A Project Report on

SECURE INNOVATIVE VOTING SYSTEM WITH BIOMETRIC AND OTP AUTHENCTICATION

submitted in partial fulfillment of the requirement for the award of the Degree of

BACHELOR OF TECHNOLOGY

to

G. PULLAIAH COLLEGE OF ENGINEERING AND TECHNOLOGY

(Autonomous)

Approved by AICTE | NAAC Accreditation with 'A' Grade Accredited by NBA (CSE, ECE & EEE) | Permanently Affiliated to JNTUA Nandikotkur Road, Venkayapalli (V), Kurnool - 518452, Andhra Pradesh

by

SYED NOORULLAH BASHA (21AT5A3507)
BACHU GURU JASHWANTH (20AT1A3512)
VADLA JASWANTH ACHARI (20AT1A3518)
MUHAMMAD IMAAD UL HAQ (20AT1A3517)

Under the Guidance of

Mrs. N. S. SWAPNA M.Tech

Assistant Professor

GPCET
Pioneering Innovative Education

DEPARTMENT OF COMPUTER SCIENCE ENGINEERNG-

INTERNET OF THINGS

G. PULLAIAH COLLEGE OF ENGINEERING AND TECHNOLOGY (AUTONOMOUS)

(Approved by AICTE | NAAC Accreditation with 'A' Grade | Accredited by NBA (ECE,CSE &EEE) | Permanently Affiliated to JNTUA)

2020-2024

G. PULLAIAH COLLEGE OF ENGINEERING AND TECHNOLOGY

(Autonomous)

(Approved by AICTE | NAAC Accreditation with 'A' Grade |Accredited by NBA (CSE, ECE & EEE) | Permanently Affiliated to JNTUA)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING-

INTERNET OF THINGS

CERTIFICATE

This is to certify that the project report entitled **"Secure Innovative Voting System with Biometric and OTP Authentication"** being submitted by **Syed Noorullah Basha (21AT5A3507), Bachu Guru Jashwanth (20AT1A3512), Vadla Jaswanth Achari (20AT1A3518), Muhammad Imaad Ul Haq (20AT1A3517)** in partial fulfillment of the requirement for the award of the degree of Bachelor of Technology in Computer Science and Engineering-Internet of Things of Jawaharlal Nehru Technological University Anantapur, Ananthapuramu is a record of bonafide work carried out by them under my guidance and supervision.

Mrs. N.S. SWAPNA M.Tech.
Assistant Professor
Project Supervisor

Dr. P. SUMAN PRAKASH M. Tech., Ph.D.
Associate Professor
Head of the Department

Date of Viva-Voce_____

INTERNAL EXAMINER

EXTERNAL EXAMINER

DECLARTION

We hereby declare that this project work entitled "**Secure Innovative Voting System with Biometric and OTP Authentication"** in partial fulfillment of requirements for the award of degree of computer Science and engineering- Internet of Things is a bonafied work carried out by us during the academic year 2020-24.

We further declare that this project is a result of our effort and has not been submitted for the award of any degree by us to any institute.

By

SYED NOORULLAH BASHA (21AT5A3507)

BACHU GURU JASHWANTH (20AT1A3512)

VADLA JASWANTH ACHARI (20AT1A3518)

MUHAMMAD IMAAD UL HAQ (20AT1A3517)

ACKNOWLEDGEMENT

We are extremely grateful to Chairman, **Sri G.V.M.Mohan Kumar,** of G.Pullaiah College of Engineering and Technology, Kurnool, Andhra Pradesh for their good blessings.

We owe indebtedness to our Principal **Dr.C.SrinivasaRao, M.E., Ph.D.** G.Pullaiah College of Engineeringand Technology, Kurnool for providing us the required facilities.

We would like to express our deep sense of gratitude and our sincere thanks to **HOD Dr .P. Suman Prakash M.Tech., Ph.D.** Department of Internet of Things G.Pullaiah College of Engineering and Technology, Kurnool for providing the necessary facilities and encouragement towards the project work.

We thank our project supervisor, **Mrs. N. S. Swapna M.Tech** for her guidance, valuable suggestions and support in the completion of the project.

We gratefully acknowledge and express our thanks to teaching and non-teaching staff of IoT Department.

We would like to express our love and affection to our parents for their encouragement throughout thisproject work.

Project Associates

SYED NOORULLAH BASHA (21AT5A3507)

BACHU GURU JASHWANTH (20AT1A3512)

VADLA JASWANTH ACHARI (20AT1A3518)

MUHAMMAD IMAAD UL HAQ (20AT1A3517)

ABSTRACT

The Secure Innovative Voting System with Biometric and OTP Authentication represents a robust solution for enhancing the security and reliability of the voting process. This system integrates biometric authentication using a fingerprint module and strengthens security with one-time password (OTP) verification. Key components include a fingerprint module, servo motor, buzzer, LCD display, IR sensor, SIM900L module, and switches, all centrally controlled by an Arduino microcontroller. Upon successful fingerprint verification, an OTP is sent to the voter's mobile number through the SIM900L module for additional authentication. Voters proceed to cast their votes using switches, after which the servo motor secures the voting compartment, preventing unauthorized access. The LCD display and buzzer offer intuitive guidance and feedback to users throughout the process. By combining biometric and OTP authentication mechanisms, the system ensures the integrity and transparency of elections, safeguarding against fraud and manipulation.

Keywords: Smart Voting System, Fingerprint Authentication, OTP Authentication, Arduino, SIM900L Module, Biometric Security, Secure Elections.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF SCREENSHOTS

CHAPTER 1

INTRODUCTION

The democratic process stands as the cornerstone of modern governance, embodying principles of representation and citizen participation. Central to this process is the electoral system, where citizens exercise their right to vote and elect their representatives. However, traditional voting methods have faced challenges such as logistical complexities, long queues, and concerns about security and transparency. In response to these challenges, there is a growing interest in leveraging technology to modernize the voting process and address these issues.

One promising technology that has gained traction in recent years is the use of biometric authentication, specifically fingerprint recognition, in voting systems. Biometric authentication offers a secure and reliable means of verifying voters' identities, reducing the risk of fraud and impersonation. By integrating fingerprint recognition into voting systems, governments and election authorities aim to enhance the security, integrity, and efficiency of the electoral process.

Moreover, alongside biometric authentication, the integration of additional security measures such as one-time password (OTP) authentication further strengthens the voting system's resilience against unauthorized access and fraudulent activities. OTP authentication adds an extra layer of verification by sending a unique code to voters' mobile devices, ensuring that only authorized individuals can cast their votes.

In this context, the Smart Voting System with Fingerprint and OTP Authentication emerges as a comprehensive solution to modernize the electoral process. By combining biometric and OTP authentication mechanisms with advanced hardware components and the Arduino microcontroller platform, this system offers a secure, transparent, and user-friendly voting experience. In the following sections, we delve into the components, functionalities, and implementation of the Smart Voting System, highlighting its potential to revolutionize the way elections are conducted while upholding the principles of democracy and civic engagement.

The integration of biometric and OTP authentication mechanisms in the Smart Voting System represents a significant advancement in electoral technology, addressing longstanding challenges and enhancing the credibility of the electoral process. By leveraging fingerprint recognition and OTP verification, the system ensures secure and reliable authentication of voters, mitigating the risks of fraud, impersonation, and manipulation. Moreover, the use of advanced

hardware components such as servo motors, buzzers, and LCD displays, controlled by the Arduino microcontroller, offers a seamless and intuitive voting experience for users.

1.1 Introduction To Embedded Systems

An embedded system can be defined as a computing device that does a specific focused job. Appliances such as the air-conditioner, VCD player, DVD player, printer, fax machine, mobile phone etc. are examples of embedded systems. Each of these appliances will have a processor and special hardware to meet the specific requirement of the application along with the embedded software that is executed by the processor for meeting that specific requirement. The embedded software is also called “firm ware”. The desktop/laptop computer is a general purpose computer. You can use it for a variety of applications such as playing games, *word* processing, accounting, software development and so on. In contrast, the software in the embedded systems is always fixed listed below:

Embedded systems do a very specific task, they cannot be programmed to do different things. Embedded systems have very limited resources, particularly the memory. Generally, they do not have secondary storage devices such as the CDROM or the floppy disk. Embedded systems have to work against some deadlines. A specific job has to be completed within a specific time. In some embedded systems, called real-time systems, the deadlines are stringent. Missing a deadline may cause a catastrophe-loss of life or damage to property. Embedded systems are constrained for power. As many embedded systems operate through a battery, the power consumption has to be very low. Some embedded systems have to operate in extreme environmental conditions such as very high temperatures and humidity.

1.2 Advantages of Embedded systems

1. They are designed to do a specific task and have real time performance constraints which must be met.
2. They allow the system hardware to be simplified so costs are reduced.
3. They are usually in the form of small computerized parts in larger devices which serve a general purpose.

1.3 Application Areas

Nearly 99 per cent of the processors manufactured end up in embedded systems. The embedded system market is one of the highest growth areas as these systems are used in very market segment- consumer electronics, office automation, industrial automation, biomedical engineering, wireless communication, data communication, telecommunications, transportation, military and so on.

1.4 Introduction IOT (Internet Of Things)

Connecting everyday things embedded with electronics, software, and sensors to internet enabling to collect and exchange data without human interaction called as the Internet of Things (IoT).The term "Things" in the Internet of Things refers to anything and everything in day to day life which is accessed or connected through the internet.

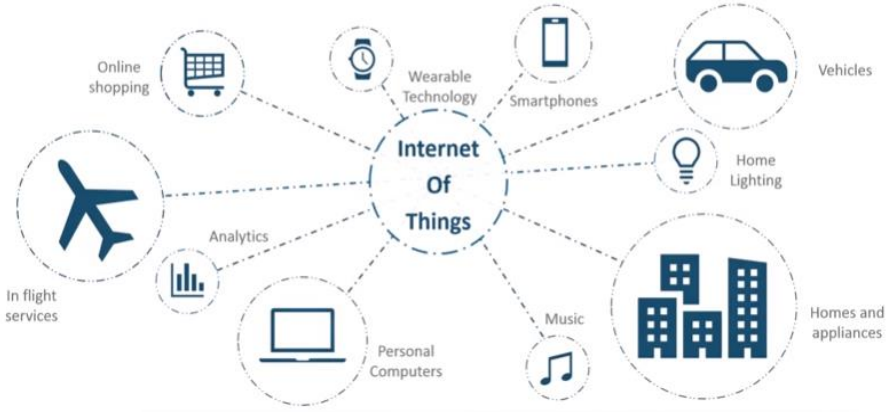


Fig 1.1: Internet of Things

IoT is an advanced automation and analytics system which deals with artificial intelligence, sensor, networking, electronic, cloud messaging etc. to deliver complete systems for the product or services. The system created by IoT has greater transparency, control, and performance. As we have a platform such as a cloud that contains all the data through which we connect all the things around us. For example, a house, where we can connect our home appliances such as air conditioner, light, etc. through each other and all these things are managed at the same platform. Since we have a platform, we can connect our car, track its fuel meter, speed level, and also track the location of the car.

1.5 Characteristics Of IOT

- Massively scalable and efficient
- IP-based addressing will no longer be suitable in the upcoming future.
- An abundance of physical objects is present that does not use IP, so IoT is made possible.
- Devices typically consume less power. When not in use, they should be automatically programmed to sleep.
- A device that is connected to another device right now may not be connected in another instant of time.
- Intermittent connectivity – IoT devices aren't always connected. In order to save bandwidth and battery consumption, devices will be powered off periodically when not in use. Otherwise, connections might turn unreliable and thus prove to be inefficient.

1.6 Advantages And Disadvantages Of IOT

Any technology available today has not reached to its 100 % capability. It always has a gap to go. So, we can say that **Internet of Things** has a significant technology in a world that can help other technologies to reach its accurate and complete 100 % capability as well.

Internet of things facilitates the several advantages in day-to-day life in the business sector. Some of its benefits are

- **Efficient resource utilization:** If we know the functionality and the way that how each device work we definitely increase the efficient resource utilization as well as monitor natural resources.
- **Minimize human effort:** As the devices of IoT interact and communicate with each other and do lot of task for us, then they minimize the human effort.
- **Save time:** As it reduces the human effort then it definitely saves out time. Time is the primary factor which can save through IoT platform.

- **Improve security:** Now, if we have a system that all these things are interconnected then we can make the system more secure and efficient.

1.7 Disadvantages of IoT

As the Internet of things facilitates a set of benefits, it also creates a significant set of challenges. Some of the IoT challenges are

- **Security:** As the IoT systems are interconnected and communicate over networks. The system offers little control despite any security measures, and it can be lead the various kinds of network attacks.
- **Privacy:** Even without the active participation on the user, the IoT system provides substantial personal data in maximum detail.
- **Complexity:** The designing, developing, and maintaining and enabling the large technology to IoT system is quite complicated.

