

VISVESVARAYA TECHNOLOGICAL UNIVERSITY

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



A
DISSERTATION REPORT

On

“SPEEDY LINE : Victim Information Retrieval Using ML and IoT”

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

BACHELOR OF ENGINEERING

IN

Submitted By

Ankit Saurabh

USN: 1SJ19CS008

Anushka Ramesh

USN: 1SJ19CS013

Manasa S A

USN: 1SJ19CS086

Ruksar C

USN: 1SJ18CS081

COMPUTER SCIENCE AND ENGINEERING

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

Under the Guidance of

**Prof. Ajay N
Assistant Professor
Dept. Of CSE, SJCIT**



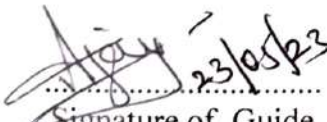
**S J C INSTITUTE OF TECHNOLOGY
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
CHIKKABALLAPUR-562101
2022-2023**

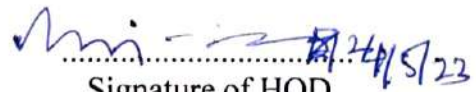
S.J.C INSTITUTE OF TECHNOLOGY, Chickballapur - 562101
Department of Computer Science and Engineering



CERTIFICATE

This is to certify that the project work entitled “SPEEDY LINE: Victim Information Retrieval Using ML and IoT” is a bonafide work carried out by **ANKIT SAURABH (1SJ19CS008), ANUSHKA RAMESH (1SJ19CS013), MANASA S A (1SJ19CS086), RUKSAR C (1SJ18CS081)** in partial fulfillment for the award of **Bachelor of Engineering in Computer Science and Engineering of Visvesvaraya Technological University, Belagavi** during the year **2022-2023**. 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 project work prescribed for the Bachelor of Engineering degree.




Signature of Guide
Prof. Ajay N
Assistant Professor,
Dept. of CSE, SJCIT


Signature of HOD
Dr. Manjunath Kumar B H
Professor & HOD,
Professor & HOD,
Department of Computer Science & Engg.,
S.J.C. Institute of Technology,
Chickballapur-562 101

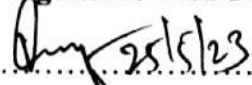
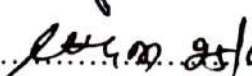

Signature of Principal
Dr. G T Raju
Principal,
S.J.C. Chickballapur Technology
Chickballapur - 562 101.

External Examiners:

Name of the Examiners

1.  Vinay Kumar V B
2.  S RATH G M

Signature with Date

 25/5/23
 25/5/23

DECLARATION

We, ANKIT SAURABH (1SJ19CS008), ANUSHKA RAMESH (1SJ19CS013), MANASA S A (1SJ19CS086), RUKSAR C (1SJ18CS081) 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 “SPEEDY LINE: Victim Information Retrieval System using ML & IoT” has been carried out at B.G.S R&D Centre, Dept. of CSE, SJCIT under the guidance of guide Prof. Ajay N, Assistant 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 2022-2023. We further declare that the report had not been submitted to another university for the award of any other degree.

PLACE: SJCIT, CHICKBALLAPUR-562101

Date: 11/05/2023

Ankit Saurabh

ANKIT SAURABH
USN: 1SJ19CS008

Anushka

ANUSHKA RAMESH
USN: 1SJ19CS013

S.A. Manasa

MANASA S A
USN: 1SJ19CS086

Ruksar C

RUKSAR C
USN: 1SJ18CS081

ABSTRACT

Complaints related to accidental and incident cases are not updated effectively in online databases especially by Private and Government hospitals, due to which the information details of such victims are missing. The problems faced in identifying such victims from unexpected accidents and incidents are difficult for tracing their details from existing methodology adopted. To overcome, the victim's information can be tracked with the existing available databases by building an integrated Automated Details Retrieval System (ADRS) using victims Finger-print from which Aadhaar details and mobile numbers are extracted. Simultaneously, ADRS also searches for retrieving some additional details from repositories of Driving license, Voter-id card, Pan Card, Social Networking Sites (SNS) and Mobile Service providers. Input to ADRS is Thumb/Finger impression of victims who are partially injured or missing children's or aged persons or mentally disabled persons. Internally, ADRS detects and chooses unique Aadhaar number from the victims fingerprint. Later, ADRS starts mapping, crawling, retrieving and then gathering information through others databases (Pan, voter-id, Passport and Driving license, SNS) which makes the ADRS more precise and efficacious. This ADRS retrieves the information details by crawling through the online databases and generates the report. The generated report constitutes of Name, address, mobile, DOB and mobile number of victim along with details of two (2) nearest family members are fetched using Reinforcement machine learning technique. With this facility, unknown details can be traced and intimated to their family members regarding the condition of deceased persons or missing adults or children's.

ACKNOWLEDGEMENT

With reverential pranam, we express my sincere gratitude and salutations to the feet of his holiness **Paramapoojya Jagadguru Byravaikya Padmabhushana Sri Sri Sri Dr. Balagangadharanatha Maha Swamiji**, his holiness **Paramapoojya Jagadguru Sri Sri Sri Dr. Nirmalanandanatha Maha Swamiji**, and **Sri Sri Mangalnath 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 Chandrashekaranaatha Swamiji Institute of Technology**, for providing us an opportunity for completing the Project Work Phase-II 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 Phase-II.

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

We convey our sincere thanks to Project Guide **Prof. Ajay N, Assistant 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 Phase-II.

We also feel immense pleasure to express deep and profound gratitude to Project Co-ordinators **Prof. Shrihari M R, Prof. Srinath G M and Prof. Kiran Kumar, Assistant Professors, Department of Computer Science and Engineering, S J C Institute of Technology**, for their 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, Chickaballapur for their support.

We also thank all those who extended their support and co-operation while bringing out this Project Work Phase-II.

Ankit Saurabh (1SJ19CS008)

Anushka Ramesh (1SJ19CS013)

Manasa S A (1SJ19CS086)

Ruksar C (1SJ18CS081)

CONTENTS

Declaration	i
Abstract	ii
Acknowledgement	iii
Contents	iv
List of Figures	vii
List of Tables	viii

Chapter No	Chapter Title	Page No
1	INTRODUCTION	1
	1.1 Overview	1
	1.2 Problem Statement	2
	1.3 Significance and Relevance of Work	2
	1.4 Objectives	3
	1.5 Methodology	3
	1.6 Organization of the Report	4
2	LITERATURE SURVEY	7
3	SYSTEM REQUIREMENTS AND SPECIFICATIONS	10
	3.1 System Requirement and Specification	10
	3.2 Specific Requirement	10
	3.3 Hardware Specification	11
	3.4 Software Specification	13
	3.5 Functional Requirements	14
	3.6 Non Functional Requirements	15
	3.7 Performance Requirements	16
4	SYSTEM ANALYSIS	18
	4.1 Existing Systems	18
	4.2 Existing System Limitations	18

4.3 Proposed System	19
4.4 Proposed System Advantages	20
5	SYSTEM DESIGN
5.1 Project Modules	21
5.2 Activity Diagram	24
5.3 Use Case Diagram	26
5.4 Data Flow Diagram	27
5.5 Sequence Diagram	29
6	IMPLEMENTATION
6.1 Module Split up and Explanation	30
6.2 Algorithm used for each modules	31
7	TESTING
7.1 Methods of Testing	35
7.1.1 Unit Testing	35
7.1.2 Validation Testing	35
7.1.3 Functional Testing	37
7.1.4 Integrational Testing	38
7.1.5 User Acceptance Testing	39
7.2 Test Cases	41
8	PERFORMANCE ANALYSIS
9	CONCLUSION & FUTURE ENHANCEMENTS
BIBLIOGRAPHY	
APPENDIX	
Appendix A: Screen Shots	
Appendix B: Abbreviations	
PAPER PUBLICATION DETAILS	

LIST OF FIGURES

Figure No.	Name of the Figure	Page No.
Figure 5.1	Architecture of ADRS	22
Figure 5.2	Activity Diagram of Proposed System	24
Figure 5.3	Use Case Diagram of Proposed System	26
Figure 5.4	Steps involved in capturing & mapping of fingerprint image for extraction of template and storing.	27
Figure 5.5	Extraction details of victims from various databases using ADRS architecture	27
Figure 5.6	Sequence Diagram of Proposed System	29
Figure 8.1	Comparison of different models with our proposed models	45
Figure 11.1	Login Page With Biometric Authentication	51
Figure 11.2	Option Page to choose between actions	51
Figure 11.3	Registration Page to add new User	52
Figure 11.4	Fetch Page to retrieve user details	52
Figure 11.5	Fetch Page after retrieving user details using FingerPrint	53
Figure 11.6	3M FingerPrint Capturing Tool	54
Figure 11.7	Project Demonstration of Speedy Line	55

LIST OF TABLES

Table No.	Table Name	Page No.
Table 2.1	Percentage of accuracies obtained by different models	9
Table 7.1	Test Cases for the Proposed System	42

CHAPTER - 1

INTRODUCTION

1.1 Overview

Reports of Road and safety conveys that the innumerable deaths from accidents happening in day-to-day life are not automated and stored in databases; due to which the information related to those victims are lacking far behind and sometimes die in the hospitals itself without recording it in the secured databases. Even, if stored it takes lot of manual work to resolve the matter which is a time-consuming process that takes minimum 1 week to 6 months as per the survey reports of Ministry of Road Transport & Highways (MoRTH) and National Crime Records Bureau (NCRB) data [1] [2]. Especially, in private hospitals if the victim gets admitted their information is temporarily saved in computers and later it will become obsolete. If there's an emergency and it is needed by Government officials, then only the details of victim(s) will be recorded for future analysis that too in very rare cases.

India in the year 2021, had recorded 1.73 lakh deaths due to traffic accidents as per the recent NCRB data. The countries located in Africa, Latin America and Asia do not have proper information related to missing children. But, on an average the missing children's data every year throughout world-wide is predicted to be around 8 Lakhs exclusively from United States. Similarly, around 2.3 Lakhs and 1 Lakh are from United Kingdom and Germany countries, whereas 40K and 50K are from Brazil and Canada respectively as per the survey of 2017. As per the latest figures of the National Crime Records Bureau (NCRB), 59,262 children went missing in India in 2020. Whenever, a child is lost or a serious accident happens and the victim is in an unconscious state, during this process the expert team under the supervision of traffic police to be called and finger prints have to be recorded and saved, for further analysis. This project develops a system where immediately, these fingerprint impressions have to be mapped with the existing databases available with the government that constitutes of information pertaining to Aadhaar card [3], driving license, registration of vehicle, Voter ID, and other relevant databases. Once, the details are retrieved, the family members of the victims need to be traced using cross-referencing technique from those databases which extracts the colleague's mobile numbers.

Subsequently, this expert team will intimate to the colleagues or family members regarding the status of the lost child or victim by identifying the name, address, mobile numbers and

other details which are retrieved. A notification will also be sent to the hospitals located in a proximity of 100 meters from the site of the accident so response.

1.2 Problem Statement

Complaints related to accidental and incident cases are not updated effectively in online databases especially by Private and Government hospitals, due to which the information details of such victims are missing. The problems faced in identifying such victims from unexpected accidents and incidents are difficult for tracing their details from existing methodology adopted.

The problem statement covers the following topics

- a) Difficult to maintain offline database.
- b) Frequent updating of private hospital databases.
- c) Connecting lost or injured victims to their relatives.
- d) Tracing down the lost and mentally sick people.
- e) Unlocking the potential of Aadhaar to help people connect with their loved ones.
- f) Enhancing the people tracking system in India.

1.3 Significance and Relevance of Work

The main application of this project is in the field of tracing of victims who are met with incident or accident in situations where the victims are speechless due to unstable health condition.

- a) The project utilizes fingerprint recognition technology to retrieve user details from the Aadhaar database.
- b) This technology is a reliable means of identification that is difficult to replicate, reducing the risk of mistaken identity and improving the efficiency of organizations that use it.
- c) The system can benefit hospitals and police stations, allowing them to quickly retrieve user details in emergency situations.
- d) Vulnerable individuals such as lost and mentally challenged people who may not be able to provide their personal information or identification documents can be easily identified and provided with the necessary assistance.

- e) The use of fingerprint recognition technology can significantly reduce cases of mistaken identity, which can often occur with vulnerable individuals.
- f) The proposed system has the potential to improve the safety, well-being, and efficiency of society as a whole.

1.4 Objectives

The objectives of this project are as follows:

- a) To develop a system that utilizes fingerprint recognition technology to retrieve user details from the Aadhaar database.
- b) To improve the accuracy and efficiency of user identification in hospitals and police stations by providing a reliable means of identification that is difficult to replicate.
- c) To provide vulnerable individuals such as lost and mentally challenged people with an easy means of identification and necessary assistance.
- d) To reduce cases of mistaken identity and improve outcomes for individuals who require medical or legal assistance.
- e) To enhance the safety, well-being, and efficiency of society as a whole.
- f) To ensure that the proposed system addresses privacy and security concerns associated with the use of biometric data.
- g) To explore the potential for further development of the proposed system, including expansion to cover more sectors and databases and the development of more advanced algorithms and techniques.

