diagram

In UML, an activity diagram is utilized to show the arrangement of activities in the system. Activity outlines demonstrate the work process from the beginning point to the completion point specifying the numerous choice ways that exist in the movement of occasions contained in the action. They may be utilized to detail circumstances where parallel preparing may happen in the execution of a few activities. Activities outlines are helpful for businesses demonstrating where they are utilized for specifying the procedures included in business activity.

- Make Vote (Doing by the voter)

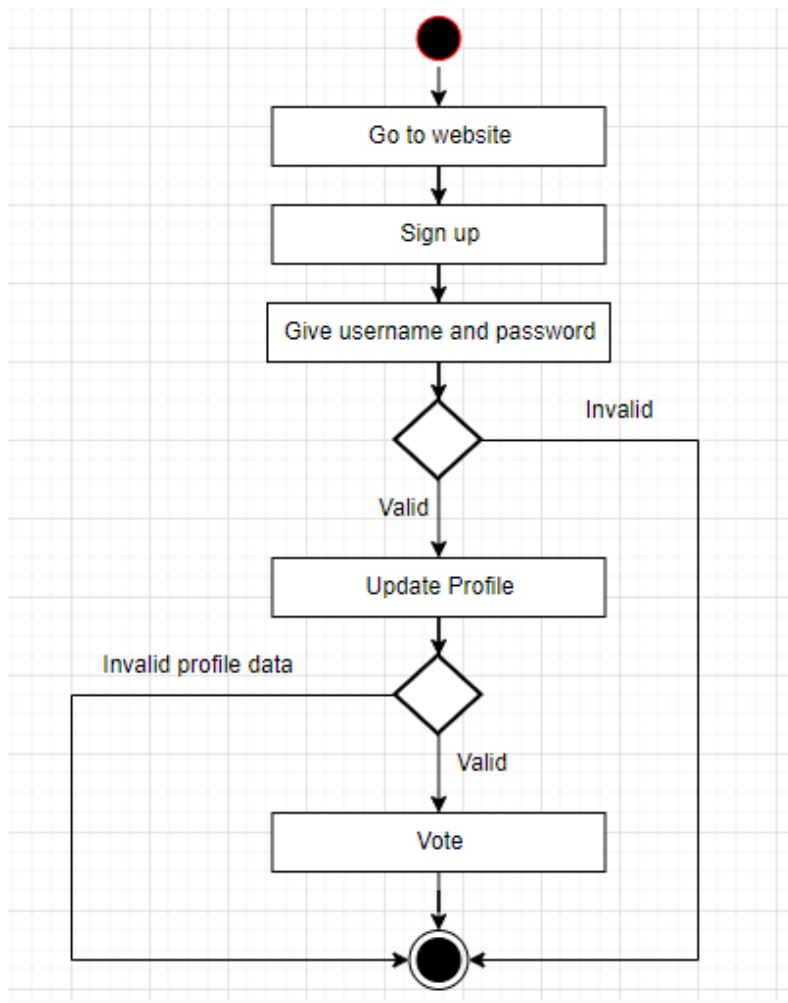


Figure 4 - Activity Diagram 1

- Create a poll (Done by the admin)

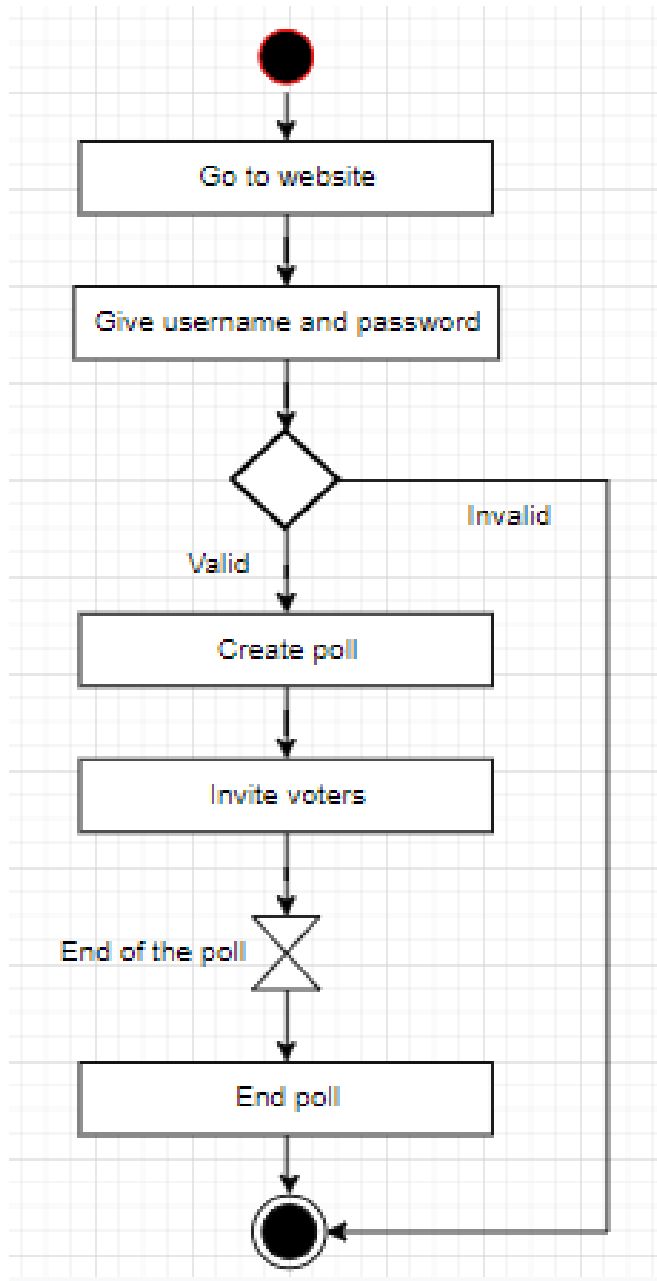


Figure 5 - Activity diagram 2



### 3.4.3 Sequence diagram

This is also called an interaction diagram. Sequence diagrams show how processes are interacting with each other and in what order. It represents the interaction between objects in a time sequence and message exchanges between those objects to finish functionalities of scenarios

- Make a vote sequence diagram

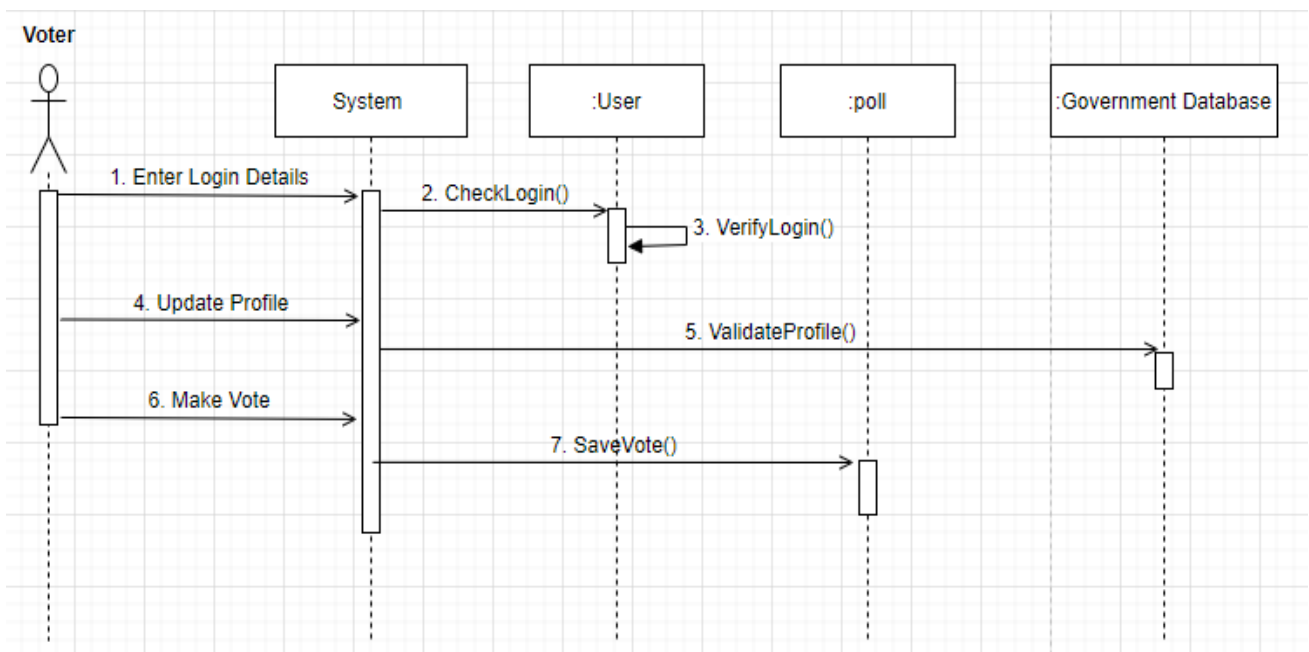


Figure 6 - Sequence Diagram 1

➤ Create a poll sequence diagram

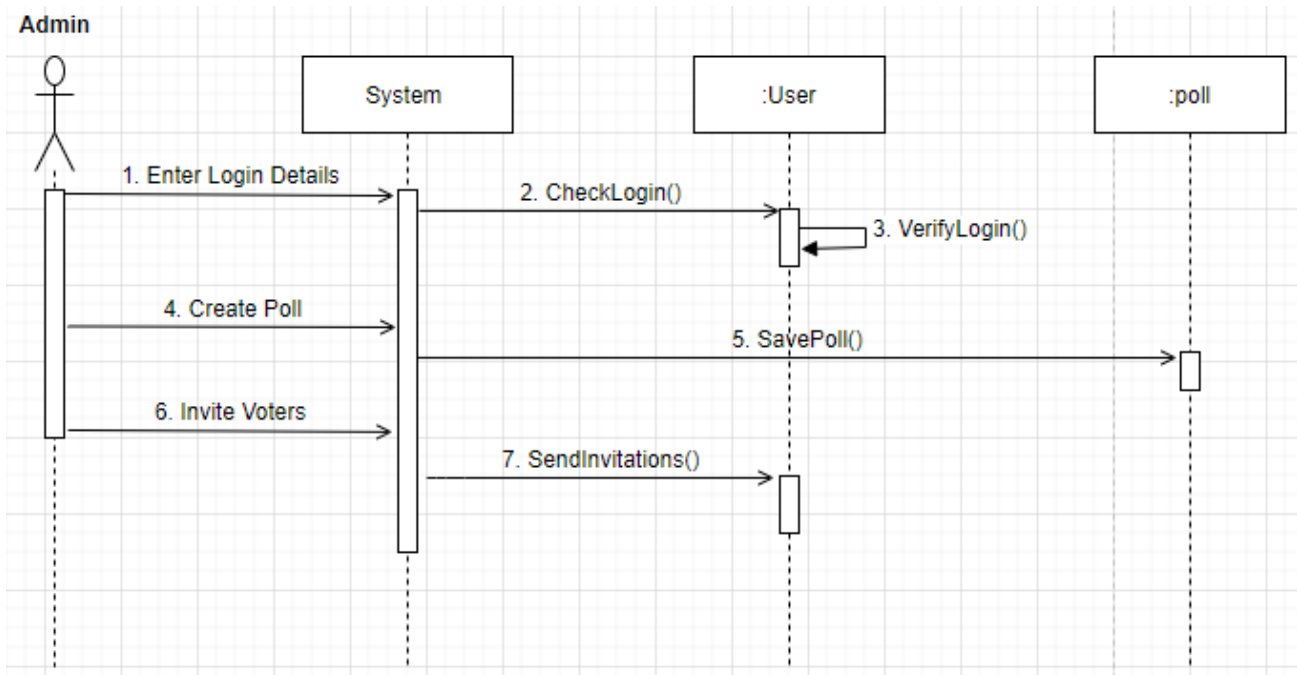


Figure 7 - Sequence Diagram 2

### 3.5 Database Designing

Database design is the process of producing a detailed data model of a database. Important data can be kept in a proper order without losing them by a better database design. To fulfill the objectives of the system identified functional and non-functions requirements of the system. Then draw the entity relationship diagram to identify the entities required to build the system and then created the database according to the ER diagram.