CHAPTER 2

LITERATURE SURVEY

M. A. Uddin, A. Stranieri, I. Gondal and V. Balasubramanian, "A Decentralized Patient Agent Controlled Blockchain for Remote Patient Monitoring,"

Blockchain emerging for healthcare provides a secure, decentralized and patient driven record management system. However, the storage of data generated from IoT devices in remote patient management applications requires a fast consensus mechanism. In this paper, we propose a lightweight consensus mechanism and a decentralized patient software agent to control a remote patient monitoring (RPM) system. The decentralized RPM architecture includes devices at three levels; 1) Body Area Sensor Network- medical sensors typically on or in patient's body transmitting data to a Smartphone, 2) Fog/Edge, and 3) Cloud. We propose that a Patient Agent(PA) software replicated on the Smartphone, Fog and Cloud servers processes medical data to ensure reliable, secure and private communication. Performance analysis has been conducted to demonstrate the feasibility of the proposed Blockchain leveraged, distributed Patient Agent controlled remote patient monitoring system.

S. B, R. T. V, N. Krishna M P, B. R. J, S. Arvindh M and D. M. Alagappan, "Secured Electronic Voting System Using the Concepts of Blockchain,"

Electronic Voting Machines (EVMs) are replacing paper ballots because of (i) the errors involved in the manual counting process and (ii) the large time period required to count the votes. Even though these digital recording electronic systems are advancements, they are prone to tampering and electoral frauds. The suspected vulnerabilities in EVMs are (i) the possibilities of tampering with the EVM's memory chip or switching it with a fake one, (ii) their simplicity, which enables them to be tampered without requiring much skill and (iii) the chances of double voting. This paper attempts to solve the above problems by storing the vote data in a decentralized network and securing the data using a security mechanism derived from the blockchain framework (which is currently used for securing cryptocurrencies). The vote data is shared among all the devices in the network and peer to peer verification is done to verify the authenticity of the vote data. In order

to successfully tamper with the system, the data stored in all the nodes must be changed. This makes the proposed system more efficient and reliable. In this paper, we propose an algorithm, which is based on the concepts of blockchain, to secure the votes stored in an EVM. A fingerprint authentication system is also added as an extra layer of security to prevent double voting. The proposed EVM will be tamperproof, and any attempt to manipulate the registered votes, will be detected and the tampered data will be replaced with the correct data.

X. I. Selvarani, M. Shruthi, R. Geethanjali, R. Syamala and S. Pavithra, "Secure voting system through SMS and using smart phone application,"

Mobile voting system is used to cast their votes in secure manner. Previously the votes were casted through the traditional methods of polling booths, punch cards, lever voting, optical voting machine, which are now replaced through some electronic mediums. All of these consumes more time to cast their votes. The proposed system are developed to select their candidate through smart phone application. This process consists of three steps: online registration of voter, vote casting of voter and display of results, through the concept of SMS (short messaging service). It provides more efficiency for voters to cast their vote from anywhere, at any time through internet. The important aspect of this is to provide more security till the core, since every vote counts and each of the votes are to be remained confidential. This prevents voters to cast their vote more than once with the use of OTP (one time password) for every sign in and login. It also reduces the paper work and eliminates the manual counting process. Here the security is provided through the RSA encryption algorithm.

B. Rogers, S. Chhabra, M. Prvulovic and Y. Solihin, "Using Address Independent Seed Encryption and Bonsai Merkle Trees to Make Secure Processors OS- and Performance-Friendly,"

In today's digital world, computer security issues have become increasingly important. In particular, researchers have proposed designs for secure processors which utilize hardware-based memory encryption and integrity verification to protect the privacy and integrity of computation

even from sophisticated physical attacks. However, currently proposed schemes remain hampered by problems that make them impractical for use in today's computer systems: lack of virtual memory and inter-process communication support as well as excessive storage and performance overheads. In this paper, we propose 1) address independent seed encryption (AISE), a counter-mode based memory encryption scheme using a novel seed composition, and 2) Bonsai Merkle trees (BMT), a novel Merkle tree-based memory integrity verification technique, to eliminate these system and performance issues associated with prior counter-mode memory encryption and Merkle tree integrity verification schemes. We present both a qualitative discussion and a quantitative analysis to illustrate the advantages of our techniques over previously proposed approaches in terms of complexity, feasibility, performance, and storage. Our results show that AISE+BMT reduces the overhead of prior memory encryption and integrity verification schemes from 12% to 2% on average, while eliminating critical system-level problems

R. Bosri, A. R. Uzzal, A. A. Omar, A. S. M. T. Hasan and M. Z. A. Bhuiyan, "Towards a Privacy-Preserving Voting System Through Blockchain Technologies,"

The voting system is the process to take the opinion of people to run the constitution properly. Fairness, independence, and unbiasedness should be present in the voting system. Hence, it must be a transparent and secured process so that everybody can express their own opinion freely. Worldwide vote manipulation is an intriguing problem in existing voting systems. Since people in different countries are using digital technology in the voting process (e.g., Optical Scan Voting system, Internet Voting system, Electronic Voting system) instead of traditional way (e.g., Ballot Box). Only digitization could not solve the issues completely. Because still there are numerous ways to manipulate or tamper digital technology and hamper the voting process. To build a secure electronic voting environment, we introduce an application of blockchain technology as a service for the distributed electronic voting system. With the use of blockchain, we achieve data integrity which is a necessary attribute of a voting environment. The anonymity of the voters, privacy, and security of the voting environment is the main goal of this work. Through the design of our system and with the help of blockchain we have solved all the security issues in the voting environment.

R. Bulut, A. Kantarcı, S. Keskin and Ş. Bahtiyar, "Blockchain-Based Electronic Voting System for Elections in Turkey,"

Traditional elections satisfy neither citizens nor political authorities in recent years. They are not fully secure since it is easy to attack votes. It threatens also privacy and transparency of voters. Additionally, it takes too much time to count the votes. This paper proposes a solution using Blockchain to eliminate all disadvantages of conventional elections. Security and data integrity of votes is absolutely provided theoretically. Voter privacy is another requirement that is ensured in the system. Lastly, waiting time for results decreased significantly in proposed Blockchain voting system.

CHAPTER 3

SYSTEM ANALYSIS

In the system analysis phase of the project, a comprehensive understanding of the requirements, constraints, and objectives is established. This phase serves as the foundation for the entire development process, ensuring that the final system meets the needs of the project effectively and efficiently.

3.1 Software Development Lifecycle (SDLC):

The Software Development Lifecycle (SDLC) is a structured approach to software development that outlines the stages involved in building and maintaining software systems. Our project follows the SDLC model to ensure systematic and organized development. The SDLC stages include:

1. **Requirement Analysis:**
    - Identify and document the functional and non-functional requirements of the system.
    - Conduct interviews, surveys, and workshops with stakeholders to gather requirements.
    - Define the scope, objectives, and constraints of the project.
2. **System Design:**
    - Develop a detailed system architecture based on the gathered requirements.
    - Design the user interface, database schema, and system components.
    - Create prototypes or mockups to visualize the system's appearance and functionality.
3. **Implementation:**
    - Write code according to the design specifications and programming standards.
    - Test individual modules for functionality and integration.
    - Address any bugs or issues identified during the testing phase.

4. **Testing:**

    - Execute various testing strategies such as unit testing, integration testing, and system testing.
    - Validate that the system meets the specified requirements and performs as expected.
    - Conduct user acceptance testing (UAT) to ensure that the system meets the needs of end-users.

5. **Deployment:**

    - Prepare the system for deployment in the production environment.
    - Install the necessary hardware and software components.
    - Train end-users and administrators on how to use and maintain the system.

6. **Maintenance:**

    - Provide ongoing support and maintenance to address any issues or updates.
    - Monitor system performance and make enhancements as needed.
    - Conduct periodic reviews and evaluations to ensure the system's effectiveness.

3.2 Additional Considerations

1. **Technology Stack:**

    - Describe the technologies and tools used in the development process, such as programming languages, frameworks, and databases.

2. **Security Measures:**

    - Outline the security measures implemented to protect sensitive data and ensure the integrity of the system, including encryption, access controls, and authentication mechanisms.

3. **Data Management:**

    - Discuss how data is collected, stored, processed, and managed within the system, including database design, data validation, and backup procedures.

4. **Integration and Interoperability:**

    - Explain how the system interacts with external systems or services, including APIs, protocols, and data exchange formats.

5. **Scalability and Performance:**

    - Address the scalability and performance considerations of the system, including load balancing, caching, and optimization techniques.

6. **Regulatory Compliance:**

    - Ensure that the system complies with relevant regulations, standards, and industry best practices, such as data privacy laws and accessibility requirements.

7. **User Training and Documentation:**

    - Develop user manuals, training materials, and documentation to help users understand and utilize the system effectively.

8. **Risk Management:**

    - Identify potential risks and mitigation strategies throughout the development lifecycle, including risk assessment, monitoring, and contingency planning.

By addressing these topics in the system analysis phase, we lay the groundwork for a successful development process that delivers a robust, secure, and user-friendly system that meets the needs of our project.

3.3 Existing System

Existing voting systems have long relied on conventional methods such as paper-based ballots or electronic voting machines (EVMs) to conduct elections. These methods, while familiar and widely used, come with their set of limitations and challenges. Paper-based ballots, for instance, require manual counting by election officials, which can be time-consuming and prone to errors. Moreover, they may be susceptible to tampering or fraud, raising concerns about the integrity of the election process. On the other hand, EVMs offer advantages in terms of faster

counting and reduced paper usage. However, they have faced criticism over issues such as security vulnerabilities, reliability, and transparency, with instances of malfunctioning machines and hacking casting doubts on the credibility of election results.

One of the primary challenges faced by existing voting systems is security. Both paper-based ballots and EVMs are vulnerable to various security threats, including tampering, hacking, and fraud. These vulnerabilities can undermine the integrity of the election process and erode public trust in democratic institutions. Furthermore, the lack of transparency in some voting systems, particularly EVMs, where the inner workings are not always visible or auditable, has raised concerns about the verifiability of election results. Without robust security measures and transparency mechanisms in place, the credibility of election outcomes may be called into question, leading to doubts and disputes.

Legal and regulatory frameworks governing existing voting systems vary widely across different jurisdictions. These frameworks dictate procedures for voter registration, ballot casting, counting, and result declaration, and may differ significantly in terms of transparency, accountability, and accessibility. While some countries have robust legal frameworks in place to ensure free and fair elections, others may lack adequate safeguards against electoral fraud or manipulation. Inconsistent or ambiguous regulations can lead to confusion, disputes, and legal challenges, undermining the legitimacy of election outcomes and threatening political stability. Public perception and trust in the electoral process are essential for the functioning of democracy. Any perceived flaws or irregularities in the voting process can erode public confidence in democratic institutions and undermine faith in the electoral system. Instances of electoral fraud, manipulation, or mismanagement can lead to protests, unrest, and even violence, posing significant challenges to the stability and legitimacy of governments. Therefore, ensuring the integrity, transparency, and credibility of the electoral process is essential for upholding democratic values and principles.

3.3.1 Disadvantages of Existing System:

1. Insufficient authentication methods: Existing systems often rely on outdated or inadequate authentication methods, leading to concerns regarding the integrity of voter identities.

2. Vulnerability to manipulation: Traditional voting systems are susceptible to various forms of manipulation, such as booth capturing and rigging, which undermine the fairness and transparency of the electoral process.

3. Security risks: The reliance on fingerprint-based authentication alone poses security risks, as these systems can be compromised through the forging of duplicate fingerprints, potentially allowing unauthorized individuals to cast votes.

3.4 Proposed System

Proposed Smart Voting System with Fingerprint and OTP Authentication offers a modernized and secure approach to conducting elections, addressing the shortcomings of traditional voting systems. This system integrates advanced technologies such as biometric authentication and one-time password (OTP) verification to enhance security, transparency, and efficiency in the electoral process.

Central to the proposed system is the utilization of biometric authentication through fingerprint recognition. By incorporating a fingerprint module, voters are required to authenticate their identity using their unique fingerprints before casting their votes. This biometric authentication mechanism provides a robust and reliable means of verifying voters' identities, significantly reducing the risk of fraud, impersonation, and multiple voting.

In addition to biometric authentication, the proposed system integrates OTP verification as an additional layer of security. Upon successful fingerprint authentication, an OTP is sent to the voter's registered mobile number using the SIM900L module. The voter must then enter the OTP via the keypad interface to proceed with casting their vote. This two-factor authentication process adds an extra layer of verification, ensuring that only authorized individuals can participate in the voting process.

Furthermore, the proposed system includes various hardware components such as a servo motor, buzzer, LCD display, IR sensor, and switches, all controlled by an Arduino microcontroller. The servo motor is used to secure the voting compartment, preventing unauthorized access and ensuring the integrity of the voting process. The buzzer and LCD display provide feedback and guidance to voters, making the voting process intuitive and user-friendly. The IR sensor detects

the presence of voters, triggering the authentication process, while switches allow voters to select their preferred candidates.