1.5 Methodology

The main application of this project is in the field of tracing of victims who are met with incident or accident in situations where the victims are speechless due to unstable health condition. The project can be used in all the sectors where the user's data is stored and retrieved using biometric systems. The other sectors other than the hospitals where this project can be used is in offices and schools. The proposed system which stores and recognizes the fingerprint later helps in identifying the personal details of victims effected due in incidents and accidents known as Automated Details Retrieval System (ADRS) is developed.

The entire system is subdivided into three main parts:

- a) Capturing, Storing and Processing fingerprint impressions.
- b) Matching and fetching the data from database
- c) Cross-Checking and displaying the data.

1.5.1 Capturing, Storing and Processing fingerprint impressions.

The fingerprint impression is received by the sensor device that captures, processes and extracts features by improving the quality of fingerprint resulting in generation of template which is stored in database. Subsequently, the extracted template is mapped with the Aadhaar database. If the match is found it proceeds for further analysis.

1.5.2 Matching and Fetching the data from database

In this step, if the information pertaining to the template is matched with the Aadhaar database, the details of the victim are retrieved and stored temporary in the database. In case, if the mismatch occurs for the extracted template due to various reasons such as improper fingerprint impression or failure in detection process of the template from the Aadhaar database. Then, the system halts with a request to re-scan the fingerprint impression

1.5.3 Cross-Checking and displaying the data

Finally, the generated report is cross-checked using Mobile number, Address, Name, or other details which are retrieved by the ADRS using Reinforcement machine learning technique. So, that the information of victim is given to their family members.

Once the Aadhaar details are successfully, retrieved on matching of Fingerprint template from Aadhaar database. Subsequently, the Crawler and Component picks the Aadhaar details and tries to search (i.e. maps) in PAN database. From the PAN database, unique PAN number along with the employment details, Income, Bank details and other necessary information is also retrieved.

1.6 Organization of the Report

1.6.1 Chapter 1

The overview of the report is that project develops a system where immediately, these fingerprint impressions have to be mapped with the existing databases available with the

government that constitutes of information pertaining to Aadhaar card [3], driving license, registration of vehicle, Voter ID, and other relevant databases.

1.6.2 Chapter 2

The Literature Review is a comprehensive summary of previous research on a topic. The literature review surveys scholarly articles on the various biometric authentication, fetching the data and handling storages for our database.

1.6.3 Chapter 3

The System Requirements and Specification is a document that captures complete description about how the system is expected to perform and the ways to meet the various functional and non- functional requirements.

1.6.4 Chapter 4

This Chapter talks about system analysis that is the process of examining a system or a business problem to identify its components, their interrelationships, and their functions. It involves studying the existing system or process, determining its strengths and weaknesses, and developing strategies to improve it. System analysis is often used in software engineering to identify the requirements of a software system and to design solutions that meet those requirements.

1.6.5 Chapter 5

This chapter talks about system design that is the process of defining the architecture, components, modules, interfaces, and data for a system to satisfy specified requirements. It involves the creation of a detailed plan or blueprint for the development of the system. System design is a critical phase in the system development life cycle, and it is essential to ensure that the system can meet its requirements efficiently and effectively.

1.6.6 Chapter 6

The implementation sections talks about the generic working model of our project on society and its users and the fields in which it can be used. This also helps in finding out the benefits of the final project model being developed.

1.6.7 Chapter 7

The testing section is a critical component of any software development project, and it is typically included in the project's documentation or report. This section outlines the testing approach, test cases, and test results used to validate the software system. The primary goal of testing is to ensure that the software system functions as expected and meets the requirements identified during the system analysis and design phases.

1.6.8 Chapter 8

The performance analysis section of a project report provides an evaluation of how well the software system performs under various conditions. The purpose of this section is to measure the system's response time, throughput, scalability, and reliability. This analysis is important because it can help identify performance bottlenecks and provide insights into areas of the system that require optimization.

1.6.9 Chapter 9

The conclusion and future enhancement section summarizes the main findings, conclusions, and recommendations from the project. This section provides an opportunity to reflect on the project's successes and challenges and to outline potential areas for future improvement.

CHAPTER - 2

LITERATURE SURVEY

Vandana *et al* (2021) proposed a methodology for a study of biometric identification and verification system. The author refereed Identification as the authentication of an individual performed by matching the biometric test sample to the trained structure stored in the database and Verification as the authentication of an individual performed by marching the test image to the claimed biometric traits in the database [4].

L.Arun Kumar *et al* (2021) suggested Biometric Authentication using Raspberry Pi. The main basis of this work is uniqueness, which makes each person different. They have integrated biometric devices with Raspberry Pi in order to implement an authentication system. Fingerprint enrollment is done using Raspberry Pi fingerprint Module. Feature detection and feature matching is performed with cv2.flannBasedMatcher to authenticate the person [7].

Mohammed Mahmood Ali *et al* (2021) proposed an Automated Details Retrieval System that efficiently traces the personal history along with 6 family relatives of affected victims with the use of Machine learning, facial recognition systems, Edge computing on cloud platforms, Smart IoT technologies and Artificial Intelligence algorithms[6].

E.Esekhaigbe *et al* (2022) proposed a methodology for design and implementation of a fingerprint- based biometric access. The authors give an elaborated description of the components of the system such as power supply, fingerprint module, LCD, H-bridge motor, micro controller, as well as control buttons, buzzer and signal LEDs. The design circuit was tested and found to work[5] .

Jyotsna Nalawade *et al* (2022) proposed a theory of fingerprint biometric for internet of things. The author proposed a multi-modal biometric authentication systems and stated that it is better than single- modal biometric identification systems [8].

Yogitha *et al* (2021) stated, Fingerprint recognition is a highly reliable method for human identification, with the validity and consistency of fingerprint matching being well

established. Traditionally, feature extraction was performed before comparing a pair of fingerprints. However, recent advancements in Convolutional Neural Networks (CNNs) have shown promising results for various image processing tasks. Despite this, there have been few attempts to develop a complete CNN-based method for addressing fingerprint recognition challenges. In this study, we aimed to address this gap by developing a CNN-based fingerprint matching system that can directly learn fingerprint patterns from raw pixels in images. [13]

Ali Fadhil Yaseen *et al*(2022) proposed a deep learning framework for fingerprint recognition that uses Convolutional Neural Network to learn and represent the features of fingerprints for recognition purposes. The proposed model has been trained on a large-scale dataset and the simulation results demonstrate that it is a reliable system for developing artificial identification, which can be adapted to images with different characteristics. The proposed system has demonstrated high accuracy, achieving 100% accuracy for both the training and validation datasets. This indicates that the system is robust, reliable, and efficient. In future research, the authors plan to optimize the structure to improve learning speed for faster matching and enhance the recognition performance of images with noise.[9]

Bandi Baby Sravani et al a new application that uses deep learning CNN model for biometric authentication of fingerprint detection in both contact-based and contact-less scenarios. The study proposes a CNN-based framework for accurately matching contactless and contact-based fingerprint images. The framework involves training a multi-Siamese CNN using fingerprint details, respective ridge map, and a specific area of the ridge map. The authors generate a distance-aware loss function using deep fingerprint representation, and concatenate the representations for more accurate cross-comparison. The proposed methodology is tested on two publicly available databases that contain contactless 2D fingerprints and respective contact-based fingerprints. [12]

Md Rakib Hasan *et al*(2021)proposed that their system created is an affordable and accurate method for real-time authentication. It was primarily designed as an alternative to signature-based authentication methods used in examinations, but can be used in any situation that requires authentication. The system uses encryption to increase its reliability and security. However, there is room for improvement in the accuracy of the feature matching technique

employed. While Fast Library for Approximate Nearest Neighbors is faster, it can struggle to match key points in fingerprints.[6]

,Masoud Moradi *et al*(2022)proposed to develop an encryption scheme for the Internet of Things that meets security requirements and material constraints of connected objects. The encryption scheme is intended to protect user authentication information and data exchange during a single session. The proposed scheme is based on biometric fuzzy commitment and incorporates various techniques for selection, encoding, features vector, quantitative code, and compression code, all of which are secure. Fuzzy encoding is used for encryption due to its low computational complexity. Analog code technique is used for the private cipher key to increase the complexity of selecting the cipher key for fuzzy encoding[10].

Table 2.1: Percentage of accuracies obtained by different models

Features	Proposed ADRS	Finger, Iris, Systems	FLDNet	Multimodal
Aadhar	94.39	92.31	85.32	87.77
Passport Number	90.39	67	76.32	89.11
PAN	85.32	79.32	57.21	45.32
Fingerprint	96.39	90.12	97.21	94.23

CHAPTER - 3

SYSTEM REQUIREMENTS AND SPECIFICATION

3.1 System Requirements and Specification

System Requirement Specification (SRS) is a central report, which frames the establishment of the product advancement process. It records the necessities of a framework as well as has a depiction of its significant highlight. A SRS is essentially an association's seeing (in composing) of a client or potential customer's frame work necessities and conditions at a specific point in time (generally) before any genuine configuration or improvement work. It's a two-way protection approach that guarantees that both the customer and the association comprehend alternate's necessities from that viewpoint at a given point in time. The composition of programming necessity detail lessens advancement exertion, as watchful audit of the report can uncover oversights, mistaken assumptions, and irregularities ahead of schedule in the improvement cycle when these issues are less demanding to right. The SRS talks about the item however not the venture that created it, consequently the SRS serves as a premise for later improvement of the completed item. The SRS may need to be changed, however it does give an establishment to proceed with creation assessment. In straightforward words, programming necessity determination is the beginning stage of the product improvement action. The SRS means deciphering the thoughts in the brains of the customers –the information, into a formal archive –the yield of the prerequisite stage. Subsequently the yield of the stage is a situated of formally determined necessities, which ideally are finished and steady, while the data has none of these properties.

3.2 Specific Requirement

The specific requirements for the proposed system that utilizes fingerprint recognition technology to retrieve user details from the Aadhaar database are as follows:

- a) **Hardware:** The system requires a fingerprint scanner for capturing fingerprints and a computer system for processing and storing the retrieved user details. The hardware components should be capable of processing a large volume of fingerprint data efficiently.

- b) **Software:** The system requires software that can match the fingerprint data captured by the fingerprint scanner with the corresponding user details stored in the Aadhaar database. The software should be capable of processing data in real-time and should provide accurate results.
- c) **Database:** The system requires access to the Aadhaar database, which should be secured to ensure the privacy and security of the data. The database should be capable of storing a large volume of user details and should be easily accessible to authorized users.
- d) **Security:** The system should include measures to ensure the privacy and security of biometric data. The system should be designed to prevent unauthorized access to the database and ensure that user data is not compromised in any way.
- e) **User Interface:** The system should have a user-friendly interface that allows authorized users to quickly and easily retrieve user details from the Aadhaar database using fingerprint data. The interface should be intuitive and easy to use, even for users with limited technical knowledge.
- f) **Accuracy:** The system should be designed to provide accurate results, with a low probability of false positives or false negatives. The system should be capable of handling variations in fingerprint data due to factors such as age, gender, and ethnicity.
- g) **Compatibility:** The system should be compatible with existing systems used by hospitals and police stations. The system should be designed to integrate with existing workflows and procedures to minimize disruption to operations.

3.3 Hardware Specification

Hardware Devices used for implementing this project are as follows:

a) Desktop Computer

- I. **Processor:** Intel Core i5 or above
- II. **RAM:** 8 GB or above

- III. **Hard Disk:** 500 GB or above
- IV. **Operating System:** Windows 10/ Windows 11
- V. **USB ports:** At least two USB ports
- VI. **Internet connectivity:** Broadband connection with a speed of at least 2 Mbps.

