ADDIS ABABA UNIVERSITY

ADDIS ABABA INSTITUTE OF TECHNOLOGY

AFRICAN RAILWAY CENTER OF EXCELLENCE

ANALYSIS AND DESIGN OF AN OPTIMIZED AUTOMATIC FARE COLLECTION SYSTEM FOR ADDIS ABABA LIGHT RAIL TRANSIT

A Thesis in Railway Engineering (Traction and Train Control)

By: Mola Ayenew Teferi

Id: GSR 6419/10

August, 2019

Addis Ababa, Ethiopia

A Thesis

Submitted in Partial Fulfillment of the Requirements for the Degree of Master of Science

APPROVAL

The undersigned have examined the thesis entitled '**Analysis and Design of an Optimized Automatic Fare Collection System for Addis Ababa Light Rail Transit'** presented by **Mola Ayenew**, a candidate for the degree of **Master of Science** and hereby certify that it is worthy of acceptance.

Dr.Eng Yihenew Wondie		
Advisor	Signature	Date
Internal Examiner	Signature	Date
External Examiner	Signature	Date
Chair person	Signature	Date

UNDERTAKING

I certify that the research work titled "**Analysis and Design of an Optimized Automatic Fare Collection System for Addis Ababa Light Rail Transit"** is my own work. The work has not been presented elsewhere for assessment. Where materials have been used from other sources have been properly acknowledged/referred.

Mola Ayenew

ABSTRACT

Recently, everything in the world becomes smart and digitalized. Many advances have been made in the transportation sector too. However, railway transport in Ethiopia has been an area where such new advances have turned their faces out. Therefore, this paper, will introduce an Automatic Fare Collection system (AFC) to up bring the railway transportation system in Ethiopia to the best standard. In addition, this paper will focus on different ways of implementing AFC, which will be suitable for AA-LRT. AFC is one of the important technologies in railway transportation around the world. Different countries use different ways of implementation of AFC to their railway transportation. One of the aims of this paper is to study different implementation mechanisms of countries and analyses the best and suitable standard for AA-LRT. The cost analysis should also be considered when such kinds of technologies are implemented. Nowadays, in AA-LRT every train is controlled by a conductor. The conductor will collect money from each passenger and issues tickets and the printed paper or tokens are used as tickets. This process needs man power and the passengers whom are utilizing the transportation services will not be satisfied. Because, they will wait a lot of time on the queuing to get ticket from the cashier. In the proposed system, every AA-LRT stations have one or more RFID (Radio-Frequency Identification) Reader. RFID is a technology whereby digital data encoded in RFID tags or smart labels are captured by a reader through radio waves. The Reader senses the RF signals coming from the passenger identification card and thus recognizes the entry, existence and exit of the passenger. Based on the signals from the card is cut off, the fare of the journey is calculated and is deducted from the passenger's account, which is linked with the backend system. The RFID card will recharge in different ways, it may recharge from stations, mobile account and/or bank account. The system will deliver an end-to-end solution for fare collection, ticketing, and payments that provides secure and faster transactions, more convenience and smooth passenger flow during peak hours, and efficient collection of fare ensuring no fraud takes place.

Keywords: AFC, RFID, LF, HF, UHF, RF, Radio Wave, AA-LRT

ACKNOWLEDGEMENT

First and foremost, praises and thanks to God, the Almighty, for his showers of blessings throughout my research work to complete the research successfully.

I would like to express my deepest appreciation to all those who provided me the possibility to complete this research. A special gratitude goes to my advisor, Dr.Eng Yihenew Wondie, for his contribution in stimulating suggestions and encouragement.

Furthermore, I am extremely grateful to my parents for their love, prayers, caring and endless support in educating and preparing me for the future. Finally, I express my thanks to my brothers and sisters for their love, understanding, prayers and continuing support to complete this research work.

LIST OF ABBRIVATIONS

AA-LRT	Addis Ababa Light Rail Transit
AFC	Automatic Fare Collection
AVLS	Automatic Vehicle Location System
CSC	Contactless Smart Card
GSM	Global System for Mobile communication
GPS	Global Positioning System
GUI	Graphical User Interface
HF	High Frequency
IDE	Integrated Development Environment
IOT	Internet of Things
LF	Low Frequency
MySQL	My Structured Query Language
OD	Origin-Destination
PT	Public Transport
RF	Radio Frequency
RFID	Radio Frequency Identification
UHF	Ultra-High Frequency
XAMPP	Cross platform Apache server MySql PHP Perl programming language

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1
INTRODUCTION

1.1 Background

A public transport is one of the most important building blocks for the economy of our country. In an economy, where there is some form of transport network, people without their own means of transport are reliant on public transport. In Addis Ababa city, there are three main modes of transport system, which includes trains, buses and taxis. Taxis are not only the most accessible but they are the most common mode of transport within the Addis Ababa environment with the basic objective of transporting commuters from one point to another.

Recently, passenger flow in AA-LRT is increasing due to its long distance coverage and low cost relative to other mode of transportation. AA-LRT is a new mode of transportation in Addis Ababa city. The system uses manual ticketing system for passenger to travel from one point to another.

Every train is controlled by a conductor. The conductor will collect money from each passenger and issues tickets and the printed paper or tokens are used as tickets. This system has many disadvantages. The passenger have to keep the ticket until the stopping of their destination, the conductor should ensures that everyone has got the ticket, the time taken for ticketing is comparatively more and too much paper is needed to print the ticket. For example, if a passenger wishes to travel in train, he/she must have money with their pocket, and then the conductor will collect the money and give ticket. This process is repeated for all passengers. Hones, it takes more time and man power. This paper will introduce an Automatic Fare Collection system, which will help both passengers and the company in many aspects.

An AFC system integrates various functionalities of process control, monitoring, and maintenance of various operations needed throughout the process of issuing, distribution, sale and validation of transport tickets [1].

An AFC system is one of the modern technologies in public transport around the world with its convenience to use, efficiency and effectiveness. This system records a transaction for each passenger boarding (origin and destination), containing attributes

regarding to the route, the train and the travel card used, along with the timestamps using the coordination of RFID and Micro-controller [2].

This paper is based on the concept of changed ticketing structure by using RFID technique. RFID (Radio Frequency Identification) is a technology whereby digital data encoded in RFID tags or smart labels are captured by a reader through radio waves. RFID tags are operated at three different frequencies: Low Frequency (LF), High Frequency (HF) and Ultra High Frequency (UHF) rate. The RFID tags/cards which is using LF range is up to 10 cm, HF will travel up to 1m and the UHF will travel much distance up to 10 to 15 m [2]. Different countries use different frequency of operational range depending on the design of the AFC system.

In this proposed system, every train stations have RFID reader. A radio frequency identification reader (RFID reader) is a device used to gathering the information from the RFID tag, which is used to track individual objects. Radio waves are used to transfer data from the tag to a reader. RFID is a technology similar in theory to bar codes. The RFID tag must be within the range of an RFID reader, which ranges from 3 to 300 feet, in order to be read. RFID technology allows several items to be quickly scanned and enables fast identification of a particular object, even when it is surrounded by several other items [3].

1.2 Automatic Fare Collection System

An Automatic Fare Collection (AFC) system is a platform which is used to integrate various functionalities of process control, monitoring and maintenance of the various operations needed throughout the process of issuing, distribution, sale and validation of transport tickets. The system also manages the access control to transportation and the processes of consolidation, settlement and clearing in multi-modal systems [4].

AFC solution of each city or region has to be adapted to the different transportation operational models. That requires the analysis and definition of all processes and functionalities of the tariff model.

AFC system is a contactless, end-to-end solution for collecting fare payment, replacing traditional ticketing system with automated ticketing and fare collection systems. It comprises automatic gate machines, ticket vending machines, recharging terminals, and ticket checking machines. This system enables efficiency in fare collection, improves the

overall transaction rate and saves extensive amount of time, thereby minimizing waiting time in queues and searching for cash to buy tickets. In addition, it is the easiest method for revenue collection. It offers travelers to carry one card for use in all transit modes.

An automated fare collection system has various software enabled features including Oracle, Sybase support, support multitier architecture, and XML web services. Fare collection technology platforms such as Smart cards, NFC, RFID and OCR systems ensure full interoperability and independence of service providers. Factors such as increase in demand for smartphones, enhanced efficiency, reduced frauds, flexible fare plans, and enhanced security system strengthen the growth of automated fare collection system market. However, high installation and maintenance costs of automated fare collection system hinder the market growth. Opportunities such as increase in high speed of public transport, rise of technology payment, and global interoperability are likely to foster automated fare collection system market growth [4].

1.3 Ticketing Technology

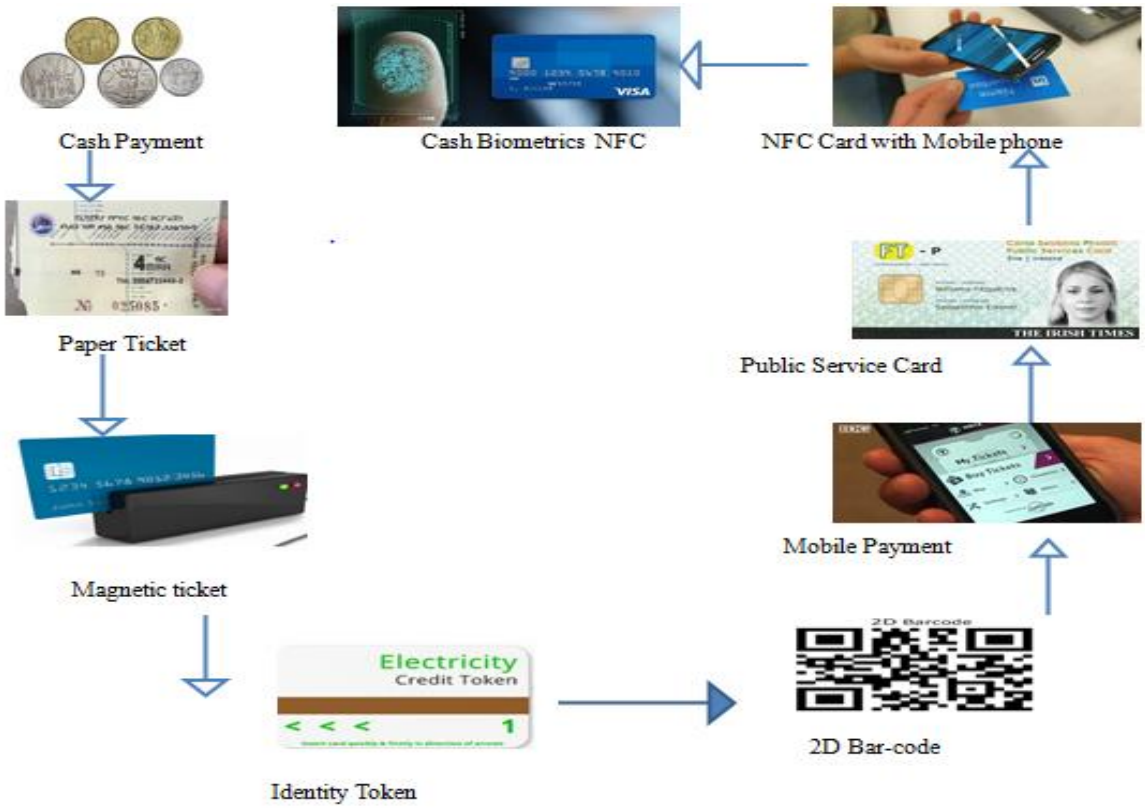

Figure 1-1: Evolution of Ticketing Technology [5, 6, 7, 8]

Based on a review of international experience, it can be concluded that no unified and standardized solutions have emerged with regard to existing AFC systems in major cities covering the totality of the operation of the system. Instead, there is competition between alternative models that include national standards, and proprietary systems. Example, Oyster in London, Octopus in Hong Kong, ITSO, Calypso, OV-chipkaart in the Netherlands. In the case of London Transport for London magnetic stripe, smartcard, EMV bank cards and mobile EMV. Travel media are always devised and developed in accordance with the business and security demands and IT capabilities of the time [5].

1.3.1 Smart Cards

A smart card is a credit-card-sized piece of plastic which has a microchip embedded in it. This chip is the 'smartness' of the smart card, and performs all the functions required by the card, which is used to storing data, processing data, writing data, etc. Smart-card chips come in two broad varieties memory-only chips and microprocessor chips. Memory-only chips have storage space for data and a reasonable level of built-in security. Microprocessor chips in addition to memory employ a processor controlled by a card operating system with the ability to process data on-board, as well as carrying small programs capable of local execution [6].

A smart card is therefore characterized uniquely by its chip, the ability to read and write data, and with its ability to store much more data up to about 64 000 bytes all within an extremely secure environment. This makes the card a reusable ticket or token which can hold credit or subscription rights, details of the card holder and access to a range of services and applications.

Paper [7, 8], describes the main issues associated with smart card technology

Interfaces: when selecting a smart card for a particular application, one of the most important criteria that must be considered is how that smart card communicates with other devices.

There are two basic choices of smart card interface. **Smart card with contacts** is the card has a set of six or eight metal contacts on the surface of the card. When the card is inserted in a reading device, spring-loaded contacts make a connection to each of these metal pads (such cards can be seen in use for telephone payments and bankcards). **Contactless** smart card is the card uses inductive radio frequency coupling to communicate with the reading device. Generally, the card needs to be placed in very

close proximity to the reader (known as 'proximity' cards), although a new standard for cards and tags being read at a distance of up to 1 m was published in June 2000, known as the 'vicinity' card standard.

The various types of smart card: Standard contact card, with one chip-set and one interface; Contactless card, with one chip-set and one interface; Combi-card, with one chip-set and two interfaces; and Hybrid-case, with two chip-sets and two interfaces

Memory: The size of the dynamic memory on a smart card into which data can be written or changed is limited, at present, both by the cost of this kind of memory (EEPROM) and by the physical size of the memory chip within the card's processor.

Reliability: Card reliability remains an important objective. Concerning the overall mechanical reliability of the cards, clearly the more complex the card, with more chips and bonding wires, the more likely is the card to fail before expected. This, to some extent, can be overcome by reducing the number of chips through greater integration.

Security: Security of smart cards and their associated systems is always an issue that is endlessly debated and focuses on three levels. Physical level describes how good the card's tamper-resistant module is. Logical level describes about the Cryptographic capabilities and others security of the contactless and other systems' interface.

1.3.2 Near-Field Communication (NFC)

The NFC technology is a wireless and contactless technology of transmission of radio data over short distances, enabling an easy and secure bidirectional communication. The use of this technology in practice would mean a significant change in the quality of passenger transport. It is possible to integrate payment or loyalty cards, IDs and driving licenses, electronic keys or travel documents into the NFC technology. The use of the NFC technology has many advantages, which result from noncash payments, and the new advantages include: In case of the card expiry date, it is not necessary to visit a bank institute to get a new card, the NFC technology can also replace physical or virtual keys (car, home, log into the PC), the NFC technology facilitates communication between all devices, which are currently widely used [9, 10].

NFC is a standards-based, short-range wireless connectivity technology that enables simple and intuitive two-way interactions between electronic devices. With NFC technology, consumers can perform contactless transactions, access digital content and

connect NFC-enabled devices with a single touch. NFC simplifies setup of some longer-range wireless technologies, such as Bluetooth and Wi-Fi.

The following chart shows how NFC compares in range and speed with other wireless technologies that can be used in a mobile phone.

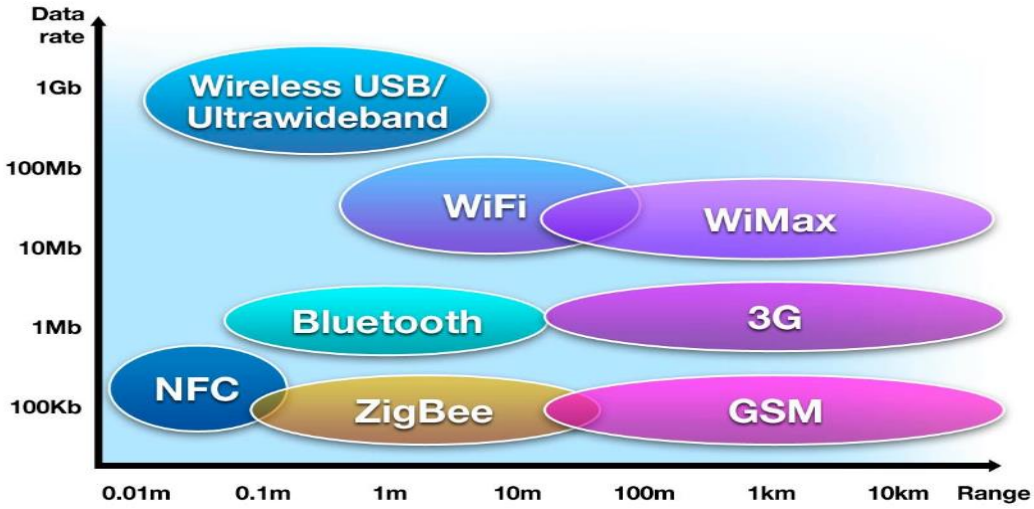

Figure 1-2: Different Contactless Systems [9]

Communication occurs when two NFC-compatible devices are brought within about four centimeters of each other. By design, NFC requires close proximity and it offers instant connectivity, which provides an intuitive consumer experience that can be readily applied to the transit environment [10].

1.3.3 Ticketing Technology in Railway Transport

Train tickets have served railways well for the 150 years or more that they have existed. Until relatively recently, there seemed no better option for ensuring that passengers have paid the correct fare. But the inexorable march of technology, coupled with exploding passenger numbers prompting new demands for a faster fare system, has brought a new innovation to the rail market paperless ticketing [11].

