VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"Jnana Sangama", Belgavi-590 018, Karnataka, India

A

Project Report

On

"E-Voting Using Blockchain Technology"

Submitted in Partial Fulfillment of the requirements for the award of the degree of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

Submitted By

NANDINI L REDDY	1SJ18CS062
POOLA BALAJI	1SJ18CS073
PUTTAPARTHI THARUN SAI	1SJ18CS076
SUBHASH K V	1SJ18CS101

Carried out at
B G S R&D Centre,
Dept of CSE, SJCIT

Under the guidance of

Dr. Vikas Reddy S BE, MS, PHD

Associate Professor

S J C INSTITUTE OF TECHNOLOGY

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CHIKKABALLAPUR-562 101

2021-2022

||Jai Sri Gurudev||

Sri Adichunchanagiri Shikshana Trust®

S J C INSTITUTE OF TECHNOLOGY.CHICKBALLAPUR – 562101

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the project work entitled **"E-Voting using BlockChain Technology"** is a bonafied work carried out by **Nandini L Reddy (1SJ18CS062), Poola Balaji(1SJ18CS073), Puttaparthi Tharun Sai(1SJ18CS076), Subhash K V (1SJ18CS101)** in partial fulfillment for the award of **Bachelor of Engineering in Computer Science and Engineering in Eighth Semester of the Visvesvaraya Technological University, Belagavi** during the year **2021-2022.** It is certified that all corrections/suggestions indicated for internal assessment have been incorporated in the report. The project report has been approved as it satisfies the academic requirements with respect to eighth semester project work prescribed for the above said degree.

12/07/2022

Signature of Guide
Dr. Vikas Reddy S
Associate Professor
Dept. of CSE, SJCIT

5/12/22

Signature of HOD
Dr. Manjunatha Kumar BH
Professor & HOD
Dept. of CSE, SJCIT

Professor & HOD,
Department of Computer Science & Engg.,
S.J.C. Institute of Technology,
Chickballapur-562 101

Signature of Principal
Dr. G T Raju
Principal, SJCIT,

Principal
S J C Institute of Technology
Chickballapur - 562 101

External Examiners:

Name of the Examiners | **Signature with date**

1. pradeepkumar SM

2. Dr. Murthy SVN

DECLARATION

We Nandini L Reddy (1SJ18CS062), Poola Balaji (1SJ18CS073), Puttaparthi Tharun Sai(1SJ18CS076), Subhash K V (1SJ18CS101) Student of VIII semester B.E in Computer Science and Engineering at S J C Institute of Technology, Chickballapur, hereby declare that this dissertation work entitled "E-voting using Blockchain Technology" has been carried out at B.G.S R&D Centre, Dept. of CSE, SJCIT under the guidance of guide [Dr. Vikas Reddy S, Associate Professor], Dept. of CSE, SJC Institute of Technology, Chickballapur and submitted in the partial fulfilment for the award of degree Bachelor of Engineering in Computer Science and Engineering of Visvesvaraya Technological University, Belagavi during the academic year 2021-2022. We further declare that the report had not been submitted to another university for the award of any other degree.

Place: Chikkaballapur

Date:

NANDINI L REDDY

1SJ18CS062

POOLA BALAJI

1SJ18CS073

PUTTAPARTHI THARUN SAI

1SJ18CS076

SUBHASH K V

1SJ18CS101

ABSTRACT

The electronic voting has emerged over time as a replacement to the paper-based and electronic machine voting to reduce the redundancies and inconsistencies. The historical perspective presented in the last two decades suggests that it has not been so successful dueto the security and privacy flaws observed over time. This project suggests a framework byusing effective hashing techniques to ensure the security of the data. The concept of blockcreation and block sealing is introduced in this project. The introduction of a block sealinghelps in making the blockchain adjustable to meet the need of the polling process. The useof consortium blockchain is suggested, which ensures that the blockchain is owned by a governing body (e.g., election commission), and no un-authorized access can be made fromoutside. The framework proposed in this project discusses the effectiveness of the polling process, hashing algorithms utility, block creation and sealing, data accumulation, and result declaration by using the adjustable blockchain method. this project claims to apprehend the security and data management challenges in blockchain and provides an improved manifestation of the electronic voting system.

ACKNOWLEDGEMENT

With reverential pranam, we express our sincere gratitude and salutations to the feetof his holiness **Paramapoojya Jagadguru Byravaikya Padmabhushana Sri Sri Sri Dr. Balagangadharanatha Maha Swamiji**, his holiness **Paramapoojya JagadguruSri Sri Sri Dr. Nirmalanandanatha Maha Swamiji** and **Pramapoojya Sri Sri Mangalnatha Swamiji**, Sri Adichunchanagiri Mutt for their unlimited blessings.

First and foremost, we wish to express our deep sincere feelings of gratitude to our institution, **Sri Jagadguru Chandrashekaranatha Swamiji Institute of Technology**, for providing us an opportunity for completing our Project Work successfully.

We extend deep sense of sincere gratitude to **Dr. G T Raju**, **Principal, S J C Institute of Technology, Chickballapur**, for providing an opportunity to complete the Project Work.

We extend special in-depth, heartfelt, and sincere gratitude to HOD **Dr. Manjunatha Kumar B H, Head of the Department, Computer Science andEngineering, S J C Institute of Technology, Chickballapur,** for his constant support andvaluable guidance of the Project Work.

We convey our sincere thanks to Project Guide **Dr.Vikas Reddy S, Associate Professor, Department of Computer Science and Engineering, S J C Institute of Technology,** for his constant support, valuable guidance and suggestions of the Project Work.

We also feel immense pleasure to express deep and profound gratitude to Project coordinators **Prof. PradeepKumar G M and Prof Shrihari M R, Assistant Professors, Department of Computer Science and Engineering, S J C Institute of Technology,** fortheir guidance and suggestions of the Project Work.

Finally, we would like to thank all faculty members of Department of Computer Science and Engineering, S J C Institute of Technology, Chickballapur for their support.

We also thank all those who extended their support and co-operation while bringingout this Project work report.

Nandini L Reddy(1SJ18CS062)

Poola Balaji(1SJ18CS073)

Puttaparthi Tharun Sai(1SJ18CS076)

Subhash K V (1SJ18CS101)

CONTENTS

LIST OF FIGURES

LIST OF TABELS

CHAPTER 1
INTRODUCTION

CHAPTER - 1

INTRODUCTION

1.1 Overview

Will of the people is a well-respected phenomenon for representation of opinion information of electoral bodies. These electoral bodies vary from the college unions to the parliaments. Over the years, 'vote 'has emerged as a tool for representing the will of the people when a selection is to be made among the available choices. The voting tool has helped improving the trust of people over the selection they make by a vote of majority. This has certainly helped in democratization of the voting process and the value of voting system to elect the parliaments and governments. In 2018, there are 167 counties out of little over 200 who have some kind of democracy; full, flawed, or hybrid etc. Since the trustof people is increasing in democracies it is important that they don't lose their trust on voteand voting system. By virtue of the emerging trust on the democratic institutions, the votingsystem emerged as a platform to help people to elect their representatives, who consequently form the governments. The power of representation empowers the people with a trust that the government shall take care of the national security, national issues likehealth and education policies, international relations, and taxation for the benefit of the people.

In order to make the voting process more effective the institutions like 'Election Commission' came into existence in different parliamentary democracies. The institutions,along with setting up the process and legislation for conducting the elections, formed the voting districts, electoral process, and the balloting systems to help in conduct of transparent, free, and fair elections. The concept of secret voting was introduced since the beginning of the voting system. Since the trust on democratic systems is increasing it is important to uphold that the trust on voting should not decrease. In the recent past there have been several examples where it was noted that the voting process was not completely hygienic and faced several issues including transparency and fairness, and the will of peoplewas not observed to be effectively quantified and translated in terms of formation of the governments. Such examples can be vastly found in countries like Nigeria, India, Brazil, Pakistan, and Bangladesh.

To resolve issues, technology comes in existence where people can cast vote from where ever they are and securely cast their vote without any influence from the external sources. By taking this present issue of voting, this project is to implement secure, efficient and burden-free voting using blockchain.

1.2 Problem Statement

As India is the second most populated country with 63.6% adults (who have the right to cast their vote), but the percentage of people voted is approximately 61% to 74%, ie only 18,27,936 has casted their vote out of 138 crore (2018 election). Among all of the reason for not participating in voting campaign, the most important one is the traditional method of voting is not convenient like, transportation is the main reason for not casting their vote.

The system is developed to raise the percentage of people voting, and to design user friendly system.

1.3 Significance and Relevance of Work

- Although India is prone to low voting rates and fairness conducted in the system. To overcome these issues and provide fair and faster voting system, this approachis deployed.

1.4 Objectives

- To develop web-based application for voting using blockchain, HTML and CSSfor voting purpose.
- To provide security and trusted voting system, using elliptical curve algorithm and three step authentications.
- To store data and classify the user image information, a database is used
- To provide graphical- oriented result using matplotlib and sklearn

1.5 Methodology

- People's biometrics (finger-prints, voter-id, verified mobile number) are collected and stored in the database for verification purpose.
- When the user is allowed to cast their vote, he/she is first verified and then allowed to use the website.
- By selecting the certain area or region polling, the user can vote accordingly
- Each individual vote is concomitant by using block-chain technology.
- Every individual vote is collected and stored for the further statistical purpose.
- After the polling, each region vote is counted using the algorithms and statistics.
- The obtained statistics are visualized in the form of graph, to represent the polling result.

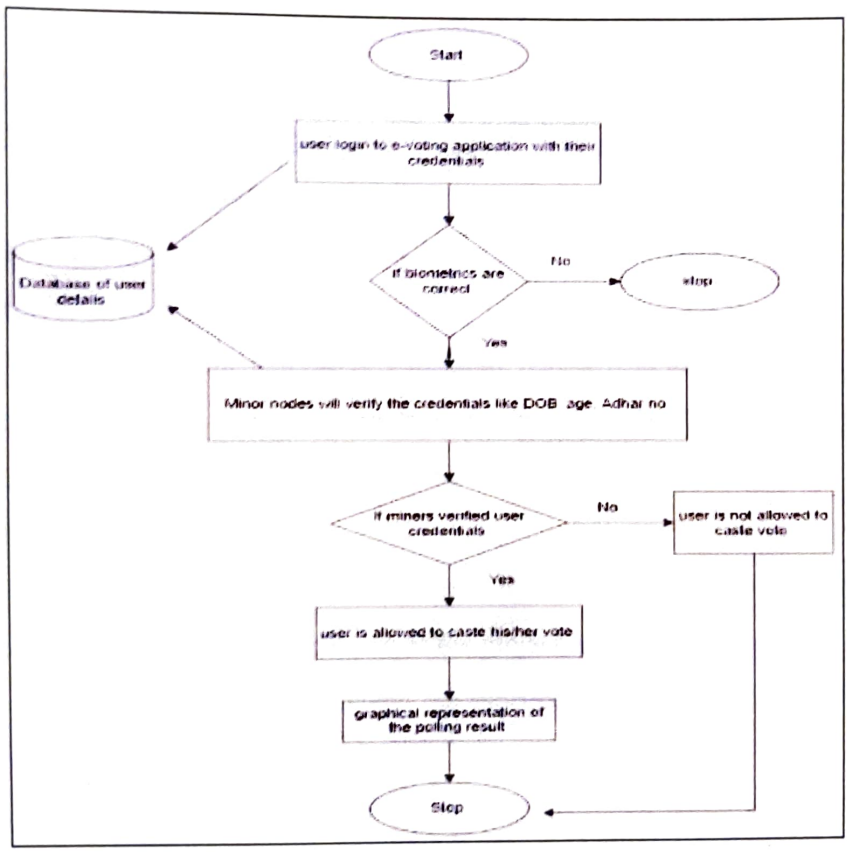


Figure 1.1 Flow chart

1.5 Organization of the Report

This report is organized into majorly into 4 different sections and each section provides detailed/ brief description about the project.

Chapter 1:

1. Overview: the overview provides the basic layout and the insight about the paper work proposed. It briefs the entire need of the currently proposed work.
2. Problem Statement: A problem statement is a concise description of an issue to be addressed or a condition to be improved upon. We have identified the gap between addressed or a condition to be improved upon. We have identified the gap between the current existing work and the desired work. Considering the needs of society, we have made an effort to approach the existing problems.
3. Significance and Relevance of work: we have mentioned about the contribution of our work to the society.
4. Objectives: A project objective describes the desired results of the work. We have mentioned about the work we are trying to accomplish in this section.
5. Methodology: a methodology is a collection of methods, practices, processes and techniques. We have explained in this section about the working of the project in a brief way.

Chapter 2

1. Literature survey: the purpose of a literature review is to gain an understanding of the existing resources to a particular topic or area of study. We have referred to many research papers relevant to our work in a better way.

Chapter 3:

1. System Requirements and Specifications: System requirement and specification is a document that describes the nature of the project, software or application. This section contains the brief knowledge about the functional and non-functional that needed to implement the project.

Chapter 4:

1. System Analysis: System analysis is a document that describes about the existing system and proposed system in the project. And also describes about advantages and disadvantages in the project.

Chapter 5:

1. System Design: System design is a document that describes about the project models, activity diagram, use case diagram, data flow diagram, and sequence diagram detailed in the project.

Chapter 6:

1. Implementation: implementation is a document that describes about the detailed concept of the project. Also describes about the algorithm with their detailed steps. And also, about the codes for implementation of the algorithm.

Chapter 7:

1. Testing: testing is a document that describes about the
   a. Methods of testing: this contains information about the unit testing, validation testing, functional testing, integration testing, user acceptance testing.
   b. Test cases: in test cases contains the detailed description about program test cases.

Chapter 8:

1. Performance Analysis: performance analysis is a document that describes about the study system in a detailed manner.

Chapter 9:

1. Conclusion and Future Enhancement: conclusion in future enhancement is a document that describes about the brief summary of the project and undetermined event that will occur in that time.

CHAPTER 2
LITERATURE SURVEY

CHAPTER – 2

LITERATURE SURVEY

FREYA SHEER HARDWICK etdl[1]: The author uses, the smart contracts and the PKIs for the verification and digital signaturesfor the first step of e-voting which is highly reliable and effective, also explains the e-votingusing decentralized e-voting system with the voter privacy rights. The protocol has been designed to adhere to fundamental e-voting properties as well as degree of decentralizationand allow the voter to change or update their vote.

Limitations: Implementing the changing or updating the vote, may leads to the less productive system, which increases the complexity of the algorithm, the effects the ongoing process of e-voting statistics.

ALI KAAN KOC etdl[2] Author clearly put forwards the idea of using Ethereum blockchain technique in the e- voting systems, this paper uses smart contracts for the verification and digital signatures ofthe blocks, which is safer, cheaper, more secure, more transparent and easier to use e-votingsystems. The idea of using Ethereum blockchain and smart contracts for an e-voting is itselfan high-minded, if implementation is successful, then e-voting will be secure enough to process all the voting through e-voting systems.