Overall, the proposed Smart Voting System with Fingerprint and OTP Authentication offers a comprehensive solution for modernizing the electoral process. By leveraging advanced biometric and OTP authentication mechanisms, combined with innovative hardware components and the Arduino microcontroller platform, this system ensures secure, transparent, and tamper-proof elections. With its emphasis on security, integrity, and user-friendliness, the proposed system represents a significant advancement in electoral technology, paving the way for fair and credible elections in the digital age.

3.4.1 Advantages of the proposed system:

1. Enhanced security: Incorporating biometric authentication and OTP verification adds an extra layer of security, reducing the risk of identity fraud and unauthorized access.

2. Real-time authentication: The system enables real-time verification of voter identities, ensuring that only legitimate voters are allowed to cast their ballots.

3. Swift response to irregularities: Immediate GSM alerts facilitate rapid response to any irregularities or attempted fraud, enabling election authorities to take prompt action to safeguard the integrity of the voting process.

4. Improved efficiency: By automating the authentication process and streamlining voter verification, the system enhances the efficiency of the voting process, reducing waiting times and improving overall voter experience.

5. Restoration of trust: By addressing the shortcomings of traditional voting systems and implementing robust security measures, the proposed system helps restore public trust in the electoral process, ensuring fair and transparent elections.

CHAPTER 4

SYSTEM DESIGN

The system design of our Smart Fingerprint-Based Voting System encompasses several key components and functionalities aimed at ensuring robustness, reliability, and efficiency in the electoral process. The design incorporates a modular architecture to facilitate scalability and flexibility, allowing for seamless integration of various hardware and software elements.

4.1 KEY COMPONENTS:

1. **Fingerprint Recognition Module:** This module is responsible for capturing and authenticating voters' fingerprints during the enrollment and voting processes. It utilizes advanced algorithms to ensure accurate identification and verification.

2. **OTP Authentication Mechanism:** The OTP authentication mechanism enhances security by generating one-time passwords that voters must enter to validate their identity before casting their votes. This additional layer of authentication adds an extra level of security to the system.

3. **GSM Communication Interface:** The GSM communication interface enables real-time communication between the voting system and election authorities. It facilitates the transmission of alerts and notifications, ensuring rapid response to any irregularities or security breaches.

4. **User Interface:** The user interface provides a user-friendly interaction platform for voters, election officials, and administrators. It includes features such as LCD displays, keypad inputs, and status indicators to guide users through the voting process and provide feedback on system status.

4.2 SYSTEM WORKFLOW:

1. **Enrollment Process:** During the enrollment process, eligible voters register their fingerprints and mobile numbers in the system. This information is securely stored in the database for future authentication.

2. **Voting Process:** When voters arrive to cast their votes, they undergo a two-step authentication process. First, they scan their fingerprints using the fingerprint recognition module. If the fingerprint matches the registered data, an OTP is generated and sent to their registered mobile number. The voter must then enter the OTP to validate their identity and proceed to cast their vote.

3. **Alert Mechanism:** In case of any discrepancies or security breaches, such as mismatched fingerprints or invalid OTPs, the system triggers an alert via the GSM communication interface. This alert notifies election authorities in real-time, enabling prompt action to resolve the issue and maintain the integrity of the voting process.

4.3 OVERALL DESIGN GOALS:

- **Security:** The system design prioritizes security by implementing multi-factor authentication mechanisms and real-time monitoring capabilities.

- **Efficiency:** The design aims to streamline the voting process, reducing waiting times and minimizing administrative overhead.

- **Reliability:** Robustness and reliability are central to the system design, ensuring consistent performance and accurate results under various operating conditions.

4.4 CIRCUIT DIAGRAM:

1. **Arduino Uno:** At the core of the system lies the Arduino Uno microcontroller board, serving as the central processing unit. It manages the operation of all other components and executes the voting logic based on the programmed instructions.

2. **AS608 Optical Fingerprint Reader Sensor Module:** This module interfaces with the Arduino Uno and is responsible for capturing and processing the fingerprints of voters during enrollment and voting. It communicates with the Arduino Uno through serial communication protocols to transmit fingerprint data for authentication.

3. **SIM800C GSM Module:** The GSM module facilitates communication between the voting system and external devices, such as mobile phones or election authorities' systems. It

enables the transmission of alerts and notifications in real-time, ensuring swift responses to any irregularities or security breaches detected by the system.

4. **LCD Display:** Serving as the output interface, the LCD display presents instructions, prompts, and status messages to voters during enrollment and voting. It offers visual feedback, guiding users through the voting process and indicating the current system status.

5. **Push Button Switches:** These switches perform various control functions within the system, such as initiating the enrollment process, starting the voting process, or triggering manual alerts in emergencies or system malfunctions.

6. **Servo Motor:** The servo motor controls the opening and closing of the door to the voting booth or compartment. Upon successful voter authentication, it activates to grant access to the voting area for casting ballots.

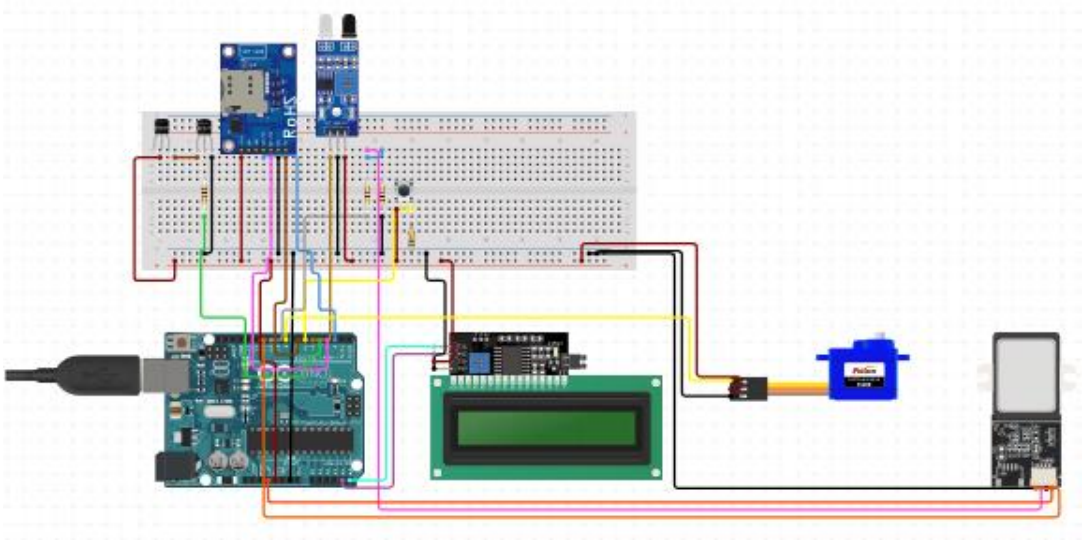


Fig 4.1: Circuit Diagram

4.5 WORKFLOW DIAGRAMS:

Workflow diagrams visually represent the sequential flow of tasks, actions, or processes within a system or project. In the context of our project, workflow diagrams serve as graphical representations of the various stages and interactions involved in the voting process. The workflow diagrams illustrates the step-by-step process of initializing the system, enrolling

voters, authenticating them using fingerprints and OTPs, enabling voting, recording data, and maintaining security measures throughout the process. Adjustments can be made based on specific functionalities or interactions within our project.

Here's the explanation of workflow diagrams:

Initialization Phase:

- Power-up sequence: The system undergoes a series of checks and initializations to ensure all components are functioning correctly. This includes checking power sources, initializing microcontrollers, and calibrating sensors.
- System initialization: Once powered up, the system initializes its software components, establishes connections with peripheral devices, and verifies communication channels.

Voter Registration:

- Voter presents for registration: Individuals seeking voter registration approach the system.
- Fingerprint enrollment: The system prompts the voter to place their finger on the fingerprint reader for enrollment. The fingerprint data is captured and stored securely.
- Mobile number registration: Voters provide their mobile numbers, which are linked to their fingerprint data for authentication purposes.

Voting Process:

- Voter authentication: Registered voters authenticate themselves by placing their fingerprints on the reader.
- Fingerprint sensor verification: The system compares the scanned fingerprint with the enrolled data to verify the identity of the voter.
- OTP generation: Upon successful fingerprint verification, the system generates a one-time password (OTP) for additional security.
- OTP sent to registered mobile: The OTP is sent to the voter's registered mobile number via SMS for verification.
- Verify OTP: The voter enters the received OTP through the keypad connected to the system. The system validates the OTP to ensure authenticity.
- Proceed to voting: If the OTP verification is successful, the voter gains access to the voting interface and can proceed to cast their vote.

Vote Casting:

- Voter casts their vote: The voter selects their desired candidate or option using the voting interface provided by the system.
- Vote recording: The system records the cast vote securely to prevent tampering or manipulation.
- Update vote count: The overall vote count is updated in real-time to reflect the latest voting statistics.

Security & Alerts:

- Failed authentication: If authentication fails at any stage, the system triggers an alert to notify authorities and prevent unauthorized access.
- Irregularities detected: The system continuously monitors voting activities for any irregularities or anomalies. If detected, alerts are sent to election authorities for immediate action.

End:

- The voting process concludes once all eligible voters have cast their votes, and the system enters a standby mode awaiting further instructions or shutdown.