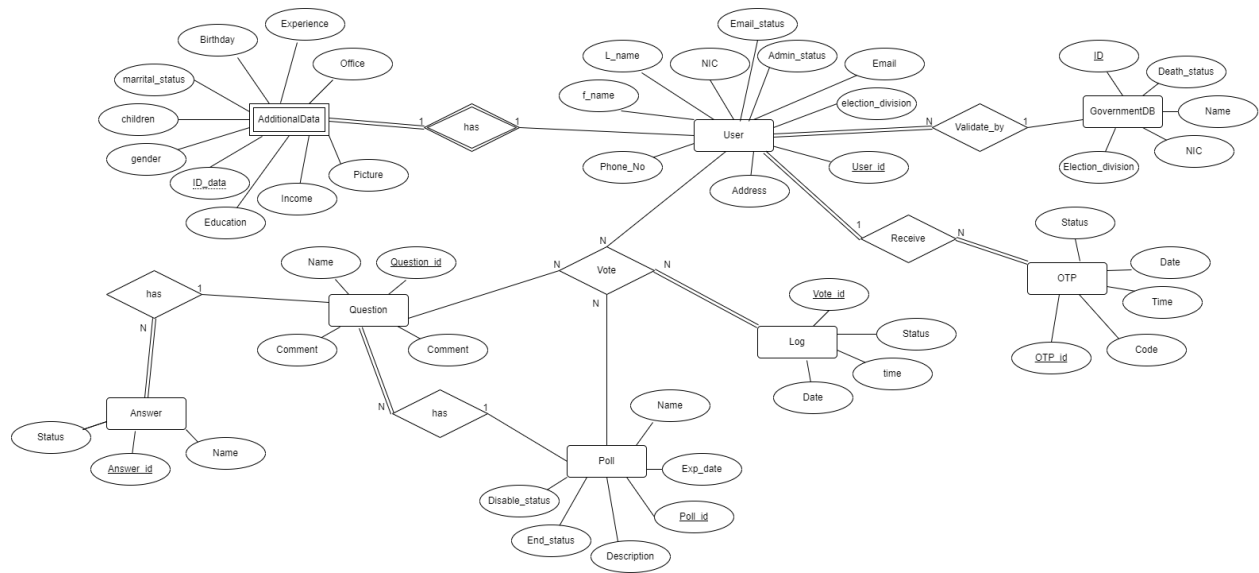


Figure 8 - ER Diagram

### 3.6 User Interface Designing

User interface design plays a major part in designing and it is important because users are directly interacting with user interfaces. I used the paper-based user interface design technique because it can be used to identify the initial idea of the UI early and easy to understand the navigations. Also, it can be modified quickly during the implementation process. The below figures show the main user interfaces designed by using pape prototypes

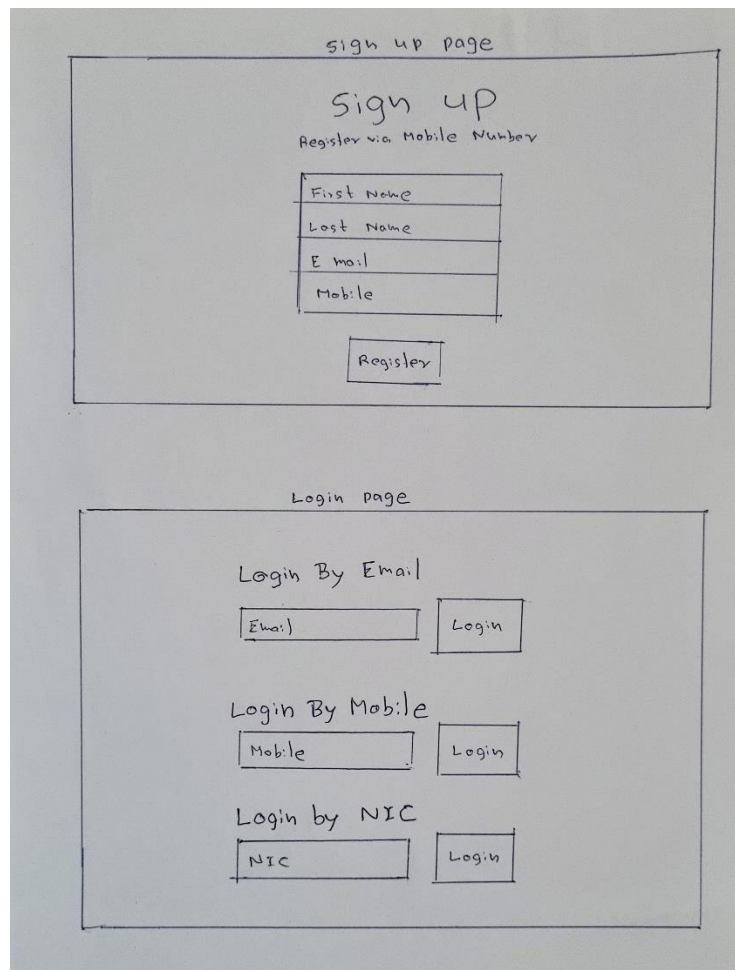


Figure 9 - Sign up and Login Prototype

update user

Photo Name

First Name

Last Name

NIC

E-mail

Mobile Number

Address

City

update

user profile photo

select image

choose file

upload image

Additional data

Figure 10 - Update User Prototype

Poll Creation

Poll Name

Name

End Date

mm/dd/yyyy

Description

Description

Create

Add a question

Create question

question

create

Figure 11 - Poll Creation Prototype

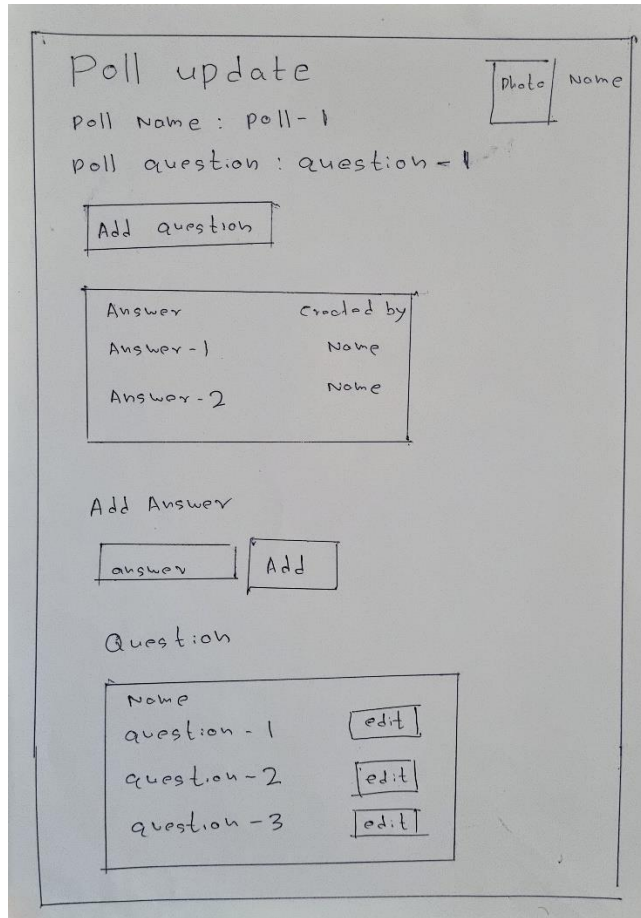


Figure 12 - Add Answers Prototype

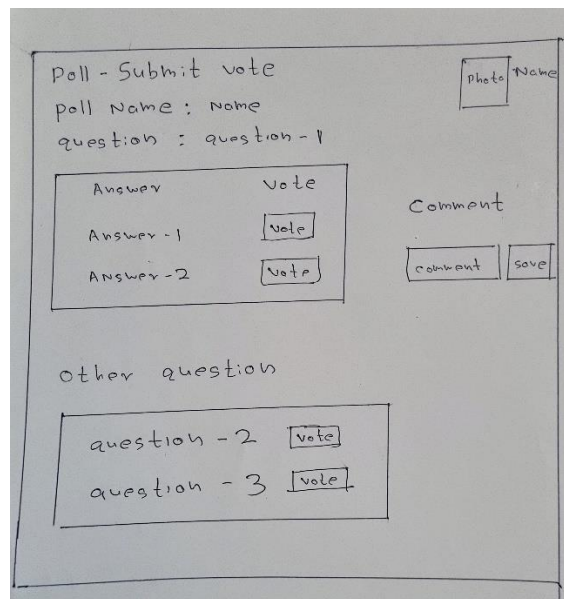


Figure 13 - Voting Page Prototype

# Chapter 4 - Implementation

## 4.1 Introduction

After the successful completion of the design phase, the implementation phase is carried out according to what the design stage planned by using appropriate tools and techniques. This is a very important stage of a software development life cycle. The output of this stage is an effective executable system. Initially, the system has been divided into modules that are functionally separated and finally integrated into all the developed modules as a whole system. The major code segments, tools, and techniques that are used to implement the system in the implementation environment are mainly focused on in this chapter.

## 4.2 Implementation Environment

The implementation environment has two major categories. The minimum hardware and the software environment are mentioned here. The following table tells what a computer is required to have in the hardware aspect as well as the software aspect.

Hardware Environment	Software Environment
Intel(R) Core(TM) i5-5200U CPU @ 2.20GHz	Microsoft Windows 8.1
4GB RAM	PHP Version 5.4.12
512GB HDD	Apache 2.4.4
	MySQL 5.6.12

Table 2 - Implementation Requirements

## 4.3 System Development Tools and Technologies

The following tools and techniques are used while developing the system. A brief description is also given in addition.

➤ **Microsoft Visio – 2013**

This software is used to draw the diagrams for the system. It supports almost all UML diagrams and is easy to draw the complex UML diagrams.

➤ **Codeigniter 3**

This software is used to code the system. CodeIgniter is a popular web development framework that supports Model-View-Controller architecture. This framework is selected because it provides better security and also support good search engine optimization.

➤ **Adobe Photoshop CS6**

Photo editing is done by using this software.

➤ **PHP**

This is a server-side scripting language that is used when developing the back end of the system. This is selected because it provides fast loading capability, flexible for use and documents are available for learning.

➤ **MySQL**

This is the Database software that is used to handle the database of the system. When we conder security and data protection this is the best database management system. This poll management system requires security on data and hence MySQL is the best.

➤ **HTML**

This is the basic web language to give the structure of the webpage.



➤ **CSS / Bootstrap**

This is used for the styling of the system. Most of the users use their mobiles to connect with this poll management system. The responsive feature of bootstrap is the best for these mobile users to use this website without any inconvenience.

➤ **JavaScript**

This is a client-side scripting language that is used when client-side validation and to dynamic the system components.

➤ **jQuery**

This is a JavaScript library to create JavaScript functions with less code and dynamic content.

➤ **JSON**

It is used for storing and exchanging text-related information. It is faster and easier to parse.

➤ **Visual studio code**

Basic text editor. It has features such as automated code suggestions and tab indents. This code editor is selected because it is instantly productive with bracket, syntax highlighting, and auto-indentation.

➤ **Apache**

Used as a web server software.

➤ **Filezilla**

Cross-platform FTP application used to connect with the server.

➤ **DataTables**

DataTables is a plug-in for the jQuery JavaScript library. It is a very adaptable tool

## **4.4 Major Code Segments**

The system's code is organized into reusable modules. Below is a list of the major code modules along with a brief explanation of what each one does. For the purpose of identifying a specific code's functionality, comments are included.

### **4.4.1 Database Connection**

All systems whether they are web-based or standalone, if the system is developed to serve the business, and the database must be integrated into it. Add, delete and update are the most important and basic actions in the system. If the above-mentioned actions are used in the system, the system must be connected to the database. The following code segments are used to connect the system with the related database. PHP's MySQLi (MySQL Improved) is the main API option when connecting to a MySQL database server. A predefined MySQLi function called "mysqli( )" connects when four parameters are passed namely hostname, username, password, and the database name.

```
database.php X
H: > > 2nd year project lap > files > final > httpdocs > application > config > database.php
67 | The $active_group variable lets you choose which connection
68 | make active. By default there is only one group (the 'default')
69 |
70 | The $query_builder variable lets you determine whether or not
71 | the query builder class.
72 */
73 $active_group = 'default';
74 $query_builder = TRUE;
75
76 $db['default'] = array(
77     'dsn' => '',
78     'hostname' => 'localhost',
79     'username' => 'evotinguser',
80     'password' => 'evoting321',
81     'database' => 'evoting_db',
82     'dbdriver' => 'mysqli',
83     'dbprefix' => '',
84     'pconnect' => FALSE,
85     'db_debug' => TRUE,
86     // 'db_debug' => (ENVIRONMENT !== 'production'),
87     'cache_on' => FALSE,
88     'cachedir' => '',
89     'char_set' => 'utf8',
90     'dbcollat' => 'utf8_general_ci',
91     'swap_pre' => '',
92     'encrypt' => FALSE,
93     'compress' => FALSE,
94     'stricton' => FALSE,
95     'failover' => array(),
96     'save_queries' => TRUE
97 );
98
```

Figure 14 - Database Connection Code

## 4.4.2 Login Handling Page

The code segment given below links the database connection page and also catches the user inputs for further processing. It also checks whether the user inputs such as username and password are empty or not. If the username (mobile, email, or NIC) is given, then check the one-time password, if yes, continue the next process, otherwise redirect to the login page with an error.

