



Islington college
(इस्लिंग्टन कॉलेज)

Module Code & Module Title

CS6P05NI Final Year Project / Interim Report

Assessment Weightage & Type

75% Individual Coursework

Year and Semester

2019-2020

Student Name: Rojin Kalu

London Met ID: 17031129

External Supervisor: Ishwor Shrestha

Internal Supervisor: Subeskshya Shrestha

Assignment Submission Date: 5-May-2020

Word Count: 8128

I confirm that I understand my coursework needs to be submitted online via Google Classroom under the relevant module page before the deadline in order for my assignment to be accepted and marked. I am fully aware that late submissions will be treated as non-submission and a mark of zero will be awarded.

Acknowledgement

I might want to express my most profound gratefulness to each one of those individuals who gave me the likelihood to finish the report. An appreciation to my mentors Ishwor Shrestha and Subekshya Shrestha, who have the major role to helped me to arrange my coursework particularly in report and Monil Adhikari a module leader.

An exceptionally extraordinary much gratitude goes to my school mates who helped me in settling this coursework inside constrained time spans. I might likewise want to express my exceptional thanks of appreciation to my college London metropolitan college which gave me the brilliant chance to do this superb coursework on the subject networking and working framework, which helped in doing bunches of research I came to know numerous things I am extremely grateful to them.

Abstract

In this coursework, we present the design and implementation of the online voting system. The online voting system gives voters a reliable and highly secure environment to cast their votes using internet terminals, such as PCs, mobile phones.

In this system the voter does not have to go to the polling booth to cast their vote. They can use their personal computer to cast their votes. There is a database which is maintained in which all the name of the voters with their complete information is stored. The System Administrator registers the voters by simply filling a registration form to register the voters. After registration they can vote to party and candidate.

The counting process is fully automated, giving no chance for manipulation of the results. When the results are ready, it will be posted on the result page along with statistics studies and graphics that explain the results.

Table of Contents

Chapter 1: Introduction	1
1.1 Project Description.....	1
1.2 Current Scenario.....	2
1.3 Problem Domain and Project as a Solution.....	2
1.4 Aims and Objectives	3
1.5 Structure of the Report.....	4
1.5.1 Background	4
1.5.2 Development	4
1.5.3 Testing and Analysis.....	4
1.5.4 Conclusion	4
Chapter 2: Background	5
2.1 About the End Users	5
2.2 Understanding the Solution.....	5
2.3 Similar Projects.....	6
2.4 Comparisons	6
Chapter 3: Development	8
3.1 Considered Mythologies	8
3.1.1 Waterfall Model	8
3.1.2 Agile Methodology	10

3.1.3 RUP Methodology	11
3.2 Selected Methodology	12
3.3 Phases of Methodology.....	12
3.4 Survey Results	15
3.4.1 Pre-Survey Results.....	15
3.4.2 Post Survey Results.....	18
3.5 Requirement Analysis.....	21
3.6 Design	26
3.6.1 Use case of Online Voting system.....	26
3.6.2 High Level Use Case	27
3.6.3 Expanded Use Case Description.....	29
3.6.4 ERD of online voting system.....	32
3.6.5 Activity Diagram	34
3.6.6 Data Flow Diagram (DFD)	35
3.6.7 Collaboration diagram	38
3.6.8 Sequence Diagram	39
3.6.9 Wireframes.....	40
3.6.10 Work Break Down Structure	48
3.6.11 Gantt Chart.....	49
3.7 Implementation	50

Chapter 4: Testing and Analysis	58
4.1 Test Plan.....	58
4.1.1 Unit Testing, Test Plan	58
4.1.2 System Testing, Test Plan.....	58
4.2 Unit Testing	59
4.3 System Testing.....	65
4.4 Critical Analysis.....	85
Chapter 5: Conclusion.....	86
5.1 Legal, Social and Ethical Issues.....	86
5.1.1 Legal Issues.....	86
5.1.2 Social Issues.....	86
5.1.3 Ethical Issues	87
5.2 Advantages.....	87
5.3 Limitations	88
5.3 Future Work.....	88
Chapter 6: References	89
Chapter 7: Appendix	92
7.1 Appendix A: Pre-Survey.....	92
7.1.1 Pre-Survey Form.....	92
7.1.2 Sample of Filled Pre-Survey Forms.....	93

7.1.3 Pre-Survey Result	94
7.2 Appendix B: Post-Survey	97
7.2.1 Post-Survey Form	97
7.2.2 Sample of Filled Pre-Survey Forms.....	98
7.2.3 Post-Survey Result.....	99
7.3 Appendix C: Sample Codes	101
7.3.1 Sample Code of the UI.....	101
7.3.2 Sample Code for the Project Script.....	105
7.4 Appendix D: Designs	113
7.4.1 Gantt Chart.....	113
7.4.2 Work Breakdown Structure	114
7.4.3 Algorithms and Flowchart	115
7.4.4 Hardware Architecture.....	116
7.4.5 Data Flow Diagrams	117
7.4.7 Use Case.....	119
7.4.8 Wireframe	120
7.5 Appendix E: Screenshots of the system.....	128
7.6 Appendix F: User Feedback.....	136
7.6.1 User Feedback Form	136
7.6.2 Sample of filled User Feedback forms.....	137

7.7 Appendix G: Future Work 138

 7.7.1 Readings for Future Work 138

Table of Figures

Figure 1: screenshot of similar voting app.....	7
Figure 2: screenshot of polling page demo	7
Figure 3: Waterfall model.....	9
Figure 4: Agile Methodology.....	11
Figure 5: Rational Unified Process (RUP) Methodology	13
Figure 6: pre survey result 1	15
Figure 7: pre survey result 2	15
Figure 8: pre survey result 3	16
Figure 9: pre survey result 4	16
Figure 10: pre survey result 5	17
Figure 11: pre survey result 6	17
Figure 12: pre survey result 7	18
Figure 13: post survey results 1	18
Figure 14: post survey result 2.....	19
Figure 15: post survey result 3.....	19
Figure 16: post survey result 4.....	20
Figure 17: post survey result 5.....	20
Figure 18: post survey result 6.....	21
Figure 19: XAMPP	25
Figure 20: Balsamiq logo.....	25
Figure 21: Use case of online voting system	26
Figure 22: Conceptual ERD of online voting system	32

Figure 23: ERD of online voting system	33
Figure 24: Activity Diagram	34
Figure 25: 0-level dfd of online voting system.....	35
Figure 26: 1-level dfd for admin	36
Figure 27: 1-level dfd for users.....	37
Figure 28: Collaboration diagram.....	38
Figure 29: Sequence Diagram.....	39
Figure 30: Home page of online voting system	40
Figure 31: Wireframe of About page.....	41
Figure 32: wireframe of Login page	42
Figure 33: wireframe of register page.....	43
Figure 34: wireframe of voting page of party	44
Figure 35: voting page of candidate.....	45
Figure 36: wireframe of my profile page	46
Figure 37: Wireframe of contact page	47
Figure 38: Work Break Down Structure	48
Figure 39: Gantt Chart	49
Figure 40: System Architecture	50
Figure 41: Login Page of online voting system	51
Figure 42: Register page of online voting system	52
Figure 43: Home page of online voting system	53
Figure 44: Vote Party page	54
Figure 45: Vote candidate page	55

Figure 46: My profile page	56
Figure 47: Contact page	57
Figure 48: Unit Testing 1.1	59
Figure 49: Unit testing 1.2	60
Figure 50: unit testing 2.1	61
Figure 51: unit testing 2.2	61
Figure 52: Unit testing 3.1	62
Figure 53: unit testing 3.2	62
Figure 54: unit testing 4	63
Figure 55: Unit testing 5	64
Figure 56:Testing 1.1	65
Figure 57: Testing 1.2	66
Figure 58: System Testing 2	67
Figure 59: Testing 3	68
Figure 60: Testing 4.1	69
Figure 61: Testing 4.2	70
Figure 62: Testing 5.1	71
Figure 63: Testing 5.2	71
Figure 64: Testing 6	72
Figure 65: Testing 7	73
Figure 66: Testing 8	74
Figure 67: Testing 9	75
Figure 68: Testing 10	76

Figure 69: Testing 11.1	77
Figure 70: Testing 11.2	78
Figure 71: testing 12.1	79
Figure 72: Testing 12.2	79
Figure 73: Testing 13.1	80
Figure 74: Testing 13.2	81
Figure 75: Testing 14.1	82
Figure 76: Testing 14.2	82
Figure 77: System testing 15.....	83
Figure 78: System Testing 16	84
Figure 79: Pre-Survey Form	92
Figure 80: SAMPLE OF FILLED PRE-SURVEY FORMS	93
Figure 81: result of pre-survey form.....	96
Figure 82: Post-Survey Form.....	97
Figure 83: SAMPLE OF FILLED POST-SURVEY FORMS.....	99
Figure 84: Result of post survey form	101
Figure 85: Screenshot of login page code	101
Figure 86: Screenshot of register page code	102
Figure 87: Screenshot pf coding of home page	102
Figure 88: Screenshot of coding of about page	103
Figure 89: Screenshot of coding account page	103
Figure 90: Screenshot of coding of voting page	104
Figure 91: Screenshot of coding of candidate voting page	104

Figure 92: Screenshot of coding of contact page.....	105
Figure 93: Home Controller.....	105
Figure 94: Vote party controller	106
Figure 95: Vote candidate controller	106
Figure 96: Account controller.....	107
Figure 97: admin controller	107
Figure 98: area controller.....	108
Figure 99: district controller	108
Figure 100: province controller	109
Figure 101: candidate controller	109
Figure 102: location controller.....	110
Figure 103: party controller	110
Figure 104: setting controller.....	111
Figure 105: user controller.....	111
Figure 106: vote count controller.....	112
Figure 107: Gantt Chart	113
Figure 108: Work Break down Structure.....	114
Figure 109: Flow Chart.....	115
Figure 110: System Architecture	116
Figure 111: 0-level DFD	117
Figure 112: 1-Level DFD for admin.....	117
Figure 113: 1-level DFD for voters	118
Figure 114: use case of online voting system	119

Figure 115: login page	120
Figure 116: Register page	121
Figure 117: home page.....	122
Figure 118: About page	123
Figure 119: vote party page	124
Figure 120: vote candidate page	125
Figure 121: My profile page	126
Figure 122: Login Page.....	128
Figure 123: Register Page.....	129
Figure 124: Home page.....	130
Figure 125: About page	131
Figure 126: Vote Party Page	132
Figure 127: Vote Candidate Page	133
Figure 128: My Profile Page.....	134
Figure 129: contact page.....	135
Figure 130: User feedback form	136
Figure 131: Sample of filled User Feedback forms	137

Table of Tables

Table 1: Description of project with RUP methodology in different phases.....	14
Table 2: Expanded Use case of Register.....	29
Table 3: Expanded use case of login.....	30
Table 4: Expanded use case of cast vote.....	31
Table 7: unit testing 1	59
Table 8: unit testing 2	60
Table 9: Unit testing 3.....	62
Table 10: unit testing 4	63
Table 11: Unit testing 5.....	64
Table 12: System Testing 1.....	65
Table 13: Testing 2	67
Table 14: Testing 3	68
Table 15: Testing 4	69
Table 16: Testing 5.1	70
Table 17: Testing 6	72
Table 18: Testing 7	73
Table 19: Testing 8	74
Table 20: Testing 9	75
Table 21: Testing 10	76
Table 22: Testing 11	77
Table 23: Testing 12	78
Table 24: Testing 13	80

Table 25: Testing 14	81
Table 26: Testing 15	83
Table 27: System Testing 16.....	84

Chapter 1: Introduction

The 2000 presidential elections forced most of the organizations around the world to think of a better way of voting that enables as much people to cast their ballots in a convenient way. It was suggested to introduce a system that enables voters to use the internet is a cheap, fast, and effective way to transfer data. Many researchers have studied the benefits and threats of using internet for voting. Most studies raised concerns about voter's authentication, security of ballots during transfer through internet, and maintaining the secret ballot, which means to separate the identity of the voter from the ballot.

Online voting is a not a new concept. In the last three decades, studies tried to come up with a way to create online voting systems that enable peoples to vote while they are at their homes. It was not until the late 90's when the idea became practically possible. Internet revolution helped the idea to grow. Studies expect that online voting will replace the present voting systems sooner or later. (Elleithy et al., 2005)

1.1 Project Description

Online voting systems are software platforms used to securely conduct votes and elections. As a digital platform, they eliminate the need to cast your votes using paper or having to gather in person. Online voting is a not a new concept. Well this Online voting system is based on web application system which user can vote to party and candidate. Nowadays people are getting smarter and developing, so online voting system will be suitable for them as they can vote from their devices like mobile, laptop and other devices.

In this online voting system, user cannot take part in voting without registration. User should first register to vote as system have to count the number of votes from users. In this online voting

system, user should vote to party first. User cannot vote candidate without voting to party. User can vote candidate after party is filtered. What new in this project is user can edit their profile. But user cannot edit citizenship number as they can be double value. This online voting system also have the encryption method which is of Laravel. The username and password are encrypted and decrypted in registration process.

1.2 Current Scenario

In current voting system, which is manual voting system, people have to travel to polling place to vote which is very time consuming. Some people have to travel back to village to vote. Sometimes, there is threat for people in polling area. In manual voting system, it occurs counting error.

1.3 Problem Domain and Project as a Solution

In today's context, there are numbers of people living in abroad. They cannot participate in voting. They cannot choose and vote the party they like. Since, with the help of online voting system, people from abroad also can vote and express their view. Inside the country also, people from villages are living in Kathmandu valley. They have not return to their village to cast a vote. Many political issues are also there in voting area. Sometimes, the voting box keep missing and exchange with another voting box. Hence the online voting system is very useful in Nepal. Online Voting is a web-based voting system that will help you manage your elections easily and securely. Also, the willingness of people to sell or buy votes in elections is still unknown. Online voting system may prevent blackmailing, threatening as people can vote easily from home.

1.4 Aims and Objectives

The main aim of this project is to develop online voting system. The objectives are as follows:

- To reduce the human effort and saving time.
- To increase the voting accuracy.
- To reduce the threat for the people that can arise in voting area.
- The advantage of online voting is that the voters have the choice of voting at specific allocated time.
- It also minimizes vote counting error.
- People living in abroad also can cast a vote through online voting system.
- Online voting provides convenience, cost-saving, and saves the voters dealing with heavy traffic, and bad weather condition.
- It helps millions of disabled and blind people to cast their votes without any assistance.

1.5 Structure of the Report

1.5.1 Background

In this topic, the background of the project is described such as languages, framework and application that will be used in project. Furthermore, this topic will include the similar website of this project. Similarities and Dissimilarities will be discussed in this topic.

1.5.2 Development

In this topic, the detail development of this project is discussed which is up to date. Such as use case, ERD and wireframes are discussed.

1.5.3 Testing and Analysis

In this topic, the progress of the project is analyzed. Does the project is heading with project plan or not by reviewing the Gannt chart.

1.5.4 Conclusion

In this topic, the overall experience of developing this project are expressed and described. Such as hard works and difficulties.

Chapter 2: Background

2.1 About the End Users

The term "end user" refers to the consumer of a good or service, with the added connotation that he or she has an innate know-how unique to consumers. In a literal sense, the term end user is used to distinguish the person who uses the application and good or service from individuals who are involved in the stages of its design, development, and production. (Kenton, 2019)

End user's requirement:

- Application should be error free.
- System user interface should be friendly.
- Final report/Result should be fair. There should be no any flaws in vote Counting process
- Easy in data manipulation.
- User can edit their profile
- Is there is any problem in system, user should let message the admin.

2.2 Understanding the Solution

In today's context, there are numbers of people living in abroad. They cannot participate in voting. They cannot choose and vote the party they like. Since, with the help of online voting system, people from abroad also can vote and express their view. Inside the country also, people from villages are living in Kathmandu valley. They have not return to their village to cast a vote. Many political issues are also there in voting area. Sometimes, the voting box keep missing and exchange with another voting box. Hence the online voting system is very useful in Nepal. Online Voting is a web-based voting system that will help you manage your elections easily and securely. Also, the willingness of people to sell or buy votes in elections is still unknown. Online voting system may prevent blackmailing, threatening as people can vote easily from home.

2.3 Similar Projects

Since there is no online voting system, this system is first one in Nepal.

The Rijnland Internet Election System (RIES) is a system designed for voting in public elections over the internet. The Rijnland Internet Election System (RIES) processed around 90000 votes in public elections in the Netherlands in 2004 and 2006. Based on total votes processed in public elections, RIES is one of the largest internet voting systems worldwide. As an interesting feature, RIES offers cryptographic end to end verifiability. This enables the voter to use cryptography to verify that her or his vote was counted as cast. After some delay, the source code to RIES was published on June 24th, 2008. This paper describes the result of a few days of looking at the source code and documentation of a rather complex internet voting system. This study began when the source code of RIES was published. The first preliminary results of this study were available to the Dutch media and members of parliament four days later on June 28th.
(Ryan & Schoenmakers, 2009)

2.4 Comparisons

Since there is no online voting system, this system is first one in Nepal. Following images are the voting websites.

Similar systems

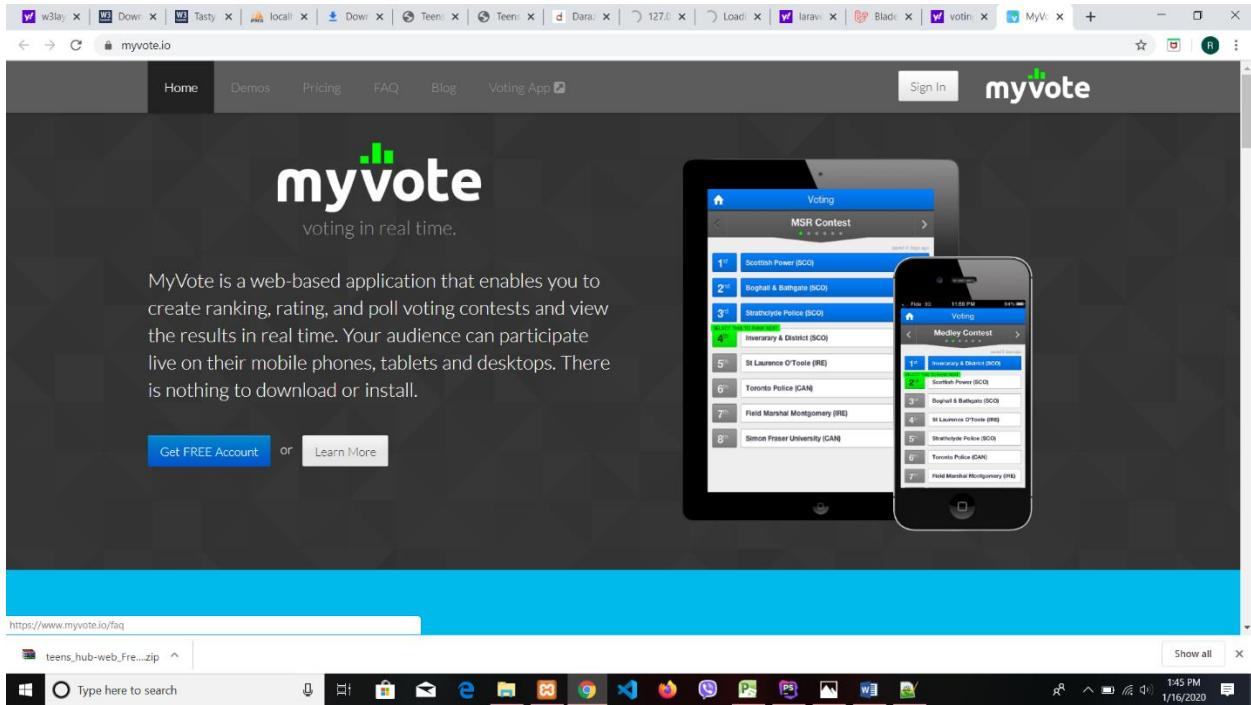


Figure 1: screenshot of similar voting app

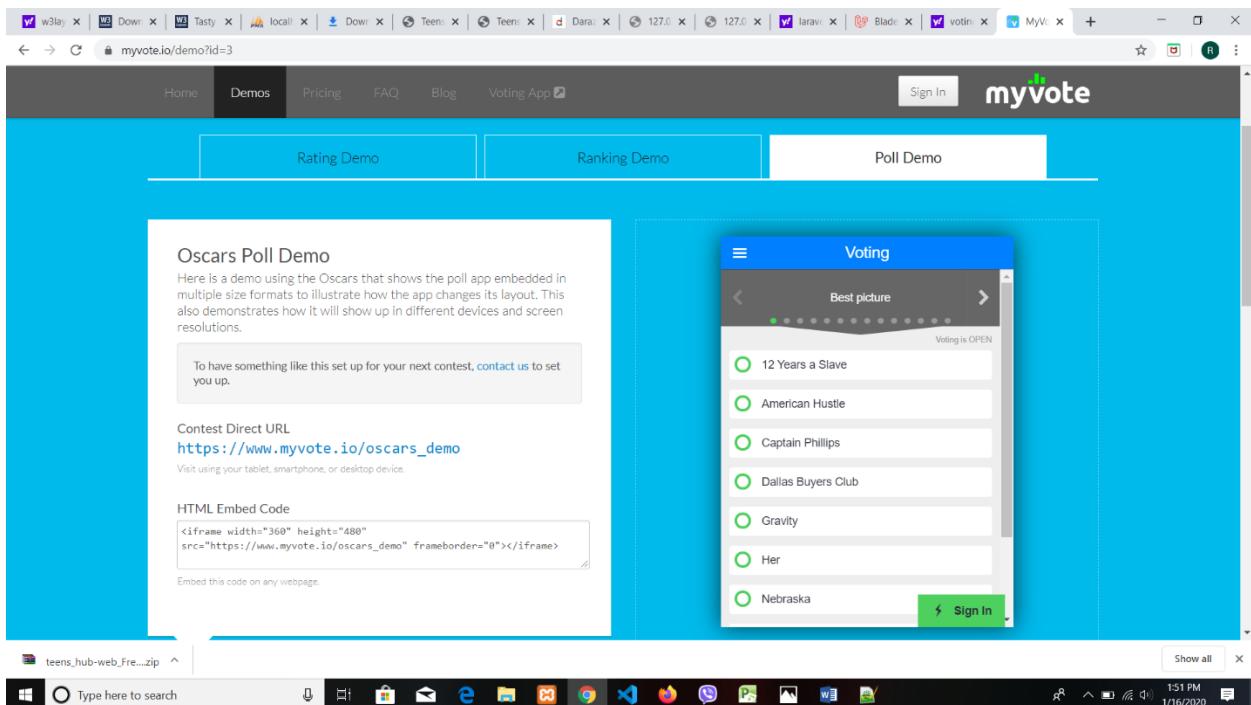


Figure 2: screenshot of polling page demo

Chapter 3: Development

3.1 Considered Mythologies

Software development methodology is a framework that is used to structure, plan, and control the process of developing an information system. This kind of development methodologies are only concerned with the software development process, so it does not involve any technical aspect of, but only concern with proper planning for the software development. There are different project management methodologies such as waterfall model, Agile, Rational Unified Process (RUP) and many more. (TatvaSoft, 2020)

3.1.1 Waterfall Model

The waterfall model is one of the most traditional and commonly used software development methodologies for software development. This model clarifies the software development process in a linear sequential flow that means that any phase in the development process begins only if the earlier phase is completed. This development approach does not define the process to go back to the previous phase to handle changes in requirements.

Advantages of Waterfall Model:

- Waterfall model is very simple and easy to understand and use a method that is why it is beneficial for the beginner or novice developer
- It is easy to manage, because of the rigidity of the model. Moreover, each phase has specific deliverables and individual review process
- In this model phases are processed and completed are at once in a time thus it saves a significant amount of time

- This type of development model works more effectively in the smaller projects where requirements are very well understood.

Disadvantages of Waterfall Model:

- This model can only be used when very precise up-front requirements are available
- This model is not applicable for maintenance type of projects
- The main drawback of this method is that once an application is in the testing stage, it is not possible to go back and edit something
- There is no possibility to produce any working software until it reaches the last stage of the cycle
- This model is good for a small project but not ideally suitable for long and ongoing projects
- Not ideal for the projects where requirements are very moderate, and there is great scope for modification

(TatvaSoft, 2020)

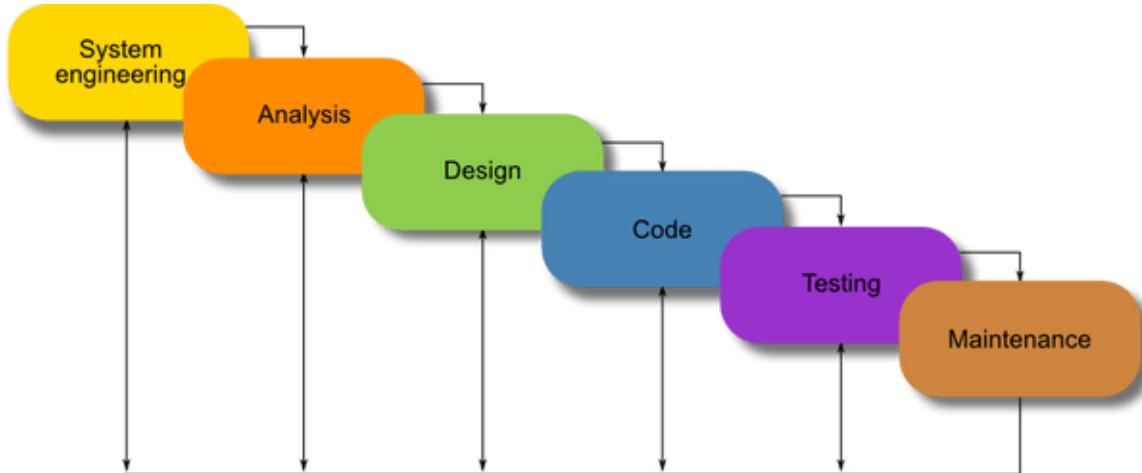


Figure 3: Waterfall model

In waterfall model, if we find an error or need to change something, we must essentially start the project from the beginning. This substantially increases the risk of project failure. But in this project, I may add some features later and it will be difficult to return in development phase when testing is already finished. That's why I didn't chose the waterfall model.

3.1.2 Agile Methodology

Agile Software Development is an approach that is used to design a disciplined software management process which also allows some frequent alteration in the development project. This is a type of software development methodologies which is one conceptual framework for undertaking various software engineering projects. The advantages and disadvantages are discussed below:

Advantages of Agile Development Methodology:

- Agile methodology has an adaptive approach which can respond to the changing requirements of the clients
- Direct communication and constant feedback from customer representative leave no space for any guesswork in the system

Disadvantages of Agile Development Methodology:

- This methodology focuses on working software rather than documentation; hence it may result in a lack of documentation.
- The software development project can get off track if the customer is not very clear about the final outcome of his project.



Figure 4: Agile Methodology

In Agile methodology, it is difficult to accurately determine the amount of time and money that will be needed to complete the project due to constantly changing requirements. But in this project, I need to finish the task and submit in time and I need brief documentation.

3.1.3 RUP Methodology

In RUP methodology, the development process is divided into four distinct phases that each involves business modeling, analysis and design, implementation, testing, and deployment. This is an object-oriented and web-enabled program development methodology. This model also helps software developer for providing them guidelines, templates, and examples for all aspects and stages of software development. As my project is online voting system and based on website, I chose RUP methodology. Furthermore, more details are discussed in (Chapter 3.2 Chosen Methodology).

3.2 Selected Methodology

Rational Unified Process methodology is shortly known as an RUP is a one modern software development process. This methodology divides the development process into four distinct phases that each involves business modeling, analysis and design, implementation, testing, and deployment. This is an object-oriented and web-enabled program development methodology. This model also helps software developer for providing them guidelines, templates, and examples for all aspects and stages of software development.

Advantages of RUP Software Development Methodology:

- This methodology emphasizes on accurate documentation
- It is proactively able to resolve the project risks that are associated with the clients evolving requirements for careful changes and request management
- Very less need for integration as the process of integration goes on throughout the development process

(TatvaSoft, 2020)

3.3 Phases of Methodology

Based on UML, RUP organizes the development of software into four phases, each consisting of one or more executable iterations of the software at that stage of development. The four phases of RUP are as follows:

Inception

In this stage, the projects business case is stated, and the team decides if the project is worth doing or if it is even possible. It is important to the process to first formulate the scope of the project and determine what resources will be needed.

Elaboration

In this stage, the developers take a closer look at the project to determine its architecture foundation and to evaluate the architecture in relation to the project. This stage is important to the RUP because it is here that developers analyze the risks associated with changing the scope of the project or adding new technologies along the way.

Construction

In this stage, the development of the project is completed. The application design is finished, and the source code is written. It is in this stage that the software is tested to determine if the project has met its goal laid out in the inception phase.

Transition

In this stage, any fine-tuning is performed. Any final adjustments can be based on user feedback, usability or installation issues.

(Webopedia, 2020)

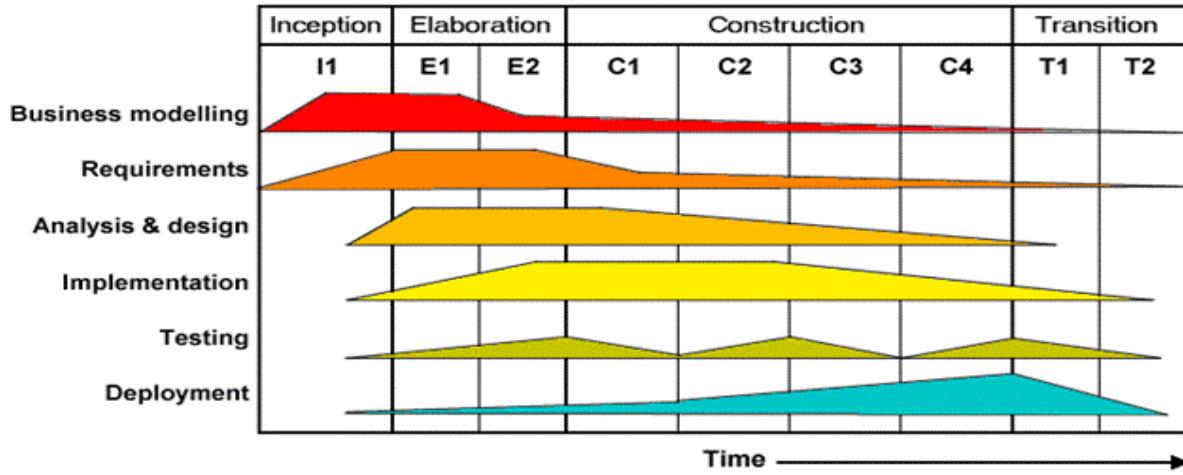


Figure 5: Rational Unified Process (RUP) Methodology

Description of project with RUP methodology in different phases.

Phases	Description
Inception	The topic was decided. The scope of this project (online voting system) was determine as main scope of this project is to eliminate manipulation of the votes and to make voting more accurate.
Elaboration	Analysis of the project and detail process or working method of the online voting system was done. Work process is separated based on timeline and Gannt chart is created which help in making strategies to complete the project in time.
Construction	In this phase, use case and ERDs is created, designing and coding of the website is done. Wireframes are design as per to meet the requirements of this project. With the help of wireframes, the website is developed by coding. Back end codes are also developed in this phase such as connecting database.
Transition	In this phase, the testing process is done. For examples, by clicking the result page, will it show results or not? Will the login account of the voters will store in database or not? Documentation of the project will be completed.

Table 1: Description of project with RUP methodology in different phases

3.4 Survey Results

3.4.1 Pre-Survey Results

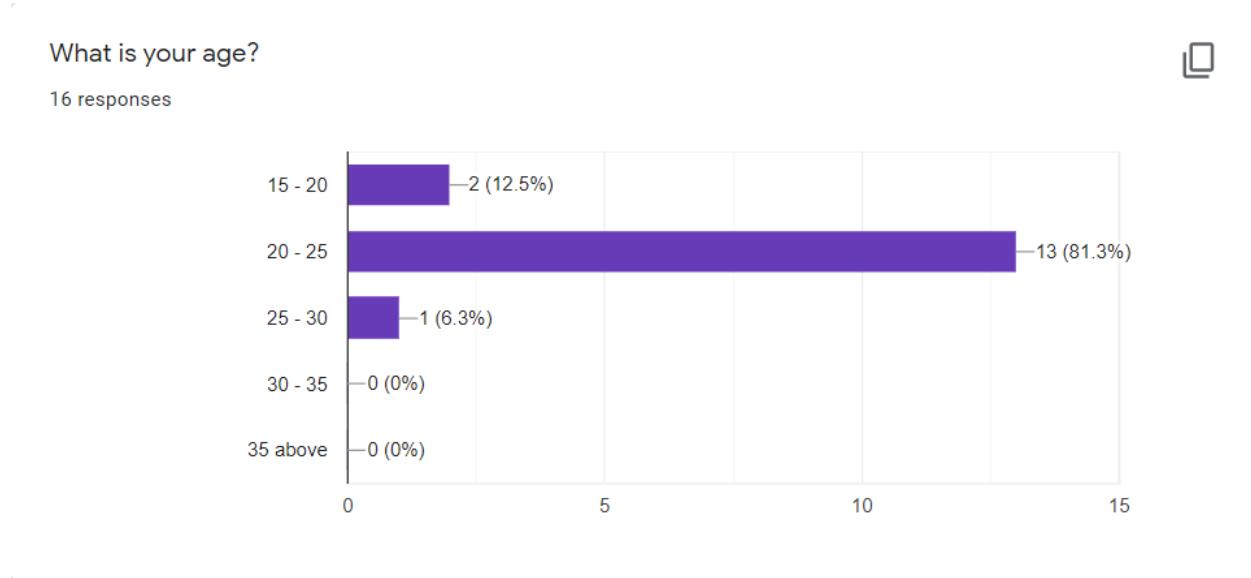


Figure 6: pre survey result 1

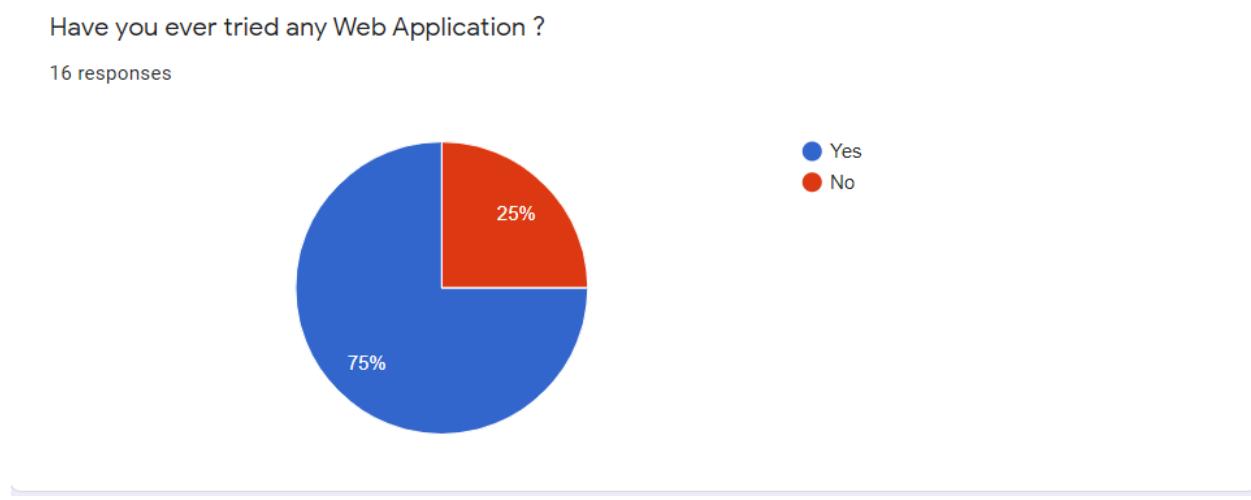


Figure 7: pre survey result 2

Do you have knowledge about online voting system?

16 responses

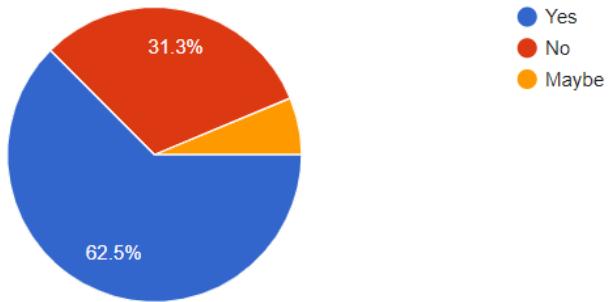


Figure 8: pre survey result 3

Would you prefer online voting system or manual (open) voting system?

