

# **BIOMETERIC AUTHENTICATION FOR PASSENGERS IN METRO RAILWAY SYSTEM**

**A PROJECT REPORT**

**SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE  
AWARD OF BACHELOR OF TECHNOLOGY**

**IN**

**COMPUTER SCIENCE AND ENGINEERING**

**BY**

**PATAN SANAULLA KHAN 2210313746**

**KRISHNA CHAITANYA 2210313730**

**VUSIRIKAPALLY SOWMYA 2210313761**

**M. RENUKA 2201313735**

**UNDER THE GUIDANCE OF**

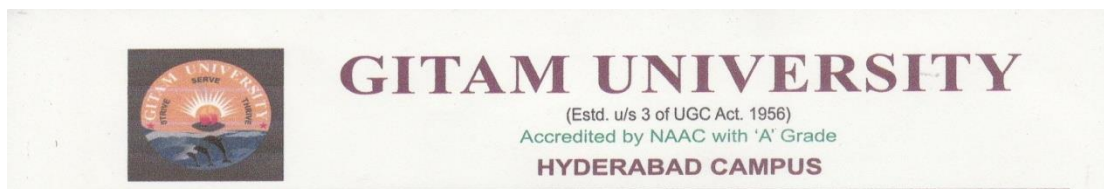
**A.PHANI SHEETAL**

**ASSISITANT PROFESSOR**



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**GITAM SCHOOL OF TECHNOLOGY  
GITAM UNIVERSITY  
HYDERABAD CAMPUS**



## **DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

### **CERTIFICATE**

**This is to certify that the project titled BIOMETRIC AUTHENTICATION FOR PASSENGERS IN METRO RAILWAY SYSTEM was presented satisfactorily in the Department of Computer Science and Engineering , School of Technology, GITAM University, Hyderabad by PATAN SANAULLA KHAN (2210313746), KRISHNA CHAITANYA (2210313730), VUSIRIKAPALLY SOWMYA (2201313761), M. RENUKA (2210313735) in partial fulfillment of requirement for their project work carried out under my guidance and supervision.**

A.Phani Sheetal

Project guide

B.Rajendra Prasad Babu  
Project Coordinator

Dr.S.PhaniKumar  
HOD-CSE

**Name & signature of the project panel member(s).**

- 1.
- 2.
- 3.

## **ACKNOWLEDGEMENT**

The satisfaction that accompanies the successful completion of any task would be incomplete without the mention of people whose ceaseless cooperation made it possible, whose constant guidance and encouragement crown all efforts with success.

We are grateful to our project guide **Mrs. A Phani Sheetal**, Assistant Professor, Department of Computer Science and Engineering, GITAM School of Technology-(HYD) for the guidance, inspiration and constructive suggestions that helped us in the preparation of this project.

We thank our project coordinator **Rajendra Prasad Babu.B**, Assistant Professor for his support and encouragement throughout the course of the project work.

At the outset we thank our Head of the Department **Dr.S. Phani Kumar** and our Honourable Pro-Vice Chancellor **Prof. N. Siva Prasad** for the moral support and the excellent facilities provided. I would also like to thank all the teaching and non-teaching staff members of Computer Science department who have extended their full cooperation during course of my industrial training.

We wish to express our warm and grateful thanks to our project coordinator for the guidance and assistance he provided in completing our project successfully.

I thank all my friends who helped me sharing knowledge and by providing material to complete the project in time.

## **ABSTRACT**

Metro Rails being developed as one of the largest modern transport systems in the world based upon the PPP model, and such projects are very common in major metropolitan cities. Unfortunately, with traditional membership cards and travel documents being easily forged and manipulated, collecting reliable, truthful, and relevant information about individuals travelling is difficult. Recognizing this important need, designing a biometric identification system to provide the ability for law enforcement and security personnel to quickly and accurately retrieve important information about individuals in the metro using biometrics is a key solution.

Biometric authentication has been receiving extensive attention over the past decade. Biometrics is to identify individuals using physiological or behavioral characteristics, such as fingerprint, face, iris, retina, palm-print, etc. Among all the biometric techniques, fingerprint recognition is the most popular method. Every individual who wishes to travel through the metro transport needs to provide the fingerprint, enter the details (i.e destination) and canthus enter into the metro station. All the details including the scanned fingerprint are to be saved into a database such that it maintains the records and makes it easier to recognize for later use.

Passenger is considered to take any train to their destination by validating and hence paying the travel money before boarding the metro and hence after reaching the destination the person has to scan their finger only then the person is allowed to leave the metro station. Therefore to minimize the time for the process and also to enhance the security we consider best and the fastest way to implement the biometric system into the application.

# **CONTENTS**

- i. PROJECT TITLE**
- ii. CERTIFICATE**
- iii. ACKNOWLEDGEMENT**
- iv. ABSTRACT**
- v. CONTENTS**
- vi. LIST OF FIGURES AND TABLES**

## **1. CHAPTER 1: INTRODUCTION**

- 1.1 Motivation
- 1.2 Problem Description
  - 1.2.1 Traffic & Transportation scenario of the city
  - 1.2.2 Real Estate Development
  - 1.2.3 Institutional Arrangement
- 1.3 Using Biometric
- 1.4 What is fingerprint?
- 1.5 Why use Fingerprint?
- 1.6 Using Fingerprint Recognition in Metro System
  - 1.6.1 Patterns:
  - 1.6.2 Fingerprint processing
  - 1.6.3 Minutiae features:
- 1.7 Limitation of using Biometric

## **2. CHAPTER 2: SYSTEM ANALYSIS**

### 2.1 Introduction

### 2.2 Existing System

#### 2.2.1 Automatic Fare Collection System

### 2.3 Disadvantages of Existing System

### 2.4 Fingerprint Identification System

### 2.5 How Fingerprint Reorganization works?

## **3. CHAPTER 3: ANALYSIS**

### 3.1 Introduction

#### 3.1.1 To seamlessly develop biometrics (Fingerprint)

### 3.2 Minimum Requirements

#### 3.2.1 Minimum Hardware Requirement

#### 3.2.2 Minimum Software Requirement

### 3.3 MICROSOFT VISUAL STUDIO 2010/2013

#### 3.3.1 Visual Studio 2010

### 3.4 C SHARP (C#)

#### 3.4.1 Features of C#

### 3.4 Structured Query Language (SQL) Server

## **4. CHAPTER 4: DESIGN**

### 4.1 Introduction

### 4.2 Usecase Diagram

4.3 Dataflow Diagram

4.4 Class Diagram

4.5 Activity Diagram

4.6 Sequence Diagram

## **5. CHAPTER 5: TESTING**

5.1 Introduction

5.2 Types of Tests

5.2.1 Unit Testing

5.2.2 Integration Testing

5.2.3 Functional Testing

5.2.4 System Testing

5.2.5 White Box Testing

5.2.6 Black Box Testing

5.3 Unit Testing:

5.4 Integration Testing

5.5 Acceptance Testing

5.6 Validation

## **6. CHAPTER 6: IMPLEMENTATION**

6.1 Introduction

6.2 Launch Page

6.3 Login Page

6.4 Home Screen Form

6.5 Welcome Form

6.6 Registration Form

6.7 Authentication Form

6.8 Validation Success Form

6.9 Passenger Details Form

6.10 Login Table

6.11 Boarding Details

6.12 Passenger Details

## **7. CHAPTER 7: CONCLUSION**

## **8. CHAPTER 8: BIBLOGRAPHY**



## **LIST OF FIGURES:**

Figure 1.1: Example of a ridge ending and a bifurcation

Figure 1.2: Example of a ridge ending, Bifurcation and a short ridge

Figure 2.3: Architectural Diagram of the Proposed System

Figure 2.2: Finger to BITS Conversion

Figure 4.1: USECASE DIAGRAM

Figure 4.2: DATAFLOW DIAGRAM

Figure 6.2: Launch Page

Figure 6.3: Login page

Figure 6.4: Home Screen Page

Figure 6.5: Welcome Page

Figure 6.6: Registration Form

Figure 6.7: Validate or Boarding Form

Figure 6.8: Successfully validating a User

Figure 6.9: Boarding details/ Passenger in train

Figure 6.10.a Login Table

Figure 6.10.b Login Table Data

Figure 6.11.a Boarding Table

Figure 6.11.b Boarding Table Data

Figure 6.12.a Passenger Table

Figure 6.12.b Passenger Table Data

# **CHAPTER 1: INTRODUCTION**

## **1.1 MOTIVATION**

Designing a railway metro passenger system based on fingerprint recognition, which works faster one to many identifications that manages records of passengers in who travel in metro rail and enhances the authentication process for the entry into the metro.

Every organization whether it be an or business organization, it should maintain a proper and secure record of customers for effective functioning of organization.

Image processing and fingerprint recognition are very advanced today in terms of technology. It was our responsibility to improve fingerprint identification system. We decreased matching time by partitioning the database and improve matching using one to many matchings.

## **1.2 PROBLEM DESCRIPTION**

The City Hyderabad, the capital city of the state of Telangana, is one of the fastest growing metros in India. The core area of the city covers about 625 sq. km (under the jurisdiction of the Greater Hyderabad Municipal Corporation). Apart from being the centre for pharma and some other traditional industries, the city has now emerged as a major IT, ITES and other service sector industries hub. With a number of universities, R&D institutions and other centres of higher education, modern corporate hospitals and a world-class new international airport, the city is a thriving urban agglomeration.

The city's strategic geographical location, image as a multilingual cosmopolitan city and the pro-active policies of the state government are making it an attractive destination and a flourishing centre of trade and commerce. The population of the city, which is currently around 8 million, is projected to grow to 13.6 million by 2021.

### 1.2.1 Traffic & Transportation scenario of the city

In the absence of a comfortable and efficient public transportation system, the traffic and transportation scenario of the city is a matter for concern. The existing public transport in the city is mainly the buses run by the State Road Transport Corporation, a public sector undertaking of the state government. Out of about 8 million motorized trips in the city, the share of public transport is only about 44%.

The rapid growth of the city, the rising income levels and the lack of a good public transportation system are resulting in phenomenal increase in private vehicles, causing frequent traffic jams and high pollution levels in the city. Presently there are over 3 million vehicles and about 0.2 million vehicles are getting added every year.

### 1.2.2 Real Estate Development

To enhance the financial viability and bankability of the project, the Concessionaire is allowed to undertake real estate development through commercial exploitation of air space over the lands provided for creation of project facilities such as depots and parking and circulation areas at select stations. The real estate development can be undertaken by the Concessionaire above the ground floor at the three depots at Miyapur (99 acres), Nagole (96 acres) and Falaknuma (17 acres) and above the parking and circulation areas at 25 select stations (57 acres at Category I, Category II and Category III Stations) in accordance with the provisions of the Concession Agreement, Applicable Laws and Good Industry Practice.

The cumulative built up area to be utilized for real estate development is 18.5 million sq. ft. (about 12.5 million sq. ft. over the 3 depots and 6 million sq. ft. at the 25 select stations). The ownership of the land will always remain with Govt. The built-up area can only be used for rentals during the Concession Period and it will have to be handed over to Govt. at the end of the Concession Period in accordance with the provisions of the Concession Agreement.

### 1.2.3 Institutional Arrangement

With a view to providing a single point nodal agency to coordinate with various government, semi-government and private agencies and facilitating implementation of the project by the Concessionaire, Government has established a Special Purpose Vehicle (SPV) in the form of Hyderabad Metro Rail Ltd. (HMR) as it's fully owned undertaking. Consisting of a small but highly motivated group of experienced engineers and senior government officers, HMR has initiated several measures for acquisition of the required lands, identification of obstacles, R&R, obtaining right of way for the three corridors, etc. The Heads of utilities in the city, such as Hyderabad Metro Water Supply & Sewerage Board (HMWSSB) and the Southern Power Distribution Company of Telangana, and other connected senior officers of the government are on the Board of Directors of HMR.

Rail-based 'Mass Rapid Transit System' has been widely accepted as a solution for most of the traffic and environmental pollution related problems which major cities throughout the world are facing now. Metro rail construction activities are being undertaken in a big way in India, existing metro rail network of the city of Kolkata and Delhi are being expanded, while it is under various stages of construction in cities like Bengaluru, Chennai, Mumbai and Hyderabad. In the present article, important environmental and other critical issues have been discussed in the Indian context which are equally relevant in other developing counties.

The infrastructure projects are important for development of a nation and are also a mirror of any country's development. However, most of the infrastructure projects on account of their sheer size and nature (namely, type, site/location, urban settling, etc.) are invariably accompanied by significant environmental and social impacts during different phases (namely, pre-construction, construction and operational phase) of the project. The nature of these impacts could be either positive or negative, depending upon their potential to favorably or adversely affect the surrounding environment and also the resident community.