Under these new systems, which are now in operation at many metro and light rail networks around the world, fare information can be stored on convenient smart cards, with contactless technology allowing commuters to pass through ticket barriers without any need to insert and reclaim a paper ticket. This is a boon for overcrowded metro services that need to speed up the ticketing system to reduce congestion.

Ticketless systems are predominantly found on city subways and light rail networks, which are smaller and easier to manage than national rail lines. But with an increasing amount of sophistication being packed into smart card devices, it seems only a matter of time before the world will bid a fond farewell to traditional paper tickets for good [11].

1.3.4 Benefits of Contactless System for Railway Transport

For operators, there are several important advantages. The use of this technology makes for much better data collection and the ability to offer more products and more opportunities to customers. Operators get much more information coming from every transaction realized with a smart card, allowing them to understand the real needs of users.

There is also the reduction of maintenance cost of the contactless systems in comparison to traditional magnetic systems. Eliminating the mechanical components which process paper tickets, combined with the reliability and sturdiness of the contactless readers and the elimination of elements that need to be replaced periodically makes the maintenance of these new systems much cheaper [12].

1.4 Statement of the Problem

The problem can be seen from AA-LRT's and passenger's perspectives. From AA-LRT's perspective; the current cash based payment of the passengers to the cashier may lead to cash theft and money laundering by the cashiers. Cash theft and money laundering activities are the major challenges increasing in the current payment system. The cost paid to the cashier of the man power who work in all stations is very large.

From passenger's perspective, passengers have to keep the ticket until their stopping destination; the conductor should ensure that everyone has got the ticket to travel by the train. In case, if the passengers do not have money in cash, he/she will not able to use railway transportation system even if they have money on their mobile/bank account. In addition, passengers will wait a lot of time on the queuing to get ticket from the cashier.

Generally, AA-LT use the old way to collect fare and give service to the passengers and this leads the passengers not comfortable by the service and they may look to other modes transportation. As a result, this will reduce the overall revenue of the company.

1.5 Objective of the Research

1.5.1 General Objective

The main aim of this paper is to analyze the different AFC system implementations and best practices and design an Optimized Automatic Fare Collection System using RFID for Addis Ababa Light Rail Transit.

1.5.2 Specific Objective

In order to achieve this aim, the following specific objectives require to be investigated.

- Study about AFC system and RFID technology and their application on public transportation.
- Analyze the different AFC implementation on different countries and understand its pros and cons.
- Study the effective implementation of AFC system for AA-LRT
- Design the overall platform of the system
- Develop an AFC system using Arduino Uno and RFID as a prototype.

1.6 Research Methodology

The objective of this paper is to design user friendly AFC system that allows a passenger to enter to a station if he/she has a registered RFID tag and calculating the fare based on the information provided in the screen. When a passenger taps his/her card to the machine, fare collection is done and the gate opened automatically. The RFID tag is rechargeable, where as it can be recharged in train depot or nearest central station.

The proposed methodology introduces an Automated Fare Collection System (AFCS) to bring up the railway transportation system of AALRT into the best standard. The prevailing ticketing system had many malfunction, malicious argument among public and corruption. It also aims to reduce fare-related fraud and revenue loss through open standard, secured transaction technology.

In this paper, an AFC system using RFID technology will be introduced in AA-LRT, the system will have both the hardware and software components interacting to each other. Every user has their own unique RFID card and there are many station machines at every station. Therefore, the system includes some hardware components (such as Micro-

controller, LCD Display, Servo Motor, LED lights, RFID Reader and Tags) that are used to develop the prototype of the proposed system.

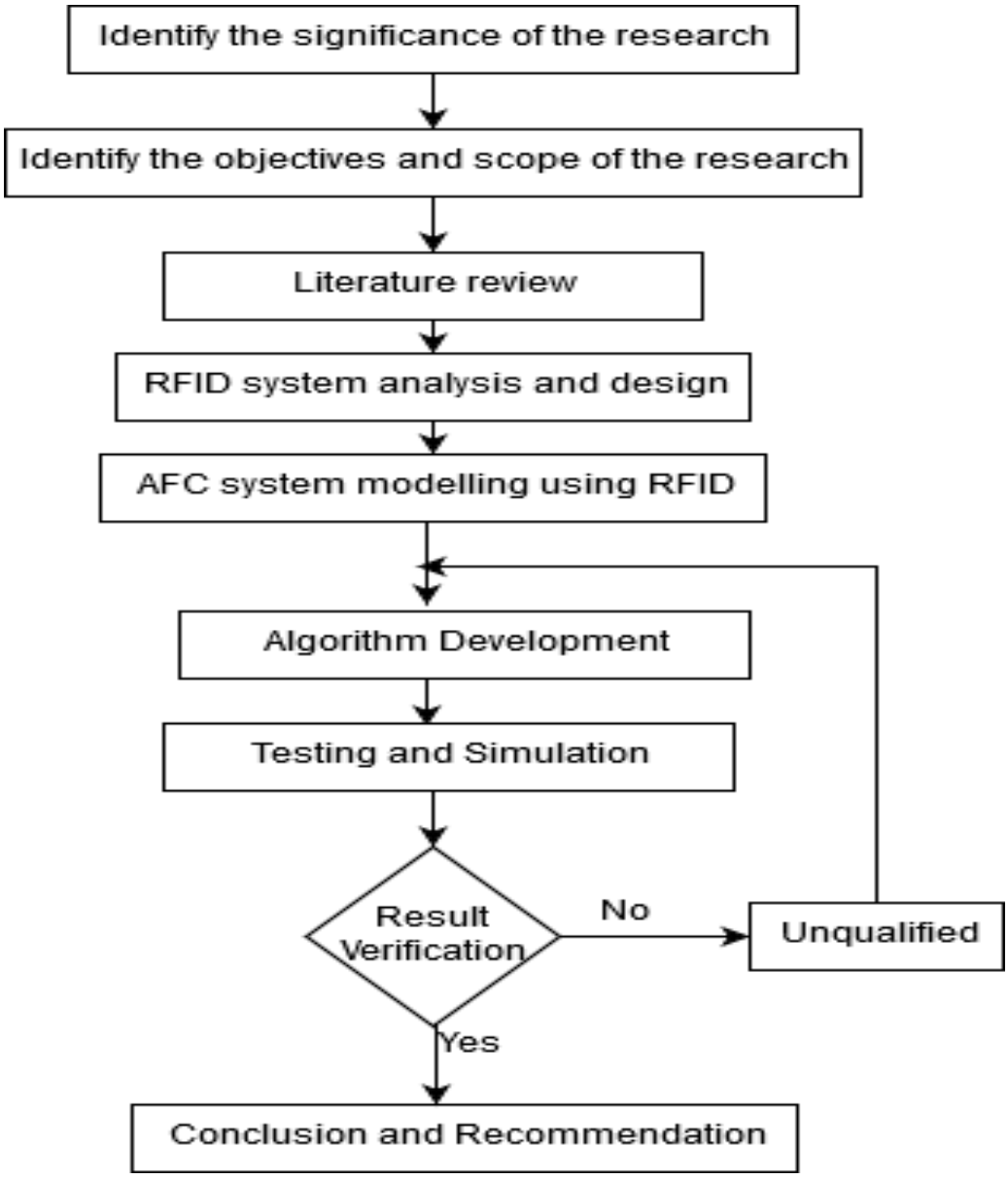

Figure 1-3: Research Methodology

A micro-controller is use to program this system by interfacing Arduino Uno IDE with RFID module. On the other hand, the back-end system is designed by using Microsoft Visual Studio and XAMPP Server. Connection between the hardware and software components, passenger registration, update, delete of user data have done at the back-end system. Using the information provided on the screen, a passenger can provide the start and end of his/her journey. Based on the information provided, the fare will be reduced

from this card and the updated information will be stored in the system. By implementing this system the efficient ticketing can be implemented and cash loose will be avoided.

1.7 Significance of the Research

The study and analysis of RFID technology is a new learning paradigm in the design of Automatic Fare collection system for Addis Ababa Light Rail Transit and other public transports. The research goal is to design an AFC system that helps passenger to use railway transport in more efficient and convenience way.

1.8 Scope of the Research

This research intends to introduce, familiarize, analysis, design and implement an AFC for AA-LRT. The research will try to deliver a simple prototype for the entire system using one of the suitable implementation systems, which will drive for the development of the real system in the future.

CHAPTER 2
RESEARCH BACKGROUND AND RELATED WORKS

2.1 Radio Frequency Identification

Radio Frequency Identification (RFID) is wireless communication technologies that use radio wave to automatically identify objects or people. It is a device used to gather information from an RFID tag and read that data back to the user, which is used to track individual objects. Radio wave is a communication medium used to transfer data from the tag to reader. RFID is an emerging technology and one of the most rapidly growing segments of today's automatic identification and data capture technology around the globe [12].

An RFID system primarily comprises of RFID Tags, RFID Reader, Middleware and a Backend system. RFID Tags are uniquely and universally identified by an identification sequence, governed by the rubrics of EPC global Tag Data Standard 2. A tag can either be passively activated by an RFID reader or it can actively transmit RF signals to the reader [13]. The RFID reader, through its antenna, reads the information stored on these tags when it's in its vicinity. The reader, whose effective range is based on its operational frequency, is designed to operate at a certain frequency. The operational frequency of the reader ranges from 125 KHz – 2.4 GHz [14]. The Middleware encompasses all those components that are responsible for the transmission of germane information from the reader to the backend management systems [15]. The Middleware can include hardware components like cables and connectivity ports and software components like filters that monitor network performance of the system [16]. The Backend database stores individual tag identifiers to uniquely identify the roles of each tag. The database stores record entries pertaining to individual tags and its role in the system application. The RFID system is interdependent on its core components to achieve maximum efficiency and optimum performance of the application. Due to its high degree of flexibility, the system can be easily adopted for an array of applications ranging from small scale inventory cabinets to multifarious and highly agile supply chain management

systems [17, 18]. Although, the cost of incorporating this technology has restricted its outreach, the technology promises to have untapped potential.

RFID has been a widespread tool for both tracking the transit transports and for the public ticketing system. It's already been an outstanding achievement throughout the globe including big cities like London, Helsinki, Shanghai, Istanbul, Moscow, Porto and many more. The system can be implemented for subways, railways and public bus services for the sake of systematic operations in corresponding cases [19, 20].

2.2 Classification of RFID Technology

According to the frequency band in which individual RFID components and systems work on; **Low Frequency RFID** systems operating between 30 kHz to 500 kHz. They have mainly short-latency and low operating costs. **High Frequency RFID** systems operating between 850 MHz to 950 MHz and a frequency between 2.4GHz to 2.5 GHz. They offer a greater impact than low frequency RFID [21].

According to the type of the identification elements to; **Active RFID** systems are composed from active RFID elements. **Passive RFID** systems are made up of passive RFID elements that do not have built-in power source and receive energy from its activities from the magnetic field of the reader.

According to their applications; **Electronic Article Surveillance (EAS) system** at present they are probably the most using application of RFID in practice with their encounter almost every day. It is the system that is primarily used to protect traders and shops against theft. Readers are stationary and are located at the exit of the store. **Portable Data Capture (PDC) system** is a system for monitoring the movement of goods and people. The specific feature is the using of mobile RFID readers. **Network Systems** these are similar to the previous RFID group in terms of use. However, use EAS systems as well as stationary sensors, which scan the identification information from the RFID tags in the movement of goods or people nearby sensor and sent it to the central system. **Systems for determining position** are used RFID to identify the exact position of the object or the vehicle identification while passing a checkpoint [21].

2.3 Historic Development of RFID

The first RFID application was the Identification Friend or Foe (IFF) system [22] and it was used by the British in the Second World War. Transponders were placed into fighter planes and tanks, and reading units could query them to decide whether to attack. Successors of this technology are still used in armies around the world.

The first commercial RFID application was the Electronic Article Surveillance (EAS). It was developed in the 1970s as a theft prevention system. It was based on tags that can store a single bit. That bit was read when the customer left the store and the system would sound alarm when the bit was not unset. In the end of 1970s RFID tags made its way into the agriculture for example for animal tagging.

In the 1980s RFID technology got a boost when Norway and several US states decided to uses RFID for toll collection on roads. In addition to toll collection the following decade brought a vast number of new applications, such as ski pass, gasoline cards [22], money cards, etc.

In 1999 the Auto-ID Center at MIT was founded. Its task was to develop a global standard for item-level tagging. The Auto-ID was closed in 2003 after completing the work on the Electronic Product Code (EPC). At the same time the newly founded EPC global Inc. continues the work. The probably first paper related to RFID technology was the landmark paper by Harry Stockman, "Communication by Means of Reflected Power" in October 1948. The first patent on RFID was issued in 1973 for a passive radio transponder with memory [23].

2.4 Working Principle of RFID

An RFID system consists of two main components, a transponder or a tag which is located on the object that we want to be identified, and a transceiver or a reader.

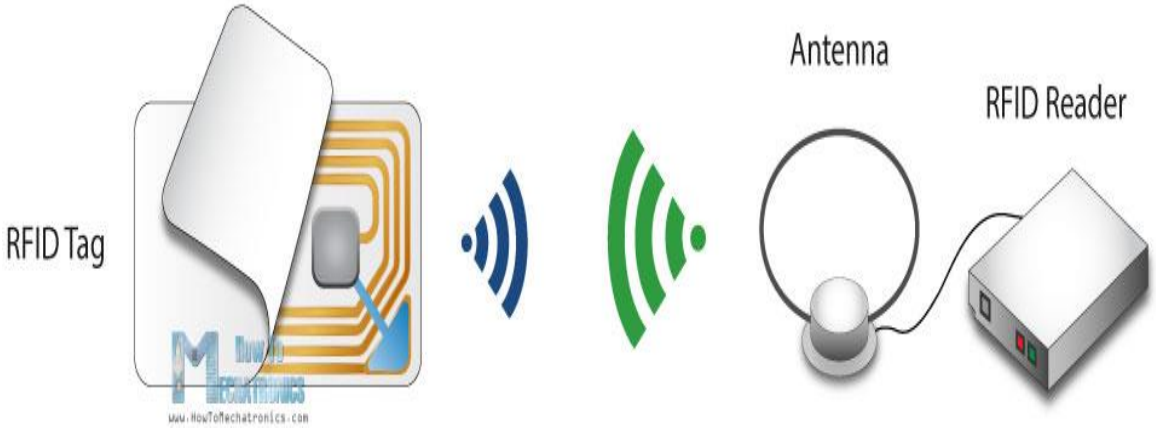

Figure 2-1: RFID Tag and Reader [24] Working principle of RFID [24]

The RFID reader consists of a radio frequency module, a control unit and an antenna coil which generates high frequency electromagnetic field. On the other hand, the tag is usually a passive component, which consists of just an antenna and an electronic microchip, so when it gets near the electromagnetic field of the transceiver, due to induction, a voltage is generated in its antenna coil and this voltage serves as power for the microchip [24].

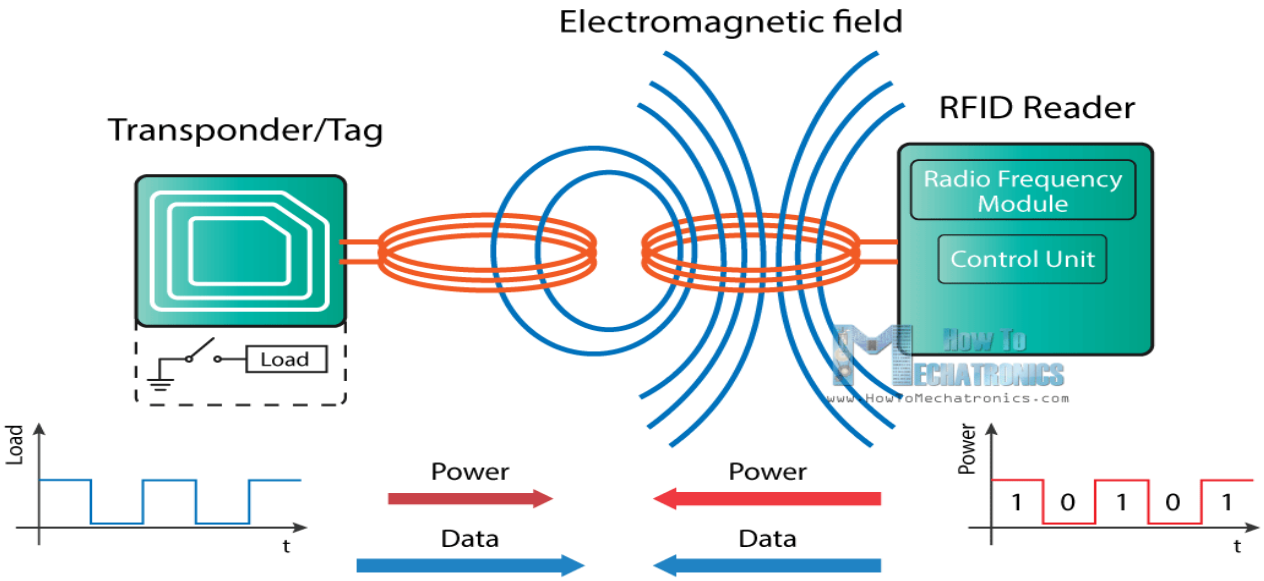

Figure 2-2: Working principle of RFID [24]

Now as the tag is powered it can extract the transmitted message from the reader and sending message back to the reader, it uses a technique called load manipulation. Switching on and off a load at the antenna of the tag will affect the power consumption of the reader's antenna which can be measured as voltage drop. These changes in the voltage will be captured as ones and zeros and that's the way the data is transferred from the tag to the reader [24]. There's also another way of data transfer between the reader and the tag, called backscattered coupling. In this case, the tag uses part of the received power for generating another electromagnetic field which will be picked up by the reader's antenna.