16 responses

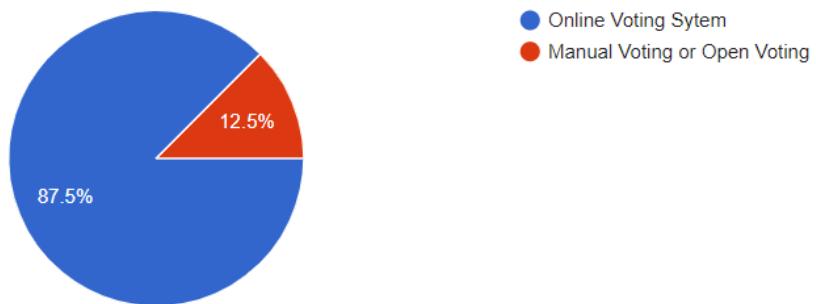


Figure 9: pre survey result 4

Do you trust online system (web application)?

16 responses

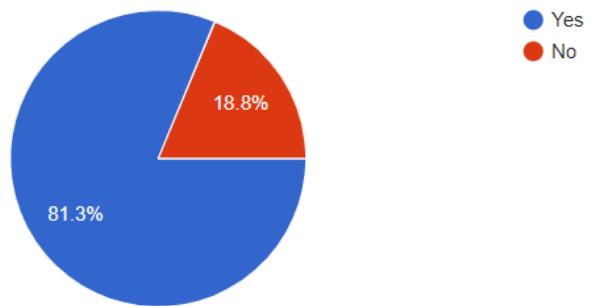


Figure 10: pre survey result 5

If online voting system launch in Nepal, Would you find easy or difficult?

16 responses

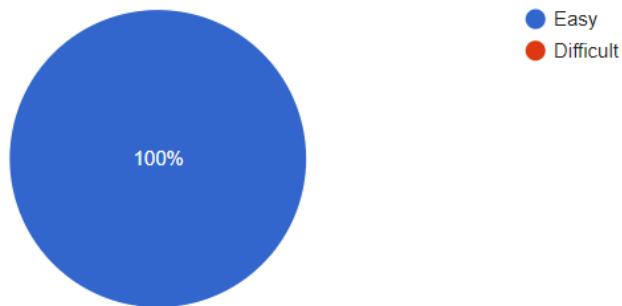


Figure 11: pre survey result 6

Feedback/Suggestions

6 responses

Nothing....

Good !!

Good Job !

No idea about online voting system but may not be trustable at all

Everything everywhere is upgraded so we must be upgraded as well.

It should be heavily secured so no one could hack the system.

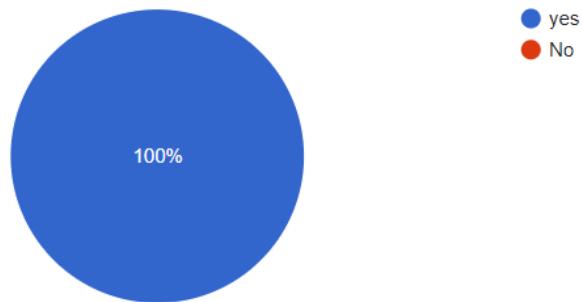
Figure 12: pre survey result 7

3.4.2 Post Survey Results

Do you like the overall features of online voting system?



5 responses

*Figure 13: post survey results 1*

Do you think design is user friendly?

5 responses

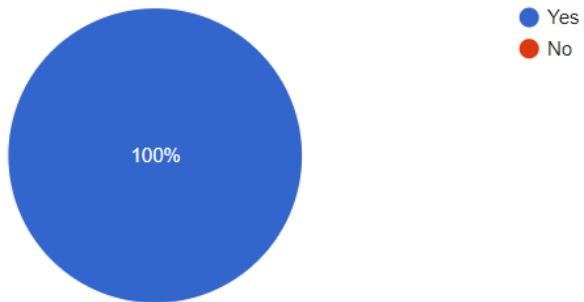


Figure 14: post survey result 2

Do you found the voting process easy or difficult?

5 responses

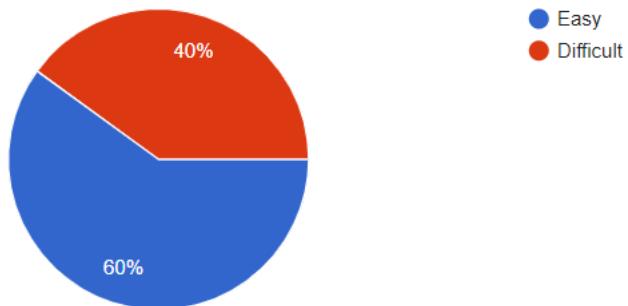


Figure 15: post survey result 3

How would you rate the prototype?

5 responses

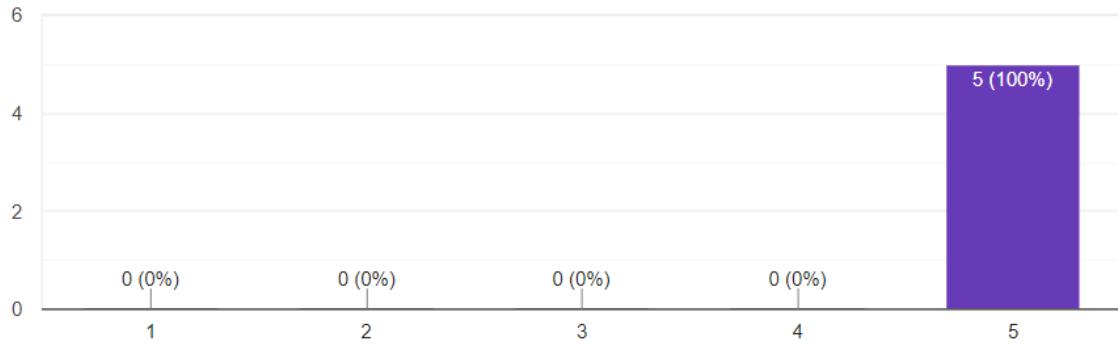


Figure 16: post survey result 4

Would you go on manual voting or online voting?

5 responses

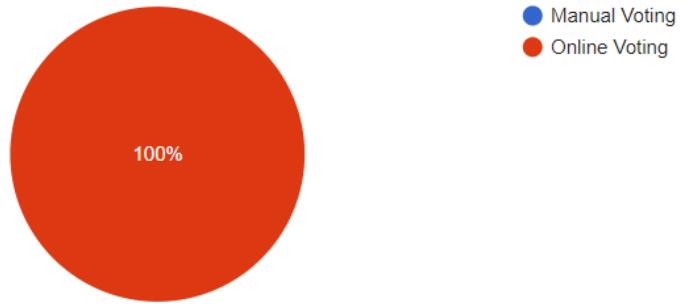


Figure 17: post survey result 5

Feedback?

3 responses

problem in register

Register process was difficult

yeah man...keep going

Figure 18: post survey result 6

3.5 Requirement Analysis

Requirements Analysis is the process of defining the expectations of the users for an application that is to be built or modified. It involves all the tasks that are conducted to identify the needs of different stakeholders. Therefore, requirements analysis means to analyze, document, validate and manage software or system requirements. High-quality requirements are documented, actionable, measurable, testable, traceable, helps to identify business opportunities, and are defined to facilitate system design.

Requirements Analysis Technique:

- UML (Unified Modeling Language)
- Flowchart Technique
- Data Flow Diagram
- Activity Diagram
- Gantt Chart

(Requirements Analysis , 2018)

Software Requirement

The required software to develop the online voting system are described below:

PHP

PHP is known as the general-purpose programming language. It is used as a server-side scripting language that is mainly used for the development of web sites. The PHP frameworks also make web development easier. This framework helps in reusing the same code and there is no need to write the lengthy and complex code for the web applications. PHP frameworks are mainly open source and can be used easily. The advantage of using php are as follow:

- PHP is open source and free of cost, which helps us to install it quickly and readily available for use.
- PHP is mainly supported by all the operating systems like Windows, Unix, Linux etc. The PHP based developed web applications can be easily run on any platform.
- It helps in saving a lot of effort and cost.
- PHP is easily connected with the database and make the connection securely with databases. It has a built-in module that is used to connect to the database easily.
- PHP based web applications can be easily tested. PHP unit uses to perform the unit testing quickly and easily. It also helps the programmers to write test cases and perform the testing smoothly.

(EDUCBA, 2019)

Laravel

Laravel is an open-source PHP framework, which is robust and easy to understand. It follows a model-view-controller design pattern. Laravel reuses the existing components of different frameworks which helps in creating a web application. The web application thus designed is more structured and pragmatic.

The advantages of using Laravel are:

- Laravel has a pre-installed powerful and lightweight template engine, helping developers in making some extraordinary layouts with intensive content seeding. Laravel template offers various solid widgets with robust CSS and JS coding.
- Laravel is the framework that facilitates you at best by being the only provider having dynamic pre-installed libraries. These libraries include Object Oriented libraries that cannot be found in other PHP frameworks.
- Laravel also assist in generating URLs which becomes very helpful for building links in your templates. All Laravel corridors are effectively laden by the framework which is delineated in the app/Http/routes.php file.
- Generally, a developer needs to interact with the Laravel framework using a command line that develops and manages the Laravel project environment. Laravel provides an integrated command line tool called Artisan. This tool helps to create skeleton code and database architecture as well as their migrations. Database management becomes easier as a result.

(Khalid, 2019)

HTML

Hypertext markup language (HTML) is the major markup language used to display Web pages on the Internet. In other words, Web pages are composed of HTML, which is used to display text, images or other resources through a Web browser. (Technopedia, 2020)

CSS

CSS is the language for describing the presentation of Web pages, including colors, layout, and fonts. It allows one to adapt the presentation to different types of devices, such as large screens, small screens, or printers. CSS is independent of HTML and can be used with any XML-based markup language. The separation of HTML from CSS makes it easier to maintain sites, share style sheets across pages, and tailor pages to different environments. (W3C, 2020)

JavaScript

JavaScript (JS) is a scripting language, primarily used on the Web. It is used to enhance HTML pages and is commonly found embedded in HTML code. JavaScript is an interpreted language. Thus, it doesn't need to be compiled. JavaScript renders web pages in an interactive and dynamic fashion. This allowing the pages to react to events, exhibit special effects, accept variable text, validate data, create cookies, detect a user's browser, etc. (Technopedia, 2020)

XAMPP

XAMPP is a free and open-source cross-platform web server solution stack package developed by Apache Friends, consisting mainly of the Apache HTTP Server, MariaDB database, and interpreters for scripts written in the PHP and Perl programming languages. Since most actual web

server deployments use the same components as XAMPP, it makes transitioning from a local test server to a live server possible. (XAMPP, 2020)



Figure 19: XAMPP

Balsamiq

Balsamiq is a graphical user interface website wireframe builder application. It allows the designer to arrange pre-built widgets using a drag-and-drop WYSIWYG editor.



Figure 20: Balsamiq logo

3.6 Design

3.6.1 Use case of Online Voting system

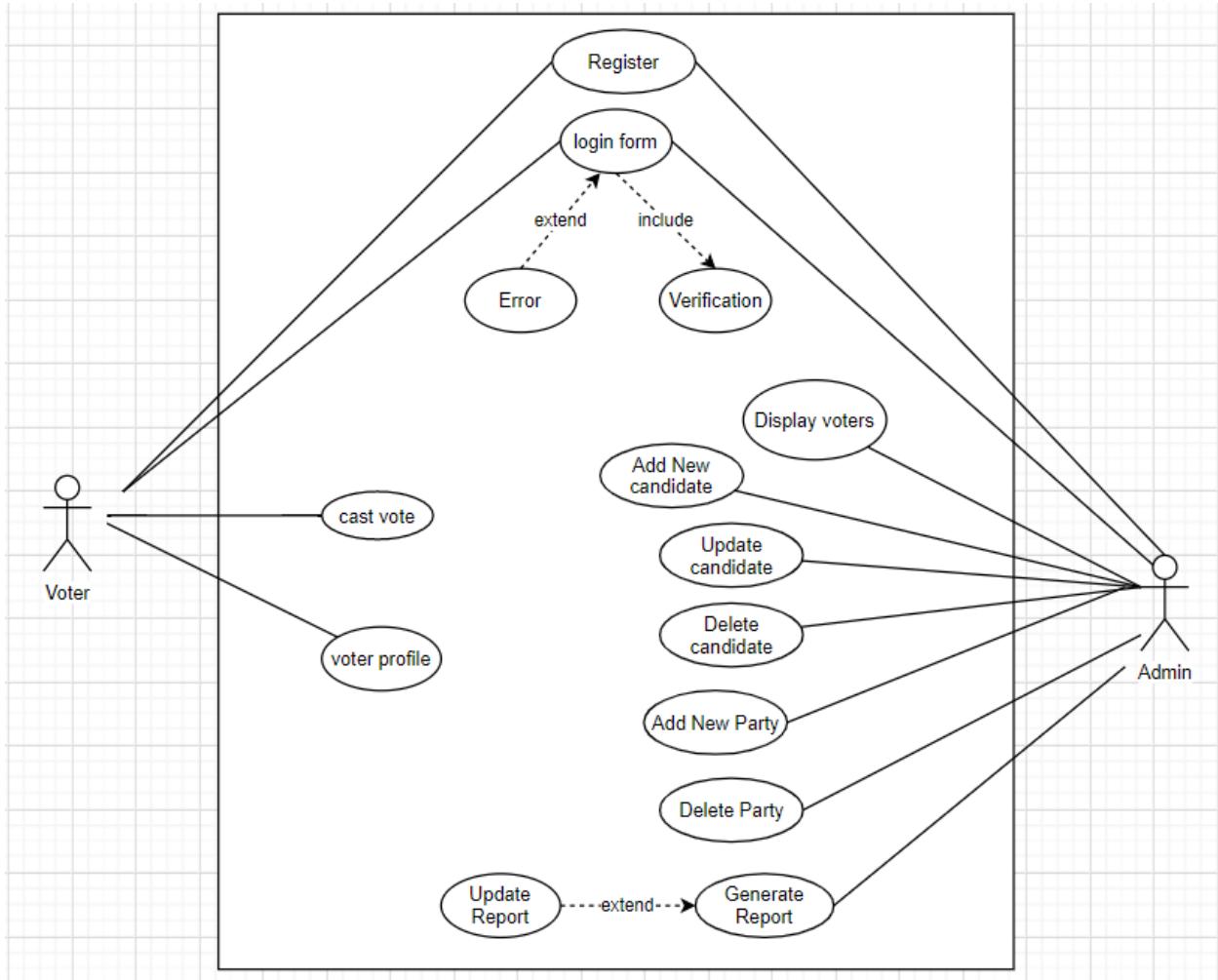


Figure 21: Use case of online voting system

Above picture is of use case of online voting system. Here the function of voter and admin is shown in the figure.

3.6.2 High Level Use Case

High Level Use Case for Register

Use case: Register

Actor: Voter, admin

Description: A new voter provides the details of his/ her information to the system.

High Level Use Case for Login

Use case: Login

Actor: Voter, admin

Description: After registration voter, login to the system.

High Level Use Case for Cast vote

Use case: Cast vote

Actor: Voter

Description: After login, voter cast a vote to different party and candidate.

High Level Use Case for Generate Result

Use case: Display Result

Actor: Admin

Description: To display the result after casting a vote.

High Level Use Case for Add Party

Use case: Add Party

Actor: Admin

Description: To add new party to the system as new party may be formed and take part in election.

High Level Use Case for Add Candidate

Use case: Add Candidate

Actor: Admin

Description: To add new candidate as many new candidates take part in election.

High Level Use Case for Delete Party

Use case: Delete Party

Actor: Admin

Description: To delete party that are no longer part of election as parties may separate into pieces.

High Level Use Case for Delete Candidate

Use case: Delete Candidate

Actor: Admin

Description: To delete candidate that are no longer part of election as many candidates left election.

3.6.3 Expanded Use Case Description

Expanded Use Case Description for Register

Name: Register

Actor: Voter

Purpose: A new member provides the details of his/ her information to the system in order to cast a vote.

Action Steps

Actor action	System Response
1. Voters will visit the website entering suitable URL.	
	2. System will display the website and display the login form that include username and password and in case voter doesn't have account, voter should register
3. Voters will click on register button.	
	4. System will display the registration form that include details of voters.
5. Voters will fill the form and click on submit button.	
	6. System will show the successful message.

Table 2: Expanded Use case of Register

Expanded Use Case Description for Login

Name: Login

Actor: Voter

Purpose: When user opens the browser then enters the respective URL of website then user login to the website by entering suitable username and password.

Action Steps

Actor action	System Response
1. User will visit the website entering suitable URL.	
	2. System will display the website and display the login form that include username and password.
3. User will input the username and password.	
	4. Find username and verify
	5. System will display the polling page.

Table 3: Expanded use case of login

Expanded Use Case Description for Cast Vote

Name: Cast Vote

Actor: Voter

Purpose: When user login into system, system should redirect into voting page and user should vote peacefully.

Action Steps

Actor action	System Response
1. User will visit the website entering suitable URL.	
	2. System will display the website and display the login form that include username and password.
3. User will input the username and password.	
	4. Find username and verify
	5. System will redirect to voter profile.
6. User will vote the party	
	7. System will display the navigation link to vote candidate or user simply can vote by clicking the navigation link of header.
8. User will vote to candidates.	
	9. System will display successful message.

Table 4: Expanded use case of cast vote

3.6.4 ERD of online voting system

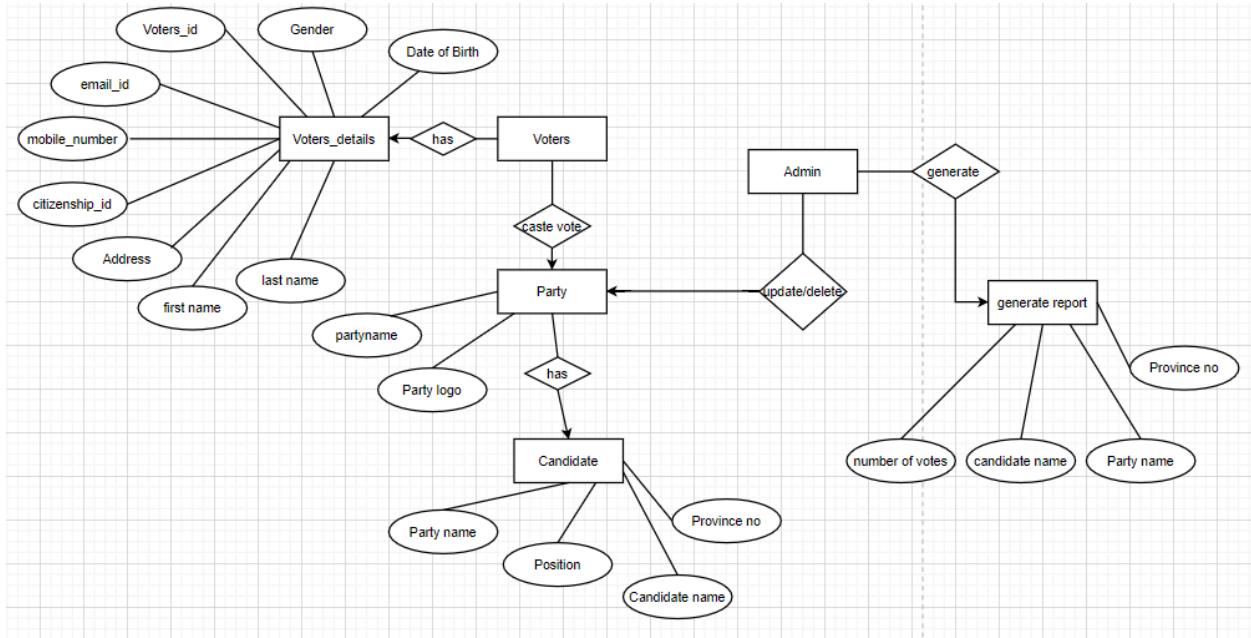


Figure 22: Conceptual ERD of online voting system

An entity relationship model, also called an entity-relationship (ER) diagram, is a graphical representation of entities and their relationships to each other, typically used in computing in regard to the organization of data within databases or information systems. An entity is a piece of data—an object or concept about which data is stored. (Beal, 2020)

The above diagram is the conceptual diagram of online voting system and the following diagram is the Entity Relational Diagram of online voting system. Here, there are 6 entities, they are Voters, Candidate, Party, Cast Vote, count vote and Generate Report. In Voters entity, the details of voters are recorded. In party entity, the detail of parties is displayed. In candidate entity, the details of candidate is displayed. In cast vote entity, the information of parties, voters and candidates are recorded to count the vote. In count vote entity, the information of parties, voters and candidates are recorded and count the vote. In Generate report entity, the result is displayed with number of votes alongside party and candidate.

The relation is also shown in the figure. As shown in figure, the relation between voters and candidate is many to one as many voters vote to a candidate. Similarly, relation between voters and party is many to one as many voters vote to a party. The relation between voters and cast vote entity is also many to one as numbers of voters are recorded. Similarly, relational between candidate and party with cast vote is also many to one.

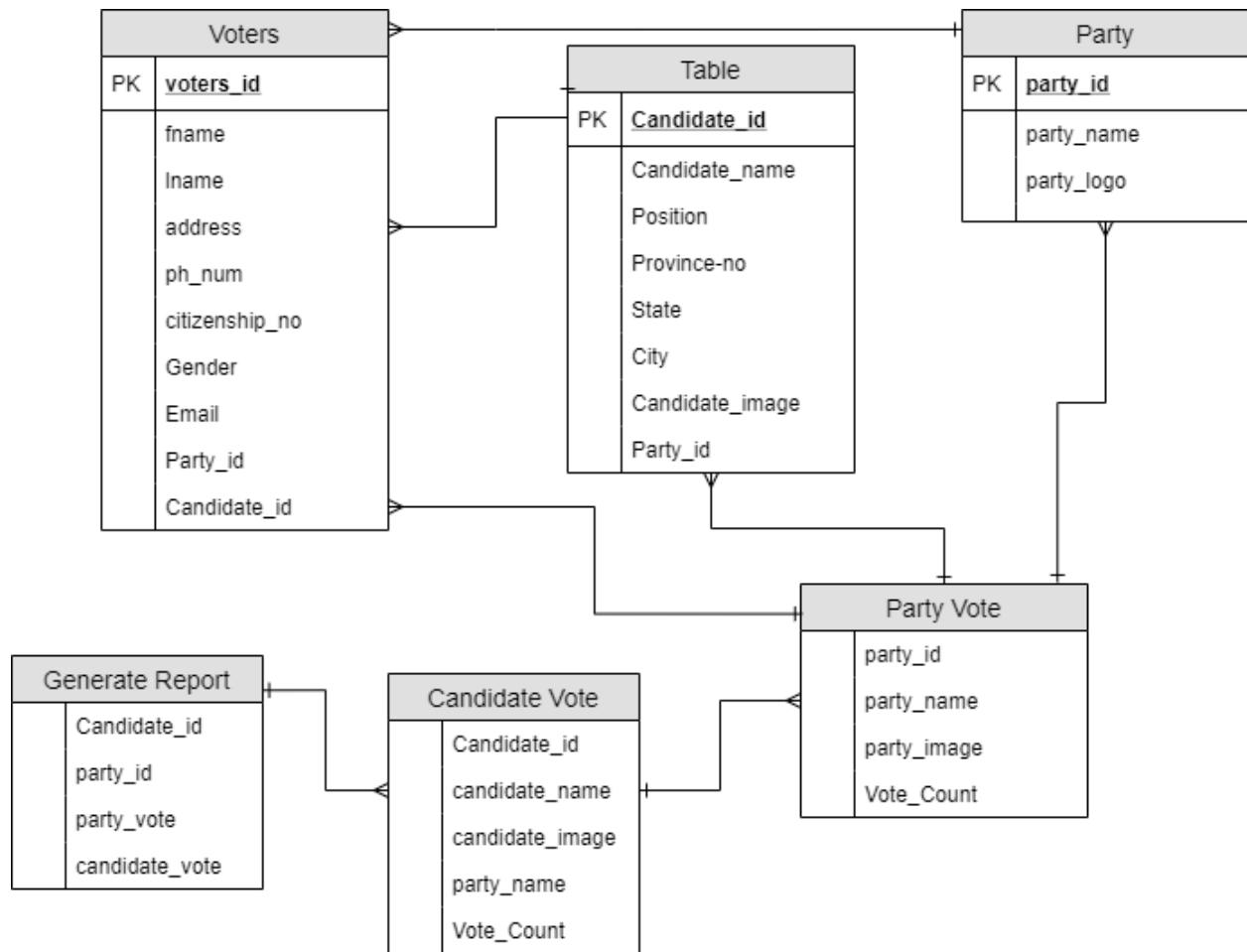


Figure 23: ERD of online voting system

3.6.5 Activity Diagram

An activity diagram is a type of flow chart. We use activity diagram to visually model the dynamic behavior of a given part of our system. Activity diagrams can be used to model the dynamic behavior of a number of elements of an object-oriented system. One element that activity diagrams can be very effective in modelling is the behavior of an operation on a class. The use of activity diagram is to model use case. (Windle & Abreo, 2003)

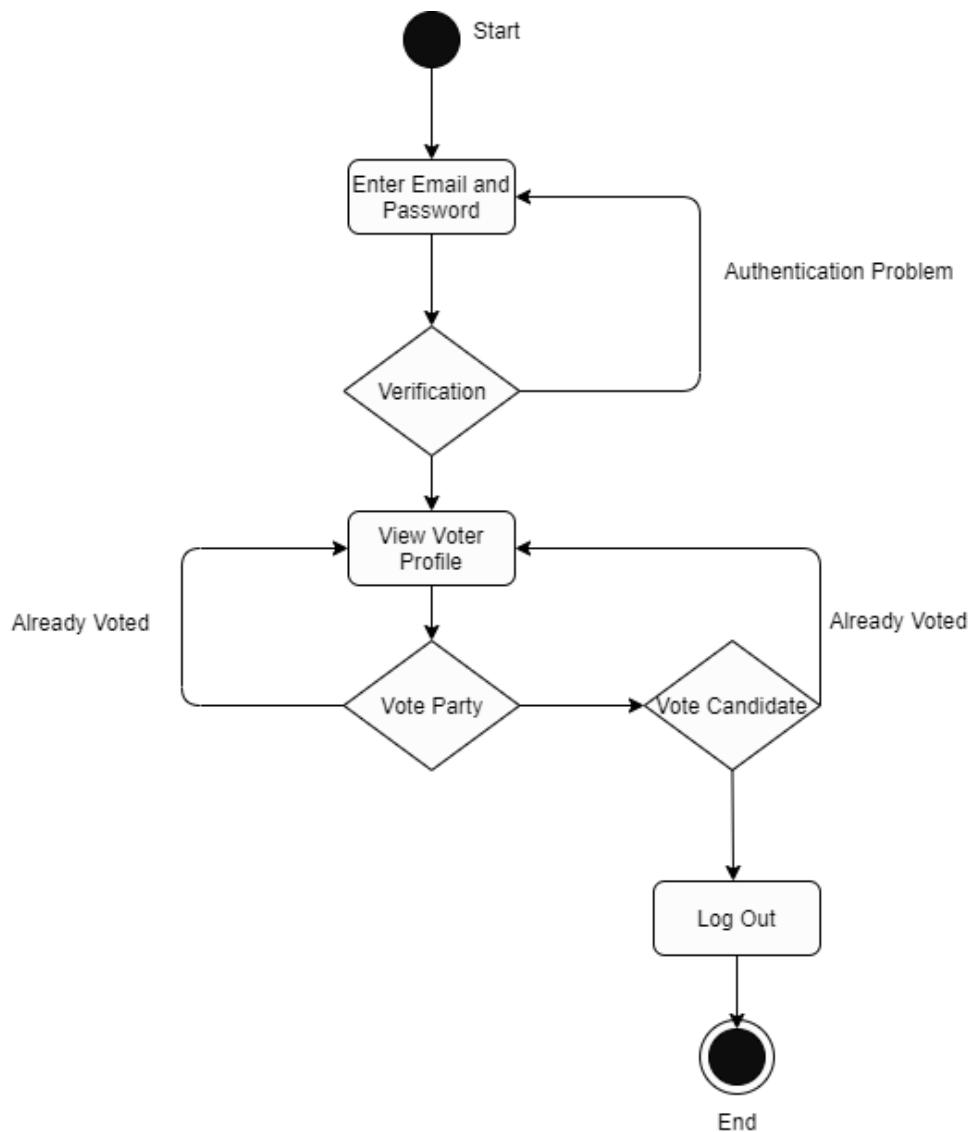


Figure 24: Activity Diagram

3.6.6 Data Flow Diagram (DFD)

A data flow diagram (DFD) maps out the flow of information for any process or system. It uses defined symbols like rectangles, circles and arrows, plus short text labels, to show data inputs, outputs, storage points and the routes between each destination. Data flowcharts can range from simple, even hand-drawn process overviews, to in-depth, multi-level DFDs that dig progressively deeper into how the data is handled. They can be used to analyze an existing system. (LucidChart, 2020)

0-Level DFD

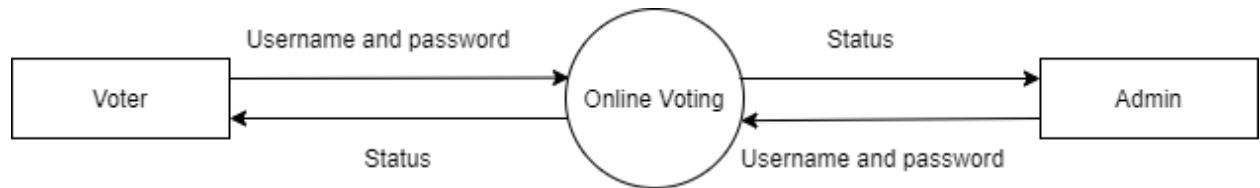
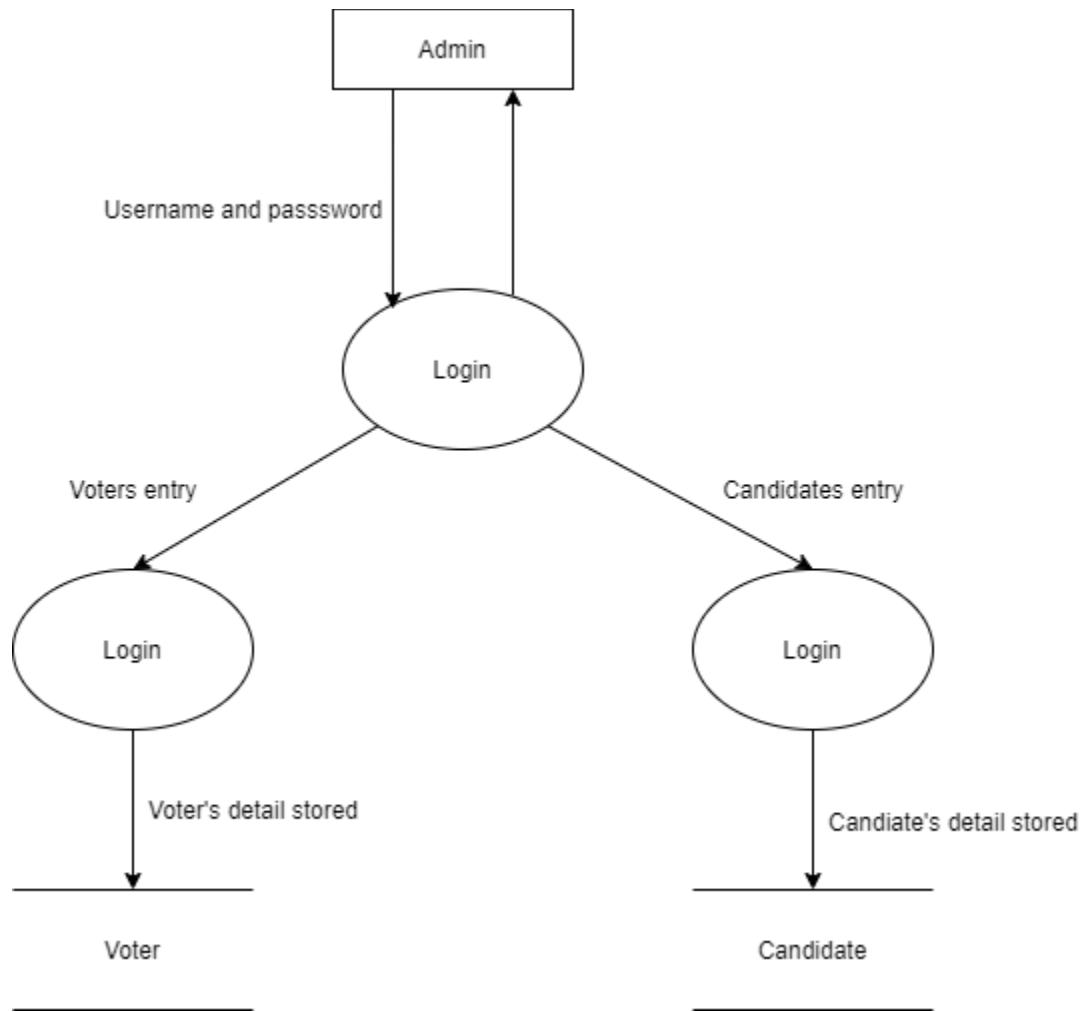


Figure 25: 0-level dfd of online voting system

1-level DFD for admin*Figure 26: 1-level dfd for admin*

1-level DFD for users

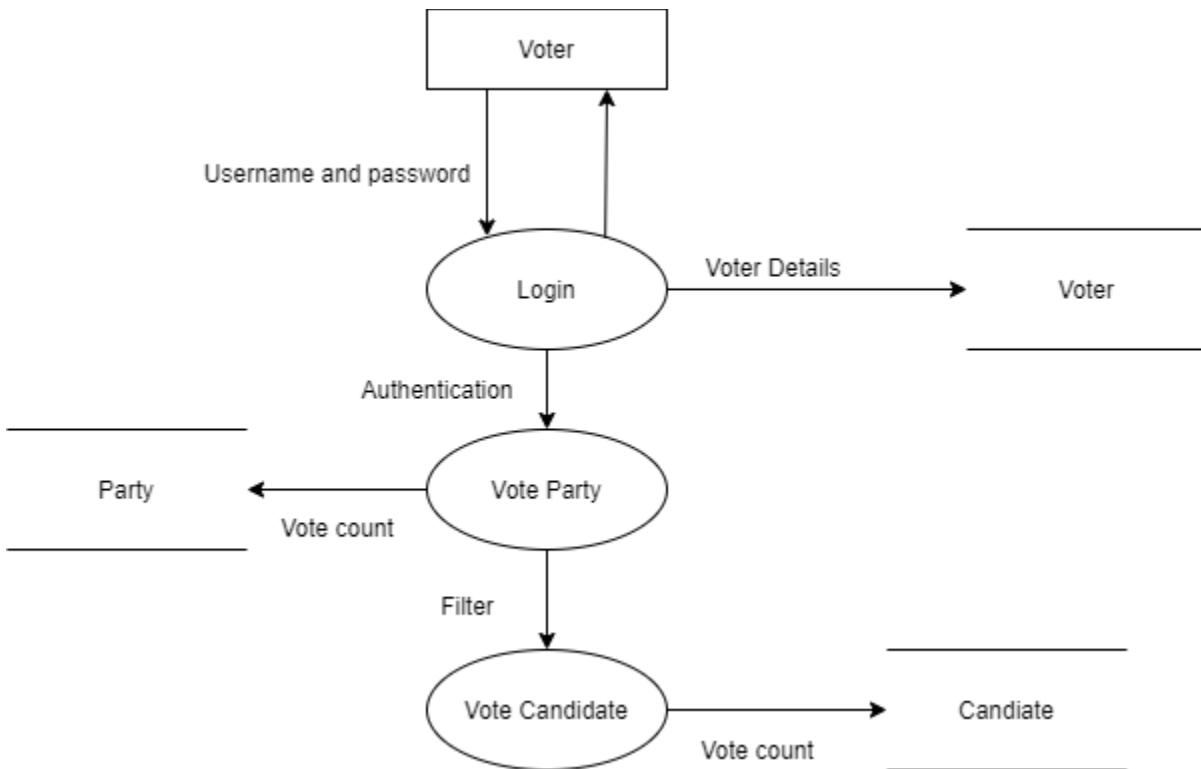


Figure 27: 1-level dfd for users

3.6.7 Collaboration diagram

A collaboration diagram is an interaction diagram that emphasizes the organization of and relationships among objects, rather than the sequential progression of the message passed among them, as sequence diagram do. The primary elements of the collaboration diagram are objects, links and messages. When creating a collaboration diagram, it is useful to begin by modelling the object that are going to participate in the interaction. With the objects established, we can then model the relationships among them and finally the messages that they send among themselves to complete the intersection.