With increasing traffic demand, coupled with increasing number of vehicles on road, the problems related to traffic congestion, road accidents and environmental pollution have also increased significantly over the last few years in various urban centres around the world. One of the most accepted methods of improving traffic and environmental conditions in these cities has been to provide an efficient public transportation system, so that the private vehicle owners are encouraged to shift to public transportation system (Fouracre et al. 2003).

In case of developing countries like India, the public transportation system in most of the cities is grossly inadequate and can be considered inefficient as well as insufficient. In order to improve the public transportation system, the Mass Rapid Transit System (MRTS) have been provided or being planned in various parts of the world. Although most of the developed countries have already provided MRTS in their major cities, the same is lacking in most of the cities in the developing countries, including India. Ideally, as the population of a city grows share of public transportation (road or rail based) should increase for a city. With a population of 1 or 2 million, the share of public transportation should be about 50 per cent–60 per cent. Moreover the percentage share of public transport should progressively increase with the population growth of the city, and should reach about 75 percent when the population of the city reaches the 5 million mark. Selection of a public transportation system on a corridor in the city, whether it should be road based (High Capacity Bus Systems [HCBS]) or rail-based (for example, metro rail, mono rail, etc.) primarily depends on the traffic density during the peak hour(s) on that particular corridor. Experience from Indian cities have shown that under mixed traffic conditions, comprising slow and fast moving vehicles, road-based public transportation system can optimally carry 8,000 persons per hour per direction (phpdt). When traffic density crosses that mark, traffic and environmental pollution related issues/problems increase; under these circumstances provisions of a rail-based mass transit system (that is, a metro rail system) should be considered.

### **1.3 USING BIOMETRICS**

Biometric verification is any means by which a person can be uniquely identified by evaluating one or more distinguishing biological traits. Unique identifiers include fingerprints, hand geometry, earlobe geometry, retina and iris patterns, voice waves, DNA, and signatures. The oldest form of biometric verification is fingerprinting. Historians have found examples of thumbprints being used as a means of unique identification on clay seals in ancient China. Biometric verification has advanced considerably with the advent of computerized databases and the digitization of analog data, allowing for almost instantaneous personal identification.

Iris-pattern and retina-pattern authentication methods are already employed in some bank automatic teller machines. Voice waveform recognition, a method of verification that has been used for many years with tape recordings in telephone wiretaps, is now being used for access to proprietary databanks in research facilities. Facial-recognition technology has been used by law enforcement to pick out individuals in large crowds with considerable reliability. Hand geometry is being used in industry to provide physical access to buildings. Earlobe geometry has been used to disprove the identity of individuals who claim to be someone they are not (identity theft). Signature comparison is not as reliable, all by itself, as the other biometric verification methods but offers an extra layer of verification when used in conjunction with one or more other methods.

No matter what biometric methodology is used, the identification verification process remains the same. A record of a person's unique characteristic is captured and kept in a database. Later on, when identification verification is required, a new record is captured and compared with the previous record in the database. If the data in the new record matches that in the database record, the person's identity is confirmed.

Biometric Identification Systems are widely used for unique identification of humans mainly for verification and identification. Biometrics is used as a form of identity access management and access control. So use of biometrics in any management system is a secure approach. There are many types of biometric systems like fingerprint recognition,

face recognition, voice recognition, iris recognition, palm recognition etc. In this project, we used fingerprint recognition system.

## **1.4 WHAT IS FINGERPRINT?**

A **fingerprint** in its narrow sense is an impression left by the friction ridges of a human finger. Fingerprints are easily deposited on suitable surfaces (such as glass or metal or polished stone) by the natural secretions of sweat from the eccrine glands that are present in epidermal ridges. These are sometimes referred to as "Chanced Impressions".

In a wider use of the term, fingerprints are the traces of an impression from the friction ridges of any part of a human or other primate hand. A print from the sole of the foot can also leave an impression of friction ridges. Deliberate impressions of fingerprints may be formed by ink or other substances transferred from the peaks of friction ridges on the skin to a relatively smooth surface such as a fingerprint card. Fingerprint records normally contain impressions from the pad on the last joint of fingers and thumbs, although fingerprint cards also typically record portions of lower joint areas of the fingers.

Human fingerprints are detailed, nearly unique, difficult to alter, and durable over the life of an individual, making them suitable as long-term markers of human identity. They may be employed by police or other authorities to identify individuals who wish to conceal their identity, or to identify people who are incapacitated or deceased and thus unable to identify themselves, as in the aftermath of a natural disaster. Fingerprint analysis, in use since the early 20th century, has led to many crimes being solved. This means that many criminals consider gloves essential.

A fingerprint is the pattern of ridges and valleys on the surface of a fingertip. The endpoints and crossing points of ridges are called minutiae. It is a widely accepted assumption that the minutiae pattern of each finger is unique and does not change during

one's life. Ridge endings are the points where the ridge curve terminates, and bifurcations are where a ridge splits from a single path to two paths at a Y-junction.

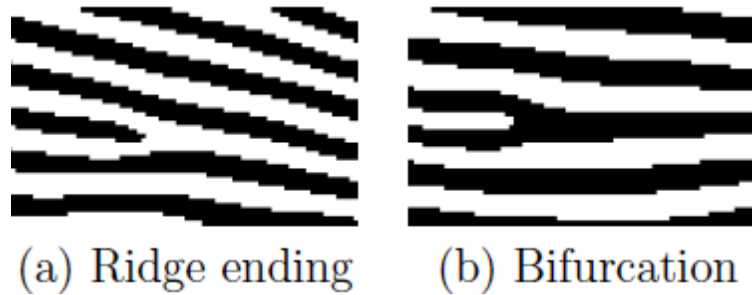


Figure 1.1: Example of a ridge ending and a bifurcation

When human fingerprint experts determine if two fingerprints are from the same finger, the matching degree between two minutiae pattern is one of the most important factors. Thanks to the similarity to the way of human fingerprint experts and compactness of templates, the minutiae-based matching method is the most widely studied matching method.

When human fingerprint experts determine if two fingerprints are from the same finger, the matching degree between two minutiae pattern is one of the most important factors. Thanks to the similarity to the way of human fingerprint experts and compactness of templates, the minutiae-based matching method is the most widely studied matching method.

## 1.5 WHY USE FINGERPRINTS?

Fingerprints are considered to be the best and fastest method for biometric identification. They are secure to use, unique for every person and do not change in one's lifetime. Besides these, implementation of fingerprint recognition system is cheap, easy and accurate up to satisfiability. Fingerprint recognition has been widely used in both forensic and civilian applications. Compared with other biometrics features, fingerprint-based biometrics is the most proven technique and has the largest market shares. Not only it is faster than other techniques but also the energy consumption by such systems is too less.



## **1.6 USING FINGERPRINT RECOGNITION IN METRO SYSTEM**

The analysis of fingerprints for matching purposes generally requires the comparison of several features of the print pattern. These include patterns, which are aggregate characteristics of ridges, and minutia points, which are unique features found within the patterns. It is also necessary to know the structure and properties of human skin in order to successfully employ some of the imaging technologies.

### **1.6.1 Patterns**

The three basic patterns of fingerprint ridges are the arch, loop, and whorl:

1. Arch: The ridges enter from one side of the finger, rise in the center forming an arc, and then exit the other side of the finger.
2. Loop: The ridges enter from one side of a finger, form a curve, and then exit on that same side.
3. Whorl: Ridges form circularly around a central point on the finger. Scientists have found that family members often share the same general fingerprint patterns, leading to the belief that these patterns are inherited.

### **1.6.2 Fingerprint processing**

Fingerprint processing has three primary functions: enrollment, searching and verification. Among these functions, enrollment which captures fingerprint image from the sensor plays an important role. A reason is that the way people put their fingerprints on a mirror to scan can affect to the result in the searching and verifying process. Regarding to verification function, there are several techniques to match fingerprints such as correlation-based matching, minutiae-based matching, ridge feature-based matching and minutiae-based algorithm. However, the most popular algorithm was minutiae based matching algorithm due to its efficiency and accuracy.

### 1.6.3 Minutiae features:

The major minutiae features of fingerprint ridges are ridge ending, bifurcation, and short ridge (or dot). The ridge ending is the point at which a ridge terminates. Bifurcations are points at which a single ridge splits into two ridges. Short ridges (or dots) are ridges which are significantly shorter than the average ridge length on the fingerprint. Minutiae and patterns are very important in the analysis of fingerprints since no two fingers have been shown to be identical.

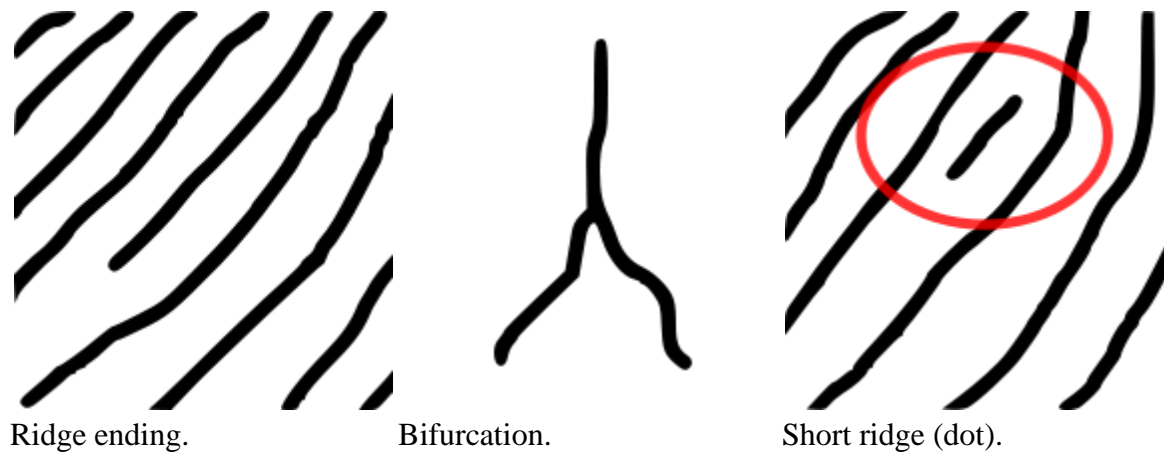


Figure 1.2: Example of a ridge ending, Bifurcation and a short ridge

To make all the passengers related work automatic and online, we plan to design an metro rail system which could be implemented and runs mainly on finger prints. It uses a fingerprint identification system developed in this project. This fingerprint identification system uses existing as well as new techniques in fingerprint recognition and matching. A new one to many matching algorithm for large databases has been introduced in this identification system.

## **1.7 LIMIMTATIONS OF USING BIOMETRIC**

Each one of the Technologies used in our days bring us a manner to restrict the access to a system, allowing the entrance only to those persons who know a specific code, own a card or have determined physic marks. The more complex is the system, the most difficult is to be attacked, although it will be more expensive and will require more software and hardware resources. When a new authentication system is implanted, it is essential a judgement between simplicity, price and efficiency, as well as social acceptability.

The password method is the cheapest and simplest technology, because it only requires elementary software resources. On the other hand, this system is easily attackable, since he is quite simple to obtain the data from a person, either extracting the information to the person itself using deceits, or attacking the software of the system. For example, it can be easily installed in the computer, a program that simulates the “user name and password” window, so that when the user introduces his data in that window, that will be collected by the “Spy” program. Immediately after this, it appears the true window, identical, and the user will simply believe that he has been mistaken. So, this method, despite being usually used, for example, to access banking accounts, is not at all the most indicated if we want a safe system, and in a short-time future is tried to be changed by most immune methods.

The Smart Cards are very useful since they can be easily combined with other authentication systems, serving as storage system. Self-containment of smart card makes it resistant to attack as it does not need to depend upon potentially vulnerable external resources. But its small size and bend requirements (which are designed to protect the card physically), limits the memory and processing resources. And used like the only identification system, is not excessively trustworthy, since it can be easily stolen, lost or simply forgotten at home. Besides, sometimes they are combined with cryptography methods, which makes them more difficult (more expensive) to implement.

The advantage that Biometrics presents is that the information is unique for each individual and that it can identify the individual despite variations in the time (it does not matter if the first biometric sample was taken year ago). The pillars of e-learning security are: authentication, privacy (data confidentiality) authorization (access control), data integrity and non-repudiation. Biometric is a technique that can provide all these requirements with quite lot reliability.

Although biometrics is considered the most effective and safe method (is very difficult to falsify), we have to bear in mind its disadvantages, for example, that since it is a relative new technology, it is not still integrated in PC, so IT departments need to make a conscious decision before making the purchase and change its structure.