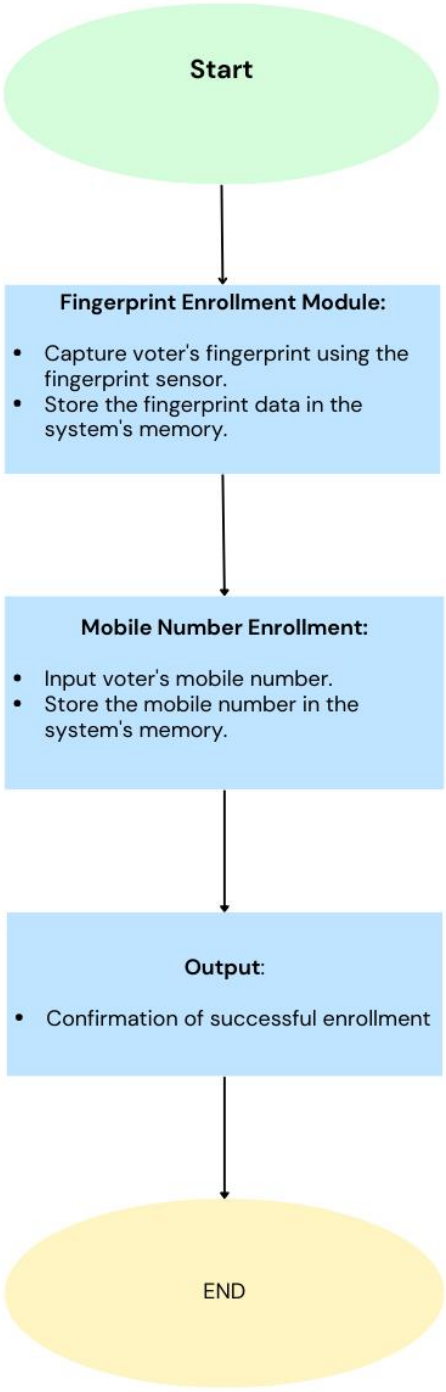


Fig 4.2: Voter enrollment workflow

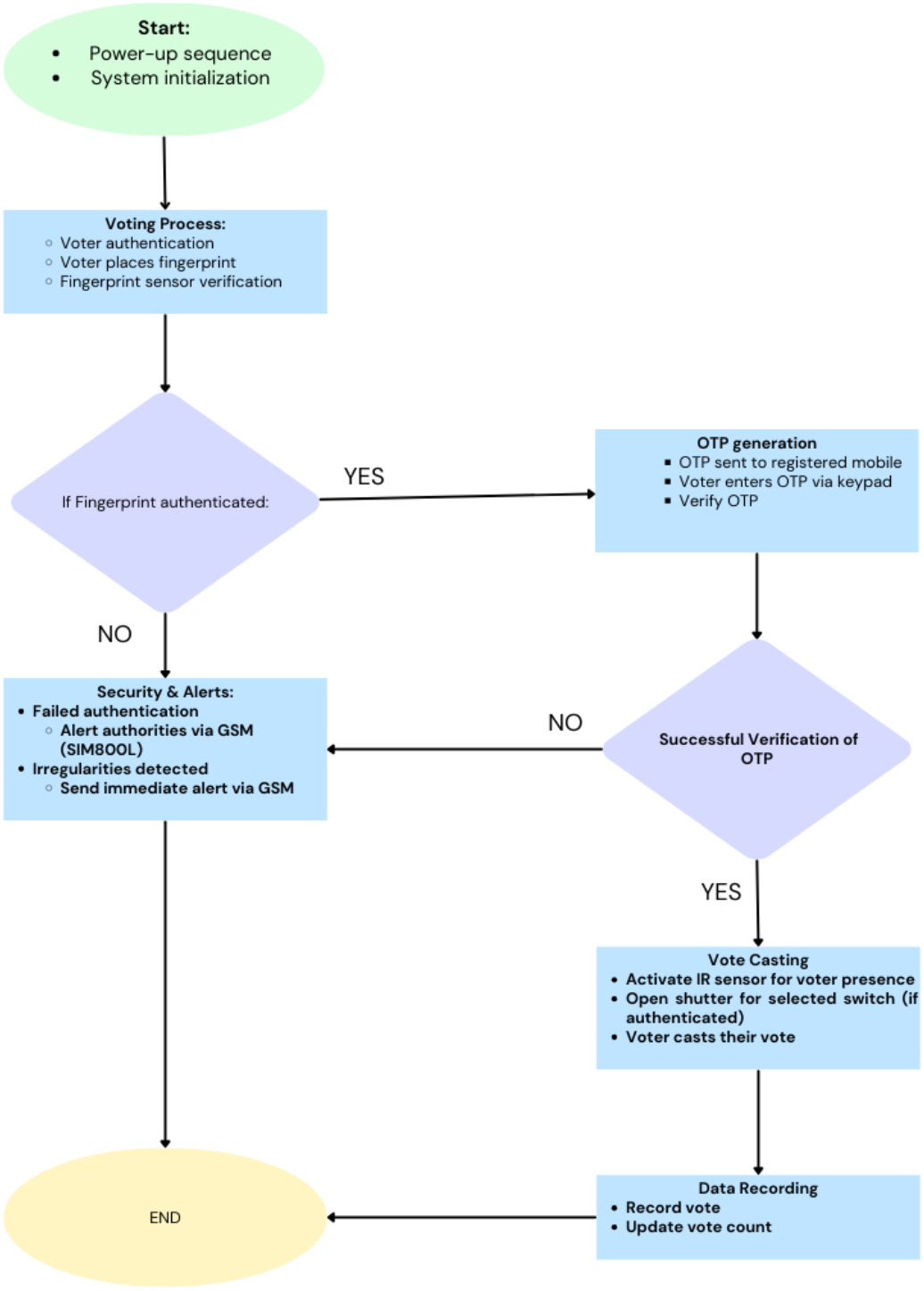


Fig 4.3: Verification and voting workflow

4.6 USE CASE DIAGRAM:

The use case diagram provides a visual representation of the interactions between actors and the system functionalities in our Secure Innovative Voting System with biometric and OTP authentication.

The diagram outlines the various actions that users and system components can perform within the voting system:

1. **Register Voter:** This use case allows eligible voters to register in the system by providing their personal information and biometric data. The registration process is facilitated by the election officer.

2. **Authenticate Voter:** In this use case, the system verifies the identity of voters during the authentication process. This involves comparing the biometric data provided by the voter with the registered data stored in the system. Additionally, OTP authentication is used to enhance security during the voter authentication process.

3. **Generate OTP:** The system generates a one-time password (OTP) for voter authentication purposes. This OTP is sent to the registered mobile device of the voter to validate their identity during the voting process.

4. **Poll Vote:** This use case enables registered voters to cast their votes securely after successful authentication. Once authenticated, voters can use the provided switches to poll their votes electronically.

5. **Alert Election Officer:** In case of failed authentication or any irregularities detected by the system, an alert is sent to the election officer. This allows for timely intervention and resolution of any issues that may arise during the voting process.

Relationships:

- The voters (actors) initiate the "Register Voter," "Authenticate Voter," "Generate OTP," and "Poll Vote" use cases.

- The election officer (actor) participates in all use cases as they facilitate the registration process and manage system operations.
- The system triggers the "Alert Election Officer" use case when irregularities are detected, ensuring swift action by the authorities.

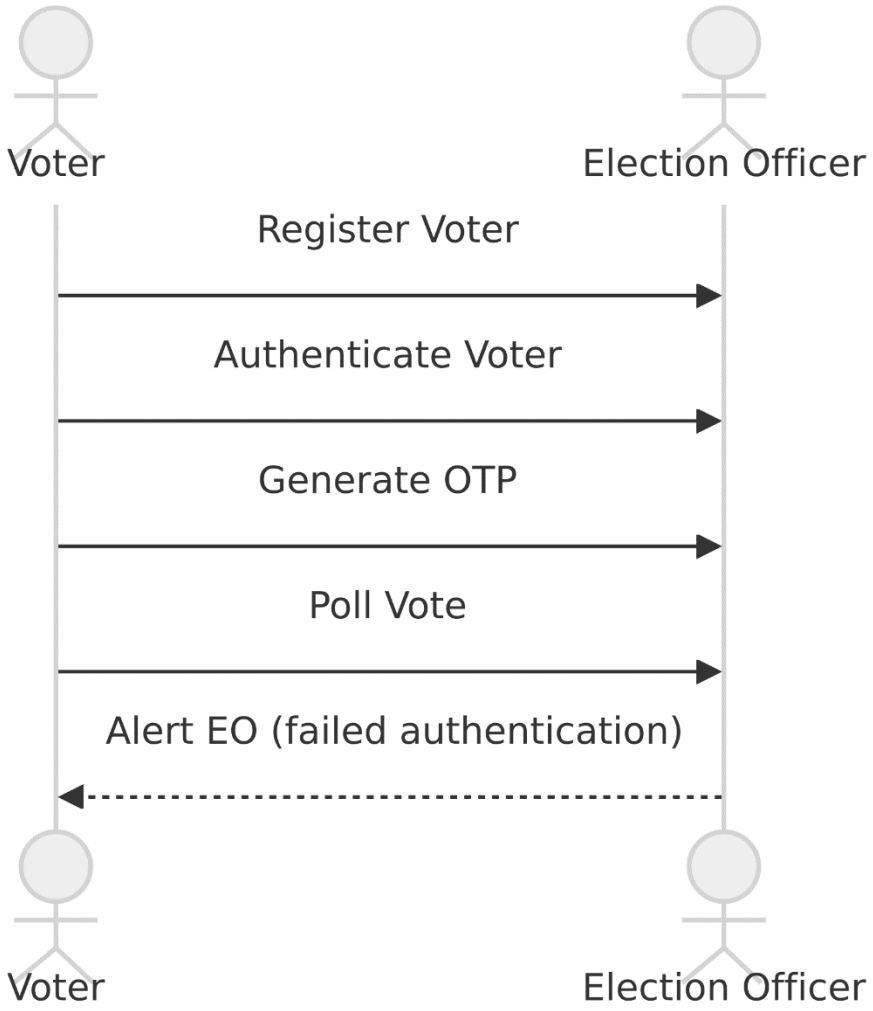


Fig 4.4: Use Case Diagram

4.7 CLASS DIAGRAM:

The class diagram outlines the structure and relationships between the key classes in our Secure Innovative Voting System with biometric and OTP authentication.

The diagram provides an overview of the classes and their attributes and methods within the system:

1. **Voter Class:** Represents individuals participating in the voting process. It holds attributes such as name, personal information, and registered mobile number. The class also includes methods for voter registration.

2. **BiometricData Class:** Manages biometric data, specifically fingerprint data, of registered voters. It contains attributes to store the fingerprint image and methods to store and compare fingerprint data.

3. **OTPAuthentication Class:** Handles the generation and verification of one-time passwords (OTPs) for voter authentication. It maintains attributes for OTPs and associated mobile numbers and includes methods to generate and verify OTPs.

4. **ElectionSystem Class:** Orchestrates system functionalities such as voter authentication and vote polling. While this class may not have specific attributes, it includes methods for voter authentication, vote polling, and alerting authorities in case of irregularities.

Relationships:

- The Voter class aggregates BiometricData and OTPAuthentication classes to associate each voter with their biometric and authentication data.
- The ElectionSystem class depends on Voter, BiometricData, and OTPAuthentication classes to perform authentication and voting processes effectively.

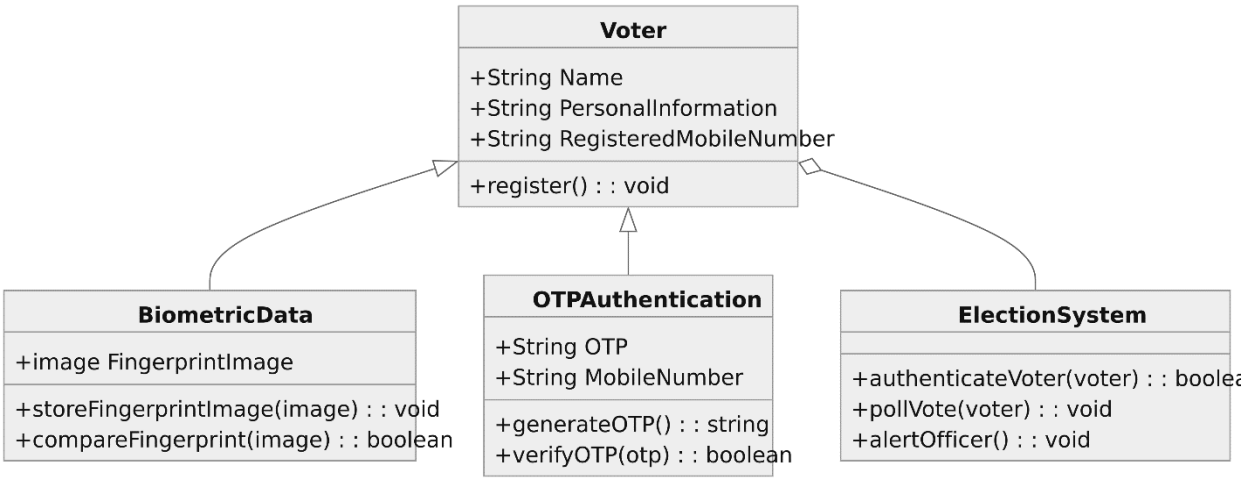


Fig 4.5: Class Diagram

4.8 SEQUENCE DIAGRAM:

The sequence diagram illustrates the sequence of interactions between system components during various processes in our Secure Innovative Voting System with biometric and OTP authentication.

The diagram provides detailed insights into the sequence of actions and communications between system components during different processes:

1. **Voter Registration:** Depicts the sequence of actions during voter registration, including providing personal information, capturing biometric data, and storing it securely in the system.

2. **Voter Authentication:** Shows the process of voter authentication using biometric data and OTP. It outlines the steps involved in verifying the voter's identity and generating and validating the OTP for authentication.

3. **Polling Vote:** Illustrates the steps involved in casting a vote after successful authentication. It includes interactions between the voter, the system components, and the voting interface.

4. **Alerting Authorities:** Describes the sequence of actions when the system detects failed authentication or irregularities during the voting process. It outlines how alerts are triggered and sent to the election authorities for timely intervention.

Interactions:

- Each sequence diagram outlines the interactions between voters, the system components (such as Arduino Mega, Fingerprint Sensor, GSM Module, etc.), and election authorities during specific processes, providing a comprehensive understanding of the system's functionality and communication flow.