➤ OTP sending code segment

```
Login.php
64 private function sendOpt($mobile) {
65
66     $res = $this->UserModel->valid_mobile($mobile);
67
68     if($res['login_status']){
69         $data['status'] = 1;
70         $data['user_id'] = $res["login_data"][0]->idusers;
71         $code = $this->opt_code_generate($res["login_data"][0]->idusers);
72         $this->otpSend($mobile,$code);
73         return $data;
74     }else{
75         $data['status'] = 0;
76         $data['error'] = 'Not a registered Mobile..!';
77         return $data;
78     }
79
80 }
81 private function sendOptEmail($email) {
82
83     $res = $this->UserModel->valid_mobileByEmail($email);
84
85     if($res['login_status']){
86         $data['status'] = 1;
87         $data['user_id'] = $res["login_data"][0]->idusers;
88         $code = $this->opt_code_generate($res["login_data"][0]->idusers);
89         $this->otpSend($res["login_data"][0]->mobilenumber,$code);
90         return $data;
91     }else{
92         $data['status'] = 0;
93         $data['error'] = 'Not a registered EMAIL..!';
94         return $data;
95     }
96
97 }
```

Figure 15 - OTP Sending Code 1

```
Login.php
8     private function sendOptNic($nic) {
9
10        $res = $this->UserModel->valid_mobileByNic($nic);
11
12        if($res['login_status']){
13            $data['status'] = 1;
14            $data['user_id'] = $res["login_data"][0]->idusers;
15            $code = $this->opt_code_generate($res["login_data"][0]->idusers);
16            $this->otpSend($res["login_data"][0]->mobilenumber,$code);
17            return $data;
18        }else{
19            $data['status'] = 0;
20            $data['error'] = 'Not a registered NIC..!';
21            return $data;
22        }
23    }
24 }
```

Figure 16 - OTP Sending Code 2

```
Login.php
286
287 // Send otp via request - !!private!!!
288 private function otpSend($number,$otp){
289
290     $curl = curl_init('https://pos.janitha.biz/pb/sms_send_otp/lkjs025s7kljdjd6wsd6fgdkjdfx798sdfuhsd257dfh/'.$number.'/'.$otp);
291     curl_setopt($curl, CURLOPT_RETURNTRANSFER, true);
292
293
294     // Send the request
295     $result = curl_exec($curl);
296
297     // Free up the resources $curl is using
298     curl_close($curl);
299 //     echo $result;
300     return $result;
301 }
302
303 }
304 }
```

Figure 17 - OTP Sending Code 3

- OTP generating code segment

```
Login.php
5 private function opt_code_generate($id){
6     $rand_code =mt_rand(1000,9999);
7     $data = array(
8         'code' => $rand_code,
9         'user_id' => $id
10    );
11    $this->UserModel->save_opt($data);
12
13    return $rand_code;
14 }
15
```

Figure 18 - OTP Generate Code

- OTP verification code segments

```
Login.php
6 public function verify_otp($user_id){
7     if (!empty($this->session->auth)) {
8         return redirect('home', 'refresh');
9     }
10
11     if (!$this->input->post('otp')){
12         return redirect('login', 'refresh');
13     }
14     $data['otp'] = $this->input->post('otp');
15
16     if (is_numeric($data['otp'])){
17         $result = $this->UserModel->login_code($user_id,$data['otp']);
18
19         if ($result == "1"){
20             $this->log_in($user_id);
21             return redirect('home', 'refresh');
22         }else{
23             $data['user_id'] = $user_id;
24             $data['error'] = 'OPT Not correct';
25
26             $this->load->view('layouts/front_h');
27             $this->load->view('login/code_index',$data);
28             $this->load->view('layouts/front_f');
29         }
30     }
31 }
32
33 }
```

Figure 19 - OTP Verification Code

➤ Registration code segment

```
Login.php
0 public function register(){
1     if (!empty($this->session->auth)) {
2         return redirect('home', 'refresh');
3     }
4     if (!$this->input->post('mobile')){
5         return redirect('login', 'refresh');
6     }
7     $mobile = $this->input->post('mobile');
8     $res = $this->UserModel->valid_mobile($mobile);
9
10    if(!$res['login_status']){
11        $firstname = $this->input->post("firstname");
12        $lastname = $this->input->post("lastname");
13        $email = $this->input->post("email");
14        $password = hash("sha256", rand());
15
16        $this->UserModel->saveData($firstname, $lastname, $email, $password,$mobile);
17        $this->index();
18        $data['user_name'] = $mobile;
19        $data['error_r'] = 'Register Success - Please login';
20        $this->session->set_flashdata('login_data',$data);
21        return redirect('login', 'refresh');
22    }else{
23        $data['status'] = 0;
24        $data['error_r'] = 'Already registered Mobile..!';
25        $this->load->view('layouts/front_h');
26        $this->load->view('login/index',$data);
27        $this->load->view('layouts/front_f');
28    }
29 }
30 }
```

Figure 20 - User Registration Code

### 4.4.3 Email validation

The code segment given below shows the email validation of the user.

### 4.4.4 Vote page validation

The code segment given below shows user validation done before allowing the voting process. Location validation, NIC validation, email validation, and death status validation are the main validations checked before the voting process.

```
Election.php
0 public function user_death_status(){
1     $user_data = $this->session->userdata('data');
2     $curl = curl_init('https://evoting.techyfish.com/api/get_death_status_by_nic/12ddk3ok3kk1fdd1kfkjewqpamdfnjvnrke4neem2n/'.$user_data->
3
4     curl_setopt($curl, CURLOPT_RETURNTRANSFER, true);
5     // Send the request
6     $result = curl_exec($curl);
7
8     // Free up the resources $curl is using
9     curl_close($curl);
0
1     $res = json_decode($result,true);
2     if ($res['valid'] == 0){
3         return -1;
4     }else{
5         if ($res['valid'] == 1){
6             return $res['death'];
7         }
8     }
9 }
```

Figure 21 – Death Status Validation Code



```

Election.php
33
34 public function vote($id,$key,$poll_id = 0){
35     if (!$this->session->has_userdata('auth')) {
36         return redirect('login', 'refresh');
37     }
38
39     $useridsession = $this->session->id;
40
41     $loginUser = $this->UserModel->getUserData($useridsession);
42     $addData = json_decode(json_encode($this->UserModel->getUserAdditionalDataByUserId($useridsession)), true);
43     $today = date("Y-m-d");
44     $diff = date_diff(date_create($addData[0]['birthday']), date_create($today));
45     $age = $diff->format('%y');
46
47     $userData = array(
48         'data' => $loginUser[0],
49         'auth' => true,
50         'id' => $useridsession,
51         'nic' => $addData[0]['nic'],
52         'birthday' => $addData[0]['birthday'],
53         'age' => $age,
54         'name' => $loginUser[0]->firstname,
55         'super' => $loginUser[0]->superuser,
56         'admin' => $loginUser[0]->admin,
57     );
58
59     $this->session->set_userdata($userData);
60
61     //Update End Date
62     $this->PollModel->updateElectionEndDate();
63
64     $data['election'] = json_decode(json_encode($this->PollModel->getElectionByIdKeyUser($id, $key,0)), true);
65
66

```

Figure 22 - Vote Page Validation Code 1

```

Election.php
7
8     $death_status = $this->user_death_status();
9     if ($death_status == -1){
10         $data['error'] = "Please update your NIC. Your NIC not found in government database !";
11         return $this->load->view('error_page', $data);
12     }else{
13         if ($death_status == 1){
14             $data['error'] = "Deceased Person cannot vote .!";
15             return $this->load->view('error_page', $data);
16         }
17     }
18
19     if ($data['election'][0]['end_status'] == 1){
20         $data['error'] = "Election Has Ended .!";
21         return $this->load->view('error_page', $data);
22     }
23
24     $data['e_id'] = $id;
25     $data['e_key'] = $key;
26     $data['poll_id'] = $poll_id;
27     $data['poll_data'] = [];
28     $data['poll_empty'] = false;
29     $data['poll_answers'] = [];
30     $data['poll_key'] = 0;
31     $data['polls'] = [];
32     $data['polls'] = json_decode(json_encode($this->PollModel->getPollsByElectionId($id)), true);
33     $data['answer_id'] = 0;
34     $user_id = $this->session->id;
35     $data['age_18'] = 0;
36

```

Figure 23 - Vote Page Validation Code 2

```

Election.php
6
7     if ($this->session->age > 18){
8         $data['age_18'] = 1;
9     }else{
10        $data['error'] = "Age Restriction: Cannot vote below 18 years old";
11        return $this->load->view('error_page', $data);
12    }
13    if (empty($data['polls'])){
14        $data['poll_empty'] = true;
15    }else{
16        if ($poll_id == 0){
17            $data['poll_id'] = $data['polls'][0]['id'];
18        }else{
19            $data['poll_id'] = $poll_id;
20        }
21        $data['poll_data'] = json_decode(json_encode($this->PollModel->getPollById($data['poll_id']), true);
22        $data['poll_answers'] = json_decode(json_encode($this->PollModel->getPollAnswersByPollId($data['poll_id']), true);
23        $data['poll_key'] = $data['poll_data'][0]['verify_key'];
24
25        $check_vote_log = json_decode(json_encode($this->PollModel->getVoteLogByPollIdUserId($data['poll_id'],$user_id), true);
26
27        if (!empty($check_vote_log)){
28            $data['answer_id'] = $check_vote_log[0]['answer_id'];
29        }
30
31    }
32
33    $data["userid"] = $this->session->id;
34
35    $this->load->view('layouts/header');
36    $this->load->view('election/vote', $data);
37    $this->load->view('layouts/footer');
38
39
40 }

```

Figure 24 - Vote Page Validation Code 3

## 4.4.5 Invitation sending process

The code segment given below shows the vote invitation sending process to the user's Email address when the poll is created by an admin.

```
Election.php
4
5 public function send_invite($type,$id){
6     $email_list = "janithae@gmail.com";
7     $selection = json_decode(json_encode($this->PollModel->getElectionById($id)), true);
8
9     if ($type == "all"){
10        $email= json_decode(json_encode($this->UserModel->getAllUserEmail()), true);
11        foreach($email as $i){
12            $email_list = $email_list.", ".$i['email'];
13        }
14    }
15    if ($type == "city"){
16        $city = $this->input->post('city');
17        $email= json_decode(json_encode($this->UserModel->getUserDetailsByCity($city)), true);
18        foreach($email as $i){
19            $email_list = $email_list.", ".$i['email'];
20        }
21    }
22    if ($type == "il1"){
23        $email= json_decode(json_encode($this->UserModel->getUserWithAdditionalDataEmailByIncomeLevel(1)), true);
24        foreach($email as $i){
25            $email_list = $email_list.", ".$i['email'];
26        }
27    }
28    if ($type == "il2"){
29        $email= json_decode(json_encode($this->UserModel->getUserWithAdditionalDataEmailByIncomeLevel(2)), true);
30        foreach($email as $i){
31            $email_list = $email_list.", ".$i['email'];
32        }
33    }
34    if ($type == "il3"){
35        $email= json_decode(json_encode($this->UserModel->getUserWithAdditionalDataEmailByIncomeLevel(3)), true);
36        foreach($email as $i){
37            $email_list = $email_list.", ".$i['email'];
38        }
39    }
40 }
```

Figure 25 - Invitation Sending Code 1

```

Election.php
9
10
11 if ($type == "il4"){
12     $email= json_decode(json_encode($this->UserModel->getUserWithAdditionalDataEmailByIncomeLevel(4)), true);
13     foreach($email as $i){
14         $email_list = $email_list." ".$i['email'];
15     }
16 }
17
18 if ($type == "ed1"){
19     $email= json_decode(json_encode($this->UserModel->getUserWithAdditionalDataEmailByEducationLevel(1)), true);
20     foreach($email as $i){
21         $email_list = $email_list." ".$i['email'];
22     }
23 }
24
25 if ($type == "ed2"){
26     $email= json_decode(json_encode($this->UserModel->getUserWithAdditionalDataEmailByEducationLevel(2)), true);
27     foreach($email as $i){
28         $email_list = $email_list." ".$i['email'];
29     }
30 }
31
32 if ($type == "ed3"){
33     $email= json_decode(json_encode($this->UserModel->getUserWithAdditionalDataEmailByEducationLevel(3)), true);
34     foreach($email as $i){
35         $email_list = $email_list." ".$i['email'];
36     }
37 }
38
39 if ($type == "ed4"){
40     $email= json_decode(json_encode($this->UserModel->getUserWithAdditionalDataEmailByEducationLevel(4)), true);
41     foreach($email as $i){
42         $email_list = $email_list." ".$i['email'];
43     }
44 }
45
46 }

```

Figure 26 - Invitation Sending Code 2

```

Election.php
3
4 $curl = curl_init('https://pos.janitha.biz/pb/send_email/1kjs025s7kljdjqd6wsd6fgdkjdfx798sfsdfsdf453fsdsdfsdfuhsd257dfh/' . $email_list.");
5
6 curl_setopt($curl, CURLOPT_RETURNTRANSFER, true);
7
8 // Send the request
9 $result = curl_exec($curl);
10
11 // Free up the resources $curl is using
12 curl_close($curl);
13
14 $res = json_decode($result, true);
15 //
16 // var_dump($res);
17 if ($res['status'] == 1){
18     $data['error'] = "Email Sent .";
19     return $this->load->view('error_page', $data);
20 }else{
21     $data['error'] = "Email Sent..";
22     return $this->load->view('error_page', $data);
23 }
24
25 }
26
27
28
29
30

```

Figure 27- Invitation Sending Code 3

## 4.4.6 Result sending process

The code segment given below shows the sending results to a voter's Email address when the poll is ended by an admin.