We also have to consider the advantages and disadvantages of fingerprint biometric system. we will make an enumeration of the problems that these techniques can present:

#### **1.7.1 Disadvantages:**

- For some people it is very intrusive, because is still related to criminal identification.
- It can make mistakes with the dryness or dirty of the finger's skin, as well as with the age (is not appropriate with children, because the size of their fingerprint changes quickly).
- Image captured at 500 dots per inch (dpi). Resolution: 8 bits per pixel. A 500 dpi fingerprint image at 8 bits per pixel demands a large memory space, 240 Kbytes approximately → Compression required (a factor of 10 approximately).

## **CHAPTER 2: SYSTEM ANALYSIS**

### **2.1 INTRODUCTION**

The present system involves traditional membership cards and travel documents being easily forged and manipulated by the passengers. Hence collecting reliable, truthful, and relevant information about individuals travelling is also difficult. And the existing system does not provide the ability for law enforcement and security personnel quickly and accurately. The difficulty to make reports and retrieve information of the passengers were in the passenger details are not received.

Hence the existing system is time consuming and also prone to crime. Thus, fingerprint based system is more efficient and useful for the present generation.

### **2.2 EXISTING SYSTEM**

Based on many Traffic and Transportation studies conducted by various agencies, the then Government of Andhra Pradesh (GoAP) approved development of Hyderabad Metro Rail (MRTS) project in three high density traffic corridors of the city spanning across 72 km in phase-I. Detailed Project Reports (DPRs), Traffic Survey Reports, and other related reports were prepared by Delhi Metro Rail Corporation (DMRC) for the project.

The three corridors spanning a length of approximately 72 km to be taken up in phase-I are as under.

- Corridor I: Miyapur – LB Nagar : 29 kms; 27 stations.
- Corridor II: JBS – Falaknuma : 15 kms; 16 stations.
- Corridor III: Nagole – Shilparamam : 28 kms; 23 stations.

The Metro Rail system under construction is a completely elevated system, generally run in the central median of the road. The viaduct structure for the elevated system is a box girder carrying two tracks on a single pier located on the median of the road.

The system is designed to cater to 50,000 PHPDT for Corridors I and III and 35,000 PHPDT for Corridor II. It will have to be built, operated, maintained and transferred at the end of the Concession Period as per the provisions of the Concession Agreement. The performance specifications and safety standards are enunciated in the Manual of Specifications & Standards (MSS), a well deliberated and published document of GoAP, based on "output oriented" performance philosophy. The miss provides scope for enough design flexibility and innovation, and it forms part of the Concession Agreement.

#### Features of Metro Railway System:

- It is an elevated metro rail, with two tracks (up and down lines) on a deck erected on pillars generally in the central median of the road, without obstructing the road traffic.
- Stations are located at an average interval of 1KM – elevated stations with passenger access through staircases, escalators and lifts.
- Adequate parking space and circulating areas are being provided for multi modal integration at the stations.
- With a frequency of 3 to 5 minutes during peak hours, the system is expected to carry about 17 lakh passengers per day by 2017 and 22 lakh by 2024.
- With a maximum speed of 80 kmph, the average speed of the trains will be 34 kmph – an international standard for MRT systems.
- The travel time by metro rail from one end to another is:
- 45 minutes for Corridor I (Miyapur-L.B.Nagar – 29Kms) as against 1 hr 46 minutes by bus.
- 22 minutes for Corridor II (Jubilee Bus Station-Falaknuma-15 Kms) as against 1 hr 10 minutes by bus.

- and 39 minutes for Corridor III (Nagole-Shilparmam-28 Kms) as against 1 hr 26 minutes by bus.
- Rails will be continuously welded to minimize noise levels.
- Signaling system ensures safety and specified speeds through Automatic Train Control (ATC), and Automatic Train Protection (ATP).
- Telecommunication facilities will be state of the art, facilitating continuous communication between Central Control, train drivers and station masters.
- Good inter-modal integration will be provided at all the rail terminals, bus stations, and the MMTS (existing joint venture of GoT and Railways) stations.
- All stations will have air conditioned "Merry go round" mini-bus services, connecting nearby colonies, business establishments and other popular places.
- Different categories of aesthetic stations are being designed to reflect the local architecture, latest trends, and to avoid/ minimise demolitions. As far as possible, parking and circulation areas will be developed on nearby Govt. lands.
- Eco-friendly Travel Mode – it will reduce air and noise pollution in the city.
- Smart card-based Automatic ticketing & gate systems for passenger convenience and seamless travel.
- Standard gauge track (1435 mm) to allow sharper curves and gradients.

### **1.2.1 AUTOMATIC FARE COLLECTION SYSTEM:**

The ticketing solution for Metro Rail is a state of the art Automated Fare Collection System based on Contactless Fare Media Technology.

The fare collection system divides the Metro Stations in two areas namely 'paid' and 'un-paid' areas. The paid and unpaid areas are separated by automatic entry / exit gates and fencing. It ensures only person having valid ticket enters the system. The system is designed as a closed one with ticket checking at entry and exit, hence ensuring 100% ticket check in the system.

#### **AFC System Key Features:**

- State of the art Automatic Fare Collection System
- Technologies best suited for MRTS and recognized world-over

- Web application to top-up the contactless smart card through internet
- System accepts banknotes, coins, bank cards (credit/debit) and smart cards as a payment mechanism
- TVMs facilitated with Coin recyclers and Bank Note recyclers
- Integrated Parking Solution for future implementation

#### **AFC System Key Sub-systems:**

The system comprises of following key sub-systems:

- Automatic Gates
- Ticket Vending Machines
- Add Value Machines
- Ticket Office Machines
- Ticket Readers
- Portable Ticket Analyzers

#### **Fare Media & Fare Products:**

The tickets are of two types: Contactless Smart Card (CSC) and Contactless Smart Token (CST). Contactless Tokens are pre-paid fare media for journey between fixed set of stations. Token can be purchased for single trip or return trip and are more suitable fare media option for non-frequent users.

Contactless Smart Cards are stored value cards and are available in the form of either monetary (e-Purse) or non-monetary (Pass) or combination of both with various options to attract the regular and frequent users. Money Value equal to fare between any two stations is deducted from the stored value at the exit Gate.

#### **Few of the proposed fare products are as listed below:**

- Electronic/Stored Value Purse (e-Purse)
- Tourist Pass
- Trip Pass (with the options like 10/30/50, etc.)
- Daily Pass
- Weekly Pass
- Monthly Pass
- Holiday Pass



### **General Operation:**

The fare media either can be procured from the Ticket Offices or Ticket Vending Machines. Patron then proceeds to automatic Entry Gate Arrays and present the fare media to the contactless reader writer located on Automatic Gate. If the authentication is valid the gate will permit the patron pass through the gate to enter from unpaid area to paid area, else gates will restrict the entry.

In case of exit, patron using the contactless token will deposit his/her token at Exit Gate and passenger using a contactless smart card will present the card to contactless reader writer on the Automatic Gate. If all the conditions for a proper exit are met, flaps of the Automatic Gate will be opened to allow access to the passenger.

### **2.3 DISADVANTAGES OF THE EXISTING SYSTEM:**

The present existing system promotes the use of travel cards or ticket tokens, which have multiple disadvantages. They are:

- Cards or tokens can be lost while travelling
- Forging or manipulating of the cards is an easier compared to unique personal fingerprint
- Manufacturing of new cards is quite a longer time period
- Secure information of the personnel is not stored in the databases
- Difficult to make reports, like of the people travelling in the metro to a location
- Requires special scanners to detect the various kinds of tokens and travel passes.

### **2.4 FINGERPRINT IDENTIFICATION SYSTEM:**

An identification system is one which helps in identifying an individual among many people when detailed information is not available. It may involve matching available features of candidate like fingerprints with those already enrolled in database. The project requires a fingerprint reader/scanner for finger detection.

Every passenger can login to the system through finger detection. The fingerprint of the passenger is compared with the one stored in database and if it matches then the person is marked for that particular passenger. The system also generates a brief report of the database according to destination-wise or person-wise as required. A defaulter list can be generated through system. Admin has the option to take a print of the reports and defaulter list thus generated.

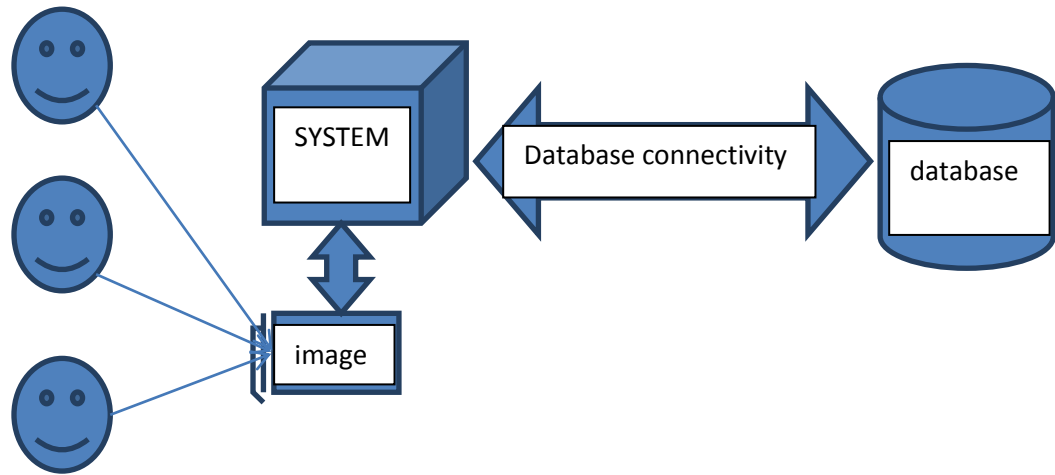


Figure 2.3: Architectural Diagram of the Proposed System

## 2.5 HOW FINGERPRINT RECOGNIZATION WORKS?

Fingerprint images that are found or scanned are not of optimum quality. So were move noises and enhance their quality. We extract features like minutiae and others for matching. If the sets of minutiae are matched with those in the database, we call it an identified fingerprint. After matching, we perform post-matching steps which may include showing details of identified passenger, marking his details etc. A brief flow chart is shown in next section.

Biometrics are automated methods of recognizing a person based on a physiological or behavioral characteristic. Among the features measured are face, fingerprints, hand geometry, handwriting, iris, retinal, vein, and voice. Biometric data are separate and distinct from personal information. Biometric templates cannot be reverse-engineered to

recreate personal information and they cannot be stolen and used to access personal information.

Using a unique, physical attribute of your body, such as your fingerprint or iris, to effortlessly identify and verify that you are who you claim to be, is the best and easiest solution in the market today. That is the simple truth and power of Biometrics Technology today. Although biometric technology has been around for many years, modern advances in this emerging technology, coupled with big reductions in cost, now make biometrics readily available and affordable to consumers, small business owner, larger corporations and public sector agencies alike.

A fingerprint scanner system has two basic jobs, it needs to get an image of your finger, and it needs to determine whether the pattern of ridges and valleys in this image matches the pattern of ridges and valleys in pre-scanned images.

Only specific characteristics, which are unique to every fingerprint, are filtered and saved as an encrypted biometric key or mathematical representation. No image of a fingerprint is ever saved, only a series of numbers (a binary code), which is used for verification.

The algorithm cannot be reconverted to an image, so no one can duplicate your fingerprints.

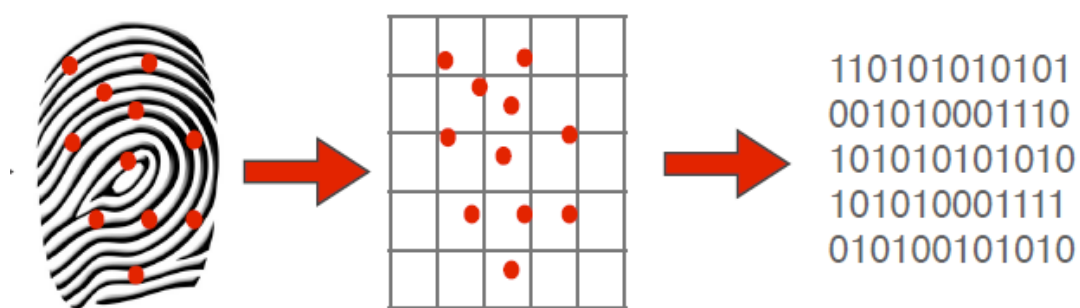


Figure 2.2: Finger to BITS Conversion

## CHAPTER 3: ANALYSIS

### 3.1 INTRODUCTION

As the technology behind biometrics become cheaper and more reliable, many companies have begun to integrate various biometrics into their existing security system. The following information will explain how to implement and build biometric technology to augment current security systems while explaining specific issues that need to be addressed.