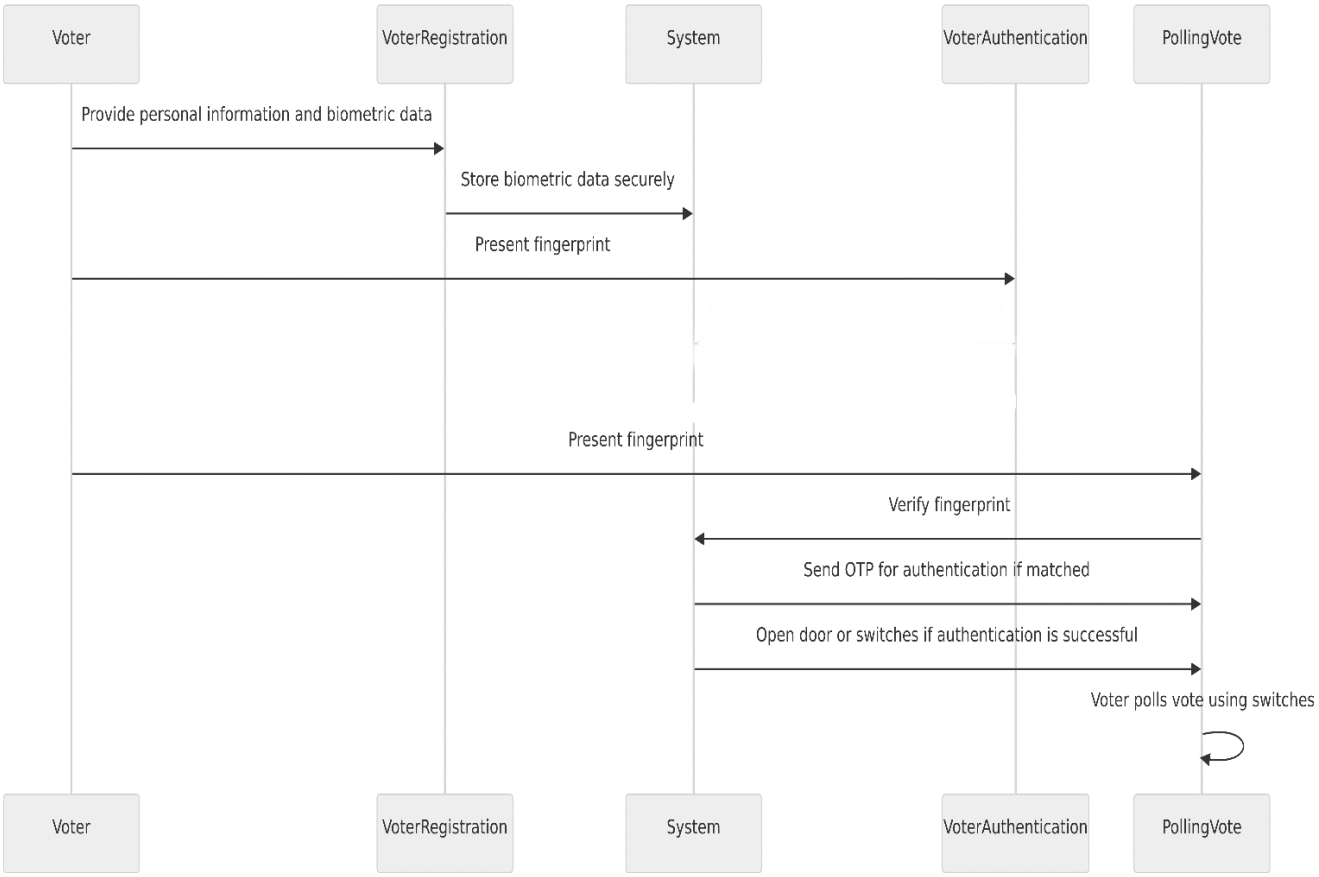


Fig 4.6: Sequence Diagram

CHAPTER 5

SYSTEM REQUIREMENTS

5.1 Power Supply

The input to the circuit is applied from the regulated power supply. The A.C input i.e., 230V from the mains supply is step down by the transformer to 12V and is fed to a rectifier. The output obtained from the rectifier is a pulsating D.C voltage. So, in order to get a pure D.C voltage, the output voltage from the rectifier is fed to a filter to remove any A.C components present even after rectification. Now, this voltage is given to a voltage regulator to obtain a pure constant D.C voltage.

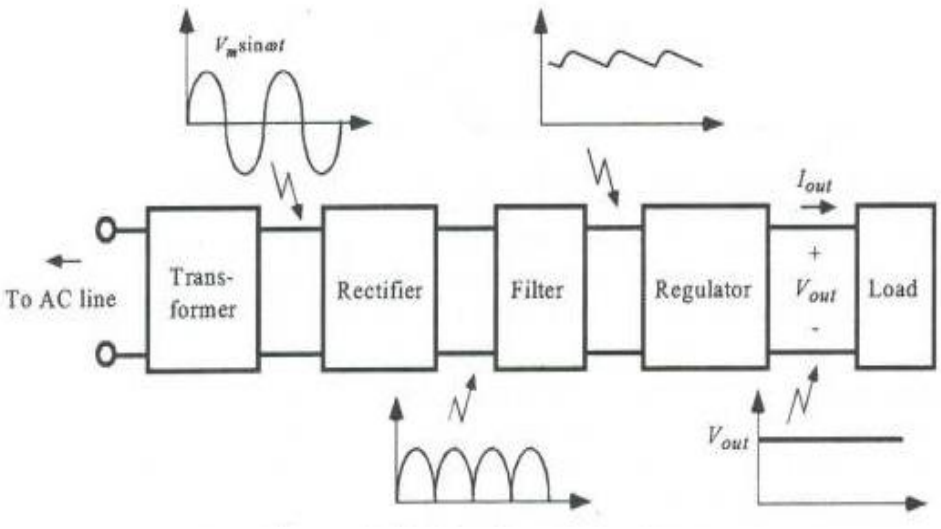


Fig 5.1 Power supply

Transformer:

Usually, DC voltages are required to operate various electronic equipment and these voltages are 5V, 9V or 12V. But these voltages cannot be obtained directly. Thus the a.c input available at the mains supply i.e., 230V is to be brought down to the required voltage level. This

is done by a transformer. Thus, a step down transformer is employed to decrease the voltage to a required level.

Rectifier:

The output from the transformer is fed to the rectifier. It converts A.C. into pulsating D.C. The rectifier may be a half wave or a full wave rectifier. In this project, a bridge rectifier is used because of its merits like good stability and full wave rectification.

Filter:

Capacitive filter is used in this project. It removes the ripples from the output of rectifier and smoothens the D.C. Output received from this filter is constant until the mains voltage and load is maintained constant. However, if either of the two is varied, D.C. voltage received at this point changes. Therefore a regulator is applied at the output stage.

Voltage regulator:

As the name itself implies, it regulates the input applied to it. A voltage regulator is an electrical regulator designed to automatically maintain a constant voltage level. In this project, power supply of 5V and 12V are required. In order to obtain these voltage levels, 7805 and 7812 voltage regulators are to be used. The first number 78 represents positive supply and the numbers 05, 12 represent the required output voltage levels.

Beyond their basic functionality, voltage regulators like the 7805 and 7812 offer additional features such as overcurrent protection, thermal shutdown, and short-circuit protection, safeguarding both the regulators themselves and the connected circuits from potential harm. This adds an extra layer of security and reliability to the overall system.

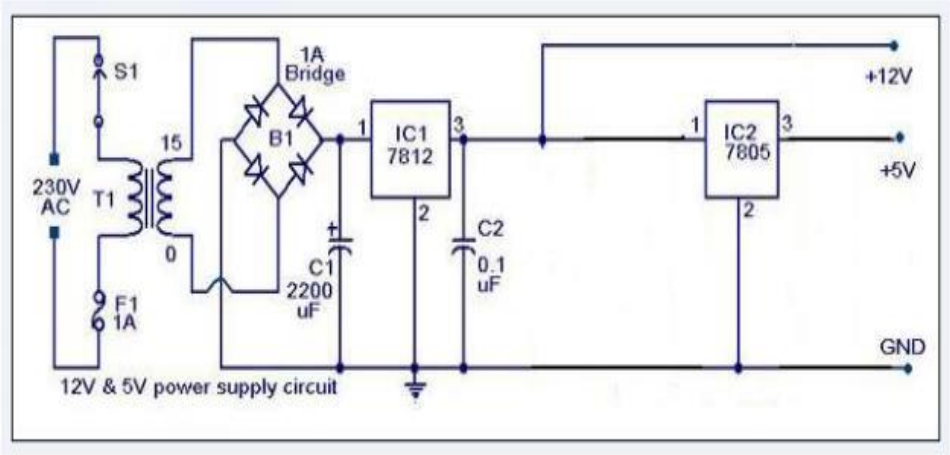


Fig 5.2 Power supply circuit diagram

5.2 ARDUINO UNO

Fig 5.3 ARDUINO UNO

Power (USB / Barrel Jack)

Every Arduino board needs a way to be connected to a power source. The Arduino UNO can be powered from a USB cable coming from your computer or a wall power supply (like this) that is terminated in a barrel jack. In the picture above the USB connection is labeled **(1)** and the barrel jack is labeled **(2)**.

The USB connection is also how you will load code onto your Arduino board. More on how to program with Arduino can be found in our Installing and Programming Arduino tutorial.

NOTE: Do NOT use a power supply greater than 20 Volts as you will overpower (and thereby destroy) yourArduino. The recommended voltage for most Arduino models is between 6 and 12 Volts.

Pins (5V, 3.3V, GND, Analog, Digital, PWM, AREF):

The pins on your Arduino are the places where you connect wires to construct a circuit (probably in conjuction with a breadboard and some wire. They usually have black plastic 'headers' that allow you to just plug a wire right into the board. The Arduino has several different kinds of pins, each of which is labeled on the board and used for different functions.

- **GND (3)**:Short for 'Ground'. There are several GND pins on the Arduino, any of which can be used to ground your circuit.
- **5V (4) & 3.3V (5)**: As you might guess, the 5V pin supplies 5 volts of power, and the 3.3V pin supplies 3.3 volts of power. Most of the simple components used with the Arduino run happily off of 5 or 3.3 volts.
- **Analog (6)**: The area of pins under the 'Analog In' label (A0 through A5 on the UNO) areAnalogIn pins. These pins can read the signal from an analog sensor (like a temperature sensor) and convert it into a digital value that we can read.
- **Digital (7)**: Across from the analog pins are the digital pins (0 through 13 on the UNO). These pins can be used for both digital input (like telling if a button is pushed) and digital output (like powering an LED).
- **PWM (8)**: You may have noticed the tilde (~) next to some of the digital pins (3, 5, 6, 9, 10, and 11 on the UNO). These pins act as normal digital pins, but can also be used for something called Pulse-Width Modulation (PWM). We have a tutorial on PWM, but for now, think of these pins as being able to simulate analog output (like fading an LED in and out).

- **AREF (9)**: Stands for Analog Reference. Most of the time you can leave this pin alone. It is sometimes used to set an external reference voltage (between 0 and 5 Volts) as the upper limit for the analog input pins.

Reset Button

Just like the original Nintendo, the Arduino has a reset button **(10)**. Pushing it will temporarily connect the reset pin to ground and restart any code that is loaded on the Arduino. This can be very useful if your code doesn't repeat, but you want to test it multiple times. Unlike the original Nintendo however, blowing on the Arduino doesn't usually fix any problems.

Power LED Indicator

Just beneath and to the right of the word "UNO" on your circuit board, there's a tiny LED next to the word 'ON' **(11)**. This LED should light up whenever you plug your Arduino into a power source. If this light doesn't turn on, there's a good chance something is wrong. Time to re-check your circuit!

TX RX LEDs

TX is short for transmit, RX is short for receive. These markings appear quite a bit in electronics to indicate the pins responsible for serial communication. In our case, there are two places on the Arduino UNO where TX and RX appear – once by digital pins 0 and 1, and a second time next to the TX and RX indicator LEDs . These LEDs will give us some nice visual indications whenever our Arduino is receiving or transmitting data (like when we're loading a new program onto the board).

Main IC

The black thing with all the metal legs is an IC, or Integrated Circuit . Think of it as the brains of our Arduino. The main IC on the Arduino is slightly different from board type to board type, but is usually from the ATmega line of IC's from the ATMEL company. This can be important, as you may need to know the IC type (along with your board type) before loading up a new program from the Arduino software. This information can usually be found in writing on the top side of the

IC. If you want to know more about the difference between various IC's, reading the datasheets is often a good idea.

Voltage Regulator

The voltage regulator is not actually something you can (or should) interact with on the Arduino. But it is potentially useful to know that it is there and what it's for. The voltage regulator does exactly what it says – it controls the amount of voltage that is let into the Arduino board. Think of it as a kind of gatekeeper; it will turn away an extra voltage that might harm the circuit. Of course, it has its limits, so don't hook up your Arduino to anything greater than 20 volts.

5.3 ESP8266 Module

ESP-01 WiFi module is developed by Ai-thinker Team. core processor ESP8266 in smaller sizes of the module encapsulates Tensilica L106 integrates inddustry-leading ultra low power 32-bit MCU micro, with the 16-bit short mode, Clock speed support 80 MHz, 160 MHz, supports the RTOS, integrated Wi-Fi MAC/BB/RF/PA/LLNA, on-board antenna.The module supports standard IEEE802.11 b/g/n agreement, complete TCP/IP protocol **s**tack. Users can use the add modules to an existing device networrking, or building a separate network controller.ESP8266 is high integration wireless SOCs, designed for space and power constrained mobile platform designers. It provides unsurpassed ability to embed Wi-Fi capabilities within other systems, or to function as a standalone application, with the lowest cost, and minnimal space requirement.

ESP8266EX offers a complete and self-contained Wi-Fi networking solution; it can be used to host the application or to offload Wi-Fi networking functions from another application processor When ESP8266EX hosts the application, it boots up directly from an external flash. In has integrated cache to improve the performance of the system in such applications. Alternately, serving as a Wi-Fi adapter, wireless internet access can be added to any micro controller based design with simple connectivity (SPI/SDIO or I2C/UART interface).

ESP8266EX also integrates an enhanced version of Tensilica's L106 Diamond series 32-bit processor, with on-chip SRAM, besides the Wi-Fi functionalities. ESP8266EX is often integrated with external sensors annd other application specific devices through its GPIOs; codes for such applications are provided in examples in the SDK.

Espressif Systems' Smart Connectivitty Platform (ESCP) demonstrates sophisticated system-level features include fast sleep/wake context switching for energy-efficient VoIP, adaptive radio biasing. for low-power operation, advance signal processing, and spur cancellation and radio co-existence features for common cellular, Bluetooth, DDR, LVDS, LCD interference mitigation.