(Far, 2005)

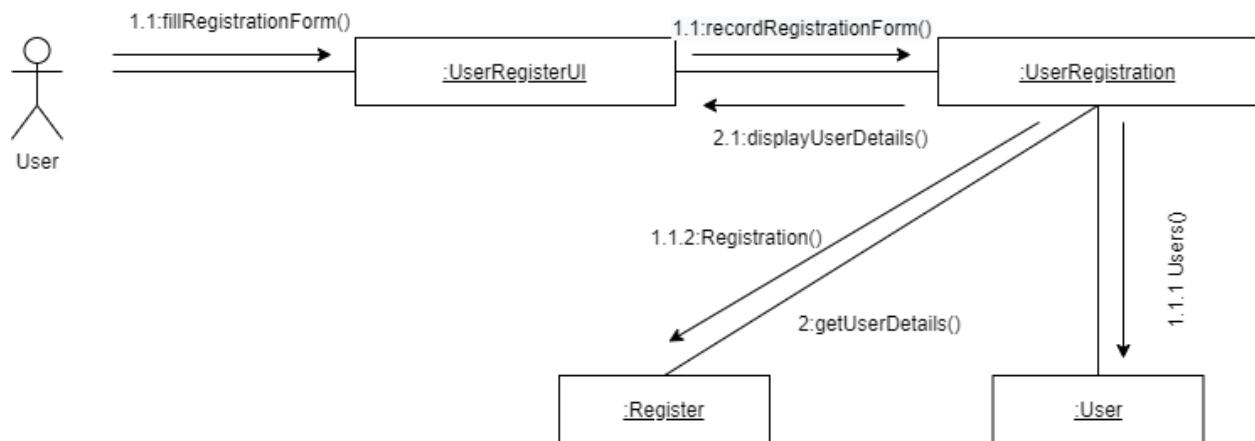


Figure 28: Collaboration diagram

3.6.8 Sequence Diagram

A sequence diagram is a collection of objects interacting to accomplish a given task or series of tasks over time. Objects appear at the top of the diagram. A dashed line extends from the object to the bottom of the sequence diagram. Time is represented on the vertical axis. Methods that appear higher on the diagram occur earlier than methods that appear lower on the diagram. The thick bar in the sequence diagram indicates the focus of control. Focus of control implies that one method called another control will return to the first method. The arrowhead points to the method that is to be called. Information can flow both ways. (Windle & Abreo, 2003)

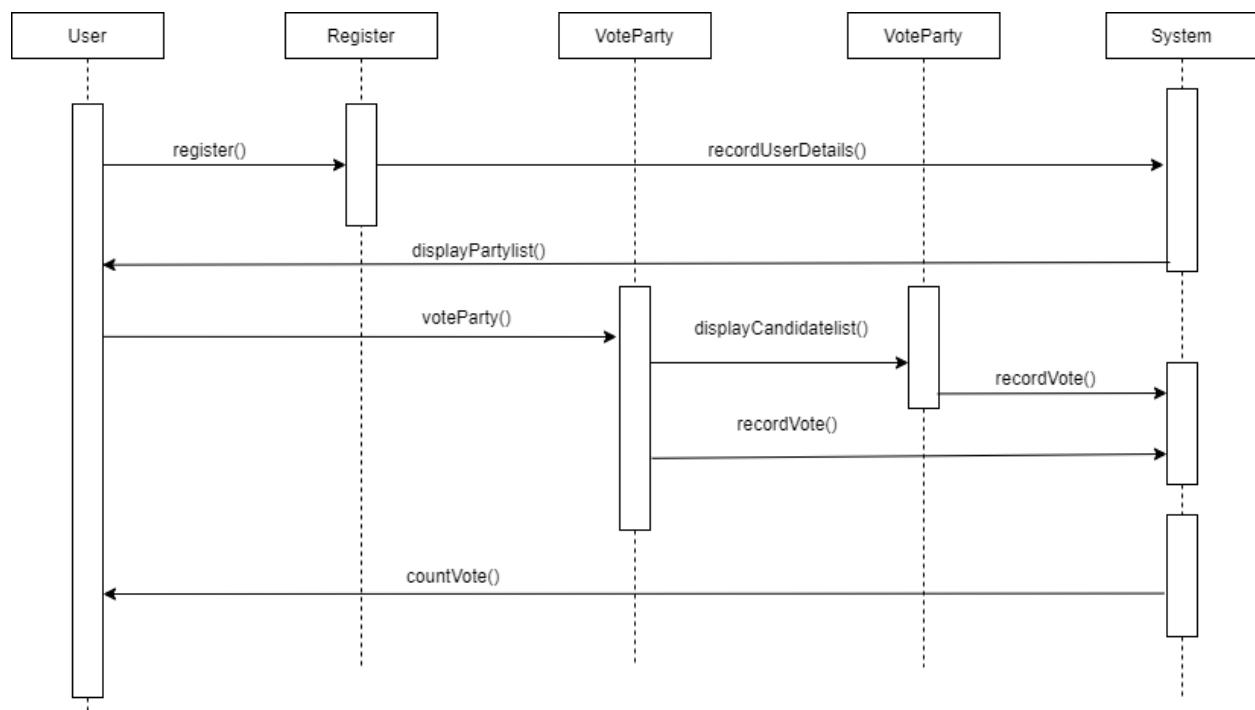


Figure 29: Sequence Diagram

3.6.9 Wireframes

Homepage

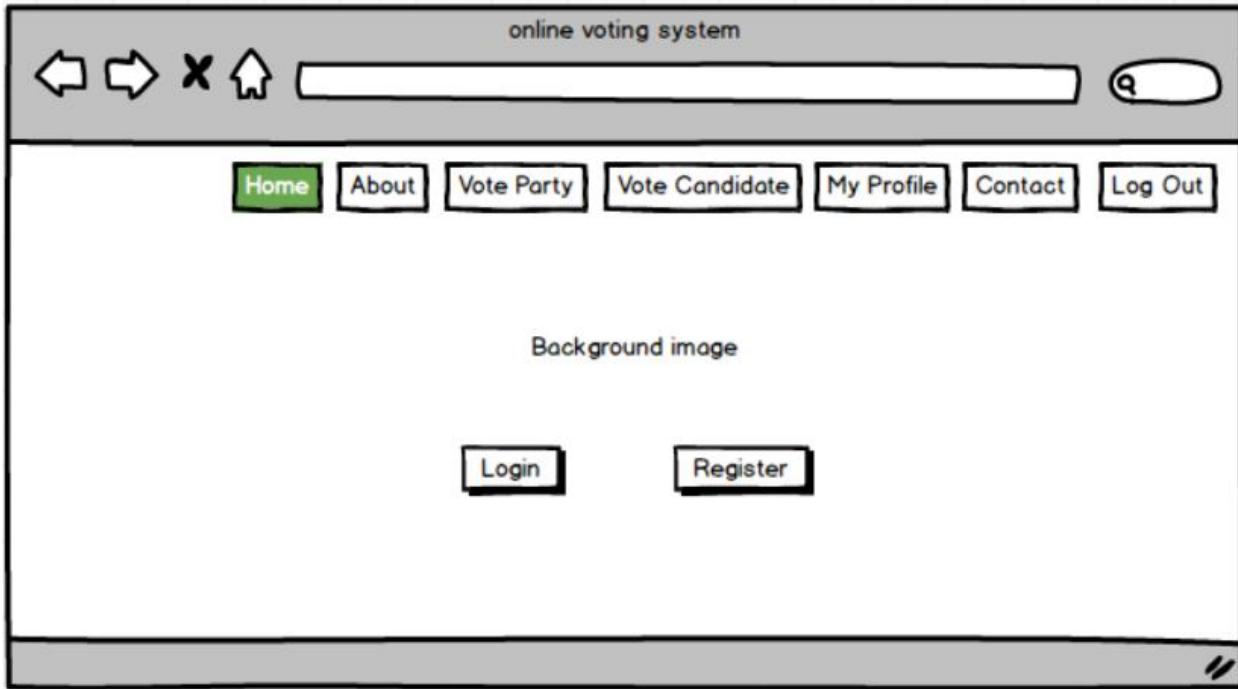


Figure 30: Home page of online voting system

Here, the above picture is the screenshot of front page of online voting system. It has the login button for voters and admin separately. There is register button also for those who doesn't have an account. They cannot register for admin.

About page

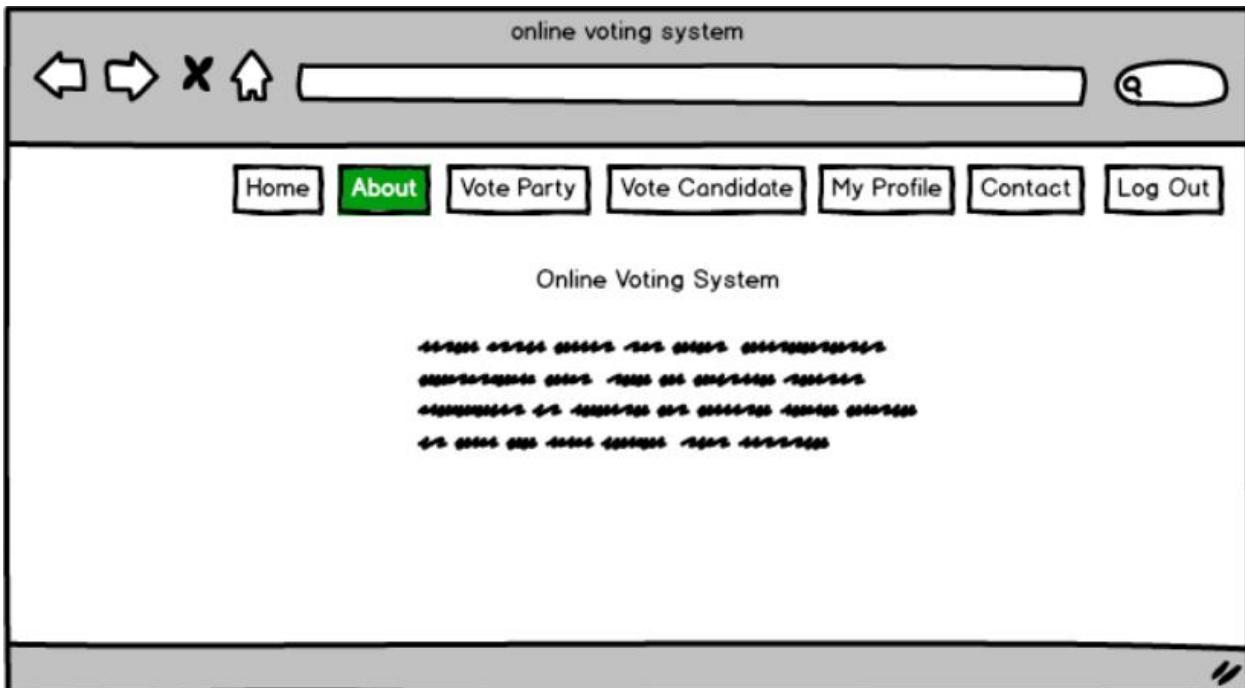


Figure 31: Wireframe of About page

Here, the above picture is of about page of online voting system. In this page, the online voting system is described.

Login Page

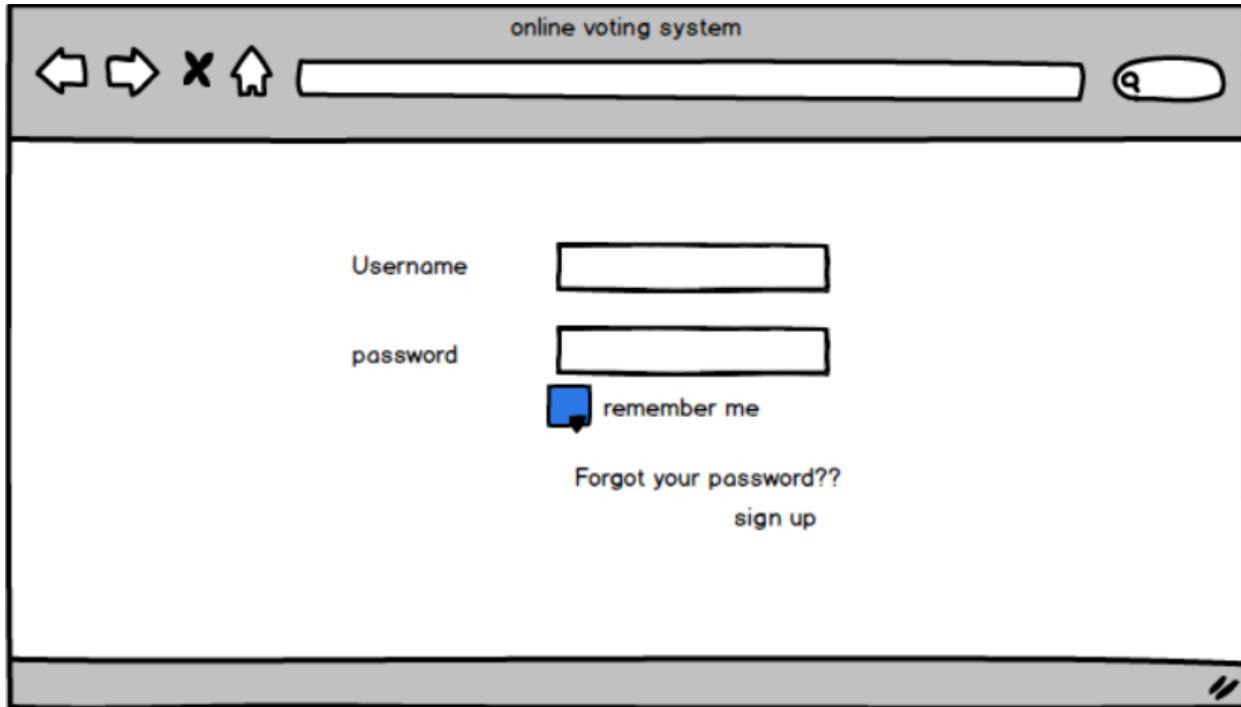


Figure 32: wireframe of Login page

Here, the above figure is the screenshot of login page for voters. This page content username and password to login. After clicking login button, voters successfully enter in polling page. There is also sign up button for those who doesn't have an account.

Register Page

The wireframe shows a web browser window titled "online voting system". The header includes standard navigation icons (back, forward, stop, home) and a search bar. The main content area contains a form for user registration. The fields are labeled on the left and include: first name, last name, email, password, confirm password, province, district, area, phone number, citizenship number, gender, and image. A large "Register" button is positioned at the bottom of the form.

Figure 33: wireframe of register page

Here, the above figure is the screenshot of wireframe of register page of online voting system. User have to fill the details form in order to register. Same citizenship number is not valid while register process. User should fill every details in order to register.

Voting Page of Party

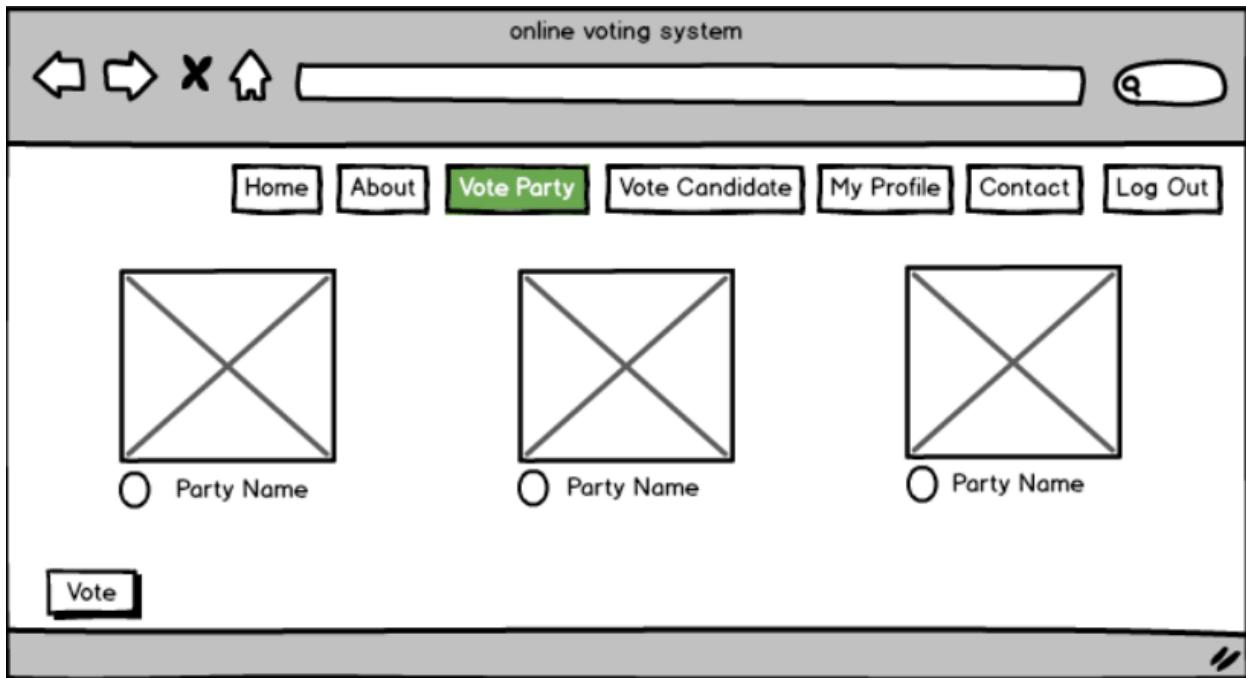


Figure 34: wireframe of voting page of party

Here, the above figure is of polling page for party. After clicking login button, voters successfully enter in polling page. Here the party logo and party page are displayed. Voter can vote by clicking the vote button that are presented under party logo.

Vote Candidate Page

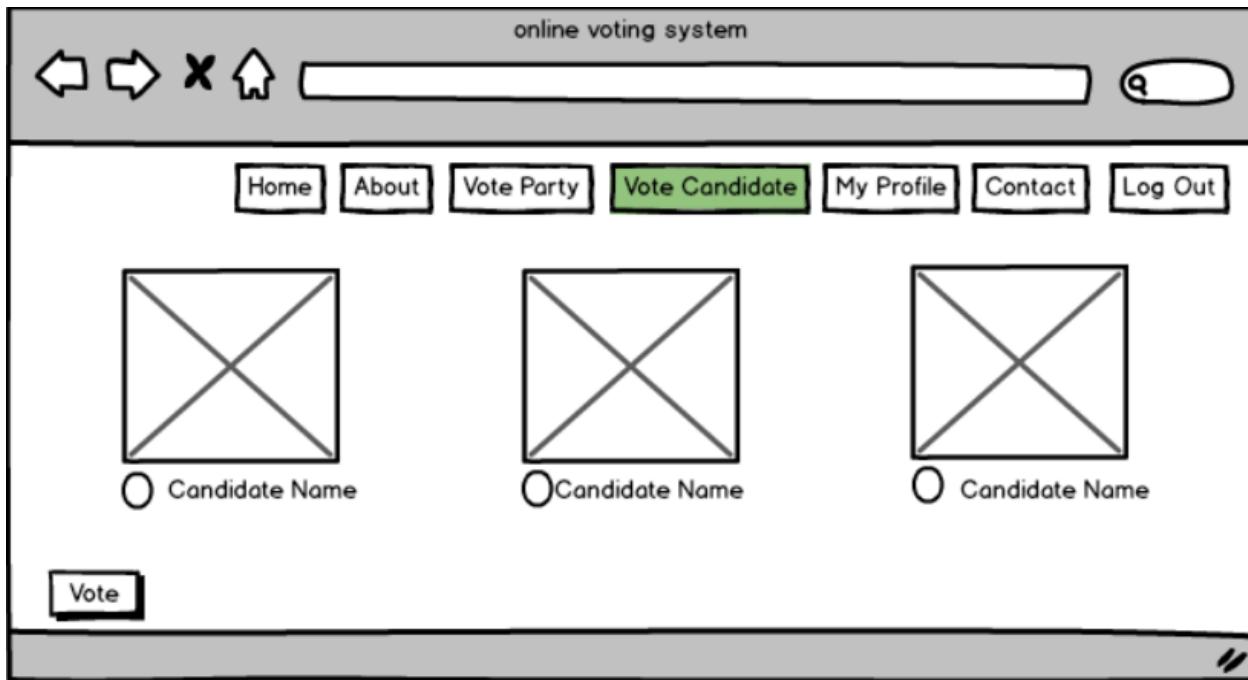


Figure 35: voting page of candidate

Here, the above figure is of polling page for candidate. After voting for party, voters successfully enter in polling page for candidate. Here the candidate photo and name are displayed. Voter can vote by clicking the vote button that are presented under candidate's picture.

My Profile Page

The wireframe shows a top navigation bar with icons for back, forward, search, and refresh, followed by the text "online voting system". Below this is a horizontal menu bar with buttons for Home, About, Vote Party, Vote Candidate, My Profile (which is highlighted in green), Contact, and Log Out. To the left, there is a placeholder image of a user profile picture with a large 'X' over it, labeled "User name" with a "Change" button below it. To the right, there is a form for updating profile details, including fields for Name, Email, phone, Address, and password, each with its own input box. A "Update" button is located at the bottom right of the form area.

Figure 36: wireframe of my profile page

Here, the above picture is of my profile page of online voting system. User can view their own profile. User can edit their details but cannot edit citizenship number.

Contact Page

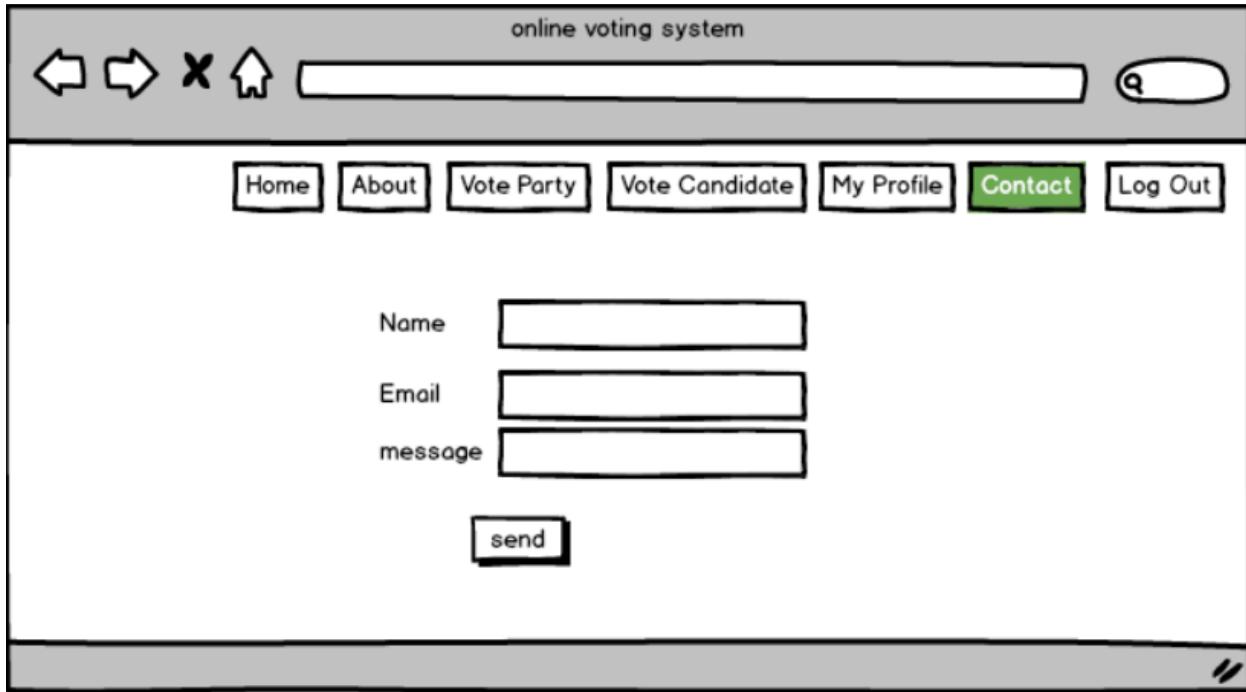


Figure 37: Wireframe of contact page

Here, the above picture is of contact page of online voting system. User can contact the system or admin as the email address and phone number is given in this page. User can ask the admin about the process or system or any problem.

3.6.10 Work Break Down Structure

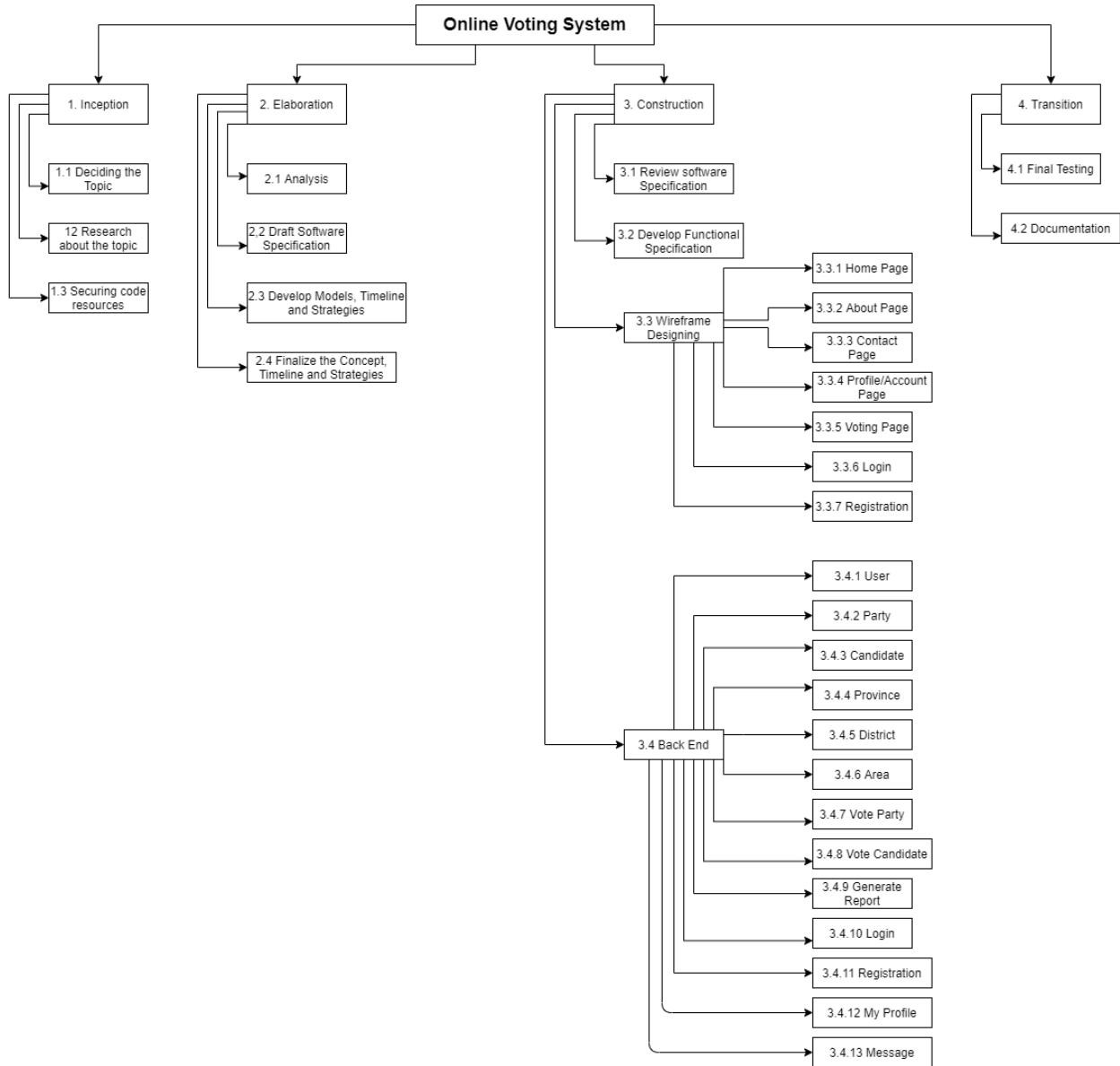


Figure 38: Work Break Down Structure

3.6.11 Gantt Chart

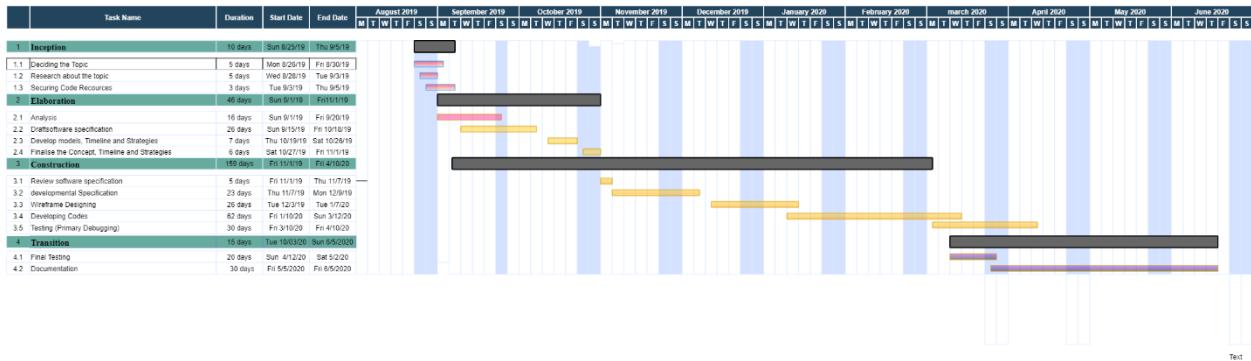


Figure 39: Gantt Chart

Gantt Chart is a chart in which a series of horizontal lines shows the amount of work done or production completed in certain periods of time in relation to the amount planned for those periods. Here, according to the Gantt chart, the tasks are finished up to wireframe designing part of construction phase. The start date of the project is August 8th, 2019 and finished date is September 5th. As the project is developing based on RUP methodology. There are four phases in RUP methodology. In inception phase, there are sub tasks such as deciding the topic, research about the topic and securing code resources. For the completion of these tasks of inception phase, it took 10 days for me. After completion of inception phase, Elaboration phase is started. The start date of Elaboration phase is September 1st, 2019 and finished date is November 1st, 2019. It took 46 days for the completion of this phase. The sub tasks of elaboration phase are analysis, draft software specification, develop models, timeline and strategies. In this phase, analysis of the project and detail process or working method of the online voting system was done. Work process is separated based on timeline and Gantt chart is created which help in making strategies to complete the project in time. After completion of elaboration phase, Construction phase is started. The start date of Construction phase is October 1st, 2019 and expected to finish in March 2nd, 2020. It took 2 months to complete up to wireframe designing part. In this phase, use case and ERDs are created.

Wireframes are design as per to meet the requirements of this project. Hence the project is completed in June 5, 2020.

3.7 Implementation

System Architecture

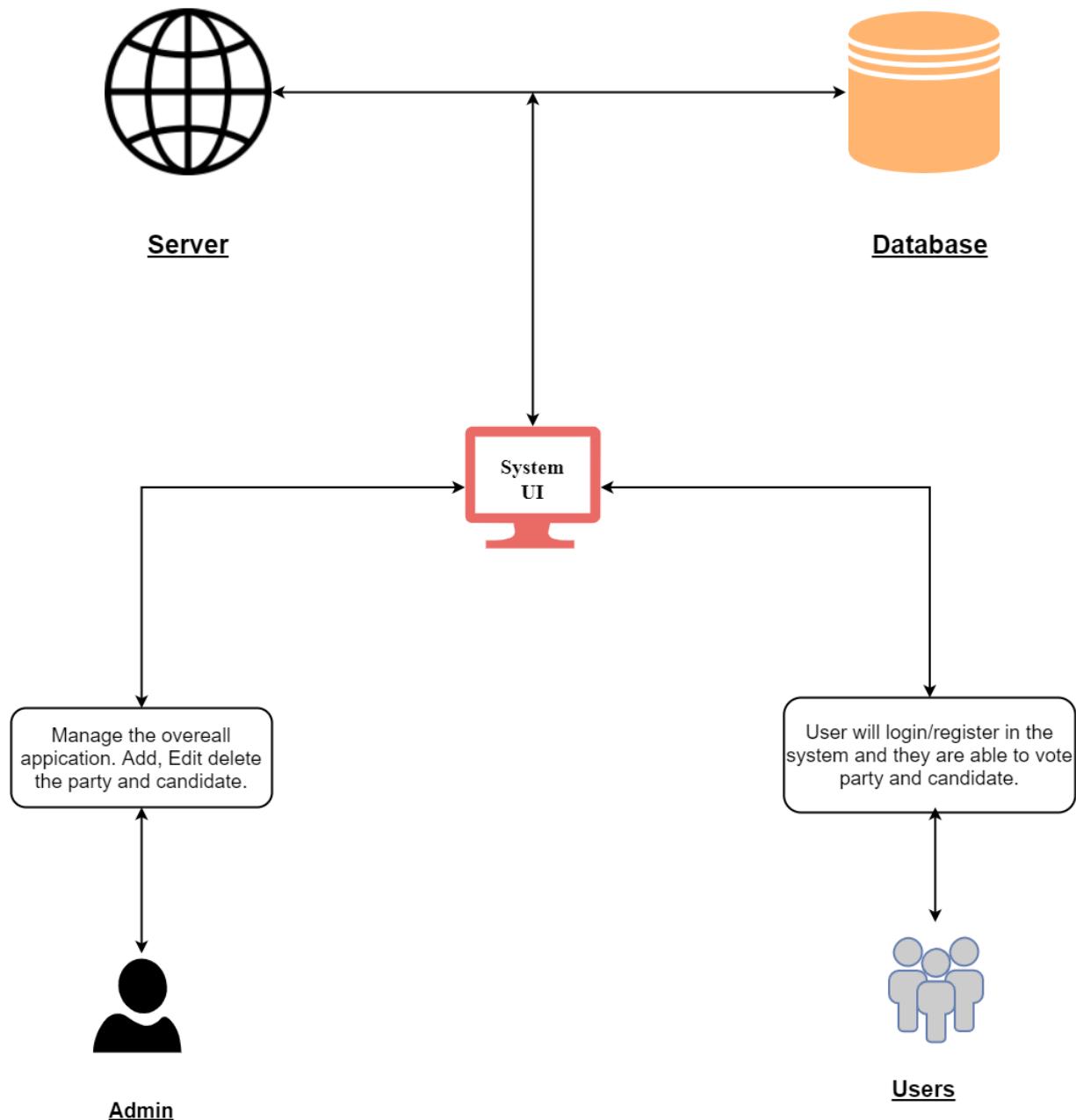


Figure 40: System Architecture

Login Page

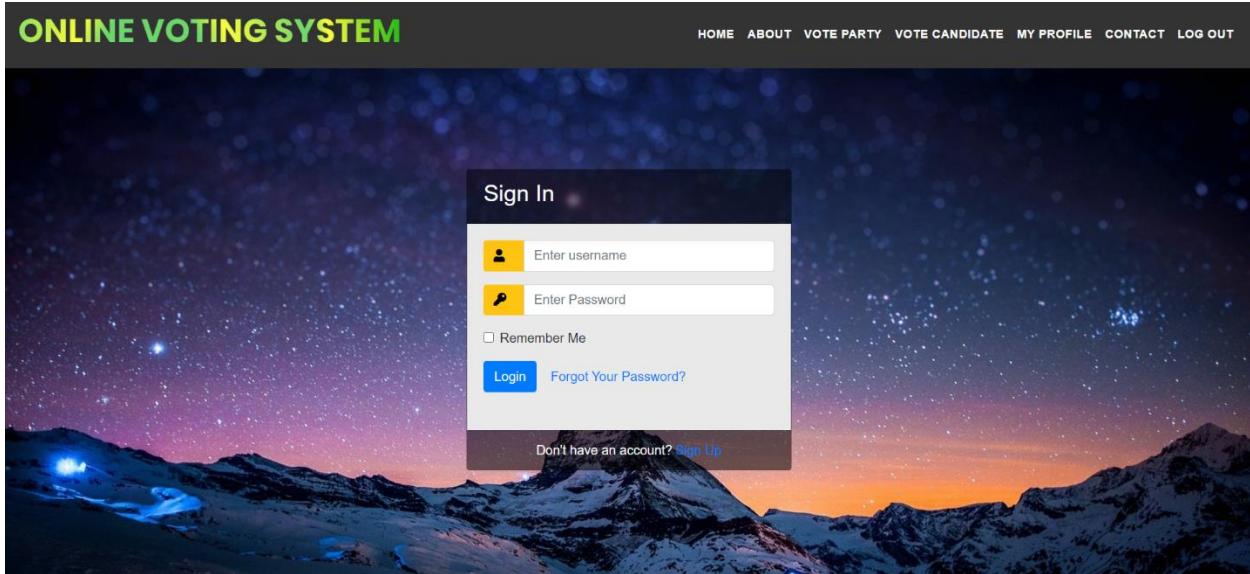


Figure 41: Login Page of online voting system

Here the above figure is of the login page. User should fill the required area in order to login. If user have not account, he/she can redirect into register page by clicking sign up.