Limitations: Using smart contracts and Ethereum alone takes up to more storage.

NIR KSHETRI etdl[3] Authors explain the requirements and need of using blockchain effectively in the e-votingprocess, and provides a detailed survey for the e-voting technology in a certain region. It mainly put forwards the challenges faced before implementing blockchain based e-voting and the challenges to overcome by implementing blockchain technology in the e-voting.

Limitations: Author explains only the challenges to be faced by implementing blockchain technology, in e-voting but doesn't provide the solution for those issues.

SHEKHAR MISHRA etdl[4] In this paper, the author institutes about various authentication types like using biometric finger print using Aadhar card authentication, which enables the user to access the e-votingsystem using biometric finger print and verifies the Aadhar number for further processing of voting system.

Limitations: With only the biometric system with finger-print doesn't provide enough security to the user.

Instead of only using only the fingerprint authentication, instead 3-step authentication and face recognition can be used to provide more security.

AMNA QUERESHI retdl[5] In this paper "Secure and Electronic polling system", the authors AMNA QURESHI, DAVID MEGÍAS, HELENA RIFA-POUS described Se-VEP, an e-polling system enabledby Internet which provides and protects the voter's integrity, security, voters unique details,poll integrity, third party breaching, prevention of double voting , fairness in election, andcoercion resistance, and preventing devices with virus which change the users decision in voting and giving false results which leads to lot of problems.

RIFA HANIFATUNISA etdl[6] Author insinuates about, whole blockchain process and how it works in the process of e- voting, and explains how the hashing and Elliptic curve digital signature algorithm,provides the same amount of security as DSA but with smaller key length, allowing for faster calculations. This algorithm is a development of generalized digital signature algorithm using ECC algorithm in the digital signature generation process and its verification.

Limitations: Using Elliptic curve digital signature provides relatable security, with the smaller storage and faster calculations.

KANIKA GRAB, PAVI SARADWAT etdl[7] This paper states the different types of voting systems present, and compares the different types of voting systems, it propounds the different types of e-voting system present currently, and differentiates the different technologies used in different e-voting system andprovides the relatable and productive e-voting system with the technology.

ISHANI MANDAI etdl[8] In this paper "Secure and Hassle Free EVM through deep learning face recognition" author"Ishani Mondal", used neural networks after extracting the facial features of the voter andwith that a reference to vote during election. If the details match the existing details the user is allowed to vote.

CHAPTER 3

SYSTEM REQUIREMENTS SPECIFICATION

CHAPTER – 3

SYSTEM REQUIREMENTS AND SPECIFICATION

3.1 System Requirement Specification:

A Software Requirements Specification (SRS) is a document that describes the nature of a project, software or application. In simple words, SRS document is a manual of a project provided it is prepared before you kick-start a project/application. This document is primarily prepared for a project, software or any kind of application.

3.2 Specific Requirement:

Specific Requirements describes the external interface requirements, logical database requirements etc.

In this system following are the specific requirements:

- Merkel tools
- Block mining
- Django utils

3.3 Hardware specifications:

PROCESSOR	:	Intel i5
RAM	:	4GB
HARD DISK	:	16GB

3.4 Software Requirements:

OPERATING SYSTEM	:	Windows
BACK-END	:	Python3
DATABASE	:	Django
FRAMEWORKS	:	Blockchain
FRONT-END	:	HTML, CSS, Json
OTHER REQUIREMENTS	:	HTML5 Enabled browser

3.5 Functional Requirements:

Functional Requirement defines a function of a software system and how the system must behave when presented with specific inputs or conditions. These may include calculations, data manipulation and processing and other specific functionality.

In this system following are the functional requirements: -

- Voter's initial(biometric) details have to be registered priorly.
- The details of the voter must be encrypted using smart contracts and blocks.
- The user has to determine the preferred political party and cast his/her vote accordingly.
- The application shows the real time results of the polling with show vote count option

3.6 Non-Functional Requirements:

Non-functional requirements are the requirements which are not directly concerned with the specific function delivered by the system. They specify the criteria that can be used to judge the operation of a system rather than specific behaviors. They may relate to emergent system properties such as reliability, response time and store occupancy.

The non-functional requirements are:

- The project is not consume more space and the processing is done quickly.
- Portable: the system can be used for all the operating system which support python

3.7 Performance Requirement:

These are the requirements which define how well the software system accomplishes certain functions under specific conditions.

CHAPTER 4
SYSTEM ANALYSIS

CHAPTER – 4

SYSTEM ANALYSIS:

4.1 Existing system:

As Traditional voting systems like ballot paper voting and EVM voting failed to provide the required security, Anonymity and Integrity to the voters vote, due to this e-voting system became prominent and effective.

Many people have worked on different ways to promote voting on internet platform, for easier and secure voting system. Out of those a journal paper [1] related to "Electronic voting system using an enterprise blockchain" explained in detail how the block-chain hyper ledger fabrics would dwell into e-voting system. This paper explains clearly, how blockchain is implemented in e-voting system using Suffrage Net Networks and DAPP, where SUFFRAGE.MINTER and SUFFRAGE.VOTER acts as a one main authorization to manage and issue voter transactions, while Validator is the one which validates the entire electoral process; using these networks block-chain is implemented and with the hyper-ledger the whole voting process is validated and kept secure.

In another paper [2], "Blockchain Based E-voting System" this journal paper has enhanced the security of the e-voting system, by adding finger-print to the existing e-voting system for anonymity and verification.

4.2 Proposed system:

The simple rationalization could be a 'Chain 'of blocks. A block is associate degree mass set of information. knowledge square measure collected and methods to suit in an exceedingly block through a process known as mining. every block may be known employing a science hash (also referred to as a digital face authentication). The block shaped can contain a hash of the previous block, so blocks will kind a sequence from the primary block ever (known because the Genesis Block) to the shaped block. during this method, all the information may be connected via a connected list structure. And to display the live result of the poll in a defined graph manner, to improve the visualization of the project

CHAPTER 5
SYSTEM DESIGN

CHAPTER – 5

SYSTEM DESIGN:

5.1 Project Modules:

1. **Registering to the Network:**

The user has to register to the network by providing, the valid details like username, age, gender, address, email-id and valid Aadhar number

2. **Aadhar Input and Verification:**

After registering to the network, the user can access the website; and enter registered Aadhar number, once the Aadhar number is fed to website, it verifies the Aadhar number and validates user whether to proceed in the voting process or not.

3. **Casting vote:**

Authorized users are allowed to select the election party and the representative of the respective region and cast their vote.

4. **Poll results:**

Once the election poll is completed, the user can access the website to notice which election party has won the election poll.

5.1.1 System Architecture:

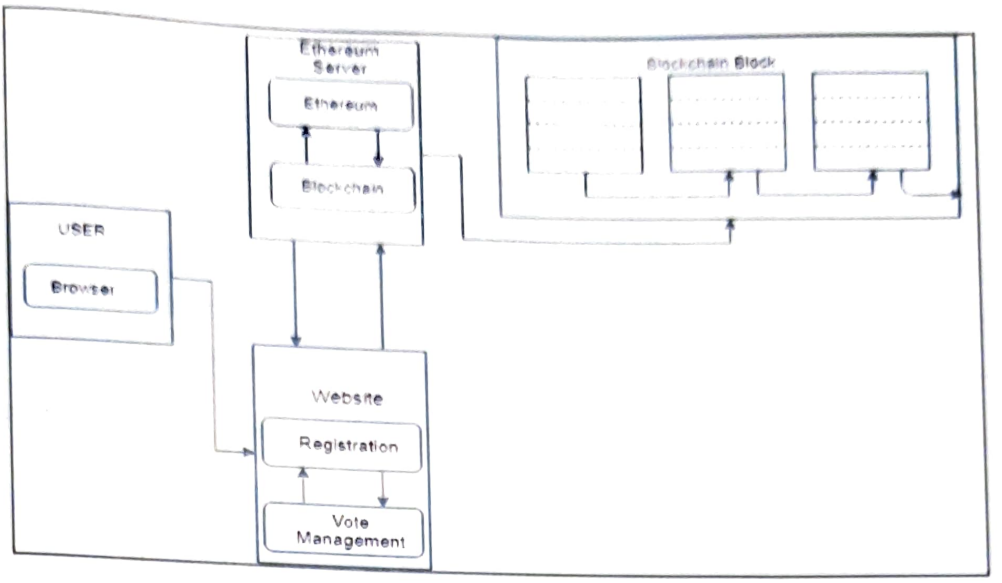


Figure 5.1 System Architecture

The system architecture is divided into three-parts, namely

1. Pre-voting phase
2. Voting phase
3. Post-voting phase

Pre-voting phase:

1. The user has to register to the network, and to get permission to vote in the described chain of networks.

The user has to create an id according to his official voter id information, and login to the network, here the previous blocks verify the new block(user) information, whether it is according to the data stored in the database or not. If the new block is validated correctly, the network allows the block to process to next stage in voting process.

2. The user has to verify his identification using Aadhar authentication, and then process to further steps in voting process

Voting phase:

After successful Aadhar authentication, the user is allowed to vote, the user's vote will be encrypted by the public key encryption and once the user has voted, then he cannot re-login and vote in the network, this re-voting using same id is restricted using smart contracts.

The vote cannot be manipulated, if one has to change his/her vote after voting, then it is almost impossible because to change one vote, the entire system of blocks should be changed, and every node in the network should authenticate, as every block contains every block information and transaction.

Post-voting phase:

Once the election poll is completed, the user can access the website to notice which election party has won the election poll.

5.2 Use-Case diagram:

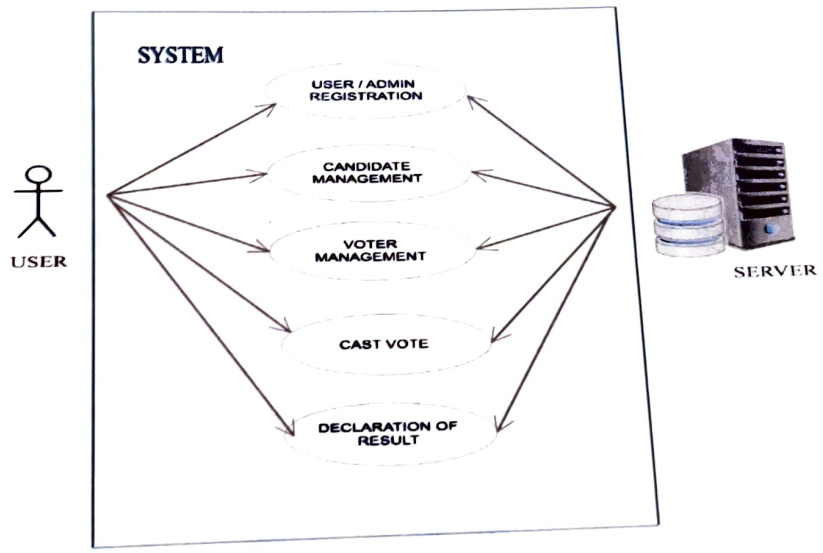


Figure 5.2 Use-Case Diagram

Use case consists of user and admin, where user is allowed to login and caste his/her vote. and admin manages all the required processings like, managing candidate details; and authorising the right voter to cast his/her vote, after the verification of the user complete biometrics, face identification; and after the election poll ends, the admin is responsible for declaration of result in a graphical representation.

5.3 Data Flow Diagram:

A Data-Flow Diagram is a way of representing a flow of a data of a process or an information system. The DFD also provides information about the outputs and inputs of each entity and the process itself. A dataflow diagram has no control flow, there are no decision rules and no loops. Specific operations based on the data can be represented by a flowchart.

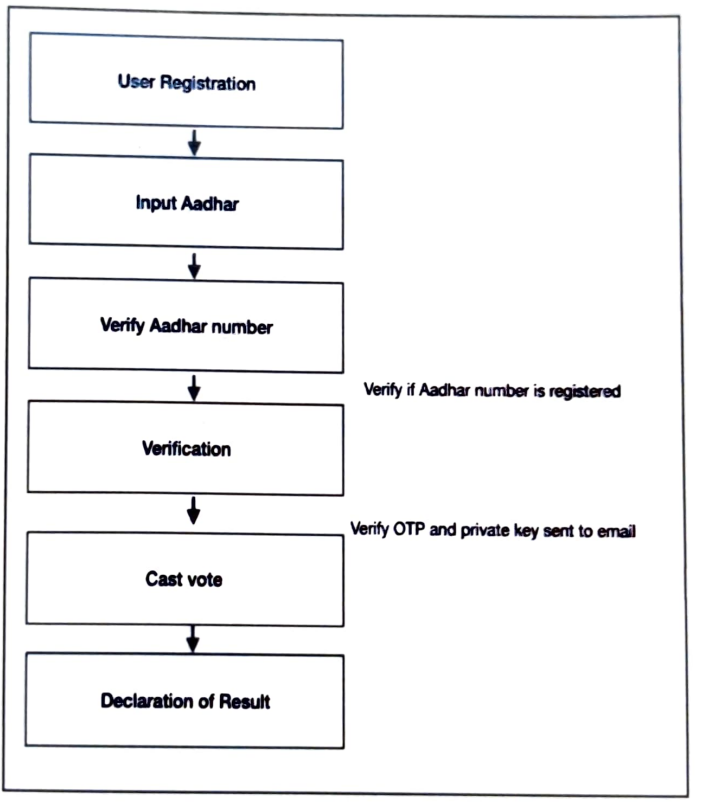


Figure 5.3 Flowchart

5.4 Level 1 dataflow:

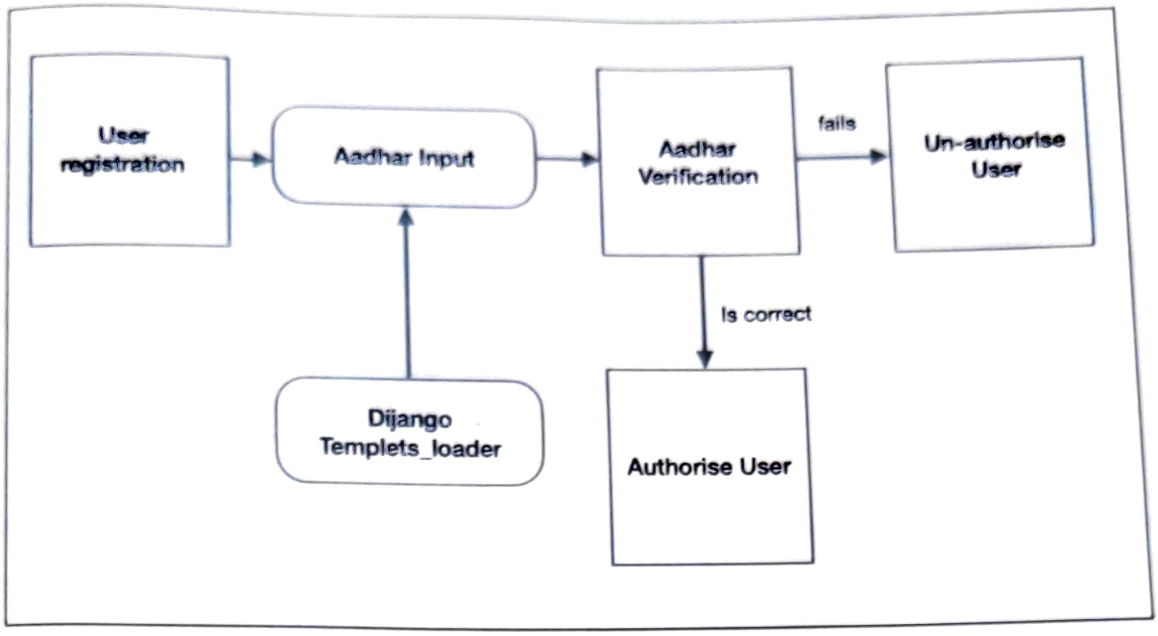


Figure 5.4 Level-1 Dataflow Diagram

Above mentioned diagram is the representation of DFD1 which provides u the content diagram or say overview of the whole system. It is designed to be an at- a-glance view, showing the system as single high-level process. Here, the user has to register before-hand with all his details like voter biometrics, email-id, etc. once the user registers his details with the administrator, he can proceed to further step in the process, on the e-voting website the voter has to input his Aadhar number and has to verified with the registered Aadhar number. If the Aadhar number is not registered previously then the voter is not allowed to processed in further process. If the Aadhar number is previously registered and verified, that voter is allowed to process in the further steps.