## 4.4.7 Report Generation

The code segment given below shows user-wise report generation. There are three Donut charts displayed by using this code

```
Report.php
9 public function poll_report()
10 {
11     if (!$this->session->has_userdata('auth')) {
12         return redirect('login', 'refresh');
13     }
14     if ($this->session->userdata('admin') == 0){
15         $data['error'] = "Only Admin Can Access This .!";
16         return $this->load->view('error_page', $data);
17     }
18     $data['poll'] = json_decode(json_encode($this->PollModel->getElections()), true);
19     $data['poll_count'] = json_decode(json_encode($this->PollModel->getElectionCountByUser()), true);
20
21     $this->load->view('layouts/header');
22     $this->load->view('report/poll_report' , $data);
23     $this->load->view('layouts/footer');
24 }
25
26 public function user_report($location="Colombo", $incomelevel=1, $edulevel=1, $numchild=0, $gender=1)
27 {
28     if (!$this->session->has_userdata('auth')) {
29         return redirect('login', 'refresh');
30     }
31     if ($this->session->userdata('admin') == 0){
32         $data['error'] = "Only Admin Can Access This .!";
33         return $this->load->view('error_page', $data);
34     }
35
36     $data['users']= json_decode(json_encode($this->ReportModel->getUserswithparam($location, $incomelevel, $edulevel, $numchild, $gender))
37
38     $this->load->view('layouts/header');
39     $this->load->view('report/user_report' , $data);
40     $this->load->view('layouts/footer');
```

Figure 28 – Poll Report Generation Code

```

Report.php
6     public function user_report_search()
7     {
8         if (!$this->session->has_userdata('auth')) {
9             return redirect('login', 'refresh');
10        }
11        if ($this->session->userdata('admin') == 0) {
12            $data['error'] = "Only Admin Can Access This .!";
13            return $this->load->view('error_page', $data);
14        }
15
16        $location = $this->input->post('location');
17        $incomelevel = $this->input->post('incomelevel');
18        $edulevel = $this->input->post('educationlevel');
19        $numchild = $this->input->post('numofchild');
20        $gender = $this->input->post('gender');
21
22        if(isset($location) == false){
23            $location = "";
24            $incomelevel = 1;
25            $edulevel = 1;
26            $numchild = 1;
27            $gender = 0;
28        }
29
30        $data['users'] = json_decode(json_encode($this->ReportModel->getUserswithparam($location, $incomelevel, $edulevel, $numchild, $gender)));
31
32        $this->load->view('layouts/header');
33        $this->load->view('report/user_report' , $data);
34        $this->load->view('layouts/footer');
35    }
36
37 }
38

```

Figure 29 - User Report Generation Code

## 4.5 Reusable Components

- Htaccess is used to rewrite URLs to improve user-friendliness. (Hendrix, 2022)
- DataTables are used to generate graphs for reports in the system. (Add advanced interaction CONTROLSTO YOUR HTML tables the Free & Easy Way)

# Chapter 5 – Testing and Evaluation

## 5.1 Related Testing Types

The success of every project is relay on the testing process. It aims at determining whether the promises about the product by the supplier and the needs of the customer are met on an acceptable level. This chapter describes the process of verifying and validating the system in a structured predefined manner.

The testing process includes verification and validation. Verification is done at all stages of system development to check whether the system is developed correctly. Validation is done after the implementation of the executable system to check the system requirements given by the clients at the initial stages against the features of the final system.

### 5.1.1 Unit Testing

Software programs are subjected to unit testing while being developed (coded). Unit testing aims to isolate a specific piece of code and validate its accuracy. When I'm developing this system I tested every functions and modules functionality immediately after the coding was done. It helped me to correct issues before user testing. So it minimized the major issues and helped to increase user satisfaction.

### 5.1.2 Regression Testing

In this testing, we have to rerun functional and non-functional testing methods to check whether there are any failures due to new modifications. This will confirm that there are not any failures in the system due to new modifications. This testing needs to run many times and if we can use the automated testing technique for this testing it will be very effective.

### **5.1.3 Integrated Testing**

An integrated testing technique will be used in the final phase. Multiple methods are used to assemble and test modules together. The smallest testable component of the system is referred to as a unit in this scenario. Before issues arise during actual program execution, integration testing can reveal interface issues between software components.

### **5.1.4 User Acceptance Test**

User acceptance testing will go in line with the implementation of the system. The manual process also goes parallel with the preliminary implementation stage and the system will be tested by the exact end user. Most of the time minor changes will capture at this stage. The testing process is almost done at the end of this stage.



## 5.2 Test Cases

This is a document of test data with pre-conditions, steps, and actual and expected results. A properly planned test case should verify the relevant system component function. The system is developed and tested module by module to reduce the complexity of both development and testing. Following are test cases for some of the major system modules

<b>ID</b>	<b>Description</b>	<b>Pre-condition</b>	<b>Steps</b>	<b>Expected Result</b>	<b>Actual Result</b>
1	Validate user login with correct input.	<ol style="list-style-type: none"> <li>1. Need an internet connection.</li> <li>2. The user should be on the login page.</li> </ol>	<ol style="list-style-type: none"> <li>1. The user should give either a valid email, mobile number, or NIC.</li> <li>2. Enter OTP which send to the mobile number.</li> </ol>	The user should be successfully logged into the system.	The user logged into the system.
2	Validate user login with incorrect input.	<ol style="list-style-type: none"> <li>1. Need an internet connection.</li> <li>2. The user should be on the login page.</li> </ol>	Either email, NIC, Mobile Number, or OTP is invalid.	Send the user to the login page after displaying the error message.	Displayed the error message and redirect to the login page.
3	Verify the user email with the correct OTP.	<ol style="list-style-type: none"> <li>1. The user should be logged into the system.</li> <li>2. User in the profile update page.</li> </ol>	<ol style="list-style-type: none"> <li>1. The user should click verify email button.</li> <li>2. Then enter OTP send to that email address.</li> </ol>	If OTP is correct then the email is verified.	The email status changed to verified.

<b>ID</b>	<b>Description</b>	<b>Pre-condition</b>	<b>Steps</b>	<b>Expected Result</b>	<b>Actual Result</b>
4	Verify user email with incorrect OTP.	<ol style="list-style-type: none"> <li>1. The user should be logged into the system.</li> <li>2. The user should be logged into the system.</li> </ol>	<ol style="list-style-type: none"> <li>1. The user should click verify email button.</li> <li>2. Then enter OTP send to that email address.</li> </ol>	If OTP is incorrect then the email is not verified.	Email status not changed to verified.
5	Verify the correct NIC Via the government database.	<ol style="list-style-type: none"> <li>1. The user should be logged into the system.</li> <li>2. User in the profile update page.</li> </ol>	The user should enter a valid NIC number.	If NIC is matched with the government database, NIC is verified.	NIC successfully saved.
6	Verify incorrect NIC Via the government database.	<ol style="list-style-type: none"> <li>1. The user should be logged into the system.</li> <li>2. User in the profile update page.</li> </ol>	The user should enter an invalid NIC number.	If NIC is not matched with the government database NIC is rejected.	Display an error message that the NIC is not in a government database.
7	Verify correct user location (election division)	<ol style="list-style-type: none"> <li>1. The user should be logged into the system.</li> <li>2. User in the profile update page.</li> </ol>	The user should enter a valid location	The location should add to the user profile.	Location successfully added to the user profile.
8	Verify incorrect user location (election division)	<ol style="list-style-type: none"> <li>1. The user should be logged into the system.</li> <li>2. User in the profile update page.</li> </ol>	The user should enter an invalid location.	Display the invalid location error message.	Displayed invalid location error message.

<b>ID</b>	<b>Description</b>	<b>Pre-condition</b>	<b>Steps</b>	<b>Expected Result</b>	<b>Actual Result</b>
9	Verify the make-vote process without validation.	<ol style="list-style-type: none"> <li>1. The user should be logged into the system.</li> <li>2. User in the voting page.</li> <li>3. All validations should be completed</li> </ol>	<ol style="list-style-type: none"> <li>1. Select relevant poll</li> <li>2. Select question</li> <li>3. Vote for the answer</li> </ol>	The given vote should be marked on the voting page.	Successfully added vote to the given answer.
10	Verify make vote process without validation	<ol style="list-style-type: none"> <li>1. The user should be logged into the system.</li> <li>2. User in the voting page.</li> <li>3. All validations are not completed</li> </ol>	<ol style="list-style-type: none"> <li>1. Select relevant poll</li> <li>2. Select question</li> <li>3. Vote for the answer.</li> </ol>	Display error message with relevant validation is not completed.	Displayed an error message with relevant validation is not completed.
11	Verify poll creation	<ol style="list-style-type: none"> <li>1. Admin should be logged into the system.</li> <li>2. Admin in the create poll page.</li> </ol>	<ol style="list-style-type: none"> <li>1. Create a new poll.</li> <li>2. Add questions for the poll.</li> <li>3. Add answers to questions.</li> </ol>	A new poll should appear in the poll list.	A new poll added to the poll list
12	Verify the invitation to the poll	<ol style="list-style-type: none"> <li>1. Admin should be logged into the system.</li> <li>2. Admin in the poll list page.</li> </ol>	<ol style="list-style-type: none"> <li>1. Select the poll.</li> <li>2. Press the invite button and select the audience.</li> <li>3. Send invitation.</li> </ol>	Email invitations should be sent to the selected audience	Email received with a link to vote for the relevant poll.
<b>ID</b>	<b>Description</b>	<b>Pre-condition</b>	<b>Steps</b>	<b>Expected Result</b>	<b>Actual Result</b>

13	End poll	<ol style="list-style-type: none"> <li>Admin should be logged into the system.</li> <li>Admin in the poll list page.</li> </ol>	<ol style="list-style-type: none"> <li>Select the poll.</li> <li>Press the End button.</li> </ol>	The Poll should be disabled and can't vote further.	Display an error message that the poll is ended.
14	Verify the death status of the user	<ol style="list-style-type: none"> <li>The user should be logged into the system.</li> <li>User in the voting page.</li> </ol>	Select a relevant poll to vote.	Display an error message with the death status.	The error message is displayed and blocked the voting.
15	Verify age before voting	<ol style="list-style-type: none"> <li>The user should be logged into the system.</li> <li>User in the voting page</li> </ol>	Select a relevant poll to vote.	Display an error message that the voter is under 18 years old.	The error message is displayed and blocked the voting.

Table 3 - Important Test Cases

### 5.3 User Evaluation

The system is tested in the user environment to get the user acceptance testing. User acceptance testing is also called beta testing or end-user testing. This is done after the implantation process in the system development life cycle. This is the stage in the software development process in which the system is tested within the user’s environment.

To collect data for the evaluation process, the sample has to identify. The sample was selected according to the user level defined in the system. The system sends to various users depending on their educational level, income level, location, etc. Above users evaluated the prototype thoroughly and find the strengths and weaknesses of the prototype, and gave their feedback through the evaluation form. This evaluation form was given after the system implementation. Refer to the below table.