Designed to benefit both technical and non-technical professionals, this real world information will enable developers to develop biometric solutions without compromising the intended security enhancement.

#### 3.1.1 To seamlessly develop biometrics (Fingerprint):

##### a) Template storage and management issues

Which consists for the storage of the databases with the existing and new data records. Databases to store the information regarding the different users of the application.

##### b) Template encryption issues

Making the data to be encrypted and decrypted such that it could be accessed and be checked for accuracy with the existing records.

##### c) Security and integrity of biometric data from source to output

Maintaining the security and integrity by a login form such that the user and the authorized admin can access the information and make updates on the data of the users.

##### d) Export restrictions regarding certain biometric implementations

Maintaining a restriction on the size of the images and data or content to be taken as input from the user. Hence the user need to follow the standards while uploading the content over the application.

e) Common UI (User Interface) issues regarding biometrics

The user interface that provide smooth communication with the user to share the user related biometric data to the application. Thus the user interface needs to be user friendly and easy to use.

f) Client/Server programming issues to consider

Making sure the server that stores the databases of the information which related to the biometric content is always connected over a strong programming without any loop holes. Thus, debug for any issues related to the server programming.

### **3.2 MINIMUM REQUIREMENTS:**

The following is the basic requirement needed to build the basic biometric scanner for the Metro Railway System.

#### **3.2.1 Minimum Hardware Requirement:**

- i3 Based Processor
- 256 MB-RAM
- 80 GB hard disk
- Monitor

#### **3.2.2 Minimum Software Requirement:**

- Visual Studio 2010 ( C# )
- SQL Server
- Window 7 or higher

### 3.3 MICROSOFT VISUAL STUDIO 2010/2013

**Microsoft Visual Studio** is an integrated development environment (IDE) from Microsoft. It is used to develop computer programs for Microsoft Windows, as well as web sites, web apps, web services and mobile apps. Visual Studio uses Microsoft software development platforms such as Windows API, Windows Forms, Windows Presentation Foundation, Windows Store and Microsoft Silverlight. It can produce both native code and managed code.

Visual Studio includes a code editor supporting IntelliSense (the code completion component) as well as code refactoring. The integrated debugger works both as a source-level debugger and a machine-level debugger. Other built-in tools include a code profiler, forms designer for building GUI applications, web designer, class designer, and database schema designer. It accepts plug-ins that enhance the functionality at almost every level—including adding support for source control systems (like Subversion) and adding new toolsets like editors and visual designers for domain-specific languages or toolsets for other aspects of the software development lifecycle (like the Team Foundation Server client: Team Explorer).

Visual Studio supports different programming languages and allows the code editor and debugger to support (to varying degrees) nearly any programming language, provided a language-specific service exists. Built-in languages include C, C++ and C++/CLI (via Visual C++), VB.NET (via Visual Basic .NET), C# (via Visual C#), F# (as of Visual Studio 2010) and TypeScript (as of Visual Studio 2013 Update 2). Support for other languages such as Python, Ruby, Node.js, and M among others is available via language services installed separately. It also supports XML/XSLT, HTML/XHTML, JavaScript and CSS. Java (and J#) were supported in the past.

### 3.3.1 VISUAL STUDIO 2010:

The Visual Studio 2010 IDE was redesigned which, according to Microsoft, clears the UI organization and "reduces clutter and complexity". The new IDE better supports multiple document windows and floating tool windows, while offering better multi-monitor support. The IDE shell has been rewritten using the Windows Presentation Foundation (WPF), whereas the internals have been redesigned using Managed Extensibility Framework (MEF) that offers more extensibility points than previous versions of the IDE that enabled add-ins to modify the behavior of the IDE.

Visual Studio 2010 comes with .NET Framework 4 and supports developing applications targeting Windows 7. It supports IBM DB2 and Oracle databases, in addition to Microsoft SQL Server. It has integrated support for developing Microsoft Silverlight applications, including an interactive designer. Visual Studio 2010 offers several tools to make parallel programming simpler: in addition to the Parallel Extensions for the .NET Framework and the Parallel Patterns Library for native code, Visual Studio 2010 includes tools for debugging parallel applications. The new tools allow the visualization of parallel Tasks and their runtime stacks.<sup>[104]</sup> Tools for profiling parallel applications can be used for visualization of thread wait-times and thread migrations across processor cores. Intel and Microsoft have jointly pledged support for a new Concurrency Runtime in Visual Studio 2010 and Intel has launched parallelism support in Parallel Studio as an add-on for Visual Studio.

The Visual Studio 2010 code editor now highlights references; whenever a symbol is selected, all other usages of the symbol are highlighted. It also offers a *Quick Search* feature to incrementally search across all symbols in C++, C# and VB.NET projects. Quick Search supports substring matches and camelCase searches. The *Call Hierarchy* feature allows the developer to see all the methods that are called from a current method as well as the methods that call the current one. IntelliSense in Visual Studio supports a *consume-first* mode which developers can opt into. In this mode, IntelliSense does not auto-complete identifiers; this allows the developer to use undefined identifiers (like variable or method names) and define those later. Visual Studio 2010 can also help in this by automatically defining them, if it can infer their types from

usage. Current versions of Visual Studio have a known bug which makes IntelliSense unusable for projects using pure C (not C++).

Visual Studio 2010 features a new Help System replacing the MSDN Library viewer. The Help System is no longer based on Microsoft Help 2 and does not use Microsoft Document Explorer. Dynamic help containing links to related help topics based on where the developer was in the IDE has been removed in the shipping product, but can be added back using a download from Microsoft. Microsoft Visual Studio 2013 is a freeware to let the computer programmers to develop software and it is provided by the Microsoft. It allows the programmers to develop applications and websites in platforms such as Visual Basic, Visual C#, Visual C++ etc. The Express Editions is more suitable for novice developers.

### **3.4 C SHARP (C#)**

C# (pronounced as *see sharp*) is a multi-paradigm programming language encompassing strong, typing, imperative, declarative, functional, generic, object-oriented (class-based), and component-oriented programming disciplines. It was developed by Microsoft within its .NET initiative and later approved as a standard by Ecma (ECMA-334) and ISO (ISO/IEC 23270:2006). C# is one of the programming languages designed for the Common Language Infrastructure.

C# is a general-purpose, object-oriented programming language. Its development team is led by Anders Hejlsberg. The most recent version is C# 7.0 which was released in 2017 along with Visual Studio 2017

During the development of the .NET Framework, the class libraries were originally written using a managed code compiler system called *Simple Managed C* (SMC). In January 1999, Anders Hejlsberg formed a team to build a new language at the time called Cool, which stood for "C-like Object Oriented Language". Microsoft had considered keeping the name "Cool" as the final name of the language, but chose not to do so for trademark reasons. By the time the .NET project was publicly announced at the July



2000 Professional Developers Conference, the language had been renamed C#, and the class libraries and ASP.NET runtime had been ported to C#.

C#'s principal designer and lead architect at Microsoft is Anders Hejlsberg, who was previously involved with the design of Turbo Pascal, Embarcadero Delphi (formerly CodeGear Delphi, Inprise Delphi and Borland Delphi), and Visual J++. In interviews and technical papers he has stated that flaws in most major programming languages (e.g. C++, Java, Delphi, and Smalltalk) drove the fundamentals of the Common Language Runtime (CLR), which, in turn, drove the design of the C# language itself.

### **3.4.1 Features of C#:**

#### **a) Portability**

By design, C# is the programming language that most directly reflects the underlying Common Language Infrastructure (CLI). Most of its intrinsic types correspond to value-types implemented by the CLI framework. However, the language specification does not state the code generation requirements of the compiler: that is, it does not state that a C# compiler must target a Common Language Runtime, or generate Common Intermediate Language (CIL), or generate any other specific format. Theoretically, a C# compiler could generate machine code like traditional compilers of C++ or Fortran.

#### **b) Typing**

C# supports strongly typed implicit variable declarations with the keyword `var`, and implicitly typed arrays with the keyword `new[]` followed by a collection initializer.

C# supports a strict Boolean data type, `bool`. Statements that take conditions, such as `while` and `if`, require an expression of a type that implements the `true` operator, such as the Boolean type. While C++ also has a Boolean type, C# disallows this "integer meaning true or false" approach, on the grounds that forcing programmers to use expressions that return exactly `bool` can prevent certain types of programming mistakes

such as `if (a = b)` (use of assignment `=` instead of equality `==`, which while not an error in C or C++, will be caught by the compiler anyway).

C# is more type safe than C++. The only implicit conversions by default are those that are considered safe, such as widening of integers. This is enforced at compile-time, during JIT, and, in some cases, at runtime. No implicit conversions occur between Booleans and integers, nor between enumeration members and integers (except for literal 0, which can be implicitly converted to any enumerated type). Any user-defined conversion must be explicitly marked as `explicit` or `implicit`, unlike C++ copy constructors and conversion operators, which are both implicit by default.

C# has explicit support for covariance and contravariance in generic types, unlike C++ which has some degree of support for contravariance simply through the semantics of return types on virtual methods.

The C# language does not allow for global variables or functions. All methods and members must be declared within classes. Static members of public classes can substitute for global variables and functions.

### **c) Meta programming**

Meta programming via C# attributes is part of the language. Many of these attributes duplicate the functionality of GCC's and VisualC++'s platform-dependent preprocessor directives.

### **d) eMethods and functions**

Like C++, and unlike Java, C# programmers must use the keyword `virtual` to allow methods to be overridden by subclasses.

*Extension methods* in C# allow programmers to use static methods as if they were methods from a class's method table, allowing programmers to add methods to an object that they feel should exist on that object and its derivatives.

The type `dynamic` allows for run-time method binding, allowing for JavaScript-like method calls and run-time object composition.

C# has support for strongly-typed function pointers via the keyword `delegate`. Like the Qt framework's pseudo-C++ *signal* and *slot*, C# has semantics specifically surrounding publish-subscribe style events, though C# uses delegates to do so.

#### **e) Property**

C# provides properties as syntactic sugar for a common pattern in which a pair of methods, accessor (getter) and mutator (setter) encapsulate operations on a single attribute of a class. No redundant method signatures for the getter/setter implementations need be written, and the property may be accessed using attribute syntax rather than more verbose method calls.

#### **f) Namespace**

A C# `namespace` provides the same level of code isolation as a Java `package` or a C++ `namespace`, with very similar rules and features to a `package`.

#### **g) Memory access**

In C#, memory address pointers can only be used within blocks specifically marked as *unsafe*, and programs with unsafe code need appropriate permissions to run. Most object access is done through safe object references, which always either point to a "live" object or have the well-defined null value; it is impossible to obtain a reference to a "dead" object (one that has been garbage collected), or to a random block of memory. An unsafe pointer can point to an instance of a value-type, array, string, or a block of memory allocated on a stack. Code that is not marked as unsafe can still store and manipulate pointers through the `System.IntPtr` type, but it cannot dereference them.

Managed memory cannot be explicitly freed; instead, it is automatically garbage collected. Garbage collection addresses the problem of memory leaks by freeing the programmer of responsibility for releasing memory that is no longer needed.

#### **h) Exception**

Checked exceptions are not present in C# (in contrast to Java). This has been a conscious decision based on the issues of scalability and versionability.

### **i) Polymorphism**

Unlike C++, C# does not support multiple inheritance, although a class can implement any number of interfaces. This was a design decision by the language's lead architect to avoid complication and simplify architectural requirements throughout CLI. When implementing multiple interfaces that contain a method with the same signature, C# allows implementing each method depending on which interface that method is being called through, or, like Java, allows implementing the method once, and have that be the one invocation on a call through any of the class's interfaces.

However, unlike Java, C# supports operator overloading. Only the most commonly overloaded operators in C++ may be overloaded in C#.

### **j) Functional programming**

Though primarily an imperative language, C# 2.0 offered limited support for functional programming through first-class functions and closures in the form of anonymous delegates. C# 3.0 expanded support for functional programming with the introduction of a lightweight syntax for lambda expressions, extension methods (an affordance for modules), and a list comprehension syntax in the form of a "query comprehension" language.

## **3.4 STRUCTURED QUERY LANGUAGE (SQL) SERVER**

SQL Server is a local database storage provided by the Microsoft along with the Microsoft Visual Studio. It is provided to let the computer programmers to manage and store information while develop the applications and websites. SQL Server appears to be a Relational Database Management System.

**Microsoft SQL Server** is a relational database management system developed by Microsoft. As a database server, it is a software product with the primary function of storing and retrieving data as requested by other software applications—which may run

either on the same computer or on another computer across a network (including the Internet).