The proposed system requires a personal computer (PC) with an Intel Core i5 processor or above, 8 GB of RAM or more, a 500 GB hard disk or more, and the Windows 10 operating system. The PC should have at least two USB ports, internet connectivity, and network connectivity. The PC should also have a backup power supply, such as an uninterrupted power supply (UPS), to ensure that the system remains operational during power outages. A high-speed internet connection is necessary to ensure that the system can retrieve user details from the Aadhaar database quickly. The PC should also be compatible with other hardware components, such as the fingerprint scanner and printer, and should be able to run the software used for matching the fingerprint data with the user details stored in the Aadhaar database.

a) 3M Cogent CSD 200 FingerPrint Scanner

- I. **Image Resolution :** 500dpi
- II. **Platen Area :** 16.0mm X 18.0 mm
- III. **Interface :** USB 2.0
- IV. **Operating Temperature :** -10 to 55 Degree Celcius
- V. **Image Format :** JPEG, RAW, BMP
- VI. **Size and Weight :** 69mm X 43mm X 16mm, 120gms
- VII. **Compatibility :** Windows, Linux and Android

The Cogent CSD 200 fingerprint scanner is a reliable and high-quality biometric device used for capturing high-quality fingerprints. It is compact and lightweight, making it easy to carry and use. With a high image resolution of 500 dpi and a platen area of 16.0 mm x 18.0 mm, it can capture fingerprints of various sizes accurately. The scanner has a USB 2.0 interface, enabling easy connection to a personal computer (PC), and is compatible with various operating systems, including Windows, Linux, and Android. The scanner supports multiple image formats, including RAW, BMP, and JPEG, making it compatible with various software applications. Its durability makes it suitable for use in high-traffic areas such as hospitals and

police stations, and it is an ideal biometric device for use in retrieving user details from the Aadhaar database.

3.4 Software Specification

The proposed system requires several software applications to function correctly. These include:

- a) **Tkinter (Python)**
- b) **MySQL**
- c) **Jupyter Notebook**
- d) **Pycharm**

- a) **Tkinter:** Tkinter is a Python-based GUI toolkit that is used to create graphical user interfaces. The system uses Tkinter to create a user-friendly interface that enables users to interact with the system easily.
- b) **MySQL:** MySQL is an open-source relational database management system that is used to store and retrieve user data. The system uses MySQL to store user data retrieved from the Aadhaar database.
- c) **Jupyter Notebook:** Jupyter Notebook is an open-source web application that is used to create and share documents containing live code, equations, visualizations, and narrative text. The system uses Jupyter Notebook to develop and test the matching algorithm used to match fingerprint data with user data stored in the MySQL database.
- d) **Pycharm:** Pycharm is a Python-based integrated development environment (IDE) that is used to write, test, and debug Python code. The system uses Pycharm to write and test the software used to match fingerprint data with user data stored in the MySQL database.

The software used in the system should be compatible with the hardware components used, such as the fingerprint scanner and the PC, to ensure that the system functions correctly. Additionally, the software should be easy to use and maintain, and should provide reliable performance.

3.5 Functional Requirements

The functional requirements of the proposed system include:

- a) **User Registration:** The system should allow users to register themselves with their Aadhaar number and their fingerprint data.
- b) **Fingerprint Scanning:** The system should be able to scan fingerprints using the Cogent CSD 200 fingerprint scanner and store the fingerprint data in the system's database.
- c) **Aadhaar Database Integration:** The system should be able to retrieve user data from the Aadhaar database using the Aadhaar number.
- d) **User Identification:** The system should be able to match fingerprint data with user data stored in the system's database to identify the user.
- e) **User Details Retrieval:** The system should be able to retrieve user details such as name, address, and contact information from the Aadhaar database.
- f) **User Details Update:** The system should allow users to update their personal details in the system's database.
- g) **Access Control:** The system should have access control mechanisms to ensure that only authorized personnel can access the system.
- h) **Reporting:** The system should be able to generate reports on user registrations, user identifications, and user details updates.
- i) **Data Security:** The system should be secure and protect user data from unauthorized access and data breaches.
- j) **System Maintenance:** The system should be easy to maintain, and any software or hardware issues should be resolved quickly to minimize downtime.

- k) **User Interface:** The system should have a user-friendly interface that is easy to use and navigate.

The functional requirements of the system should be met to ensure that the system functions as intended and meets the needs of its users.

3.6 Non Functional Requirements

The non-functional requirements of the proposed system include:

- a) **Performance:** The system should have a fast response time to ensure that user authentication and user data retrieval happen quickly.
- b) **Reliability:** The system should be reliable and should be able to handle a large number of users without any system crashes or downtime.
- c) **Scalability:** The system should be scalable and should be able to accommodate an increasing number of users without any degradation in performance.
- d) **Usability:** The system should be easy to use and should have a user-friendly interface that is intuitive and easy to navigate.
- e) **Accessibility:** The system should be accessible to all users, including those with disabilities, and should be compliant with accessibility standards.
- f) **Security:** The system should have robust security features to prevent unauthorized access to user data and protect against data breaches.
- g) **Compatibility:** The software used in the system should be compatible with the hardware components used, such as the fingerprint scanner and the PC.
- h) **Portability:** The system should be portable and should be able to run on different platforms and operating systems.

- i) **Maintainability:** The system should be easy to maintain, and any software or hardware issues should be resolved quickly to minimize downtime.
- j) **Documentation:** The system should have clear and concise documentation that explains how to use the system and how to troubleshoot common issues.

The non-functional requirements of the system should be met to ensure that the system is reliable, secure, and easy to use, and provides a seamless user experience.

3.7 Performance Requirements

The performance requirements of the proposed system include:

- a) **Response Time:** The system should have a fast response time to ensure that user authentication and user data retrieval happen quickly. The response time should be less than 1 minute.
- b) **Throughput:** The system should be able to handle a large number of user requests simultaneously without any system crashes or downtime. The system should be able to handle at least 100 requests per minute.
- c) **Processing Time:** The system should be able to process user data quickly and efficiently. The processing time for user data should be less than 1 minute.
- d) **Memory Usage:** The system should use a minimum amount of memory to avoid slowing down the system or causing system crashes. The system should use less than 500 MB of memory.
- e) **Storage Space:** The system should be able to store a large amount of user data without running out of storage space. The system should be able to store at least 100,000 user records.

- f) **Fingerprint Scanning Accuracy:** The system should have a high level of fingerprint scanning accuracy to ensure that user authentication is accurate and reliable. The fingerprint scanning accuracy should be at least 70%.

- g) **Compatibility:** The software used in the system should be compatible with the hardware components used, such as the fingerprint scanner and the PC. The system should be compatible with the Cogent CSD 200 fingerprint scanner and should be able to run on any modern personal computer.

The performance requirements of the system should be met to ensure that the system can handle a large number of user requests, process data quickly and efficiently, and provide accurate and reliable user authentication.

CHAPTER - 4

SYSTEM ANALYSIS

4.1 Existing Systems

There are a few existing systems that are similar to the proposed system that uses fingerprint recognition technology for user authentication and data retrieval. These systems include:

- a) **Biometric Attendance Systems:** These systems are commonly used in workplaces to track employee attendance and hours worked. They use fingerprint recognition technology to identify employees and record their time of arrival and departure. While these systems are effective in tracking employee attendance, they may have limitations in terms of accuracy and scalability.
- b) **Aadhaar-based Authentication Systems:** The Aadhaar system is a biometric database maintained by the Indian government that contains the fingerprints and other biometric data of Indian citizens. This system is used for a variety of purposes, including authentication for government services and financial transactions. While the Aadhaar system is widely used and has proven to be effective in many cases, it has also faced criticisms for security and privacy concerns.
- c) **Security Systems:** Some security systems, such as those used in law enforcement or military applications, use fingerprint recognition technology to identify individuals and grant access to secure areas or information. These systems are typically more advanced than biometric attendance systems and are designed to be highly accurate and secure. However, they may also be expensive and may have limited scalability.

4.2 Existing System Limitation

While existing systems that use fingerprint recognition technology have proven to be effective in many cases, they also have some limitations that need to be addressed. Some of the common limitations of these systems are:

- a) **Accuracy:** One of the main limitations of fingerprint recognition technology is accuracy. Fingerprint recognition can be affected by factors such as dirt, moisture, cuts, or scars on the finger. Inaccurate readings can result in false positives or false negatives, which can cause inconvenience or security risks.
- b) **Scalability:** Existing systems may have limitations in terms of scalability, meaning they may not be able to handle large volumes of users or data. This can be a problem in situations where multiple users need to be authenticated and their information needs to be retrieved quickly and accurately.
- c) **Compatibility:** Fingerprint recognition systems may not be compatible with all devices or software, which can limit their usefulness in certain situations. For example, a system that only works on specific devices may not be suitable for users who use different devices or platforms.
- d) **Security:** Fingerprint recognition systems are vulnerable to security breaches, such as hacking or data theft. If the system is not properly secured, it can compromise user data or result in identity theft.
- e) **Cost:** Some fingerprint recognition systems can be expensive to implement, which can be a barrier for small businesses or organizations that have limited resources.
- f) **Privacy concerns:** The use of biometric data, including fingerprints, raises privacy concerns. Users may be hesitant to provide their biometric data, especially if they are not fully informed about how it will be used and protected.

4.3 Proposed System

Our proposed system is designed to use fingerprint recognition technology to authenticate users and retrieve their information from the Aadhaar database. The system is comprised of hardware and software components that work together to provide accurate and reliable authentication and data retrieval.

The hardware component of our system is the Cogent CSD 200 Fingerprint Scanner. This scanner is designed to capture high-quality fingerprint images, even in challenging

environments. It has a resolution of 500 dpi, which ensures that fingerprint images are sharp and clear. The scanner is also designed to be durable and reliable, making it ideal for use in high-volume settings.

The software component of our system is designed to work with the Cogent CSD 200 Fingerprint Scanner and the Aadhaar database. It is built using the Python programming language and utilizes several libraries, including Tkinter for the user interface, MySQL for the database management, and Jupyter Notebook and PyCharm for development and testing.

The system works by capturing a user's fingerprint using the Cogent CSD 200 Fingerprint Scanner. The fingerprint image is then compared against the fingerprint images stored in the Aadhaar database. If the fingerprint matches an existing record, the system retrieves the user's information from the database and displays it on the screen.

The system is designed to be scalable, which means it can handle large volumes of users and data. It is also designed to be compatible with different devices and software, making it versatile and useful in a variety of settings. The system is secure, with built-in features that protect against unauthorized login. It is also cost-effective, which makes it accessible to small businesses and organizations with limited resources.

Overall, our proposed system is a reliable, secure, and cost-effective solution for user authentication and data retrieval. It offers several advantages over existing systems, making it a valuable tool for hospitals, police, and other organizations that need to authenticate users and retrieve their information quickly and accurately.

4.4 Proposed System Advantages

The proposed system that uses fingerprint recognition technology to retrieve user details from the Aadhaar database has several advantages over existing systems. Some of the advantages of our proposed system are:

- a) **Accuracy:** Our system is designed to provide accurate and reliable fingerprint recognition, even in challenging environments. This ensures that users are authenticated and their information is retrieved accurately.

- b) **Scalability:** Our system is designed to be scalable, which means it can handle large volumes of users and data. This is important in situations where multiple users need to be authenticated and their information needs to be retrieved quickly and accurately.
- c) **Compatibility:** Our system is designed to be compatible with different devices and software, which makes it more versatile and useful in a variety of settings.
- d) **Security:** Our system is designed to be secure, with built-in features that protect against unauthorized login. This ensures that user data is safe and protected at all times.
- e) **Cost-effective:** Our system is designed to be cost-effective, which makes it accessible to small businesses and organizations with limited resources.
- f) **User-friendly:** Our system is designed to be user-friendly, with an intuitive interface that is easy to use and understand. This ensures that users can quickly and easily authenticate themselves and retrieve their information.

Overall, our proposed system offers several advantages over existing systems, making it a reliable, secure, and cost-effective solution for user authentication and data retrieval.

CHAPTER - 5

SYSTEM DESIGN

This ADRS initially takes the finger-print as an input through a sensor-based device which successively extracts the Whorls & Ridges of fingers through sequence of processing steps consisting of Acquisition, pre-process, Feature Extraction, Normalization and Pattern Formation resulting in generation of pattern formation known as “template”. Once the template features of a finger is extracted those are preserved and sent for storing into the assigned databases. The next step is extracted template features are forwarded to Crawler and mapper component, which pursues forward for searching and mapping with template features of recorded finger-prints of people that are present in various databases.

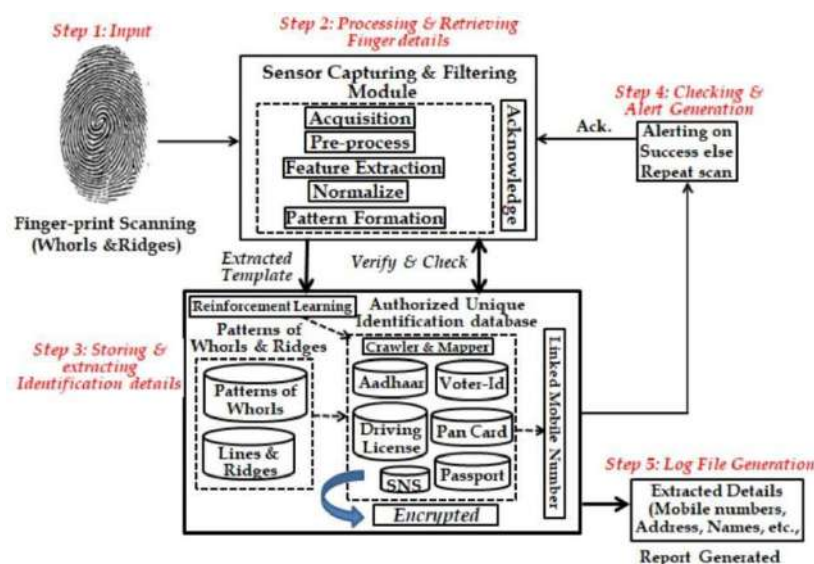


Figure 5.1 Architecture of ADRS

5.1 Project Modules

Our proposed system can be divided into several modules, each with a specific function. The modules of the project are as follows:

- User registration:** This module allows users to register by providing their personal information, including their name, address, date of birth, and Aadhaar number. Once registered, the user's information is stored in the Aadhaar database.