Register page

The screenshot shows the 'Register' page of an online voting system. At the top, there is a dark header bar with the text 'ONLINE VOTING SYSTEM' in green. To the right of the header are navigation links: HOME, ABOUT, VOTE PARTY, VOTE CANDIDATE, MY PROFILE, CONTACT, and LOG OUT. Below the header, the word 'Register' is centered in green text. The main form area has a light gray background and rounded corners. It contains the following fields:

- First Name: A text input field with placeholder text 'Enter First Name'.
- Last Name: A text input field with placeholder text 'Enter Last Name'.
- Email: A text input field with placeholder text 'Enter Email'.
- Password: A text input field with placeholder text 'Enter Password'.
- Confirm Password: A text input field with placeholder text 'Enter Password'.
- Province: A dropdown menu set to 'Bagmati'.
- Phone Number: A text input field with placeholder text 'Enter phone number'.
- Citizenship Number: A text input field with placeholder text 'Enter citizenship number'.
- Gender: A dropdown menu set to 'Male'.
- Profile Image: A file upload section with a 'Choose file' button, a 'Browse' button, and an 'Upload' button.

At the bottom of the form is a large, dark blue 'REGISTER' button.

Figure 42: Register page of online voting system

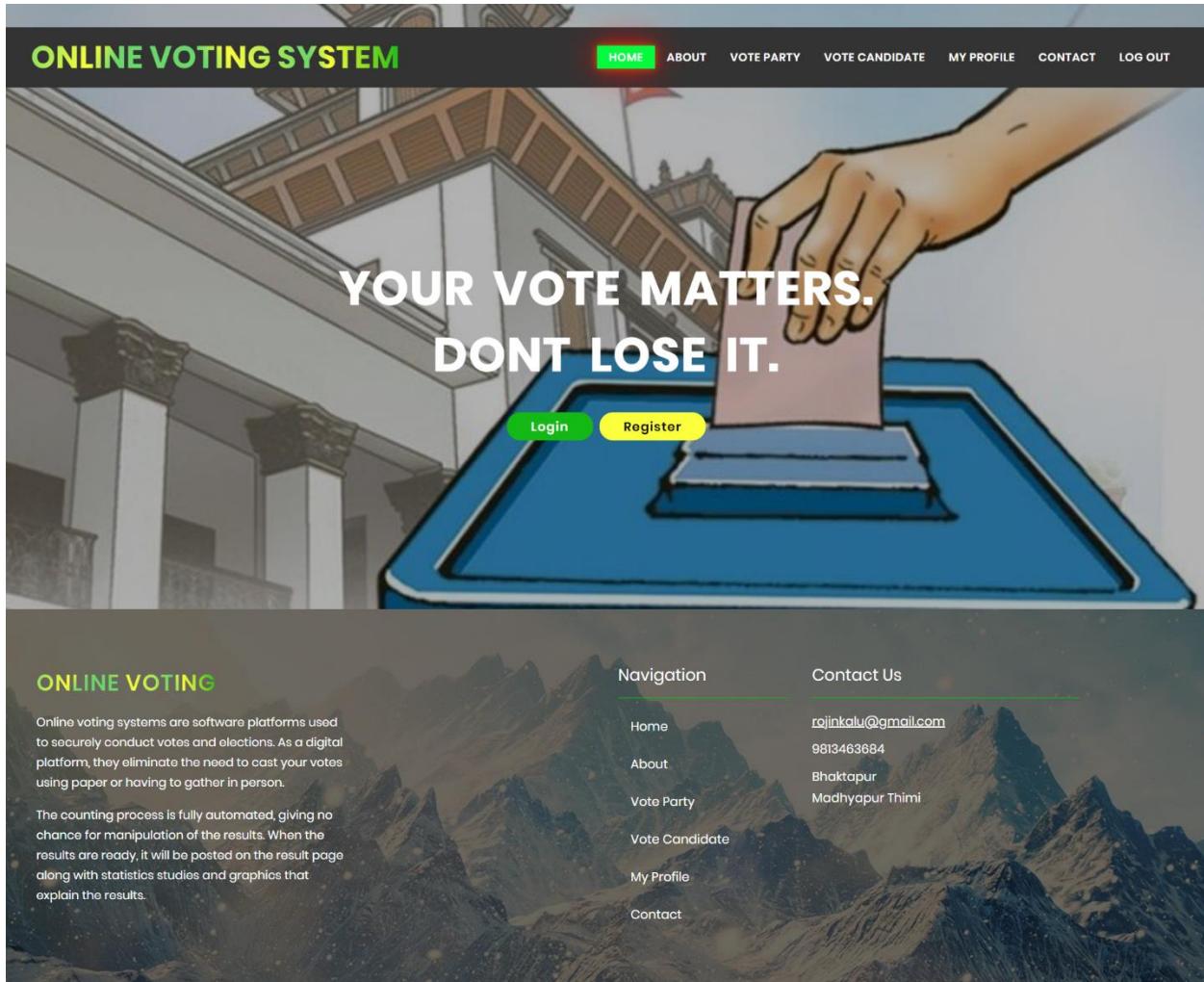
Home Page

Figure 43: Home page of online voting system

Here, above picture is of home page of online voting system. As we can see there are many navigation bars. User redirect to the page which they click. There are also login and register button.

Vote Party Page

The screenshot displays the 'Vote Party' section of the 'ONLINE VOTING SYSTEM'. At the top, there is a navigation bar with links: HOME, ABOUT, VOTE PARTY (which is highlighted in green), VOTE CANDIDATE, MY PROFILE, CONTACT, and LOG OUT. Below the navigation bar, the text 'Vote Your Selective Party' is centered. There are four boxes representing different parties: Congress (a green tree icon), Communist (a red flag with a hammer and sickle icon), Yamalo (a red sunburst icon with the text 'YAMALO' and 'CPY'), and Samajbadi (an icon of a hand holding a torch). Below these, there is another box for the Rastriya Prajatanta Party, which features a yellow elephant icon on a red, yellow, and blue background. A message indicates that the user has voted for Yamalo and provides a link to 'View Candidate'. The bottom section of the page contains a large image of snow-capped mountains, with a sidebar containing information about online voting, a navigation menu, and contact details.

ONLINE VOTING SYSTEM

HOME ABOUT VOTE PARTY VOTE CANDIDATE MY PROFILE CONTACT LOG OUT

Vote Your Selective Party

Congress

Communist

Yamalo

You Vote for Yamalo
[View Candidate](#)

Samajbadi

Rastriya Prajatanta Party

ONLINE VOTING

Online voting systems are software platforms used to securely conduct votes and elections. As a digital platform, they eliminate the need to cast your votes using paper or having to gather in person. The counting process is fully automated, giving no chance for manipulation of the results. When the results are ready, it will be posted on the result page along with statistics studies and graphics that explain the results.

Navigation

- Home
- About
- Vote Party
- Vote Candidate
- My Profile
- Contact

Contact Us

- rojinakalu@gmail.com
- 9813463684
- Bhaktapur
- Madhyapur Thimi

Figure 44: Vote Party page

Vote Candidate Page

The screenshot shows a web application titled "ONLINE VOTING SYSTEM". At the top, there is a navigation bar with links: HOME, ABOUT, VOTE PARTY, VOTE CANDIDATE (which is highlighted in green), MY PROFILE, CONTACT, and LOG OUT. Below the navigation bar, there is a heading "Vote Your Selective Candidate from Yamale". A grid of seven images of actors is displayed, each with a name below it:

- Henry Cavill
- Ben Affleck
- Jason Momoa
- Gal Gadot
- Anne Hathaway
- Berry Allan
- Chris Pine

At the bottom of the page, there is a footer section with a background image of mountains. It contains the following information:

- ONLINE VOTING**
- Online voting systems are software platforms used to securely conduct votes and elections. As a digital platform, they eliminate the need to cast your votes using paper or having to gather in person.
- The counting process is fully automated, giving no chance for manipulation of the results. When the results are ready, it will be posted on the result page along with statistics studies and graphics that explain the results.
- Navigation** links: Home, About, Vote Party, Vote Candidate, My Profile, Contact.
- Contact Us** information: rojinikalu@gmail.com, 9813463684, Bhaktapur, Madhyapur Thimi.

Figure 45: Vote candidate page

My Profile

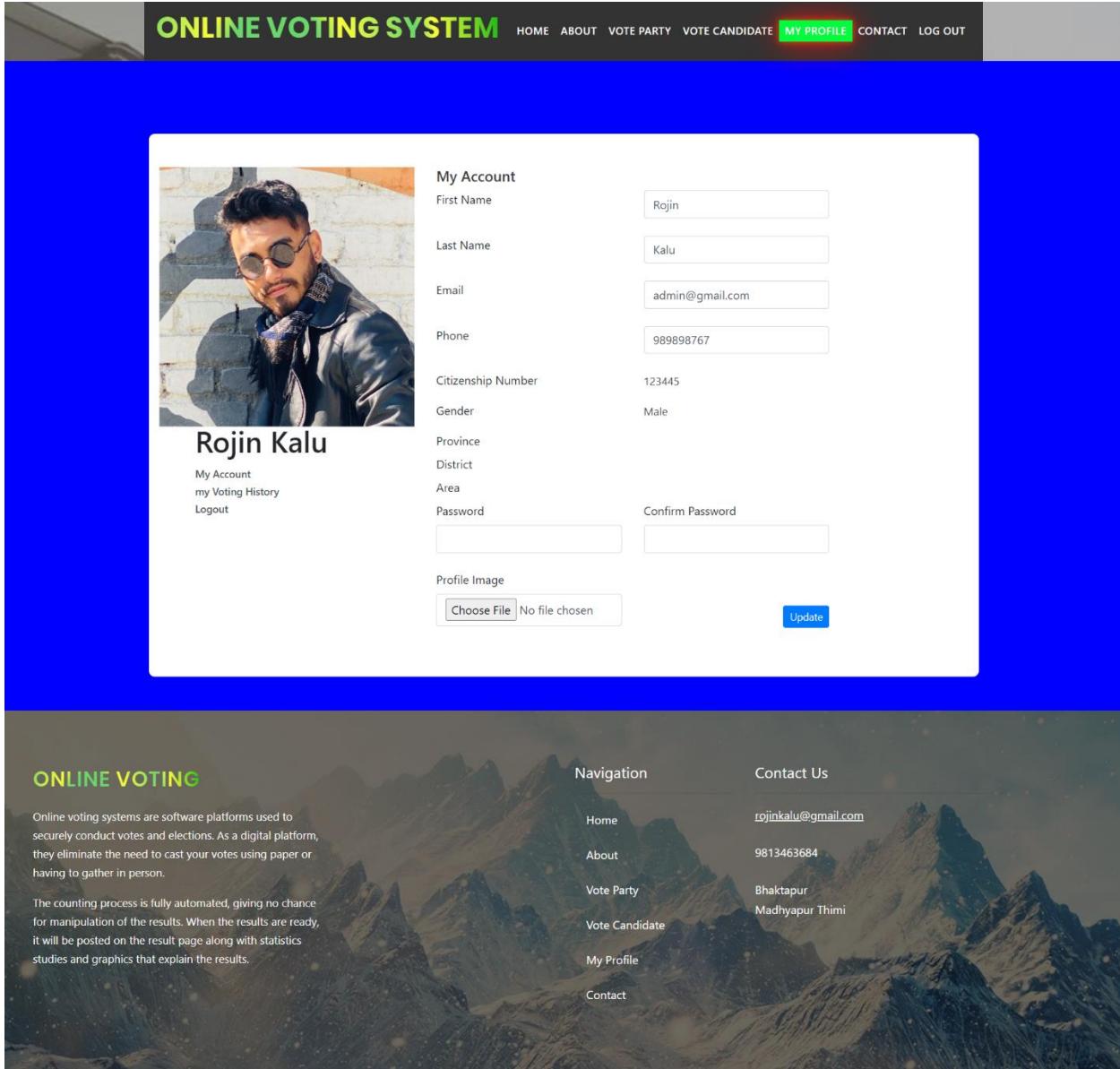


Figure 46: My profile page

Here, above picture is of account page. User can edit their profile. But cannot edit citizenship number.

Contact Page

ONLINE VOTING SYSTEM

HOME ABOUT VOTE PARTY VOTE CANDIDATE MY PROFILE **CONTACT** LOG OUT

Contact Us

Company Email	rojinakalu@gmail.com rojinakalu23@gmail.com	First Name	<input type="text"/>
Company Phone	9813463684 rojinakalu23@gmail.com	Last Name	<input type="text"/>
Company Address	Madhyapur Thimi, Nepal	Email	<input type="text"/>
		subject	<input type="text"/>
		Message	<input type="text"/>

Send

ONLINE VOTING

Online voting systems are software platforms used to securely conduct votes and elections. As a digital platform, they eliminate the need to cast your votes using paper or having to gather in person.

The counting process is fully automated, giving no chance for manipulation of the results. When the results are ready, it will be posted on the result page along with statistics studies and graphics that explain the results.

Navigation

- Home
- About
- Vote Party
- Vote Candidate
- My Profile
- Contact

Contact Us

- rojinakalu@gmail.com
- 9813463684
- Bhaktapur
- Madhyapur Thimi

Figure 47: Contact page

Chapter 4: Testing and Analysis

4.1 Test Plan

4.1.1 Unit Testing, Test Plan

- To show that admin can add party.
- To show that admin can delete party.
- To show that admin can add candidate according to party and location.
- To display confirmation message for deletion process.
- To display the successful message after adding process.
- To display confirmation message for deletion process.

4.1.2 System Testing, Test Plan

- To check whether register process works or not.
- To show the error that user cannot register with same citizenship number.
- To display the user should fill all the valid details while register.
- To check whether user can login after logout or not.
- To show that user can edit their profile but cannot edit their location.
- To show that user cannot vote without login
- To show that user cannot vote candidate without voting party.
- To show that candidate is filtered according to selective party.
- To show that candidate is filtered according to selective party
- To check whether candidate voting process works or not.
- To check whether message is sent to admin by the users or not.
- To check whether user can access the function of admin or not. Such as add party, delete party
- To check whether user can vote twice to party and candidate or not.
- To check whether logout process is valid or not.
- To display the error that email address and password should be matched, or user should register first.
- To show that password should matched with confirm password.

4.2 Unit Testing

Unit Testing 1

Objective	To show that admin can add party.
Expected Result	Party should be added successfully.
Actual Result	Party is added successfully.
Conclusion	Successful

Table 5: unit testing 1

Adding party details

Figure 48: Unit Testing 1.1

Display of Party after Adding Party details

Party Name	Party Logo	Delete	Edit
Congress		<button>Delete</button>	<button>Edit</button>
Communist		<button>Delete</button>	<button>Edit</button>
Yamale		<button>Delete</button>	<button>Edit</button>
Samajbadi		<button>Delete</button>	<button>Edit</button>
Rastriya Prajatanta Party		<button>Delete</button>	<button>Edit</button>

Figure 49: Unit testing 1.2

Unit Testing 2

Objective	To show that admin can delete party.
Expected Result	Party should be deleted successfully.
Actual Result	Party is deleted successfully.
Conclusion	Successful

Table 6: unit testing 2

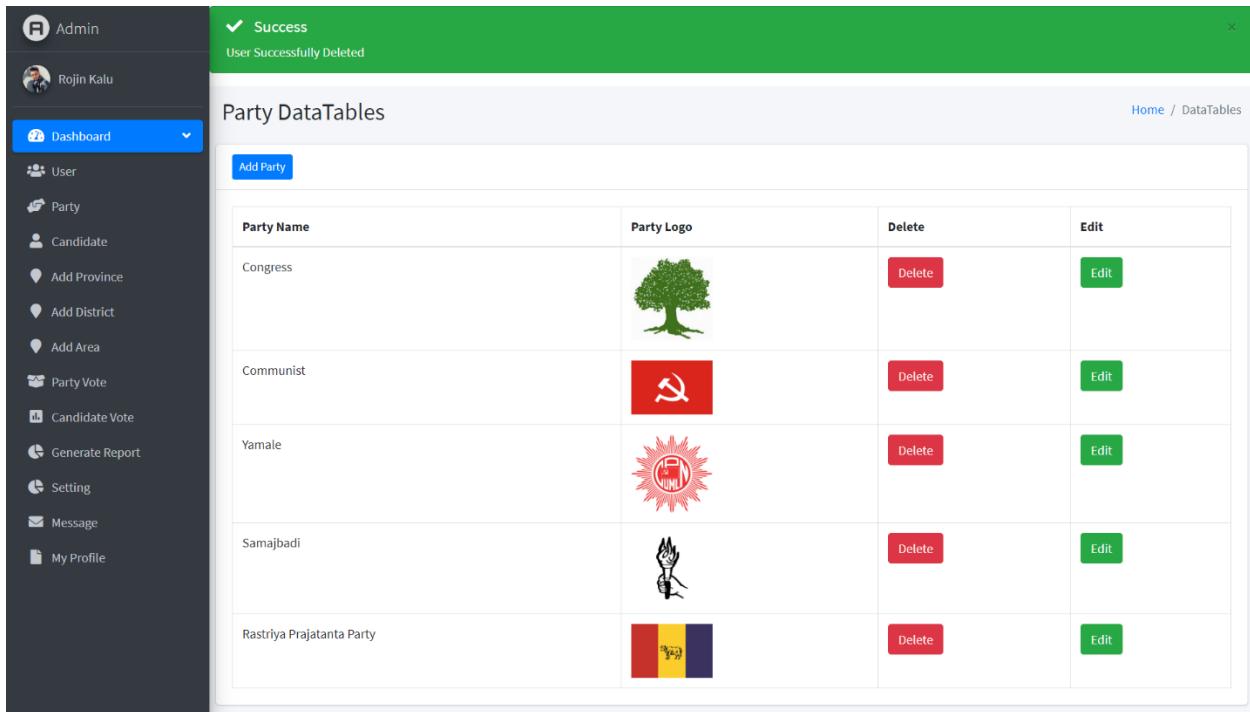
Before deleting party name “test”



Party Name	Party Logo	Delete	Edit
Congress		<button>Delete</button>	<button>Edit</button>
Communist		<button>Delete</button>	<button>Edit</button>
Yamale		<button>Delete</button>	<button>Edit</button>
Samajbadi		<button>Delete</button>	<button>Edit</button>
Rastriya Prajatanta Party		<button>Delete</button>	<button>Edit</button>
Test		<button>Delete</button>	<button>Edit</button>

Figure 50: unit testing 2.1

After deleting party name “test”



Party Name	Party Logo	Delete	Edit
Congress		<button>Delete</button>	<button>Edit</button>
Communist		<button>Delete</button>	<button>Edit</button>
Yamale		<button>Delete</button>	<button>Edit</button>
Samajbadi		<button>Delete</button>	<button>Edit</button>
Rastriya Prajatanta Party		<button>Delete</button>	<button>Edit</button>

Figure 51: unit testing 2.2

Unit testing 3

Objective	To show that admin can add candidate according to party and location.
Expected Result	System should display the list of party that candidate to show candidate party name and candidate should add successfully.
Actual Result	Candidate is added successfully.
Conclusion	Successful

Table 7: Unit testing 3

The screenshot shows a web application interface for an administrator. On the left is a dark sidebar with user profile information (Rojin Kalu) and a navigation menu with items like Dashboard, User, Party, Candidate, Add Province, Add District, Add Area, Party Vote, Candidate Vote, Generate Report, Setting, Message, and My Profile. The main content area is titled 'General Form' and contains a form titled 'Fill the Form'. The form fields are: Candidate Name (input: Test Test), Party ID (dropdown: Samajbadi), Position (dropdown: Chair Person), Province (dropdown: Province 1), District (dropdown: Jhapa), Area (dropdown: Area 2), and an 'Upload Candidate Image' section with 'Choose file' (input), 'Browse' (button), and 'Upload' (button). A large blue 'Submit' button is at the bottom of the form. The top right corner shows a breadcrumb path: Home / General Form.

Figure 52: Unit testing 3.1

Setting	Message	My Profile	Reon bbb	1	Bagmati	Kathmandu	Area 2	Chair Person		Delete	Edit
			Test Test	4	Province 1	Jhapa	Area 2	Chair Person		Delete	Edit

Figure 53: unit testing 3.2

Unit testing 4

Objective	To display confirmation message for deletion process.
Expected Result	System should display confirmation message
Actual Result	System displayed confirmation message.
Conclusion	Successful

Table 8: unit testing 4

The screenshot shows a user interface for managing political parties. On the left is a dark sidebar with navigation links: Admin, User, Party, Candidate, Add Province, Add District, Add Area, Party Vote, Candidate Vote, Generate Report, Setting, Message, and My Profile. The 'Party' link is highlighted. The main area has a table titled 'Party Name' with rows for Congress, Communist, Yamale, Samajbadi, Rastriya Prajatanta Party, and Test. Each row contains a small party logo icon and 'Delete' and 'Edit' buttons. A modal dialog box is centered over the first row (Congress). It displays the text '127.0.0.1:8000 says' and 'Are you sure you want to Delete?'. It has 'OK' and 'Cancel' buttons. The background table rows are partially visible behind the dialog.

Figure 54: unit testing 4

Unit testing 5

Objective	To display the successful message after adding process.
Expected Result	System should display the successful message after adding process.
Actual Result	System displayed the successful message after adding process.
Conclusion	Successful

Table 9: Unit testing 5

The screenshot shows a user interface for managing political parties. On the left, there is a sidebar with a dark background containing icons and text for Admin, Rojin Kalu, Dashboard (selected), User, Party, Candidate, Add Province, Add District, Add Area, Party Vote, and Candidate Vote. The main area has a green header bar with a checkmark icon and the text "Success" followed by "Party Successfully Added". Below this is a "General Form" section with a blue header "Fill the Form". It contains fields for "Party Name" (with placeholder "Enter Party Name") and "Upload Party Logo" (with a "Choose file" input, "Browse" button, and "Upload" button). At the bottom is a blue "Submit" button.

Figure 55: Unit testing 5

4.3 System Testing

System Testing 1

Objective	To check whether register process works or not.
Expected Result	System transfer to profile account.
Actual Result	Profile Account page opens.
Conclusion	Successful

Table 10: System Testing 1

Before Registration system

Figure 56: Testing 1.1

After Registration process

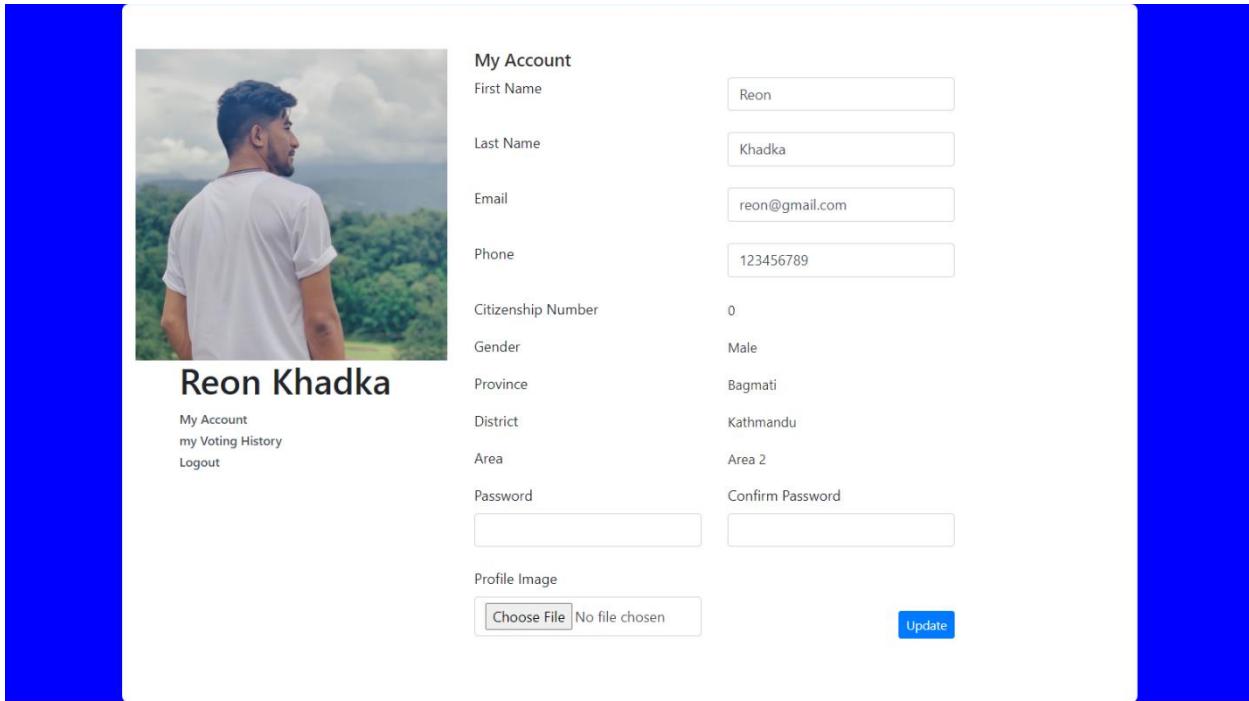


Figure 57: Testing 1.2

System Testing 2

Objective	To show the error that user cannot register with same citizenship number.
Expected Result	System should display message.
Actual Result	Displayed error message.
Conclusion	Successful

Table 11: Testing 2

The screenshot shows the 'Register' page of the 'ONLINE VOTING SYSTEM'. The page has a dark header bar with the title 'ONLINE VOTING SYSTEM' and a navigation menu with links: HOME, ABOUT, VOTE PARTY, VOTE CANDIDATE, MY PROFILE, CONTACT, and LOG OUT. Below the header is a registration form with fields for First Name, Last Name, Email, Password, Confirm Password, Province (dropdown menu set to 'Bagmati'), Phone Number, Citizenship Number (text input field containing 'Enter citizenship number'), and Gender (dropdown menu set to 'Male'). A red error message 'The citizenshipno has already been taken.' is displayed above the citizenship number field. At the bottom of the form is a 'REGISTER' button.

Figure 58: System Testing 2

System Testing 3

Objective	To display the user should fill all the valid details while register.
Expected Result	System should display message.
Actual Result	System displayed message.
Conclusion	Successful

Table 12: Testing 3

Register

First Name

Last Name

Email
 ! Please fill out this field.

Password
 ! Please fill out this field.

Confirm Password

Province ▼

Phone Number

Citizenship Number

Gender ▼

Profile Image
 Browse Upload

REGISTER

Figure 59: Testing 3

System Testing 4

Objective	To check whether user can login after logout or not.
Expected Result	User should login into system.
Actual Result	User logged in.
Conclusion	successful

Table 13: Testing 4

Before Login

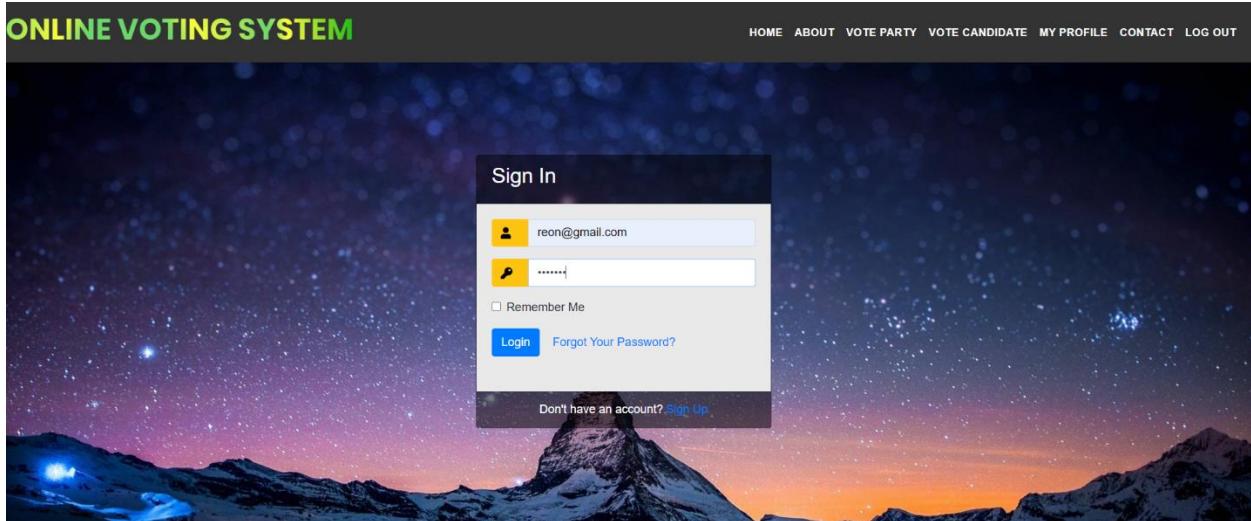


Figure 60: Testing 4.1

After login

My Account

First Name	Reon
Last Name	Khadka
Email	reon@gmail.com
Phone	123456789
Citizenship Number	0
Gender	Male
Province	Bagmati
District	Kathmandu
Area	Area 2
Password	[empty]
Confirm Password	[empty]
Profile Image	<input type="button" value="Choose File"/> No file chosen
<input type="button" value="Update"/>	

Figure 61: Testing 4.2

System Testing 5

Objective	To show that user can edit their profile but cannot edit their location.
Expected Result	The phone number should be changed.
Actual Result	Phone number is changed.
Conclusion	successful

Table 14: Testing 5.1

Before Editing



Rojin Kalu

My Account
my Voting History
Logout

My Account	
First Name	Rojin
Last Name	Kalu
Email	rojinkalu@gmail.com
Phone	981346368
Citizenship Number	1234
Gender	Male
Province	Bagmati
District	Bhaktapur
Area	Area 1
Password	<input type="password"/>
Confirm Password	<input type="password"/>
Profile Image	<input type="file"/> No file chosen
<input type="button" value="Choose File"/> <input type="button" value="Update"/>	

Figure 62: Testing 5.1

After Editing



Rojin Kalu

My Account
my Voting History
Logout

My Account	
First Name	Rojin
Last Name	Kalu
Email	rojinkalu@gmail.com
Phone	9813432342
Citizenship Number	1234
Gender	Male
Province	Bagmati
District	Bhaktapur
Area	Area 1
Password	<input type="password"/>
Confirm Password	<input type="password"/>
Profile Image	<input type="file"/> No file chosen
<input type="button" value="Choose File"/> <input type="button" value="Update"/>	

Figure 63: Testing 5.2

System Testing 6

Objective	To show that user cannot vote without login.
Expected Result	System should display login page while trying to vote the party.
Actual Result	System displayed login page.
Conclusion	successful

Table 15: Testing 6

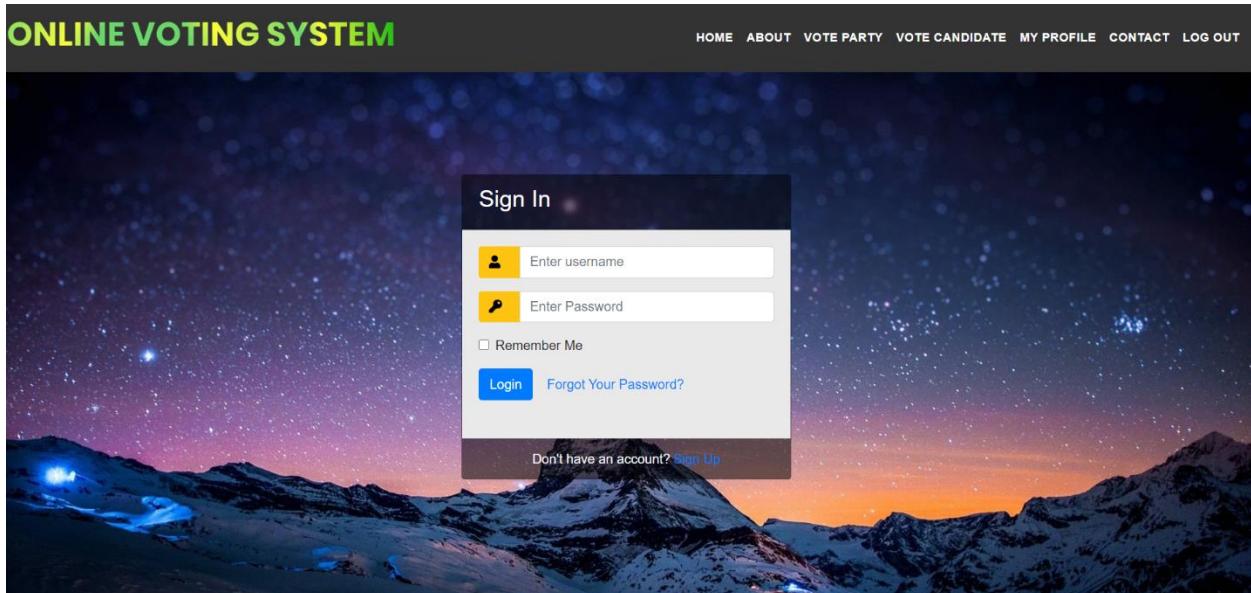


Figure 64: Testing 6

System Testing 7

Objective	To show that user cannot vote candidate without voting party.
Expected Result	System should display a message to vote party.
Actual Result	System displayed a message.
Conclusion	successful

Table 16: Testing 7

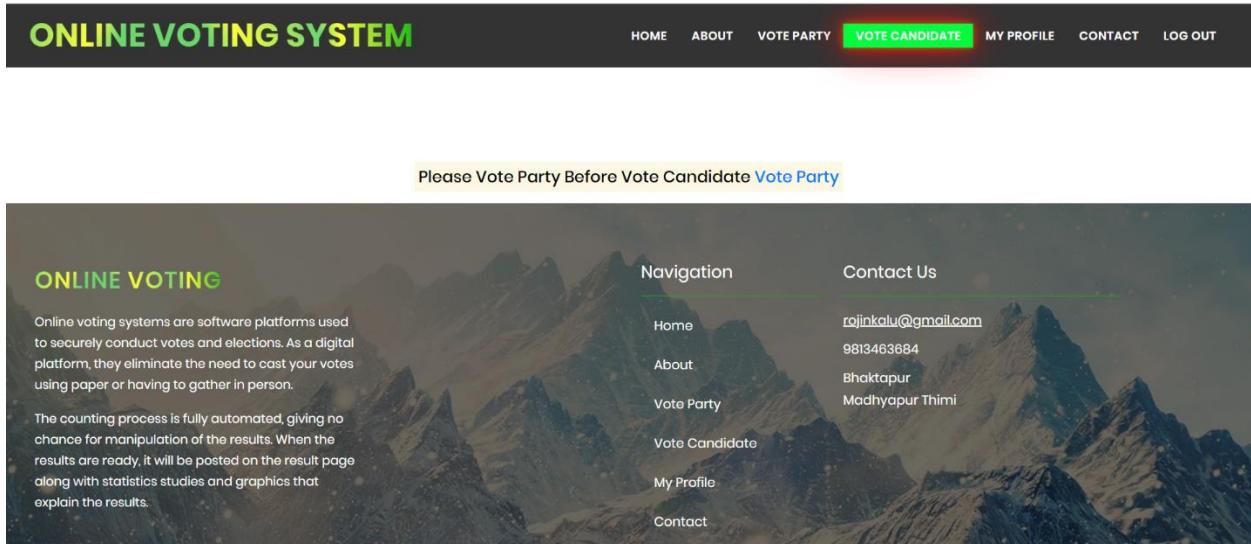


Figure 65: Testing 7

System Testing 8

Objective	To check whether voting process is successful or not.
Expected Result	System should display a message and send navigation to vote candidate.
Actual Result	System displayed a message and send navigation to vote candidate.
Conclusion	successful

Table 17: Testing 8

The screenshot shows a dark-themed web application. At the top, there is a navigation bar with links: HOME, ABOUT, VOTE PARTY (which is highlighted in red), VOTE CANDIDATE, MY PROFILE, CONTACT, and LOG OUT. Below the navigation bar, a green banner displays the text "You Vote For Yamale" with a close button ("X"). Underneath the banner, the heading "Vote Your Selective Party" is centered. Four party logos are displayed in boxes:

- Congress:** An illustration of a large green tree.
- Communist:** A red flag featuring a white hammer and sickle symbol.
- Yamale:** A red sunburst logo containing a white emblem with the text "YAMALE" and "JUMLA".
- Samajbadi:** An illustration of a hand holding a lit torch.