2.4.1 RFID Tags and Smart Labels

The RFID tag is composed of a protective material that holds the pieces together and shields them from various environmental conditions. The protective material depends on the application. For example, employee ID badges containing RFID tags are typically made from durable plastic, and the tag is embedded between the layers of plastic. RFID tags come in a variety of shapes and sizes and are either passive or active. Passive tags are the most widely used, as they are smaller and less expensive to implement. Passive tags must be "powered up" by the RFID reader before they can transmit data. Unlike passive tags, active RFID tags have an onboard power supply (e.g., a battery), thereby enabling them to transmit data at all times.

Smart labels differ from RFID tags in that they incorporate both RFID and barcode technologies. They are made of an adhesive label embedded with an RFID tag inlay, and they may also feature a barcode and/or other printed information. Smart labels can be encoded and printed on-demand using desktop label printers, whereas programming RFID tags are more time consuming and requires more advanced equipment.

2.5 Components of RFID System

An RFID system consists of various components that are connected to one another by a dedicated communication path. The individual components are integrated into the system to implement the benefits of RFID solution [26]. The lists of components are described as follow:

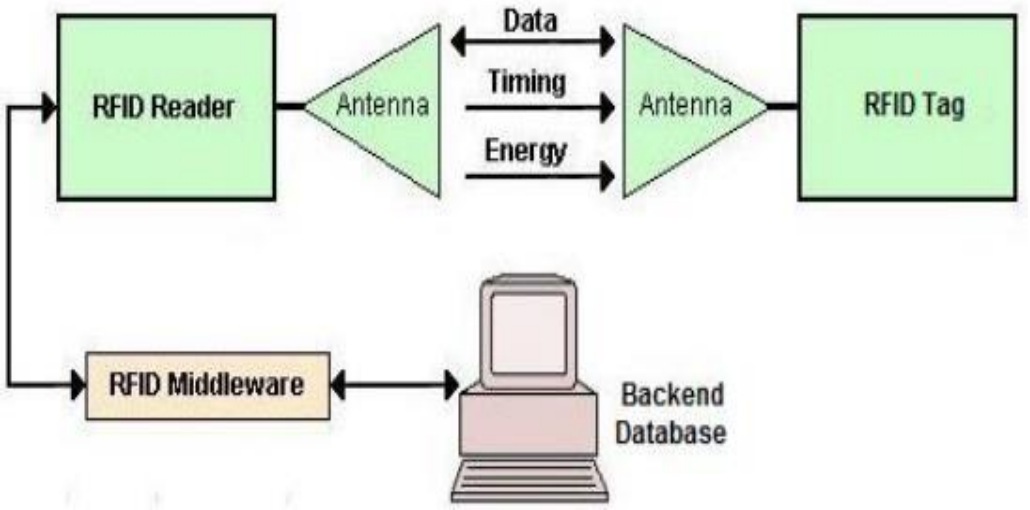

Figure 2-3: Components of RFID System [26]

2.5.1 RFID Tag

Tags - an object that is attached to any product and uses a unique sequence of characters to define it. A tag consists of a microchip that stores a unique sequence identifier that is useful in identifying objects individually. The sequence is a numeric serial, which is stored in the RFID memory. The microchip includes minute circuitry and an embedded silicon chip [15, 17]. The tag memory can be permanent or re-writable, which can be re-programmed electronically by the reader multiple times. Tags are designed specific to its applications and environment. For example, paper-thin tags are attached to books in a library management system [27].

Tags are available in various shapes and sizes. Tags that are initiated by the reader are known as Passive tags, whilst those that do not require external initiation are called Active tags. A Semi-Passive tag exists, which has the features of both Active and Passive tags [20]. Each tag type has its distinct characteristics. Tags are operable on Microwave

(2.4 – 2.5 GHz), Ultra High Frequency (UHF) (860 – 1500 MHz), High Frequency (HF) (13.56 MHz) and Low Frequency (LF) (125 kHz) [28].

Figure 2-4: Types of RFID Tags [28]

Table 2-1: Features of Types of Tags

Feature	Type of Tag		
	Passive	**Active**	**Semi – Passive**
Read Range	Short (up to 10m)	Long (up to 100m)	Long (up to 100m)
Battery	No	Yes	Yes
Lifespan	Up to 20 years	Between 5-10 years	Up to 10 years
Cost	Cheap	Very expensive	Expensive
Availability	Only in field of reader	Continuous	Only in field of reader
Storage	128 bytes read/ write	128 Kbytes read/ write	128 Kbytes read/ write
Application	EZ-Pass toll payment booths	Monitor the condition of fresh produce	Measurement of temperature periodically

2.5.2 Antenna

Antenna – it is responsible for the transmission of information between the reader and tag using radio waves. The antenna is medium through which the tag and reader communicate with each other. An antenna can activate a passive tag and transfer

data by emitting wireless impulses that has electromagnetic properties [19]. The antenna comes in various designs. They come in following types: (1) Stick antennas, (2) Di-pole or multi-pole antennas, (3) Beam-forming or phased array element antennas, (4) Circular polarized, (5) Gate antennas, (6) Patch antennas, (7) Linear polarized, (8) Adaptive antennas, and (9) Omni directional antennas [16].

2.5.3 Reader

Reader – a scanning device that uses the antenna to realize the tags that are in its vicinity. It transmits signals at a certain frequencies. The reader is the most fundamental part of the RFID system. It reads raw data from the tag and transmits it to the Middleware for further processing. The reader attempts to interrogate the tags at varying frequencies. The reader communicates by transmitting a beam of impulses, which encapsulate commands to the tag and listens for the tag's response [29]. The reader also contains built in anti-collision processes, which allows the reader to read multiple tags simultaneously. The reader is connected to the computer for data processing via a USB cable or over a wireless connection [26].

2.5.4 Middleware

Middleware – it is a communication interface to interpret and process data being fed by the readers into information. It takes into account all relevant ports of communication and a software application to represent this information. The middleware is an interface required to manage the flow of data from the reader and to transmit it efficiently to the backend database management systems. The middleware monitors the number of tags present in the system and extracts relevant information from the readers [27].

2.5.5 Backend Database

Backend database – a repository of information, which is designed specific to the application. The database stores records of data specific to individual tags. The backend database primarily deals with the storage of relevant information recorded by the reader and communicated by the middleware. For example, the middleware in an automated

security control system will store all tag readings taken by the reader in the database. This helps create log entries for the system [28].

2.6 Application of RFID Technology

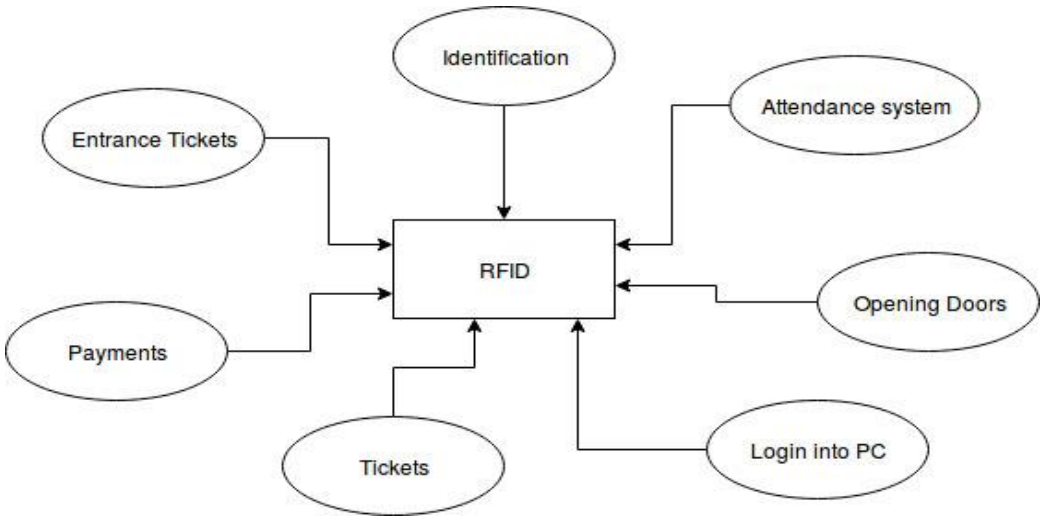

Figure 2-5: Application of RFID

2.7 Wireless Technology

Wireless technology is a technology that allows communication without using cables or wires. It is used to transfer data and other entities through wireless communication medium over very long distance. Wireless technology includes Radio Frequency (RF) and Infrared Frequency (IR) waves. Radio Frequency is the rate of oscillation of electromagnetic radio waves in the range of 3 kHz to 300 GHz, as well as the alternating currents carrying the radio signals. IF referred to simply as infrared, is a region of the electromagnetic radiation spectrum where wave lengths range from 700nm to 1 mm [37].

There are two main advantages of using wireless technology

Mobility: accessing a network without any physical contact freely, within home, business or even every city and maintain a connection to other components in the same network.

Cost Saving: installing cabling can be very expensive, especially in difficult terrain or established building. So it can save a considerable amount to use a wireless network instead, provided the environment is suitable.

2.8 Related Work

In this paper, the implementation of the RFID system in AALRT for improvement of fare collection system will be present. An RFID tag is an object that can be applied to or inserted into a product, person, or animal for identification and tracking using radio waves. Some identifiers can be read from several centimeters or meters away and beyond the line of sight of the reader.

Several related works exist in works, application of RFID Technology to different areas and specifically to the area of public transportation problem.

In paper [35,36] designed and employed a model of a secured and portable embedded reader system to read the biometric data from the electronic passport. The paper attempted to solve problems of trustworthiness, security and confidentiality in E-passports by authenticating holder online using Global System of Mobile Communications (GSM) network. The GSM network is the main edge between identification center and the e passport reader. The communication data is protected between server and e-passport reader by using AES to translate data for protection while transferring through GSM network.

Reference [30], Octopus cards are an electronic payment system based on a wireless RFID technology developed in Hong Kong. Users simply hold their con tactless smart cards over an electronic reader,and the payment is deducted from the card automatically.
If users link their cards to their credit card to upload money ,there is no cash transaction involved. Launched in 1997,Octopus cards are the world's most widely accepted contactless RFID electronic payment system. The system generates value for customers,service providers,and societies.
Octopus cards contain electronic microchips that store and update payment information and monetary value. When a customer uses it, the card is scanned by an Octopus reading processor,which deducts the transaction amount from the card. The operation mode is consistent in all payment transactions(Octopus,2010). Fig.2-4 illustrates the system

architecture of the Octopus system. The Octopus Clearing House(OCH) is the core component of the system that is responsible for transaction validations, revenue allocation, and fund transfers[30].

The front-end smart card processors are the reader machines that users interact with directly. Processors can communicate with the central computer or database in realtime, although certain readers store transaction data offline and send them through the network later.

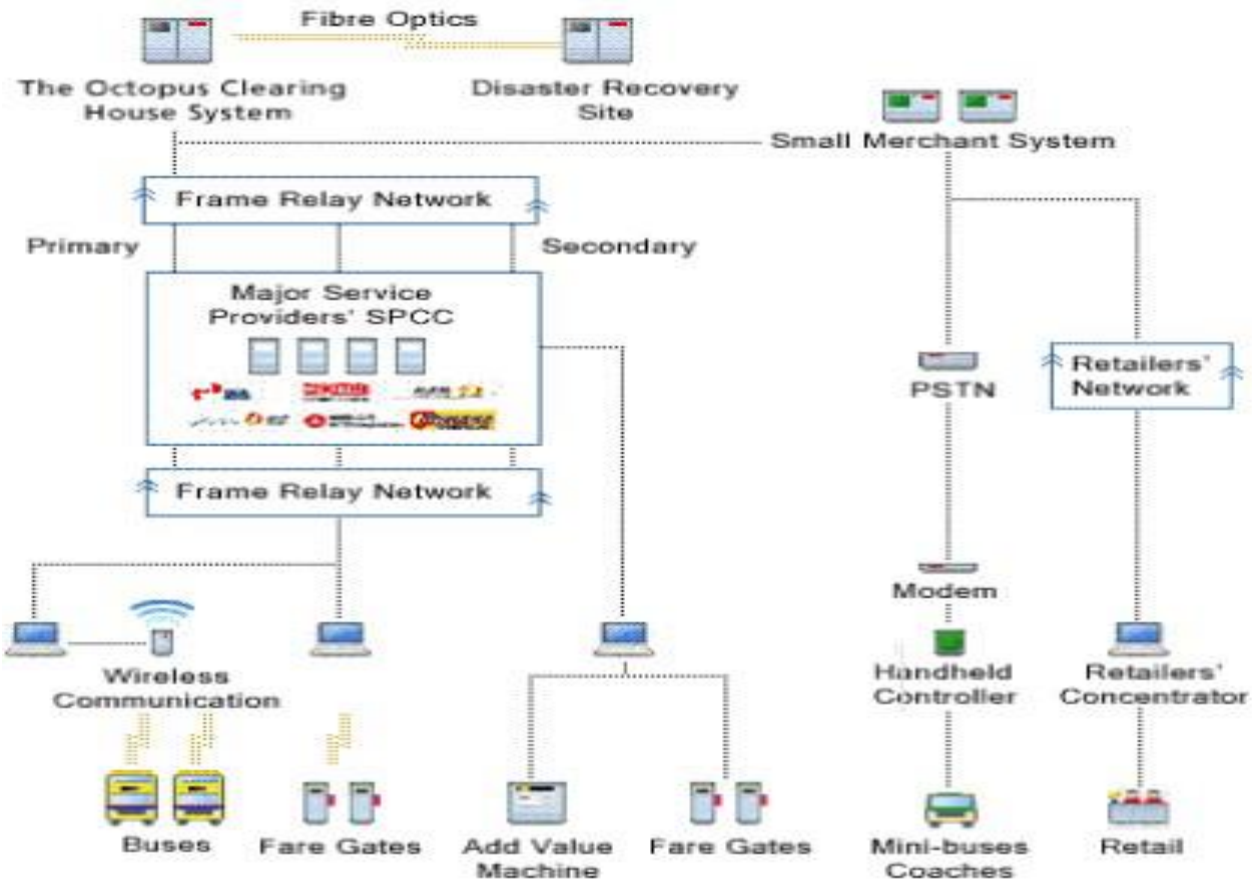

Figure 2-6: Internal Architecture of Octopus [30]

Oyster smart card for public transport in London is one of the largest and most complex smart- card enabled ticketing projects of its kind in the world. It was completed in 2004, almost six years after the contract for the Finance Initiative (PFI) contract was awarded to TranSys, a consortium of companies led by Electronic Data Systems (EDS) and Cubic Transportation Systems (CTS).

The Oyster card has a proximity range of about 80mm. The card operates as a RFID system. Oyster readers can also read other types of cards including Cubic Transportation

Systems' Go cards Oyster cards are credit card-sized plastic cards which are loaded with Philip's Mifare microchip technology. TranSys at first estimated that it would need to produce in the region of three million cards. However, the popularity of the Oyster with London Transport (LT) customers has proved so great that this initial estimate has had to be radically revised. Upwards of ten million Oyster cards have now been issued, while approximately five million are in regular use on the network.

In addition the paper talked about the working principle of Oyster as follow:
Oyster cards work through an aerial and a tiny microchip built into the (smart) cards themselves. This microchip acts as a data storage device, storing data about the ticket type purchased, card ownership and current credit. The aerial is used to send and receive data when a card is positioned in proximity to a card reader on a ticket machine or ticket gate. When this happens, a signal flows through the aerial and data flows from the card to the reader and back again. Communication is by radio signal and an individual transaction takes less than 0.20 of a second. Data can be read from the card by the card reader or can be sent to the card. Oyster cards do not have a battery. For power, they use a ‘built-in inductor’ that captures part of the incident radio-frequency interrogation signal from the card reader and employs it to charge the card for as long as it takes to conduct a transaction [31].
Paper [31] also describes the relationship between Oyster cards and the central Oyster database. Account data is held electronically on individual cards rather than on the central database. The central database does, however, keep a record of individual card use. It receives this data from ticket machines and ticket gates as customers navigate the transport system. The main database also holds data on credit and tickets bought online. Records of tickets or credit bought online are stored in the central database ready to be loaded onto a customer’s card when he or she passes through a pre selected barrier.
The Oyster card has a proximity range of about 80 mm (3 inches). The card operates as a RFID system. Oyster readers can also read other types of cards including Cubic Transportation Systems' Go cards Oyster cards are credit card-sized plastic cards which are loaded with Philip's Mifare microchip technology. TranSys at first estimated that it would need to produce in the region of three million cards. However, the popularity of the Oyster with London Transport (LT) customers has proved so great that this initial estimate has had to be radically revised. Upwards of ten million Oyster cards have now been issued, while approximately five million are in regular use on the network.