No	Statement	Strongly Agree	Agree	Neutral	Disagree	Strongly disagree
1	This system fulfills the e-voting facility for an opinion poll.	5	17	1		
2	Users can understand the system easily.	7	16			
3	This system responds quickly.	8	11	4		
4	System interfaces are very attractive.	8	12	3		
5	Users can navigate through the system easily.	10	12	1		
6	System error messages help to understand, what the mistake was.	11	11	1		
7	System reports help to make decisions quickly.	4	17	2		
8	System increases efficiency and productivity of the E-voting process	9	14			
	Total	62	110	12		

Table 4 - summary of the user evaluation result

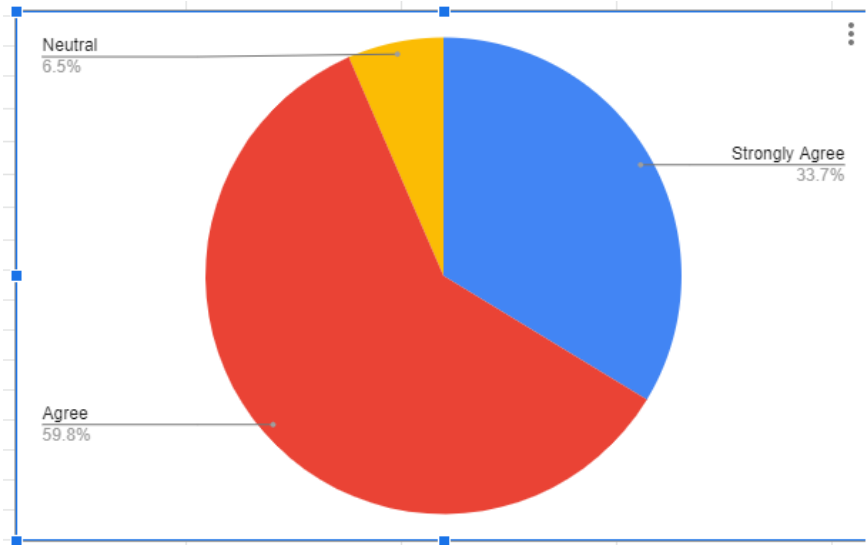


Figure 30 - Pie chart of the evaluation summary

According to the evaluation result, there is no negative feedback. That means users are happy with the new system implementation.

## 5.4 Results of the Testing

Test cases are executed throughout the system development life cycle. Some test cases failed during the initial versions of prototypes and those errors are corrected in the next prototype model till all the required test cases are working without any errors. The below table shows the test case execution status of the system

Module	Pass	Fail	On hold	Not Executed
<i>Login authentication</i>	2	0	0	0
<i>Email verification</i>	2	0	0	0
<i>Location verification</i>	2	0	0	0
<i>NIC verification</i>	2	0	0	0
<i>Death status verification</i>	1	0	0	0
<i>Age verification</i>	1	0	0	0
<i>Voting process</i>	2	0	0	0
<i>Poll creation process</i>	3	0	0	0
<b>Total</b>	<b>15</b>	<b>0</b>	<b>0</b>	<b>0</b>

Table 5 - Test execution status

# Chapter 6 - Conclusion

The conclusion chapter here illustrates a total overview of the developed system with its constraints, achievements, and drawbacks. And this will summarize the author's reflection on the whole project work including a critical appraisal of the successes and limitations of the project approach and the development of the system itself.

## 6.1 Introduction

The achievements of aims and objectives of this study can be considered as very outstanding. The creation of a secure authentication facility to check to validate users logging into the voting system using OTP, mobile number, Email, and voter NIC number was achieved successfully allowing users of the system to use the system in a user-friendly manner.

A database was created effectively to be used to store votes and user information on the system. Implementation of a security method to use to ensure that votes being casting the system will not be compromised and any outside attack was very efficient and robust. The creation of the backend administration section which will be used to enable the administration to manage the poll system effectively was done well ensuring user-friendliness for the administrator team of the system as well. Voters are also prevented from voting more than once for their chosen poll.

Therefore, in conclusion, it is essential to mention that this system would be fulfilling a prudent requirement in the current Sri Lankan society. The voters of the system are allowed to vote with the use of their own NIC number, Email, mobile number, and OTP. Then, those votes are recorded in the database. Then the internet acts as the medium interconnecting all the voters in different locations and the administrators. The administrator is provided with the privilege to log on to the system from his account created and to analyze the votes cast by the voters. The system is a new development to the current market software since the security and validation features are achieved interestingly and solidly which allows the users to be more confident in using the system. The advantages such as user-friendliness, inexpensiveness, and the easiness to analyze the votes cast are very crucial in considering this newly developed system.

In conclusion, it is a pleasure to note that the efforts that are invested in creating this system after properly analyzing the requirements have been successful with the output of this newly designed system software. With the modern world transformation to the Global village concept with the evolution of the internet, the world this type of e-voting concept will be highly demanded and highly useable. The internet is accessible to most parts of the world and the IT literacy of humans is also improving drastically. So this type of system can be very easily adopted to enhance the performance of the opinion poll process of the country.

## **6.2 Lesson Learnt**

As a student studying a recognized master's program, I have learned more in a theoretical way. This project provided me with a lot of practical-oriented knowledge by applying those theories to a real-world working system. I am trained to complete this project with conditions such as a limited period. It also supported me to improve my confidence, coding ability, communication skills, and report-writing skills.

I am experienced with new web languages such as Ajax and also got more knowledge of web technologies. I could learn to apply those technologies properly and apply coding standards.

## **6.3 Critical Assessment of the project**

At the beginning of the project according to the requirements I planned to validate user location by getting their longitude and latitude. But when we consider the practical scenario user can travel to any place and can add a fake location to their profile, this is not the requirement that needs to implement and this can lead to fake profiles in this system. So then change the plan and used a sample government database to validate this user's location and NIC number along with the death status of the user. This will avoid creating profiles with incorrect user location NIC and block make the voting process for deceased users.



## 6.4 System Weaknesses

The system consists of some minor weaknesses and some of the important points are highlighted below.

### ➤ **Can't trace voters**

The implemented system does not support a vote trace facility for administrators or any other person. In some situations, this can be a weakness of the system because according to the poll results admin make decisions, and sometimes if they need to contact the voter it is not possible by using this system.

### ➤ **Sign up with stolen data**

If a user steals data such as NIC number and also he knows the election division of the stolen person he can validate his profile with fake details. But he can't enter fake email and fake mobile number to his profile because of the validation process.

## 6.5 Future Enhancement

This is an online e-voting system. There are some suggestions to improve the system in the future. Following are them

### ➤ **Connecting with API (Application Programming Interface)**

I'm validating most of the user details such as NIC, Death status, and Election division from a sample government database. But if there is a real Application Programming Interface then this validation will be more accurate and up to date with user data.

### ➤ **User polls with approval**

The current system does not allow the creation of user polls with admin approval. Adding create a poll function with admin approval will limit adding useless polls and the system's overall performance can be improved.

➤ **Automatic poll ending**

Polls should be ended by admins after analyzing the specific requirement or end date of the poll. If we can add the pre-requirements for the poll then it can automatically end the poll after meeting the specific requirement and this will be a more effective way to end the poll.

## List of References

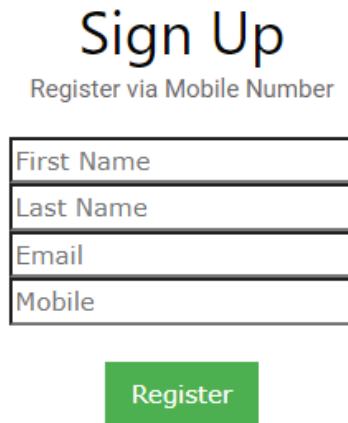
- Srilankalaw. Lk. 2021. Parliamentary Elections Act | Volume VI. [online] Available at: <<https://www.srilankalaw.lk/Volume-VI/parliamentary-elections-act.html>> [Accessed 26 November 2021].
- DataReportal – Global Digital Insights. 2021. Digital in Sri Lanka: All the Statistics You Need in 2021 — DataReportal – Global Digital Insights. [online] Available at: <<https://datareportal.com/reports/digital-2021-sri-lanka>> [Accessed 26 November 2021].
- Aceproject.org. 2021. Internet Voting —. [online] Available at: <<https://aceproject.org/ace-en/topics/et/eth/eth02/eth02b/eth02b4>> [Accessed 25 December 2021].
- 2021. ONLINE VOTING SYSTEM - The IEEE Maker Project. [online] Available at: <<https://transmitter.ieee.org/makerproject/view/91d35>> [Accessed 25 December 2021].
- Google Books. 2022. *Software Systems Safety*. [online] Available at: <[https://books.google.lk/books?hl=en&lr=&id=KuqbCAAAQBAJ&oi=fnd&pg=PA135&dq=e+voting+systems+&ots=royN9EHy58&sig=4Zxkjm\\_dEmS5LGDYmhDZNjW2XXU#v=onepage&q=e%20voting%20systems&f=false](https://books.google.lk/books?hl=en&lr=&id=KuqbCAAAQBAJ&oi=fnd&pg=PA135&dq=e+voting+systems+&ots=royN9EHy58&sig=4Zxkjm_dEmS5LGDYmhDZNjW2XXU#v=onepage&q=e%20voting%20systems&f=false)> [Accessed 8 June 2022].
- Estoniaevoting.org. 2022. *Research Paper | Independent Report on E-voting in Estonia*. [online] Available at: <<https://estoniaevoting.org/findings/paper/>> [Accessed 8 June 2022].
- Jhalderm.com. 2022. *Ethical Issues in E-Voting*. [online] Available at: <<https://jhalderm.com/pub/papers/voting-wecsr11.pdf>> [Accessed 8 June 2022].
- Avirubin.com. 2022. *Analysis of an Electronic Voting System*. [online] Available at: <<https://avirubin.com/vote.pdf>> [Accessed 8 June 2022].

- Jatit.org. 2022. *NEW APPLIED E-VOTING SYSTEM*. [online] Available at: <<http://www.jatit.org/volumes/research-papers/Vol25No2/4Vol25No2.pdf>> [Accessed 8 June 2022].
  
- *UML 1.4 versus UML 2.0 as languages to describe software ... - researchgate* (no date). Available at: [https://www.researchgate.net/publication/221105847\\_UML\\_14\\_versus\\_UML\\_20\\_as\\_Languages\\_to\\_Describe\\_Software\\_Architectures](https://www.researchgate.net/publication/221105847_UML_14_versus_UML_20_as_Languages_to_Describe_Software_Architectures) (Accessed: November 2, 2022).
  
- *Add advanced interaction CONTROLSTO YOUR HTML tables the Free & Easy Way* (no date) *DataTables*. Available at: <https://datatables.net/> (Accessed: November 18, 2022).
  
- Hendrix, N. (2022) *Using .htaccess files for pretty urls, Code Envato Tuts+*. Envato Tuts. Available at: <https://code.tutsplus.com/tutorials/using-htaccess-files-for-pretty-urls--net-6049> (Accessed: November 18, 2022).

# Appendix A – User Documentation

## Sign up

- Go to the <https://evoting.techyfish.com/> URL then you will get the following user interface.
- Then enter your first name, last name, valid email address, and valid mobile phone number, and Click the “Register” button to sign up as a user of the opinion poll system.



Sign Up

Register via Mobile Number

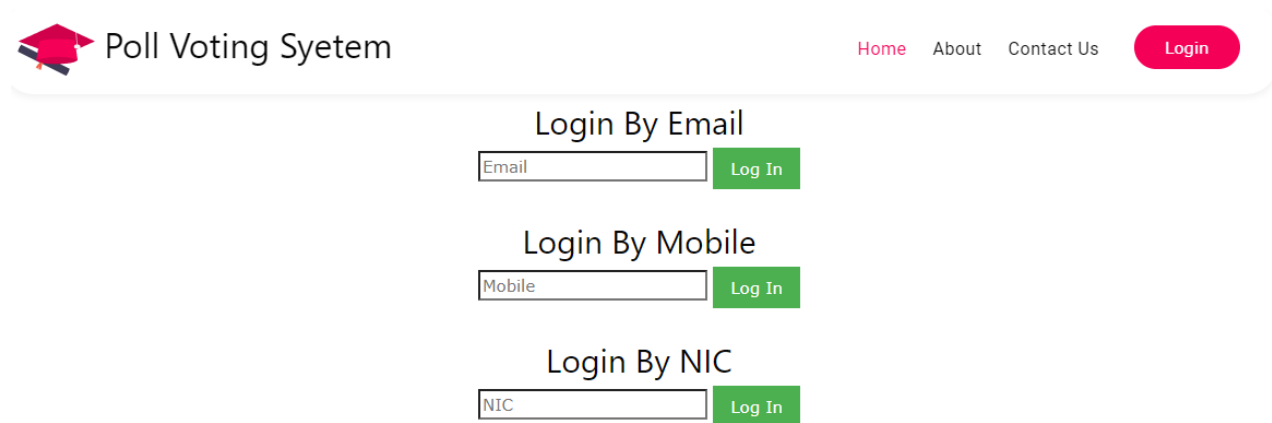
First Name
Last Name
Email
Mobile

Register

Figure 31 - Sign Up Page

## Login

- Enter the registered Email, Mobile, or NIC number and press the “Log In” button.



Poll Voting System

Home About Contact Us Login

Login By Email

Email Log In

Login By Mobile

Mobile Log In

Login By NIC

NIC Log In

Figure 32 - Login Page

- Then you will get OTP to your given mobile number. Enter the OTP and press the “Verify” button.