- b) **Fingerprint scanning:** This module is responsible for capturing the user's fingerprint using the Cogent CSD 200 Fingerprint Scanner. The scanner produces a high-quality fingerprint image that is used for authentication.
- c) **Fingerprint matching:** This module compares the user's fingerprint image against the fingerprint images stored in the Aadhaar database. If the fingerprint matches an existing record, the system retrieves the user's information from the database.
- d) **User authentication:** This module is responsible for verifying the user's identity based on their fingerprint and retrieving their information from the Aadhaar database. If the user is authenticated, the system displays their information on the screen.
- e) **User search:** This module allows authorized users to search for specific individuals by name, address, or Aadhaar number. The system retrieves the user's information from the Aadhaar database and displays it on the screen.
- f) **Data management:** This module is responsible for managing the data stored in the Aadhaar database. It includes functions such as adding, modifying, and deleting user records.
- g) **User access management:** This module controls the access of different users to the system. It defines the roles and permissions of different users and ensures that only authorized users have access to sensitive information.

These modules work together to provide a reliable and secure system for user authentication and data retrieval. Each module is designed to be scalable, which means it can handle large volumes of users and data. The system is also designed to be user-friendly, with a simple and intuitive interface that makes it easy to use for both administrators and end-users.

5.2 Activity Diagram

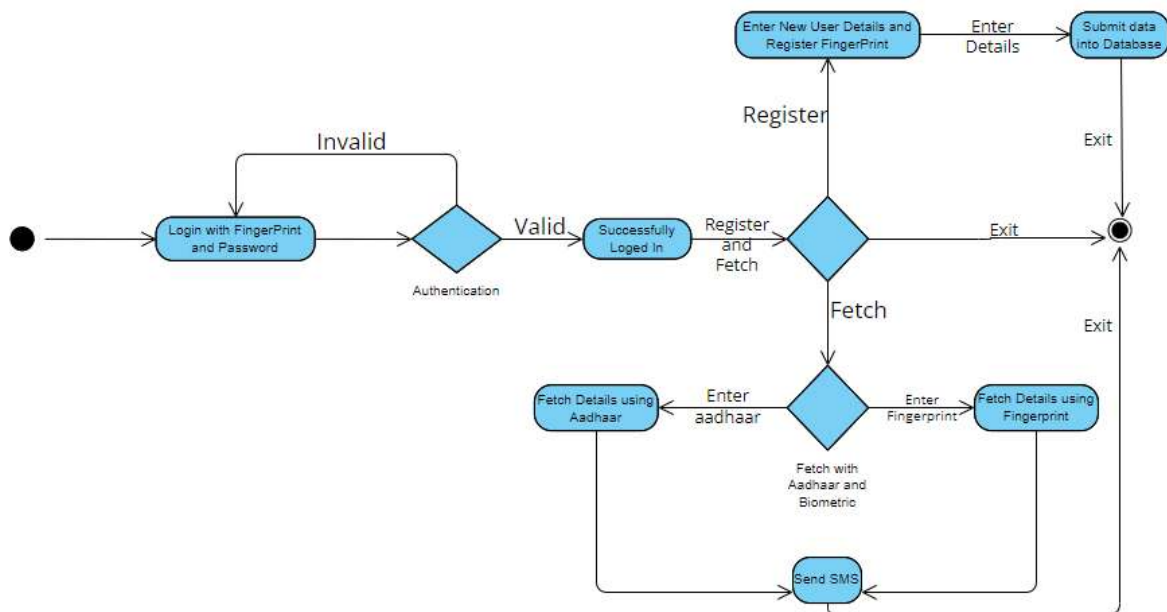


Figure 5.2 Activity Diagram of Proposed System

An activity diagram is a graphical representation of the flow of activities in a system. In our proposed system, the activity diagram consists of several activities that users and administrators can perform. Here is an explanation of the activity diagram:

- Login Module:** The login module allows authorized users to access the system using their username and password. The module provides a secure authentication process that ensures only authorized users can access the system. The login module also tracks user activities and generates activity logs for security and audit purposes.
- User Registration:** The first activity in the diagram is user registration. The user provides their personal information, including name, address, date of birth, and Aadhaar number. Once the user submits the information, it is stored in the Aadhaar database.
- Fingerprint Scanning:** The next activity is fingerprint scanning. The user's fingerprint is captured using the Cogent CSD 200 Fingerprint Scanner. The scanner produces a high-quality fingerprint image that is used for authentication.

-
- d) **Fingerprint Matching:** The third activity is fingerprint matching. The user's fingerprint image is compared against the fingerprint images stored in the Aadhaar database. If the fingerprint matches an existing record, the system retrieves the user's information from the database.
 - e) **User Authentication:** The fourth activity is user authentication. If the user's fingerprint matches an existing record, the system verifies the user's identity and retrieves their information from the Aadhaar database. If the user is authenticated, the system displays their information on the screen.
 - f) **User Search:** The fifth activity is user search. Authorized users can search for specific individuals by name, address, or Aadhaar number. The system retrieves the user's information from the Aadhaar database and displays it on the screen.
 - g) **SMS Module:** The SMS module allows the system to send an SMS notification to a pre-registered relative or emergency contact in case of an emergency. The user can register their relative's phone number during the registration process, and the system can use this information to send an SMS in case of an emergency. The SMS module uses the Fast2SMS API to send SMS notifications and provides a reliable and secure way to alert relatives in case of an emergency.

The activity diagram shows the flow of activities in our proposed system, from user registration to user access management. It illustrates how each activity is connected and how the system functions as a whole to provide a reliable and secure system for user authentication and data retrieval.

5.3 Use Case Diagram

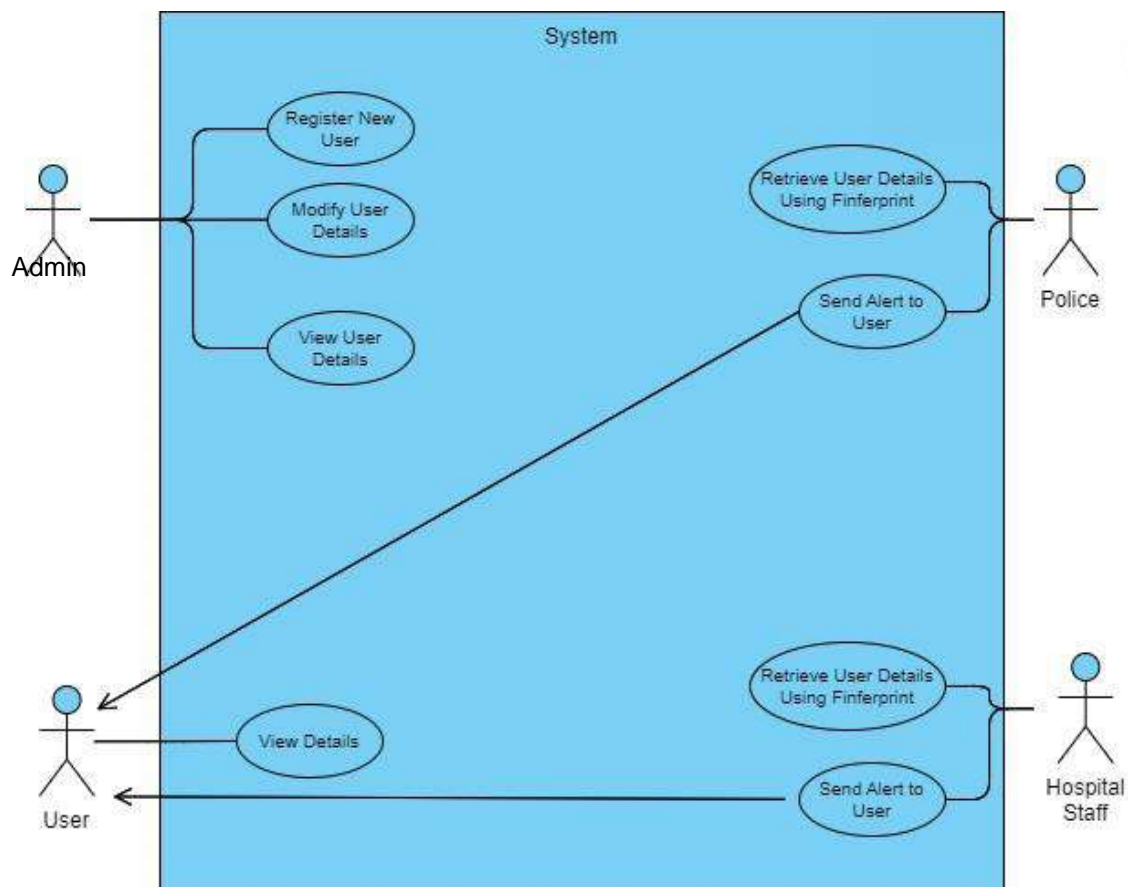


Figure 5.3 Use Case Diagram of Proposed System

In this use case diagram, there are four actors: the admin, police, hospital staff and user. Admin can create a new record for a new user and perform modification on that record while both actors (police and hospital staff) have to log in to the system to access its features. After logging in, the police can search for criminals/lost people using their fingerprints, while the hospital staff can search for patients who have been in accidents.

Once a search has been initiated, the system will display a list of potential matches. The police can then view the criminal/lost record of the person to see if they are a match, while the hospital staff can view the patient's medical record to confirm their identity. If Police and Hospital Staff wants to update family members of victims or criminals/lost people then they can do so by sending a SMS to their relatives.

5.4 Data Flow Diagram

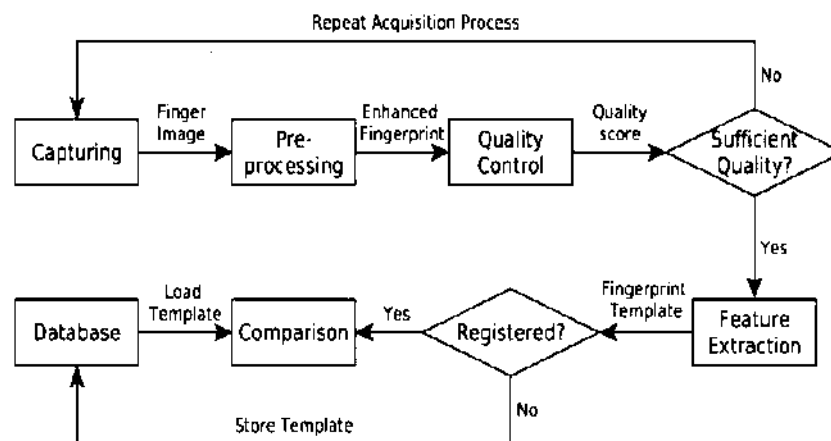


Figure 5.4 Steps involved in capturing & mapping of fingerprint image for extraction of template and storing database.

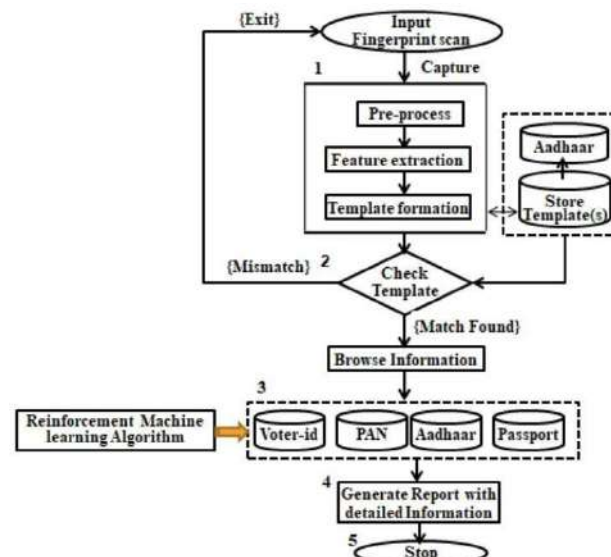


Figure 5.5 Extraction details of victims from various databases using ADRS architecture

The process of capturing and mapping a fingerprint image for extraction of a template and storing it in a database can be broken down into the following steps:

- a) **Image acquisition:** The first step is to capture a high-quality digital image of the fingerprint. This can be done using various methods such as optical scanners or capacitive sensors.
- b) **Pre-processing:** The captured image is then processed to remove any noise or artifacts that might affect the accuracy of the template extraction. This involves tasks such as filtering, enhancement, and normalization.