5.4 LCD2004 Parallel LCD Display with IIC/I2C Interface

Fig 5.4 LCD Display

If you want to add some visual output to your Arduino projects, you'll need a display. If you need only a little to display, the LCD2004 Parallel LCD Display with **IIC/I2C interface** is a quite good solution.

This Display provides a simple and cost-effective solution for adding a 20×4 White on RGB Liquid Crystal Display into your project**. The display is 20 character by 4 line display has a very clear and high contrast white text upon a blue background/backlight.**

This is a great blue backlight LCD display. It is fantastic for Arduino-based projects. This Display with Blue Backlight is very easy to interface with Arduino or Other Microcontrollers.

This display overcomes the drawback of LCD2004 Parallel LCD Display in which you'll waste about 8 Pins on your Arduino for the display to get working. Luckily in this product, an I2C adapter is directly soldered right onto the pins of the display. So all you need to connect are the I2C pins, which show a good library and little of coding.

The I2C is a type of serial bus developed by Philips, which uses two bidirectional lines, called SDA (Serial Data Line) and SCL (Serial Clock Line). Both must be connected via pulled-up resistors. The usage voltages are standard as 5V and 3.3V.

Fig 5.5 I2C Adapter

If you already have the **I2C** adapter soldered onto the board, like in this product, the wiring is quite easy. You should usually have only four pins to hook up. **VCC** and **GND** of course. This display works with 5 volts. So we go for the 5V pin.

The values shown on the display can be either simple text or numerical values read by the sensors, such as temperature or pressure, or even the number of cycles that the Arduino is performing.

5.5 AS608 Optical Fingerprint Sensor Fingerprint Module

The below figure is AS608 Optical Fingerprint Sensor Fingerprint Module.

Fig 5.6 AS608 Optical Fingerprint Sensor Fingerprint Module

The fingerprint algorithm extracts features from the acquired fingerprint image and represents the fingerprint information. The storage, comparison, and search of fingerprints are all done by operating fingerprint features. Fingerprint processing includes two processes: fingerprint registration process and fingerprint matching process (in which fingerprint matching is divided into fingerprint comparison (1:1) and fingerprint search (1:N) two ways).When the fingerprint is registered, two fingerprints are entered for each fingerprint, and the input image is processed twice. The synthesis module is stored in the module. When the fingerprint is matched, the fingerprint sensor is used to input the fingerprint image to be verified and processed, and then it is compared with the fingerprint module in the module (if it is matched with a module specified in the module, it is called fingerprint comparison mode, i.e., 1:1 mode. If matching with multiple modules is called fingerprint search, ie 1:N mode, the module gives the matching result (pass or fail).

5.6 SIM800C GPRS/GSM SHIELD WITH ANTENNA

The Sim800C GPRS/GSM Shield with Antenna provides you with a way to use the GSM phone network to receive data from a remote location and it is compatible with all boards which have the same form factor (and pinout) as a standard Arduino Board. This shield can also be applied to DIY phones for calling, receiving and sending messages, making GPS trackers or other applications like smart home, etc. SIM800C GPRS/GSM Shield delivers GSM/GPRS850/900/1800/1900MHz signals for Audio, SMS and GPRS Service. Also, it runs at the low power consumption of about 0.6mA in sleep mode but acts compliant to GSM phase 2/2+: Class 4 (2 W @850/ 900 MHz), Class 1 (1 W @ 1800/1900MHz) with a two-in-one headset jack.

Fig 5.7 Sim800C GPRS/GSM Module.

The Sim800C GPRS/GSM Shield with Antenna offers versatile connectivity solutions, enabling seamless integration with the GSM phone network for remote data retrieval. Its compatibility extends to all boards sharing the standard Arduino form factor and pinout, ensuring widespread applicability across various projects. Beyond conventional Arduino applications, this shield proves invaluable in DIY phone setups, facilitating calling, message reception, and transmission functionalities. Moreover, its capabilities extend to more advanced projects such as GPS trackers and Smart home systems, showcasing its adaptability to diverse user needs.

5.7 TOWERPRO SG90 SERVO MOTOR (180° ROTATION)-GOOD QUALITY

Fig 5.8 Servo motor

TowerPro Servo Motors are optimum-quality and affordable cost servos.! They are suitable for a wide range of applications, including RC aircraft, automobiles, and robotics, Or just to have some fun with whatever crazy project you're working on.

When you purchase TowerPro motors in India, they are almost usually NOT ORIGINAL, and these are no exception… These are NOT ORIGINAL Tower Pro Servos, either. However, they are dead cheap and Serve the purpose. We put them to the test and found them to be of good quality for the price.

You can buy E Max Servos OR Orange servos if you're seeking high-quality servos.

Wire Description:

- *RED* – Positive
- *Brown* – Negative

- *Orange* – Signal

We have imported this copy of original TowerPro SG90 9g Mini Servo from our trusted supplier. This *good quality* servo motor is in very close competition to the original TowerPro SG90 1.2kgCm 180-degree servo motor.

Main Difference between Good Quality SG90 and Standard Quality Sg90 is

Property	**Good Quality (SKU:-43849)**	**Standard Quality (SKU:-5764)**
Cable	Thick	Thin
Gears	Nylon	PolyPropelene Plastic
Motor	High torque	Slightly Lower Torque

Table 5.1: 43849 Servo motor Vs 5764 Servo motor

5.8 SOFTWARES:

This tutorial will walk you through downloading, installing, and testing the Arduino software (also known as the Arduino IDE - short for Integrated Development Environment). Before you jump to the page for your operating system, make sure you've got all the right equipment.

Fig 5.9 : Arduino Logo

Required Materials:

- A computer (Windows, Mac, or Linux)
- An Arduino-compatible microcontroller (anything from this guide should work)
- A USB A-to-B cable, or another appropriate way to connect your Arduino-compatible microcontroller to your computer (check out this USB buying guide if you're not sure which cable to get).

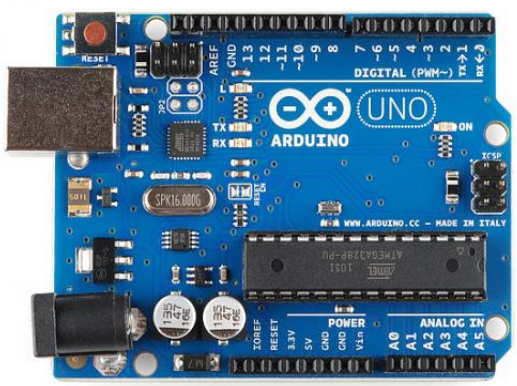


Fig 5.10 An Arduino Uno

Fig 5.11: An A-to-B USB Cable

How to install CH340 drivers (if you need them) on Windows, Mac OS X, and Linux.
If you're ready to get started, click on the link in the column on the left that matches up with your operating system, or you can jump to your operating system here.

- Windows
- Mac
- Linux
- Windows

This page will show you how to install and test the Arduino software with a Windows operating system (Windows 8, Windows 7, Vista, and XP).

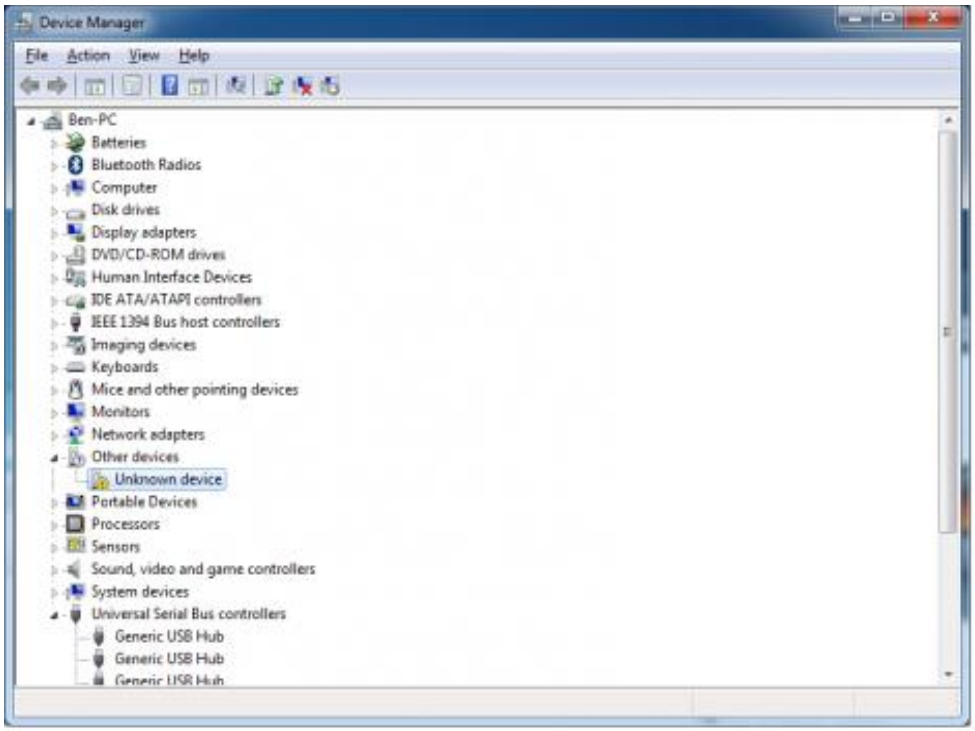
- Go to the Arduino download page and download the latest version of the Arduino software for Windows.
- When the download is finished, un-zip it and open up the Arduino folder to confirm that yes, there are indeed some files and sub-folders inside. The file structure is important so don't be moving any files around unless you really know what you're doing.
- Power up your Arduino by connecting your Arduino board to your computer with a USB cable (or FTDI connector if you're using an Arduino pro). You should see the an LED labed 'ON' light up. (this diagram shows the placement of the power LED on the UNO).

Drivers for Arduino Uno on Windows:

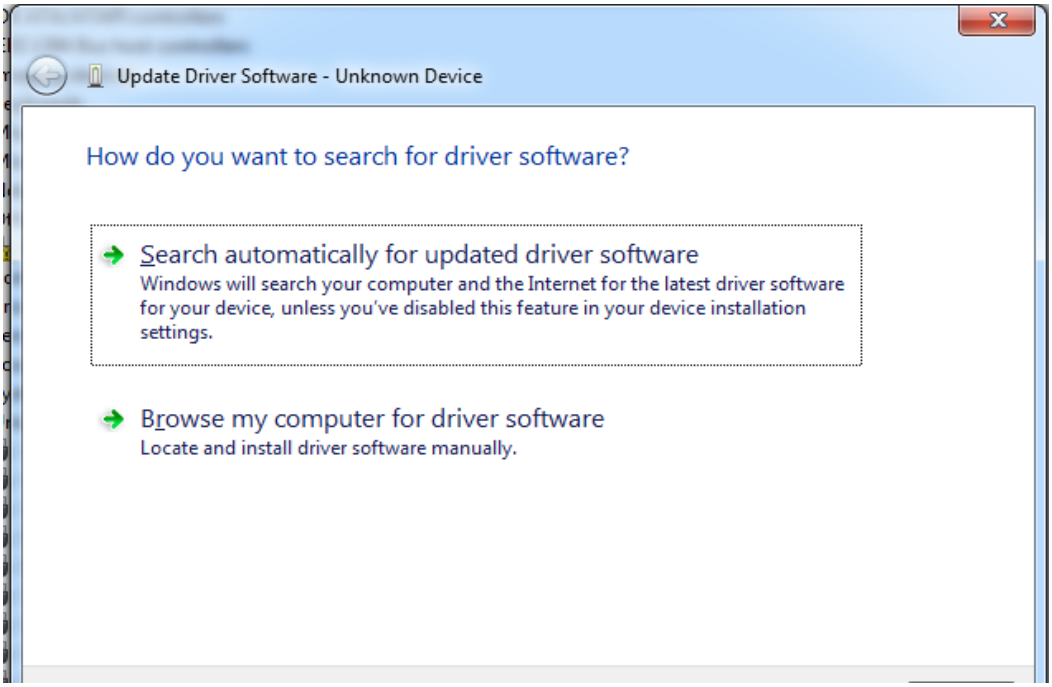
Installing the Drivers for the Arduino Uno (from Arduino.cc)

- Plug in your board and wait for Windows to begin it's driver installation process
- After a few moments, the process will fail, despite its best efforts
- Click on the Start Menu, and open up the Control Panel
- While in the Control Panel, navigate to System and Security. Next, click on System

- Once the System window is up, open the Device Manager
- Look under Ports (COM & LPT). You should see an open port named "Arduino UNO (COMxx)". If there is no COM & LPT section, look under 'Other Devices' for 'Unknown Device'.
- Right click on the "Arduino UNO (COMxx)" or "Unknown Device" port and choose the "Update Driver Software" option
- Next, choose the "Browse my computer for Driver software" option.