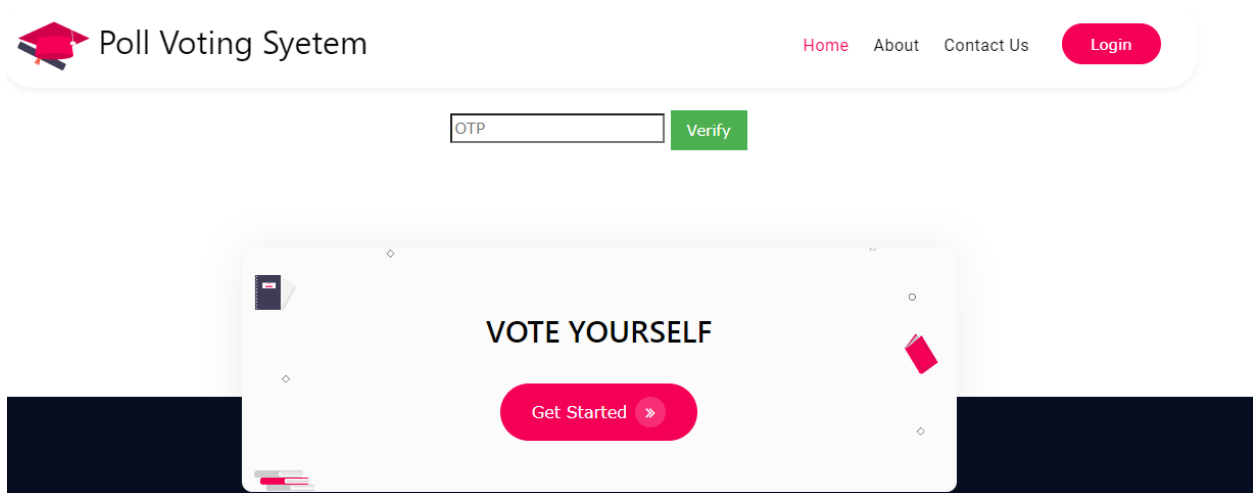


Figure 33 - OTP Entering Page

- Then you will navigate to the home page as shown below. Press the button with the label of your first name to go to the voting page.

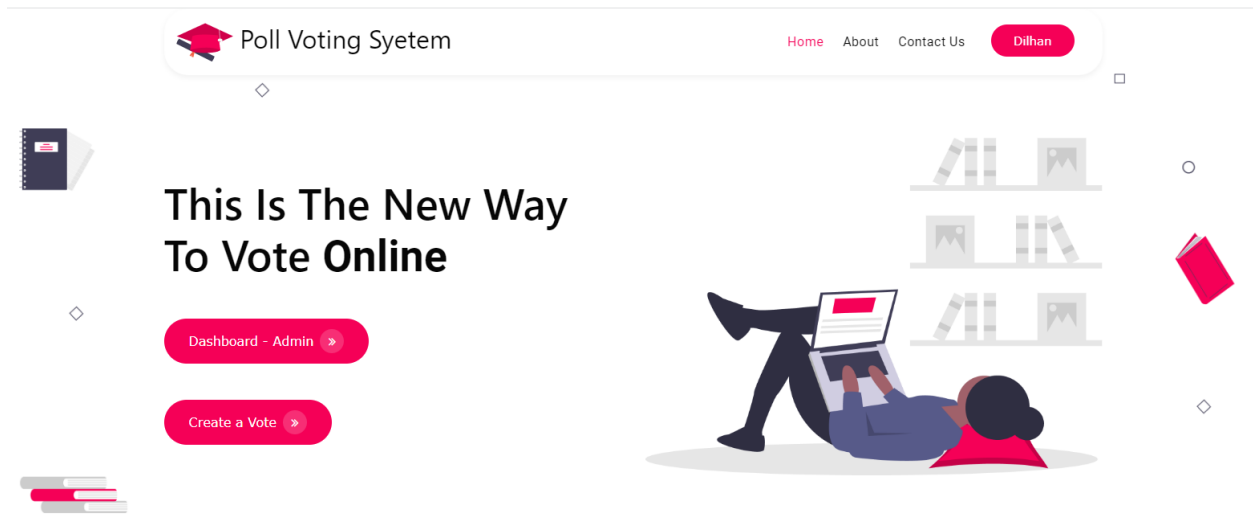


Figure 34 - Dashboard Page 1

## Voting for a poll

- Scroll down the home page and you will see the below summary of polls. You can vote by clicking the “Vote” button and also can see current progress by clicking the “Result” button.

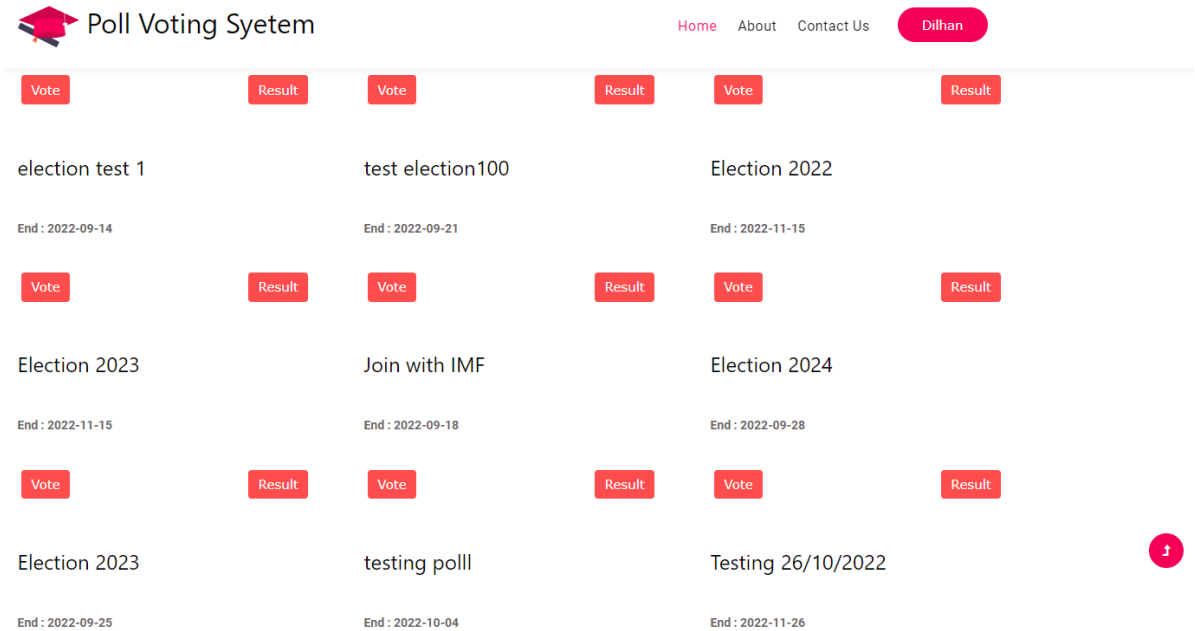


Figure 35 - Dashboard Page 2

- The following user interface shows the available polls and you can vote by clicking the “Vote” button. You can't vote for the ended polls which are highlighted.

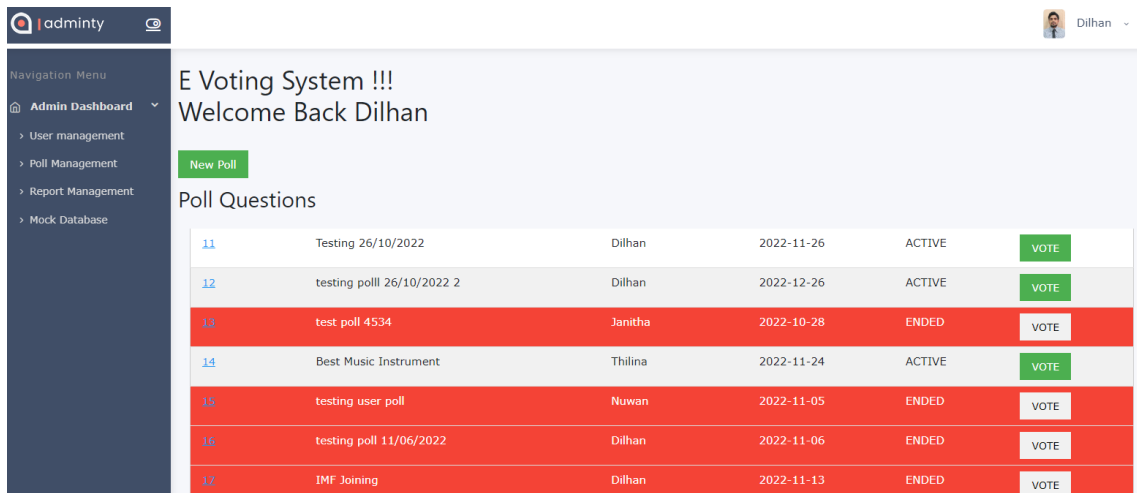


Figure 36 - Poll Selecting Page

- You can select a different question and give your answer by clicking the “Vote” button.
- Can enter a comment for each question and press the “Save ” button to add a comment.

## Poll - Submit Vote

Poll Name : IMF Joining

Question : Can we improve the economy by joining with IMF?

Answer	Vote
Yes	<input type="button" value="VOTE"/>
No	<input type="button" value="VOTE"/>
Not Sure	<input type="button" value="VOTE"/>

### Comment

Your Comment

### Other Poll Questions

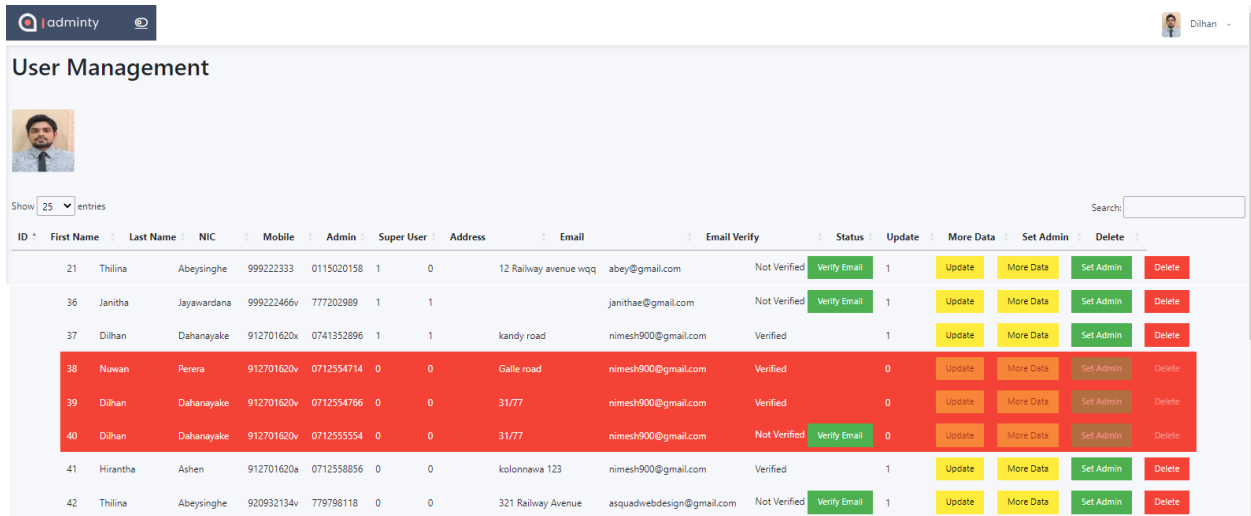
Name	Edit	
Can we improve the economy by joining with IMF?	<input type="button" value="Vote"/>	<input type="button" value="Result"/>
How much loan do we need?	<input type="button" value="Vote"/>	<input type="button" value="Result"/>

Figure 37 - Voting Page



## User Management

- By clicking “User management” on the sidebar you can the below user management page.
- Super admin can Disable users and Set users as admins by clicking the “Delete” and “Set Admin buttons respectively”



ID	First Name	Last Name	NIC	Mobile	Admin	Super User	Address	Email	Email Verify	Status	Update	More Data	Set Admin	Delete	
21	Thilina	Abeysinghe	999222333	0115020158	1	0	12 Railway avenue wqq	abey@gmail.com	Not Verified	Verify Email	1	Update	More Data	Set Admin	Delete
36	Janitha	Jayawardana	999222466v	777202989	1	1		janithae@gmail.com	Not Verified	Verify Email	1	Update	More Data	Set Admin	Delete
37	Dilhan	Dahanayake	912701620x	0741352896	1	1	kandy road	nimesh900@gmail.com	Verified	Verify Email	1	Update	More Data	Set Admin	Delete
38	Nuwari	Perera	912701620v	0712554714	0	0	Galle road	nimesh900@gmail.com	Verified	Verify Email	0	Update	More Data	Set Admin	Delete
39	Dilhan	Dahanayake	912701620v	0712554766	0	0	31/77	nimesh900@gmail.com	Verified	Verify Email	0	Update	More Data	Set Admin	Delete
40	Dilhan	Dahanayake	912701620v	0712555554	0	0	31/77	nimesh900@gmail.com	Not Verified	Verify Email	0	Update	More Data	Set Admin	Delete
41	Hirantha	Ashen	912701620a	0712558856	0	0	kolonnawa 123	nimesh900@gmail.com	Verified	Verify Email	1	Update	More Data	Set Admin	Delete
42	Thilina	Abeysinghe	920932134v	779798118	0	0	321 Railway Avenue	asquadwebdesign@gmail.com	Not Verified	Verify Email	1	Update	More Data	Set Admin	Delete

Figure 38 - user Management Page

## Update Profile

- Click on your name and you will get the below drop-down. Click on the “Profile” button to update your profile details.