2.8.1 Comparing Barcode with RFID

Table 2-2: Comparing Barcode with RFID[32]

Barcode	RFID
Rely on the user to make contact to the reader, hence cannot be read from a distance	Do not require contact with reader, hence can be read from a distance.
In Barcode, only one card read at a time is allowed.	Multiple read at a time is permitted
Embedded information cannot be updated, Hence the restriction of the repeated overwriting if the embedded election information for each card	Embedded information can be updated; this allows the repeated over-writing if embedded electronic information for each card.
It does not allow for the increase technologies like surveillance cameras to be activated with an employee being in the vicinity.	RFID has increased technologies like surveillance cameras to be activated in conjunction with an employee being in their vicinity.
It is slower and requires time of sight to function.	RFID is faster and does not require line of sight.
It has lower data storage transponder.	It has higher data storage.
This transponder is bogus and cannot be incorporated in small items.	The transponder is miniaturized and can be incorporate in other items. For example, in 2009 researchers successfully glued RFID micro-transponder to live ants

CHAPTER 3
SYSTEM DESIGN ARCHITECTURE AND FUNCTIONS

3.1 Introduction

There are three structural layers used in the proposed system: the user interface layer, the application service layer and the Backend layer. System arch architecture as shown in Figure 3-1.

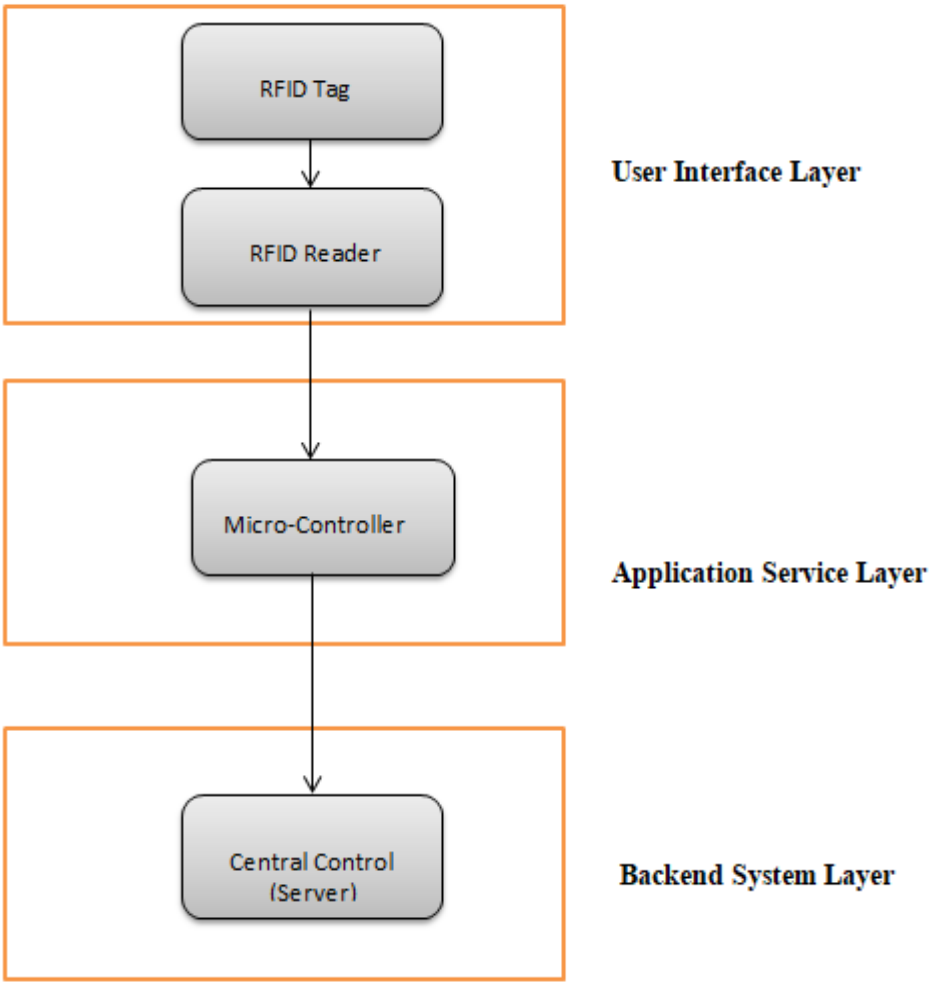

Figure 3-1: High Level Architecture of the System

3.1.1 User Interface Layer

This layer is used to communicate the passenger with the backend system, by providing user interface that handle users command. This layer includes Screens, keyboards and other tools. The user interface layer includes the RFID Tag and Reader. RFID tag consists of a microchip that stores a unique sequence identifier that is useful in identifying objects individually. It is used to store the passenger's information. Radio wave is electromagnetic radiation used in communication technologies, with a wavelength in the electromagnetic spectrum longer than infrared light. RFID reader is a static machine installed in every station. It receives information from the tag by radio wave and display the information on the screen at the same time it sends the information to the central control/server. It displays the provided services by the server to the user and vice versa. When the user selects a service, this program sends request to the server. When the server returns the processed result, this program shows it to the user.

3.1.2 Application Service Layer

The application service layer is the core of this three layer structure, the system functions and implementations are handled in this layer. In this layer, the system's logical model is encapsulated, the application service interfaces is provided for the user interface layer and the system modules between the function calls. Micro-Controller is computer hardware and software program and used to build digital devices and interactive objects that can sense and control objects in physical world. It consists of a physical programmable circuit board and an integrated development environment which is run on the computer and is used to write and upload computer code to the physical board.

Arduino is used between the user interface layer and the database layer by providing a lot of plugins for both layers. At the user interface level all the procedures/programs and hardwires will be communicated to each other. The RFID Tag's information will be compared/checked with the RFID Reader on the top Arduino. After checking the Tags/Cards Arduino is responsible to send the data to the server immediately.

3.1.3 Backend System layer

It is used to hold data, including user registration information, ticket ordering information, ticket information and all of the other information. This layer is responsible for calculate the fare based on the information received from the RFID Reader. The Backend layer also updates data in the database and RFID tag, according to the service request of the top layer. This is done using the connection of different software tools such as, XAMPP Server, Microsoft Visual Studio and others.

3.2 System Modelling/Process Flow and Flow Chart

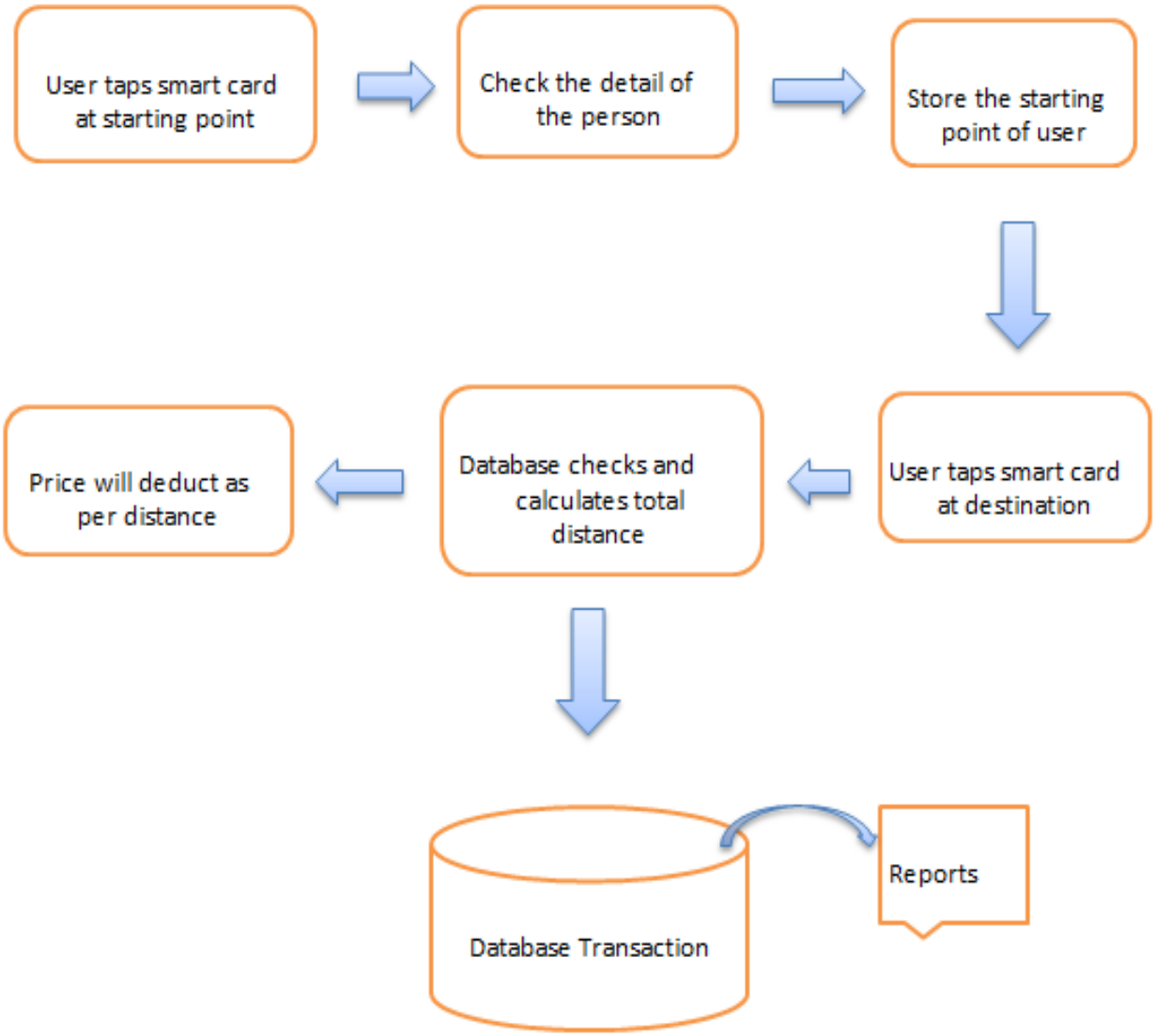

Figure 3-2: Automatic Fare Collection system Data Flow

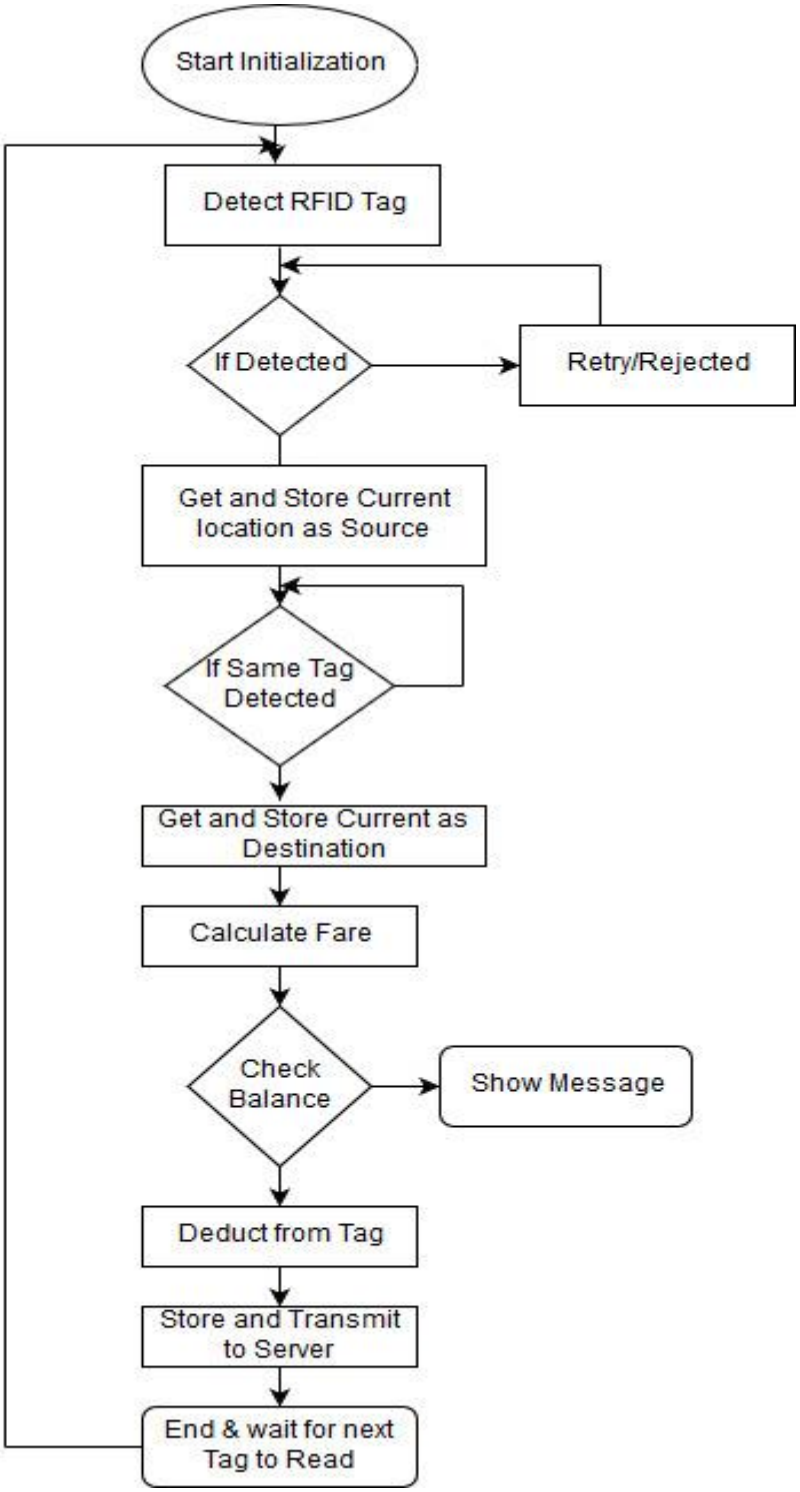

Figure 3-3: Flow Chart Algorithm of the Proposed System (Method 1)

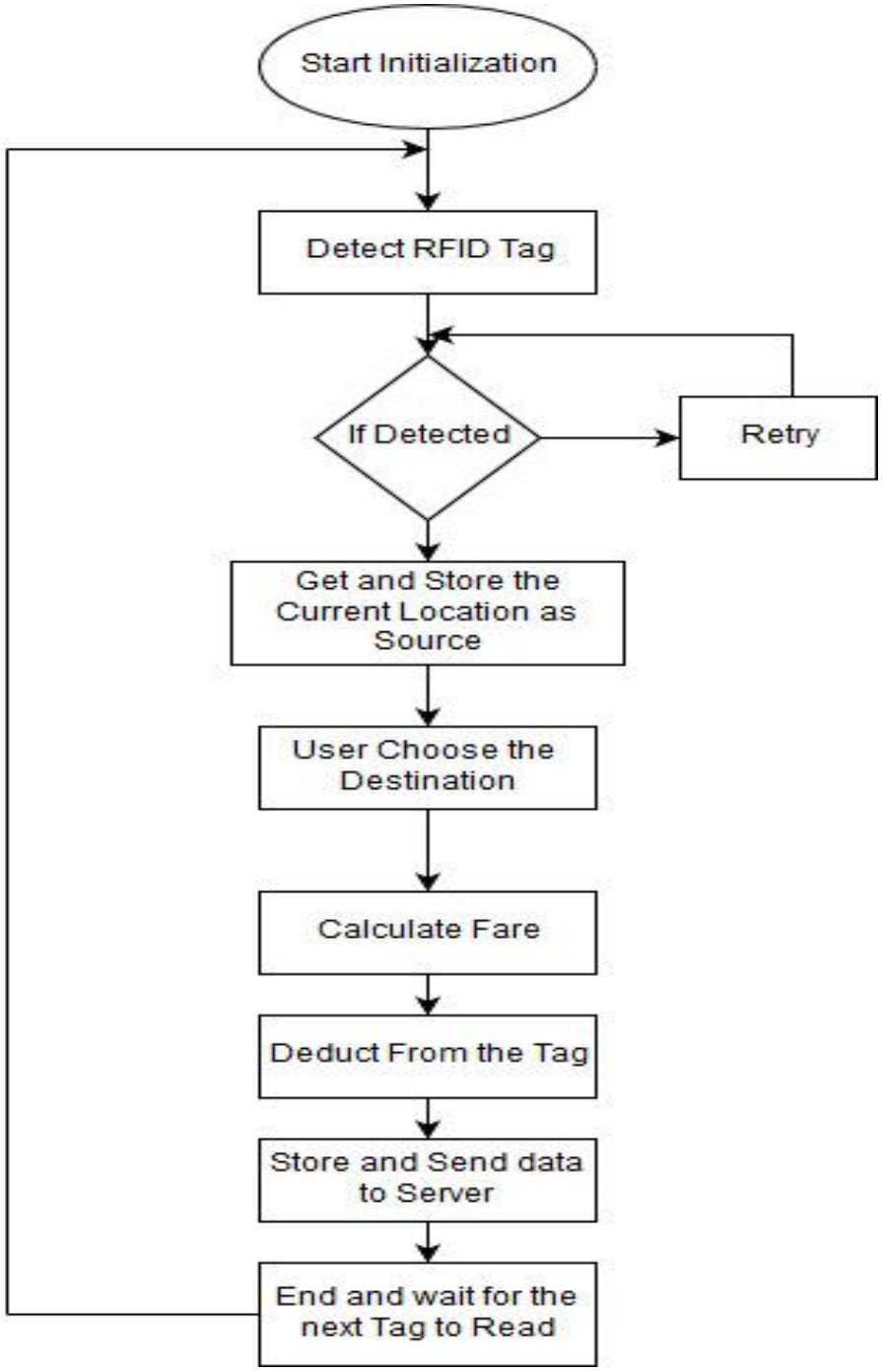

Figure 3-4: Flow Chart Algorithm of the Proposed System (Method 2)

3.2.1 Algorithm for the whole system

Step 1: Start

Step 2: Connect the hardware components

Step 3: Connect power supply

Step 4: Write the program using appropriate language

Step 5: Debug the code

Step 6: Run the module

Step 7: See the values and calculation was done or not

Step 8: See the amount is deducted or not from the person account

Step 9: See the values in web page (or server interface)

Step 10: Stop

Algorithm for RFID Tag; the RFID Tag is used to store passenger data/information. It is used for send passenger data to the RFID reader through radio wave. It is also used to receive and store data that comes from the backend system. **Algorithm for RFID Reader**; the main functions for RFID reader are receiving and send the passenger data/information to/form the backend system/RFID Tag through some middle layer applications (Microcontroller, internet….). Receive passenger data/info from the system and display to the user and at the same time store the data at the RFID tag.