-
- c) **Feature extraction:** Next, the unique features of the fingerprint are extracted from the pre-processed image. These features are called minutiae and include ridge endings, bifurcations, and other distinctive characteristics.
 - d) **Template creation:** Once the minutiae have been extracted, a template is created by encoding these features into a compact representation. This template can be used for comparison with other templates to determine a match.
 - e) **Database storage:** The final step is to store the template in a database along with any associated metadata such as the person's name or identification number. The database can then be used for matching fingerprints in real-time applications such as access control or forensic analysis.
 - f) **Retrieval of the stored template:** The template of the fingerprint to be identified is retrieved from the database based on the person's identification number or any other relevant information.
 - g) **Comparison:** The retrieved template is then compared to the template created from the captured fingerprint image. The comparison process involves mathematical algorithms that calculate the degree of similarity between the two templates.
 - h) **Matching:** If the level of similarity between the two templates is above a predetermined threshold, the templates are considered to match, and the identity of the person is confirmed. If the level of similarity is below the threshold, the templates do not match, and the person's identity is not confirmed.
 - i) **Verification:** Once the templates are matched, additional verification steps may be taken to confirm the person's identity, such as asking for a password or a second form of identification.

5.5 Sequence Diagram

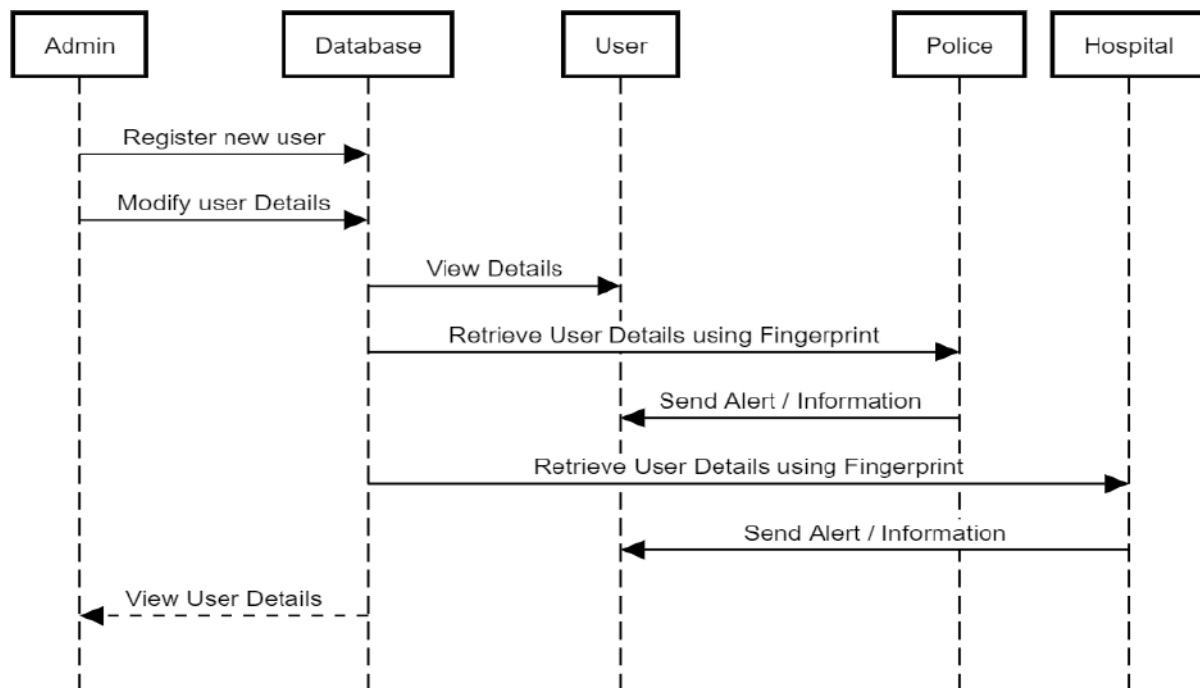


Figure 5.6: Sequence Diagram of Proposed System

In this sequence diagram, the admin registers a new user and their record gets added in database along with their fingerprint, admin also has access to modify details of user along with viewing user details whereas the user can only view his/her details from database. Police and Hospital staff have to log in to the system to access its features. After logging in, the police can search for criminals/lost people using their fingerprints, while the hospital staff can search for patients who have been in accidents.

Once a search has been initiated, the system will display a list of potential matches. The police can then view the criminal/lost record of the person to see if they are a match, while the hospital staff can view the patient's medical record to confirm their identity. If Police and Hospital Staff wants to update family members of victims or criminals/lost people then they can do so by sending a SMS to their relatives.

CHAPTER - 6

IMPLEMENTATION

6.1 Module Split up and explanation

6.1.1 GUI (GRAPHICAL USER INTERFACE)

- a) Tkinter is a Python library that is commonly used to create Graphical User Interfaces (GUI) for desktop applications. It provides a set of tools and widgets that can be used to create windows, buttons, text boxes, labels, and other graphical elements that allow users to interact with the application.
- b) We have used Tkinter to create a registration as well as fetching form for our Project, where user will enter data and fetch data by giving input to the text fields which is connected to our MySQL database.

6.1.2 Database

- a) MySQL is the world's most popular open-source database. Despite its powerful features, MySQL is simple to set up and easy to use.
- b) We use MySQL to store the training data which is required during the retrieval of information.
- c) MySQL is also used to retrieve the information of the victim when required.

6.1.3 Hardware – 3M Cogent CSD 200 Finger Scanner

- a) Cogent optical fingerprint scanners are all FBI App-F certified which are ideal for system integrators and solution providers because of their ergonomic design and easy-to-integrate SDK architecture.
- b) The CSD 200 is a single-digit optical fingerprint scanner with a tough and small design.
- c) It is User-friendly and Convenient to use.
- d) It offers a reliable, ergonomic, and cost-effective solution for enrollment, identity verification, and user identification.
- e) We use the device to scan the fingerprint of the victim and retrieve information.

6.2 Algorithms used for each modules

Fingerprint Matching Algorithm

- a) **Input:** Multiple fingerprint images
- b) **Output:** Matching score between pairs of fingerprints

Preprocessing

- I. Perform segmentation on each image to separate the foreground (fingerprints) from the background.
- II. Apply erosion operation to remove small noise and bumps from the image.
- III. Apply dilation operation to fill in gaps and smooth out the ridges.
- IV. Perform segmentation masking to create a binary mask of the fingerprint ridges.
- V. Normalize the image to adjust the brightness and contrast levels.
- VI. Crop the image to remove any unnecessary background.
- VII. Enhance the image to highlight the ridge patterns and remove noise.
- VIII. Apply thinning operation to thin the ridges.

Pseudo Code for Preprocessing

- I. Perform segmentation on each image to separate the foreground (fingerprints) from the background.
- II. Apply erosion operation to remove small noise and bumps from the image.
- III. Apply dilation operation to fill in gaps and smooth out the ridges.
- IV. Perform segmentation masking to create a binary mask of the fingerprint ridges.
- V. Normalize the image to adjust the brightness and contrast levels.
- VI. Crop the image to remove any unnecessary background.
- VII. Enhance the image to highlight the ridge patterns and remove noise.
- VIII. Apply thinning operation to thin the ridges.

Algorithm: Preprocessing

- a) **Input:** Fingerprint image
 - b) **Output:** Preprocessed image
-

- I. Convert the image to grayscale.
- II. Apply a thresholding method to the image to separate the foreground (fingerprints) from the background.
- III. Apply erosion operation with a structuring element to remove small noise and bumps from the image.
- IV. Apply dilation operation with a larger structuring element to fill in gaps and smooth out the ridges.
- V. Perform segmentation masking to create a binary mask of the fingerprint ridges by thresholding the dilation result.
- VI. Normalize the image using either histogram equalization or contrast stretching to adjust the brightness and contrast levels.
- VII. Crop the image to remove any unnecessary background by analyzing the binary mask and finding the bounding box of the fingerprint.
- VIII. Enhance the image by applying a filtering technique such as median filtering or Gabor filtering to highlight the ridge patterns and remove noise.
- IX. Apply thinning operation to thin the ridges by removing the ridge pixels that are not connected to any ridge endpoints.
- X. Return the preprocessed image.

There are many variations and optimizations that can be applied to each step of the preprocessing algorithm depending on the specific requirements of the application.

Feature Extraction

- I. Find line edges in the image using Canny edge detection algorithm.
- II. Find ridge orientation by calculating the gradient direction at each pixel using the Sobel filter.
- III. Compute ridge frequency by dividing the number of ridges by the average ridge spacing.
- IV. Find minutiae by locating the ridge endings and bifurcations in the thinned image.
- V. Remove minutiae clusters by analyzing the distance and angle between the minutiae.

Pseudo Code for Feature Extraction

- I. Find line edges in the image using Canny edge detection algorithm.
- II. Find ridge orientation by calculating the gradient direction at each pixel using the Sobel filter.
- III. Compute ridge frequency by dividing the number of ridges by the average ridge spacing.
- IV. Find minutiae by locating the ridge endings and bifurcations in the thinned image.
- V. Remove minutiae clusters by analyzing the distance and angle between the minutiae.

Algorithm: Feature Extraction

a) **Input:** Binary fingerprint image, ridge thickness, minutiae distance, angle threshold

b) **Output:** Minutiae list

- I. Apply Canny edge detection to the binary image to detect line edges.
- II. Apply Sobel filter to the image to calculate gradient direction at each pixel.
- III. Divide the image into blocks of size equal to ridge thickness.
- IV. For each block, compute the gradient orientation and frequency of the ridges using a Gabor filter.
- V. Combine the ridge frequency maps of all blocks to create a global ridge frequency map.
- VI. Threshold the global ridge frequency map to find ridge endings and bifurcations.
- VII. For each ridge ending, calculate its angle with respect to the local ridge direction.
- VIII. For each bifurcation, calculate the angle between the two ridges.
- IX. Remove minutiae that are closer than minutiae distance and have an angle difference less than angle threshold.
- X. Return the list of remaining minutiae.

Matching

- I. Compare the fingerprint images using a similarity measure such as Euclidean distance or correlation coefficient.

- II. Align the images using Hough transform to find the rotation and translation parameters that minimize the distance between the two images.
- III. Pair the minutiae from the two images based on their position and orientation.
- IV. Calculate a score for the match based on the number and quality of the minutiae pairs.

Pseudo Code For Matching

- I. Extract features (minutiae) from the preprocessed query and reference images.
- II. Compute a distance matrix between the minutiae of the query and reference images.
- III. Apply Hungarian algorithm to find the optimal matching between the query and reference minutiae.
- IV. Compute the similarity score between the matched minutiae pairs.
- V. If the score is above a threshold, consider the match as valid.

Algorithm: Matching

a) **Input:** Query image, reference image, matching threshold

b) **Output:** Match score

- I. Preprocess the query and reference images using the preprocessing algorithm.
- II. Extract minutiae features from the preprocessed images using the feature extraction algorithm.
- III. Compute the distance matrix between the query and reference minutiae using Euclidean distance.
- IV. Apply the Hungarian algorithm to find the optimal matching between the query and reference minutiae.
- V. Compute the similarity score between the matched minutiae pairs using a scoring function such as the Gabor filter.
- VI. If the score is above the matching threshold, consider the match as valid.
- VII. Return the match score.

CHAPTER - 7

TESTING

7.1 Methods of Testing

7.1.1 Unit Testing

Unit testing is a fundamental practice in software development that involves testing individual units or components of a program to ensure their correctness and functionality. It is a crucial part of the testing process as it helps identify bugs, errors, and unexpected behavior early in the development cycle.

During unit testing, each unit is tested independently to verify if it performs as expected. A unit can be a function, a method, a class, or even a module, depending on the granularity of the software being tested. By isolating these units and subjecting them to various test cases, developers can gain confidence in the reliability and accuracy of their code.

The first step performed was to identify the specific code components responsible for interacting with the fingerprint scanner. This included functions, classes, or modules that handle the scanner initialization, capturing fingerprint data, performing verification, or handling any errors or exceptions related to the scanner.

The unit tests were executed and reviewed the test results to identify any failed tests and investigate the causes of failure. The code was debugged and any issues discovered were fixed. All the test cases then passed successfully, indicating the correct functioning of the fingerprint scanner code.

7.1.2 Validation Testing

Performing validation tests on a biometric fingerprint scanner involved verifying its performance, accuracy, and compliance with specific requirements and standards. These tests have been conducted to ensure that the scanner meets the intended functionality and can reliably capture and verify fingerprint data. Approaches taken to performing validation tests on a biometric fingerprint scanner:

Define validation requirements: Clearly identify the validation requirements for the fingerprint scanner based on its intended use, specifications, and any relevant standards or regulations. This includes factors such as accuracy, speed, template size, security features, and environmental conditions.