Microsoft markets at least a dozen different editions of Microsoft SQL Server, aimed at different audiences and for workloads ranging from small single-machine applications to large Internet-facing applications with many concurrent users.

Data storage is a database, which is a collection of tables with typed columns. SQL Server supports different data types, including primary types such as *Integer*, *Float*, *Decimal*, *Char* (including character strings), *Varchar* (variable length character strings), binary (for unstructured blobs of data), *Text* (for textual data) among others.

Microsoft SQL Server also allows user-defined composite types (UDTs) to be defined and used. It also makes server statistics available as virtual tables and views (called Dynamic Management Views or DMVs). In addition to tables, a database can also contain other objects including views, stored procedures, indexes and constraints, along with a transaction log. A SQL Server database can contain a maximum of  $2^{31}$  objects, and can span multiple OS-level files with a maximum file size of  $2^{60}$  bytes (1 exabyte). The data in the database are stored in primary data files with an extension *.mdf*. Secondary data files, identified with a *.ndf* extension, are used to allow the data of a single database to be spread across more than one file, and optionally across more than one file system. Log files are identified with the *.ldf* extension.

Storage space allocated to a database is divided into sequentially numbered *pages*, each 8 KB in size. A *page* is the basic unit of I/O for SQL Server operations. A page is marked with a 96-byte header which stores metadata about the page including the page number, page type, free space on the page and the ID of the object that owns it. Page type defines the data contained in the page: data stored in the database, index, allocation map which holds information about how pages are allocated to tables and indexes, change map which holds information about the changes made to other pages since last backup or logging, or contain large data types such as image or text. While page is the basic unit of an I/O operation, space is actually managed in terms of an *extent* which consists of 8

pages. A database object can either span all 8 pages in an extent ("uniform extent") or share an extent with up to 7 more objects ("mixed extent"). A row in a database table cannot span more than one page, so is limited to 8 KB in size. However, if the data exceeds 8 KB and the row contains *varchar* or *varbinary* data, the data in those columns are moved to a new page (or possibly a sequence of pages, called an *allocation unit*) and replaced with a pointer to the data.

For physical storage of a table, its rows are divided into a series of partitions (numbered 1 to n). The partition size is user defined; by default all rows are in a single partition. A table is split into multiple partitions in order to spread a database over a computer cluster. Rows in each partition are stored in either B-tree or heap structure. If the table has an associated, clustered index to allow fast retrieval of rows, the rows are stored in-order according to their index values, with a B-tree providing the index. The data is in the leaf node of the leaves, and other nodes storing the index values for the leaf data reachable from the respective nodes. If the index is non-clustered, the rows are not sorted according to the index keys. An indexed view has the same storage structure as an indexed table. A table without a clustered index is stored in an unordered heap structure. However, the table may have non-clustered indices to allow fast retrieval of rows. In some situations the heap structure has performance advantages over the clustered structure. Both heaps and B-trees can span multiple allocation units.

## CHAPTER 4: DESIGN

### 4.1 INTRODUCTION

UML stands for Unified Modeling Language. UML is a standardized general-purpose modeling language in the field of object-oriented software engineering. The goal is for UML to become a common language for creating models of object oriented computer software. In its current form UML is comprised of two major components: A Meta-model and a notation. In the future, some form of method or process may also be added to, or associated with, UML. The Unified Modeling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business modeling and other non-software systems. The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems. The UML uses mostly graphical notations to express the design of software projects.

### 4.2. USECASE DIAGRAM

A use-case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.

### 4.3 DATAFLOW DIAGRAM

A **data flow diagram (DFD)** is a graphical representation of the "flow" of data through an information system, modeling its *process* aspects. A DFD is often used as a preliminary step to create an overview of the system without going into great detail, which can later be elaborated. DFDs can also be used for the visualization of data

processing (structured design). A DFD shows what kind of information will be input to and output from the system, how the data will advance through the system, and where the data will be stored. It does not show information about the timing of process or information about whether processes will operate in sequence or in parallel unlike a flowchart which also shows this information

#### **4.4 CLASS DIAGRAM:**

Class diagram are widely used to describe the types the types of objects in a system and their relationships. Class diagrams model class structure and contents using design elements such as classes, packages and objects. Class diagrams describe three different perspectives when designing a system, conceptual, specification, and implementation. Their perspectives when designing a system, conceptual, specification and implementation .There perspectives become evident as the diagram is created and help solidify the design. C lasses are composed of three things: a name, attributes, and operations.

#### **4.5 ACTIVITY DIAGRAMS:**

Activity diagram describe the workflow behavior of a system. Activity diagrams are Similar to state diagrams because activities are the state of doing something. The diagrams describe the state of activities by showing sequence of activities performed. Activity diagram can show activities that are conditional or parallel. Activity diagrams should be used in conjunction with other modeling techniques such as interaction diagram, state diagrams. The main reason to use activity diagram is to model the work flow behind the system being designed.

#### **4.6 SEQUENCE DIAGRAM:**

The sequence diagram is used primarily to show the interaction between objects in the sequential order that those interaction occur. Much like the class diagram, developers typically think sequence diagrams were meant exclusively for them. However, an organization business staff can find sequence diagrams useful to communicate how the business currently works by showing how various business objects interact.



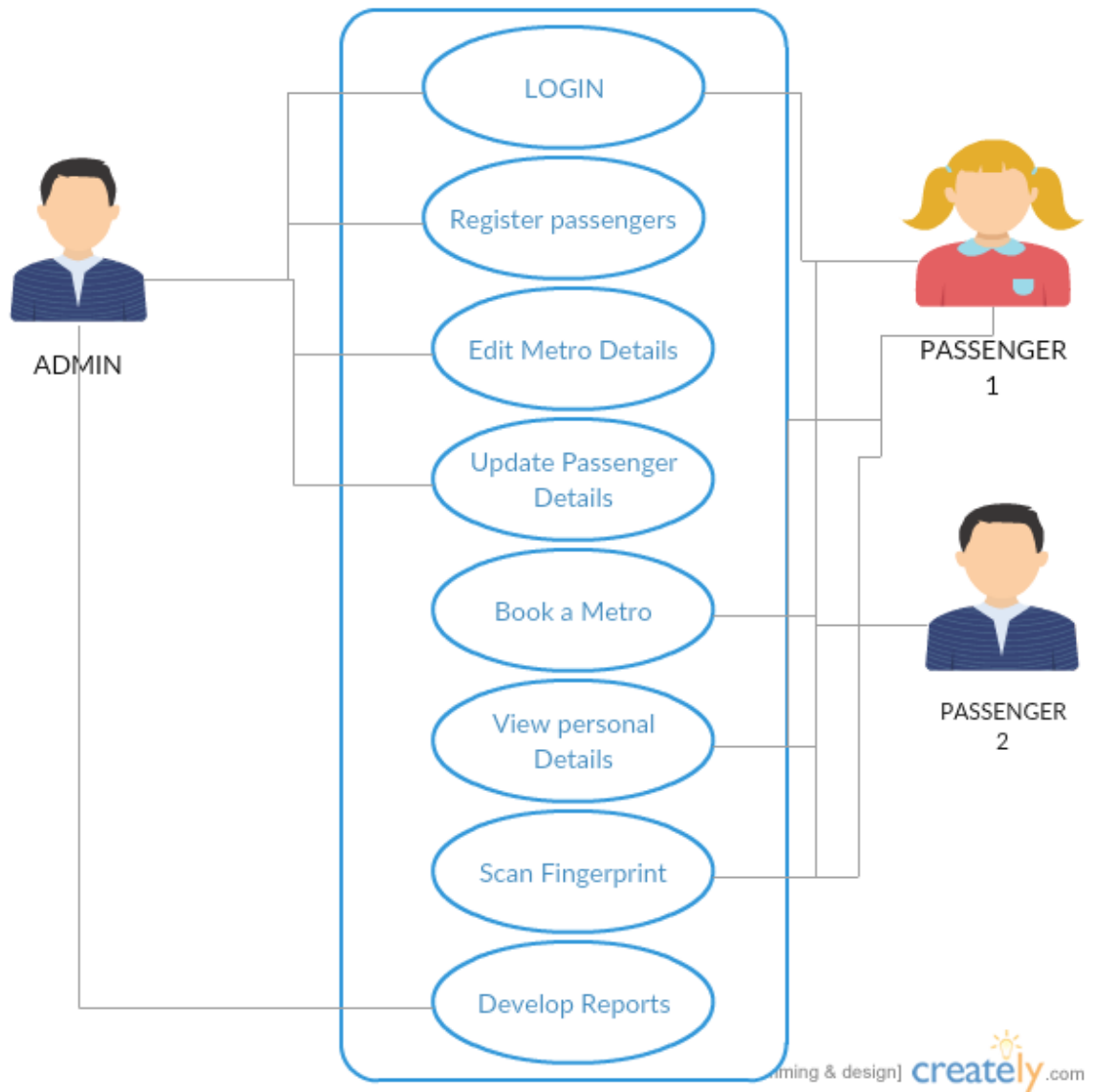


Figure 4.1:USECASE DIAGRAM

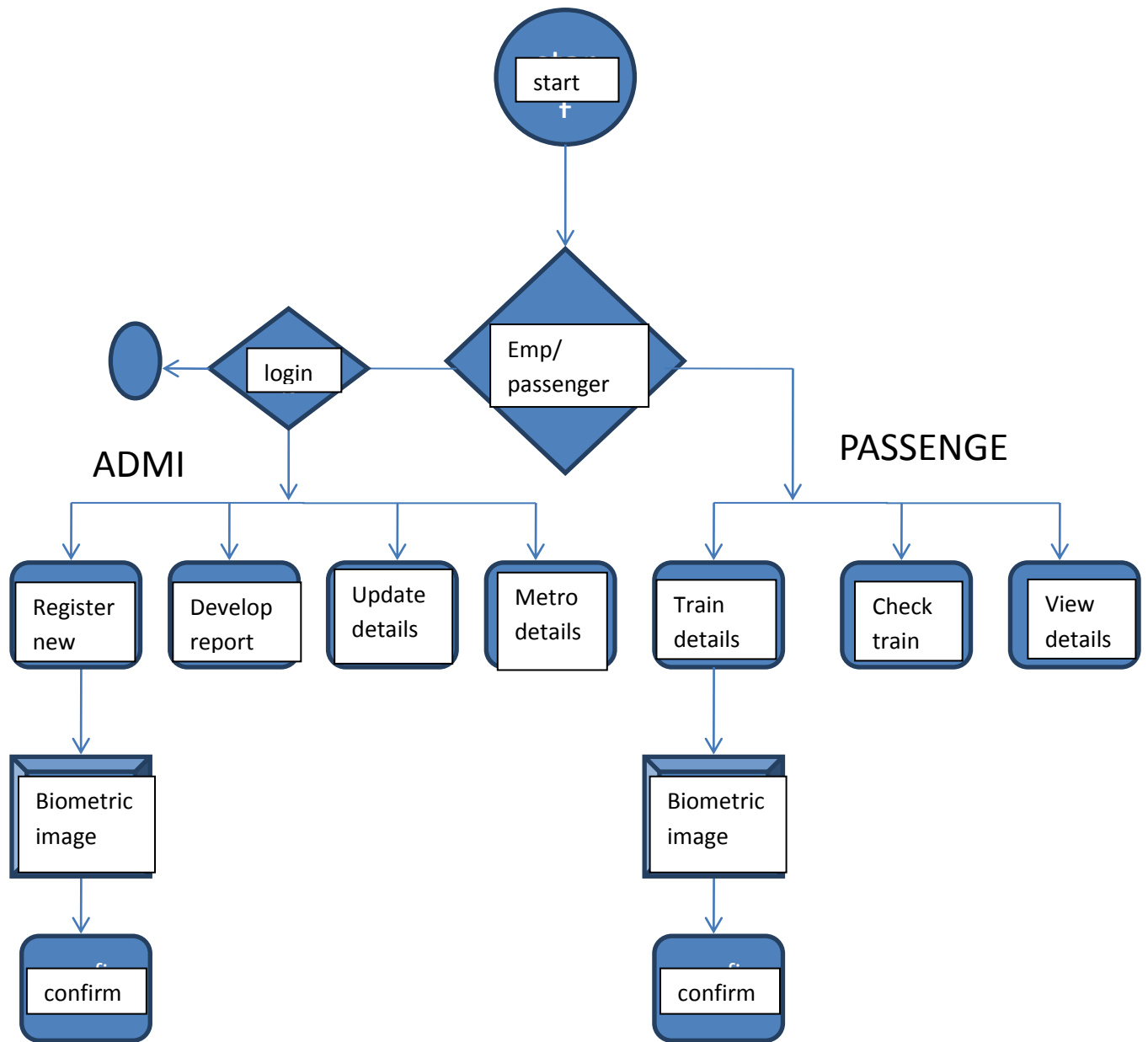


Figure 4.2: DATAFLOW DIAGRAM

# **CHAPTER 5: TESTING**

## **5.1 INTRODUCTION**

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub assemblies, assemblies and/or a finished product. It is the process of exercising software with the intent of ensuring that the

Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

## **5.2 TYPES OF TESTS**

### **5.2.1 UNIT TESTING**

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application. It is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

### **5.2.2 INTEGRATION TESTING**

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfactory, as shown by successful unit testing, the

combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

### **5.2.3 FUNCTIONAL TESTING**

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted.