Screenshot 5.1: Device Manager



Screenshot 5.2: Update driver software

- Finally, navigate to and select the Uno's driver file, named "ArduinoUNO.inf", located in the "Drivers" folder of the Arduino Software download (not the "FTDI USB Drivers" sub-directory). If you cannot see the .inf file, it is probably just hidden. You can select the 'drivers' folder with the 'search sub-folders' option selected instead.
- Windows will finish up the driver installation from there

For earlier versions of the Arduino boards (e.g. ArduinoDuemilanove, Nano, or Diecimila) check out this page for specific directions.

Drivers for RedBoard on Windows

If you are using a RedBoard programmed for Arduino, please go to How to Install FTDI Drivers, for specific instructions on how to install the drivers.

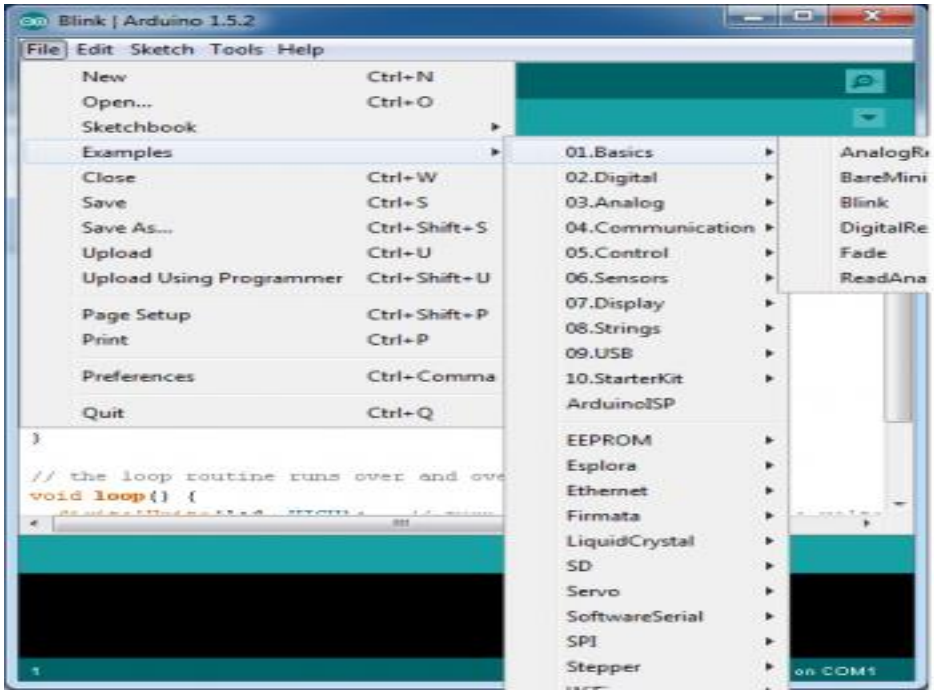
USB Serial Driver Quick Install

How to install USB serial drivers on Windows.

Launch and Blink!

After following the appropriate steps for your software install, we are now ready to test your first program with your Arduino board!

- Launch the Arduino application
- If you disconnected your board, plug it back in
- Open the Blink example sketch by going to: File > Examples > 1.Basics > Blink
- Select the type of Arduino board you're using: Tools > Board > your board type



Screenshot 5.3: Examples Code Template

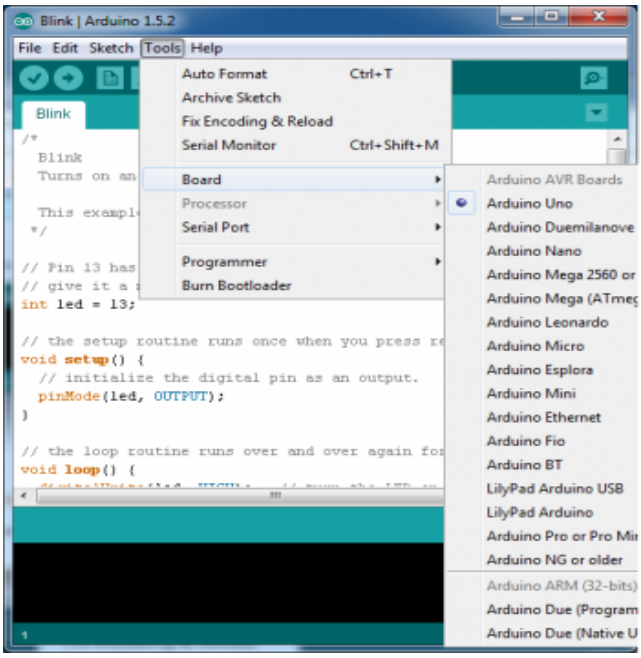
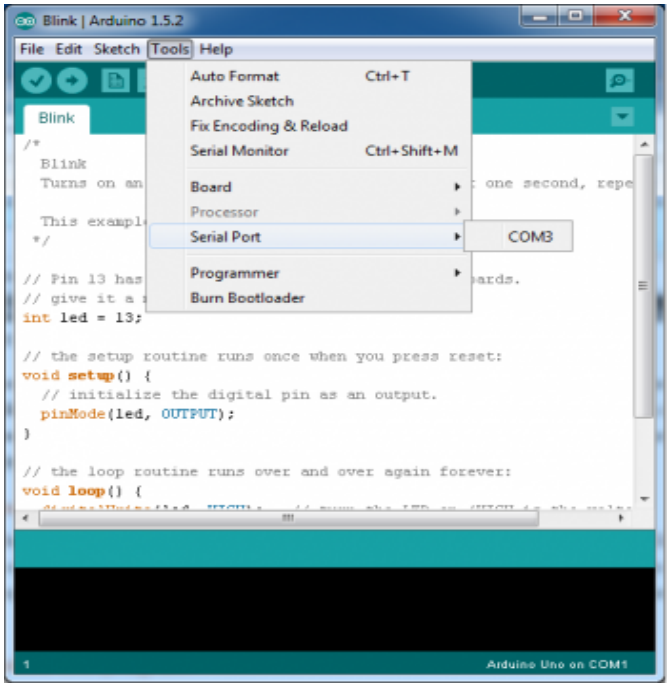


Fig 5.4 : Selection of board

- Select the serial/COM port that your Arduino is attached to: Tools > Port >COMxx



Screenshot 5.5: Selection of serial port

- If you're not sure which serial device is your Arduino, take a look at the available ports, then unplug your Arduino and look again. The one that disappeared is your Arduino.
- With your Arduino board connected, and the Blink sketch open, press the 'Upload' button.

Now, armed with the knowledge of your Arduino board and serial port, you're ready to upload your first program. But before you do, take a moment to reflect on the journey thus far. From installing the software to navigating the Arduino IDE, each step has brought you closer to realizing your creative vision. With anticipation building, press the 'Upload' button and witness the magic unfold as your code springs to life, illuminating the onboard LED in a mesmerizing display of programming prowess.

Screenshot 5.6: Example code

- After a second, you should see some LEDs flashing on your Arduino, followed by the message 'Done Uploading' in the status bar of the Blink sketch.
- If everything worked, the onboard LED on your Arduino should now be blinking! You just programmed your first Arduino!

Troubleshooting:

The Arduino Playground Linux section is a great resource for figuring out any problems with your Arduino installation.

- Board Add-Ons with Arduino Board Manager

With Arduino v1.6.4+, a new boards manager feature makes it easy to add third-party boards (like the SparkFunRedboard, Digital Sandbox, and RedBot) to the Arduino IDE.

To start, highlight and copy (CTRL + C / CMD + C) the text below for the boards manager URL. You'll need this to configure Arduino.

COPY CODEhttps://raw.githubusercontent.com/sparkfun/Arduino_Boards/master/IDE_Board_Manager/package_sparkfun_index.json

Open up Arduino:

- Configure the Boards Manager
    - For Windows and Linux, head to File>Preferences>Additional Boards Manager URLs and paste (CTRL + V / CMD + V) the link
    - For Macs, head to Arduino>Preferences>Additional Boards Manager URLs and paste (CTRL + V / CMD + V) the link
- Click on Tools>Board>Boards Manager...
- Select the Type as "Contributed" from the drop down menu.
- Click on the SparkFun AVR Boards and then click Install

That's it! Boards are all installed. This also gives you access to all of our library files as well through the built-in Library Manager tool in Arduino. Looking for more information about adding other custom boards? Check out the the following tutorial to install other Arduino cores.

CHAPTER 6

SOFTWARE ENVIRONMENT

6.1 ARDUINO IDE

The Arduino Integrated Development Environment (IDE) is the primary software tool used for programming and uploading code to the Arduino UNO microcontroller. It provides a user-friendly interface for writing, compiling, and uploading sketches (programs) to the Arduino board. The IDE supports the Arduino programming language, which is based on C/C++, making it accessible to both beginners and experienced developers. Additionally, the IDE offers a wide range of built-in libraries and examples, simplifying the development process and enabling rapid prototyping of projects.

6.2 THINGSPEAK CLOUD

ThingSpeak is an Internet of Things (IoT) platform that enables the collection, analysis, and visualization of data from IoT devices in real-time. It offers a cloud-based infrastructure for storing and managing sensor data, as well as built-in analytics tools for processing and visualizing data streams. ThingSpeak provides RESTful APIs for integration with various IoT devices and applications, making it an ideal platform for IoT projects, including our Smart Fingerprint-Based Voting System. By leveraging ThingSpeak, we can securely transmit and store voting data, enabling real-time monitoring and analysis of voting patterns and trends.

6.3 ARDUINO PROGRAMMING

- Arduino programming is primarily done using a variant of the C++ programming language. C++ is a powerful and versatile programming language known for its efficiency, flexibility, and performance. It offers a wide range of features and capabilities, making it well-suited for developing embedded systems and IoT applications on Arduino platforms.

- When writing code for Arduino boards, developers utilize the Arduino IDE, which provides a simplified programming environment with built-in libraries and examples tailored for Arduino development. Despite its simplicity, the Arduino programming

language retains the core syntax and structure of C++, allowing developers to leverage the full power of the language.

- C++ features commonly used in Arduino programming include:
    - **Variables and Data Types**: C++ supports various data types, such as integers, floating-point numbers, characters, and arrays, which are essential for storing and manipulating data in Arduino sketches.
    - **Functions**: Functions in C++ allow developers to encapsulate code into reusable blocks, improving code organization and modularity. Arduino sketches typically include setup() and loop() functions, which are automatically executed at the start and continuously during program execution, respectively.
    - **Control Structures**: C++ provides control structures such as if-else statements, for loops, and while loops, enabling developers to implement conditional logic and repetitive tasks in Arduino sketches.
    - **Object-Oriented Programming (OOP)**: While Arduino programming often focuses on procedural programming, C++ also supports object-oriented programming principles such as classes, objects, inheritance, and polymorphism. These concepts can be utilized to create modular and extensible code structures in larger Arduino projects.

6.4 NECESSARY LIBRARIES

- In addition to the Arduino IDE and ThingSpeak Cloud, our project requires the use of specific libraries for interfacing with hardware components and peripherals. These libraries enhance the functionality and compatibility of the system, facilitating seamless integration of hardware and software components. Notable libraries used in our project include:
    - **LiquidCrystal.h**: This library provides functions for controlling liquid crystal displays (LCDs) compatible with the Hitachi HD44780 driver. It enables the display of text and custom characters on the LCD screen, allowing for informative user interfaces and visual feedback in our Smart Fingerprint-Based Voting System.

- **Adafruit_Fingerprint.h**: The Adafruit Fingerprint Sensor Library facilitates communication with fingerprint sensors, such as the AS608 Fingerprint Reader module. This library offers functions for enrolling fingerprints, verifying identities, and capturing fingerprint images, enabling robust biometric authentication in our voting system.

CHAPTER 7

SYSTEM IMPLEMENTATION

7.1 PROJECT IMPLEMENTATION OVERVIEW

Our project implementation involved the integration of hardware components and software algorithms to realize the Secure Innovative Voting System with Biometric and OTP Authentication. The implementation process encompassed several key stages:

1. **Hardware Setup**: We assembled the necessary hardware components, including the Arduino Uno microcontroller, R307 Optical Fingerprint Reader Sensor Module, SIM800L GSM Module, 4×4 Matrix Keypad Membrane Switch, LCD, Push Button Switches, and Servo Motor. Each component was connected according to the system architecture to ensure seamless functionality.

2. **Software Development**: We developed the software logic and algorithms required for system operation. This included programming the Arduino Uno using the Arduino IDE, incorporating libraries such as LiquidCrystal.h and Adafruit_Fingerprint.h for interfacing with the hardware components, and implementing the necessary control logic for fingerprint authentication, OTP generation, and GSM communication.

3. **System Integration**: Once the hardware and software components were developed, we integrated them into a cohesive system. This involved testing the interaction between different modules, ensuring proper communication between hardware components and the Arduino Uno, and validating the functionality of each feature individually and in combination.