5.5 Level 2 data flow:

Above mentioned diagram is the representation of DFD2. The Level 1 DFD is broken down into more specific, Level 2 DFD. Level 2 DFD depicts basic modules in the system and flow of data among various modules. Here, the authorized user is connected to local blockchain, and to the verified email-id, the OTP is sent for verification purpose using SMTP protocols; once the OTP validation is done. The user is allowed to create block in the blockchain where each block will have hash values, and after the block creation, the Private key is sent to the registered email-id; the private key is generated using the SHA-256 algorithm, and blocks are created using Merkel tools.

Once the private key is verified, the user is allowed to cast his vote. After validating all votes, the result of the poll is decided.

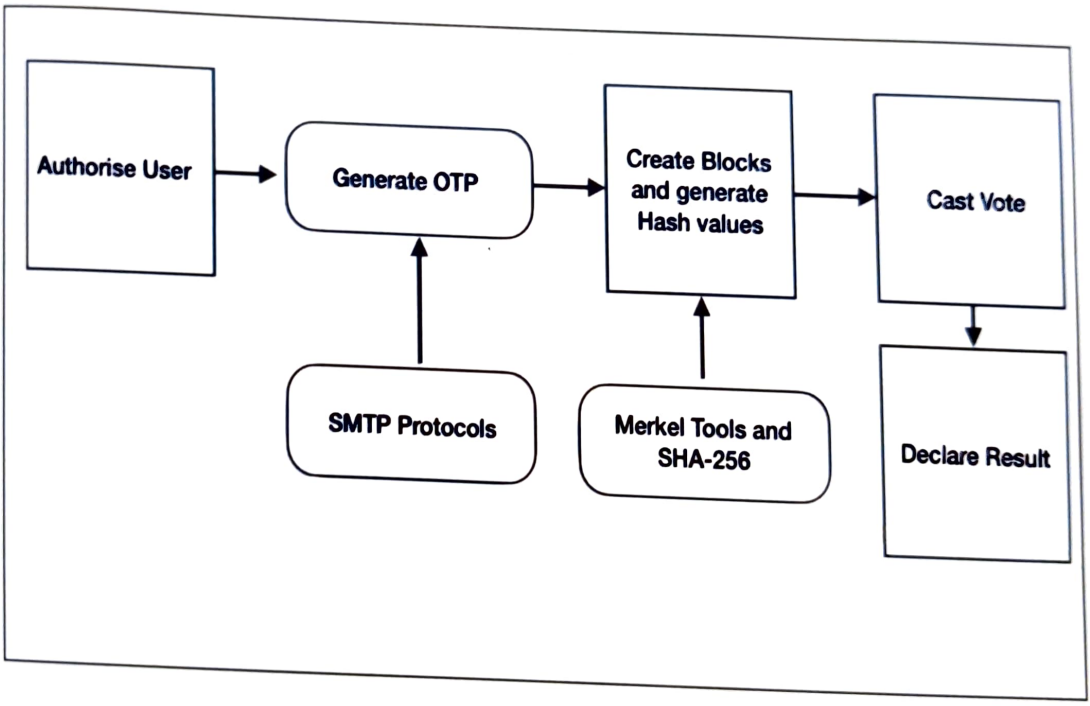


Figure 5.5 Level-2 Dataflow Diagram

CHAPTER 6
IMPLEMENTATION

CHAPTER-6

IMPLEMENTATION

6.1 Concept:

The implementation methods are as follows:

Since the project is based on Electronic Voting using Blockchain, In the first step of voting process voters need to register their votes using Aadhar card and in the next step the voter can cast their vote using Aadhar number. They can only cast their vote only if their image is matched with the Aadhar card image. After validating the image there is one more step of authentication that is otp sent to a mail, which is provided by the voter. Now multiple parties along with the nota will be visible to the voters so that they can select any one of them. To cast their vote there is one more authentication step after selecting the party, that last authentication step is based on blockchain Here need to enter the encrypted hash value that is received in the mail. If the encrypted code is verified then they can cast their vote, otherwise they cannot cast their vote. Each voter can caste vote only one time. After casting the vote, they can view the results.

6.2Algorithm:

a. **SMTP (for sending otp to email-id):**

1. Composition of Mail: A user sends an e-mail by composing an electronic mail message using a Mail User Agent (MUA). Mail User Agent is a program which is used to send and receive mail. The message contains two parts: body and header. The body is the main part of the message while the header includes information such as the sender and recipient address. The header also includes descriptive information such as the subject of the message. In this case, the message body is like a letter and header is like an envelope that contains the recipient's address.

2. Submission of Mail: After composing an email, the mail client then submits the completed e-mail to the SMTP server by using SMTP on TCP port

3. Delivery of Mail: E-mail addresses contain two parts: username of the recipient and domain name. For example, vivek@gmail.com, where "vivek" is the username of the recipient and "gmail.com" is the domain name.
If the domain name of the recipient's email address is different from the sender's domain name, then MSA will send the mail to the Mail Transfer Agent (MTA). To relay the email, the MTA will find the target domain. It checks the MX record from Domain Name System to obtain the target domain. The MX record contains the domain name and IP address of the recipient's domain. Once the record is located, MTA connects to the exchange server to relay the message.

4. Receipt and Processing of Mail: Once the incoming message is received, the exchange server delivers it to the incoming server (Mail Delivery Agent) which stores the e-mail where it waits for the user to retrieve it.

5. Access and Retrieval of Mail: The stored email in MDA can be retrieved by using MUA (Mail User Agent). MUA can be accessed by using login and password.

b. SHA-256 Bit Encryption Algorithm:

The SHA-256 algorithm is one flavour of SHA-2 (Secure Hash Algorithm 2), which was created by the National Security Agency in 2001 as a successor to SHA-1. SHA-256 is a patented cryptographic hash function that outputs a value that is 256 bits long.

In encryption, data is transformed into a secure format that is unreadable unless the recipient has a key. In its encrypted form, the data may be of unlimited size, often just as long as when unencrypted. In hashing, by contrast, data of arbitrary size is mapped to data of fixed size. For example, a 512-bit string of data would be transformed into a 256-bit string through SHA-256 hashing.

In cryptographic hashing, the hashed data is modified in a way that makes it completely unreadable. It would be virtually impossible to convert the 256-bit hash mentioned above back to its original 512-bit form. The most common reason is to verify the content of data that must be kept secret. For example, hashing is used to verify the integrity of secure messages and files. The hash code of a secure file can be posted publicly so users who download the file can confirm they have an authentic version without the contents of the file being revealed. Hashes are similarly used to verify digital signature.

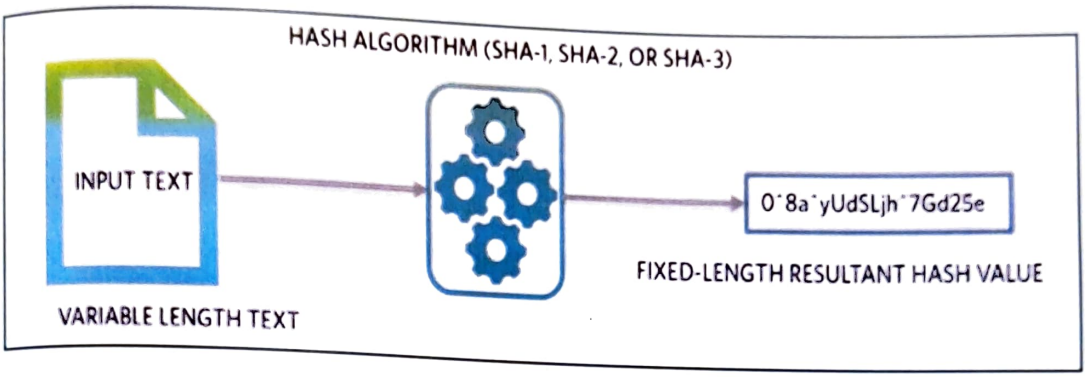


Figure 6.1 SHA-256 Bit Algorithm Working

Password verification is a particularly important application for cryptographic hashing. Storing users' passwords in a plain-text document is a recipe for disaster; any hacker that manages to access the document would discover a treasure trove of unprotected passwords. That's why it's more secure to store the hash values of passwords instead. When a user enters a password, the hash value is calculated and then compared with the table. If it matches one of the saved hashes, it's a valid password and the user can be permitted access.

C. Merkel Hash

A Merkle tree totals all transactions in a block and generates a digital fingerprint of the entire set of operations, allowing the user to verify whether it includes a transaction in the block

Merkel tress are made by hashing pairs of nodes repeatedly until only one hash remains; this hash is known as the Merkel root. They are built from the bottom, using Transaction ID's which are hashes of the individual transactions. Each non-leaf is a hash of its previous hash, and every leaf node is a hash of transactional data.

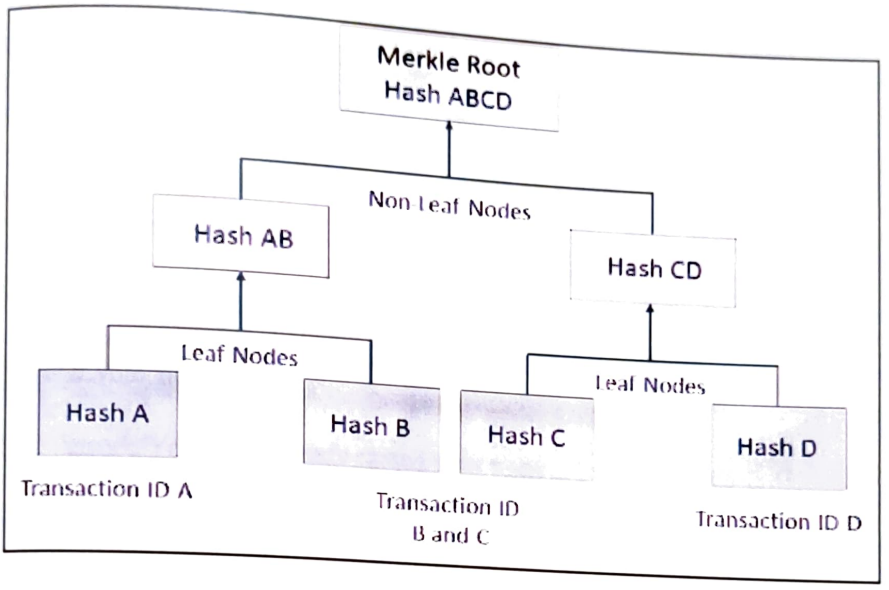


Figure 6.2 Merkel Hash Working

6.3 Functional modules

The functional modules of the electronic voting using blockchain are:

6.3.1 Input Aadhar number

To caste vote the voters need to register their votes before the elections using their Aadhar number. Only the registered candidates are allowed to vote. The voters need to enter the Aadhar number as the input in order to cast their vote.

```
def authentication(request):

  aadhar_no = '2'

  details = {'success': False}
  print(aadhar_no)
  try:
    voter = Voters.objects.get(uuid = aadhar_no)
    print(voter)
    request.session['uuid'] = aadhar_no
    render_html = loader.render_to_string('candidate_details.html', {'details': voter})
    if voter.vote_done:
      details = {
        'error': 'You have already casted your vote.'
      }
    else:
      details = {
        'success': True,
        'html': render_html,
        'details': model_to_dict(voter)
      }
  except:
    details = {
      'error': 'Invalid Aadhar, Please Enter Correct Aadhar Number!'
    }

  return JsonResponse(details)
```