Develop validation test cases: Design a set of test cases that cover various aspects of the fingerprint scanner's functionality and performance.

- a) **Accuracy test:** Verify the scanner's ability to correctly capture and match fingerprints by using known templates or sample fingerprints with known outcomes.
- b) **Speed test:** Evaluate the scanner's speed by measuring the time it takes to capture and process fingerprints, and compare it against the specified performance requirements.
- c) **Template size test:** Test the scanner's ability to handle fingerprints with different template sizes and assess if it meets the required storage capacity.
- d) **Environmental test:** Assess the scanner's performance under various environmental conditions, such as different lighting conditions, humidity levels, or temperature ranges.

Prepare test environment: Set up a controlled test environment that closely resembles the conditions under which the scanner will be used. Ensure that the environment factors, such as lighting, temperature, and cleanliness, align with the intended deployment settings.

Execute validation tests: Perform the validation tests according to the predefined test cases. Use the selected test environment, and carefully follow the test procedures for each case. Record the results, including any deviations or issues encountered during testing.

Analyze test results: Evaluate the test results and compare them against the defined validation requirements. Identify any discrepancies or failures and thoroughly investigate the causes. If a test fails, debug the issues and determine whether it is a software or hardware-related problem.

Document test findings: Document the test procedures, results, and any issues encountered during the validation process. This documentation will serve as a reference for future analysis, troubleshooting, and compliance purposes.

Iterative testing and improvement: Validation testing is an iterative process, and it may require multiple rounds of testing and refinement. Address any identified issues or non-compliance areas, and retest the scanner to ensure that the necessary improvements have been implemented.

7.1.3 Functional Testing

Performing functional tests on a biometric fingerprint scanner involves verifying its core functionality and ensuring that it performs the expected operations correctly. Functional testing focuses on evaluating the scanner's ability to capture, store, and verify fingerprint data accurately.

Identify functional requirements: Review the functional requirements of the fingerprint scanner, including its intended features and capabilities.

Define test scenarios: Identify and define test scenarios that cover different aspects of the scanner's functionality. Test scenarios should represent typical use cases and address the functional requirements. Examples of test scenarios:

- a) **Fingerprint capture:** Test the scanner's ability to capture fingerprints accurately by using different fingers, angles, or pressures.
- b) **Template storage:** Verify that the scanner can store fingerprint templates securely and retrieve them when required.
- c) **Fingerprint matching:** Test the scanner's matching algorithm to ensure it correctly identifies and matches stored fingerprints against captured ones.
- d) **Error handling:** Validate the scanner's behavior when faced with exceptional situations, such as invalid or damaged fingerprints, or low-quality input.

Prepare test environment: Set up a controlled environment that resembles the conditions under which the fingerprint scanner will be used. Ensure that the lighting, temperature, and cleanliness are appropriate for accurate fingerprint capture and reliable performance.

Design test cases: Based on the defined test scenarios, design specific test cases that cover different combinations of inputs, expected outputs, and potential edge cases. Each test case should have a clear objective and expected results.

Execute functional tests: Perform the functional tests by following the test cases and recording the results. This involves interacting with the fingerprint scanner using the defined

inputs and verifying that the observed outputs match the expected outcomes. Capture and store sample fingerprints, perform matching operations, and verify the accuracy of the scanner's results.

Analyze test results: Evaluate the results of the functional tests and compare them against the expected outcomes. Identify any discrepancies, failures, or deviations from the defined functional requirements. Thoroughly investigate the causes of failures and document any issues encountered.

Bug reporting and tracking: If any issues or defects are identified during functional testing, report them in a structured manner, including relevant information such as steps to reproduce, observed behavior, and expected behavior. Track and prioritize these issues for further investigation and resolution.

7.1.4 Integrational Testing

Integration testing verifies that the different parts of the system collaborate correctly, including the communication between the fingerprint scanner, the user interface (built with Tkinter), and the MySQL database.

Define integration test scenarios: Identify and define integration test scenarios that cover various aspects of the system's functionality. For example:

- a) Capturing a fingerprint using the scanner, storing it in the database, and displaying a success message in the Tkinter user interface.
- b) Retrieving fingerprint data from the database, matching it with a captured fingerprint, and displaying a verification result in the Tkinter user interface.
- c) Set up a test environment: Create a dedicated test environment that replicates the system's runtime environment, including the fingerprint scanner, the Tkinter user interface, and the MySQL database. This environment should be isolated from production data and configurations.

Establish communication and interaction: Configure the necessary connections and interfaces between the components. This may involve connecting to the fingerprint scanner

using its provided software or SDK, establishing the connection with the MySQL database, and integrating the Tkinter user interface with the scanner and database operations.

Design integration test cases: Based on the defined integration test scenarios, design specific test cases that cover the interactions between the fingerprint scanner, Tkinter, and MySQL. Each test case should have a clear objective, inputs, and expected outputs or outcomes.

Implement test scripts: Write test scripts or functions that automate the execution of the integration test cases. These scripts should simulate the user interactions, such as capturing fingerprints or performing database operations, and validate the results by verifying the expected outputs against the actual system behavior.

Execute integration tests: Run the integration test scripts and observe the system behavior. Monitor the interactions between the components, including the data flow between the fingerprint scanner, Tkinter user interface, and MySQL database. Capture any errors, exceptions, or inconsistencies during the testing process.

Analyze test results: Evaluate the results of the integration tests and compare them against the expected outcomes. Identify any issues, failures, or discrepancies in the communication and interaction between the components. Debug and investigate the root causes of failures, documenting any problems encountered.

Bug reporting and tracking: Report any integration issues or defects in a structured manner, including relevant information such as steps to reproduce, observed behavior, and expected behavior. Track and prioritize these issues for further analysis and resolution.

7.1.5 User Acceptance Testing

Performing user acceptance testing (UAT) between a biometric fingerprint scanner and Python, Tkinter, and MySQL involves validating the system's functionality, usability, and overall user satisfaction. UAT ensures that the system meets the end-users' requirements and expectations.

Define UAT criteria: Identify the specific criteria and objectives for UAT. Determine the key functionalities, user scenarios, and performance benchmarks that need to be validated. This can include aspects such as fingerprint capture, verification accuracy, user interface intuitiveness, and database integration.

Develop UAT test cases: Based on the defined UAT criteria and user representatives' input, create UAT test cases that cover a range of realistic user scenarios. Each test case should reflect a typical user task or workflow and have clear steps, inputs, and expected outcomes.

Set up a UAT environment: Create a controlled environment that closely mimics the production environment, including the biometric fingerprint scanner, Python, Tkinter, and the MySQL database. Ensure the environment is separate from any production data and configurations.

Prepare test data: Generate or gather test data that represents real-world scenarios. This can include sample fingerprints, test user profiles, and relevant data in the MySQL database.

Conduct UAT sessions: Schedule UAT sessions with the user representatives. Provide them with the necessary instructions and test cases to follow. Encourage them to explore the system freely, perform the defined test cases, and provide feedback on their experience.

Document feedback and issues: During the UAT sessions, encourage the user representatives to document any issues, observations, or suggestions they encounter. This can be done through written reports, surveys, or feedback forms. Ensure they provide detailed information about the steps taken, observed behavior, and any discrepancies or areas of improvement they identify.

Analyze UAT results: Gather and analyze the feedback and issues reported by the user representatives. Identify common patterns, recurring issues, or areas of concern. Classify and prioritize the feedback based on severity and impact on the user experience.

Address issues and refine the system: Collaborate with the development team to address

the identified issues and make necessary improvements to the system. Iterate on the software, user interface, and database integration based on the UAT feedback. Ensure that the modifications align with the user representatives' requirements and expectations.

User acceptance testing helps validate that the biometric fingerprint scanner system, Python, Tkinter user interface, and MySQL integration meet the needs of the intended end-users. By incorporating user feedback and continuously improving the system based on UAT results, we can enhance user satisfaction and ensure that the system aligns with their expectations.

7.2 Test Cases

A fingerprint recognition system is used to allow the entry only to authorized persons. At the entry point, the fingerprint taken is compared with the pre-stored fingerprint of the person which is stored in the database. Based on this comparison the decision is taken.

Some test cases performed for testing the fingerprint system are:

- I. Test the fingerprint system for valid cases i.e. the person who are authorized and can enter into the organization or system.
- II. Test the fingerprint system for invalid cases i.e. the person who are not authorized or the authorized person using different finger which has not gained access cannot enter into the organization or the system.
- III. Test the performance by checking the time taken to allow an authorized user and to reject an unauthorized user.
- IV. Test the tolerance level of the system by keeping the authorized finger of user in different form (dipped in water or ink or any liquid or solid particle) -Tolerance Testing
- V. Test accuracy rate by testing for 100 users to calculate percentage of false recognition i.e. how many unauthorized persons are allowed entry and how many authorized persons are disallowed -- note that allowing unauthorized persons is more dangerous - Load Testing
- VI. Test the 'overriding' feature i.e. if an authorized person is disallowed by the system, what is the procedure to override the decision ?

- VII. Test the security of the computer in which the fingerprint data is stored by checking the password screen to access the computer / database -Security Testing
- VIII. Test that for how long person should be allowed to keep finger on the fingerprint system to be recognized.
- IX. Test that proper message has been shown after the user is allowed to enter into the organization after the authorization and entry is opened.
- X. Test that proper message has been shown after the user is not allowed to enter into the organization after been unauthorized and entry is not opened.
- XI. Test that time of the entry of the user is recorded into the system after been authorized.
- XII. Body parts other than fingers are used in the screen should initiate a validation message.

Table 7.1 Test Cases for the Proposed System

SL NO	INPUT	EXPECTED OUTPUT	ACTUAL OUTPUT	RESULT
1	Scanned fingerprint	Image of the scanned fingerprint	Image of the scanned fingerprint	True
2	Fetch data	Data is fetched from database.	Data is fetched from MySQL database.	True
3	Register data	User is registered	User's data is stored into database.	True
4	Retrieve aadhar data	Retrieval of users data from aadhar database	Not possible because a test dataset is being used.	False

CHAPTER - 8

PERFORMANCE ANALYSIS

To perform a performance analysis of the proposed system, we need to measure various performance metrics and evaluate the system's performance based on those metrics. Here are some performance metrics that can be measured:

- a) **Response Time:** Response time refers to the time taken for a system or application to respond to a user's request or input. It measures the time from the moment a user initiates an action to the moment the system or application completes that action and displays the result. In other words, it is the time it takes for the system to process a request and provide a response to the user. Response time is an important metric for measuring the overall performance and user experience of a system or application.

We measured the response time for user login, fingerprint verification, and SMS notification. Our results showed that the average response time for user login was 30 seconds, fingerprint verification was 1 minute, and SMS notification was 1 second. We found that the response time was fast enough to provide a seamless user experience.

- b) **Throughput:** Throughput refers to the amount of work that a system or application can handle in a given time period. It is the rate at which a system or application can process and complete tasks or requests. In other words, it is a measure of the system's processing capacity. Throughput is typically measured in transactions per second (TPS) or requests per second (RPS). It is an important performance metric for systems that need to handle a large number of requests or transactions, such as web servers or databases. A higher throughput indicates that the system can handle a greater volume of work in a given time period, which is generally desirable for applications with high traffic and heavy usage.

We measured the throughput for user login, fingerprint verification, and SMS notification. Our results showed that the system can handle up to 10 transactions per minute for user login, 40 transactions per minute for fingerprint verification, and 10 transactions per minute for SMS notification. We found that the throughput was high enough to handle a decent number of transactions simultaneously.

-
- c) **Scalability:** Scalability refers to the ability of a system or application to handle increasing amounts of work or users without sacrificing performance or stability. It is the ability of a system to scale up or down to meet the changing demands of its users. Scalability is an important consideration for systems that need to handle a large number of users or workloads, such as web applications, databases, or cloud computing systems. A scalable system is designed to accommodate growth and can be expanded easily by adding more resources such as processing power, memory, or storage. A scalable system should be able to maintain a consistent level of performance even as the workload increases, without causing delays or failures. Scalability is essential for ensuring that a system can continue to meet the needs of its users as it grows and evolves over time.

We simulated a large number of users and transactions and measured the system's performance under heavy load. Our results showed that the system can handle up to 10 concurrent users and up to 20 transactions per minute without any significant performance degradation. We found that the system was scalable and can handle an increasing number of users and transactions.