4. **Testing and Validation**: We conducted rigorous testing to validate the functionality and reliability of the system. This included testing for fingerprint recognition accuracy, OTP generation and verification, GSM communication reliability, and overall system robustness under various operating conditions.

5. **Refinement and Optimization**: Throughout the implementation process, we iteratively refined and optimized the system to improve performance, efficiency, and user experience.

This involved identifying and addressing any issues or limitations encountered during testing, optimizing code for better resource utilization, and fine-tuning system parameters for optimal operation.

7.2 CODE OVERVIEW

The code for the project is structured into several modules, each responsible for a specific task. It begins with initializing the system and its components, including the fingerprint sensor, GSM module, and LCD display. The voter registration process involves capturing and storing fingerprint data along with the voter's mobile number. During the voting process, the system verifies the voter's identity using the fingerprint sensor and generates a one-time password (OTP) sent to the registered mobile number. The voter then enters the OTP via a keypad, and upon successful verification, the system grants access to cast the vote. Finally, the system records the vote and updates the overall count, ensuring the integrity and security of the electoral process.

1. **Library Inclusions:**
    - The code includes necessary libraries such as **LiquidCrystal** for interfacing with LCD, **Adafruit_Fingerprint** for fingerprint sensor, and **SoftwareSerial** for serial communication.
2. **Global Variable Declaration:**
    - Global variables like **party1**, **party2**, **party3** for vote count, **OTP** for storing generated OTP, **number** for storing a phone number, etc., are declared.
3. **Setup Function:**
    - Initializes various pins and components.
    - Initializes the LCD, serial communication, fingerprint sensor, and sets up the GSM module for sending SMS.
4. **Keypad Function:**
    - Allows the user to enter a password through a keypad.

- Retrieves the entered password and returns it as an integer.

5. **Loop Function:**
    - Checks for button presses to enroll fingerprints or identify registered voters.
    - If a registered voter is identified, it generates and sends an OTP to the registered mobile number.
    - The voter is then prompted to enter the OTP. If the OTP is valid, the voter can proceed to vote by pressing buttons corresponding to different parties.
    - After voting, a confirmation SMS is sent to the registered mobile number.
6. **getFingerprintEnroll Function:**
    - Guides the user through the process of enrolling fingerprints by capturing images and converting them to templates.

7.3 CODE SNIPPETS

1. **Initializing Components:**

```
#include <LiquidCrystal.h>
#include <Adafruit_Fingerprint.h>
#include <SoftwareSerial.h>

const int rs = 13, en = 12, d4 = 11, d5 = 10, d6 = 9, d7 = 8;
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);
Adafruit_Fingerprint finger = Adafruit_Fingerprint(&mySerial);
SoftwareSerial mySerial(2, 3);
```

2. **Setup Function:**
    - Initializes pins, LCD, serial communication, and fingerprint sensor.
    - Registers a mobile number for SMS communication.

```
void setup() {
```

```
    // Pin modes setup
    // LCD initialization
    // Serial communication initialization
    // Fingerprint sensor initialization
    // Register mobile number for SMS communication
}
```

3. **Keypad Function:**
    - Handles keypad input for entering the password (OTP).
    - Generates and returns the entered password as an integer.

```
int keypad() {
    // Keypad input handling for password entry
    // Generates and returns the entered password
}
```

4. **Main Loop Function:**
    - Waits for user input to enroll fingerprints or identify voters.
    - Handles OTP verification and records votes for different parties.

```
void loop() {
    // Waits for user input (enroll/identify)
    // Handles OTP verification and vote recording
}
```

5. **Fingerprint Enrollment Function:**

    - Guides the user through the process of enrolling fingerprints.
    - Handles various errors during fingerprint enrollment.

```
uint8_t getFingerprintEnroll() {
  // Guides user through fingerprint enrollment
  // Handles errors during enrollment process
}
```

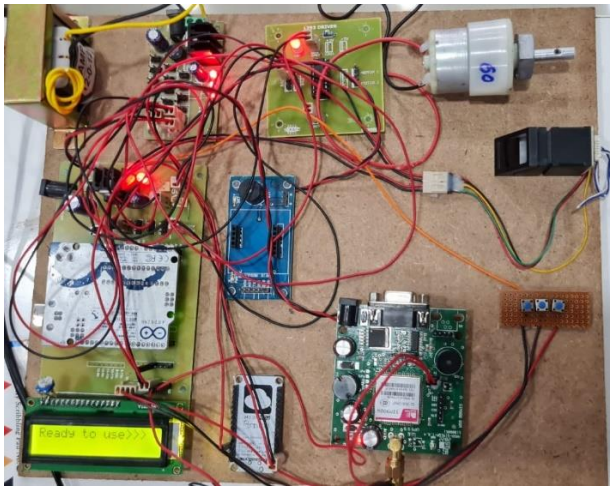
CHAPTER 8

SYSTEM RESULTS

8.1 SCREENSHOTS

8.1.1 System Overview

This image provides a comprehensive overview of the Secure Innovative Voting System with Biometric and OTP Authentication, showcasing its main components and their interactions**.**



Screenshot 8.1: System Overview

8.1.2 Welcome Screen on LCD:

The welcome screen displayed on the LCD provides a user-friendly interface for voters, guiding them through the voting process. It serves as the initial point of interaction, welcoming voters and prompting them to proceed with voter registration or authentication.

Screenshot 8.2: Ready to Use Welcome Screen

8.1.3 Voter Enrolling:

This message confirms successful enrollment of a voter in the system. It indicates that the voter's fingerprint and mobile number have been successfully registered, ensuring their eligibility to participate in the voting process.

Screenshot 8.3: Voter Enrollment

8.1.4 OTP Generation:

This image depicts the system generating a unique OTP (One-Time Password) for voter authentication. The OTP serves as an additional security measure, ensuring that only authorized voters can proceed to cast their votes.

Screenshot 8.4: OTP Generation

8.1.5 Sending OTP to Voter Mobile:

This screenshot captures the moment when the OTP is sent to the voter's registered mobile number via SMS. The SMS notification provides the voter with the necessary OTP to authenticate their identity and proceed with voting.

Screenshot 8.5 Sending of OTP to mobile

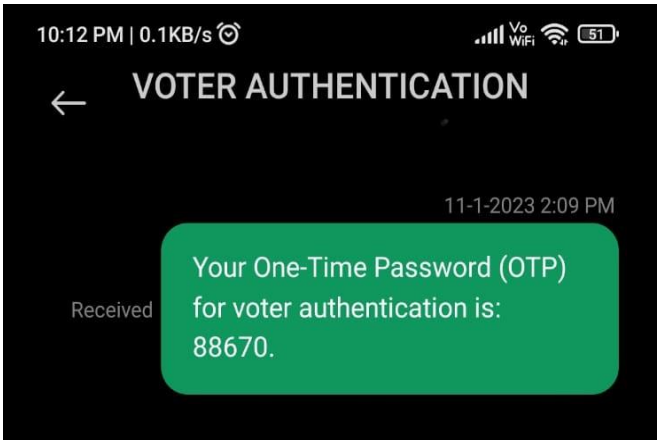
8.1.6 Entering OTP:

Here, the voter is seen entering the OTP received on their mobile device into the voting system via the keypad. This step ensures that the voter possesses both their fingerprint and the corresponding OTP, enhancing the security of the authentication process.

Screenshot 8.6: Entering of OTP

8.1.7 Received OTP SMS on Mobile Phone:

This screenshot displays the OTP received by the voter on their mobile phone via SMS. It serves as visual confirmation for the voter, allowing them to enter the OTP into the voting system for authentication and access to the voting interface.



Screenshot 6.6: OTP SMS sent to mobile

CONCLUISON

In conclusion, the Smart Voting System with Fingerprint and OTP Authentication represents a significant advancement in electoral technology, offering a modernized and secure approach to conducting elections. By leveraging biometric authentication and OTP verification mechanisms, combined with innovative hardware components and the Arduino microcontroller platform, this system addresses the shortcomings of traditional voting systems while upholding the principles of democracy and civic engagement.The integration of biometric authentication through fingerprint recognition provides a robust and reliable means of verifying voters' identities, significantly reducing the risk of fraud, impersonation, and multiple voting. Additionally, the inclusion of OTP verification adds an extra layer of security, ensuring that only authorized individuals can participate in the voting process.Furthermore, the Smart Voting System's emphasis on transparency, integrity, and user-friendliness aligns with the fundamental principles of democracy, fostering trust and confidence in the electoral process. Through its intuitive interface, clear instructions, and audible feedback, the system empowers voters to actively participate in the democratic process, promoting inclusivity and civic engagement.

FUTURE SCOPE

1. **Enhanced Security Measures**:
    - Implement multi-factor authentication and advanced security protocols to bolster system security against unauthorized access.
2. **Integration with Biometric Databases**:
    - Explore integration with national biometric databases like Aadhaar to streamline voter registration and authentication processes.
3. **Cloud-based Solutions**:
    - Develop cloud-based solutions for data storage and analysis, enabling real-time monitoring of voting activities.
4. **Blockchain Technology Integration**:
    - Investigate blockchain technology for secure and transparent recording of votes, ensuring immutable records and enhancing trust.
5. **Mobile Application Development**:
    - Develop mobile applications for remote voter registration and voting, increasing accessibility and convenience.
6. **Scalability and Flexibility**:
    - Design the system with scalability in mind to accommodate future expansions and modifications.
7. **Machine Learning for Anomaly Detection**:
    - Utilize machine learning for real-time detection of suspicious voting patterns or irregularities.
8. **Collaboration with Election Authorities**:

- Collaborate with election authorities to integrate the voting system into existing electoral infrastructure.

9. **Public Awareness Campaigns**:

    - Conduct public awareness campaigns to educate voters about the benefits and reliability of the new voting system.

REFERENCES

[1] Chandra Keerthi Pothina, Atla Indu Reddy "Smart Voting System using Facial Detection" IEEE Journal, April 2020.

[2] Anurag Chowdhury, Simon Kirchgasser, Andreas Uhl, Arun Ross "CNN Automatically Learn the Significance Of Minutiae Points for Fingerprint Matching?" IEEE Conference, Mar 2020.

[3] Samarth Agarwal, Afreen Haider, "Biometrics Based Secured Remote Electronic Voting System". IEEE Conference, Sep 2020.

[4] Suresh Kumar, Tamil Selvan G M, "Block chain Based Secure Voting System Using Lot", IEEE Journal, JAN 2020.

[5] Hanzhuo Tan, Ajay Kumar, "Towards More Accurate Contactless Fingerprint Minutiae Extraction and Pose-Invariant Matching" IEEE Conference 2020.

[6] Chengsheng, Yuan, Zhihua, Xia, "Fingerprint Liveness Detection using an improved CNN with image Scale Equalization" IEEE Journal 2019.

[7] Hui Xui, Miao Qi, "Multimodal Biometrics Based on Convolutional Neural Networks by Two-Layer Fusion" IEEE Conferences 2019.

[8] Abdelleratif EI Idrissi, Youssef El Merabet, "Plamprint Recognition using state-of the art Local texture descriptors." IEEE Conferences 2020.

[9] Uttam U. deshpande, V.S. Malemath, "A Convolution Neural Network-Based Latent Fingerprint Matching Using the Combination of Nearest Neighbor Arrangement Indexing" IEEE Conference, JAN 2020.

[10] Chengsheng Yuan, Zhihua Xia, "Fingerprint Liveness Detection using an improved CNN With Image Scale Equalization" IEEE Conference, JAN 2019.

[11] Ayushi Tamrakar, Neetesh Gupta, "Low Resolution Fingerprint Image Verification using CNN Filter and LSTM Classifier" IEEE Confernce, Jan 2020.

[12] Ishank Geol, N.B.Puhan, “Deep Convolution Neural Network for Double-Identity Fingerprint Detection”, IEEE Conference 2020.

[13] Maliha Khan, Rani Astya, “Face Detection And Recognition Using Opencv” IEEE Conference 2020.

[14] XuechaoYang, Xun Yi, Surya Nepal, Andrei Kelarev, and Fengling Han, “A Secure Verifiable Ranked Choice Online Voting System Based on Homomorphic Encryption” 10.1109/ACCESS. 2018. 2817518, IEEE Access.

[15] Alaguvel.R, Gnanavel.G, Jagadhambal.K “Biometrics using Electronic Voting System with Embedded Security”.

[16] Umang Shah, Trupt Shah, Marteen Kansagara, Saagar Daxini, „Biometric Secured Voting Machine to Avoid Bogus Voting Based on AADHAR CARD“, International Journal of Innovative Research in Computer and Communication Engineering , Vol. 3, Issue 3, March 2015

[17] Qasim Abbas, SarahJavaid, Tanzeel Hussnain Abass "Location-free Voting System with the help of IOT Technology” 2018 12th International Conference on Mathematics, Actuarial Science, Computer Science and Statistics Issue:2018,Pages:14-20.