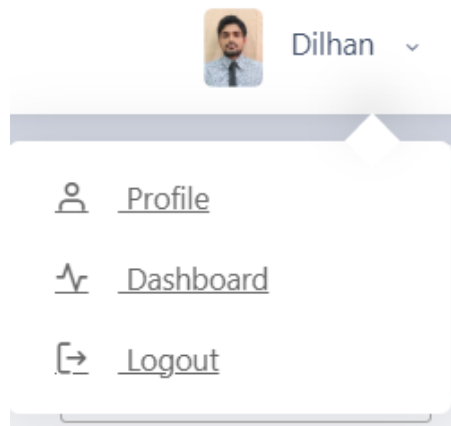


Figure 39 - Dropdown Page

- You will get the below interface and enter all fields and press the “Update” button. Select a photo with PNG type and press the “Upload Image” button to upload a profile picture.
- Press the “Additional Data” button to update additional data details.

Navigation Menu

- Admin Dashboard
- > User management
- > Poll Management
- > Report Management
- > Mock Database

## Update User

First name

Last Name

NIC


E-Mail

Mobile Number

Address

City

### User Profile Photo



Select image to upload:  No file chosen

Figure 40 - User Data Update Page

- You will get the below page to enter all required fields and press the “Save” button to add details to your profile.

Navigation Menu

- Admin Dashboard
- > User management
- > Poll Management
- > Report Management
- > Mock Database

## Update User Additional Data

Office Name

Experience

Birthday

Marital status (Yes = 1/No =0)

Children Count

Gender : Please select the gender

Highest Educational Level:

- OL
- AL
- Graduate
- Post Graduate

Income Level:

- Rs 0 - 50 000
- Rs 50 000 - 100 000
- Rs 100 000 - 250 000
- Rs 250 000 Above

Figure 41 - User Additional Data Update Page

## Poll Management

- By clicking “Poll management” on the sidebar you can the below poll management page.
- By clicking the “Result” button you can see the poll result.
- By clicking the “End” button you can end the poll and block the voting process.
- By clicking the “Edit” button you can update the poll details.
- By clicking the “Invitation - Email” button you can send invitations to users.

The screenshot displays the 'Poll Management' page in the 'adminty' dashboard. The page features a navigation sidebar on the left with options: Admin Dashboard, User management, Poll Management, Report Management, and Mock Database. The main content area is titled 'POLL Management' and includes a 'Create New Poll' button. Below this is a table listing 10 poll entries. Each entry has columns for ID, Name, User, User First Name, End Date, Description, Email Invitations, Result, Edit, Delete, and End POLL. The table data is as follows:

ID	Name	User	User First Name	End Date	Description	Email Invitations	Result	Edit	Delete	End POLL
1	election test 1	24	Janitha	2022-09-14	test description	Invitation - Email	Results	Edit	Delete	END
2	test election100	24	Janitha	2022-09-21	electiondescription100	Invitation - Email	Results	Edit	Delete	END
4	Election 2022	27	Dilhan	2022-11-15	Selecting candidates	Invitation - Email	Results	Edit	Delete	END
5	Election 2023	27	Dilhan	2022-11-15	Selecting candidates	Invitation - Email	Results	Edit	Delete	END
7	Join with IMF	27	Dilhan	2022-09-18	collecting data	Invitation - Email	Results	Edit	Delete	END
8	Election 2024	27	Dilhan	2022-09-28	Selecting candidates	Invitation - Email	Results	Edit	Delete	END
9	Election 2023	27	Dilhan	2022-09-25	Selecting candidates	Invitation - Email	Results	Edit	Delete	END
10	testing poll	27	Dilhan	2022-10-04	Selecting candidates	Invitation - Email	Results	Edit	Delete	END

Figure 42 - Poll Management Page

- After clicking the “Invitation - Email” button you will get the below page. Press any button according to the requirement to send an invite. Can send to all users and also it can send income-wise, education-wise, and location-wise.

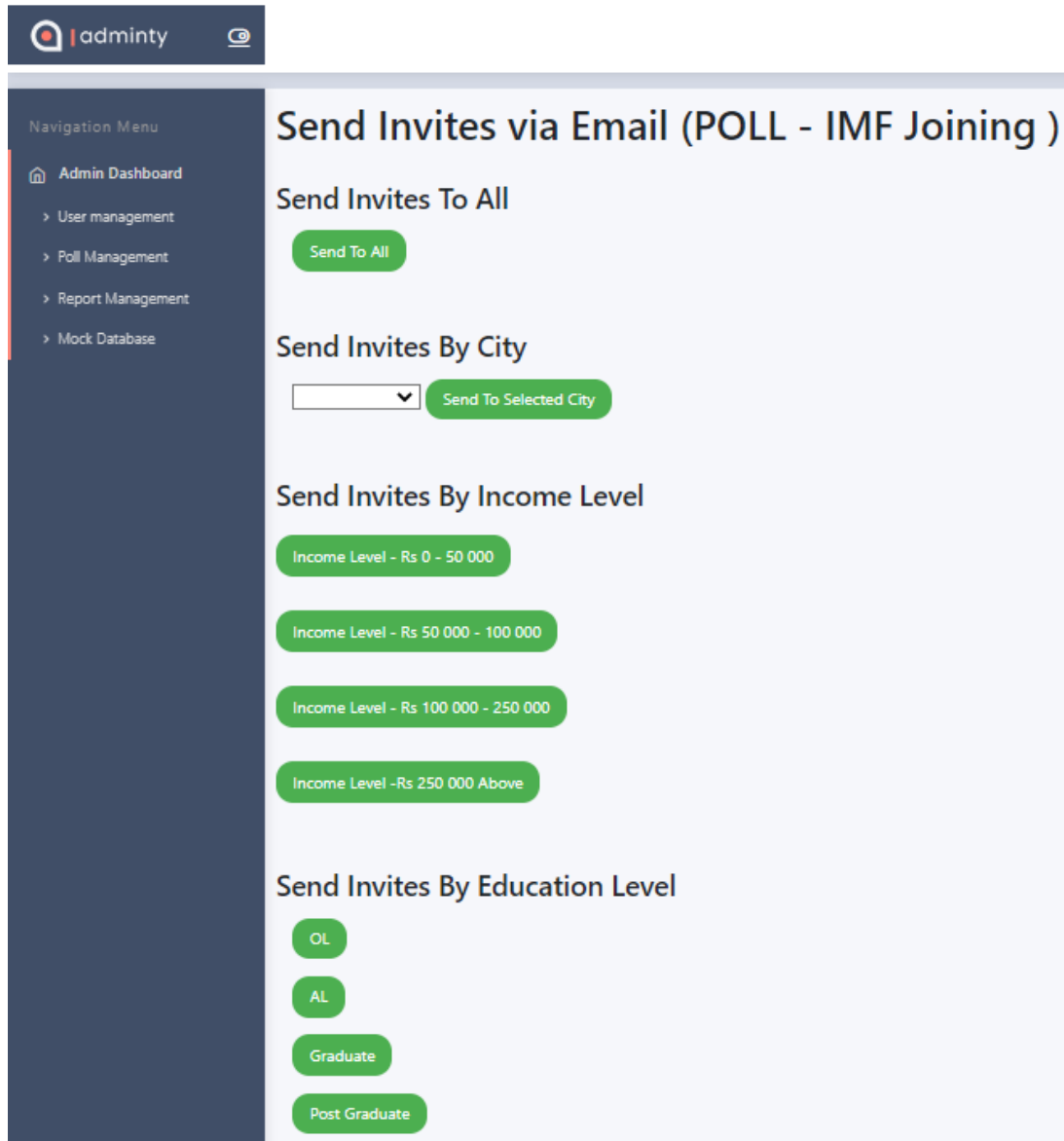


Figure 43 - Invitation Sending Page

## Poll Creation

- Go to poll management and click on the “Create New Poll” button

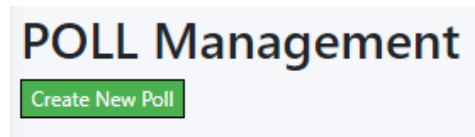


Figure 44 - Poll Creation Button

- Then you will get the below interface. Enter the Poll name, end date, and description, then press the “Create” button to create a new poll.

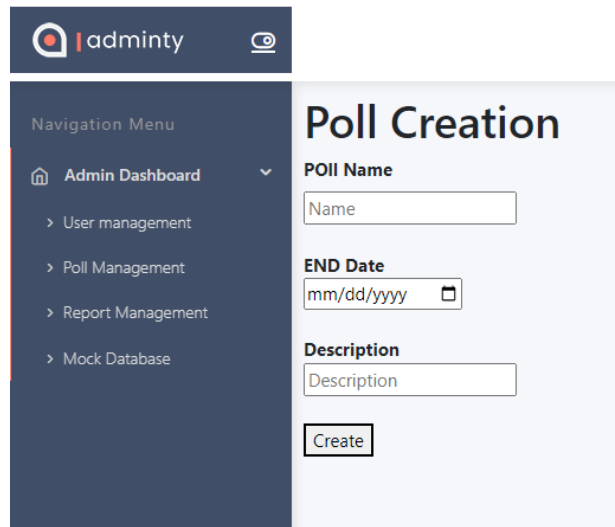


Figure 45 - Poll Creation Page

- Then click the “Add Question for pol” button to add questions
- Give a name for the question and click the “Create” button to add a new question to the poll.