Algorithm for Micro-Controller; the Arduino Uno IDE is used to receive data/info from RFID reader/backend system, send data/info to RFID Reader/backend system and middle layer application control. **Algorithm for Central Control/Backend System**; the backend system includes the MySQL database and the Visual Basic graphic user interface and performs a lot of activities. Received/send data from/to the middle layer application (Micro-Controller), update the Database based on the given input and decision processed by the system, calculate the distance covered by passenger and deduct

price from the DB, send the Report the final data/info to the RFID Tag to store the updated info and manage and control the overall transaction and communication of the system.

3.3 Hardware and Software Requirements

3.3.1 Micro-Controller

Arduino is open source computer hardware and Software Company and a user community that designs and manufactures microcontroller kits for building digital devices and interactive objects that can sense and control objects in physical world. It consists of a physical programmable circuit board and an integrated development environment which is run on the computer and is used to write and upload computer code to the physical board.

3.3.2 Liquid Crystal Display

Liquid Crystal Display (LCD) is the fundamental display technology used by monitors, televisions, tablets and smart-phones. They are thinner and lighter and draw much less power than cathode ray tubes (CRTs). LCD technology is used for displays in notebook and other smaller computers.

3.3.3 Power Supply

This unit will supply the various voltage requirements of each unit. This will be consists of transformer, rectifier, filter and regulator. The rectifier used here will be Bridge Rectifier. It will convert 230VAC into desired 5V/12/V DC.

3.3.4 Servo Motor

A servomotor is a rotary actuator that allows for precise control of angular position, velocity and acceleration. It consists of a suitable motor coupled to a sensor for position feedback. It also requires a relatively sophisticated controller, often a dedicated module designed specifically for use with servomotors. Door handling system is controlled by servomotor. For heavy loads, stepper motors can be used instead. Selection of motor depends on the weight of door, its operating mechanism etc.

3.3.5 RFID Reader and Tag

The Radio-frequency identification (RFID) is a wireless technology that uses low frequency radio signals ranging from 125 kHz to 960 MHz in order to transfer small bits of data between RFID devices. An RFID device consists of two fundamental components: tags and readers. The communication between the reader and the tag is achieved via the transmission of electromagnetic waves. A reader is used to magnetize the tag and decode the information from the tag. Tags store and process information stored in it. A reader emits radio frequency signal which interacts with the tag. This energizes the pin or bar code producing its own magnetic field with a unique interference pattern which corresponds to a unique number which is read by the tag [7].

3.4 Components and circuit schematics of the proposed system

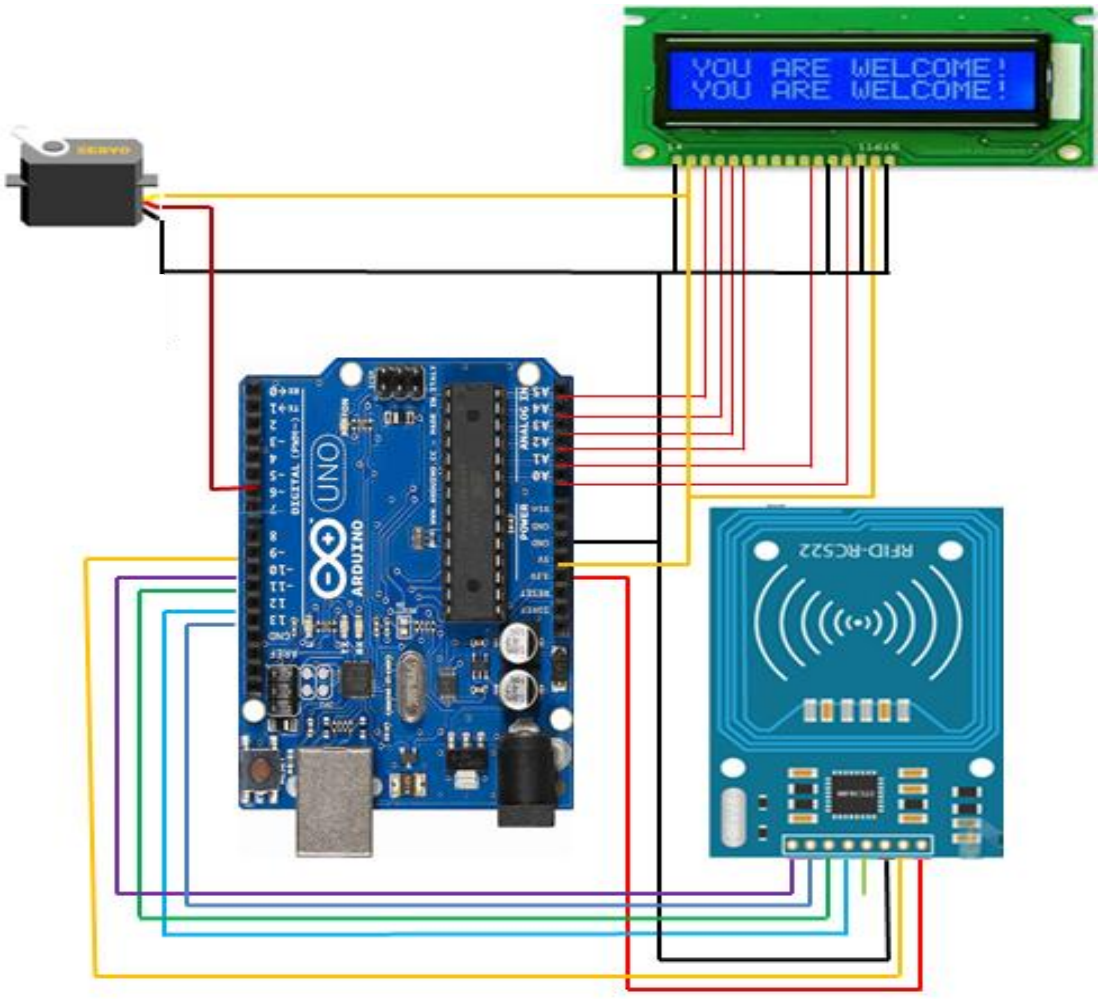

Figure 3-5: Circuit Schematics

3.4.1 RFID Reader Module and Arduino Uno Configuration

Table 3-1: RFID Module and Arduino Uno

RFID Module	Arduino Uno
3.3 v	Pin 3.3 V
RST	Pin 9
GND	GND
NC	No Connection
MISO	Pin 12
MOSI	Pin 11
SCK	Pin 13
SDA	Pin 10

3.4.2 LCD Configuration and Servo Motor with Arduino Uno

Table 3-2: Arduino and LCD Configuration

LCD	Arduino Uno
V_{SS}	GND
V_{CC}	5V
V_{EE}	GND
RS	Analog pin A0
R/W	GND

LCD	Arduino Uno
Enable	Analog pin A1
DB4	Analog pin A5
DB5	Analog pin A4
DB6	Analog pin A3
DB7	Analog pin A2
LED +	Power
LED -	GND
Servo Pin 1	Digital pin 8

3.5 Source Code

The source code for the designed program starts by include the libraries for the RFID module, the display and the servo motor, define some variables needed for the program below as well as create the instances of the libraries. All the program codes and description are stated in the Appendix A.

Some of the codes are shown below.

```
#include <SPI.h>
#include <MFRC522.h>
#include <LiquidCrystal.h>
#include <Servo.h>
#define SS_PIN 10
#define RST_PIN 9
MFRC522 mfrc522(SS_PIN, RST_PIN); // Create MFRC522 instance.
#define beep_pin 8
#define led_pin 6 //Red led
#define led_pin 5 //Green led
LiquidCrystal lcd(A0,A1,A5,A4,A3,A2);
Servo myServo; // Servo motor
void setup()
{
Serial.begin(9600); // Initiate a serial communication
myServo.attach(8); // Servo motor
myServo.write(0); // Initial lock position of the servo motor
pinMode(beep_pin,OUTPUT);
pinMode(6,OUTPUT);
pinMode(5,OUTPUT);
digitalWrite(beep_pin,LOW);
lcd.begin(16, 2);
lcd.setCursor(2, 0);
lcd.print("Welcome!!");
lcd.setCursor(0, 1);
lcd.print("Circuit is Ready");
delay(2000);
lcd.clear();
SPI.begin(); // Initiate SPI bus
mfrc522.PCD_Init(); // Initiate MFRC522
Serial.println("Put your card to the reader...");
Serial.println();
}
```

3.6 The Backend System Hardware and Software Requirement

The backend system of this thesis includes some the most popular software tools. The software development unit consists of XAMPP Control Panel, Microsoft Visual Studio 13 and Arduino Uno IDE.

3.6.1 The Backend System Hardware and Software Requirement

3.6.1.1 Database Conceptual Design

In the database conceptual design phase, the entity - relationship diagram can be used to establish the data model to form UML class model independent of the machine and BMS. UML class diagram provides entities (data object), properties, and associated methods, in order to describing the conceptual model of the real world. In UML class diagrams, a class (similar to an entity type in ER) is displayed as a box that includes three sections: The top section gives the class name (similar to entity type name); the middle section includes the attributes; and the last section includes operations that can be applied to individual objects (similar to individual entities in an entity set) of the class. Operations are not specified in ER diagrams. Relationship types are called associations in UML terminology, and relationship instances are called links. A binary association (binary relationship type) is represented as a line connecting the participating classes (entity types), and may option-ally have a name. A relationship attribute, called a link attribute, is placed in a box that is connected to the association's line by a dashed line [32].

UML Class Diagrams

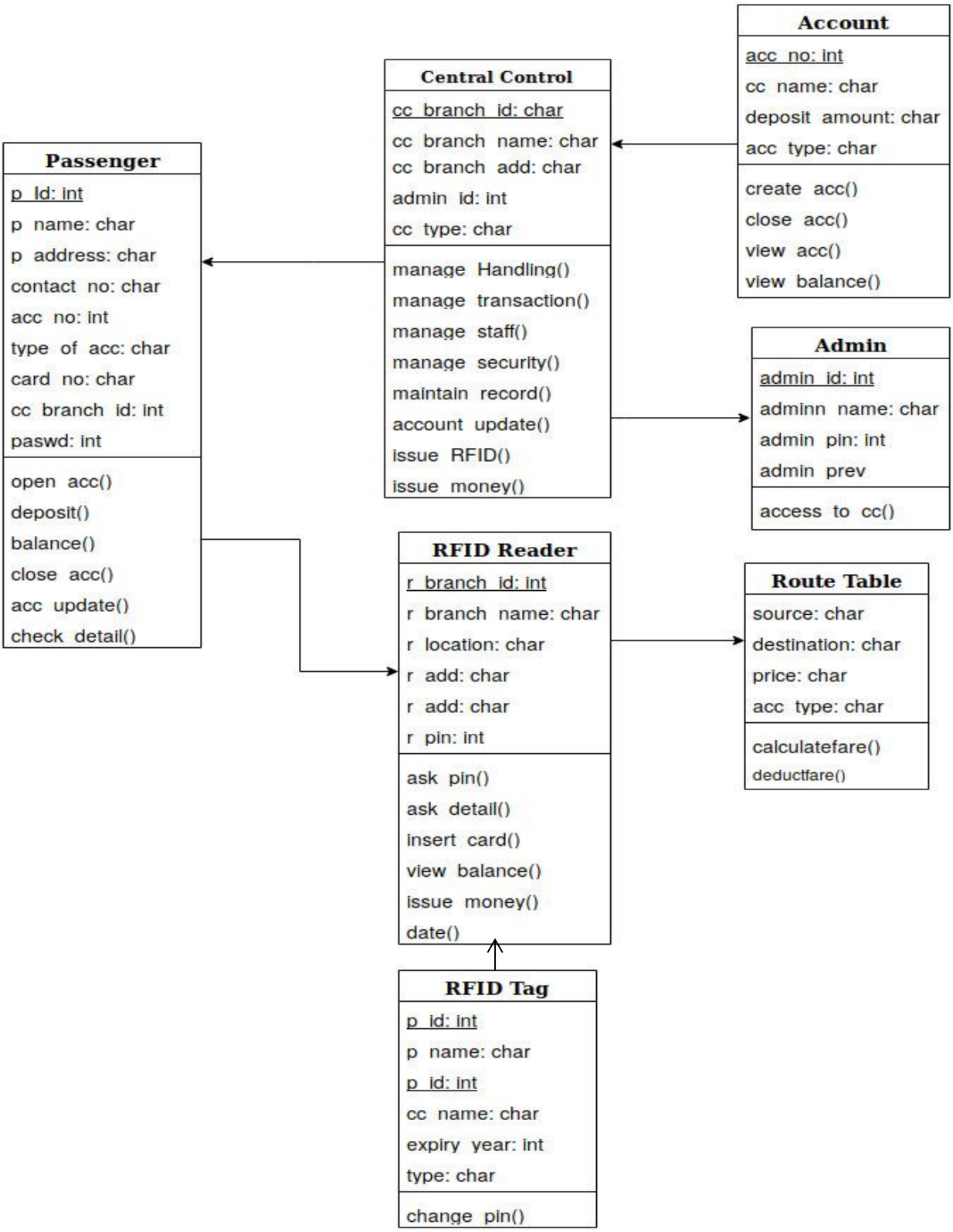

Figure 3-6: The AFC System conceptual schema in UML class diagram notation

3.6.1.2 Logical Model Design

Logic model design task is to convert the basic ER diagram of the conceptual models to logical structure of the consistent data model which was supported by DBMS products. In this paper, the user view method is used to standard tables. All the keywords of tables are listed. The contact and constraint relation is description by data collection chart. The design result of the user view is summarized. All of user view tables are composed to a complex database system.

3.6.1.3 Physical Design of the Database

The physical structure of the database mainly refers to record format, record organization and record access methods. Obviously, the physical design of the database entirely dependent on a given hardware environment and database products. In relational model system, the physical design is relatively simple because the file format is a single record type file which contains only index mechanism, space size, block size, etc.

3.6.1.4 XAMPP Control Panel

This software is used in this project to create a visual server where the activities of each passenger is being stored. There is one database designed using XAMPP Control Panel. The database is named "rifd_based_afc_system". The "rifd_based_afc_system" database contains the overall system information, user information and all transactions. This information are used by the administration or passenger for identifying and manipulating fare collection system information of each passenger. Besides that, there are seven tables inside the "rifd_based_afc_system" database which act as temporary data storage which are "CENTRAL CONTROL", "PASSENGER", "RFID READER", "RFID TAG", "ACCOUNT", "ROUTE TABLE" and "ADMINISTRATION". These tables are essential to the program flow. Figures are the pictures of the "rfid".

a) TABLE I CENTRAL CONTROL

Table 3-3: Central Control

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	cc_branch_id	int(20)			No	*None*	central control branch id	AUTO_INCREMENT
2	cc_branch_name	varchar(20)	latin1_swedish_ci		No	*None*	central control name	
3	cc_branch_add	varchar(20)	latin1_swedish_ci		No	*None*	central control address	
4	admin_id	int(20)			No	*None*	administrator id	
5	cc_type	varchar(20)	latin1_swedish_ci		No	*None*	central control type	

b) TABLE II PASSENGER

Table 3-4: Passenger Information

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	p_id	int(10)			No	*None*	passenger ID	AUTO_INCREMENT
2	p_name	varchar(20)	latin1_swedish_ci		No	*None*	passenger name	
3	p_add	varchar(20)	latin1_swedish_ci		No	*None*	passenger address	
4	contact_no	varchar(20)	latin1_swedish_ci		No	*None*	passenger contact	
5	acc_no	int(20)			No	*None*	passenegr account no	
6	type_of_accc	varchar(20)	latin1_swedish_ci		No	*None*	type of account	
7	card_no	varchar(20)	latin1_swedish_ci		No	*None*	Tag UID	
8	acc_branch_id	int(10)			No	*None*	central control	
9	passwd	int(10)			No	*None*	passenger password	

Other tables are attached at Appendex B.

3.6.2 Microsoft Visual Studio 13

Visual Studio is one of the leading technologies to design user friendly graphical user interface. The user interface includes the passenger registration form, connection creation with the XAMPP server to access the DB, Passenger information display and the travel information. Figure is the flow chart for the login transaction.

3.6.3 Communication between the front-end and back-end system (Server)

Generally, Automatic Fare Collection System (AFC) is one of basic station equipment that consists of automatic gate machine and backend Control machine. In this application, stable and integrated platforms are necessarily to keep passenger flow run smoothly at peak hours; at the same time, all data will be gathered and transmitted into center.

3.6.3.1 Industrial Ethernet Switches for AFC Systems

To enhance system performance and reliability, industrial-grade switches with reliable quality are perfect for fare collection systems. This new generation Industrial Ethernet Switch creates a bridge between devices and network, its Ethernet ports, which support high-speed data transfers (100 Mbps for EKI-3525 and EKI-3528/ Gigabit for EKI-3725 and EKI-3728) for each port, are responsible for connecting the main board, hard disk and diverse devices in fare equipment to collect more data while sending back to control center. The robust features of EKI-3000 series includes dual power input (8.4~52.4 VDC) and wide operating temperature range (-10~60°C) are dedicated to operating in

areas of unstable power and rugged environments therefore ensure a reliable and uninterrupted operation for a fare collection system [33].