 Below the Communist logo, the text "Communist" is written. Below the Yamale logo, the text "Yamale" is written. At the bottom of the page, the text "You Vote for Yamale" and a blue link "View Candidate" are visible.

Figure 66: Testing 8

System Testing 9

Objective	To show that candidate is filtered according to selective party.
Expected Result	System should filter candidate and display candidate page after user voted party.
Actual Result	System displayed candidate page after user voting party.
Conclusion	successful

Table 18: Testing 9



Figure 67: Testing 9

System Testing 10

Objective	To check whether candidate voting process works or not.
Expected Result	System should display successful message.
Actual Result	System displayed successful message.
Conclusion	successful

Table 19: Testing 10

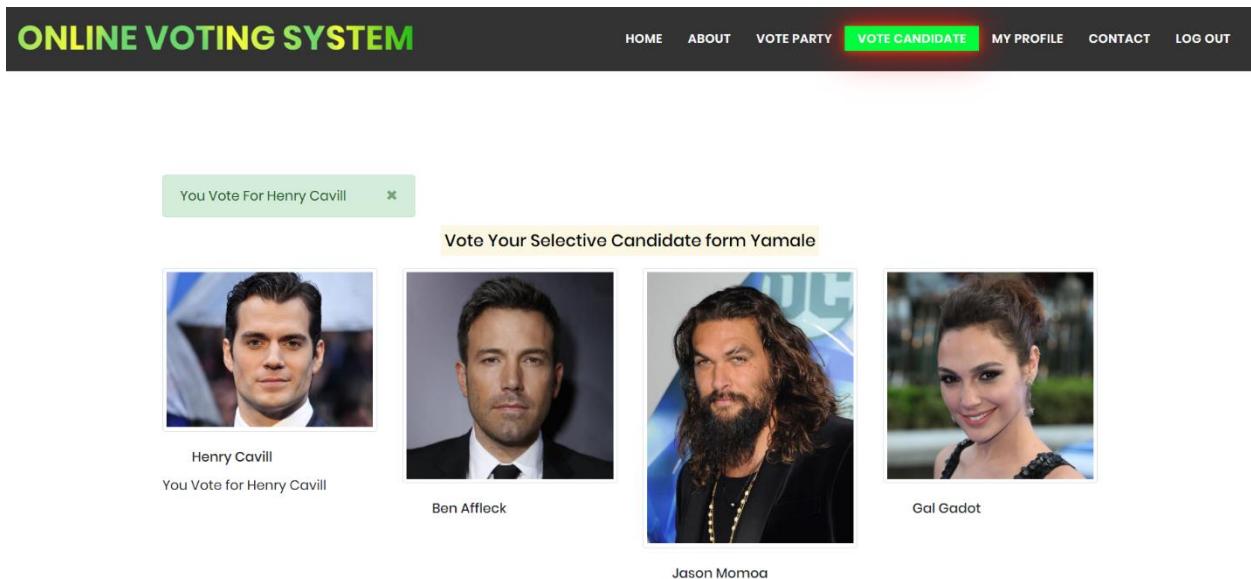


Figure 68: Testing 10

System Testing 11

Objective	To check whether message is sent to admin by the users or not.
Expected Result	Message should be recorded in database.
Actual Result	Message is recorded in database.
Conclusion	successful

Table 20: Testing 11

Filling contact form by user

Figure 69: Testing 11.1

Message sent to admin

Sn	Name	Email	Subject	Message
1	Ocean Morris	wafycunohu@mailinator.com	Laborum eaque qui ar	Consequuntur Nam ut
2	Reon Khadka	reon@gmail.com	About the system	Your system is very good.

Figure 70: Testing 11.2

System Testing 12

Objective	To check whether user can access the function of admin or not. Such as add party, delete party.
Expected Result	User will redirect to login page
Actual Result	User redirected to login page.
Conclusion	successful

Table 21: Testing 12

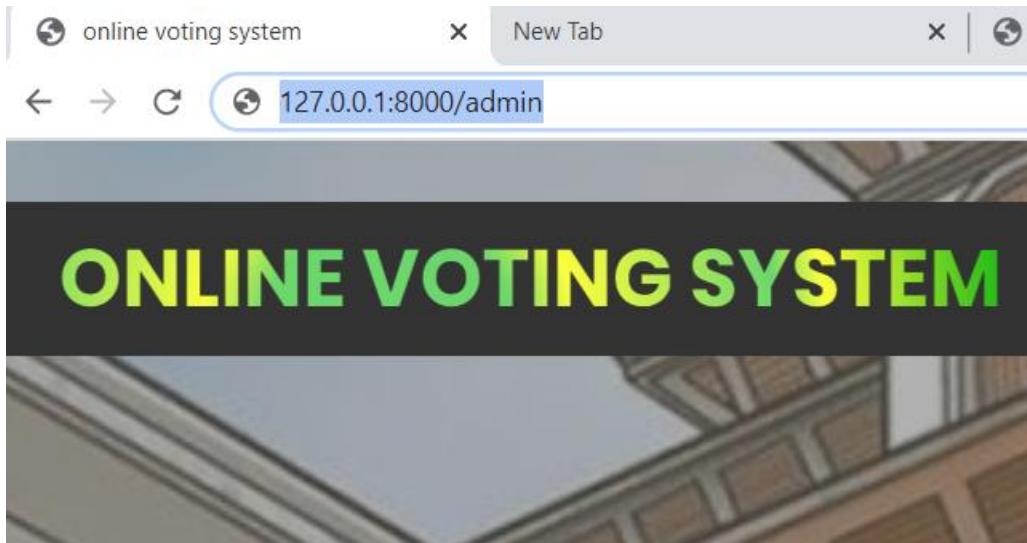


Figure 71: testing 12.1

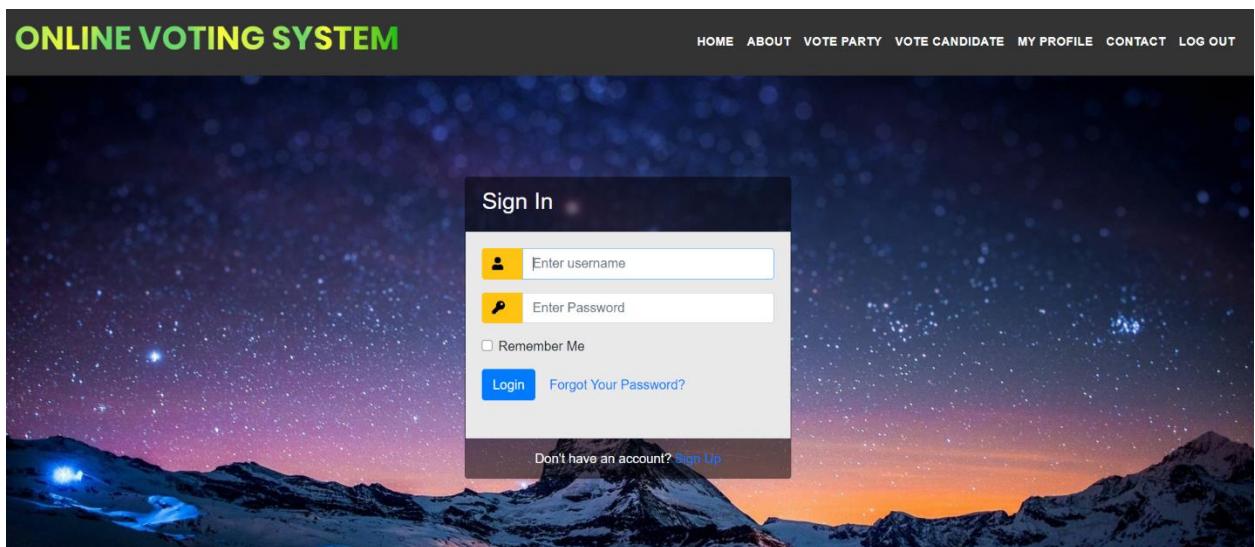


Figure 72: Testing 12.2

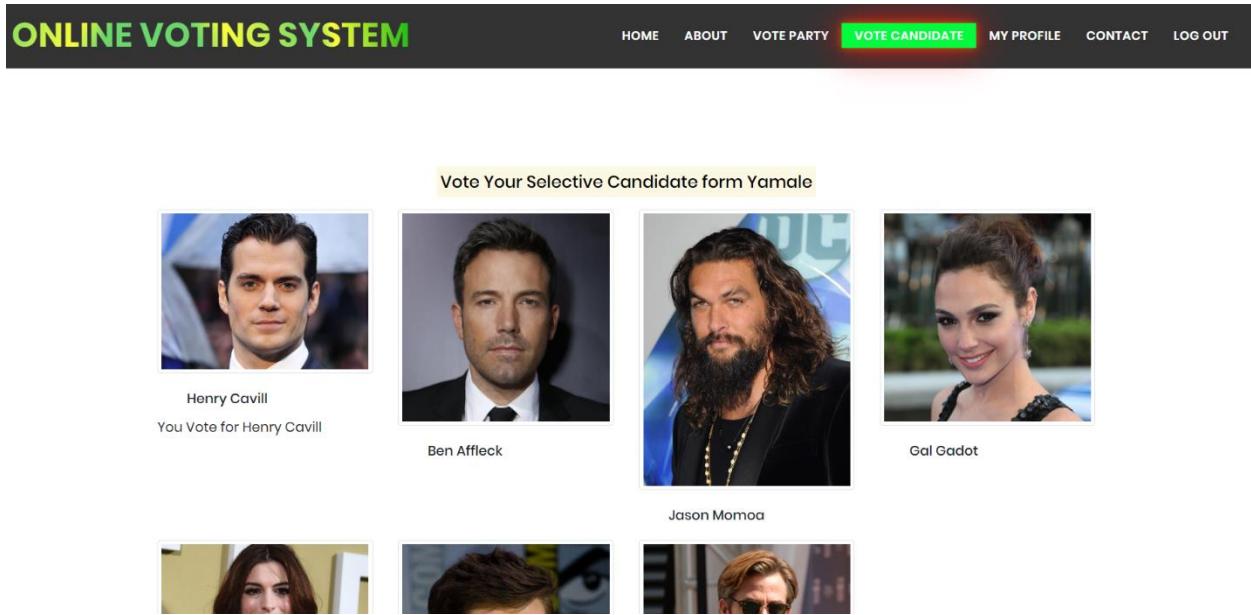
System Testing 13

Objective	To check whether user can vote twice to party and candidate or not.
Expected Result	System should display message that user already voted to selective party or candidate.
Actual Result	User redirected to login page.
Conclusion	successful

Table 22: Testing 13

The screenshot shows the 'ONLINE VOTING SYSTEM' header with a dark background and light text. Below it is a navigation bar with links: HOME, ABOUT, VOTE PARTY (highlighted in green), VOTE CANDIDATE, MY PROFILE, CONTACT, and LOG OUT. The main content area is titled 'Vote Your Selective Party'. It displays four political parties with their logos and names: Congress (green tree icon), Communist (red flag with hammer and sickle icon), Yamaile (red sunburst logo with text 'CPN (UML)'), and Samajbadi (flame icon). Below the Yamaile logo, there is a message: 'You Vote for Yamale' and a link 'View Candidate'.

Figure 73: Testing 13.1

*Figure 74: Testing 13.2*

System Testing 14

Objective	To check whether logout process is valid or not.
Expected Result	After user click on logout, system should redirect to homepage.
Actual Result	After user click on logout, system redirected to homepage.
Conclusion	successful

Table 23: Testing 14

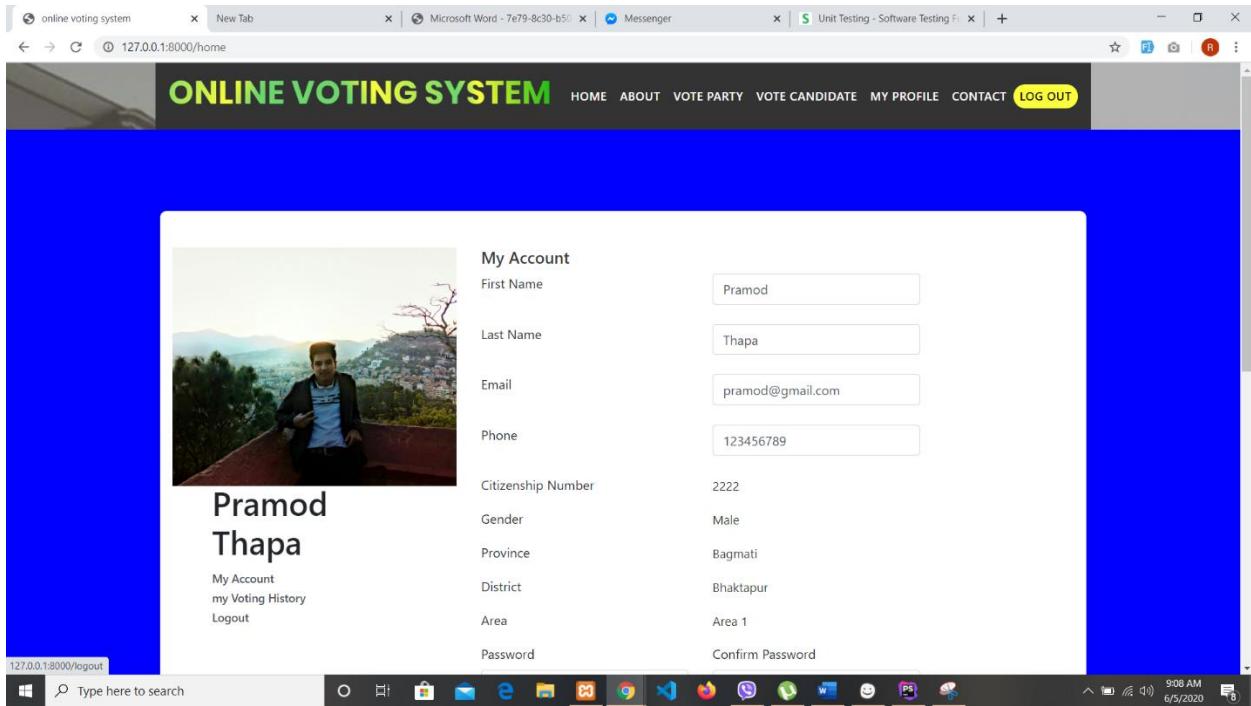


Figure 75: Testing 14.1

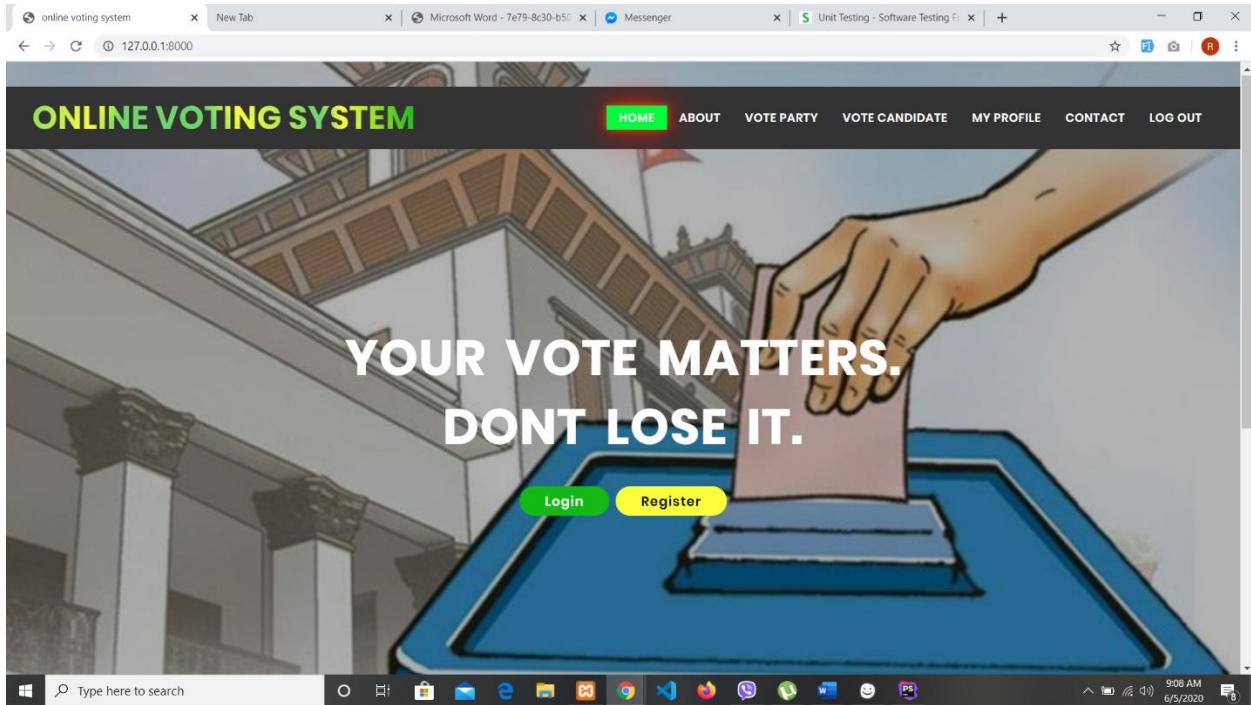


Figure 76: Testing 14.2

System Testing 15

Objective	To display the error that email address and password should be matched, or user should register first.
Expected Result	System should display error if any of email address or password is incorrect.
Actual Result	System displayed error if any of email address or password is incorrect.
Conclusion	successful

Table 24: Testing 15

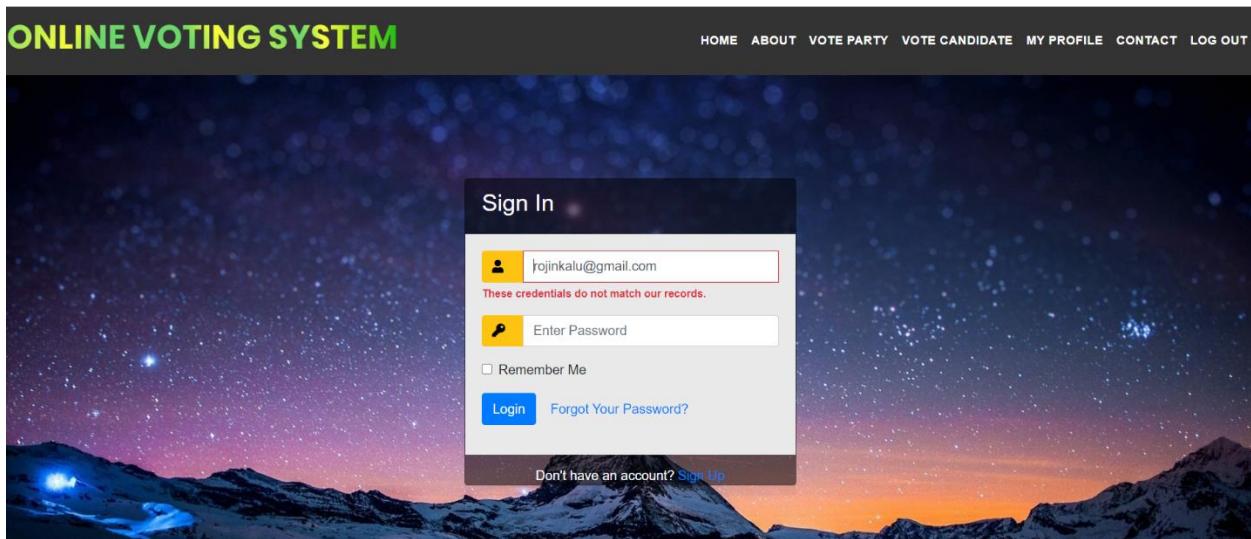


Figure 77: System testing 15

System testing 16

Objective	To show that password should matched with confirm password.
Expected Result	System should display error message.
Actual Result	System displayed error message.
Conclusion	successful

Table 25: System Testing 16

The screenshot shows the registration form for the "ONLINE VOTING SYSTEM". The form fields include First Name, Last Name, Email, Password, Confirm Password, Province (Bagmati), Phone Number, Citizenship Number, Gender (Male), and Profile Image. A red error message "The password confirmation does not match." is displayed above the Confirm Password field. The "REGISTER" button is at the bottom.

Field	Value
First Name	Enter First Name
Last Name	Enter Last Name
Email	Enter Email
Password	Enter Password
Confirm Password	Enter Password
Province	Bagmati
Phone Number	Enter phone number
Citizenship Number	Enter citizenship number
Gender	Male
Profile Image	Choose file <input type="file"/> Browse <input type="button"/> Upload <input type="button"/>

Figure 78: System Testing 16

4.4 Critical Analysis

The project is successfully developed and tested. This project is the online voting system which is web application. The entire project was fully completed through lot of trial and errors. The task assigned in the coursework was not easy at all. It required lots of labor and research. For the successful completion of the tasks, each task was carried out in steps, every step deploying the full effort. At first, lots of study and research was done on the relevant topics like Laravel, php codes, html, CSS, Laravel encryption method and many more. Here to analysis this project, it is much more efficient and user friendly than manual voting.

In this system the voter does not have to go to the polling booth to cast their vote. They can use their personal computer to cast their votes. There is a database which is maintained in which all the name of the voters with their complete information is stored. The System Administrator registers the voters by simply filling a registration form to register the voters. After registration they can vote to party and candidate.

Chapter 5: Conclusion

5.1 Legal, Social and Ethical Issues

5.1.1 Legal Issues

On legal view, the most import part of the online voting system is security. Because it is about the election and many powerful people can hires hacker or try to intervene in process. Well, this online voting system have already using Laravel encryption method.

A well-designed e-voting system should produce an audit trail that is even stronger than that of conventional systems (including paper-based systems). Remote Internet voting systems pose significant risk to the integrity of the voting process and should not be fielded for use in public elections until substantial technical and social science issues are addressed. Conclusively the paper focuses on the specific attributes an electronic voting (polling place) system should respect and ensure such as transparency, verifiability, accountability, security and accuracy in relation to the constitutional requirements such as General, Free, Equal, Secret, Direct and Democratic. (Kosmopoulos, 2004)

5.1.2 Social Issues

In social science, it will be useful to investigate the actual effects of voting at home. We are not aware of significant figures that indicate effects of an unsupervised voting environment on voting behavior. Also, the willingness of people to sell or buy votes in elections is still unknown. Online voting system may prevent blackmailing, threatening as people can vote easily from home. More research is also needed in the area of trust, especially trust in information systems and the role of computer scientists and program verification technology in this area. Internet voting is neither necessary nor inevitable, but we think society should make sure that it is ready for its adoption,

since it may show itself to be an important feature of the new democracy of the information age.
(Pieters & Becker, 2005)

5.1.3 Ethical Issues

Internet voting isn't just a matter of gathering votes in an alternate manner. The development of trust within the relation between citizens and democracy has its own dynamics and rationality. As a result of the interceding job of innovation in this connection, Internet casting a ballot can change the intersubjectively comprised picture of casting a ballot and vote based system, and subsequently add to or demolish trust. Trust in the voting process, the innovation utilized and majority rules system itself are joined. If we consider democracy as an important value, we have to make sure that new technology does not destroy this trust. (Pieters & Becker, 2005)

5.2 Advantages

In today's context, there are numbers of people living in abroad. They cannot participate in voting. They cannot choose and vote the party they like. Since, with the help of online voting system, people from abroad also can vote and express their view. Inside the country also, people from villages are living in Kathmandu valley. They have not return to their village to cast a vote. Many political issues are also there in voting area. Sometimes, the voting box keep missing and exchange with another voting box. Hence the online voting system is very useful in Nepal. Online Voting is a web-based voting system that will help you manage your elections easily and securely. Also, the willingness of people to sell or buy votes in elections is still unknown. Online voting system may prevent blackmailing, threatening as people can vote easily from home.

5.3 Limitations

Whether this online voting system uses Laravel encryption method for security, there is no guarantee that the websites cannot be hacked. While in voting process, server could be slow as many people will vote at the same time. In rural areas, networks are not available in all the area. Old age people would not be able to vote as they don't understand the system.

5.3 Future Work

Here, any project or any application cannot be fully completed. An extra feature can be added in any system. Here the final report could be displayed in map according to province, district or local area. Higher security could be implemented in this system as there is higher risks of hacking the online voting system.

Chapter 6: References

Beal, V. (2020) *Webopedia* [Online]. Available from: https://www.webopedia.com/TERM/E/entity_relationship_diagram.html [Accessed 15 January 2020].

Bright Hub Project Management. (2020) *Top 10 Benefits of a Gantt Chart* [Online]. Available from: <https://www.brighthubpm.com/templates-forms/2434-top-ten-benefits-of-a-gantt-chart/> [Accessed 16 January 2020].

EDUCBA. (2019) *Advantages of PHP* [Online]. Available from: <https://www.educba.com/advantages-of-php/> [Accessed 3 January 2020].

Elleithy, K. et al. (2005) In *Advances in Computer, Information, and Systems Sciences, and Engineering*. Dordrecht: Springer. p.219.

Far, R.B. (2005) *Mobile Computing Principles: Designing and Developing Mobile Applications*. Cambridge: Cambridge Universirt Press.

Kenton, W. (2019) *Investopedia* [Online]. Available from: <https://www.investopedia.com/terms/e/end-user.asp> [Accessed 5 june 2020].

Khalid, R.B. (2019) *Arpatech* [Online]. Available from: <https://www.arpatech.com/blog/top-10-benefits-of-using-laravel-framework/> [Accessed 13 January 2020].

Kosmopoulos, A. (2004) *Aspects of Regulatory and Legal Implications on e-Voting*. Berlin: Springer.

LucidChart. (2020) *lucidchart* [Online]. Available from: <https://www.lucidchart.com/pages/data-flow->

[diagram#:~:text=A%20data%20flow%20diagram%20\(DFD,the%20routes%20between%20each%20destination.](#) [Accessed 5 May 2020].

Pieters, W. & Becker, M. (2005) Ethics of evoting An essay on requirements and values in Internet elections. *Ethics of evoting An essay on requirements and values in Internet elections.*

Requirements Analysis. (2018) *Reqtest* [Online]. Available from: <https://reqtest.com/requirements-blog/requirements-analysis/> [Accessed 6 March 2020].

Ryan, P.Y.A. & Schoenmakers, B. (2009) *E-Voting and Identity*. Luxembourg: Springer.

TatvaSoft. (2020) *Top 12 Software Development Methodologies & its Advantages / Disadvantages* [Online]. Available from: <https://www.tatvasoft.com/blog/top-12-software-development-methodologies-and-its-advantages-disadvantages/> [Accessed 7 January 2020].

Technopedia. (2020) *Hypertext Markup Language (HTML)* [Online]. Available from: <https://www.techopedia.com/definition/1892/hypertext-markup-language-html> [Accessed 16 Janaury 2020].

Technopedia. (2020) *JavaScript(JS)* [Online]. Available from: <https://www.techopedia.com/definition/3929/javascript-js> [Accessed 16 January 2020].

W3C. (2020) *HTML and CSS* [Online]. Available from: <https://www.w3.org/standards/webdesign/htmlcss.html> [Accessed 16 January 2020].

Webopedia. (2020) *RUP - Rational Unified Process* [Online]. Available from: <https://www.webopedia.com/TERM/R/RUP.html> [Accessed 7 January 2020].

Windle, D.R. & Abreo, L.R. (2003) *Software Requirements Using the Unified Process: A Practical Approach*. New Jersey: Prentice Hall.

XAMPP. (2020) *Definitions* [Online]. Available from:

<https://wwwdefinitions.net/definition/XAMPP> [Accessed 5 May 2020].

Chapter 7: Appendix

7.1 Appendix A: Pre-Survey

7.1.1 Pre-Survey Form

The screenshot shows a Google Forms survey titled "Online Voting System". The survey consists of the following questions:

- Hello !!**
I would be grateful if you could fill this survey form to share your opinion on online voting system.
- Full Name ***
Short answer text
- What is your age?**
 - 15 - 20
 - 20 - 25
 - 25 - 30
 - 30 - 35
 - 35 above
- Have you ever tried any Web Application ?**
 - Yes
 - No
- Do you have knowledge about online voting system?**
 - Yes
 - No
 - Maybe
- Would you prefer online voting system or manual (open) voting system?**
 - Online Voting System
 - Manual Voting or Open Voting
- Do you trust online system (web application)?**
 - Yes
 - No
- If online voting system launch in Nepal, Would you find easy or difficult?**
 - Easy
 - Difficult
- Feedback/Suggestions**
Long answer text

Figure 79: Pre-Survey Form

7.1.2 Sample of Filled Pre-Survey Forms

Untitled form

16 responses

Accepting responses

Summary Question Individual

< 13 of 16 >

Responses cannot be edited

Online Voting System

Hello !!
I would be grateful if you could fill this survey form to share your opinion on online voting system.
* Required

Full Name *

Sunil bade

What is your age?

15 - 20
 20 - 25
 25 - 30
 30 - 35
 35 above

Have you ever tried any Web Application ?

Yes
 No

Do you have knowledge about online voting system?

Yes
 No
 Maybe

Would you prefer online voting system or manual (open) voting system?

Online Voting System
 Manual Voting or Open Voting

Do you trust online system (web application)?

Yes
 No

If online voting system launch in Nepal, Would you find easy or difficult?

Easy
 Difficult

Feedback/Suggestions

Everything everywhere is upgraded so we must be upgraded as well.

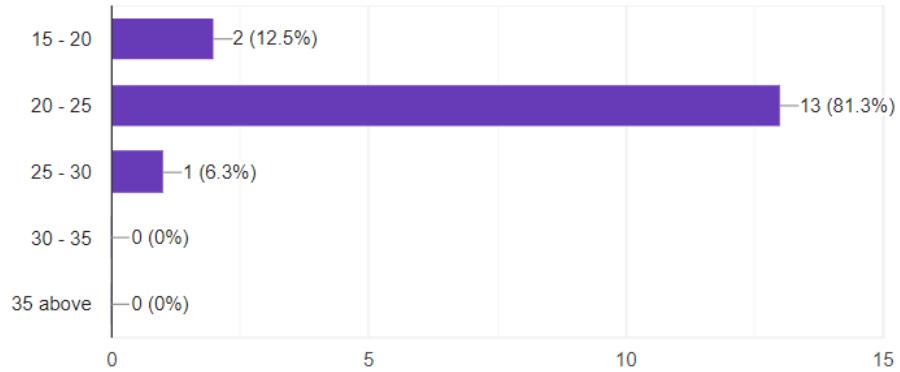
Submitted 6/3/20, 9:47 PM

Figure 80: SAMPLE OF FILLED PRE-SURVEY FORMS

7.1.3 Pre-Survey Result

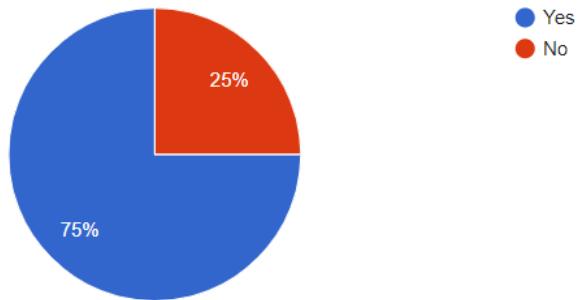
What is your age?

16 responses



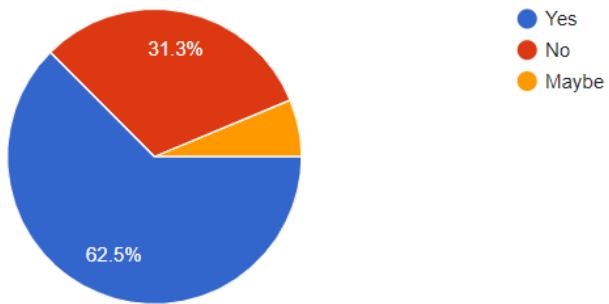
Have you ever tried any Web Application ?

16 responses



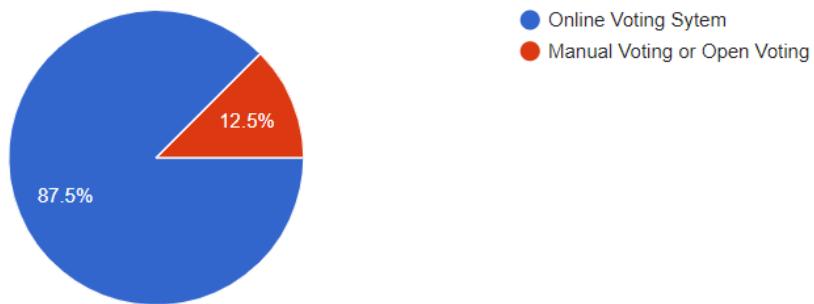
Do you have knowledge about online voting system?

16 responses



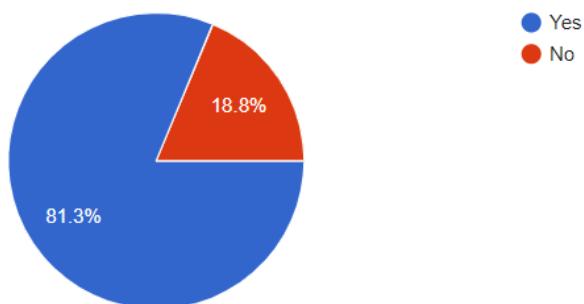
Would you prefer online voting system or manual (open) voting system?

16 responses



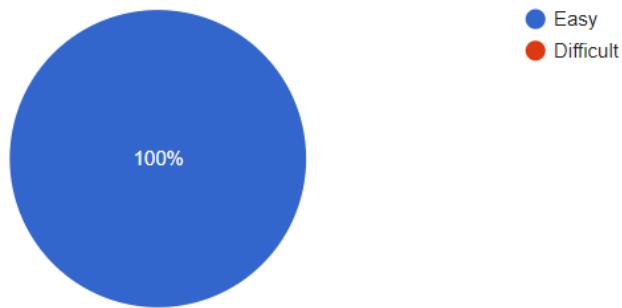
Do you trust online system (web application)?

16 responses



If online voting system launch in Nepal, Would you find easy or difficult?

16 responses



Feedback/Suggestions

6 responses

Nothing....

Good !!

Good Job !

No idea about online voting system but may not be trustable at all

Everything everywhere is upgraded so we must be upgraded as well.

It should be heavily secured so no one could hack the system.

Figure 81: result of pre-survey form

7.2 Appendix B: Post-Survey

7.2.1 Post-Survey Form

The screenshot shows a Google Forms survey titled "Online Voting System". The form consists of the following questions:

- Email Address:** A short answer text input field.
- Full Name:** A short answer text input field.
- Do you like the overall features of online voting system?** A radio button question with options "yes" and "No".
- Do you think design is user friendly?** A radio button question with options "Yes" and "No".
- Do you found the voting process easy or difficult?** A radio button question with options "Easy" and "Difficult".
- How would you rate the prototype?** A rating scale from 1 to 5.
- Would you go on manual voting or online voting?** A radio button question with options "Manual Voting" and "Online Voting".
- Feedback?** A long answer text input field.

The top right corner of the interface shows various form management icons: a plus sign, a document icon, a gear icon, a send button, and other settings and preview icons.

Figure 82: Post-Survey Form

7.2.2 Sample of Filled Pre-Survey Forms

Untitled form ☆

Questions Responses 5

5 responses

Accepting responses

Summary Question Individual

< 1 of 5 >

Responses cannot be edited

Online Voting System

Hello !
I would be grateful if you could fill this survey form to share your opinion after using online voting system.

Email Address
bishalkauri@email.com

Full Name
Bishal Kauri

Do you like the overall features of online voting system?

yes
 No

Do you think design is user friendly?

Yes
 No

Do you found the voting process easy or difficult?

Easy
 Difficult

How would you rate the prototype?

1 2 3 4 5

Would you go on manual voting or online voting?

Manual Voting
 Online Voting

Feedback?
Register process was difficult

Submitted 6/5/20, 10:50 AM

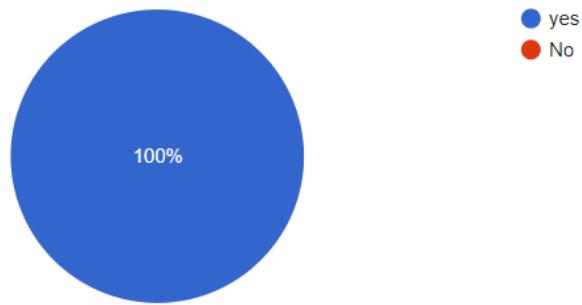
Figure 83: SAMPLE OF FILLED POST-SURVEY FORMS

7.2.3 Post-Survey Result

Do you like the overall features of online voting system?