Invalid Input : identified classes of invalid input must be rejected.

Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised.

Systems/Procedures: interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

### **5.2.4 SYSTEM TESTING**

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

### **5.2.5 WHITE BOX TESTING**

White Box Testing is a testing in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is used to test areas that cannot be reached from a black box level.

### **5.2.6 BLACK BOX TESTING**

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

## **5.3 UNIT TESTING:**

Testing will be performed manually and functional tests will be written in detail.

### **Test objectives**

- All field entries must work properly.
- Pages must be activated from the identified link.
- The entry screen, messages and responses must not be delayed.

### **Features to be tested**

- Verify that the entries are of the correct format
- No duplicate entries should be allowed
- All links should take the user to the correct page.

**Test Results:** All the test cases mentioned above passed successfully. No defects encountered.

## 5.4 INTEGRATION TESTING:

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

**Test Results:** All the test cases mentioned above passed successfully. No defects encountered.

## 5.5 ACCEPTANCE TESTING:

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

**Test Results:** All the test cases mentioned above passed successfully. No defects encountered.

## 5.6 VALIDATION

- User id & password cannot be blank while logging into the site.
- In the profile page or the new user account page there are some mandatory fields like vacancy no, name, resume title, company name etc. which cannot be left blank.
- In the modify password page user have to specify the login id as well as the old password & the new password.

## CHAPTER 6: IMPLEMENTATION

### 6.1 INTRODUCTION

The implementation and coding phase of the **Software Development Life Cycle (SDLC)** is the third phase of the **SDLC** process. The Implementation phase is when the end user of your software is foremost in your mind. During this phase we create the documentation and tools the customer uses to make informed decisions about how to deploy your software securely.

### 6.2 LAUNCH PAGE

Upon executing the project the launch setup pops on the screen, which on clicking results the login page. This is the main frame form that is achieved on executing the application.

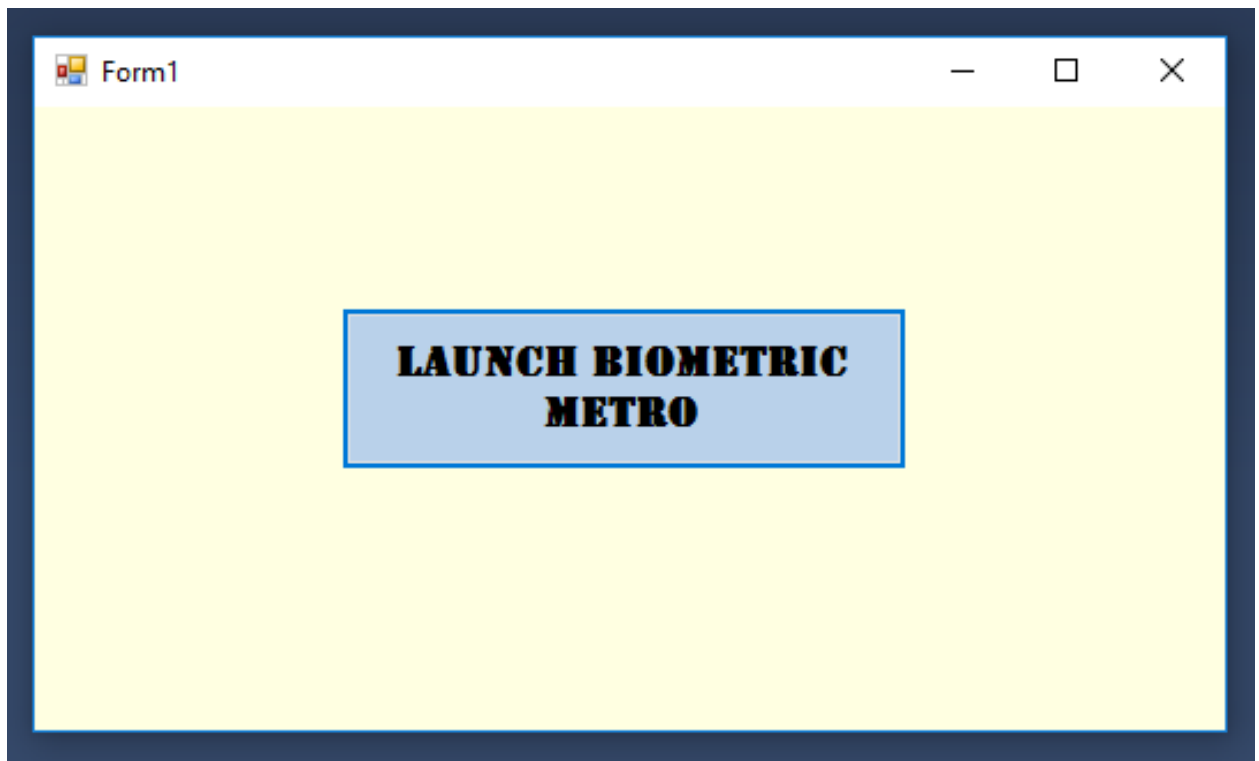


Figure 6.2: Launch Page

## 6.3 LOGIN PAGE

Upon clicking the launch biometric button on the previous form the Login Page pops on the screen. Where the User or the Admin can login into the application. This involves the authentication which checks in the backend for the give username and password, thus upon matching user name and password one can access into the application. Therefore the two possible logins are meant for:

1. Admin: Admin has more privileges when compared to the other users, like checking details, registering new, and also validating.
2. User: Only can view the database or the details and can board the train using the fingerprint.

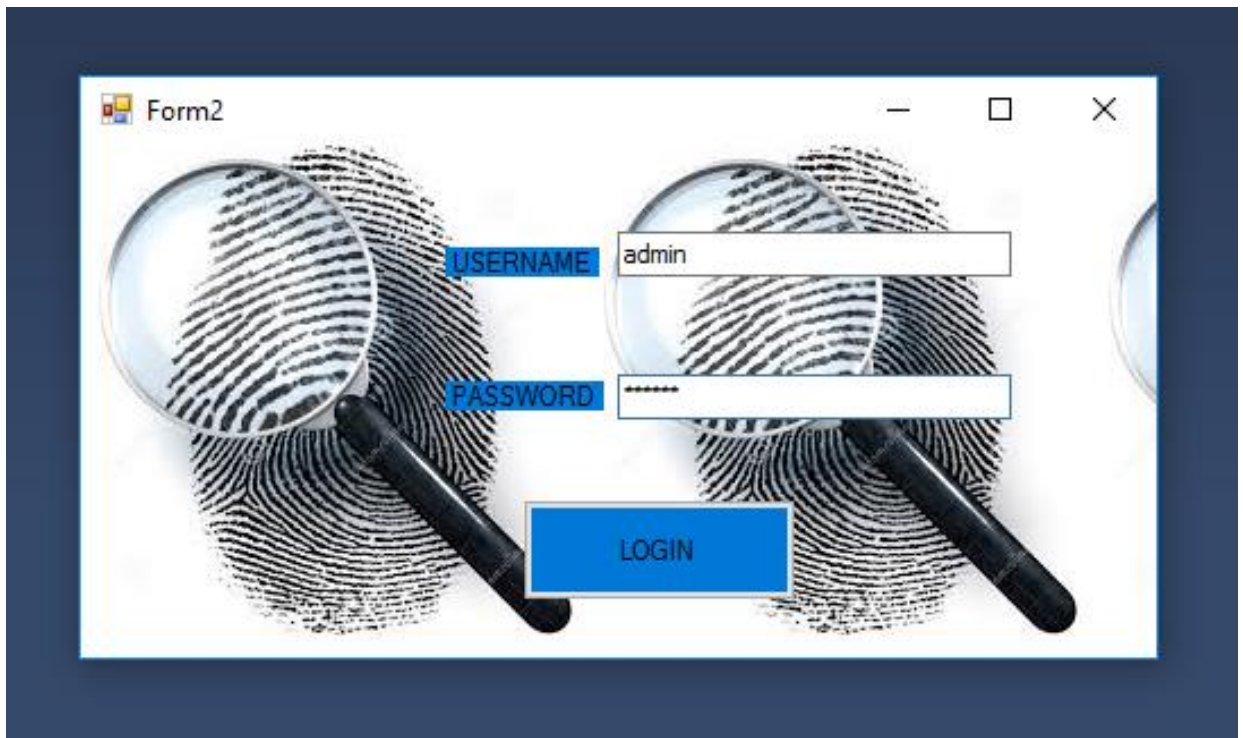


Figure: 6.3 Login page



## 6.4 HOME SCREEN FORM

Once the Login of User or Admin is confirmed, i.e the system validates the user as an official user or as the admin the following page is displayed onto the screen. It has various buttons for displaying the various pages that make the application a user friendly environment.

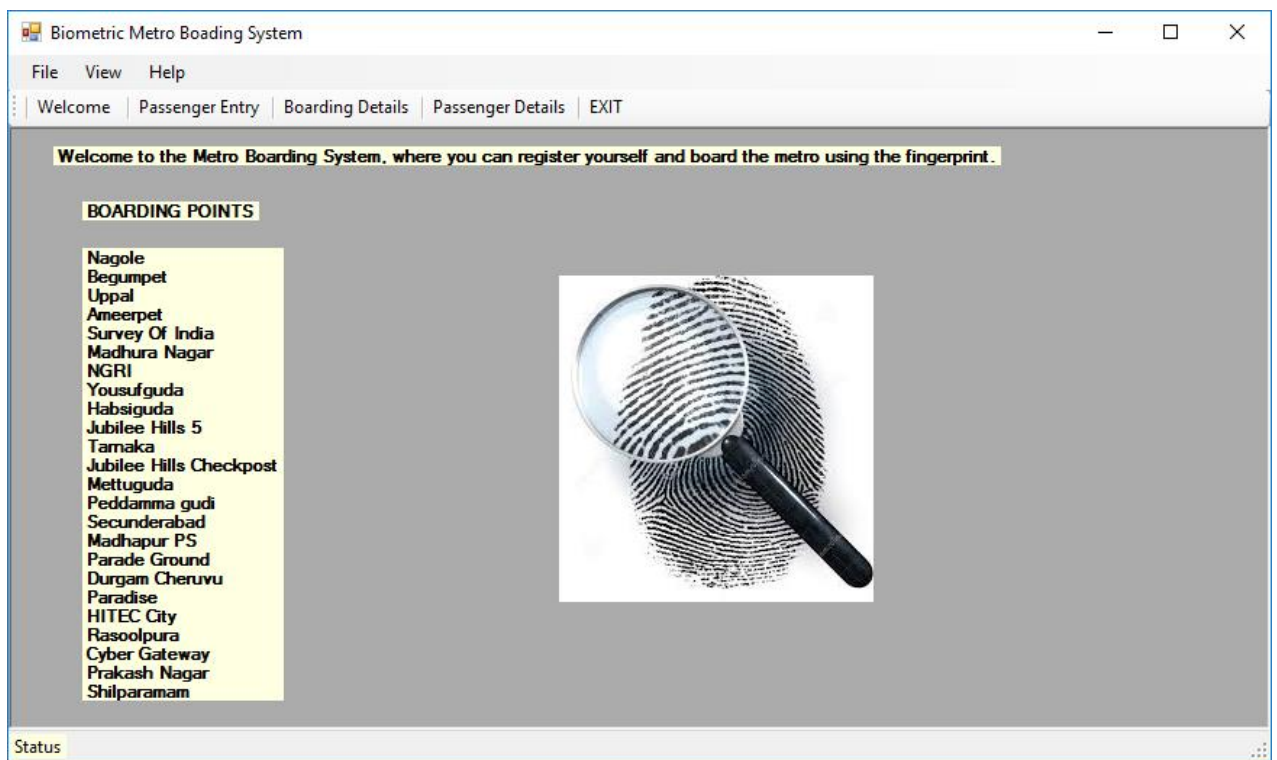


Figure 6.4: Home Screen Page

## 6.5 WELCOME FORM

Upon clicking the Welcome button on the Home Screen Page we get a pop-up that displays the information regarding the Biometric Metric Application, such that the user can know the information regards the version and etc. This pop-up closes on click of OK button at the bottom of the screen.