3.6.3.2 Ethernet Shield used to connect the Arduino to the internet

The Ethernet Shield allows an Arduino board to connect to the internet. It is based on the Wiznet W5100 ethernet chip. The Wiznet W5100 provides a network (IP) stack capable of both TCP and UDP [34].

The m2m protocol of the overall connection is based on the HTTP protocol. The communication between backend system and the station machines are always connected. A station machine sends HTTP post request to the server with the information in m2m format included in the request content as a parameter value. The server processes the request according to the m2m command and return message to the station machine. If there is any command for the station machine, like remote control command from browser, it will be brought to it in the HTTP response. Before a station machine makes an m2m communication, it must login to the server as a user. Otherwise, its HTTP request will be ignored. Because the W5100 ethernet chip only supports TCP, the HTTP post request based on TCP must be fulfilled in Uno board. The HTTP response from the server is parsed to extract session cookie. That is to find "Set-Cookie:"string and get "JSESSIONID=xxxxxxxx" substring, then the xxxxxxxx is the session cookie. The session cookie must be appended in the following HTTP post request to get user identity.

3.6.3.3 Serial Port and MySQL Connector

The MySQL connector is a new technology made for permitting to connect Microsoft Visual Studio to a XAMPP server without using an intermediate computer or a web based service. Permitting a direct access to a database server means store data acquired from different projects as well as check values stored in tables on the server and keep the network local to facilities including having a network that isn't connected to the internet or any other network [35].

The Connector allows issue queries to the database server in much the same manner as through the MySQL client application. It provides functions such as insert, delete, update data and call functions, create objects, etc. the connector reads one packet-at-a-time and since the Arduino has a limited data size, the combined length of all fields must be less than available memory. However, most projects are those that need to store data and in that case the only memory requirements are those for the SQL statements [36].

CHAPTER 4
RESULT ANALYSIS AND DISCUSSION

4.1 Introduction

In the previous chapter, the theoretical study of an electronic automatic fare collection system based on RFID has been presented. The study leads to choose some of the important the hardware and software components. In this chapter, the final simulation and analysis result will be presented. In addition, the hardware and software implementation of the overall study result will be presented.

4.2 Simulation of Arduino and RFID

The model simulated in this paper is an Automatic fare collection system using RFID and it includes one entry gate (reader station machine) and two RFID tags. When the tag taps to the station machine (reader), the system detects the tag and then the gate is opened and the passenger passed.

The required hardware components are shown below in figure 4-1 and 4-2, including outside and inside view of the station machine.

Figure 4-1: The outside view of the proposed AFC system hardware

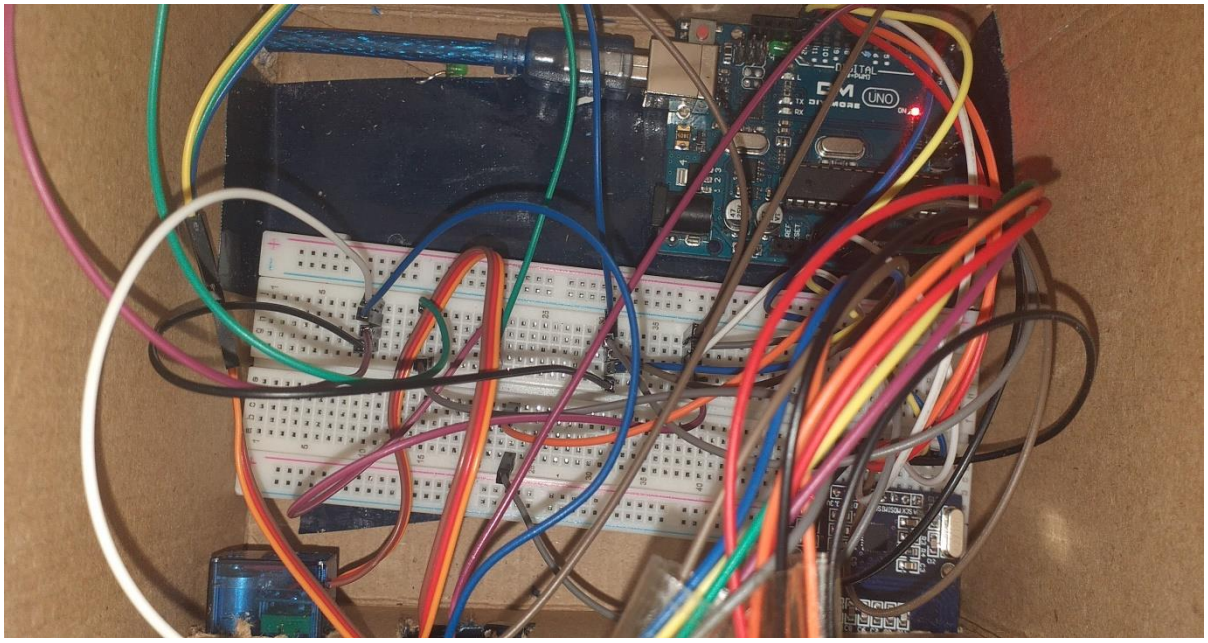

Figure 4-2: The inside view of the proposed AFC system hardware

The working principle of the AFC system and the interaction with the passenger has shown in the figure 4-3 and 4-4. The first figure shows the initial position of the station machine and the light at the gate is red. The second figure shows that when the passenger taps the tag to the station machine the gate will be opened and the light at the gate changes to green and the passenger enter. Then the station machine automatically close the gate and will waiting for another passenger.

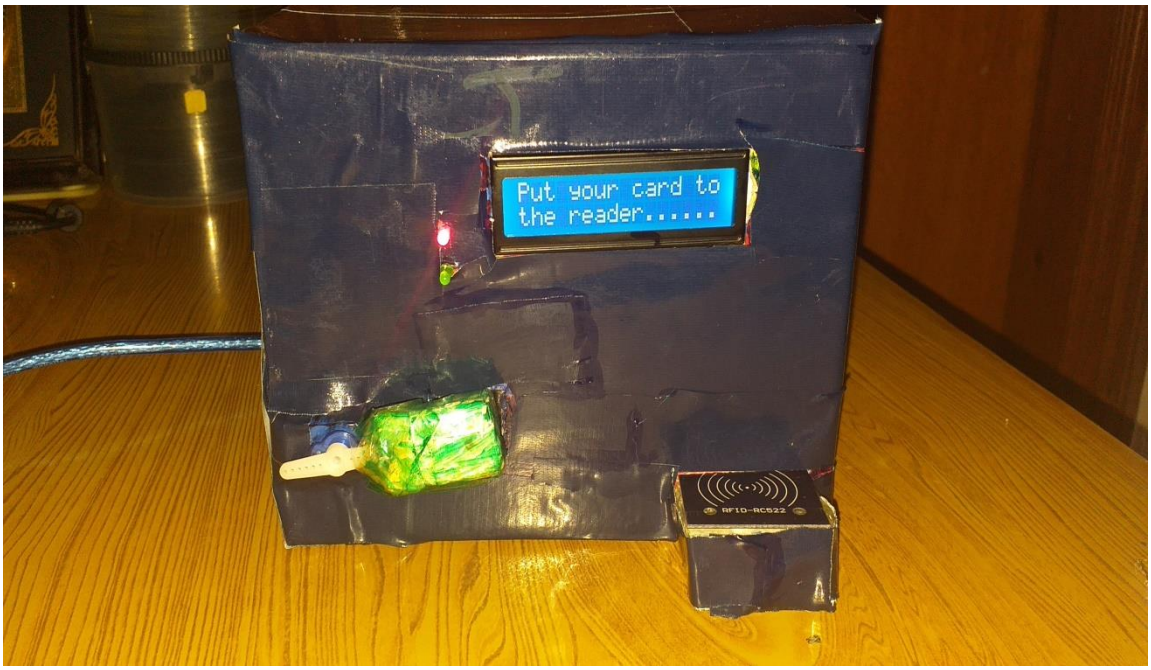

Figure 4-3: The starting position of the station machine

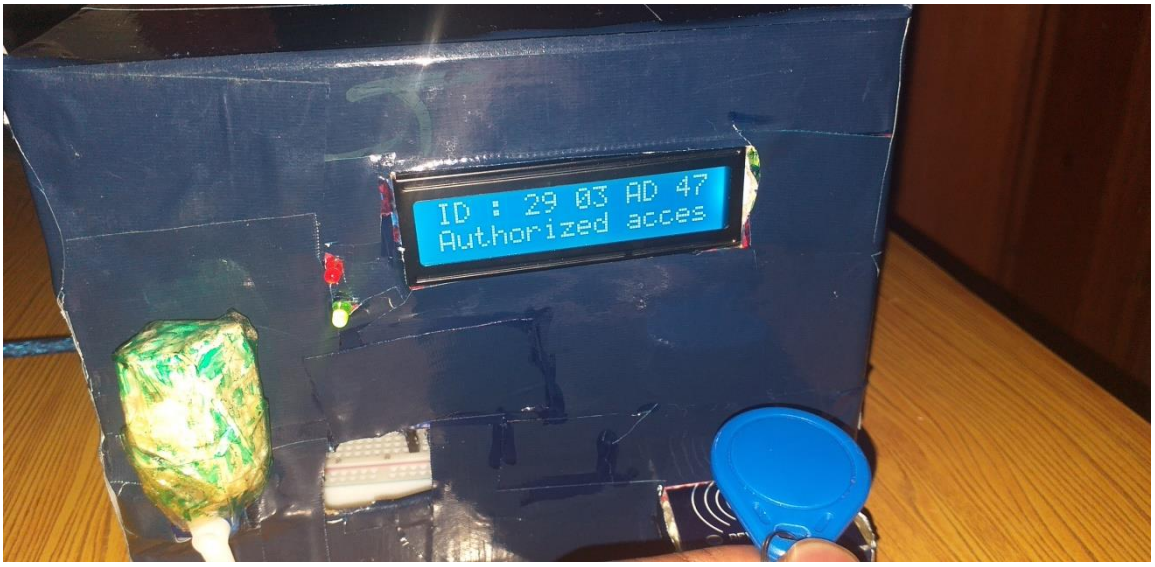

Figure 4-4: After the passenger taps his/her tag to the station machine

4.3 RFID Based Automatic Fare Collection System Management

The main form of the proposed system was built by using Visual Basic programming language. This form contains four windows (UI for connection creation with Arduino, Display passenger information and registration of new passenger). The first form shown in Fig.4-5 is used to connect the Microsoft Visual Studio to Arduino Uno IDE. After providing the Serial port and Baud Rate, click the button connect.

Figure 4-5: An interface for connection creation with Arduino Uno serial port

The second form show in the fig [4-6] displays the passenger information.

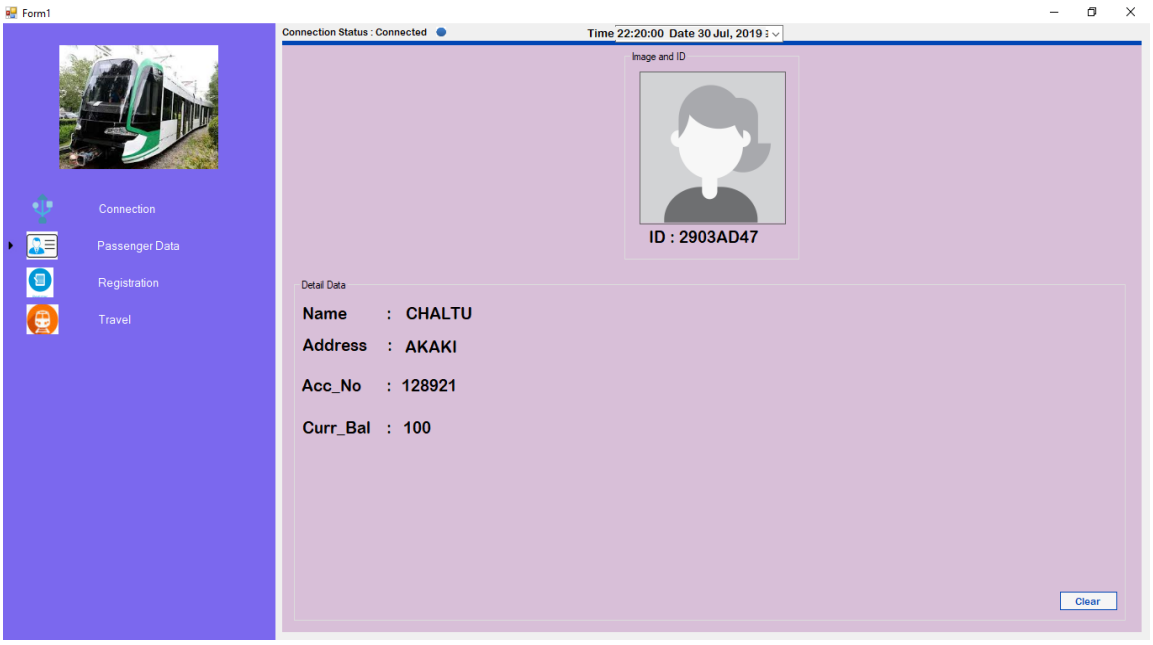

Figure 4-6: An interface to display passenger information

The Third form, which is shown in fig [4-7] is used to add new passenger comes to register to the system with their RFID tag.

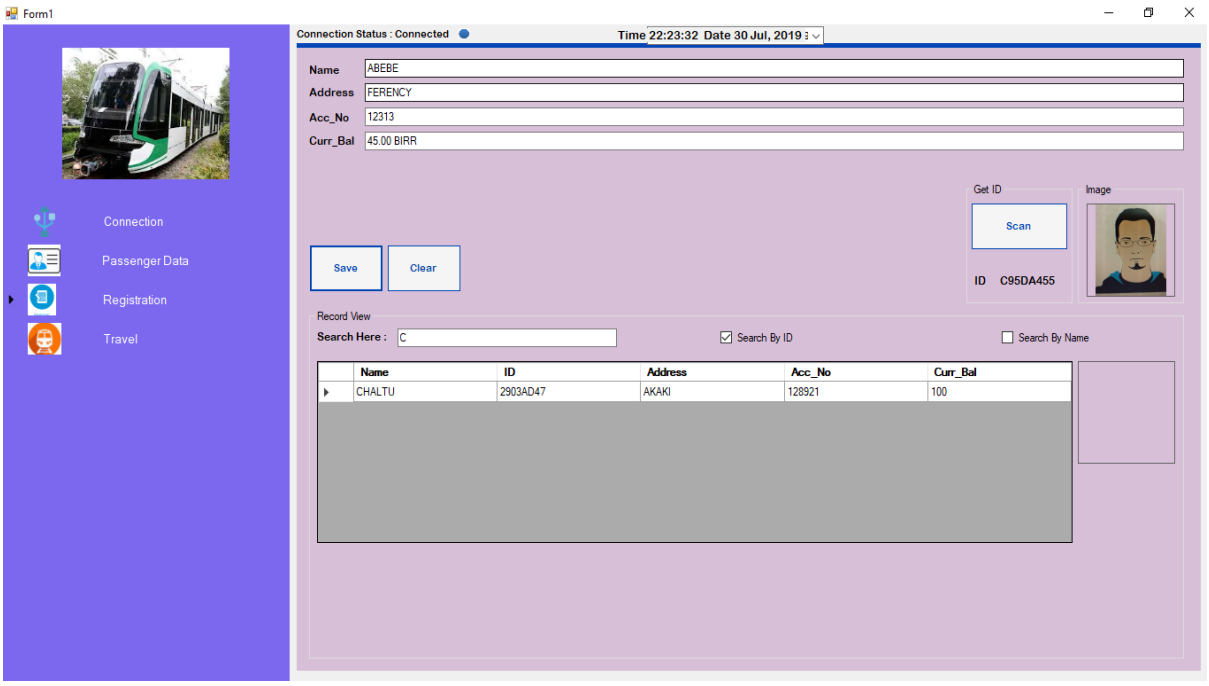

Figure 4-7: An interface for passenger registration and searching from DB

The last user interface shown below fig [4-8] is used for passenger at the station machine to choose the start and end stops. After providing the start and end stop information such as the cost, distance, and route information will be displayed to the passenger and the passenger pay the price and then the data are stored in the database.