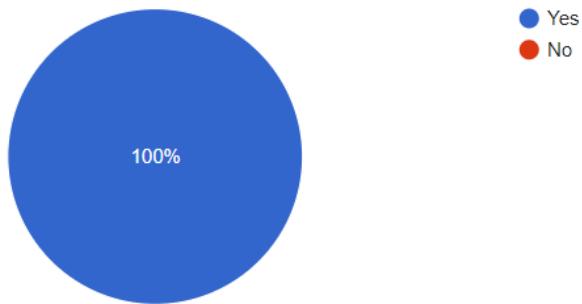


5 responses



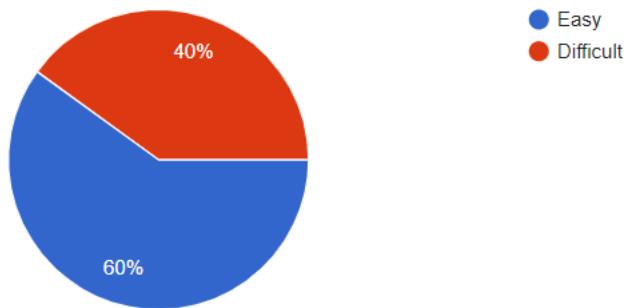
Do you think design is user friendly?

5 responses



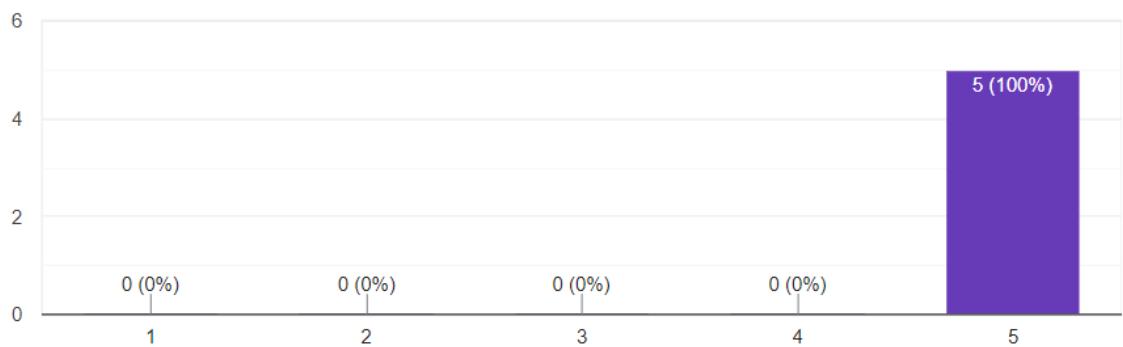
Do you found the voting process easy or difficult?

5 responses



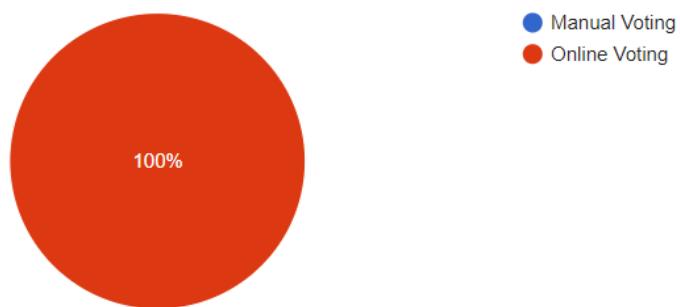
How would you rate the prototype?

5 responses



Would you go on manual voting or online voting?

5 responses



Feedback?

3 responses

problem in register

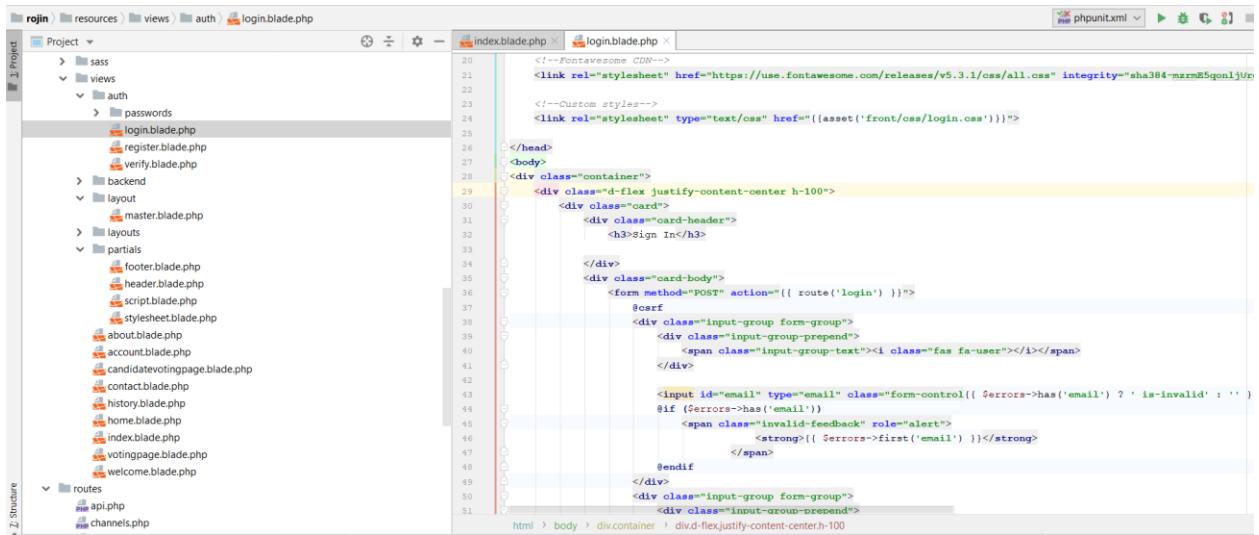
Register process was difficult

yeah man...keep going

Figure 84: Result of post survey form

7.3 Appendix C: Sample Codes

7.3.1 Sample Code of the UI



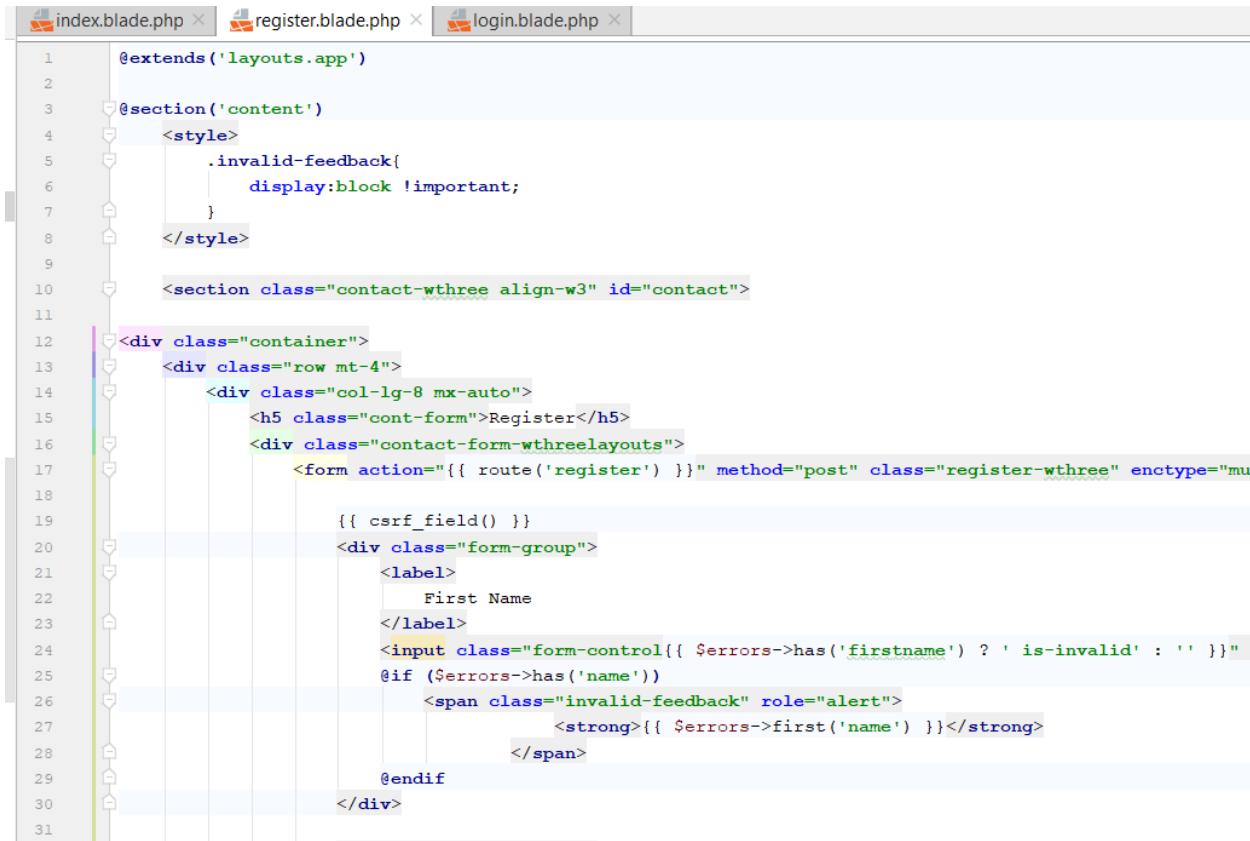
```

<!--Fontawesome CDN-->
<link rel="stylesheet" href="https://use.fontawesome.com/releases/v5.3.1/css/all.css" integrity="sha384-mzrmE5qonljUremFCDKlCcI7U7t5vNkZD5&lt;!--Custom styles-->
<link rel="stylesheet" type="text/css" href="{{ asset('front/css/login.css') }}>

</head>
<body>
<div class="container">
<div class="d-flex justify-content-center h-100">
<div class="card">
<div class="card-header">
<h3>Sign In</h3>
</div>
<div class="card-body">
<form method="POST" action="{{ route('login') }}>
    @csrf
    <div class="input-group form-group">
        <div class="input-group-prepend">
            <span class="input-group-text"><i class="fas fa-user"></i></span>
        </div>
        <input id="email" type="email" class="form-control"{{ $errors->has('email') ? ' is-invalid' : '' }}>
        @if ($errors->has('email'))
            <span class="invalid-feedback" role="alert">
                <strong>{{ $errors->first('email') }}</strong>
            </span>
        @endif
    </div>
    <div class="input-group form-group">
        <div class="input-group-prepend">
            <span class="input-group-text"><i class="fas fa-lock"></i></span>
        </div>
        <input id="password" type="password" class="form-control"{{ $errors->has('password') ? ' is-invalid' : '' }}>
        @if ($errors->has('password'))
            <span class="invalid-feedback" role="alert">
                <strong>{{ $errors->first('password') }}</strong>
            </span>
        @endif
    </div>
    <div class="form-group">
        <button type="submit" class="btn btn-primary w-100">Sign In</button>
    </div>
</form>
</div>
</div>
</div>

```

Figure 85: Screenshot of login page code

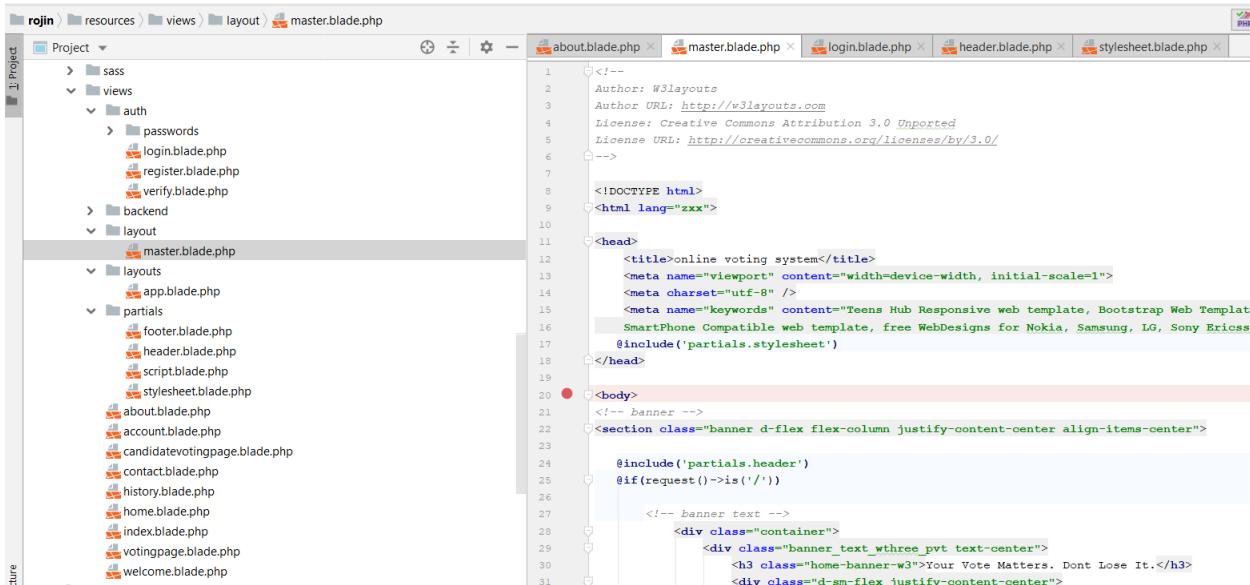


```

1 @extends('layouts.app')
2
3 @section('content')
4     <style>
5         .invalid-feedback{
6             display:block !important;
7         }
8     </style>
9
10    <section class="contact-wthree align-w3" id="contact">
11
12        <div class="container">
13            <div class="row mt-4">
14                <div class="col-lg-8 mx-auto">
15                    <h5 class="cont-form">Register</h5>
16                    <div class="contact-form-wthreelayouts">
17                        <form action="{{ route('register') }}" method="post" class="register-wthree" enctype="mu
18
19                            {{ csrf_field() }}
20                            <div class="form-group">
21                                <label>
22                                    First Name
23                                </label>
24                                <input class="form-control{{ $errors->has('firstname') ? ' is-invalid' : '' }}" type="text" name="firstname" value="{{ old('firstname') }}"/>
25                                @if ($errors->has('name'))
26                                    <span class="invalid-feedback" role="alert">
27                                        <strong>{{ $errors->first('name') }}</strong>
28                                    </span>
29                                @endif
30                            </div>
31

```

Figure 86: Screenshot of register page code



```

1 <!--
2 Author: W3layouts
3 Author URL: http://w3layouts.com
4 License: Creative Commons Attribution 3.0 Unported
5 License URL: http://creativecommons.org/licenses/by/3.0/
6 -->
7
8 <!DOCTYPE html>
9 <html lang="zxx">
10
11 <head>
12     <title>online voting system</title>
13     <meta name="viewport" content="width=device-width, initial-scale=1">
14     <meta charset="utf-8" />
15     <meta name="keywords" content="Teens Hub Responsive web template, Bootstrap Web Template
16     SmartPhone Compatible web template, free WebDesigns for Nokia, Samsung, LG, Sony Ericsson, HTC,
17     Motorola, BlackBerry, Apple iPhone, iPad, Android, iPod Touch, Samsung Galaxy, LG G2, LG G3, LG G4, LG G5, LG G6, LG V10, LG V20, LG V30, LG V40, LG V50, LG V60, LG V70, LG V80, LG V90, LG V100, LG V200, LG V300, LG V400, LG V500, LG V600, LG V700, LG V800, LG V900, LG V1000, LG V2000, LG V3000, LG V4000, LG V5000, LG V6000, LG V7000, LG V8000, LG V9000, LG V10000, LG V20000, LG V30000, LG V40000, LG V50000, LG V60000, LG V70000, LG V80000, LG V90000, LG V100000, LG V200000, LG V300000, LG V400000, LG V500000, LG V600000, LG V700000, LG V800000, LG V900000, LG V1000000, LG V2000000, LG V3000000, LG V4000000, LG V5000000, LG V6000000, LG V7000000, LG V8000000, LG V9000000, LG V10000000, LG V20000000, LG V30000000, LG V40000000, LG V50000000, LG V60000000, LG V70000000, LG V80000000, LG V90000000, LG V100000000, LG V200000000, LG V300000000, LG V400000000, LG V500000000, LG V600000000, LG V700000000, LG V800000000, LG V900000000, LG V1000000000, LG V2000000000, LG V3000000000, LG V4000000000, LG V5000000000, LG V6000000000, LG V7000000000, LG V8000000000, LG V9000000000, LG V10000000000, LG V20000000000, LG V30000000000, LG V40000000000, LG V50000000000, LG V60000000000, LG V70000000000, LG V80000000000, LG V90000000000, LG V100000000000, LG V200000000000, LG V300000000000, LG V400000000000, LG V500000000000, LG V600000000000, LG V700000000000, LG V800000000000, LG V900000000000, LG V1000000000000, LG V2000000000000, LG V3000000000000, LG V4000000000000, LG V5000000000000, LG V6000000000000, LG V7000000000000, LG V8000000000000, LG V9000000000000, LG V10000000000000, LG V20000000000000, LG V30000000000000, LG V40000000000000, LG V50000000000000, LG V60000000000000, LG V70000000000000, LG V80000000000000, LG V90000000000000, LG V100000000000000, LG V200000000000000, LG V300000000000000, LG V400000000000000, LG V500000000000000, LG V600000000000000, LG V700000000000000, LG V800000000000000, LG V900000000000000, LG V1000000000000000, LG V2000000000000000, LG V3000000000000000, LG V4000000000000000, LG V5000000000000000, LG V6000000000000000, LG V7000000000000000, LG V8000000000000000, LG V9000000000000000, LG V10000000000000000, LG V20000000000000000, LG V30000000000000000, LG V40000000000000000, LG V50000000000000000, LG V60000000000000000, LG V70000000000000000, LG V80000000000000000, LG V90000000000000000, LG V100000000000000000, LG V200000000000000000, LG V300000000000000000, LG V400000000000000000, LG V500000000000000000, LG V600000000000000000, LG V700000000000000000, LG V800000000000000000, LG V900000000000000000, LG V1000000000000000000, LG V2000000000000000000, LG V3000000000000000000, LG V4000000000000000000, LG V5000000000000000000, LG V6000000000000000000, LG V7000000000000000000, LG V8000000000000000000, LG V9000000000000000000, LG V10000000000000000000, LG V20000000000000000000, LG V30000000000000000000, LG V40000000000000000000, LG V50000000000000000000, LG V60000000000000000000, LG V70000000000000000000, LG V80000000000000000000, LG V90000000000000000000, LG V100000000000000000000, LG V200000000000000000000, LG V300000000000000000000, LG V400000000000000000000, LG V500000000000000000000, LG V600000000000000000000, LG V700000000000000000000, LG V800000000000000000000, LG V900000000000000000000, LG V1000000000000000000000, LG V2000000000000000000000, LG V3000000000000000000000, LG V4000000000000000000000, LG V5000000000000000000000, LG V6000000000000000000000, LG V7000000000000000000000, LG V8000000000000000000000, LG V9000000000000000000000, LG V10000000000000000000000, LG V20000000000000000000000, LG V30000000000000000000000, LG V40000000000000000000000, LG V50000000000000000000000, LG V60000000000000000000000, LG V70000000000000000000000, LG V80000000000000000000000, LG V90000000000000000000000, LG V100000000000000000000000, LG V200000000000000000000000, LG V300000000000000000000000, LG V400000000000000000000000, LG V500000000000000000000000, LG V600000000000000000000000, LG V700000000000000000000000, LG V800000000000000000000000, LG V900000000000000000000000, LG V1000000000000000000000000, LG V2000000000000000000000000, LG V3000000000000000000000000, LG V4000000000000000000000000, LG V5000000000000000000000000, LG V6000000000000000000000000, LG V7000000000000000000000000, LG V8000000000000000000000000, LG V9000000000000000000000000, LG V10000000000000000000000000, LG V20000000000000000000000000, LG V30000000000000000000000000, LG V40000000000000000000000000, LG V50000000000000000000000000, LG V60000000000000000000000000, LG V70000000000000000000000000, LG V80000000000000000000000000, LG V90000000000000000000000000, LG V100000000000000000000000000, LG V200000000000000000000000000, LG V300000000000000000000000000, LG V400000000000000000000000000, LG V500000000000000000000000000, LG V600000000000000000000000000, LG V700000000000000000000000000, LG V800000000000000000000000000, LG V900000000000000000000000000, LG V1000000000000000000000000000, LG V2000000000000000000000000000, LG V3000000000000000000000000000, LG V4000000000000000000000000000, LG V5000000000000000000000000000, LG V6000000000000000000000000000, LG V7000000000000000000000000000, LG V8000000000000000000000000000, LG V9000000000000000000000000000, LG V10000000000000000000000000000, LG V20000000000000000000000000000, LG V30000000000000000000000000000, LG V40000000000000000000000000000, LG V50000000000000000000000000000, LG V60000000000000000000000000000, LG V70000000000000000000000000000, LG V80000000000000000000000000000, LG V90000000000000000000000000000, LG V100000000000000000000000000000, LG V200000000000000000000000000000, LG V300000000000000000000000000000, LG V400000000000000000000000000000, LG V500000000000000000000000000000, LG V600000000000000000000000000000, LG V700000000000000000000000000000, LG V800000000000000000000000000000, LG V900000000000000000000000000000, LG V1000000000000000000000000000000, LG V2000000000000000000000000000000, LG V3000000000000000000000000000000, LG V4000000000000000000000000000000, LG V5000000000000000000000000000000, LG V6000000000000000000000000000000, LG V7000000000000000000000000000000, LG V8000000000000000000000000000000, LG V9000000000000000000000000000000, LG V10000000000000000000000000000000, LG V20000000000000000000000000000000, LG V30000000000000000000000000000000, LG V40000000000000000000000000000000, LG V50000000000000000000000000000000, LG V60000000000000000000000000000000, LG V70000000000000000000000000000000, LG V80000000000000000000000000000000, LG V90000000000000000000000000000000, LG V100000000000000000000000000000000, LG V200000000000000000000000000000000, LG V300000000000000000000000000000000, LG V400000000000000000000000000000000, LG V500000000000000000000000000000000, LG V600000000000000000000000000000000, LG V700000000000000000000000000000000, LG V800000000000000000000000000000000, LG V900000000000000000000000000000000, LG V1000000000000000000000000000000000, LG V2000000000000000000000000000000000, LG V3000000000000000000000000000000000, LG V4000000000000000000000000000000000, LG V5000000000000000000000000000000000, LG V6000000000000000000000000000000000, LG V7000000000000000000000000000000000, LG V8000000000000000000000000000000000, LG V9000000000000000000000000000000000, LG V10000000000000000000000000000000000, LG V20000000000000000000000000000000000, LG V30000000000000000000000000000000000, LG V40000000000000000000000000000000000, LG V50000000000000000000000000000000000, LG V60000000000000000000000000000000000, LG V70000000000000000000000000000000000, LG V80000000000000000000000000000000000, LG V90000000000000000000000000000000000, LG V100000000000000000000000000000000000, LG V200000000000000000000000000000000000, LG V300000000000000000000000000000000000, LG V400000000000000000000000000000000000, LG V500000000000000000000000000000000000, LG V600000000000000000000000000000000000, LG V700000000000000000000000000000000000, LG V800000000000000000000000000000000000, LG V900000000000000000000000000000000000, LG V1000000000000000000000000000000000000, LG V2000000000000000000000000000000000000, LG V3000000000000000000000000000000000000, LG V4000000000000000000000000000000000000, LG V5000000000000000000000000000000000000, LG V6000000000000000000000000000000000000, LG V7000000000000000000000000000000000000, LG V8000000000000000000000000000000000000, LG V9000000000000000000000000000000000000, LG V10000000000000000000000000000000000000, LG V20000000000000000000000000000000000000, LG V30000000000000000000000000000000000000, LG V40000000000000000000000000000000000000, LG V50000000000000000000000000000000000000, LG V60000000000000000000000000000000000000, LG V70000000000000000000000000000000000000, LG V80000000000000000000000000000000000000, LG V90000000000000000000000000000000000000, LG V100000000000000000000000000000000000000, LG V200000000000000000000000000000000000000, LG V300000000000000000000000000000000000000, LG V400000000000000000000000000000000000000, LG V500000000000000000000000000000000000000, LG V600000000000000000000000000000000000000, LG V700000000000000000000000000000000000000, LG V800000000000000000000000000000000000000, LG V900000000000000000000000000000000000000, LG V1000000000000000000000000000000000000000, LG V2000000000000000000000000000000000000000, LG V3000000000000000000000000000000000000000, LG V4000000000000000000000000000000000000000, LG V5000000000000000000000000000000000000000, LG V6000000000000000000000000000000000000000, LG V7000000000000000000000000000000000000000, LG V8000000000000000000000000000000000000000, LG V9000000000000000000000000000000000000000, LG V10000000000000000000000000000000000000000, LG V20000000000000000000000000000000000000000, LG V30000000000000000000000000000000000000000, LG V40000000000000000000000000000000000000000, LG V50000000000000000000000000000000000000000, LG V60000000000000000000000000000000000000000, LG V70000000000000000000000000000000000000000, LG V80000000000000000000000000000000000000000, LG V90000000000000000000000000000000000000000, LG V100000000000000000000000000000000000000000, LG V200000000000000000000000000000000000000000, LG V300000000000000000000000000000000000000000, LG V400000000000000000000000000000000000000000, LG V500000000000000000000000000000000000000000, LG V600000000000000000000000000000000000000000, LG V700000000000000000000000000000000000000000, LG V800000000000000000000000000000000000000000, LG V900000000000000000000000000000000000000000, LG V1000000000000000000000000000000000000000000, LG V2000000000000000000000000000000000000000000, LG V3000000000000000000000000000000000000000000, LG V4000000000000000000000000000000000000000000, LG V5000000000000000000000000000000000000000000, LG V6000000000000000000000000000000000000000000, LG V7000000000000000000000000000000000000000000, LG V8000000000000000000000000000000000000000000, LG V9000000000000000000000000000000000000000000, LG V10000000000000000000000000000000000000000000, LG V20000000000000000000000000000000000000000000, LG V30000000000000000000000000000000000000000000, LG V40000000000000000000000000000000000000000000, LG V50000000000000000000000000000000000000000000, LG V60000000000000000000000000000000000000000000, LG V70000000000000000000000000000000000000000000, LG V80000000000000000000000000000000000000000000, LG V90000000000000000000000000000000000000000000, LG V100000000000000000000000000000000000000000000, LG V200000000000000000000000000000000000000000000, LG V300000000000000000000000000000000000000000000, LG V400000000000000000000000000000000000000000000, LG V500000000000000000000000000000000000000000000, LG V600000000000000000000000000000000000000000000, LG V700000000000000000000000000000000000000000000, LG V800000000000000000000000000000000000000000000, LG V900000000000000000000000000000000000000000000, LG V100000
```

```

@extends('layout.master')

@section('section')
    <section class="single_grid w3_main align-w3" id="about" style="...>
        <div class="container">
            <div class="wthree_pvt_title text-center">
                <h4 class="w3pvt-title">{{\App\Setting::getSettings('site_title')}}</h4>
                <mark><span class="sub-title">{{\App\Setting::getSettings('site_descrip}}</span></mark>
            </div>
            <div class="row pt-md-4">
                <div class="col-lg-12">
                    <div class="single_grid_text">
                        (! \App\Setting::getSettings('about') !)
                    </div>
                </div>
            </div>
        </div>
    </section>
    <style>
        body {
            background-image: url("front/images/p..jpg");
            background-size: cover;
        }
    </style>
    <style>
        mark {
            background-color: blue;
        }
    </style>
</style>

```

Figure 88: Screenshot of coding of about page

```

@extends('layout.master')

@section('section')
    <link href="//maxcdn.bootstrapcdn.com/bootstrap/4.1.1/css/bootstrap.min.css" rel="stylesheet" id="bootstrap-css">
    <script src="//maxcdn.bootstrapcdn.com/bootstrap/4.1.1/js/bootstrap.min.js"></script>
    <script src='//cdnjs.cloudflare.com/ajax/libs/jquery/3.2.1/jquery.min.js'></script>
    <!-- Include the above in your HEAD tag -->

    <div class="container emp-profile" style="...>
        @if(\Illuminate\Support\Facades\Session::has('success'))
            <div style="...>
                <div class="alert alert-success alert-dismissible">
                    <button type="button" class="close" data-dismiss="alert" aria-hidden="true">></button>
                    {{ \Session('success') }}
                </div>
            </div>
        @endif
        <form method="post" enctype="multipart/form-data" action="{{ route('user.update') }}>
            {{ csrf_field() }}
            <input type="hidden" name="user_id" value="{{ $user->id }}>
            <div class="row">
                <div class="col-md-4">
                    <div class="profile-img">
                        
                    </div>
                    <div class="profile-work">
                        <h1> {{ $user->name }} {{ $user->lname }} </h1>
                        <a href="{{ route('account') }}>Mu Account</a>
                    </div>
                </div>
            </div>
        </form>

```

Figure 89: Screenshot of coding account page

```

votingpage.blade.php
1  @extends('layout.master')
2
3  @section('section')
4      <div class="voting--section">
5          <div class="container">
6              @if(\Illuminate\Support\Facades\Session::has('success'))
7                  <div style="...">
8                      <div class="alert alert-success alert-dismissible">
9                          <button type="button" class="close" data-dismiss="alert" aria-hidden="true">></button>
10                         {{ \session('success') }}
11                     </div>
12                 </div>
13             @endif
14
15             <h2><mark>Vote Your Selective Party</mark></h2>
16             <form method="post" action="{{ route('vote.party') }}">
17                 {{ csrf_field() }}
18             <div class="row">
19
20                 @foreach($parties as $party)
21
22                     <div class="col-md-3">
23                         <div class="thumbnail">
24                             
25                             <label class="mainradio"><h1>{{ $party->party_name }}</h1>
26
27                             @if(!$user->voteCast)
28                                 <input type="radio" value="{{ $party->id }}" name="party_id" required>
29                                 <span class="checkmark"></span>
30                             @endif
31                         </div>
32                     </div>
33                 @endforeach
34             </div>
35         </div>
36     </div>
37 
```

Figure 90: Screenshot of coding of voting page

```

candidatevotingpage.blade.php
1  @extends('layout.master')
2
3  @section('section')
4      <div class="voting--section">
5          <div class="container">
6              @if(\Illuminate\Support\Facades\Session::has('success'))
7                  <div style="...">
8                      <div class="alert alert-success alert-dismissible">
9                          <button type="button" class="close" data-dismiss="alert" aria-hidden="true">></button>
10                         {{ \session('success') }}
11                     </div>
12                 </div>
13             @endif
14             @if($candidates->isNotEmpty())
15                 <h2><mark>Vote Your Selective Candidate form {{ $party_name }}</mark></h2>
16                 <form method="post" action="{{ route('vote.candidate') }}">
17                     {{ csrf_field() }}
18                 <div class="row">
19                     @foreach($candidates as $candidate)
20
21                         <div class="col-md-3">
22
23                             <div class="thumbnail">
24                                 
25                                 <label class="mainradio"><h1>{{ $candidate->candidate_name }}</h1>
26
27                                     @if(!$user->voteCast->candidate_id)
28                                         <input type="radio" value="{{ $candidate->id }}" name="candidate_id">
29                                         <span class="checkmark"></span>
30                                     @endif
31                                 </label>
32                         </div>
33                     </div>
34                 </div>
35             </div>
36         </div>
37     </div>
38 
```

Figure 91: Screenshot of coding of candidate voting page

```

1 @extends('layout.master')
2
3 @section('section')
4     <div class="voting--section">
5         <div class="container">
6             @if(\Illuminate\Support\Facades\Session::has('success'))
7                 <div style="">
8                     <div class="alert alert-success alert-dismissible">
9                         <button type="button" class="close" data-dismiss="alert" aria-hidden="true"></button>
10                    {{ \Session('success') }}
11                </div>
12            </div>
13        @endif
14    <h2><mark>Contact Us</mark></h2>
15
16    <div class="row">
17        <div class="col-md-6">
18            <div class="row">
19                <div class="col-md-6">
20                    <label>Company Email</label>
21                </div>
22                <div class="col-md-6">
23                    {{ (\App\Setting::getSettings('site_primary_email')) }}<br>
24                    {{ (\App\Setting::getSettings('site_secondary_email')) }}
25                </div>
26            </div>
27            <hr>
28        <div class="row">
29
30
31

```

Figure 92: Screenshot of coding of contact page

7.3.2 Sample Code for the Project Script

```

1 <?php
2
3 namespace App\Http\Controllers;
4
5 use Illuminate\Http\Request;
6
7 class HomeController extends Controller
8 {
9
10     /**
11      * Create a new controller instance.
12      *
13      * @return void
14      */
15     public function __construct()
16     {
17         $this->middleware('auth');
18     }
19
20     /**
21      * Show the application dashboard.
22      *
23      * @return \Illuminate\Contracts\Support\Renderable
24      */
25     public function index()
26     {
27         return view('home');
28     }
29

```

Figure 93: Home Controller

```

<?php
namespace App\Http\Controllers;
use ...;

class FrontendController extends Controller
{
    /**
     * Display a listing of the resource.
     *
     * @return \Illuminate\Http\Response
     */
    public function index()
    {
        $parties = Party::all();
        $user = Auth::user();
        return view('votingpage', compact('parties', 'user'));
    }

    /**
     * Show the form for creating a new resource.
     *
     * @return \Illuminate\Http\Response
     */
    public function create()
    {
        return view('backend.CastVote.index');
    }
}

```

\App\Http\Controllers\FrontendController.php

Figure 94: Vote party controller

```

namespace App\Http\Controllers;
use App\Candidate;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\Auth;

class VotecandidateController extends Controller
{
    /**
     * Display a listing of the resource.
     *
     * @return \Illuminate\Http\Response
     */
    public function index()
    {
        $user = Auth::user();

        if ($user->voteCast) {
            $candidates = Candidate::where('party_id', $user->voteCast->party_id)->get();
            $party_name = $user->voteCast->party->party_name;
        } else {
            $party_name = "";
            $candidates = Candidate::where('party_id', 0)->get();
        }

        return view('candidatevotingpage', compact('candidates', 'user', 'party_name'));
    }
}

```

\App\Http\Controllers\VotecandidateController.php

Figure 95: Vote candidate controller

```

<?php
namespace App\Http\Controllers;
use ...

class AccountController extends Controller
{
    /**
     * Display a listing of the resource.
     *
     * @return \Illuminate\Http\Response
     */
    public function index()
    {
        $user = Auth::user();
        $user->area = $user->location;
        $user->districtName = Location::where('id', $user->area?$user->area->parent_id:0)->first();
        $user->provinceName = Location::where('id', $user->districtName?$user->districtName->parent_id:0)->first();
        return view('account', compact('varname: 'user'));
    }

    public function history()
    {
        $user = Auth::user();

        if($user->voteCast) {
            $user->vparty = $user->voteCast->party?$user->voteCast->party:Party::find(0);
            $user->vcandidate = $user->voteCast->candidate?$user->voteCast->candidate:Candidate::find(0);
        } else {
            ...
        }
    }
}

```

Figure 96: Account controller

```

<?php
namespace App\Http\Controllers\Admin;
use ...

class AdminController extends Controller
{
    /**
     * Display a listing of the resource.
     *
     * @return \Illuminate\Http\Response
     */
    public function index()
    {
        $user = User::all();
        $parties = Party::all();
        $scandidates = Candidate::all();
        $messages = ContactUs::all();
        return view('backend.index', compact('varname: 'user', 'parties', 'candidates', 'messages'));
    }

    /**
     * Show the form for creating a new resource.
     *
     * @return \Illuminate\Http\Response
     */
    public function create()
    {
        ...
    }
}

```

Figure 97: admin controller

```

<?php

namespace App\Http\Controllers\Admin;

use ...

class AddAreaController extends Controller
{
    /**
     * Display a listing of the resource.
     *
     * @return \Illuminate\Http\Response
     */
    public function index()
    {
        $areas = Location::where('type', 'area')->get();
        return view( VIEW: 'backend.AddArea.index', compact( varname: 'areas' ));
    }

    /**
     * Show the form for creating a new resource.
     *
     * @return \Illuminate\Http\Response
     */
    public function create()
    {
        $districts = Location::where('type', 'district')->get();
        return view( VIEW: 'backend.AddArea.add', compact( varname: 'districts' ));
    }

    /**
     * Store a newly created resource in storage.
     *
     * @param \App\Location $area
     * @return \Illuminate\Http\Response
     */
    public function store(Request $request)
    {
        $validatedData = $request->validate([
            'name' => 'required|string|max:255',
            'type' => 'required|in:area,district,province',
            'parent_id' => 'nullable|exists:locations,id'
        ]);

        $area = Location::create($validatedData);

        if ($area->id) {
            return redirect()->route('backend.AddArea.index');
        } else {
            return back()->withError('Area was not created');
        }
    }
}