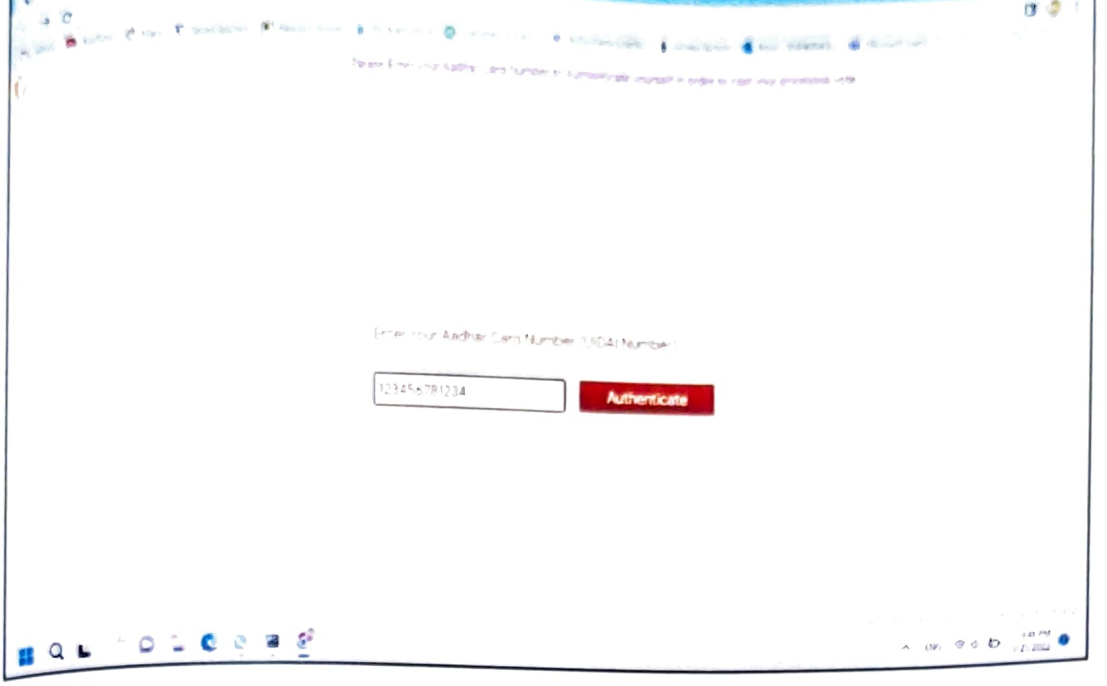


Figure 6.3 Input Aadhar Number

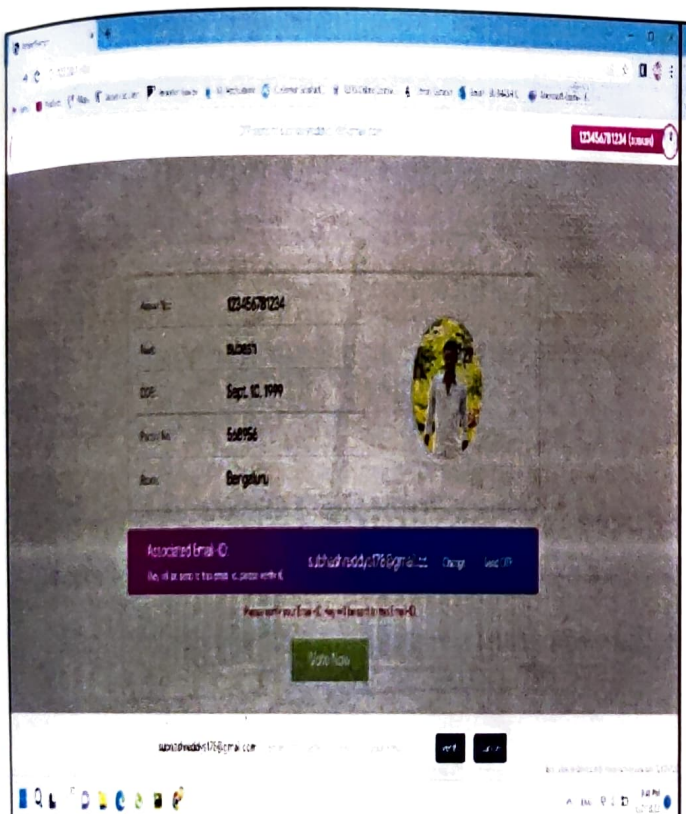
6.3.2 OTP Validation

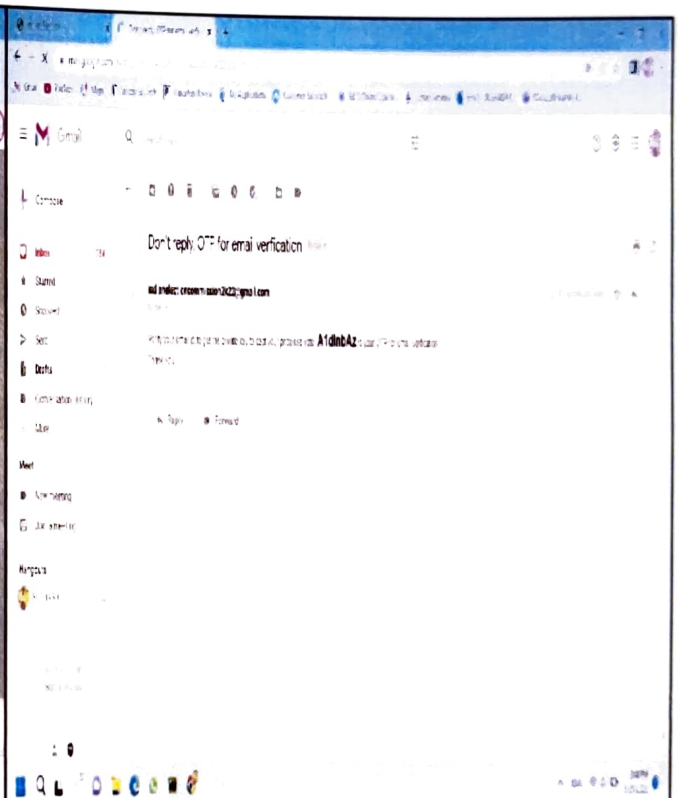
After entering the valid Aadhar number it is verified through otp it is sent to the candidate mail id which is provided at the time of registration.

This otp process is done with the help of SMTP.SMTP is an application layer protocol. The client who wants to send the mail opens a TCP connection to the SMTP server and then sends the mail across the connection. The SMTP server is an always-on listening mode. As soon as it listens for a TCP connection from any client, the SMTP process initiates a connection through port 25. After successfully establishing a TCP connection the client process sends the mail instantly.

```
def send_otp(request):
  email_input = request.GET.get('email-id')

  [success, result] = send_email_otp(email_input)
  print(result)
  #[success, result] = [True, '0']

  json = {'success': success}
  if success:
    request.session['otp'] = result
```

```
        request.session['email-id'] = email_input
        request.session['email-verified'] = False
    else:
        json['error'] = result

    return JsonResponse(json)

def verify_otp(request):

    otp_input = request.GET.get('otp-input')
    json = {'success': False}
    if otp_input == request.session['otp']:
        voter = Voters.objects.get(uuid = request.session['uuid'])
        voter.email = request.session['email-id']
        voter.save()
        json['success'] = True
        request.session['email-verified'] = True

    return JsonResponse(json)
```





6.3.3 Private key authentication

After successful completion of otp validation now the candidate able to see the party names and symbols along with the nota. If they click on the party which they are going to vote it will ask for authentication that is known as private key authentication where they need to enter the private key which they have received to their registered email id. The generation of private key is done through hashing.

Hashing is the procedure of interpreting a given key into a code. A hash function is used to substitute the data with a freshly produced hash code. Furthermore, hashing is the practice of taking a string or input key, a variable generated for saving narrative information, and defining it with a hash value, which is generally decided by an algorithm and create a much shorter string than the original.

Hashing is generally a one-way cryptographic function. Because hashes are irreversible, understanding the output of a hashing method does not enable us to regenerate the contents of a file. It allows us to assess whether two files are same without understanding their contents.

The use of hashing in information security and internet authentication is a common practice. For example, it can be used to securely save passwords in a database, but can also provide the security of other element of information including files and documents.

The hashing data structure allows arrays to effectively find and store information, supporting an effective structure for finding and storing information. Suppose that it can have a list of 20,000 numbers and it is asked to look for a specific number in that list and it can scan each number in the list to view if it matches the number that it is entered.

Hashing is the procedure of transforming a string of characters into a frequently shorter and fixed-length value. The why of using hashed keys to search for element in a database is that discovering the item using its original value is more time-consuming than using the shorter hashed key.

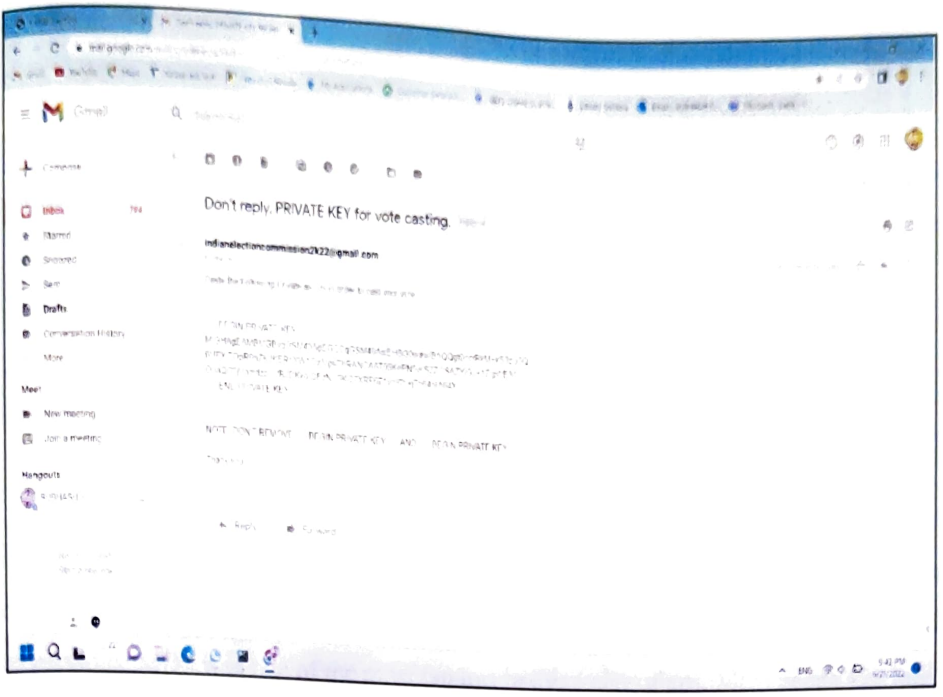


Figure 6.5 Private key Authentication

It can be used to locate or store elements effectively in collections when searching for them. For example, if it can have a list of 10,000 English words and want to check if a given word is between them, it will be inefficient to compare the given word to all 10,000 items until a match is discovered.

An array can be indexed by the values of the keys of a range, which is known as hashing. The modulo operator will be used to acquire a range of key values. In this case, it can store the following items in a 20x20 hash table. Each item is formatted as a (key, value).

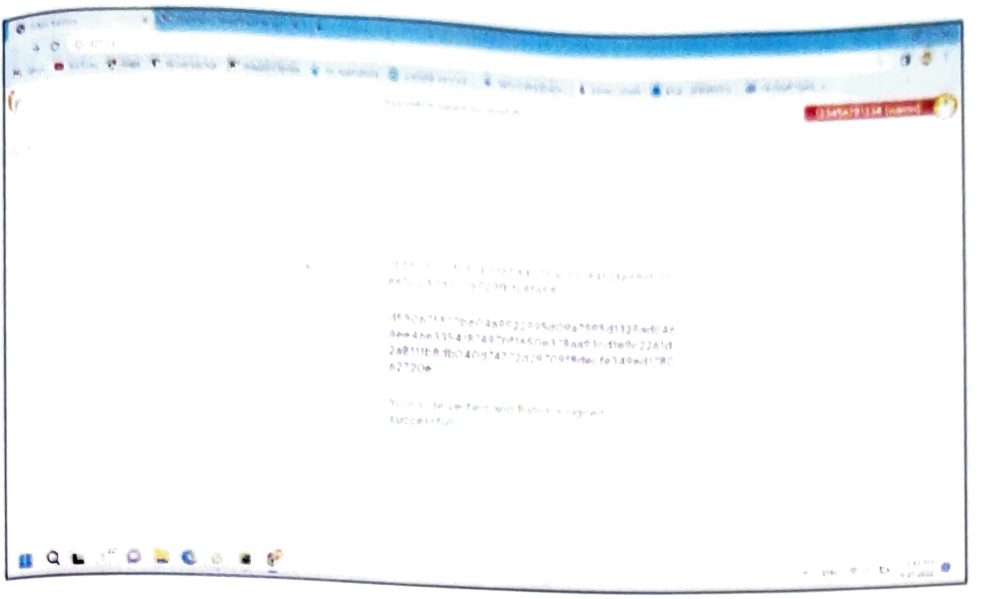


Figure 6.6 Hash Encrypted value

Hashing algorithms including MD5, SHA-1, SHA-2, NTLM, and LANMAN are all generally used in today's world. A message digest is divided down into 5 versions, this one being MD5. In the previous, MD5 was one of the most famous hashing algorithms. MD5 needs 128 bits for its outputs.

In hashing, each bit in the data block is transformed into a fixed-size bit string value. A file includes data blocks. There is a risk that two multiple inputs will create the same hash value. This is called a collision, which appears when two multiple inputs support the same hash value.

```
def blockchain(request):
    blocks = Block.objects.all()
    return render(request, 'blockchain.html', {'blocks':blocks})
def block_info(request):
    try:
        block = Block.objects.get(id=request.GET.get('id'))
        confirmed_by = (Block.objects.all().count() - block.id) + 1
        vote = Vote.objects.filter(block_id=request.GET.get('id'))
```

```
vote_hashes=
[SHA3_256.new((f'{vote.uuid}|{vote.vote_party_id}|{vote.timestamp}').encode('utf-
8')).hexdigest() for vote in votes]
    root = MerkleTools()
    root.add_leaf([f'{vote.uuid}|{vote.vote_party_id}|{vote.timestamp}' for vote in votes],
True)
root.make_tree()
    merkle_hash = root.get_merkle_root()
    tampered = block.merkle_hash != merkle_hash
    context = {
      'this_block': block,
      'confirmed_by': confirmed_by,
      'votes': zip(votes, vote_hashes),
      're_merkle_hash': merkle_hash,
      'isTampered': tampered,
    }
  return render(request, 'block-info.html', context)
except Exception as e:
  print(str(e))
  return render(request, 'block-info.html')
```

6.3.4 Casting vote

After successful completion of otp validation now the candidate able to see the party names and symbols along with the nota. If they click on the party which they are going to vote it will ask for authentication that is known as private key authentication where they need to enter the private key which they have received to their registered email id. After successful validation of private key the vote will be recorded and also after casting their vote they can also election results.

```
def show_result(request):
    vote_result = vote_count()
    vote_result = dict(reversed(sorted(vote_result.items(), key = lambda vr:(vr[1], vr[0]))))
    results = []

    political_parties = PoliticalParty.objects.all()
    i=0
    for party_id, votecount in vote_result.items():
        i+=1
        party = political_parties.get(party_id = party_id)
        results.append({
            'sr': i,
            'party_name': party.party_name,
            'party_symbol': party.party_logo,
            'vote_count': votecount
        })
    return render(request, 'show-result.html', {'results': results})
```

CHAPTER 7
TESTING

CHAPTER-7

TESTING

7.1 Methods of Testing

The various strategies that were used in testing this software were as follows:

1. Unit testing
2. Integration testing
3. Validation testing
4. User validation testing

7.1.1 Unit Testing

Unit testing, also known as component testing refers to tests that verify the functionality of a specific section of code, usually at the function level. In an object-oriented environment, this is usually at the class level, and the minimal unit tests include the constructors and destructors. Unit testing is a software development process that involves synchronized application of a broad spectrum of defect prevention and detection strategies in order to reduce software development risks, time, and costs. The following Unit Testing Table shows the functions that were tested at the time of programming. The first column gives all the modules which were tested, and the second column gives the test results. Test results indicate if the functions, for given inputs are delivering valid outputs.