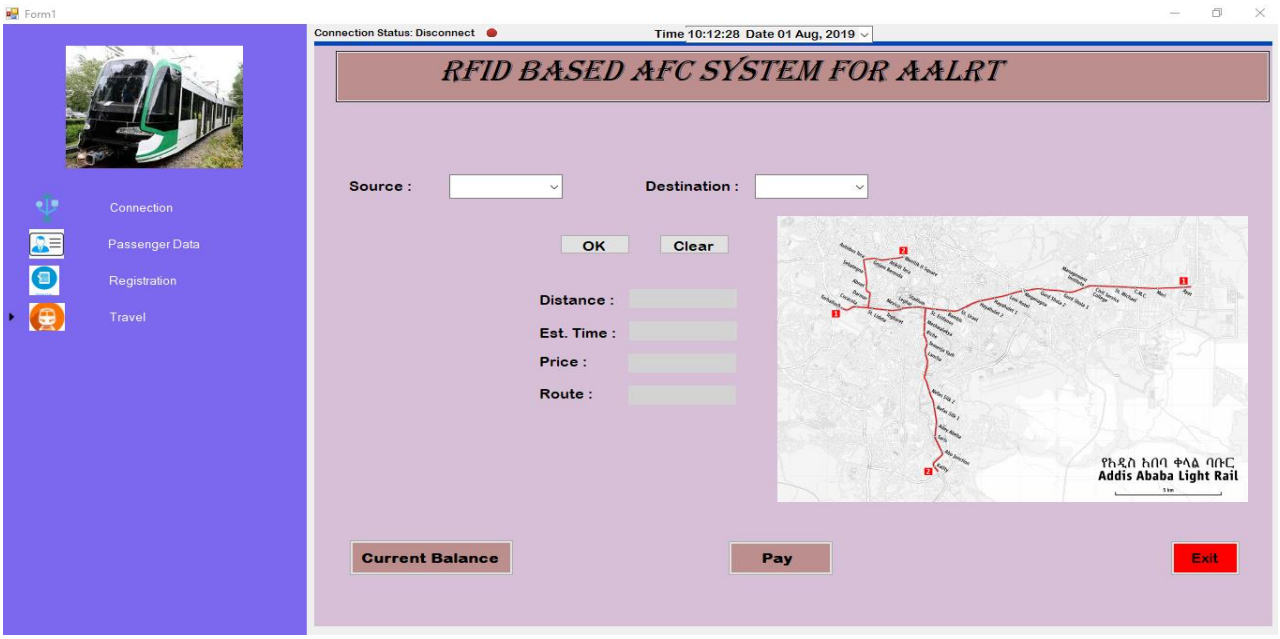

Figure 4-8: An interface for the passenger

4.4 Analysis Result

The developed AFC system is fully automated, reliable, transparent and convenient to the user. The cards are reusable and more convenient when they are compared to the paper based ticketing system. Controlling the entrance gate with the use of RFID tags and reader are made possible.

The central control is used to control the overall transactions, updates and other administration issues. Also, it is used to register new passengers and store their current information. The payment mechanism is secure and it avoids theft and money laundering. In addition, the overall system will be control by the central control and operate with few staffs, and then the company's overall revenue can increases.

The developed system helps the passenger to use railway transportation in easy way. Passengers have a registered RFID tag and use this card in every station to travel by the train. The RFID tag stores the passengers' basic information and interacts with the

central system to update the stored data and display the current status. The passengers will wait short time to process their operation and it saves time when it compared with the old system. In addition, the passengers do not expected to have tickets and wait on the queue to get the ticket. As a result, the passengers can be satisfied by the service and choose this mode of transportation frequently.

4.5 Discussion

This section discusses and shows the future research possibilities that can address the following research questions:

a) How to study about different AFC implementation trends?
b) How to design RFID based AFC system?
c) How to design a suitable system that is convenient to passenger?
d) How to develop the overall platform of the study?

One of the major finding is the hardware implementation of the proposed system. The hardware part is built using Arduino Uno IDE, RFID module and other tools. The combination of those tools make the system works as expected. C program is used in Arduino Uno IDE. Serial port communication is used to integrate the hardware part with other backend system. But this is not the only way to communicate the system and other communication mechanisms are stated in the previous chapter.

By using RFID tag, the passengers are allowed to use railway transportation. Figs [4-3 and 4-4] show the interaction of a passenger and station machine at station. The backend system will check passenger's current balance, calculate the fare and reduce the fare form passenger tag. Then updated information will be stored in the both the tag and backend system and gate will be opened.

The backend system is designed using Microsoft Visual studio and XAMP Server. Figs [4-6, 4-7 &4-8] show the implementation of the proposed system. An administrator uses the designed use interface to register new passenger and control the database. They have an access to control the day to day overall transactions done in every station. The backend system is a standalone application.

The results of this study would agree with the objective of the proposed system including the following scenarios.

Scenario1: The developed system can be installed inside the entry and exit gate of the train. At the entry gate the passenger taps their tag and the system store their current information and this gate is used only for entry of passengers. At the exit gate, the passengers do the same thing as before. The two systems interact each other internally and compare the information and the then system calculates the fare. The all station machines are connected internally.

Scenario 2: The developed system can be installed outside the train. It will be installed in every station. The procedure is the same as scenario 1, at this stage the machine is called station machine and it is static.

CHAPTER 5
CONCLUSSION AND RECOMMENDATIONS

5.1 Conclusion

The main objective of this paper was to study and design an automated fare collection system for AALRT using RFID technology. In this regard, this thesis work has achieved the following things. Thus are,

- Studying about different implementation mechanism of AFC and RFID technology.
- Based on the effectiveness and efficiency of RFID technology on AFC system, the overall design platform of the proposed AFC system is built.
- A prototype design of the AFC system using RFID technology is developed and tested successfully.
- The proposed system has included an interactive user interface for both the front and back end of the system. The passenger and system administrator can easily interact with the AFC system using the developed user interface.

The RFID based automatic fare collection system is an easy, time saving and convenient way of gaining access to public transport system; it reduces the number of traveler who are going without ticket, increases customer satisfaction and it increases the revenue for the transportation company.

Furthermore, in terms of performance and efficiency, this paper has provided a convenient method of fare collection system compared to the traditional method of ticketing system. By using databases, the data is more organized. This system is also a user-friendly system as data manipulation and retrieval can be done via the interface, making it a universal fare collection system. Thus, it can also be implemented in other mode of transportation systems. For example, in Anbessa bus, Sheger bus, Public bus and other public transportations.

5.2 Recommendations

Innovations that are characterized by good engineering design may have some limitations. In this regard, the result of this thesis work may have some limitations which will be improved in the future by considering the advancement of technology. Further research is recommended to increase the reliability and effectiveness of the recommended system. Thus are;

- An IP camera can be integrated into this system to monitor the actions like buddy-punching wherein a person cheats by scanning for another person ID card.
- A GSM can be incorporated into the system for sending messages to passenger to confirm their current balance and other travel information.
- The Ultra-High Frequency (UHF) active RFID tags (that is 3m to 9ft) can be used rather than High Frequency (HF) active RFID tags (that is 1m to 3ft) for better performance and flexibility of users.
- The communication between the hardware and the software components of the overall system must be done in very effective and efficient way, the developed system done locally and use only serial port and MySQL connector.

5.3 Future Work

It is now clear that smart cards will play a significant role in public transport ticketing system for years to come. In this paper, the main aim is to design RFID based AFC system. In addition, the prototype of the design system is built. But this is not enough to deploy fully in AA-LRT system. It needs more research and implementations at high level.

The research shown that it is possible to upgrade this manual based ticketing system to automatic fare collection system. This will be achieved if the organizations especially, transportation sectors gave a special concern/attention to develop this system.

REFERENCES

[1] António A. Nunes, Teresa Galvão Dias, and João Falcão e Cunha, "Passenger Journey Destination Estimation From Automated Fare Collection System Data Using Spatial Validation", IEEE transactions on intelligent transportation systems.

[2] M. P. Pelletier, M. Trpanier, C. Morency, "Smart Card Data Use in Public Transit: A Literature Review", in *Transportation Research Part C: Emerging Technologies*, Vol. 19, No. 4, Pp. 557-568, Aug. 2011.

[3] K. Finkenzeller, "RFID Handbook: Fundamentals and Applications in contactless Smart Cards Radio Frequency Identification and Near-Field Communication" , Chichester, United Kingdom: John Wiley & Sons Ltd., 2010.

[4] "Identification Cards - Contactless Integrated Circuit Cards - Proximity Cards - Part 2: Radio Frequency Power and Signal Interface", ISO/IEC 14443–2:2010.

[5] Brumerčík, F. & Danko, "R. Transport application of hybrid simulation. Communications", EDIS, 2014. Vol. 16. No. 2. P. 20–24.

[6] B LYTHE P. T, "Smart Card Applications in Transport", International Smart Card Industry Guide, Smart Card News, United Kingdom, 1994.

[7] "Identification cards-contactless integrated circuits(s) cards-proximity cards", international standards organization ISO, Geneva, ISO 14443.

[8] "Identification cards-contactless integrated circuits(s) cards-vicinity cards", international standards organization, Geneva, ISO 14444.

[9] L. Sandip, "RFID Sourcebook", IBM Press, USA, ISBN: 0-13-185137-3, 2005.

[10] "Information Security: Radio Frequency Identification Technology in the Federal Government", US. Government Accountability Office, Report to Congressional Requesters, GAO-05-55, 2005.

[11] Gnoni, M.G, Rollo, A, Tundo, P, "A smart model for urban ticketing based on RFID applications", Industrial Engineering and Engineering Management, IEEE

International Conference on DOI: 10.1109/IEEM.2009.5373004 Publication Year: 2009 , Page(s): 2353 – 2357, 2009.

[12] E. Zeisel & R. Sabella, "*RFID+*", Exam Cram, ISBN: 0-7897-3504-0, 2006.

[13] "Additional Guidance and Security Controls are needed over Systems using RFID and DHS", US Department of Homeland Security, Department of Homeland Security, OIG-06-53, 2006.

[14] "Information Security: Radio Frequency Identification Technology in the Federal Government", US. Government Accountability Office, Report to Congressional Requesters, GAO-05-551, 2005.

[15] Juels, A., Weis, S.A., "Authenticating Pervasive Devices with Human Protocols", Advances in Cryptology – Crypto '05. Lecture Notes in Computer Science, Volume 3621, pp 293-308, 2005.

[16] S. Shepard, "RFID Radio Frequency Identification", USA, ISBN: 0-07-144299-5, 2005.

[17] "Enhanced Security Controls needed for US-Visit's System using RFID Technology", US Department of Homeland Security, Department of Homeland Security, OIG-06-39, 2006.

[18] K. Ahsan, H. Shah, P. Kingston, "Role of Enterprise Architecture in healthcare IT", Proceeding ITNG2009, IEEE, 2009.

[19] Roy, W., "An Introduction to RFID Technology Pervasive Computing and Communications", IEEE Press, pp 25-33, 2006.

[20] "Identification cards-contactless integrated circuit(s) cards-Vicinity cards", International Organization for Standardization (ISO), ISO/IEC 14443, 2003.

[21] Maria Grazia GNONI, Alessandra ROLLO, Pier Giuseppe TUNDO, "A smart model for urban ticketing based on RFID applications", IEEM09-P-0572, IEEE International Conference on Industrial Engineering and Engineering Management (IEEM), 2009.

[22] "The invention of IFF in WWII" http://www.vectorsite.net/twiz1.html

[23] "The probably first RFID patent" http://patft.uspto.gov/netacgi

[24] "How RFID Works and How To Make an Arduino based RFID Door Lock" https://howtomechatronics.com/tutorials/arduino/rfid-works-make-arduino-based-rfid-door-lock/

[25] Jaroslav Masek, Peter Kolarovszki,"Application of RFID technology in railway transport services and logistics chains", 9th International Scientific Conference Transbaltica, 2015.

[26] Krane. J,"Benetton clothing to carry tiny tracking transmitters", Associated Press, 2009.

[27] R.Want, "Enabling Ubiquitous Sensing with RFID", Computer, vol. 37, no. 4, pp.84–86, 2004.

[28] L. Srivastava, "RFID: Technology Applications and Policy Implications", Presentation, International Telecommunication Union, Kenya, 2005.

[29] "RFID Journal Gillette Confirms RFID Purchase RFID Journal", http://www.rfidjournal.com/article/articleview/258/1/1/, 2003.

[30] W.Sariken, "Automatic Fare Collection System for Public Transport in Jabalpur City", India, 2008.

[31] "London Travelcard v Oyster Card v Contactless Card", https://www.londontoolkit.com/briefing/travelcard_oyster.htm

[32] Ramez Elmasri, Shamkant B. Navath, "Fundamentals of database systems", 6th Edition, pp,226-227,2011.

[33] "Automatic Fare collection System (AFC) for Railway transportation", https://www.advantech.com/intelligent-transportation/afc

[34] Wu-Jeng Li, Shu-Chu Tung, ShihMiao Huang, "Server-side Ladder Logic of a Web-based Supervisory Control System", The First International Conference on engineering and Technology Innovation, Taiwan, (2011) November 11-15.

[35] Ononiwu G., Chiagozie, Okorafor G. Nwaji. "Radio Frequency Identification (RFID) Based Attendance System with Automatic Door Unit", in *Academic Research*, International, ISSNL-L: 2223-9553. 2012;2(2).

[36] "Evaluation of an active RFID system for fast tag collection", Computer Communications, 2008.

[37] Carr, Joseph J., "Microwave and Wireless Communications Technology". Boston: Butterworth-Heinemann, 1997.Gralla, Preston. "How Wireless Works". Indianapolis, IN: Que, 2002.

APPENDIX A

Arduino program for RFID Reader and Tag

```
#include <SPI.h>
#include <MFRC522.h>
#include <LiquidCrystal.h>
#include <Servo.h>
#define SS_PIN 10
#define RST_PIN 9
MFRC522 mfrc522(SS_PIN, RST_PIN);   // Create MFRC522 instance.
#define beep_pin 8
#define led_pin 6 //Red led
#define led_pin 5  //Green led
LiquidCrystal lcd(A0,A1,A5,A4,A3,A2);
Servo myServo; // Servo motor
void setup()
{
  Serial.begin(9600);   // Initiate a serial communication
  myServo.attach(8);  // Servo motor
  myServo.write(0); // Initial lock position of the servo motor
  pinMode(beep_pin,OUTPUT);
  pinMode(6,OUTPUT);
  pinMode(5,OUTPUT);
  digitalWrite(beep_pin,LOW);
  lcd.begin(16, 2);
  lcd.setCursor(2, 0);
  lcd.print("Welcome!!");
  lcd.setCursor(0, 1);
  lcd.print("Circuit is Ready");
  delay(2000);
  lcd.clear();
  SPI.begin();      // Initiate  SPI bus
  mfrc522.PCD_Init();   // Initiate MFRC522
  Serial.println("Put your card to the reader...");
  Serial.println();
}
void loop()
{
  digitalWrite(beep_pin, LOW);
  digitalWrite(6, HIGH);
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Put your card to");
  lcd.setCursor(0, 1);
  lcd.print("the reader.....");
  delay(2000);
  lcd.clear();
```

```
// Look for new cards
if ( ! mfrc522.PICC_IsNewCardPresent())
{
  return;
}
// Select one of the cards
if ( ! mfrc522.PICC_ReadCardSerial())
{
  return;
}
//Show UID on serial monitor
Serial.print("UID tag :");
String content= "";
byte letter;
for (byte i = 0; i < mfrc522.uid.size; i++)
{
   Serial.print(mfrc522.uid.uidByte[i] < 0x10 ? " 0" : " ");
   Serial.print(mfrc522.uid.uidByte[i], HEX);
   content.concat(String(mfrc522.uid.uidByte[i] < 0x10 ? " 0" : " "));
   content.concat(String(mfrc522.uid.uidByte[i], HEX));
}
Serial.println();
Serial.print("Message : ");
Serial.println("User Name: ");
Serial.println("User ID: ");
content.toUpperCase();

if (content.substring(1) == "C9 5D A4 55") //change here the UID of the
card/cards that you want to give access
{
  digitalWrite(6,LOW);
  digitalWrite(beep_pin,HIGH);
  digitalWrite(5,HIGH);
  delay(200);
  digitalWrite(beep_pin,LOW);
  delay(100);
  lcd.setCursor(0, 0);
  lcd.print("ID : ");
  lcd.print(content.substring(1));
  lcd.setCursor(0, 1);
  lcd.print("Authorized access");
  delay(2000);
  Serial.println("Authorized access");
  Serial.println("User Name: Abebe Kebede");
  Serial.println("User ID: ID_1267162716");
  Serial.println("Current Balance: 5.00 Birr");
  myServo.write(90); // Unlocks the door
  delay(4800);
  myServo.write(0);//lock the door
  delay(1000);
```

```
  digitalWrite(5,LOW);
  digitalWrite(6,HIGH);
 }
 else if (content.substring(1) == "29 03 AD 47")
 {
  digitalWrite(6,LOW);
  myServo.write(0);
  digitalWrite(beep_pin,HIGH);
  digitalWrite(5,HIGH);
  delay(200);
  digitalWrite(beep_pin,LOW);
  delay(100);
  lcd.setCursor(0, 0);
  lcd.print("ID : ");
  lcd.print(content.substring(1));
  lcd.setCursor(0, 1);
  lcd.print("Authorized access");
  delay(2000);
  Serial.println("Authorized access");
  Serial.println("User Name: Kebede Kebede");
  Serial.println("User ID: ID_12671633333");
  Serial.println("Current Balance: 100.00 Birr");
  myServo.write(90); // Unlocks the door
  delay(4800);
  myServo.write(0);
  delay(1000);
  digitalWrite(5,LOW);
  digitalWrite(6,HIGH);
 }

 else   {
  digitalWrite(beep_pin,HIGH);
  lcd.setCursor(0, 0);
  lcd.print("ID : ");
  lcd.print(content.substring(1));
  lcd.setCursor(0, 1);
  lcd.print("Access denied");
  Serial.println(" Access denied");
  delay(1500);
   }
}
```