```

Figure 98: area controller

```

<?php

namespace App\Http\Controllers\Admin;

use ...

class AddDistrictController extends Controller
{
    /**
     * Display a listing of the resource.
     *
     * @return \Illuminate\Http\Response
     */
    public function index()
    {
        $districts = Location::where('type', 'district')->get();
        return view( VIEW: 'backend.AddDistrict.index', compact( varname: 'districts' ));
    }

    /**
     * Show the form for creating a new resource.
     *
     * @return \Illuminate\Http\Response
     */
    public function create()
    {
        $provinces = Location::where('type', 'province')->get();
        return view( VIEW: 'backend.AddDistrict.add', compact( varname: 'provinces' ));
    }

    /**
     * Store a newly created resource in storage.
     *
     * @param \App\Location $district
     * @return \Illuminate\Http\Response
     */
    public function store(Request $request)
    {
        $validatedData = $request->validate([
            'name' => 'required|string|max:255',
            'type' => 'required|in:area,district,province',
            'parent_id' => 'nullable|exists:locations,id'
        ]);

        $district = Location::create($validatedData);

        if ($district->id) {
            return redirect()->route('backend.AddDistrict.index');
        } else {
            return back()->withError('District was not created');
        }
    }
}

```

Figure 99: district controller

```

<?php
namespace App\Http\Controllers\Admin;
use ...

class AddProvinceController extends Controller
{
    /**
     * Display a listing of the resource.
     *
     * @return \Illuminate\Http\Response
     */
    public function index()
    {
        $AddProvinces = Location::where('type', 'province')->get();
        return view('backend.AddProvince.index', compact('varname: 'AddProvinces'));
    }

    /**
     * Show the form for creating a new resource.
     *
     * @return \Illuminate\Http\Response
     */
    public function create()
    {
        return view('backend.AddProvince.add');
    }

    /**
     * Store a newly created resource in storage.
     *
     * @param \App\Location $AddProvince
     * @return \Illuminate\Http\Response
     */
    public function store(AddProvince $AddProvince)
    {
        $AddProvince->save();
        return redirect()->route('AddProvince.index');
    }
}

```

Figure 100: province controller

```

<?php
namespace App\Http\Controllers\Admin;
use ...

class CandidateController extends Controller
{
    /**
     * Display a listing of the resource.
     *
     * @return \Illuminate\Http\Response
     */
    public function index()
    {
        $scandidates = Candidate::all();

        foreach($scandidates as $candidate){
            $area = $candidate->location;
            $candidate->districtName = Location::where('id', $area->parent_id)->first();
            $candidate->provinceName = Location::where('id', $candidate->districtName->parent_id)->first();
        }

        $AddProvinces = Location::all();
        return view('backend.candidate.index', compact('varname: 'candidates'));
    }

    /**
     * Show the form for creating a new resource.
     *
     * @return \Illuminate\Http\Response
     */
    public function create()
    {
        return view('backend.candidate.add');
    }

    /**
     * Store a newly created resource in storage.
     *
     * @param \App\Location $AddCandidate
     * @return \Illuminate\Http\Response
     */
    public function store(Candidate $AddCandidate)
    {
        $AddCandidate->save();
        return redirect()->route('candidate.index');
    }
}

```

Figure 101: candidate controller

```

<?php

namespace App\Http\Controllers\Admin;

use ...

class LocationController extends Controller
{
    /**
     * Display a listing of the resource.
     *
     * @return \Illuminate\Http\Response
     */
    public function index()
    {
        return view('backend.Location.index');
    }

    /**
     * Show the form for creating a new resource.
     *
     * @return \Illuminate\Http\Response
     */
    public function create()
    {
        //
    }

    /**
     * Store a newly created resource in storage.
     *
     * @param \Illuminate\Http\Request $request
     * @return \Illuminate\Http\Response
     */
}

```

Figure 102: location controller

```

<?php

namespace App\Http\Controllers\Admin;

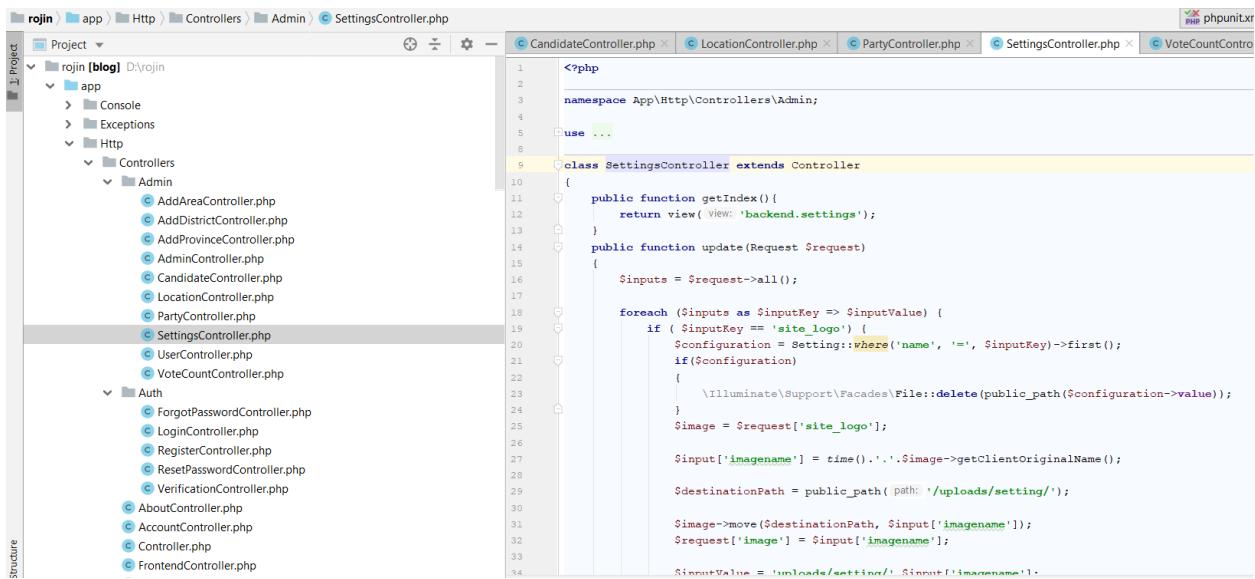
use ...

class PartyController extends Controller
{
    /**
     * Display a listing of the resource.
     *
     * @return \Illuminate\Http\Response
     */
    public function index()
    {
        $parties = Party::all();
        return view('backend.party.index', compact('parties'));
    }

    /**
     * Show the form for creating a new resource.
     *
     * @return \Illuminate\Http\Response
     */
    public function create()
    {
        return view('backend.party.add');
    }
}

```

Figure 103: party controller

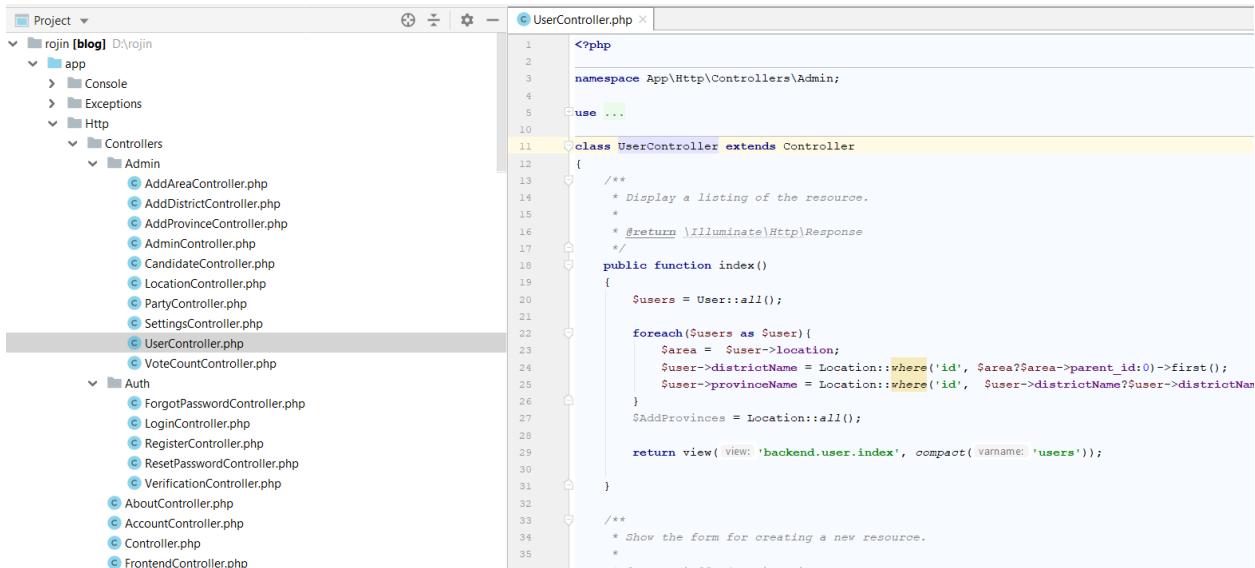


```

<?php
namespace App\Http\Controllers\Admin;
use ...
class SettingsController extends Controller
{
    public function getIndex()
    {
        return view('backend.settings');
    }
    public function update(Request $request)
    {
        $inputs = $request->all();
        foreach ($inputs as $inputKey => $inputValue) {
            if ($inputKey == 'site_logo') {
                $configuration = Setting::where('name', '=', $inputKey)->first();
                if($configuration)
                {
                    \Illuminate\Support\Facades\File::delete(public_path($configuration->value));
                }
                $image = $request['site_logo'];
                $input['imagename'] = time().'.'.$image->getClientOriginalName();
                $destinationPath = public_path(path: '/uploads/setting/');
                $image->move($destinationPath, $input['imagename']);
                $request['image'] = $input['imagename'];
                $inputValue = \Image::make($destinationPath . $input['imagename']);
            }
        }
    }
}

```

Figure 104: setting controller



```

<?php
namespace App\Http\Controllers\Admin;
use ...
class UserController extends Controller
{
    /**
     * Display a listing of the resource.
     *
     * @return \Illuminate\Http\Response
     */
    public function index()
    {
        $users = User::all();
        foreach($users as $user){
            $area = $user->location;
            $user->districtName = Location::where('id', $area? $area->parent_id:0)->first();
            $user->provinceName = Location::where('id', $user->districtName? $user->districtName->parent_id:0)->first();
        }
        $addProvinces = Location::all();
        return view( view: 'backend.user.index', compact( varname: 'users' ));
    }
    /**
     * Show the form for creating a new resource.
     *
     */
}

```

Figure 105: user controller

```

1 <?php
2
3 namespace App\Http\Controllers\Admin;
4
5 use ...
6
7 class VoteCountController extends Controller
8 {
9
10     /**
11      * Display a listing of the resource.
12      *
13      * @return \Illuminate\Http\Response
14      */
15
16     public function partyVote()
17     {
18
19         $parties= Party::all();
20
21         return view( view: 'backend.VoteCount.party', compact( varname: 'parties' ));
22     }
23
24
25     public function candidateVote()
26     {
27
28         $candidates= Candidate::all();
29
30         return view( view: 'backend.VoteCount.candidate', compact( varname: 'candidates' ));
31     }
32
33     public function report()
34     {
35
36         $castVotes = CastVote::all();
37
38         \App\Http\Controllers\Admin::VoteCountController

```

Figure 106: vote count controller

7.4 Appendix D: Designs

7.4.1 Gantt Chart

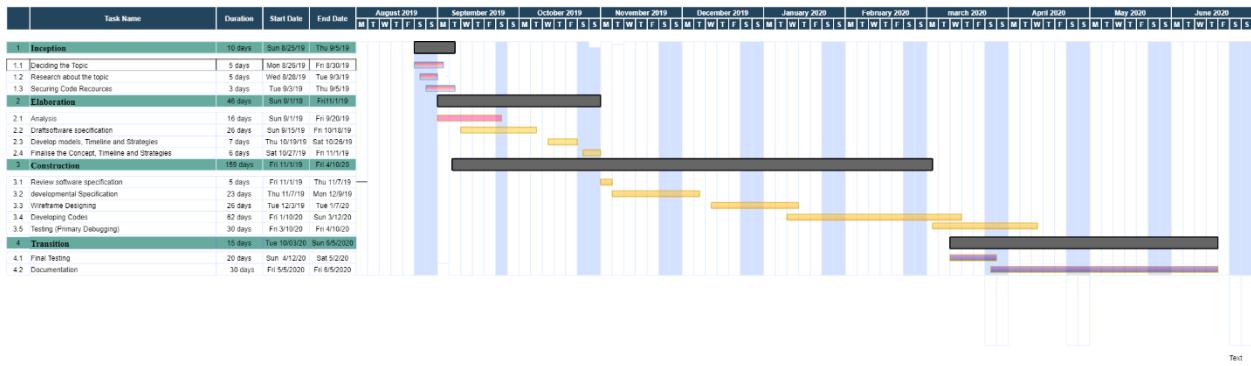


Figure 107: Gantt Chart

7.4.2 Work Breakdown Structure

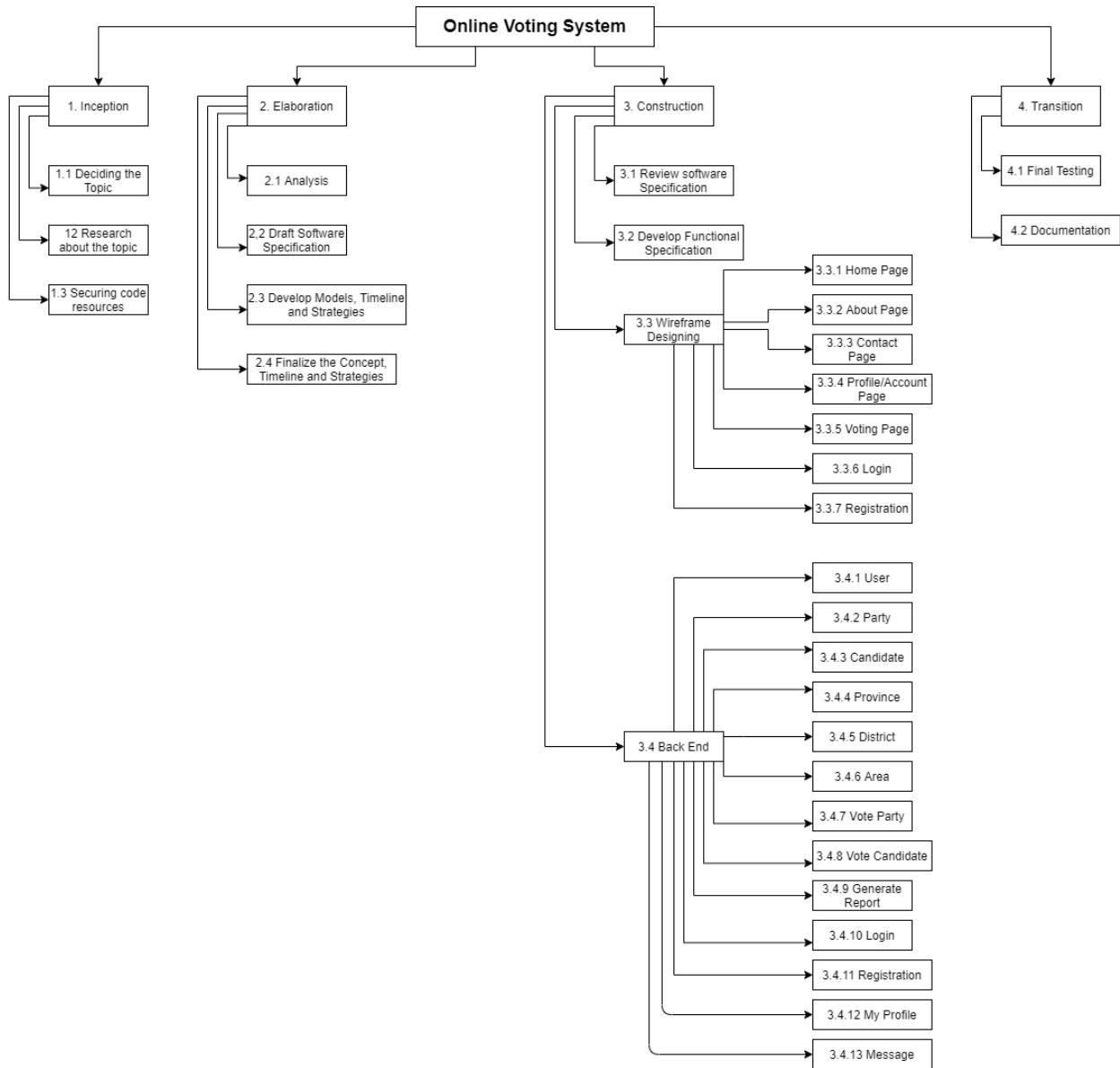


Figure 108: Work Break down Structure

7.4.3 Algorithms and Flowchart

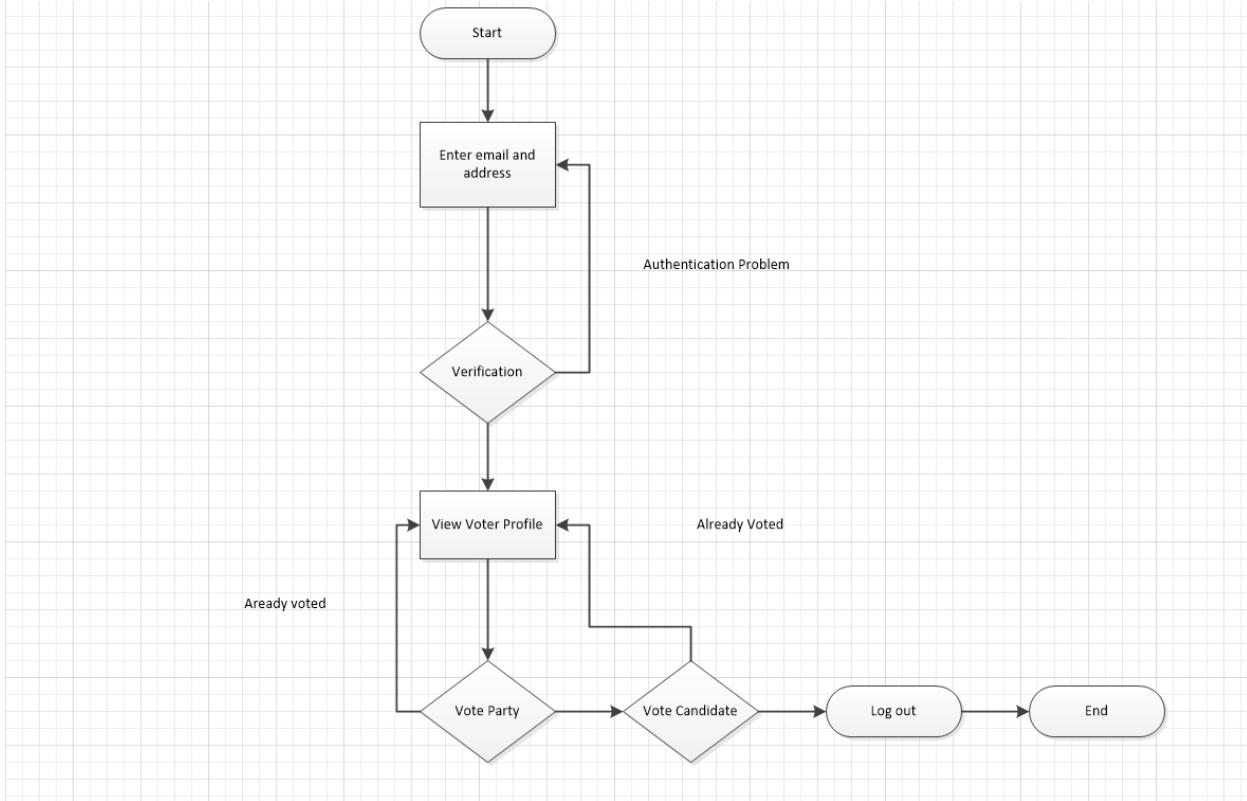


Figure 109: Flow Chart

7.4.4 Hardware Architecture

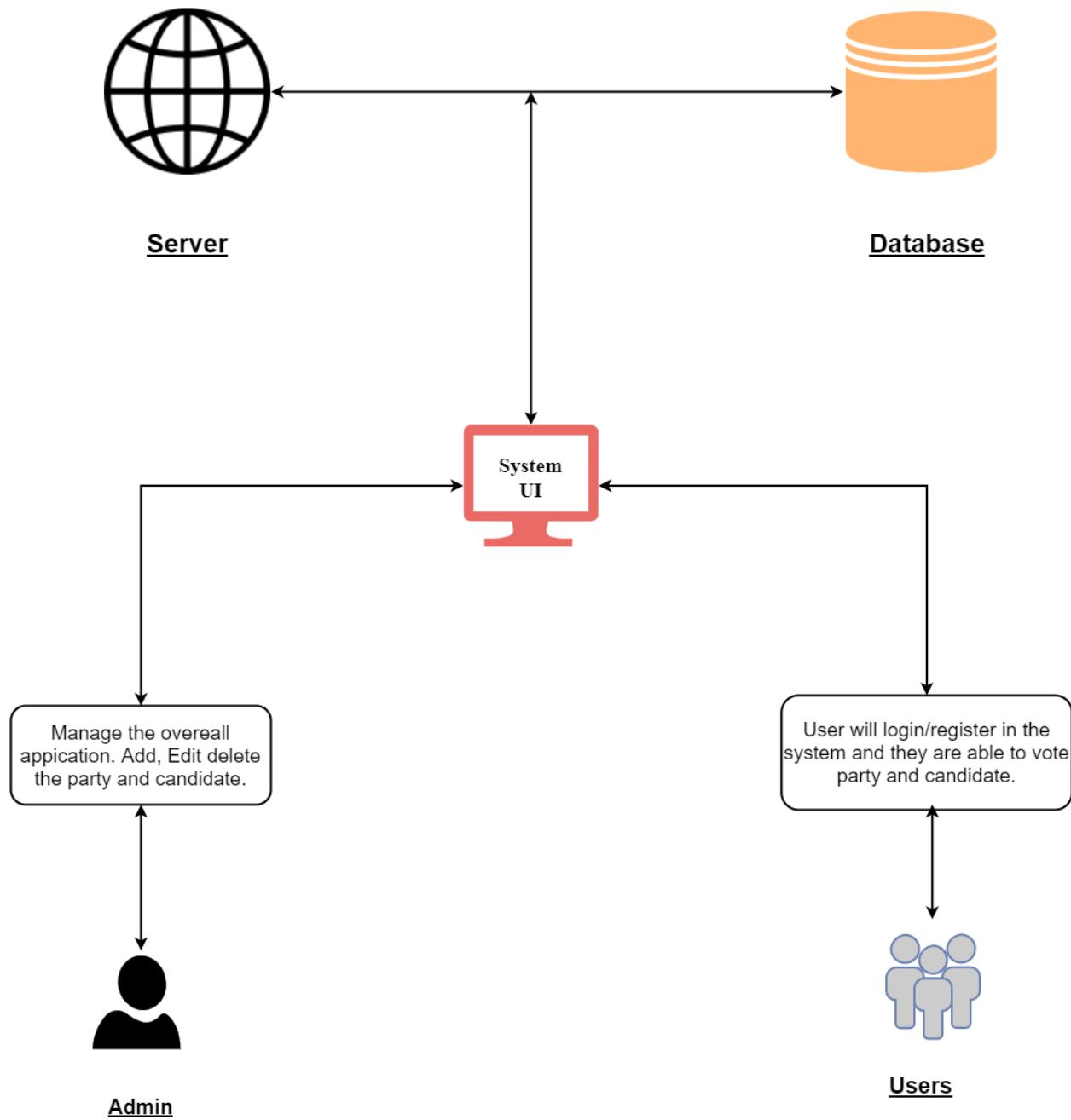


Figure 110: System Architecture

7.4.5 Data Flow Diagrams

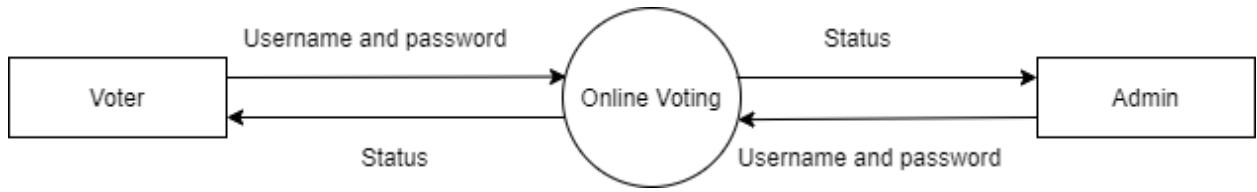


Figure 111: 0-level DFD

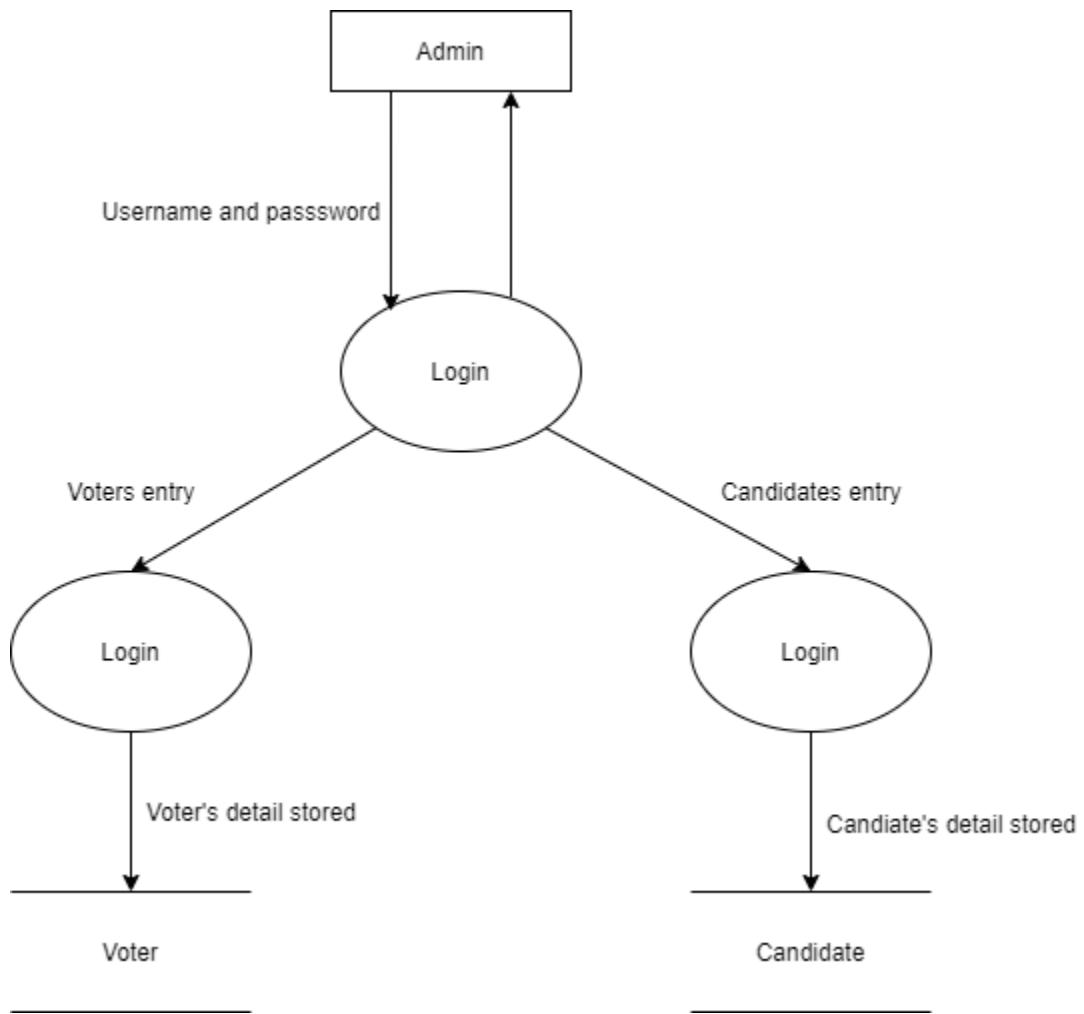


Figure 112: 1-Level DFD for admin

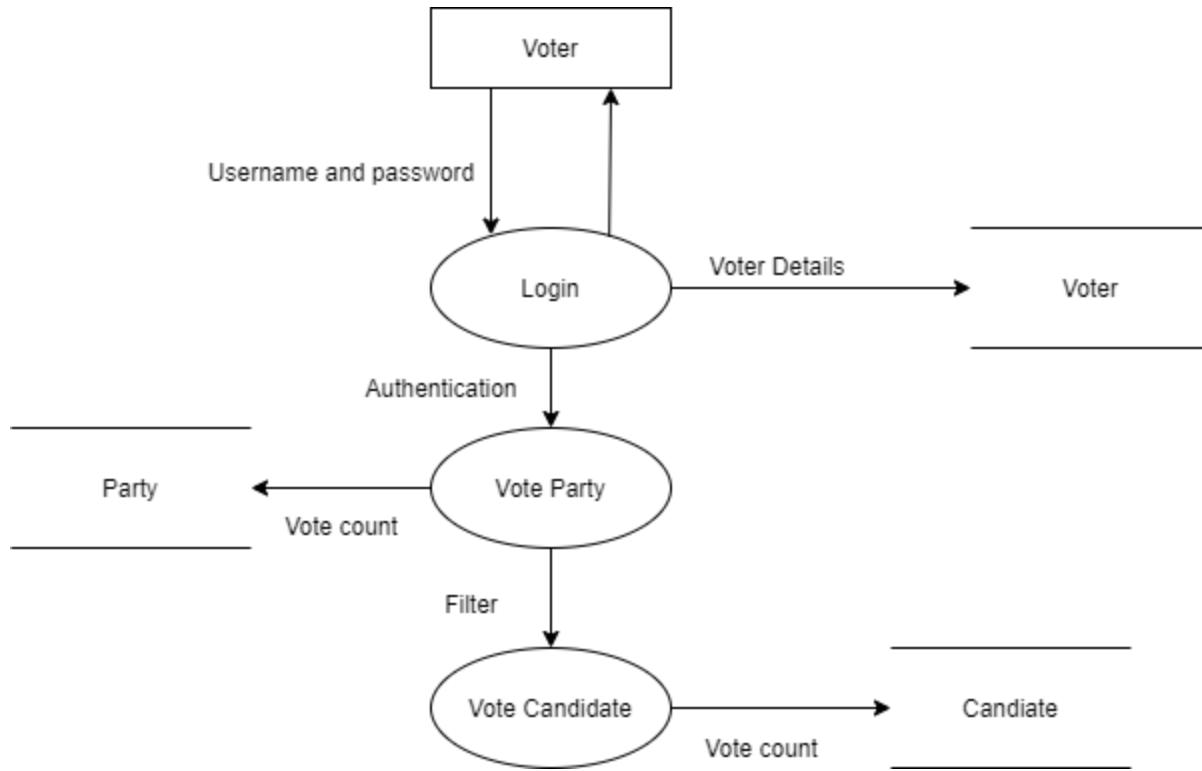


Figure 113: 1-level DFD for voters

7.4.7 Use Case

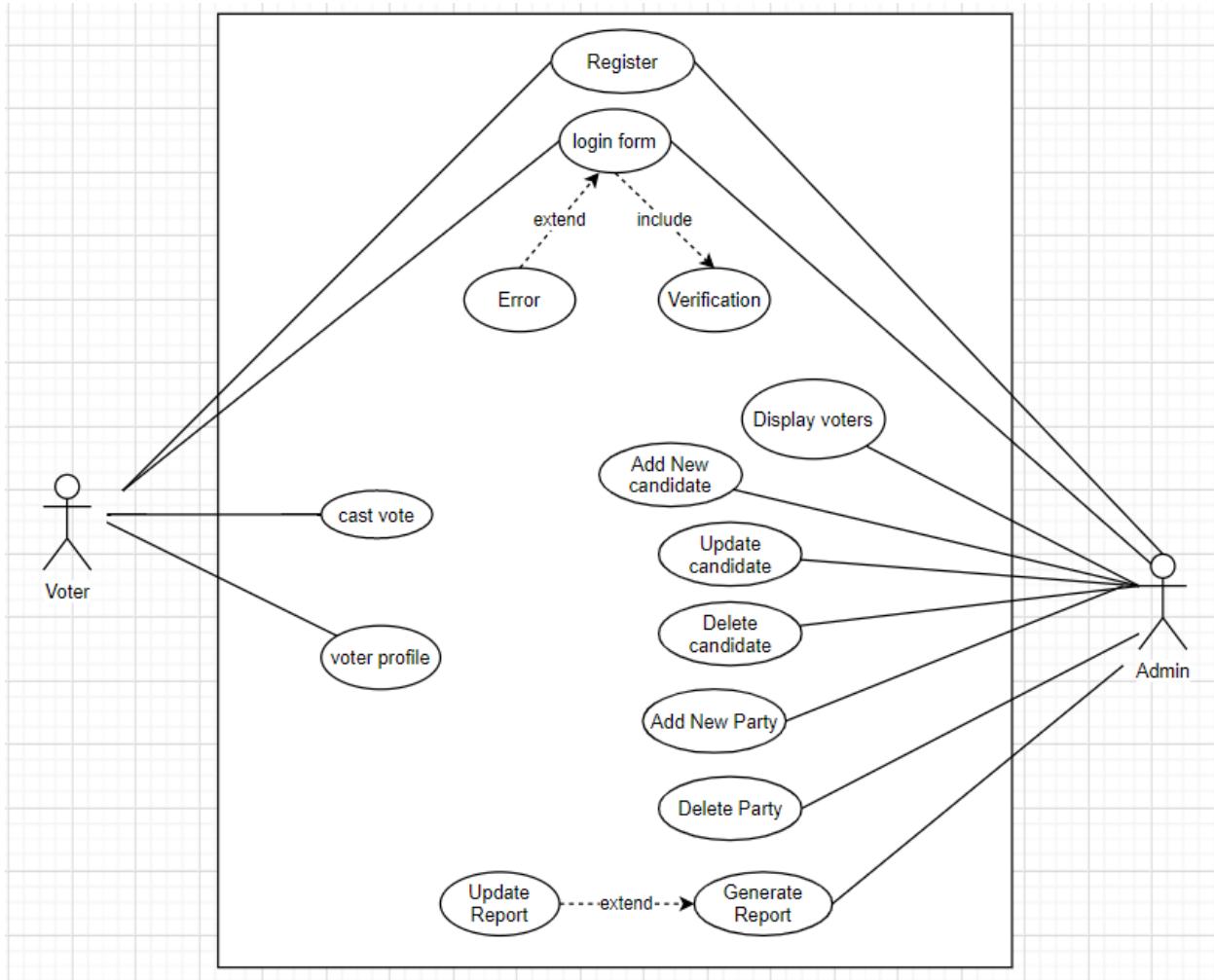


Figure 114: use case of online voting system

7.4.8 Wireframe

Login Page

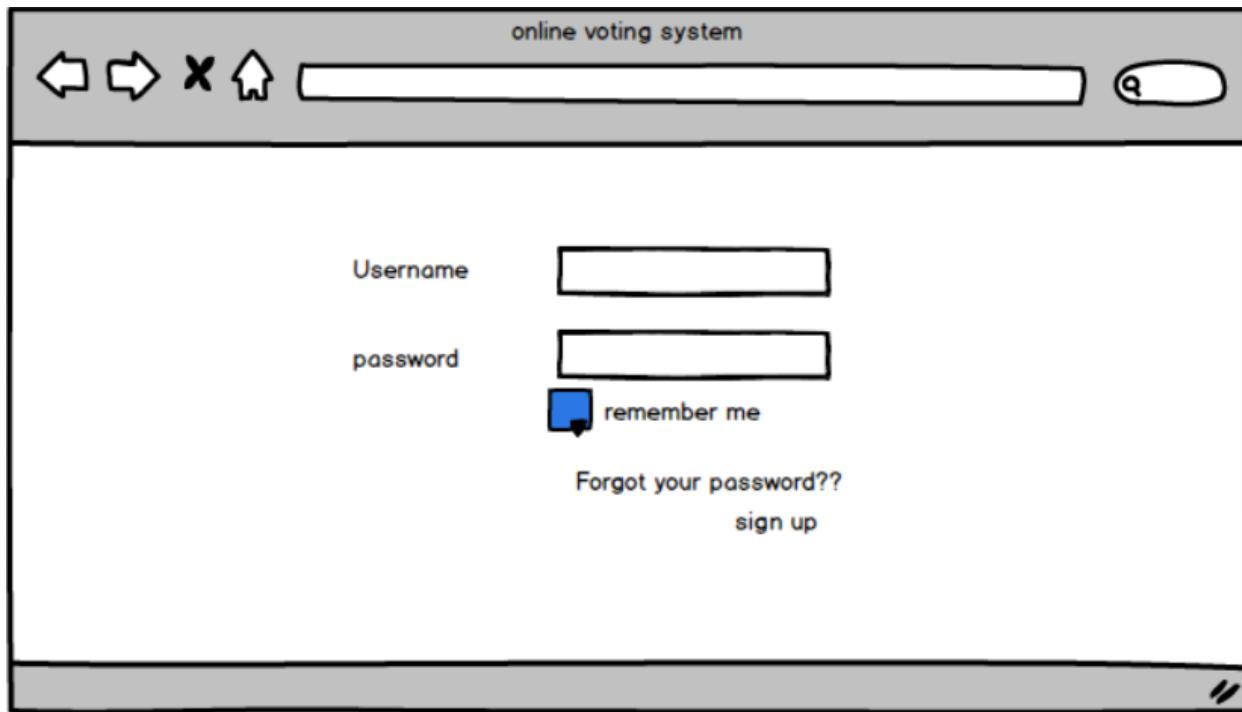


Figure 115: login page

Register Page

The diagram shows a wireframe of a web browser window. At the top, it says "online voting system". Below the title bar are standard browser controls: back, forward, stop, and refresh. To the right of the controls is a search bar with a magnifying glass icon. The main content area is a form for user registration. The fields are labeled on the left and have corresponding input boxes on the right. The fields are: first name, last name, email, password, confirm password, province, district, area, phone number, citizenship number, gender, and image. At the bottom of the form is a large, prominent "Register" button.