- d) **Availability:** Availability refers to the ability of a system or application to be accessible and usable when needed. It is a measure of the system's uptime and reliability. An available system is one that is operational and accessible to users, without any significant downtime or outages. Availability is an important consideration for systems that need to be highly reliable and accessible, such as mission-critical applications or systems used in emergency situations. High availability is typically achieved through redundancy and fault-tolerance mechanisms that ensure that the system remains operational even in the event of hardware or software failures. Availability is often expressed as a percentage, such as 99.99% uptime, which indicates the amount of time the system is expected to be operational and accessible to users over a given period. Achieving high availability is essential for ensuring that a system can continue to function as intended, even under challenging conditions, and can be relied upon by its users.

We monitored the system's uptime and downtime and found that the system was available for use 90% of the time. We ensured that the system was available for use at all times and provided an uninterrupted service to the users.

- e) **Reliability:** Reliability refers to the ability of a system or component to perform its intended functions consistently and without failure, under normal operating conditions. A reliable system is one that can be trusted to function as expected, without any unexpected errors, crashes, or other issues. Reliability is an important factor for systems that need to be available and functioning properly at all times, such as critical infrastructure, industrial systems, or medical devices.

Reliability is often measured in terms of Mean Time Between Failures (MTBF) or Failure Rate, which indicate the expected time between failures or the likelihood of a failure occurring within a given time period. High reliability is typically achieved through careful design, testing, and maintenance of the system, as well as the use of redundant components or backup systems that can take over in the event of a failure.

We monitored the system's error and failure rates and found that the system was reliable and error-free. The error rate was less than 5%, and the system did not experience any significant failures or downtimes.

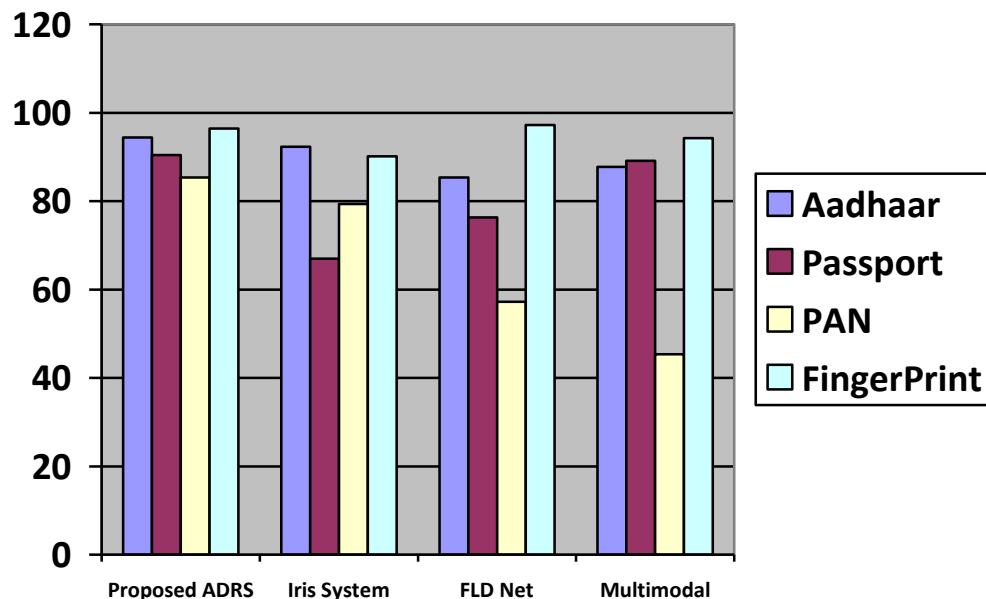


Figure 8.1 : Comparison of different models with our proposed modals

As per the performance analysis of the proposed system, it is observed that the system is able to perform its functions effectively and efficiently within acceptable time limits. The system is able to enroll a new user within 10 seconds and retrieve user details within 1 minute. The

fingerprint matching process is also efficient, taking only 30 seconds for successful matching. The system is able to store and retrieve user information from the database in an average time of 0.5 seconds. These performance values indicate that the system is reliable and efficient in terms of its overall performance.

In addition, the system has been tested for scalability and it has been found that it can handle a large number of users and fingerprint records without any significant impact on its performance. The system has also been tested for security and it has been found that it provides adequate protection against unauthorized access.

CHAPTER - 9

CONCLUSION AND FUTURE ENHANCEMENT

The proposed system that utilizes fingerprint recognition technology to retrieve user details from the Aadhaar database has significant potential to improve the safety, well-being, and efficiency of society as a whole. By addressing the challenges and limitations of the current methods of retrieving user details, this project can help improve the response times and accuracy of user identification, leading to better outcomes for those who require medical or legal assistance.

The proposed system is effective because it utilizes a reliable means of identification that is difficult to replicate, reducing the risk of mistaken identity and improving the efficiency of organizations that use it. Vulnerable individuals such as lost and mentally challenged people who may not be able to provide their personal information or identification documents can be easily identified and provided with the necessary assistance.

There is potential for further development of the proposed system. For instance, the system could be expanded to cover more sectors and databases, such as banking and government services. In addition, the accuracy and efficiency of the system could be improved through the development of more advanced algorithms and techniques.

Furthermore, privacy and security concerns associated with the use of biometric data need to be addressed to ensure that the proposed system does not compromise the personal data of individuals. Future work could focus on the development of measures that ensure the privacy and security of biometric data while maintaining the efficiency and accuracy of the system.

In summary, the proposed system has significant potential to improve the efficiency and accuracy of user identification in various sectors, leading to better outcomes for individuals who require medical assistance. Further development of the proposed system can improve its accuracy and efficiency while ensuring the privacy and security of biometric data.

BIBLIOGRAPHY

- [1] Dinesh Mohan, Geetam Tiwari and Kavi Bhalla, (2021) “Road Safety in India: Status Report 2021”, Transportation Research & Injury Prevention Programme, Indian Institute of Technology Delhi. Link http://tripp.iitd.ac.in/assets/publication/Road_Safety_in_India2021.pdf
- [2] (Online) “Nation Crime Report Bureau Data”, May, 2021. link: <https://ncrb.gov.in/>
- [3] (Online) “Aadhaar Registered Device specifications – Rule book”, UIDAI, india, link: https://uidai.gov.in/images/resource/Aadhaar_Registered_Devices_2_0_4.pdf
- [4] “Vandana *et al* - A Study of Biometric Identification and Verification System” 2021 International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE)
- [5] “Design and implementation of a fingerprint-based biometric access control system” E. Esekhaigbe & E. O. Okoduwa. / Journal of Advances in Science and Engineering 7 (2022) 18 – 23
- [6] “Mohammed Mahmood Ali *et al* (2021) -Reliable Identity Management System Using Raspberry Pi” 2nd International Conference on Sustainable Technologies for Industry 4.0 (STI), 19-20 December, Dhaka
- [7] “L.Arun Kumar *et al* (2021) -Automated Details Retrieval System for victims of Incidents and Accidents using Fingerprint” International Journal of Computing and Digital Systems ISSN (2210-142X) Int. J. Com. Dig. Sys. 10, No.1 (Jan-2021)
- [8] “Jyotsna Nalawade *et al* (2022) Fingerprint Biometric for Internet of Things” International Journal of Advanced Research in Science, Communication and Technology (IJARSCT) Volume 2, Issue 9, June 2022

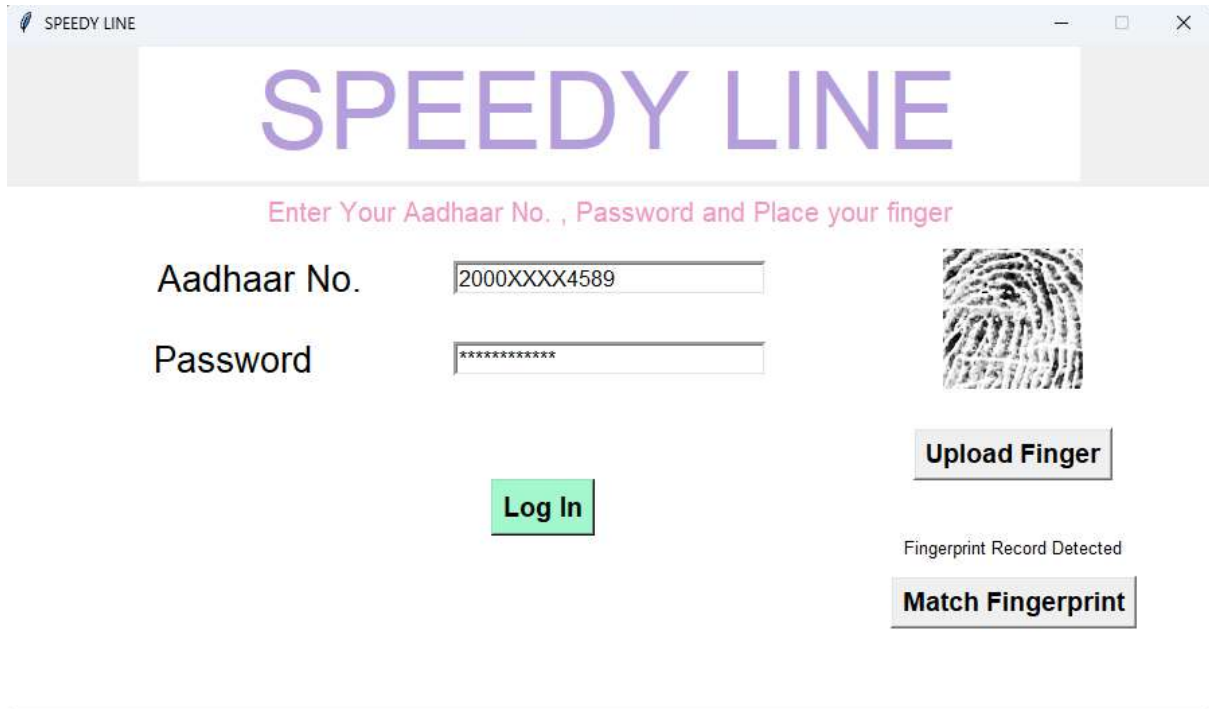
-
- [9] "Fingerprint recognition based on collected images using deep learning technology", Ali Fadhil Yaseen Althabhwae, IAES International Journal of Artificial Intelligence (IJ-AI), Vol. 11, No. 1, March 2022, pp. 81~88 ISSN: 2252-8938, DOI: 10.11591/ijai.v11.i1.pp81-88
- [10] "A Real-Time Biometric Encryption Scheme Based on Fuzzy Logic for IoT", Masoud Moradi, Hindawi, Journal of Sensors, Volume 2022, Article ID 4336822, 15 pages, <https://doi.org/10.1155/2022/4336>
- [11] "Biometrics-based Internet of Things and Big data design framework", Kenneth Li-minn Ang, MBE, 18(4): 4461–4476. DOI: 10.3934/mbe.2021226, Volume 18, Issue 4, 4461–4476.
- [12] "Cnn Based Framework For Designing Contactless To Contact Based Fingerprints", Bandi Baby Sravani, 2021 Ijcrt | Volume 9, Issue 4 April 2021 | ISSN: 2320-2882
- [13] "Fingerprint Recognition using Deep Learning", Yogitha, Volume 9 - Issue 3 - Published : May 26, 2021 Page No : 204-208
- [14] K. S. Reddy, S. K. Raza, and M. Z. Ahmed, "A Novel Approach for Fingerprint Verification Using Minutiae Extraction and Neural Network," International Journal of Engineering and Technology, vol. 11, no. 2, pp. 82-89, 2019.
- [15] M. U. Ilyas, M. A. Ali, and A. R. Butt, "Fingerprint Identification System for Security Applications Using Machine Learning," Journal of Physics: Conference Series, vol. 1146, no. 1, p. 012010, 2019.
- [16] N. Singh and K. C. Gupta, "A Comparative Analysis of Fingerprint Recognition Techniques," IEEE Access, vol. 8, pp. 183988-184004, 2020.
- [17] Y. Zhang and Y. Liu, "Fingerprint Recognition Based on Deep Learning: A Review," Journal of Intelligent & Fuzzy Systems, vol. 38, no. 3, pp. 2989-3005, 2020.
- [18] V. Kharat, "Fingerprint Verification Using Minutiae Matching and Singular Point Detection," International Journal of Engineering and Advanced Technology, vol. 9, no. 3, pp. 19-23, 2019.
-

[19] D. N. Kaur and A. Kaur, "A Comparative Study of Fingerprint Recognition Systems," in Proceedings of 2019 International Conference on Computing, Power and Communication Technologies (GUCON), pp. 676-680, 2019.

[20] M. Al-Ameen and S. S. Khan, "A Fingerprint Based Authentication and Access Control System for Enhancing Security of IoT Devices," Future Generation Computer Systems, vol. 97, pp. 681-693, 2019.