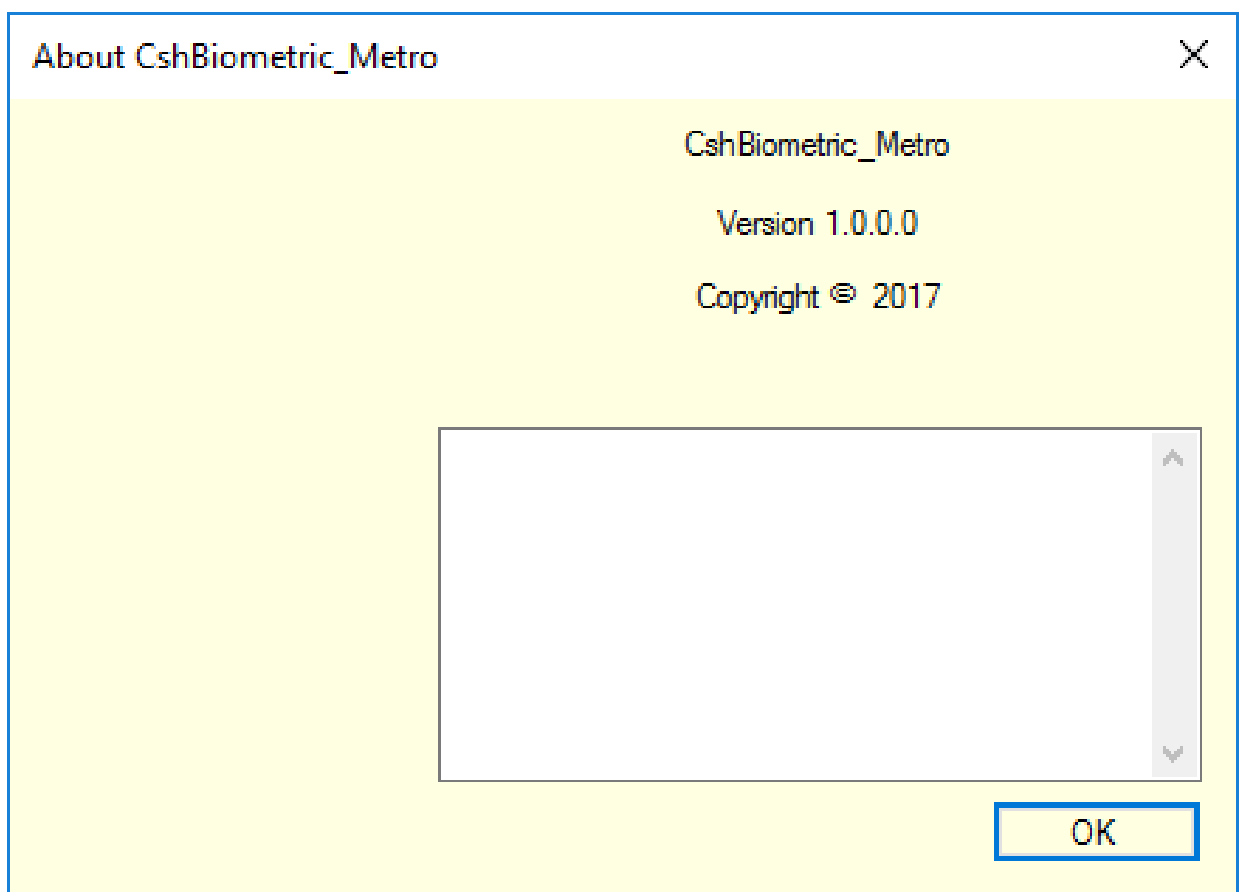


Figure 6.5: Welcome Page

## 6.6 REGISTRATION FORM

This page is visited on clicking the Passenger Details on the Welcome Screen. This page enables the admin to register new passengers with information such as Name, Aadhaar Card number, address etc. Hence here the passenger needs to scan or upload a picture of his finger, such that the image can be stored in the database and can be used later the device for validation. Along with details the user is requested to give the boarding point and destination and a cost is generated. Hence upon clicking save the details the user details are saved into the backend and can be used by the system to validate the passenger through the finger print.

The screenshot shows a web application window titled "passengerinfo". The form is organized as follows:

- Full Name:** Text input field containing "Sample".
- Aadhar Card No:** Text input field containing "67890".
- Address:** Text input field containing "xyz".
- Boarding Point:** Dropdown menu with "Begumpet" selected.
- Destination:** Dropdown menu with "Ameerpet" selected.
- Frequency:** Dropdown menu with "1 Month" selected.
- Upload Pic:** Button with an icon of a person surrounded by various symbols.
- Upload Scan:** Button with a fingerprint icon.
- Total Amount to pay:** Text input field containing "1000/-".
- Save the Details:** A large blue button at the bottom center.

Figure 6.6: Registration Form

## 6.7 AUTHENTICATION FORM

Once a passenger has been registered by the admin he/she can board the train that is in the given route. This thus is done in the form validate, where the passenger is requested to provide their fingerprint and the system performs a background task to identify the passenger and then thus the passenger is valid and boarded into the train. This information is then stored into the new database which is then displayed later in the boarding details.

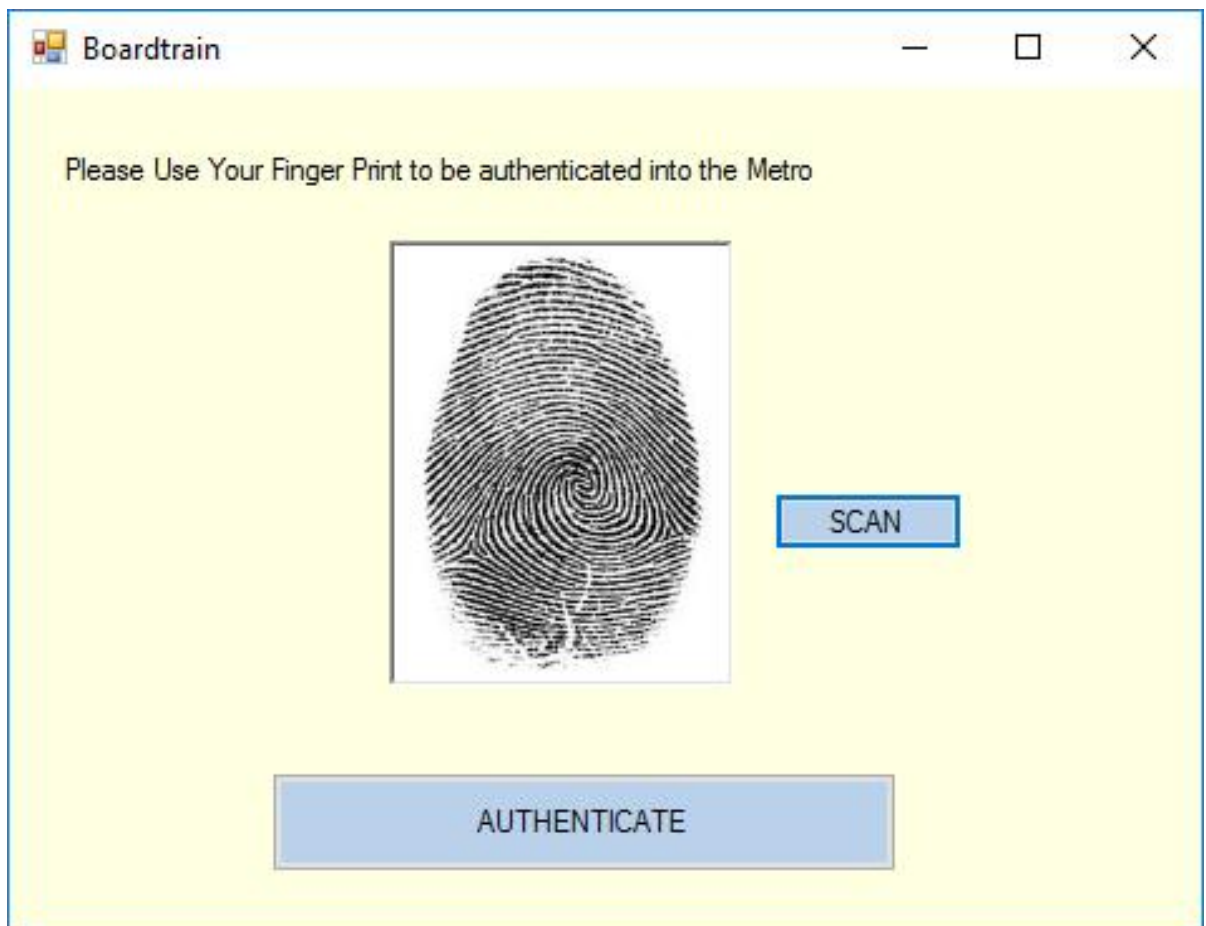
The image shows a software window titled "Boardtrain" with standard Windows window controls (minimize, maximize, close). The window has a light yellow background. At the top, it says "Please Use Your Finger Print to be authenticated into the Metro". In the center is a large rectangular area containing a grayscale fingerprint image. To the right of the fingerprint image is a blue button with the text "SCAN". Below the fingerprint image is a larger blue button with the text "AUTHENTICATE".

Figure 6.7: Validate or Boarding Form

## 6.8 VALIDATION SUCCESS FORM

Upon successfully validating a person based on the fingerprint given by the user, we create a pop-up with the name of the passenger and a welcome message. Hence this makes the passenger feel he has been identified and thus creates a confirmation message to the user of the system. This system can also generate an error message such as entered a wrong user or such as failed to validate message to the user.