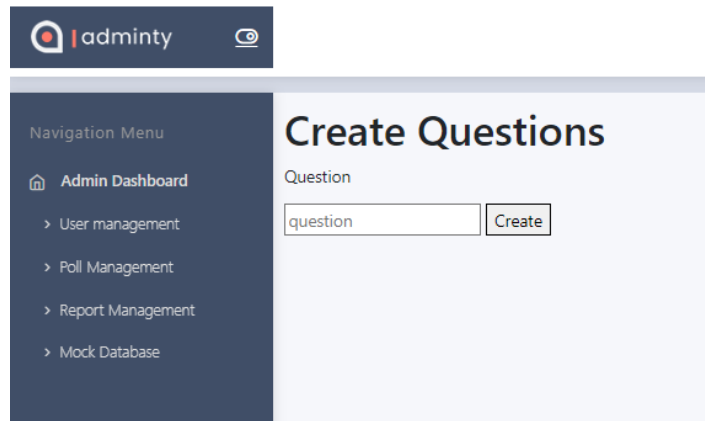


Figure 46 - Question Creation Page

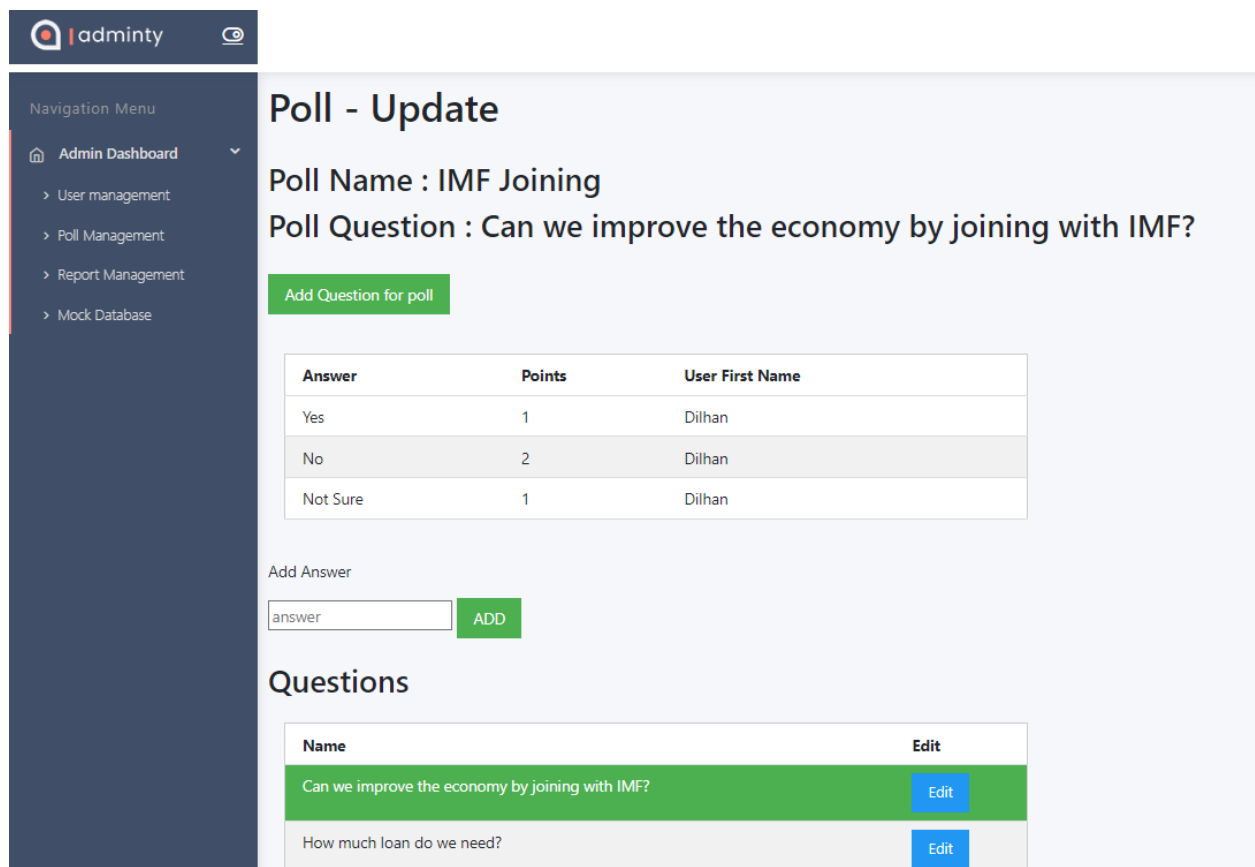


Figure 47 - Answer Adding Page

- After adding the question, you can add answers for each question by clicking the “Add” button

# **Appendix B – MIS Reports**

## **Description of MIS Report Module**

The management needs MIS reports to evaluate the organization's performance and facilitate quicker decision-making. It is a useful system that assists decision-makers in selecting the best course of action to expand their organization. And one of the parts of this well-organized system is MIS reports.

### **Decision Making**

The decision-makers need the information to make effective decisions and the MIS report will fulfill this requirement. Admin can make decisions before creating a poll. They can analyze existing user data and decide where that poll is suitable for the specific area in the country. Let's assume that we want to make a poll with users who have a monthly income over 100000 rupees and the users registered in that area have a less percentage of that income level does not mean creating such a poll in that area.

### **Tracking records**

Management information systems play a crucial role in maintaining records. They provide an accurate summary of all commercial transactions and serve as a benchmark for upcoming operations. This makes it possible for businesses to monitor their development and performance over time, which is crucial for decision-making.



# Evidence of the reports

## User wise reports

First users can filter by their location and get the reports/graphs according to income level, education level, gender, etc.

**User Report**

Location:

Education Level:  Monthly Income:  Number of Child:  Gender:

ID	First Name	Last Name	City	Mobile	Admin	Super User	Address	Experience	Married	Children	Education Level	Monthly income	Email
21	Thilina	Abeyasinghe	Maharagama	0115020158	1	0	12 Railway avenue wqq	12	1	1	Graduate	Rs 250 000 Above	abey
24	Janitha	Jayawardana	Maharagama	777202989	0	0	test address 2	3	1	2	AL	Not Set	janith
27	Dilhan	Dahanayake	Kandy	0741352896	1	1		0	0	4	OL	Not Set	Nimr
28	Dilhan	Dahanayake		0712554711	0	0	31/77	1	1	3	OL	Not Set	nime
37	Dilhan	Dahanayake	kandy	0741352896	1	1	kandy road	6	1	3	AL	Rs 250 000 Above	nime
38	Nuwan	Perera	Galle	0712554714	0	0	Galle road	2	1	2	Post Graduate	Rs 100 000 - 250 000	nime
39	Dilhan	Dahanayake	galle	0712554766	0	0	31/77	2	1	2	Post Graduate	Rs 50 000 - 100 000	nime
40	Dilhan	Dahanayake	galle	0712555554	0	0	31/77	5	1	3	Post Graduate	Rs 50 000 - 100 000	nime
41	Hiranth	Ashen	kolonnawa	0712558856	0	0	kolonnawa 123	2	1	4	Post Graduate	Rs 50 000 - 100 000	nime
43	Thilina	Abeyasinghe	pannipitya	717699107	1	1	321 Railway Avenue	10	1	2	Graduate	Rs 100 000 - 250 000	thilin

Showing 1 to 10 of 12 entries Previous   Next

Figure 48 – User-wise Report Page

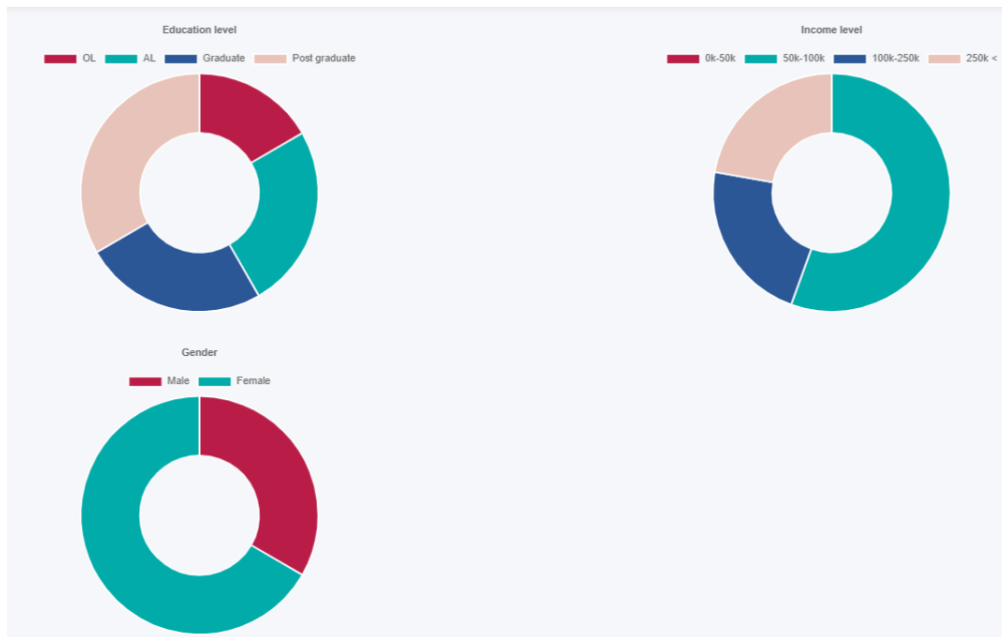


Figure 49 - User-wise Graph Page

## Poll wise reports

The result of the poll can get by the question wise. The count and the percentage of the vote distribution will be shown in the graphs.

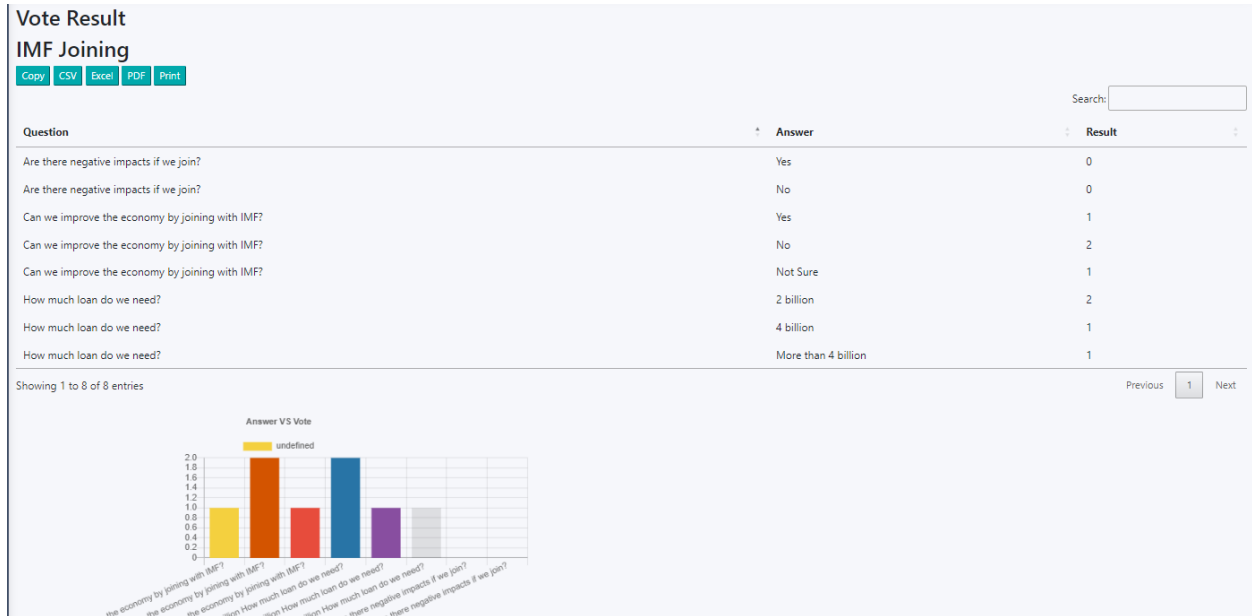


Figure 50 - Poll Result Page

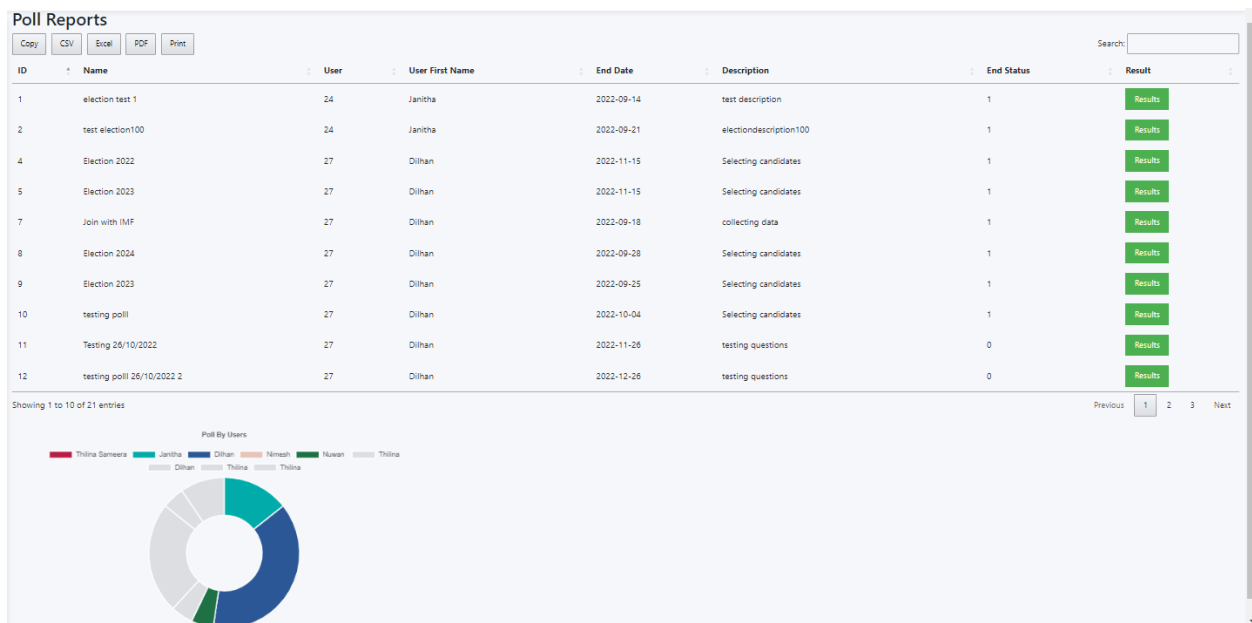


Figure 51 - Poll Report Page

## Decision wise reports

Can download user details and poll details in Excel, PDF, and CSV file formats and can use these data for data mining and decision-making process.

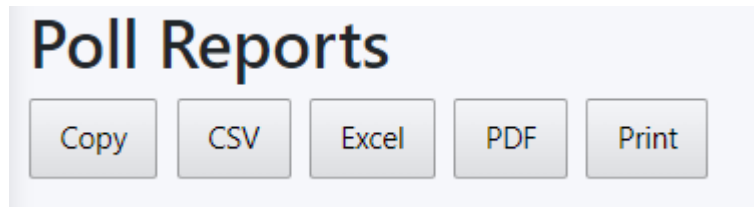


Figure 52 - File Generation Buttons