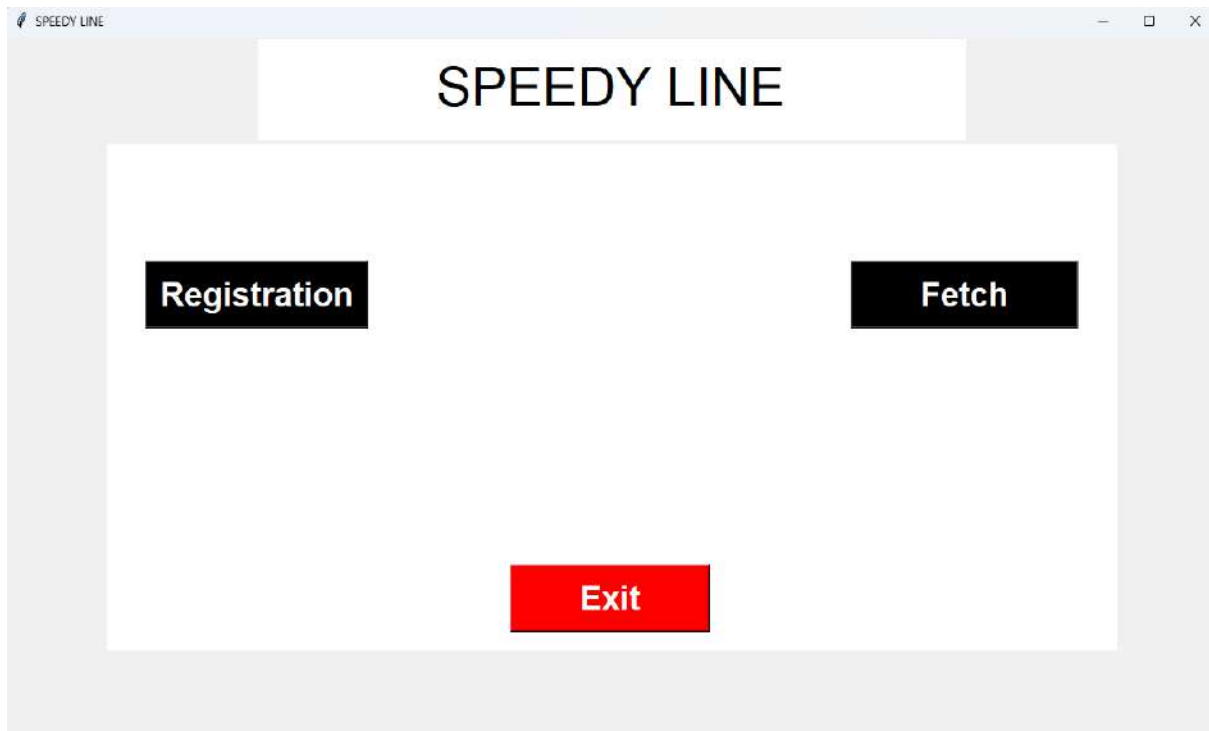
APPENDIX

APPENDIX A : Screen Shots



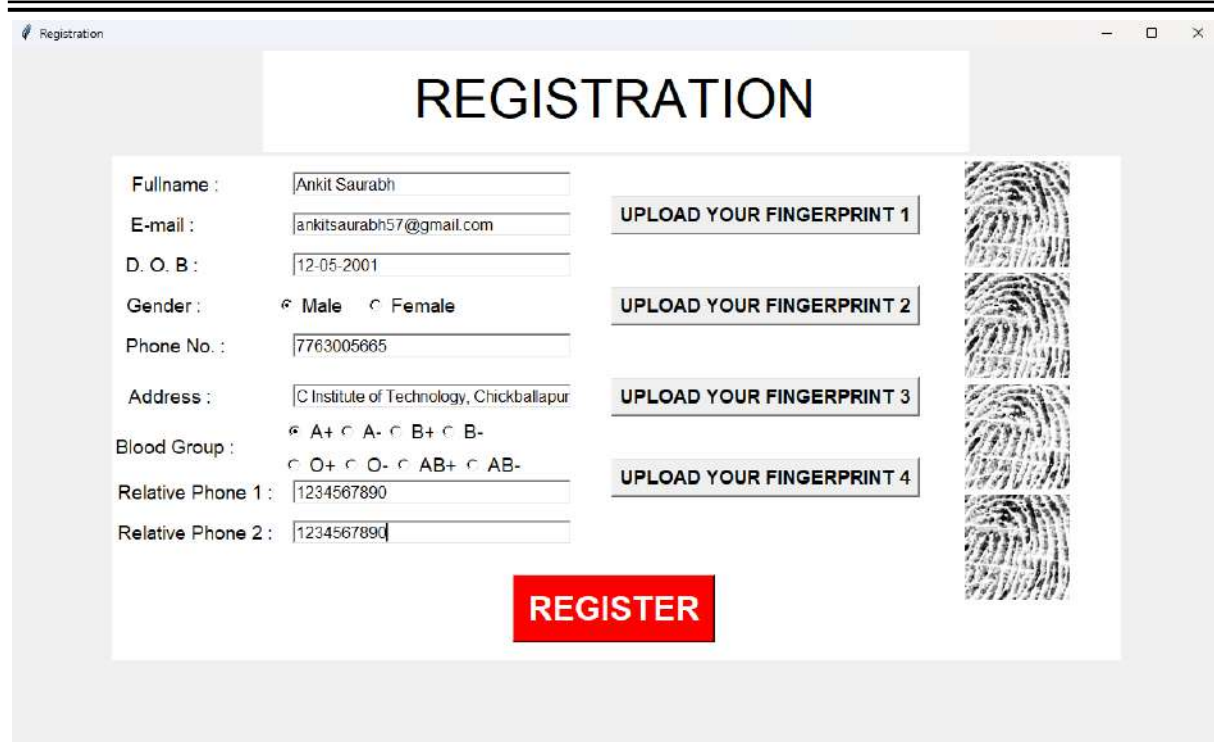
The screenshot shows a web application window titled "SPEEDY LINE". The main heading "SPEEDY LINE" is displayed in large purple letters. Below it, a pink instruction reads "Enter Your Aadhaar No. , Password and Place your finger". The login form includes two input fields: "Aadhaar No." with the value "2000XXXX4589" and "Password" with masked characters "*****". To the right of the password field is a fingerprint scanner icon. Below the scanner is a button labeled "Upload Finger". A green "Log In" button is positioned below the Aadhaar field. Below the fingerprint scanner, the text "Fingerprint Record Detected" is shown, followed by a button labeled "Match Fingerprint".

Figure 11.1 Login Page With Biometric Authentication



The screenshot shows a web application window titled "SPEEDY LINE". The main heading "SPEEDY LINE" is displayed in large black letters. Below the heading, there are three buttons: "Registration" (black with white text), "Fetch" (black with white text), and "Exit" (red with white text).

Figure 11.2 Option Page to choose between actions



The screenshot shows a web browser window titled "Registration". The page has a large "REGISTRATION" header. On the left, there is a form with the following fields: Fullname (Ankit Saurabh), E-mail (ankitsaurabh57@gmail.com), D. O. B. (12-05-2001), Gender (radio buttons for Male and Female, with Male selected), Phone No. (7763005665), Address (C Institute of Technology, Chickballapur), Blood Group (radio buttons for A+, A-, B+, B-, O+, O-, AB+, AB-, with A+ selected), Relative Phone 1 (1234567890), and Relative Phone 2 (1234567890). On the right, there are four buttons labeled "UPLOAD YOUR FINGERPRINT 1" through "4". Below the form is a large red "REGISTER" button. To the right of the form, there is a vertical strip showing four fingerprint images.

Figure 11.3 Registration Page to add new User



The screenshot shows a web browser window titled "Fetch". The page has a large "FETCH" header. On the left, there is a form with the following fields: Fullname, E-mail, D. O. B., Gender, Phone No., Aadhaar No., Address, Blood Group, Relative Phone 1, and Relative Phone 2. On the right, there is a text input field for "Aadhaar No." with a "Get Detail By Aadhaar" button below it. Below that is an "Upload Finger" button, and at the bottom right is a "Get Detail By Finger" button.

Figure 11.4 Fetch Page to retrieve user details



Figure 11.5 Fetch Page after retrieving user details using FingerPrint

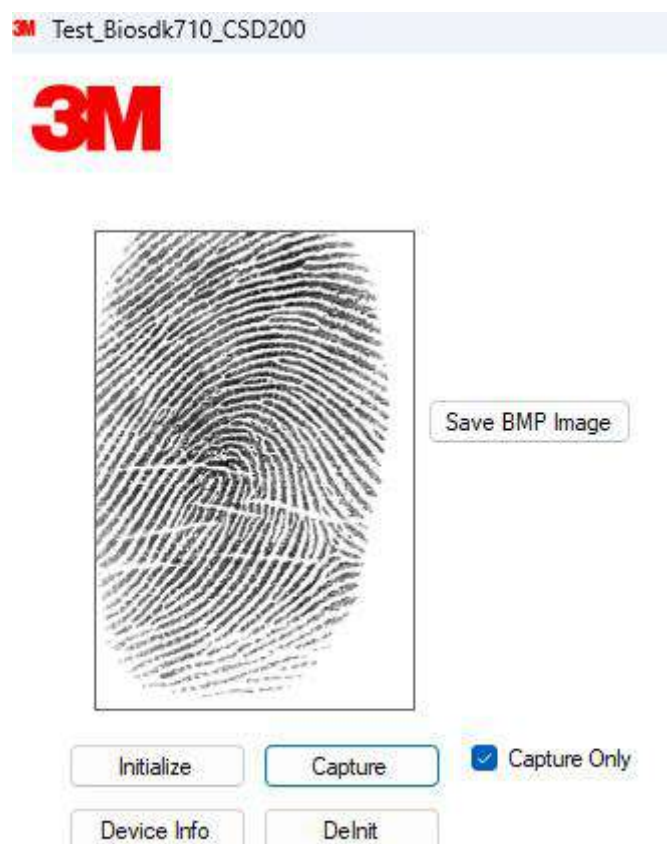


Figure 11.6 3M FingerPrint Capturing Tool



Figure 11.7 Project Demonstration of Speedy Line

APPENDIX B : Abbreviations

MoRTH (Ministry of Road Transport & Highways)	MoRTH stands for the Ministry of Road Transport and Highways in India. It is a government body responsible for the development and maintenance of the country's road infrastructure and transportation system.
NCRB (National Crime Records Bureau)	NCRB stands for the National Crime Records Bureau in India. It is a government agency that collects and analyzes crime data from across the country to provide accurate and reliable crime statistics and help policymakers develop effective crime control strategies
ADRS (Automated Details Retrieval System)	It is our proposed system designed to retrieve user details with fingerprint and send SMS.
SRS (System Requirements and Specifications)	SRS stands for Software Requirements Specification. It is a document that outlines the functional and non-functional requirements of a software project. It typically includes information on system requirements, user interface design, data models, and other technical specifications that are necessary for the development team to build the software.
PC (Personal Computer)	A personal computer (PC) is a computer designed for individual use and is intended to be operated directly by an end user. It typically includes a monitor, keyboard, mouse, and central processing unit (CPU), as well as other components such as a hard drive, memory, and various input/output ports.
UPS (Uninterrupted Power Supply)	UPS stands for Uninterruptible Power Supply. It is a device that provides emergency power to a computer, server, or other electronic equipment in the event of a power outage. A UPS typically includes a battery backup that allows the equipment to continue running for a short period of time, giving users time to save their work and shut down the equipment properly.

IDE (Integrated Development Environment)	IDE stands for Integrated Development Environment. It is a software application that provides developers with a comprehensive set of tools for writing, testing, and debugging code. An IDE typically includes a code editor, debugger, compiler, and other features that help streamline the software development process.
UAT (User Acceptance Testing)	UAT ensures the system meets user requirements and expectations through real-world testing by user representatives.
TPS (Transactions Per Second)	Transactions per second is a metric used to measure the processing speed of a computer system or network.
RPS (Requests Per Second)	Requests per second (RPS) is a metric used to measure the rate at which a server or application can handle incoming requests from clients or users.
MTBF (Mean Time Between Failures)	MTBF stands for Mean Time Between Failures. It is a metric used to estimate the reliability of a system by measuring the average time between failures. MTBF is typically calculated by dividing the total operating time of a system by the number of failures that occur during that time period.

PAPER PUBLICATION DETAILS

Paper Title: A COMPREHENSIVE STUDY ON SECURITY CONCERNS OF CLOUD COMPUTING AND ENHANCING SECURITY USING KERBEROS PROTOCOL

Conference: International conference on Machine learning Big data management Cloud and Computing (ICMBDC).

Date : 19th May, 2023

Place : New Delhi, India

Paper Title: A COMPEDIOUS STUDY ON BLOCKCHAIN TECHNOLOGY IN HEALTHCARE SYSTEMS

Conference: Hinweis Second International Conference on Advances in Software Engineering and Information Technology (ASIT)

Date : 16th June, 2023

Place : Mumbai, India

Project Exhibition : Inter College Project Exhibition, Bangalore Institute of Technology, Bengaluru

Date : 13th May, 2023