Function Name Tests Results feeding the valid enrolled Aadhar number and authorise user using blockchain hash techniques; output is tested successful when the user can cast his vote through the website

Function name	Test results
Feed enrolled Aadhar number	Tested for different input of Aadhar numbers verification
Verify the details	Authorising only valid users to vote
Display result	Output is to cast vote only once by a single user

Table 7.1: Function Name and Test Results.

7.1.2 Integrating testing

Integration testing is any type of software testing that seeks to verify the interfaces between components against a software design. Software components may be integrated in an iterative way or all together ("big bang"). Normally the former is considered a better practice since it allows interface issues to be located more quickly and fixed. Integration testing works to expose defects in the interfaces and interaction between integrated components (modules). Progressively larger groups of tested software components corresponding to elements of the architectural design are integrated and tested until the software works as a system.

7.1.3 Validation Testing

At the culmination of integration testing, software is completed assembled as a package. Interfacing errors have been uncovered and corrected. Validation testing can be defined in many ways; here the testing validates the software function in a manner that is reasonably expected by the customer. In software project management, software testing, and software engineering, verification and validation (V&V) is the process of checking that a software system meets specifications and that it fulfills its intended purpose. It may also be referred to as software quality control.

7.1.4 User Acceptance Testing

Performance of an acceptance test is actually the user's show. User motivation and knowledge are critical for the successful performance of the system. The above tests were conducted on the newly designed system performed to the expectations. All the above testing strategies were done using the following test case design.

7.2 Unit Testing Test Cases

Input Aadhar test case:

Test case	1
Name of the test	Input Aadhar
Input	Valid unique ID
Expected output	Input Aadhar feed by the user
Actual output	Valid Aadhar number is accepted as enrolled in the database
Result	Successful

Table 7.2: input Aadhar test case

Email OTP authentication test case:

Test case	1
Name of the test	Email OTP authentication
Input	Valid/ enrolled email-id
Expected output	Obtain OTP to the registered email
Actual output	Receiving unique OTP from the enrolled email ID
Result	Successful

Table 7.3: Email OTP authentication test case

Private key verification Test case;

Test case	1
Name of the test	Private key verification
Input	Valid/ enrolled email ID
Expected output	Obtaining unique hash value
Actual output	Receiving unique hash value using blockchain from the enrolled email ID
Result	Successful

Table 7.4: Private key verification Test case

CHAPTER 8
PERFORMANCE ANALYSIS

CHAPTER – 8

PERFORMANCE ANALYSIS

Performance analysis is a specialist discipline involving systematic observations to enhance performance and improve the people voting and ease people difficulty in voting.

This voting system helps people who have smart mobiles and websites enabled laptops; also helps people from faraway places to vote conveniently from the place they are. It increases, the voting rate in the country and decreases the cost expenses of conducting live booth pooling in every place.

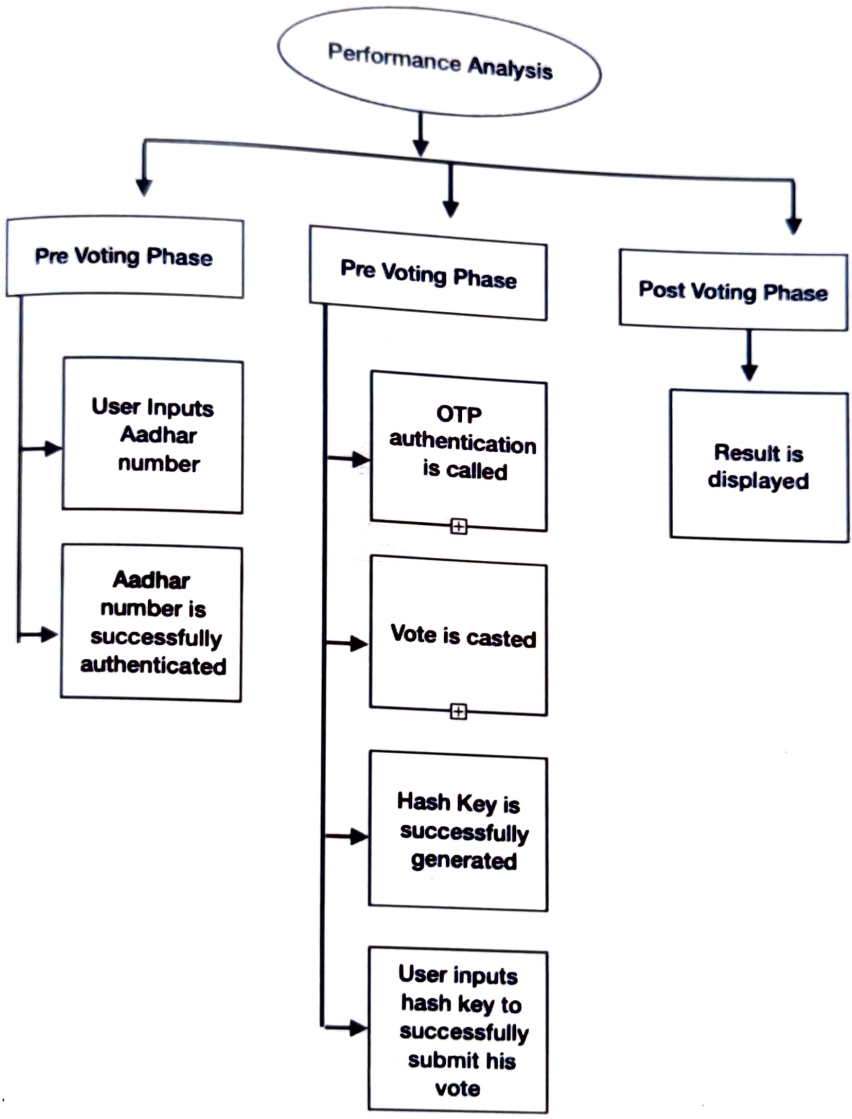


Figure 8.1 Performance Analysis

CHAPTER 9

ONCLUSION & FUTURE ENHANCEMENT

CHAPTER – 9

CONCLUSION AND FUTURE WORK

The proposed framework provides complete security to the e-voting system, with the usage of Ethereum blockchain and smart contracts to provide added security to the system. Blockchain implementation prevents vote manipulation and provides privacy, integrity for voters to cast their vote. Smart contracts ensures that the voter can vote only once using his/her unique id (Aadhar number); with the convention of different security algorithms like SHA-256, Merkel hash and SMTP prototyping, enhances the security of the system. As a result, the voter is authorized to cast his/her vote from where ever they are; provides high security standards to the system and convenient and easier ways to vote.

Future work:

1. To the proposed existing system, additional biometrics (fingerprint, face authentication) can be added to enhance the security of the system.
2. Three step authentications can also be used to provide more security to the system.

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] FREYA SHEER HARDWICK, APOSTOLOS GIOULIS, RAJA NAEEM AKRAM and KONSTANITINOS MARKANTONAKIS: "E-voting with Blockchain:An e-voting protocol with decentralization and voter privacy

[2] ALI KAAN KOC and EMRE YAVUZ: "Towards secure E-voting Using Ethereum Blockchain

[3] NIR KSHETRI and JEFFREY VOAS: "Blockchain-enabled E-voting"

[4] SHEKHAR MISHRA and Y. ROJA: "electronic voting machine using biometric ginger print with Aadhar card authentication

[5] AMNA Qureshi ''SEVEP: Verifiable, secure and privacy preserving remote polling with untrusted computing devices," in Future Network Systems and Security Feb 22(2019) IEEE.

[6] RIFA HANIFATUNISA and BUDI RAHARDJO: "blockchain based e-voting recording system design"

[7] KANIKA GRAB, PAVI SARADWAT, SACHINE BISHT, SAHIL AGGARVAL, SAI KRISHNA KOTHURI and SAHIL GUPTA: "A comparative analysis on e-voting system using blockchain"

[8] ISANI MANDAI: "Secure and hassle free EVM through deep learning face recognition"

[9] S. GANESH PRABHU, RACHEL, AGENS SHINYH, and A. R. ROSHINEE. "Tracking Real Time Vehicle and Locking System Using Lab View Applications." In 2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS), pp. 55-57.IEEE, 2020.

[10] ANNOSHMITHA DAS "VOT-EL: Three Tier Secured State Of-The-Art EVM DesignUsing Pragmatic Fingerprint Detection Annexed with NFC Enabled Voter -ID Card" (2016) IEEE.

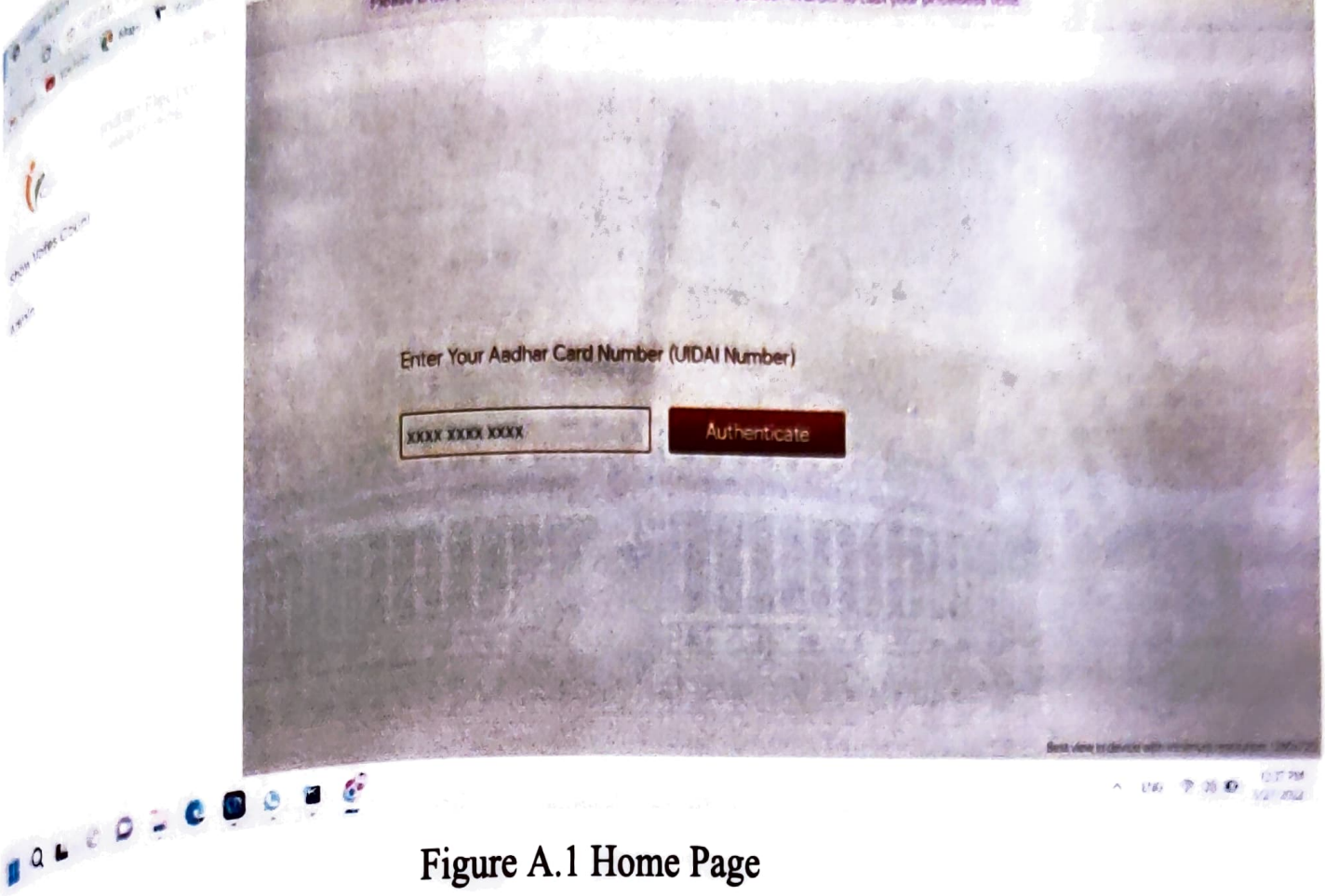
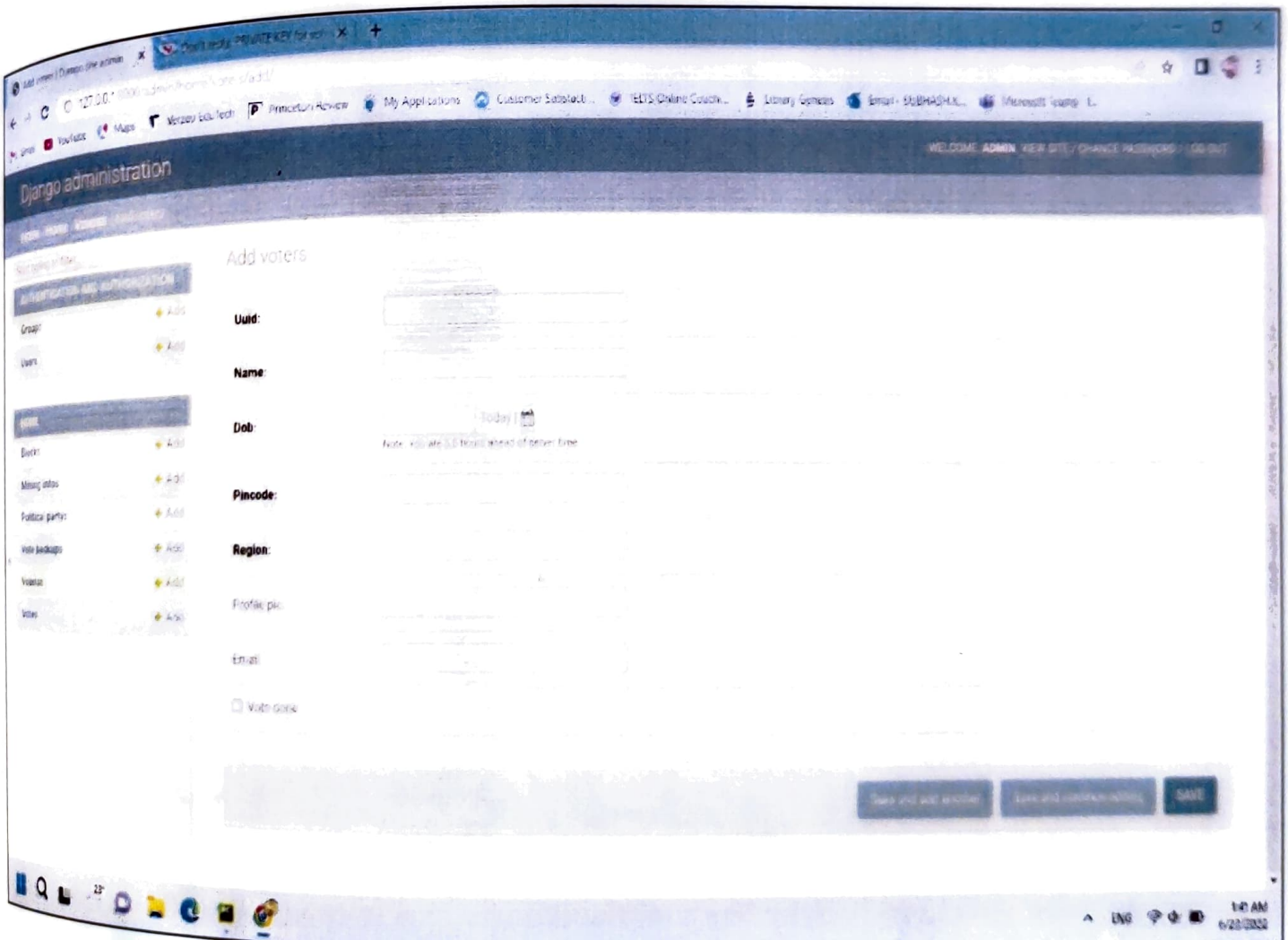