Figure 116: Register page

Home Page

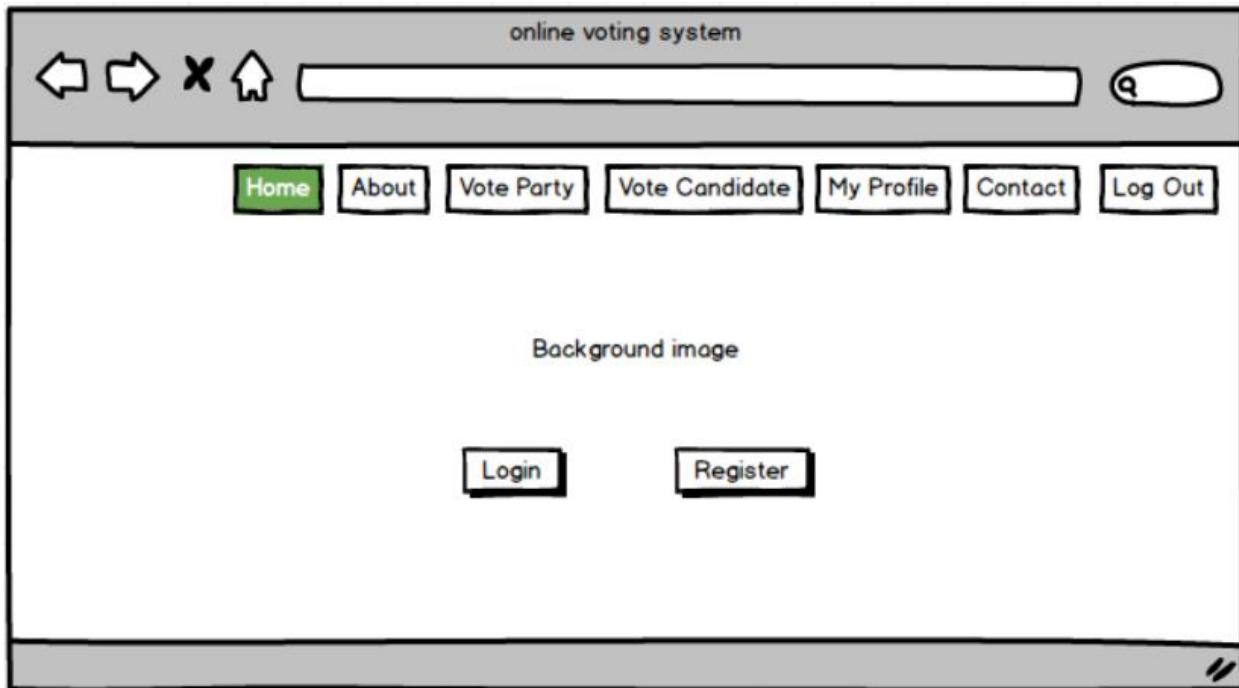
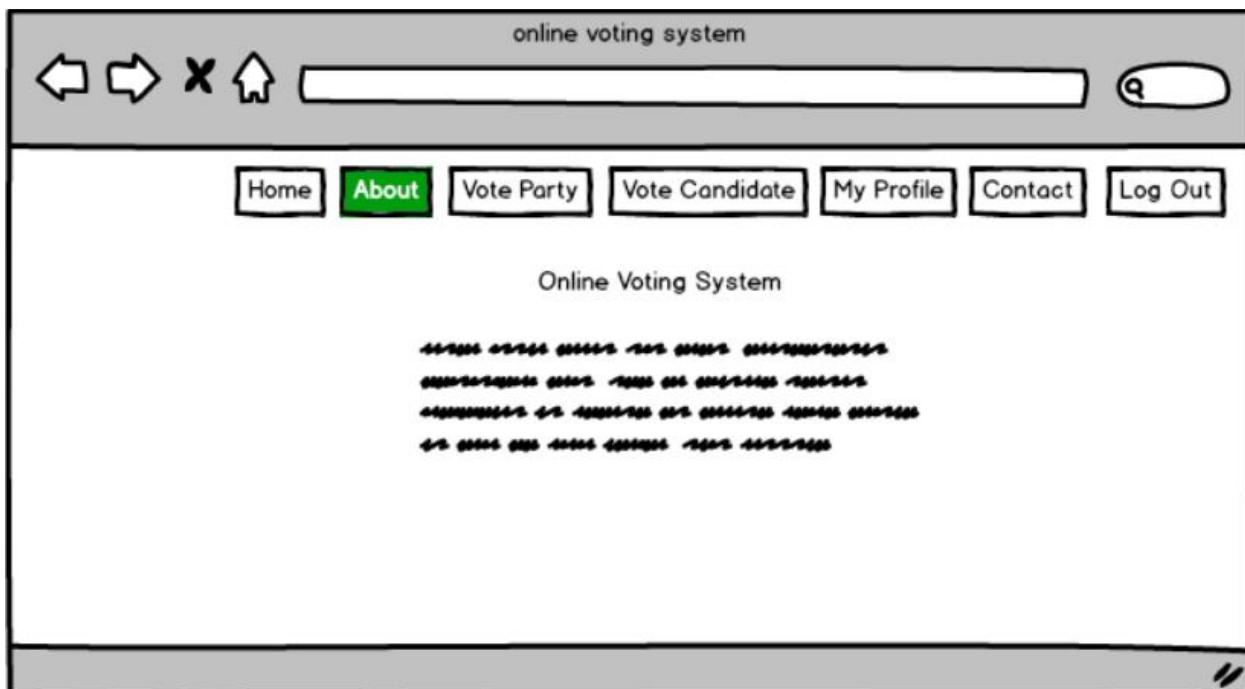


Figure 117: home page

About page*Figure 118: About page*

Vote Party Page

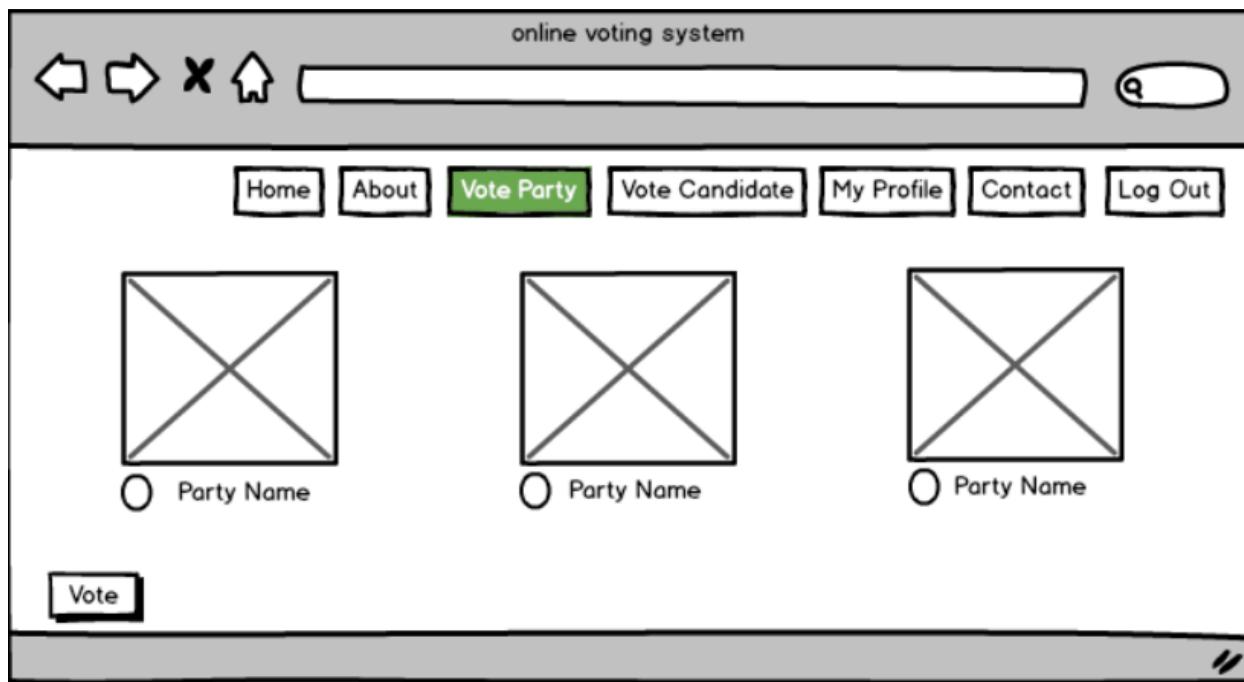


Figure 119: vote party page

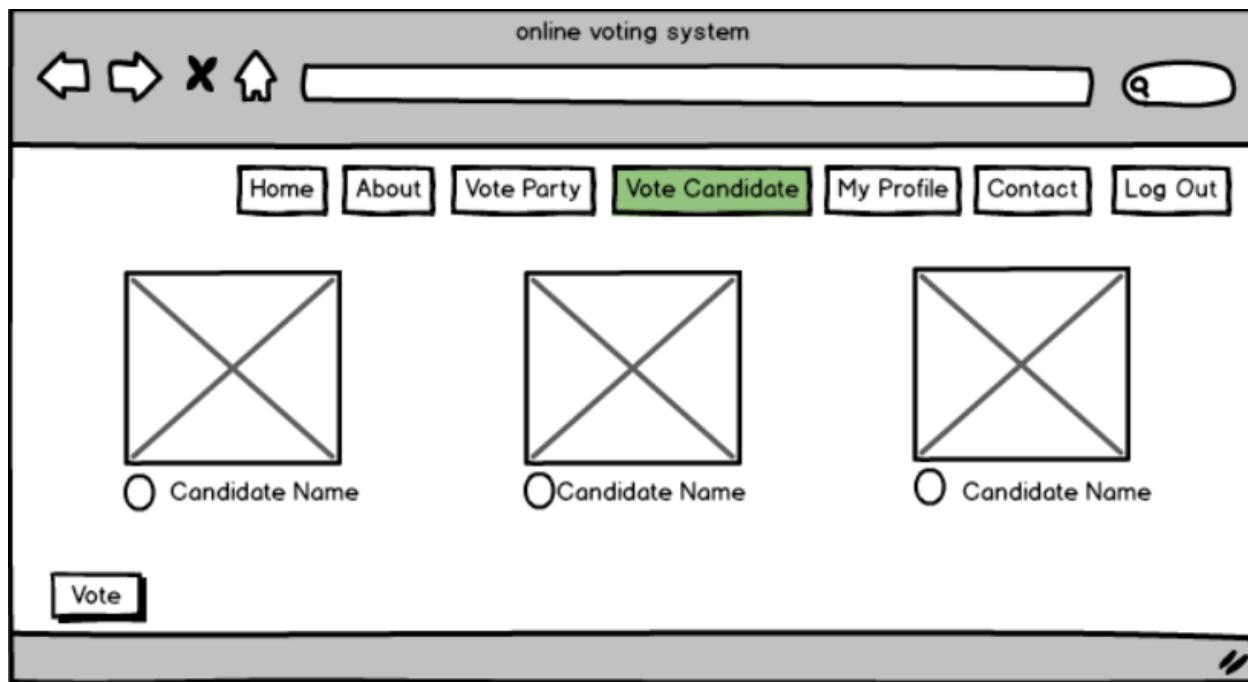
Vote Candidate page

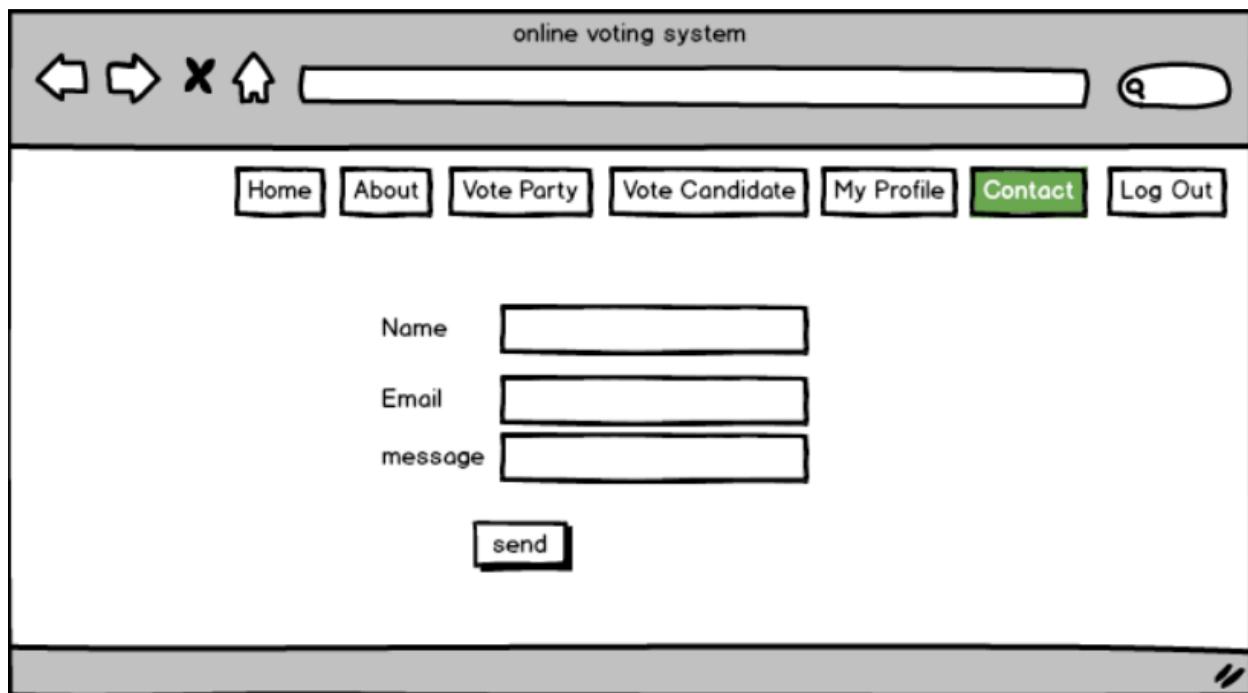
Figure 120: vote candidate page

My Profile page

The screenshot shows a web-based application interface for managing a user profile. At the top, there's a header bar with icons for back, forward, and search, followed by the text "online voting system". Below the header is a navigation menu with buttons for Home, About, Vote Party, Vote Candidate, My Profile (which is highlighted in green), Contact, and Log Out. To the left of the main content area, there's a placeholder image with a large 'X' and a "User name" input field below it. A "Change" button is positioned just below the "User name" field. On the right side, there are five input fields labeled Name, Email, phone, Address, and password, each with a corresponding empty rectangular box. At the bottom right of the form area is a "Update" button.

Figure 121: My profile page

Contact Page



7.5 Appendix E: Screenshots of the system

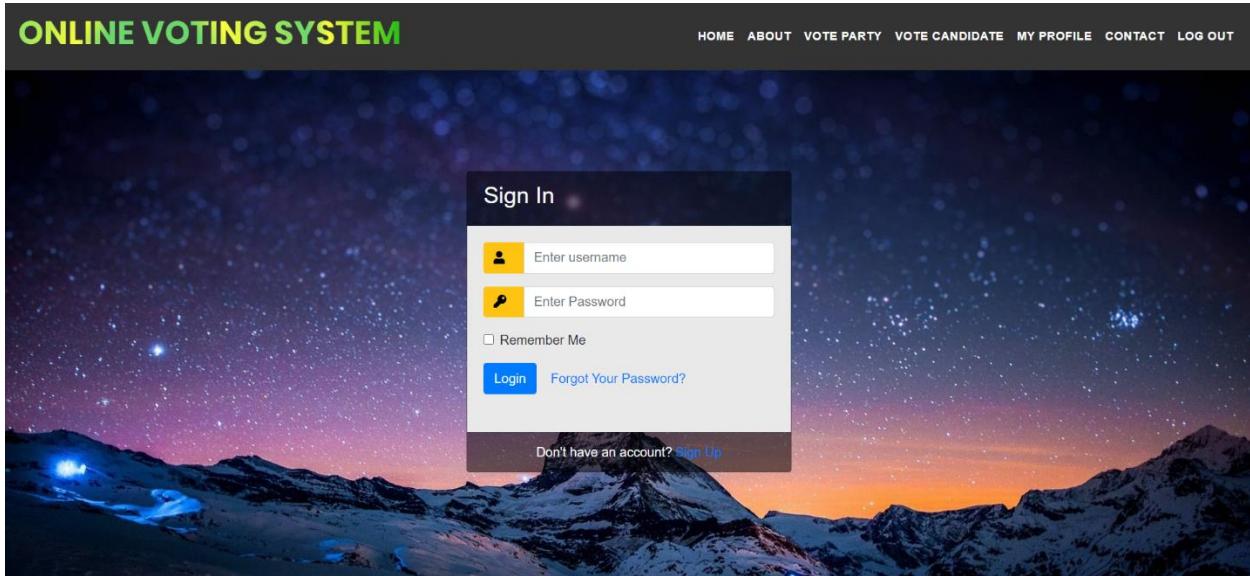


Figure 122: Login Page

The screenshot shows the 'Register' page of the 'ONLINE VOTING SYSTEM'. The page has a light gray background with a white registration form in the center. At the top right of the form, the word 'Register' is displayed in green. The form contains the following fields:

- First Name: A text input field with placeholder text 'Enter First Name'.
- Last Name: A text input field with placeholder text 'Enter Last Name'.
- Email: A text input field with placeholder text 'Enter Email'.
- Password: A text input field with placeholder text 'Enter Password'.
- Confirm Password: A text input field with placeholder text 'Enter Password'.
- Province: A dropdown menu set to 'Bagmati'.
- Phone Number: A text input field with placeholder text 'Enter phone number'.
- Citizenship Number: A text input field with placeholder text 'Enter citizenship number'.
- Gender: A dropdown menu set to 'Male'.
- Profile Image: A file upload section with 'Choose file' button, 'Browse' button, and 'Upload' button.

At the bottom of the form is a large, dark gray 'REGISTER' button.

Figure 123: Register Page

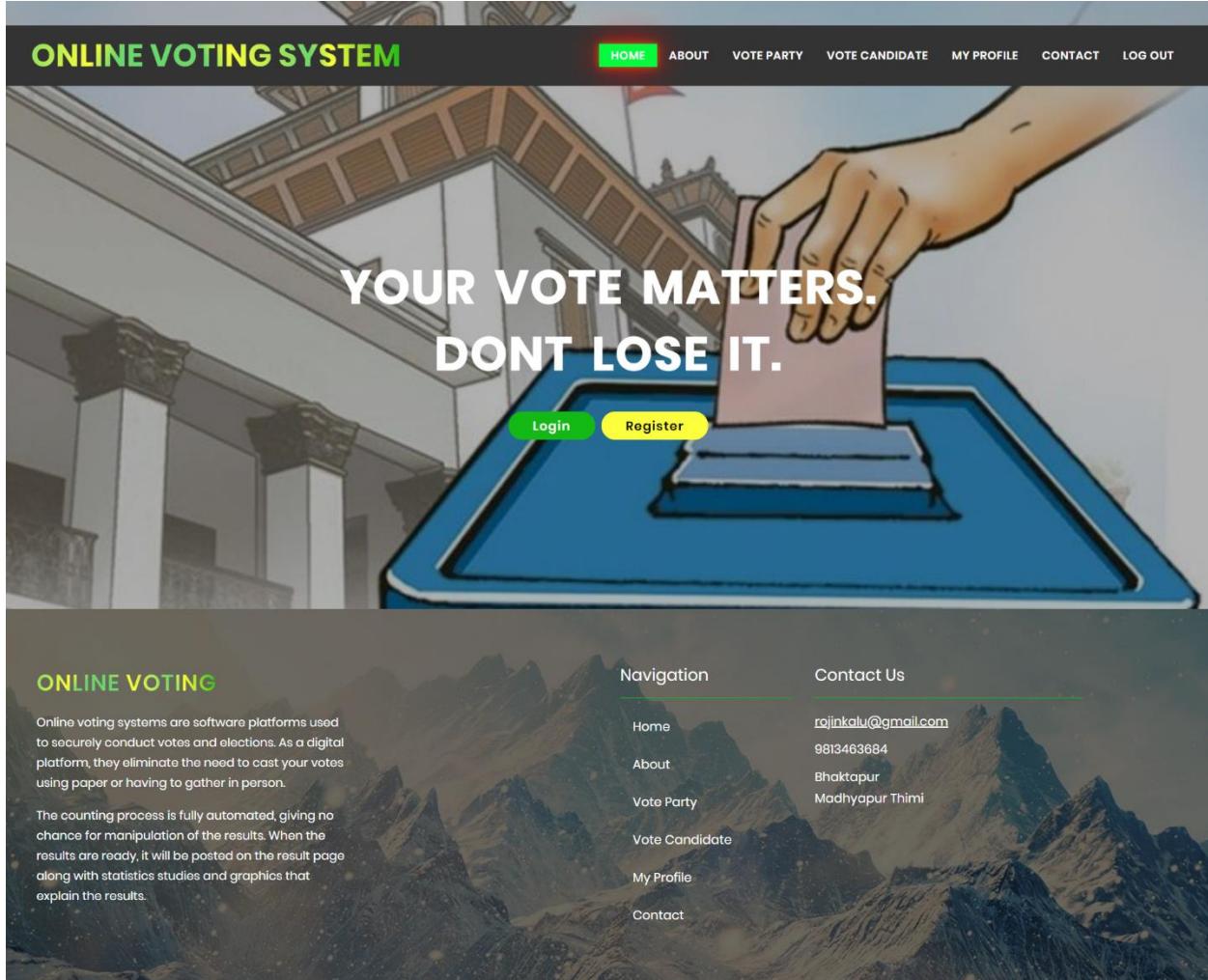


Figure 124: Home page

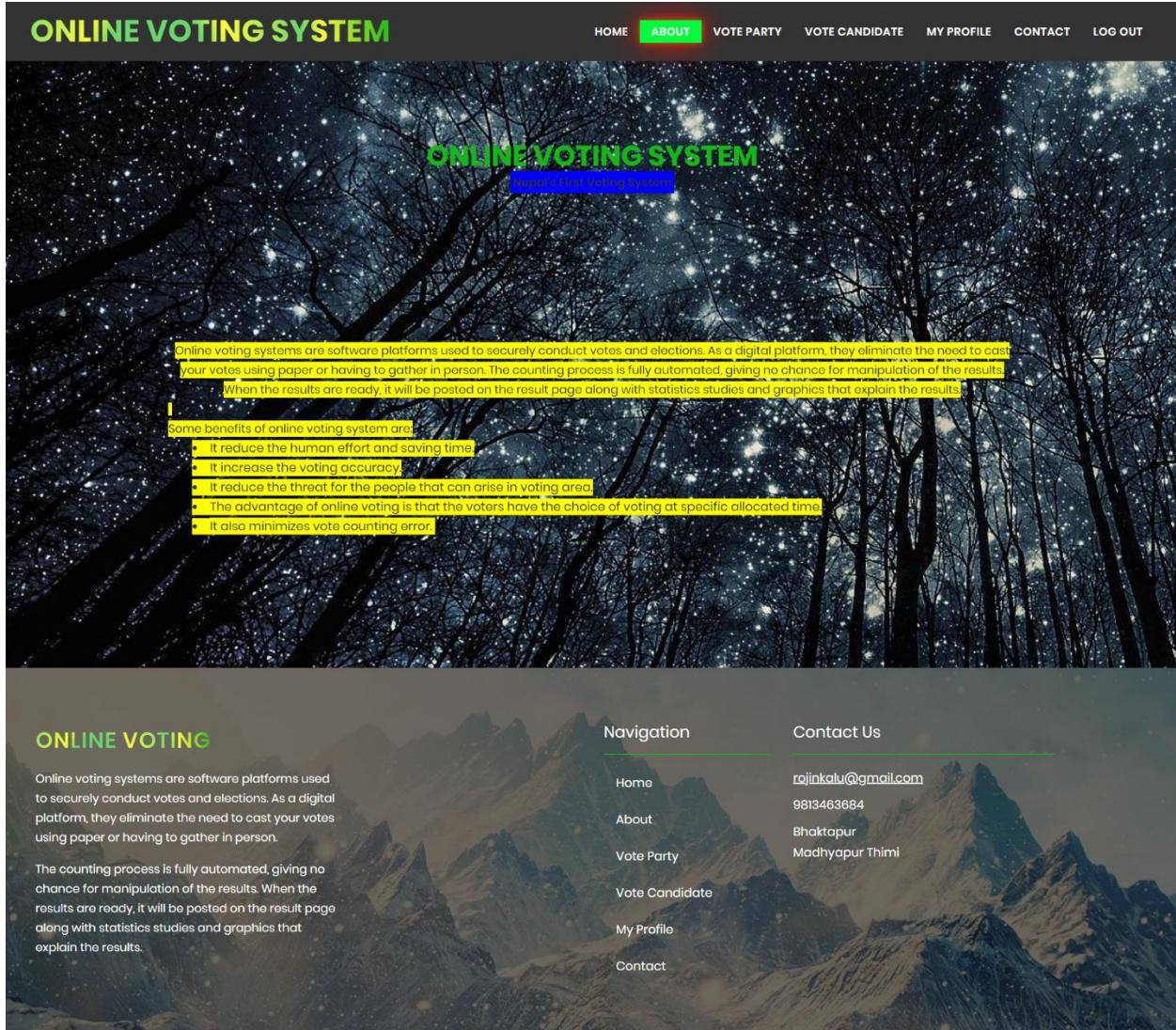


Figure 125: About page

The screenshot shows the 'VOTE PARTY' section of the Online Voting System. At the top, there is a navigation bar with links: HOME, ABOUT, VOTE PARTY (highlighted in red), VOTE CANDIDATE, MY PROFILE, CONTACT, and LOG OUT. Below the navigation bar, a yellow button says 'Vote Your Selective Party'. There are four boxes representing political parties:

- Congress**: Shows a green tree icon.
- Communist**: Shows a red flag with a hammer and sickle icon.
- Yamale**: Shows a red sunburst logo with the text 'YAMALI' and a hammer and sickle symbol.
- Samajbadi**: Shows a hand holding a torch icon.

Below these boxes, there is a link 'You Vote for Yamale' and a blue link 'View Candidate'. Further down, there is a box for the **Rastriya Prajatanta Party** featuring its flag colors (red, yellow, blue) and a lion emblem.

ONLINE VOTING

Online voting systems are software platforms used to securely conduct votes and elections. As a digital platform, they eliminate the need to cast your votes using paper or having to gather in person.

The counting process is fully automated, giving no chance for manipulation of the results. When the results are ready, it will be posted on the result page along with statistics studies and graphics that explain the results.

Navigation

- Home
- About
- Vote Party
- Vote Candidate
- My Profile
- Contact

Contact Us

- rajinikalu@gmail.com
- 9813463884
- Bhaktapur
- Madhyapur Thimi

Figure 126: Vote Party Page

The screenshot shows a web application interface for an online voting system. At the top, there is a navigation bar with links: HOME, ABOUT, VOTE PARTY, VOTE CANDIDATE (which is highlighted in red), MY PROFILE, CONTACT, and LOG OUT. Below the navigation bar, the title "ONLINE VOTING SYSTEM" is displayed in green. A sub-header "Vote Your Selective Candidate from Yamale" is centered above a grid of eight candidate portraits. The candidates are arranged in two rows of four. The first row includes Henry Cavill, Ben Affleck, Jason Momoa, and Gal Gadot. The second row includes Anne Hathaway, Chris Pine, and Berry Allan. Each portrait has a caption below it. At the bottom of the page, there is a footer section with a background image of mountains. The footer contains sections for "ONLINE VOTING", "Navigation" (with links to Home, About, Vote Party, Vote Candidate, My Profile, and Contact), and "Contact Us" (with email address rojinkalu@gmail.com, phone number 9813463684, and location Bhaktapur, Madhyapur Thimi).

ONLINE VOTING SYSTEM

HOME ABOUT VOTE PARTY **VOTE CANDIDATE** MY PROFILE CONTACT LOG OUT

Vote Your Selective Candidate from Yamale

Henry Cavill
You Vote for Henry Cavill

Ben Affleck

Jason Momoa

Gal Gadot

Anne Hathaway

Chris Pine

Berry Allan

ONLINE VOTING

Online voting systems are software platforms used to securely conduct votes and elections. As a digital platform, they eliminate the need to cast your votes using paper or having to gather in person.

The counting process is fully automated, giving no chance for manipulation of the results. When the results are ready, it will be posted on the result page along with statistics studies and graphics that explain the results.

Navigation

- [Home](#)
- [About](#)
- [Vote Party](#)
- [Vote Candidate](#)
- [My Profile](#)
- [Contact](#)

Contact Us

rojinkalu@gmail.com
9813463684
Bhaktapur
Madhyapur Thimi

Figure 127: Vote Candidate Page

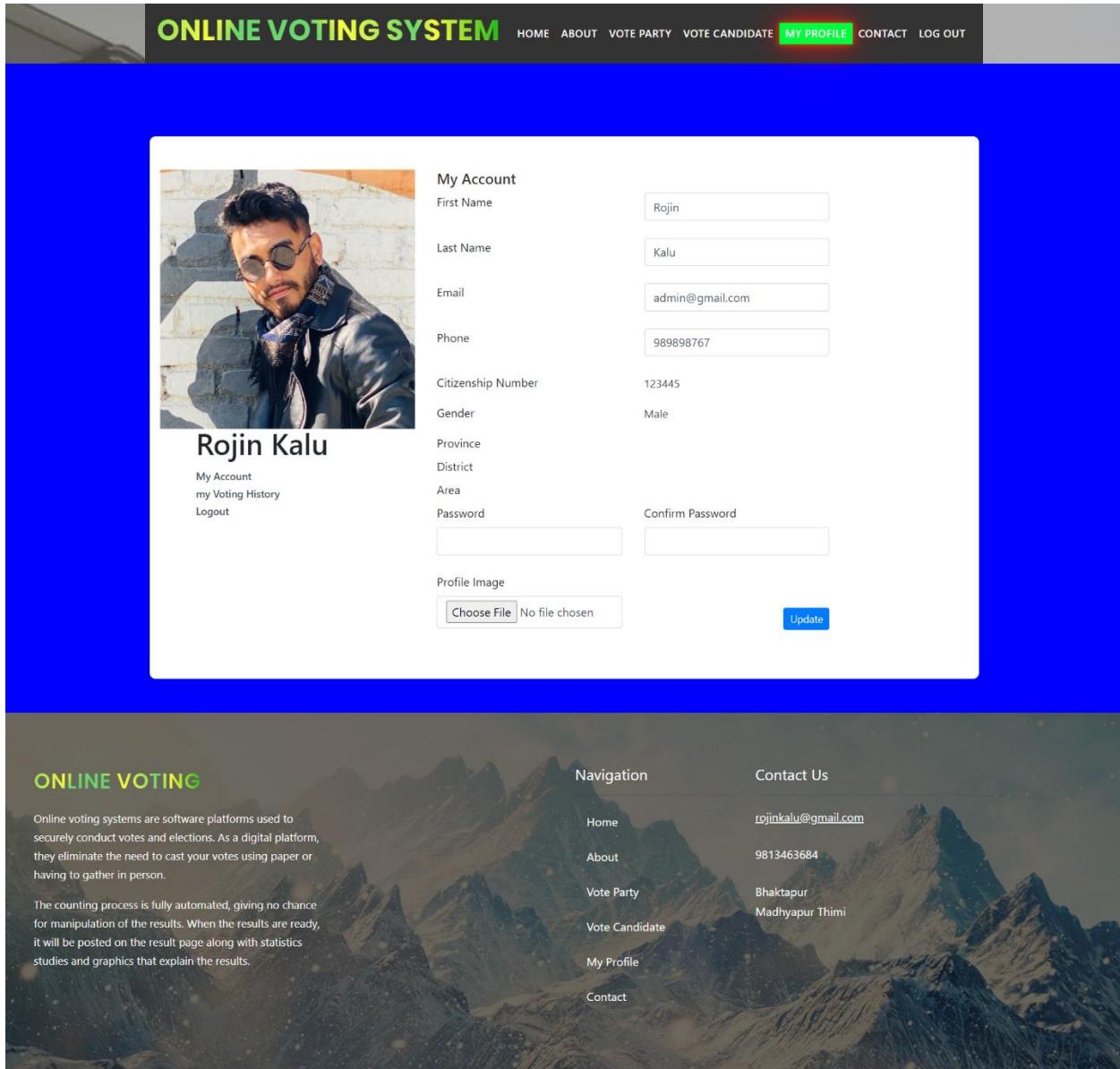


Figure 128: My Profile Page

ONLINE VOTING SYSTEM

HOME ABOUT VOTE PARTY VOTE CANDIDATE MY PROFILE **CONTACT** LOG OUT

Contact Us

Company Email	rojinkalu@gmail.com rojinkalu23@gmail.com	First Name	<input type="text"/>
Company Phone	9813463684 rojinkalu23@gmail.com	Last Name	<input type="text"/>
Company Address	Madhyapur Thimi, Nepal	Email	<input type="text"/>
		subject	<input type="text"/>
Message <input type="text"/>			

Send

ONLINE VOTING

Online voting systems are software platforms used to securely conduct votes and elections. As a digital platform, they eliminate the need to cast your votes using paper or having to gather in person.

The counting process is fully automated, giving no chance for manipulation of the results. When the results are ready, it will be posted on the result page along with statistics studies and graphics that explain the results.

Navigation

- [Home](#)
- [About](#)
- [Vote Party](#)
- [Vote Candidate](#)
- [My Profile](#)
- [Contact](#)

Contact Us

rojinkalu@gmail.com
9813463684
Bhaktapur
Madhyapur Thimi

Figure 129: contact page

7.6 Appendix F: User Feedback

7.6.1 User Feedback Form

The screenshot shows a Google Forms survey titled "Online Voting System". The survey consists of the following questions:

- Email Address:** A short answer text input field.
- Full Name:** A short answer text input field.
- Do you like the overall features of online voting system?** A radio button choice between "yes" and "No".
- Do you think design is user friendly?** A radio button choice between "Yes" and "No".
- Do you found the voting process easy or difficult?** A radio button choice between "Easy" and "Difficult".
- How would you rate the prototype?** A rating scale from 1 to 5.
- Would you go on manual voting or online voting?** A radio button choice between "Manual Voting" and "Online Voting".
- Feedback?** A long answer text input field.

The form interface includes a toolbar at the top with icons for file operations, a "Send" button, and a red circular button. On the right side, there are three vertical columns of icons for modifying the form structure.

Figure 130: User feedback form

7.6.2 Sample of filled User Feedback forms

Responses 5

Accepting responses

Summary Question Individual

1 of 5

Responses cannot be edited

Online Voting System

Hello!
I would be grateful if you could fill this survey form to share your opinion after using online voting system.

Email Address
bishalkauri@email.com

Full Name
Bishal Kauri

Do you like the overall features of online voting system?
 yes
 No

Do you think design is user friendly?
 Yes
 No

Do you found the voting process easy or difficult?
 Easy
 Difficult

How would you rate the prototype?
 1 2 3 4 5

Would you go on manual voting or online voting?
 Manual Voting
 Online Voting

Feedback?
Register process was difficult

Submitted 6/5/20, 10:50 AM

Figure 131: Sample of filled User Feedback forms

7.7 Appendix G: Future Work

7.7.1 Readings for Future Work

Here, any project or any application cannot be fully completed. An extra feature can be added in any system. Here the final report could be displayed in map according to province, district or local area. Higher security could be implemented in this system as there is higher risks of hacking the online voting system.