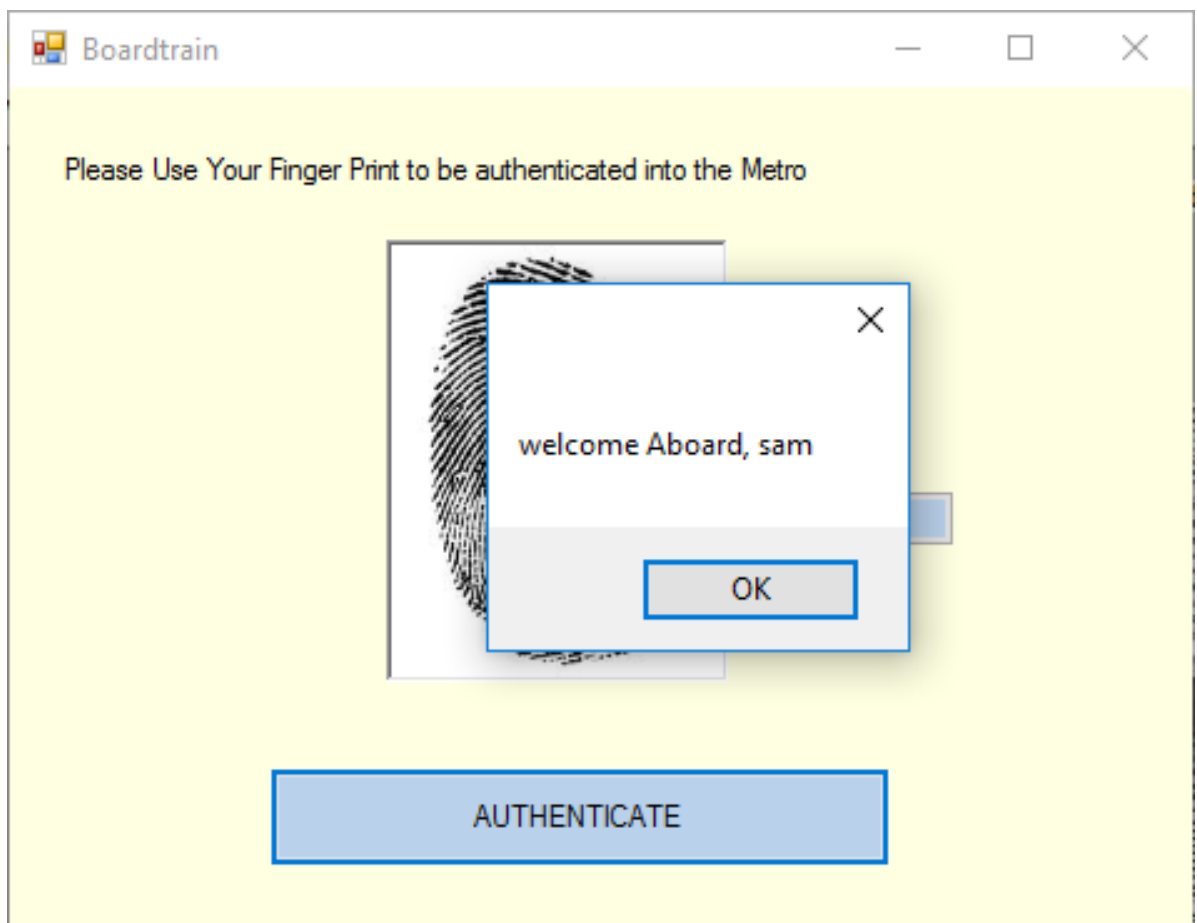
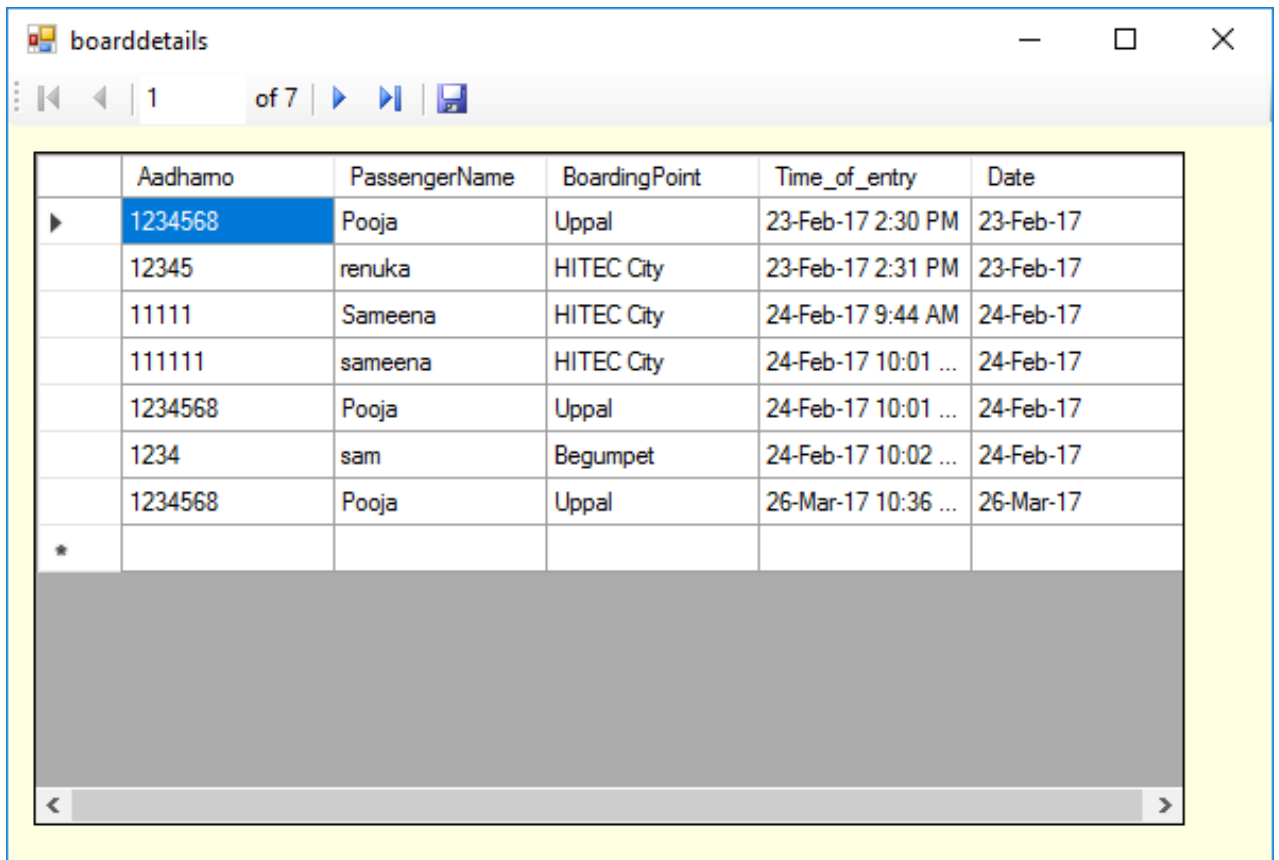


Figure 6.8: Successfully validating a User

## 6.9 PASSENGER DETAILS FORM

Once a user has successfully validated and messaged, immediately the system inserts the data of the passenger into the boarding form, where the database of the passengers who have boarded the train is shown along with the time and date of entry. This enable the users to keep track of the passengers and maintains details of the passengers.



	Aadhamo	PassengerName	BoardingPoint	Time_of_entry	Date
▶	1234568	Pooja	Uppal	23-Feb-17 2:30 PM	23-Feb-17
	12345	renuka	HITEC City	23-Feb-17 2:31 PM	23-Feb-17
	11111	Sameena	HITEC City	24-Feb-17 9:44 AM	24-Feb-17
	111111	sameena	HITEC City	24-Feb-17 10:01 ...	24-Feb-17
	1234568	Pooja	Uppal	24-Feb-17 10:01 ...	24-Feb-17
	1234	sam	Begumpet	24-Feb-17 10:02 ...	24-Feb-17
	1234568	Pooja	Uppal	26-Mar-17 10:36 ...	26-Mar-17
*					

Figure 6.9: Boarding details/ Passenger in train

## 6.10 LOGIN TABLE:

### 6.10.1 STRUCTURE

	Name	Data Type	Allow Nulls	Default	
	userid	varchar(50)	<input type="checkbox"/>		
	password	varchar(50)	<input type="checkbox"/>		
			<input type="checkbox"/>		

Figure 6.10.a Login Table

### 6.10.2 TABLE DATA

	userid	password
▶	Admin	admin
	User	user
	passenger	passenger
★	NULL	NULL

Figure 6.10.b Login Table Data

## 6.11 BOARDING DETAILS

### 6.11.1 STRUCTURE

	Name	Data Type	Allow Nulls	Default	
	Aadharno	varchar(50)	<input checked="" type="checkbox"/>		
	PassengerName	varchar(50)	<input checked="" type="checkbox"/>		
	BoardingPoint	varchar(50)	<input checked="" type="checkbox"/>		
	Time_of_entry	datetime	<input checked="" type="checkbox"/>		
	Date	date	<input checked="" type="checkbox"/>		
			<input type="checkbox"/>		

Figure 6.11.a Boarding Table

### 6.11.2 TABLE DATA

	Aadharno	PassengerName	BoardingPoint	Time_of_entry	Date
▶	1234568	Pooja	Uppal	23-Feb-17 2:30:...	23-Feb-17
	12345	renuka	HITEC City	23-Feb-17 2:31:...	23-Feb-17
	11111	Sameena	HITEC City	24-Feb-17 9:44:...	24-Feb-17
	111111	sameena	HITEC City	24-Feb-17 10:01...	24-Feb-17
	1234568	Pooja	Uppal	24-Feb-17 10:01...	24-Feb-17
	1234	sam	Begumpet	24-Feb-17 10:02...	24-Feb-17
	1234568	Pooja	Uppal	26-Mar-17 10:3...	26-Mar-17
	1234568	Pooja	Uppal	30-Mar-17 8:17:...	30-Mar-17
	1234	sam	Begumpet	30-Mar-17 8:18:...	30-Mar-17
	1234	sam	Begumpet	30-Mar-17 8:31:...	30-Mar-17
	111111	sameena	HITEC City	08-Apr-17 11:46...	08-Apr-17
★	NULL	NULL	NULL	NULL	NULL

Figure 6.11.b Boarding Table Data



## 6.12 PASSENGER DETAILS

### 6.12.1 STRUCTURE

	Name	Data Type	Allow Nulls	Default
	fullName	varchar(50)	<input checked="" type="checkbox"/>	
	aadhar	varchar(50)	<input type="checkbox"/>	
	address	varchar(MAX)	<input checked="" type="checkbox"/>	
	boardpt	varchar(50)	<input checked="" type="checkbox"/>	
	destpt	varchar(50)	<input checked="" type="checkbox"/>	
	frequency	varchar(50)	<input checked="" type="checkbox"/>	
	amtpay	varchar(50)	<input checked="" type="checkbox"/>	
	facepic	image	<input checked="" type="checkbox"/>	
	fingerprint	image	<input type="checkbox"/>	
	fpadd	varchar(MAX)	<input type="checkbox"/>	
			<input type="checkbox"/>	

Figure 6.12.a Passenger Table

### 6.12.2A TABLE DATA (A)

	fullName	aadhar	address	boardpt	destpt	frequency
▶	sameena	111111	sainikpuri	HITEC City	Rasoolpura	One ride
	sam	1234	7777777777777777	Begumpet	Uppal	One ride
	renuka	12345	kothapet	HITEC City	Begumpet	One ride
	gayatri	1234567	asdfghjk	HITEC City	Begumpet	6 Months
	Pooja	1234568	wertyuio	Uppal	HITEC City	One ride
★	NULL	NULL	NULL	NULL	NULL	NULL

Figure 6.12.b Passenger Table Data

### 6.12.2B TABLE DATA (A)

frequency	amtpay	facepic	fingerprint	fpadd
One ride	40/-	0xFFD8FFE0001...	0x49492A00080...	C:\Users\Same...
One ride	40/-	0xFFD8FFE0001...	0xFFD8FFE0001...	C:\Users\Same...
One ride	40/-	0xFFD8FFE0001...	0x49492A00080...	C:\Users\Same...
6 Months	6000/-	0xFFD8FFE0001...	0xFFD8FFE0001...	C:\Users\Same...
One ride	40/-	0xFFD8FFE0001...	0xFFD8FFE0001...	C:\Users\Same...
NULL	NULL	NULL	NULL	NULL

Figure 6.12.b Passenger Table Data (continue)

## CHAPTER 7: CONCLUSION

Among all the biometric techniques fingerprint recognition is the most popular method. Every individual who wishes to travel through the metro transport needs to provide the finger print, enter the details and thus can travel through the metro system. No one can forge or cheat this metro rail system as it is almost impossible to cheat or forge one's finger print.

Successful implementation of this project can enhance security of the passengers. Misplace or Loss of information is not possible and database of the passengers helps in law enforcement and secures the details. Thus, passenger is considered to take any train to their destination by validating and hence paying the travel money before boarding the metro and the person must scan their finger or consider the image only then the person can board the metro. Hence this minimizes the time for the process and to enhances the security which is consider to be best and the fastest way to implement a biometric system into the application.

Although the system can enhance in faster and easier entry of passenger there are certain drawbacks to the concept. Therefore, eliminating some of these issues can make the idea of biometric authentication system more efficient and thus more effective. Some of the future implementations could include:

- Acquiring a database which can store a very large number of fingerprints such that it can include the data images of fingerprints
- A very strong connectivity between Database and system(host) and System
- A faster mechanism which enables the user to identify or compare images
- Handling various fingerprints with low accuracy, and readability
- Alternative approach, which includes data mining such that it includes reduced time in accessing data

## CHAPTER 8: BIBLIOGRAPHY

### REFERENCES

- [1] A.L.H. Jin, A. Chekima, J.A. Dargham, Liau Chung Fan, Sch. of Eng. & Inf. Technol., Universiti Malaysia Sabah, Malaysia “*Fingerprint identification and recognition using backpropagation neural network*” IEEE Version, Xplore, Nov 2002
- [2] P. Stanley; Wilson Jeberson; V. V. Klinsega, 2009 International Conference on Future Networks Year: 2009: “*Biometric Authentication: A Trustworthy Technology for Improved Authentication*” Pages: 171 - 175, DOI: 10.1109/ICFN.2009.49
- [3] Wiki Biometrics Website  
<http://biometrics.pbworks.com/w/page/14811349/Advantages%20and%20disadvantages%20of%20technologies>
- [4] Sukkharak Saechia, Jeerasuda Koseeyaporn, Paramote Wardkein, "Human Identification System Based ECG Signal", TENCON 2005 2005 IEEE Region 10, pp. 1-4, 2005.
- [5] Wikipedia Wiki – Visual Studio  
[https://en.wikipedia.org/wiki/Microsoft\\_Visual\\_Studio](https://en.wikipedia.org/wiki/Microsoft_Visual_Studio)
- [6] Wikipedia Wiki – SQL Server  
[https://en.wikipedia.org/wiki/Microsoft\\_SQL\\_Server](https://en.wikipedia.org/wiki/Microsoft_SQL_Server)
- [7] Download Source here: [http://download.cnet.com/Microsoft-Visual-Studio-2010-Professional/3000-2212\\_4-10618634.html](http://download.cnet.com/Microsoft-Visual-Studio-2010-Professional/3000-2212_4-10618634.html)
- [8] BioElectronix Website  
[http://www.bioelectronix.com/what\\_is\\_biometrics.html](http://www.bioelectronix.com/what_is_biometrics.html)