Figure A.1 Home Page



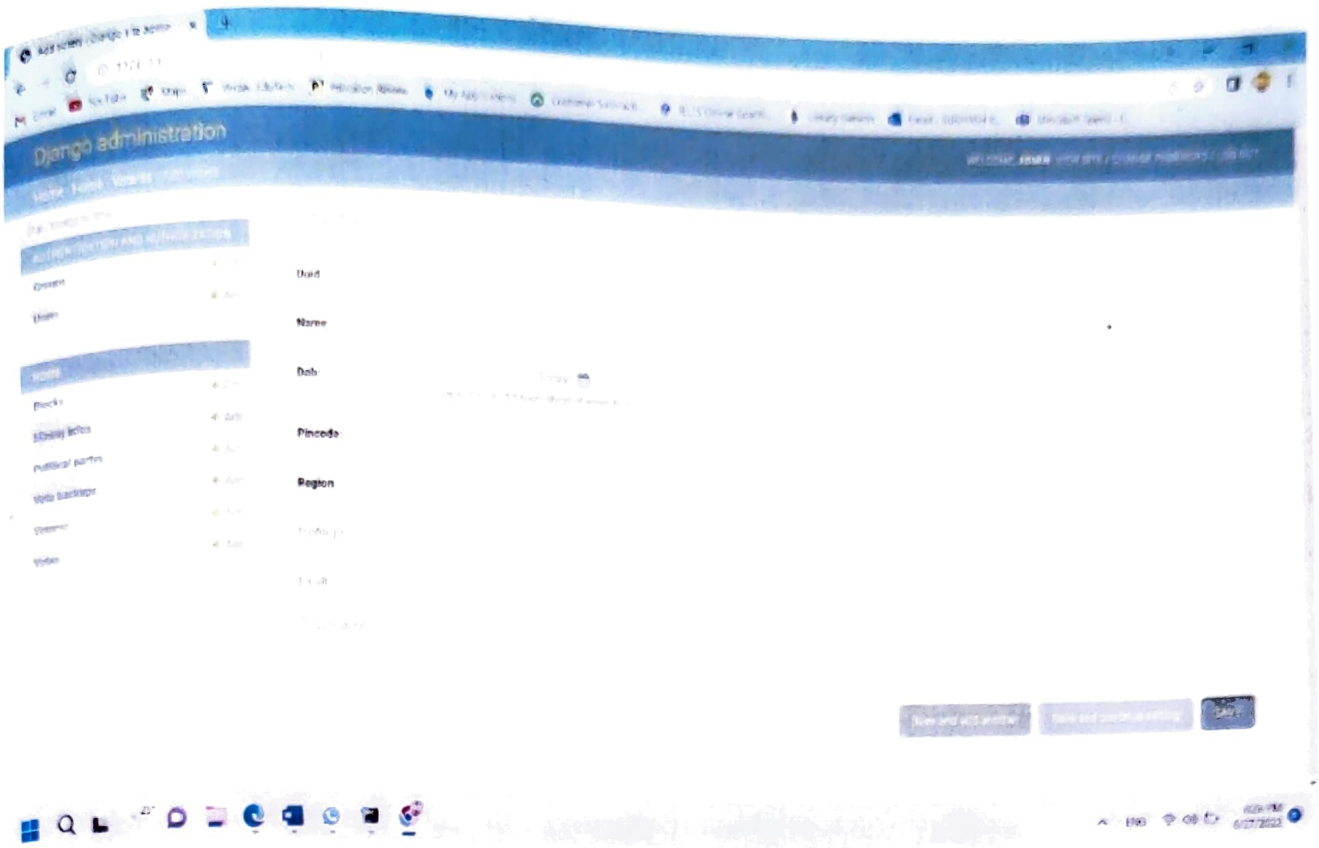
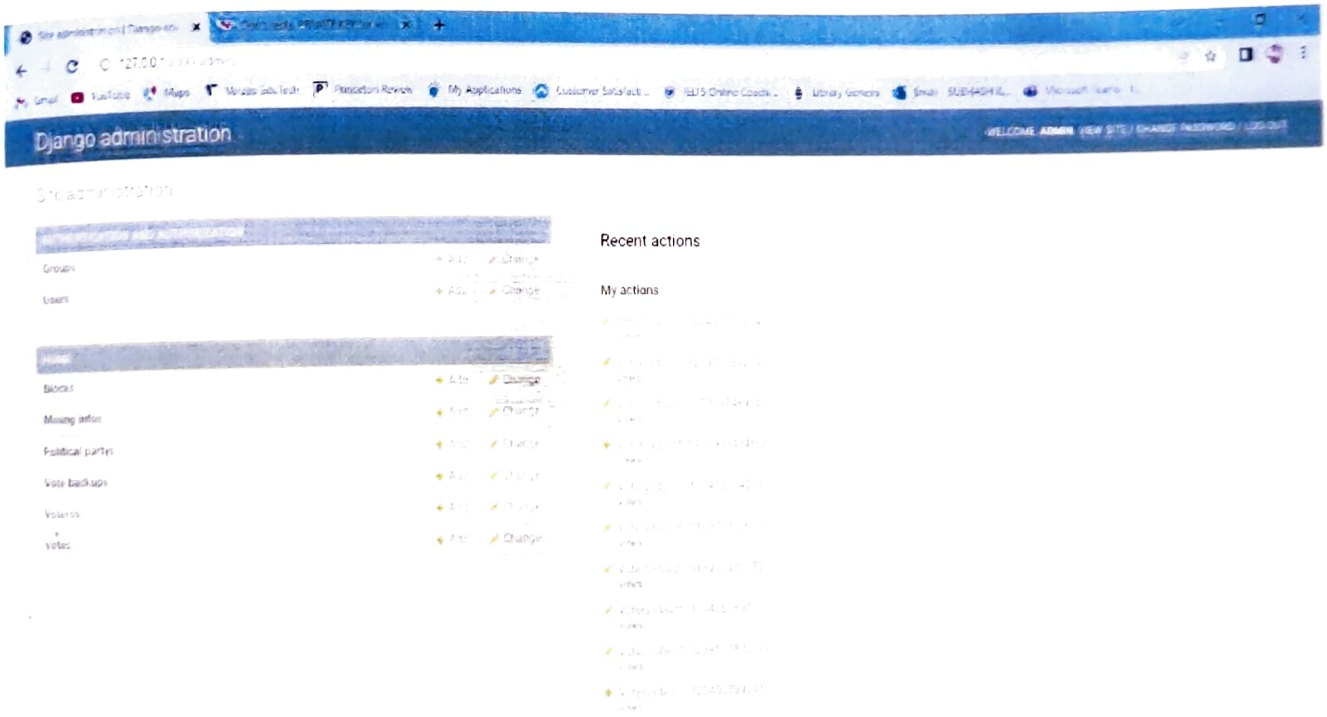


Figure A.3 Add Political Party



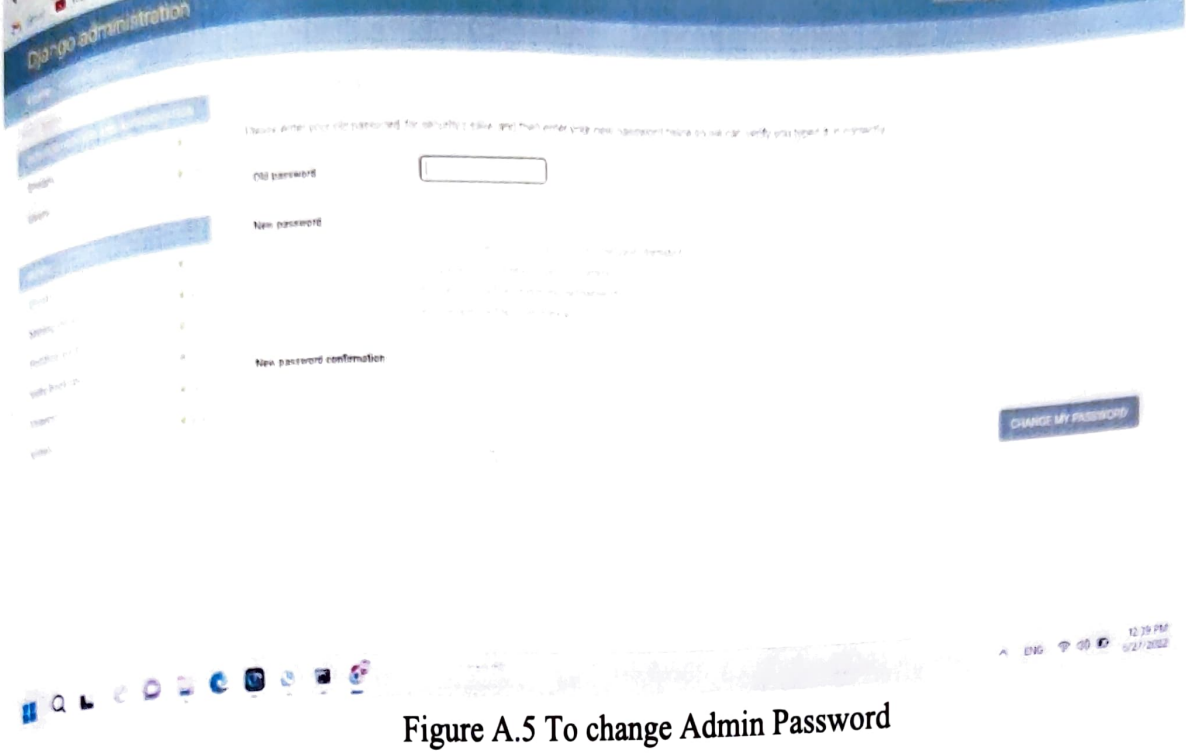


Figure A.5 To change Admin Password

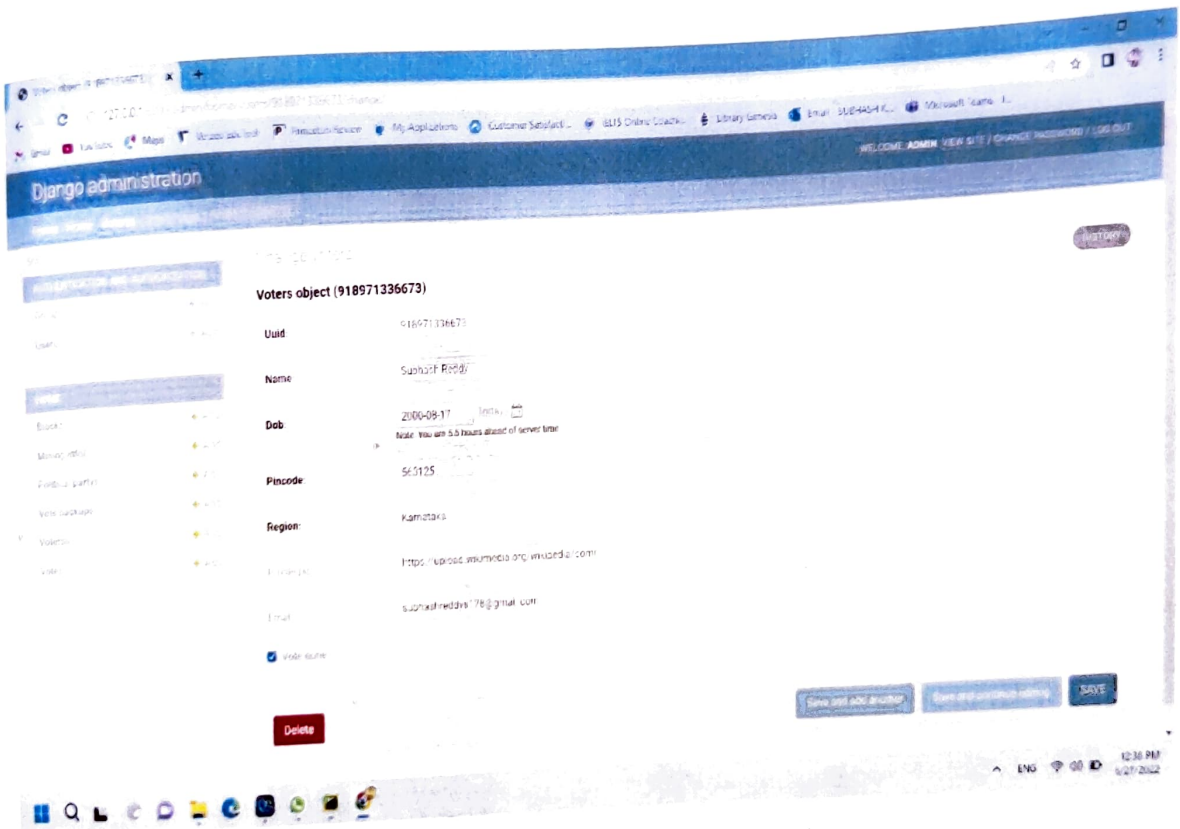


Figure A.6 to modify voters' details

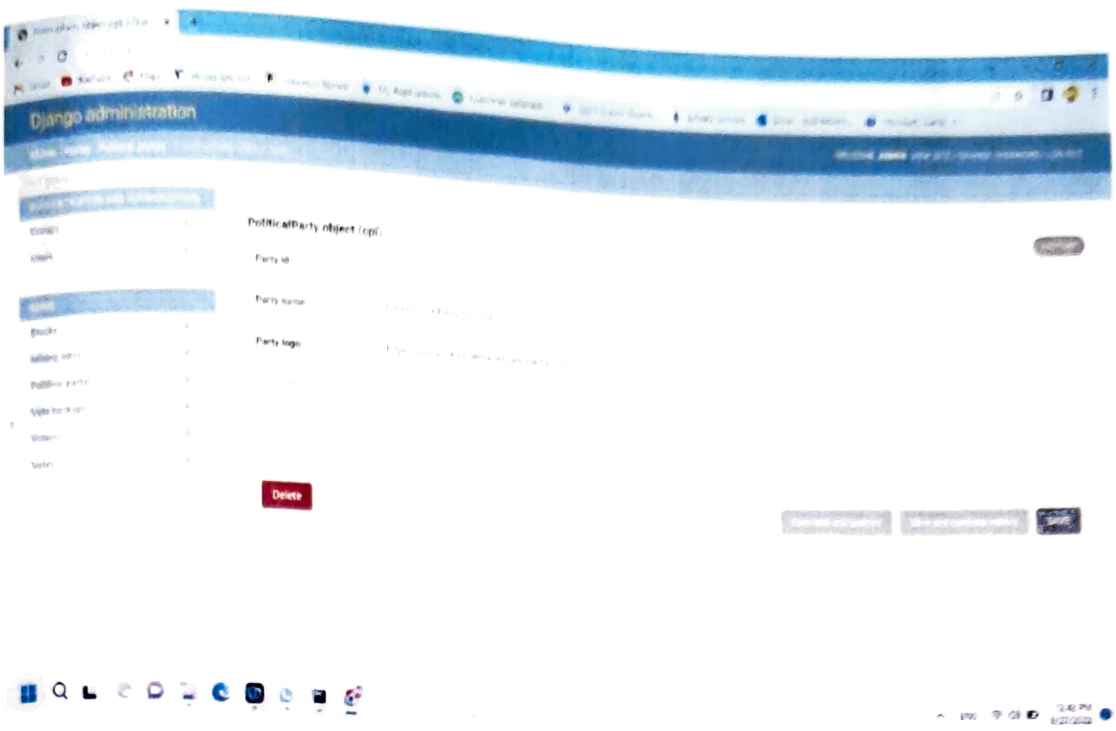


Figure A.7 To modify Political Party Details

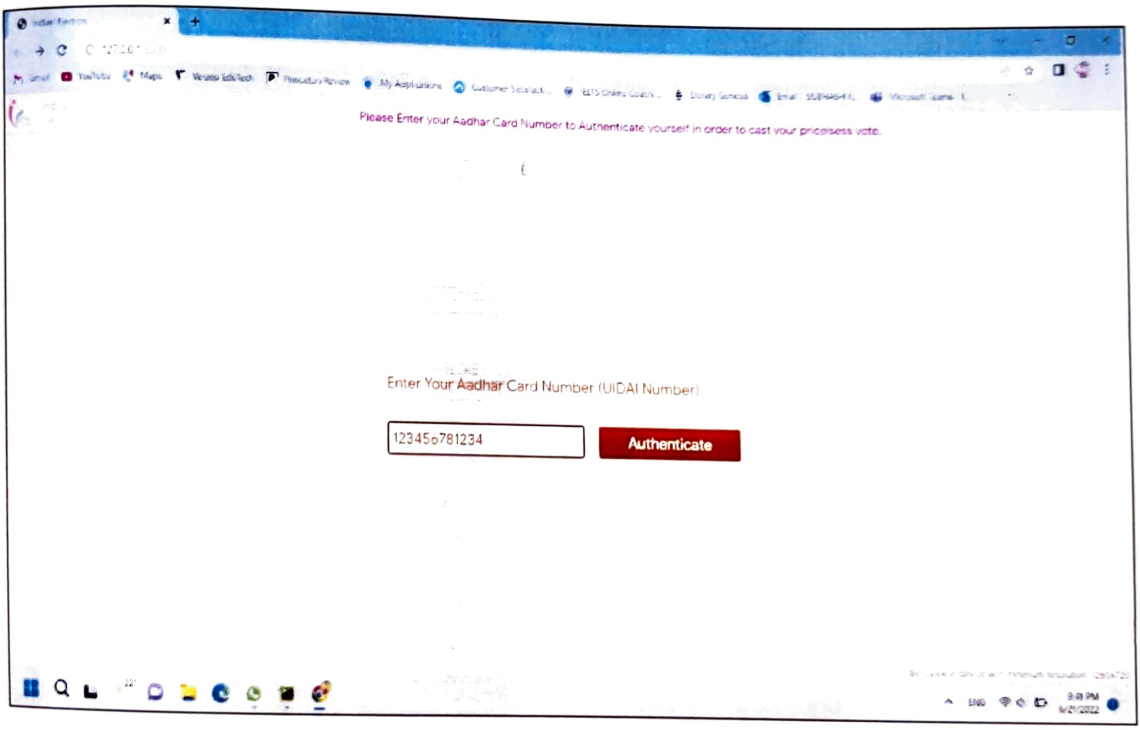


Figure A.8 Input Registered Aadhar Number

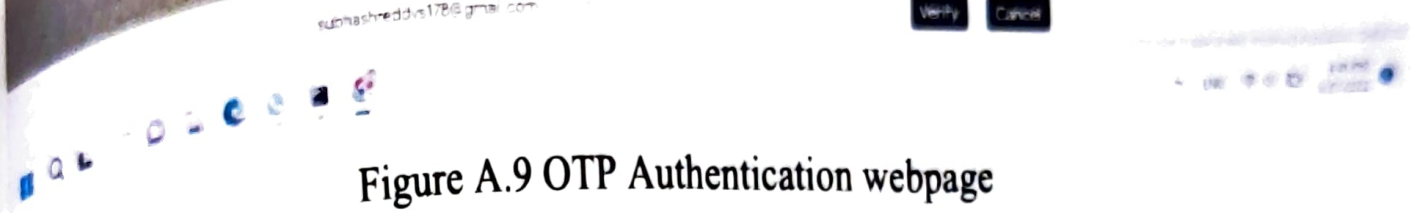


Figure A.9 OTP Authentication webpage

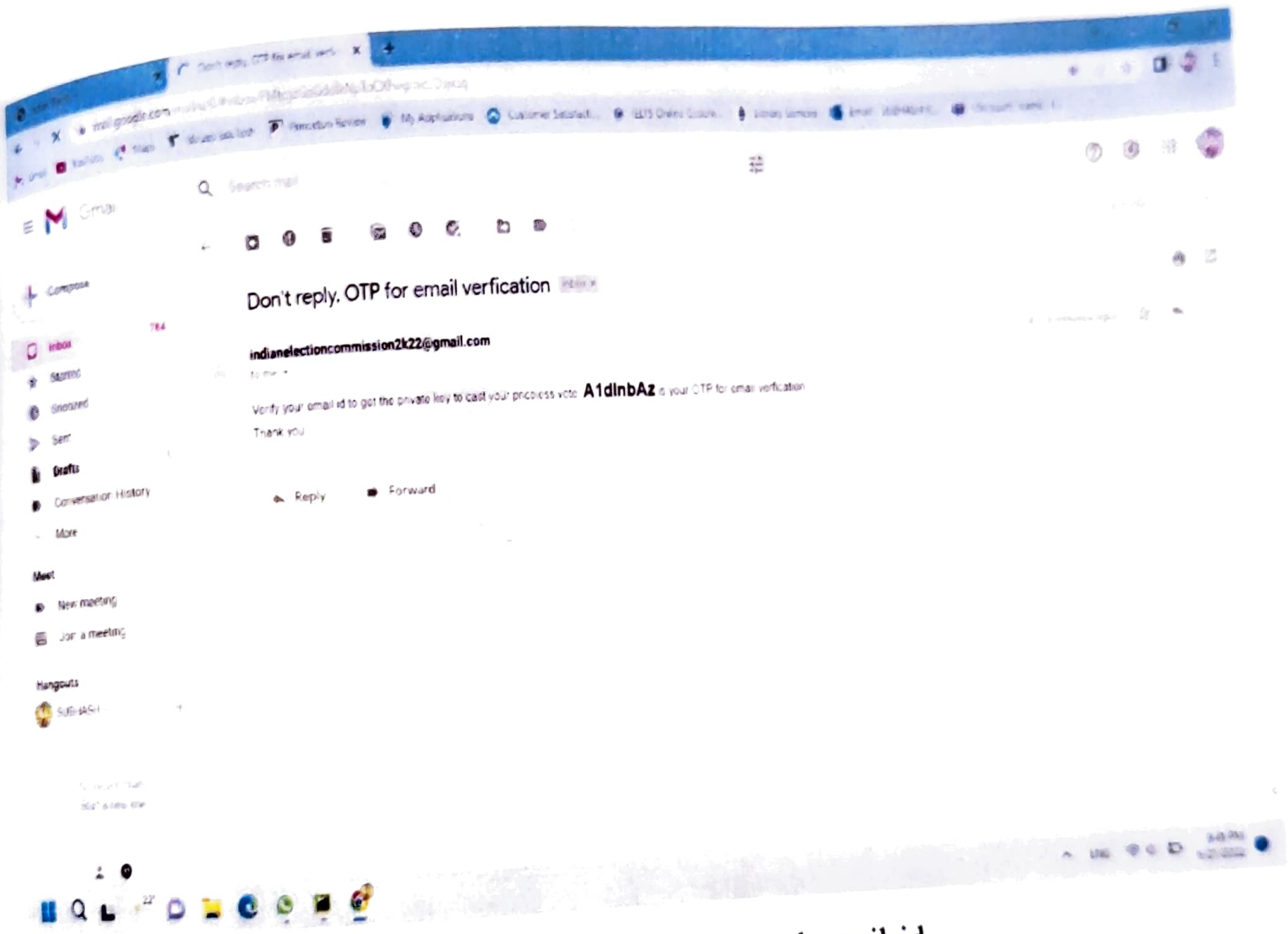


Figure A.10 sending OTP registered email-id

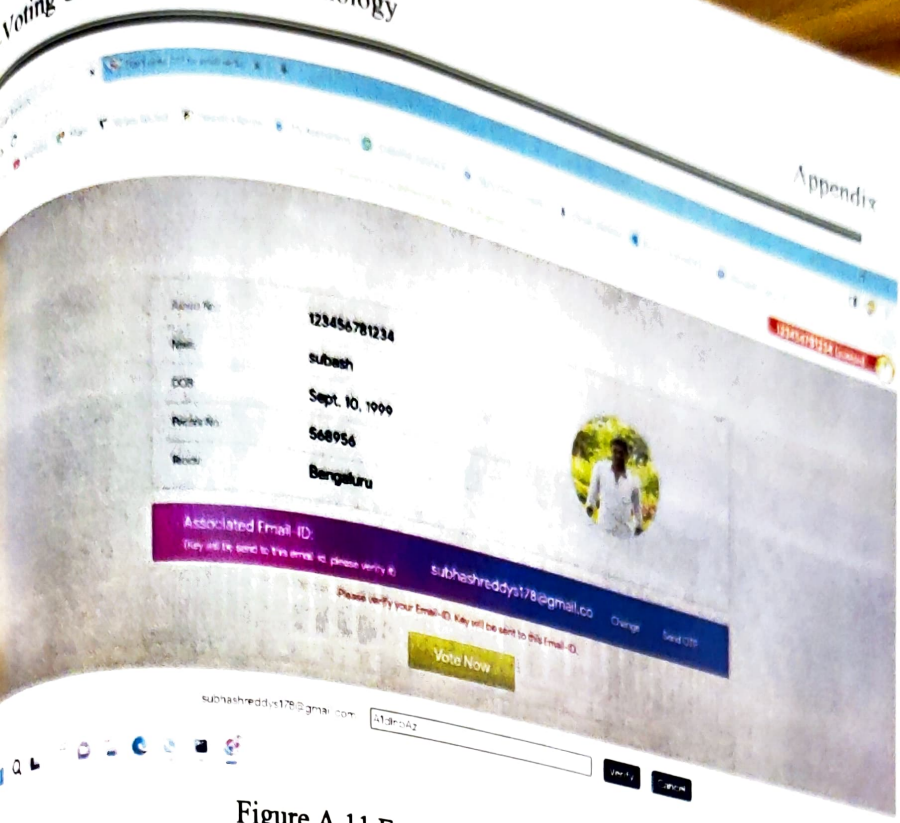


Figure A.11 Entering the received OTP

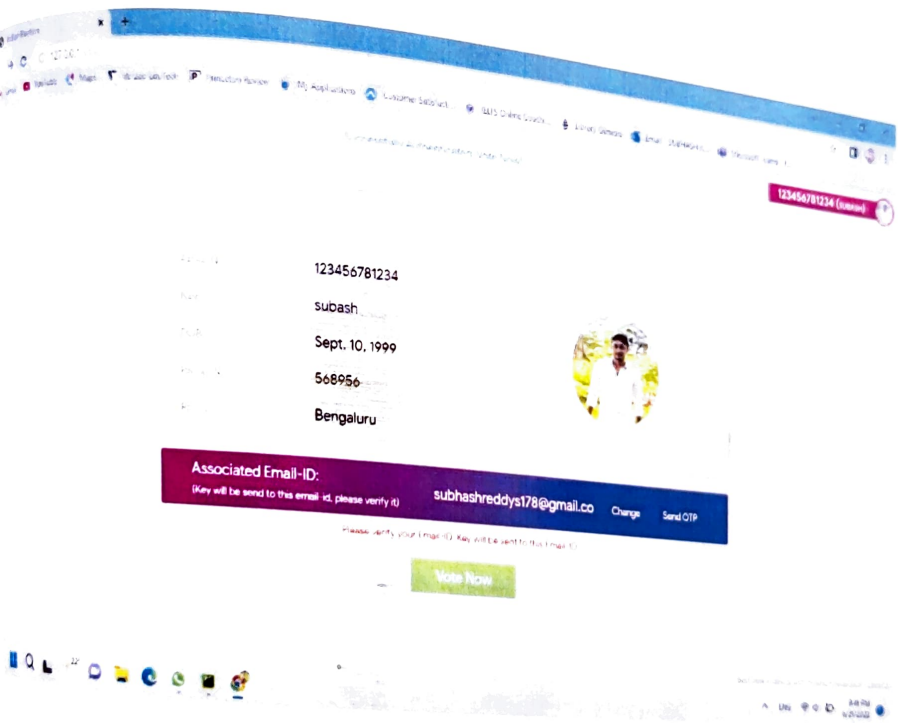


Figure A.12 Authorizing user to Vote

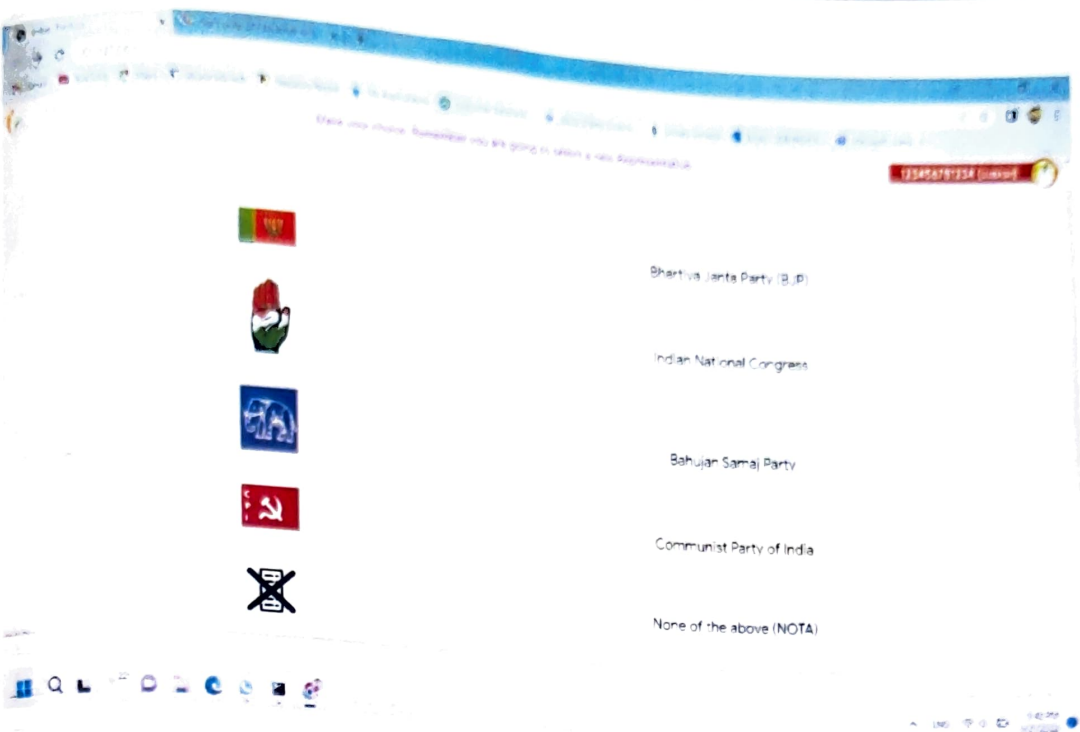


Figure A.13 User selecting his choice of party

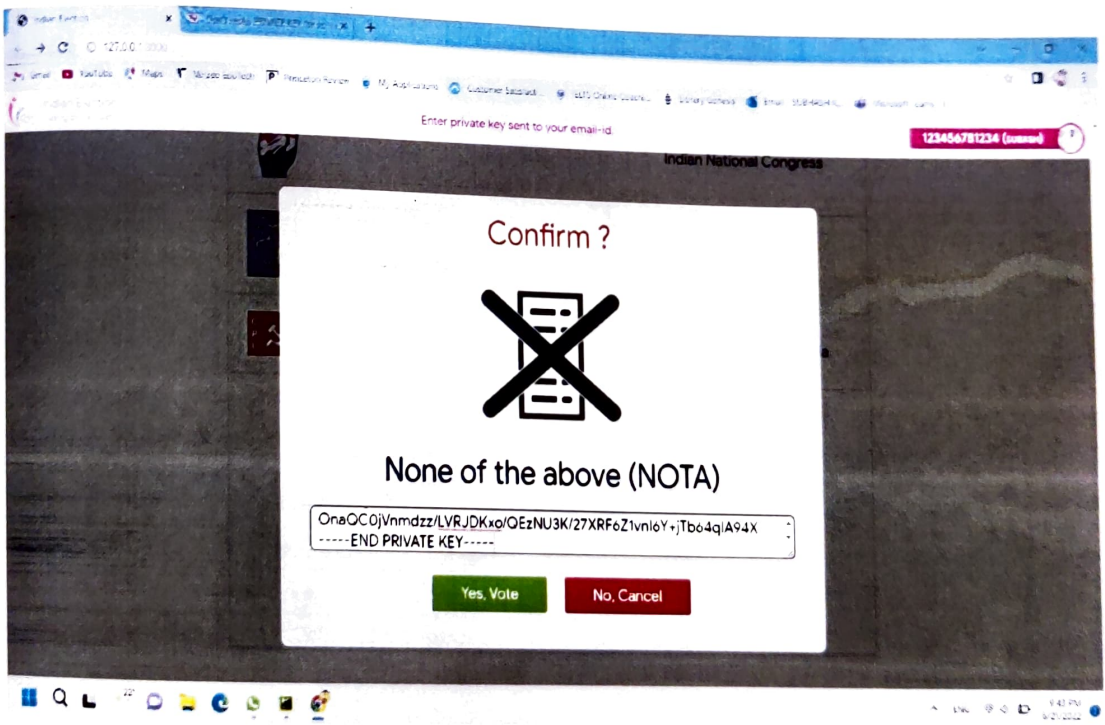


Figure A.14 User Entering the Private key

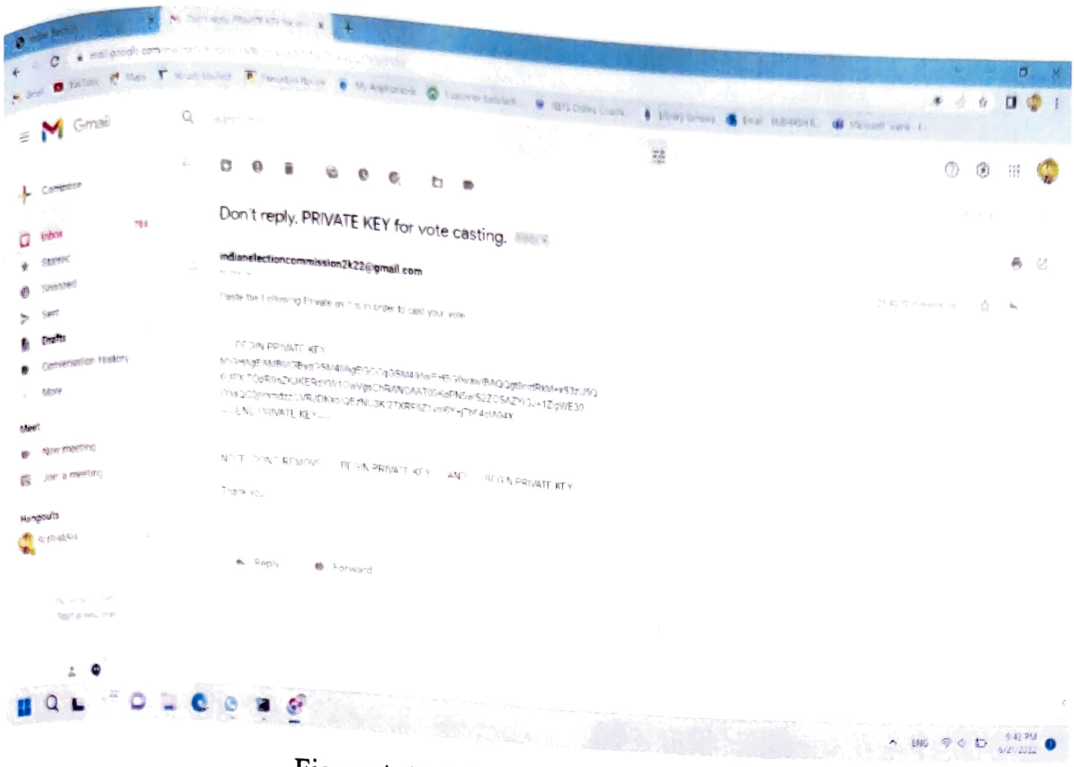


Figure A.15 Private key sent to email-id

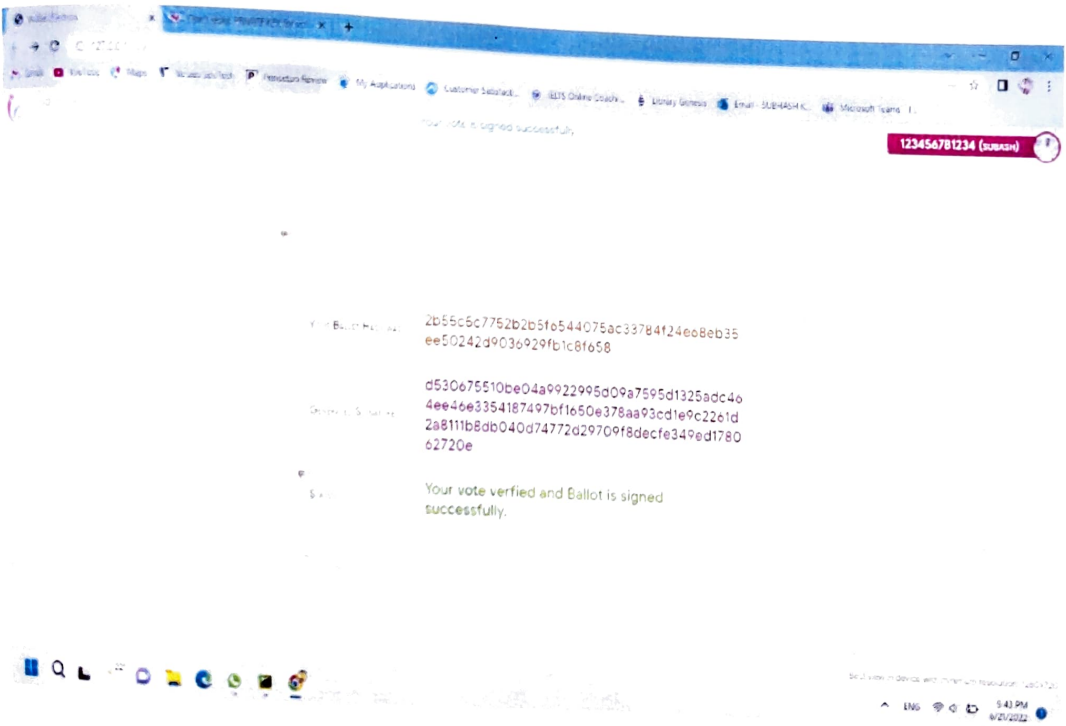


Figure A.16 Signing of Vote Successfully

Figure A.17 Vote Count

Appendix B: Abbreviations

SMTP: Simple Mail Transfer Protocol

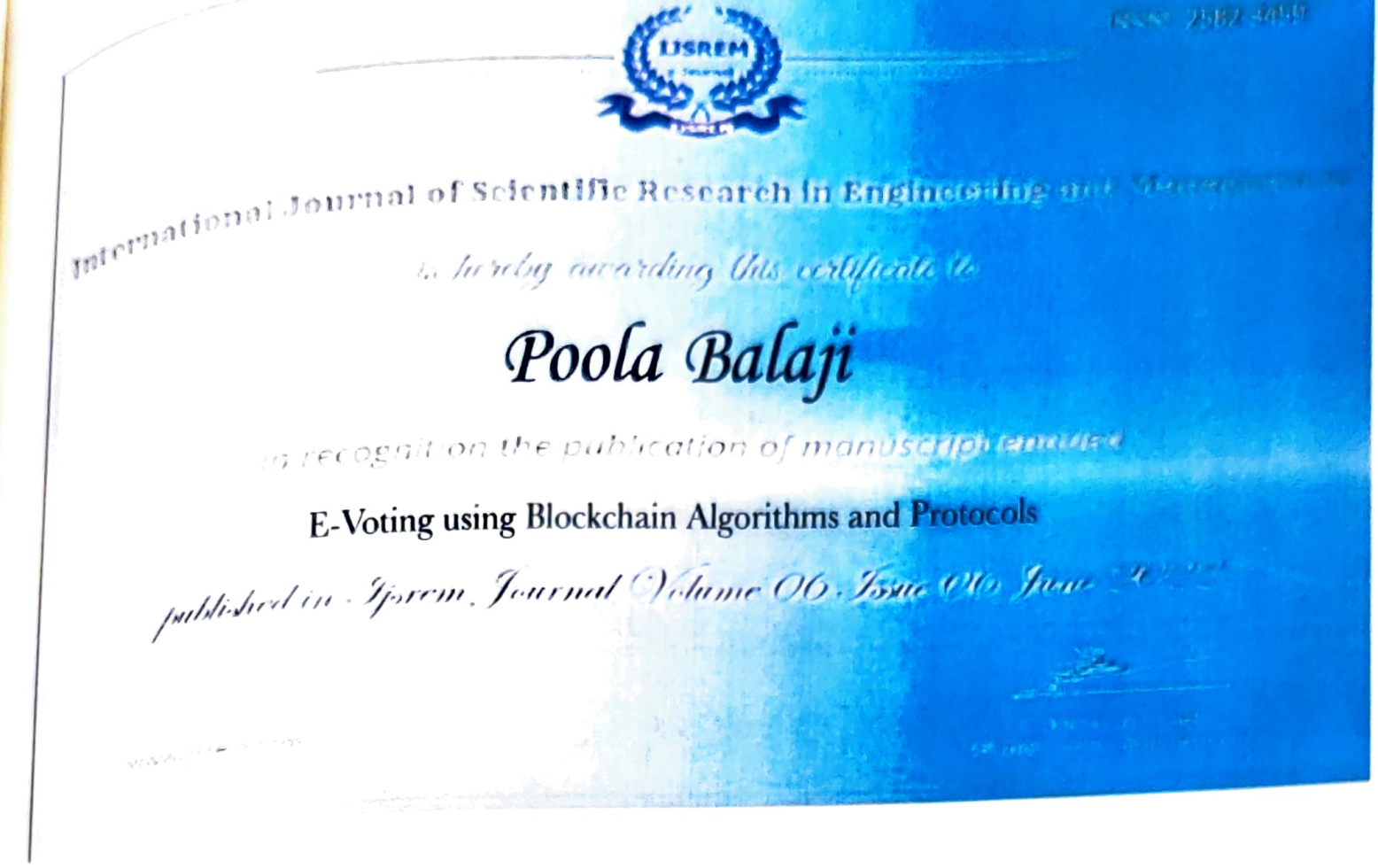
SHA-256: Secure Hash Algorithm 256 bit

MDA: Mail Delivery Agent

MUA: Mail User Agent

TCP: Transmission Control Protocol

MTA: Mail Transfer Agent


International Journal of Scientific Research in Engineering and Management
is hereby awarding this certificate to
Poola Balaji
in recognition the publication of manuscript entitled
E-Voting using Blockchain Algorithms and Protocols
published in Ijsrem Journal Volume 06 Issue 06 June


ISSN 2582-3459
International Journal of Scientific Research in Engineering and Management
is hereby awarding this certificate to
Subhash K V
in recognition the publication of manuscript entitled
E-Voting using Blockchain Algorithms and Protocols
published in Ijsrem Journal Volume 06 Issue 06 June

International Journal of Scientific Research in Engineering and Management

is hereby awarding this certificate to

Puttaparthi Tharun Sai

in recognition the publication of manuscript entitled

E-Voting using Blockchain Algorithms and Protocols

published in · Ijsrem Journal Volume 06 · Issue 06 June 2[illegible]

International Journal of Scientific Research in Engineering and Management

is hereby awarding this certificate to

Nandini L Reddy

in recognition the publication of manuscript entitled

E-Voting using Blockchain Algorithms and Protocols

published in · Ijsrem Journal Volume 06 · Issue 06 June 2[illegible]

ISSN 2582-3459