Arduino program for Passenger Registration

```
#include <SPI.h>

#include <MFRC522.h>

#include <LiquidCrystal.h>

#define SS_PIN 10
```

```
#define RST_PIN 9
MFRC522 mfrc522(SS_PIN, RST_PIN);  // Create MFRC522 instance.
LiquidCrystal lcd1(A0,A1,A5,A4,A3,A2);
int readsuccess;
byte readcard[4];
char str[32] = "";
String StrUID;
 void setup() {
  Serial.begin(9600); // Initialize serial communications with the PC
  SPI.begin();      // Init SPI bus
  mfrc522.PCD_Init(); // Init MFRC522 card
  lcd1.begin(16, 2);
  lcd1.setCursor(2, 0);
  lcd1.print("Welcome!!");
  lcd1.setCursor(0, 1);
  lcd1.print("Circuit is Ready");
}
void loop() {
  readsuccess = getid();
   if(readsuccess){
    Serial.println(StrUID);
    delay(1000);
  }
}
int getid(){
  if(!mfrc522.PICC_IsNewCardPresent()){
    return 0;
```

```
  }
  if(!mfrc522.PICC_ReadCardSerial()){
    return 0;
  }
  for(int i=0;i<4;i++){
    readcard[i]=mfrc522.uid.uidByte[i]; //storing the UID of the tag in readcard
    array_to_string(readcard, 4, str);
    StrUID = str;
  }
  mfrc522.PICC_HaltA();
  return 1;
}
void array_to_string(byte array[], unsigned int len, char buffer[])
{
    for (unsigned int i = 0; i < len; i++)
    {
        byte nib1 = (array[i] >> 4) & 0x0F;
        byte nib2 = (array[i] >> 0) & 0x0F;
        buffer[i*2+0] = nib1  < 0xA ? '0' + nib1  : 'A' + nib1  - 0xA;
        buffer[i*2+1] = nib2  < 0xA ? '0' + nib2  : 'A' + nib2  - 0xA;
    }
    buffer[len*2] = '\0';
}
```

APPENDIX B

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	r_branch_id	int(20)			No	None	RFID Reader ID	AUTO_INCREMENT
2	r_branch_name	varchar(20)	latin1_swedish_ci		No	None	RFID Reader name	
3	r_location	varchar(20)	latin1_swedish_ci		No	None	RFID Reader location	
4	r_add	varchar(20)	latin1_swedish_ci		No	None	RFID Reader address	
5	r_pin	int(20)			No	None	RFID Reader pin	

Table 5-1: RFID Reader

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	Rfid_tag_id	int(20)			No	None	FRID Tag ID	AUTO_INCREMENT
2	p_id	int(20)			No	None	passenegr ID	
3	p_name	varchar(20)	latin1_swedish_ci		No	None	passenegr name	
4	cc_name	varchar(20)	latin1_swedish_ci		No	None	central control name	
5	expiry_year	varchar(10)	latin1_swedish_ci		No	None	exipry year of tag	
6	type_of_tag	varchar(20)	latin1_swedish_ci		No	None	type of tag	

Table 5-2: RFID Tag

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	acc_no	int(20)			No	None	account number	AUTO_INCREMENT
2	cc_name	varchar(20)	latin1_swedish_ci		No	None	central control name	
3	deposit_amount	varchar(10)	latin1_swedish_ci		No	None	deposit amount	
4	acc_type	varchar(10)	latin1_swedish_ci		No	None	type of account	

Table 5-3: Account

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	route_id	int(10)			No	None	route Id	AUTO_INCREMENT
2	source	varchar(20)	latin1_swedish_ci		No	None	source of passenger	
3	destination	varchar(20)	latin1_swedish_ci		No	None	destiantion of passenegr	
4	price	varchar(10)	latin1_swedish_ci		No	None	price of the route	
5	acc_type	varchar(10)	latin1_swedish_ci		No	None	type of account	

Table 5-4: Route Table

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	admin_id	int(20)			No	None	administrator ID	AUTO_INCREMENT
2	admin_name	varchar(20)	latin1_swedish_ci		No	None	administrator name	
3	admin_pin	int(10)			No	None	administrator pin code	
4	admin_prev	varchar(10)	latin1_swedish_ci		No	None	administrator privilege	

Table 5-5: Administration

APPENDIX C

Visual Studio Programs in Visual Basics

```
Imports MySql.Data.MySqlClient
Imports System
Imports System.IO.Ports
Public Class Form1
    Dim Connection As New MySqlConnection("server=localhost;user
id=root;database=dbprofile")
    Dim MySQLCMD As New MySqlCommand
    Dim MySQLDA As New MySqlDataAdapter
    Dim DT As New DataTable
    Dim Table_Name As String = "passenger" 'your table name
    Dim Data As Integer
    Dim LoadImagesStr As Boolean = False
    Dim IDRam As String
    Dim IMG_FileNameInput As String
    Dim StatusInput As String = "Save"
    Dim SqlCmdSearchstr As String
    Public Shared StrSerialIn As String
    Dim GetID As Boolean = False
    Dim ViewUserData As Boolean = False

    Private Sub Form1_Load_1(sender As Object, e As EventArgs) Handles
MyBase.Load
        Me.CenterToScreen()
        PanelConnection.Visible = True
        PanelUserData.Visible = False
        PanelRegistrationandEditUserData.Visible = False
        PanelTravel1.Visible = False
        ComboBoxBaudRate.SelectedIndex = 3
        ComboBoxSource.Items.Add("Kality")
        ComboBoxSource.Items.Add("Adey Abeba")
        ComboBoxSource.Items.Add("Nefas Silk 1")
        ComboBoxSource.Items.Add("St. Lideta")
        ComboBoxSource.Items.Add("Menelik II Square")
        ComboBoxSource.Items.Add("Staduim")
        ComboBoxDestination.Items.Add("Staduim")
        ComboBoxDestination.Items.Add("Mexico")
        ComboBoxDestination.Items.Add("Menelik II Square")
        ComboBoxDestination.Items.Add("Kality")
      End Sub
    Private Sub ShowData()
        Try
            Connection.Open()
        Catch ex As Exception
```

```
            MessageBox.Show("Connection failed !!!" & vbCrLf & "Please check that the
server is ready !!!", "Error Message", MessageBoxButtons.OK, MessageBoxIcon.Error)
            Return
        End Try

        Try
            If LoadImagesStr = False Then
                MySQLCMD.CommandType = CommandType.Text
                MySQLCMD.CommandText = "SELECT Name, ID, Address, Acc_No,
Curr_Bal FROM " & Table_Name & " ORDER BY Name"
                MySQLDA = New MySqlDataAdapter(MySQLCMD.CommandText,
Connection)
                DT = New DataTable
                Data = MySQLDA.Fill(DT)
                If Data > 0 Then
                    DataGridView1.DataSource = Nothing
                    DataGridView1.DataSource = DT
                    DataGridView1.Columns(2).DefaultCellStyle.Format = "c"
                    DataGridView1.DefaultCellStyle.ForeColor = Color.Black
                    DataGridView1.ClearSelection()
                Else
                    DataGridView1.DataSource = DT
                End If
            Else
                MySQLCMD.CommandType = CommandType.Text
                MySQLCMD.CommandText = "SELECT Images FROM " & Table_Name &
" WHERE ID LIKE '" & IDRam & "'"
                MySQLDA = New MySqlDataAdapter(MySQLCMD.CommandText,
Connection)
                DT = New DataTable
                Data = MySQLDA.Fill(DT)
                If Data > 0 Then
                    Dim ImgArray() As Byte = DT.Rows(0).Item("Images")
                    Dim lmgStr As New System.IO.MemoryStream(ImgArray)
                    PictureBoxImagePreview.Image = Image.FromStream(lmgStr)
                    PictureBoxImagePreview.SizeMode = PictureBoxSizeMode.Zoom
                    lmgStr.Close()
                End If
                LoadImagesStr = False
            End If
        Catch ex As Exception
            MsgBox("Failed to load Database !!!" & vbCr & ex.Message,
MsgBoxStyle.Critical, "Error Message")
            Connection.Close()
            Return
        End Try
        DT = Nothing
        Connection.Close()
    End Sub
    Private Sub ShowDataUser()
```

```
        Try
            Connection.Open()
        Catch ex As Exception
            MessageBox.Show("Connection failed !!!" & vbCrLf & "Please check that the
server is ready !!!", "Error Message", MessageBoxButtons.OK, MessageBoxIcon.Error)
            Return
        End Try
        Try
            MySQLCMD.CommandType = CommandType.Text
            MySQLCMD.CommandText = "SELECT * FROM " & Table_Name & "
WHERE ID LIKE '" & LabelID.Text.Substring(5, LabelID.Text.Length - 5) & "'"
            MySQLDA = New MySqlDataAdapter(MySQLCMD.CommandText,
Connection)
            DT = New DataTable
            Data = MySQLDA.Fill(DT)
            If Data > 0 Then
                Dim ImgArray() As Byte = DT.Rows(0).Item("Images")
                Dim lmgStr As New System.IO.MemoryStream(ImgArray)
                PictureBoxUserImage.Image = Image.FromStream(lmgStr)
                lmgStr.Close()

                LabelID.Text = "ID : " & DT.Rows(0).Item("ID")
                LabelName.Text = DT.Rows(0).Item("Name")
                LabelAddress.Text = DT.Rows(0).Item("Address")
                LabelAccNo.Text = DT.Rows(0).Item("Acc_No")
                LabelCurrBal.Text = DT.Rows(0).Item("Curr_Bal")
            Else
                MsgBox("ID not found !!!" & vbCr & "Please register your ID.",
MsgBoxStyle.Information, "Information Message")
            End If
        Catch ex As Exception
            MsgBox("Failed to load Database !!!" & vbCr & ex.Message,
MsgBoxStyle.Critical, "Error Message")
            Connection.Close()
            Return
        End Try
        DT = Nothing
        Connection.Close()
    End Sub
    Private Sub ClearInputUpdateData()
        TextBoxName.Text = ""
        LabelGetID.Text = "________"
        TextBoxAddress.Text = ""
        TextBoxAccNo.Text = ""
        TextBoxCurrBal.Text = ""
        PictureBoxImageInput.Image = My.Resources.click
    End Sub
    Private Sub ButtonConnection_Click_1(sender As Object, e As EventArgs) Handles
ButtonConnection.Click
```

```
        PictureBoxSelect.Top = ButtonConnection.Top
        PanelConnection.Visible = True
        PanelUserData.Visible = False
        PanelRegistrationandEditUserData.Visible = False
        PanelTravel1.Visible = False
    End Sub
    Private Sub ButtonUserData_Click_1(sender As Object, e As EventArgs) Handles
ButtonUserData.Click
        If TimerSerialIn.Enabled = False Then
            MsgBox("Failed to open User Data !!!" & vbCr & "Click the Connection menu
then click the Connect button.", MsgBoxStyle.Information, "Information")
            Return
        Else
            StrSerialIn = ""
            ViewUserData = True
            PictureBoxSelect.Top = ButtonUserData.Top
            PanelConnection.Visible = False
            PanelUserData.Visible = True
            PanelRegistrationandEditUserData.Visible = False
            PanelTravel1.Visible = False
        End If
    End Sub

    Private Sub ButtonRegistration_Click_1(sender As Object, e As EventArgs) Handles
ButtonRegistration.Click
        StrSerialIn = ""
        ViewUserData = False
        PictureBoxSelect.Top = ButtonRegistration.Top
        PanelConnection.Visible = False
        PanelUserData.Visible = False
        PanelRegistrationandEditUserData.Visible = True
        PanelTravel1.Visible = False
        ShowData()
    End Sub
    Private Sub ButtonTravel_Click_1(sender As Object, e As EventArgs) Handles
ButtonTravel.Click
        StrSerialIn = ""
        ViewUserData = False
        PictureBoxSelect.Top = ButtonTravel.Top
        PanelConnection.Visible = False
        PanelUserData.Visible = False
        PanelRegistrationandEditUserData.Visible = False
        PanelTravel1.Visible = True
    End Sub

    Private Sub PanelConnection_Paint(sender As Object, e As PaintEventArgs)
        e.Graphics.DrawRectangle(New Pen(Color.LightGray, 2),
PanelConnection.ClientRectangle)
    End Sub
    Private Sub PanelConnection_Resize(sender As Object, e As EventArgs)
```

```
        PanelConnection.Invalidate()
    End Sub
    Private Sub PanelUserData_Paint(sender As Object, e As PaintEventArgs)
        e.Graphics.DrawRectangle(New Pen(Color.LightGray, 2),
PanelConnection.ClientRectangle)
    End Sub
    Private Sub PanelUserData_Resize(sender As Object, e As EventArgs)
        PanelUserData.Invalidate()
    End Sub
    Private Sub PanelRegistrationandEditUserData_Paint(sender As Object, e As
PaintEventArgs)
        e.Graphics.DrawRectangle(New Pen(Color.LightGray, 2),
PanelConnection.ClientRectangle)
    End Sub
    Private Sub PanelRegistrationandEditUserData_Resize(sender As Object, e As
EventArgs)
        PanelRegistrationandEditUserData.Invalidate()
    End Sub
    Private Sub PanelTravel1_Paint(sender As Object, e As PaintEventArgs)
        e.Graphics.DrawRectangle(New Pen(Color.LightGray, 2),
PanelTravel1.ClientRectangle)
    End Sub
    Private Sub PanelTravel1_Resize(sender As Object, e As EventArgs)
        PanelTravel1.Invalidate()
    End Sub
    Private Sub ButtonScanPort_Click_1(sender As Object, e As EventArgs) Handles
ButtonScanPort.Click
        ComboBoxPort.Items.Clear()
        Dim myPort As Array
        Dim i As Integer
        myPort = IO.Ports.SerialPort.GetPortNames()
        ComboBoxPort.Items.AddRange(myPort)
        i = ComboBoxPort.Items.Count
        i = i - i
        Try
            ComboBoxPort.SelectedIndex = i
        Catch ex As Exception
            MsgBox("Com port not detected", MsgBoxStyle.Critical, "Error Message")
            ComboBoxPort.Text = ""
            ComboBoxPort.Items.Clear()
            Return
        End Try
        ComboBoxPort.DroppedDown = True
    End Sub
    Private Sub ButtonScanPort_MouseLeave1(sender As Object, e As EventArgs)
Handles ButtonScanPort.MouseLeave
        ButtonScanPort.ForeColor = Color.Maroon
    End Sub
    Private Sub ButtonScanPort_MouseMove(sender As Object, e As MouseEventArgs)
Handles ButtonScanPort.MouseMove
```

```
        ButtonScanPort.ForeColor = Color.FromArgb(6, 71, 165)
    End Sub
     Private Sub ButtonConnect_Click_1(sender As Object, e As EventArgs) Handles
ButtonConnect.Click
        If ButtonConnect.Text = "Connect" Then
            SerialPort1.BaudRate = ComboBoxBaudRate.SelectedItem
            SerialPort1.PortName = ComboBoxPort.SelectedItem
            Try
                SerialPort1.Open()
                TimerSerialIn.Start()
                ButtonConnect.Text = "Disconnect"
                LabelConnectionStatus.Text = "Connection Status : Connect"
                PictureBoxStatusConnect.Image = My.Resources.connect
            Catch ex As Exception
                MsgBox("Failed to connect !!!" & vbCr & "Arduino is not detected.",
MsgBoxStyle.Critical, "Error Message")
                PictureBoxStatusConnect.Image = My.Resources.disconnect
            End Try
        ElseIf ButtonConnect.Text = "Disconnect" Then
            PictureBoxStatusConnect.Image = My.Resources.disconnect
            ButtonConnect.Text = "Connect"
            LabelConnectionStatus.Text = "Connection Status : Disconnect"
            TimerSerialIn.Stop()
            SerialPort1.Close()
        End If
    End Sub
    Private Sub ButtonConnect_MouseHover(sender As Object, e As EventArgs) Handles
ButtonConnect.MouseHover
        ButtonConnect.ForeColor = Color.Maroon
    End Sub

    Private Sub ButtonConnect_MouseLeave(sender As Object, e As EventArgs) Handles
ButtonConnect.MouseLeave
        ButtonConnect.ForeColor = Color.FromArgb(6, 71, 165)
    End Sub
    Private Sub ButtonClear_Click_1(sender As Object, e As EventArgs) Handles
ButtonClear.Click
        LabelID.Text = "ID : _____"
        LabelName.Text = "Waiting..."
        LabelAddress.Text = "Waiting..."
        LabelAccNo.Text = "Waiting..."
        LabelCurrBal.Text = "Waiting..."
        PictureBoxUserImage.Image = Nothing
    End Sub
    Private Sub ButtonClear_MouseHover(sender As Object, e As EventArgs)
        ButtonClear.ForeColor = Color.White
    End Sub
    Private Sub ButtonClear_MouseLeave(sender As Object, e As EventArgs)
        ButtonClear.ForeColor = Color.FromArgb(6, 71, 165)
```

```
    End Sub
    Private Sub ButtonSave_Click_1(sender As Object, e As EventArgs) Handles
ButtonSave.Click
        Dim mstream As New System.IO.MemoryStream()
        Dim arrImage() As Byte
        If TextBoxName.Text = "" Then
            MessageBox.Show("Name cannot be empty !!!", "Error Message",
MessageBoxButtons.OK, MessageBoxIcon.Error)
            Return
        End If
        If TextBoxAddress.Text = "" Then
            MessageBox.Show("Address cannot be empty !!!", "Error Message",
MessageBoxButtons.OK, MessageBoxIcon.Error)
            Return
        End If
        If TextBoxAccNo.Text = "" Then
            MessageBox.Show("City cannot be empty !!!", "Error Message",
MessageBoxButtons.OK, MessageBoxIcon.Error)
            Return
        End If
        If StatusInput = "Save" Then
            If IMG_FileNameInput <> "" Then
                PictureBoxImageInput.Image.Save(mstream,
System.Drawing.Imaging.ImageFormat.Jpeg)
                arrImage = mstream.GetBuffer()
            Else
                MessageBox.Show("The image has not been selected !!!", "Error Message",
MessageBoxButtons.OK, MessageBoxIcon.Error)
                Return
            End If
            Try
                Connection.Open()
            Catch ex As Exception
                MessageBox.Show("Connection failed !!!" & vbCrLf & "Please check that the
server is ready !!!", "Error Message", MessageBoxButtons.OK, MessageBoxIcon.Error)
                Return
            End Try

            Try
                MySQLCMD = New MySqlCommand
                With MySQLCMD
                    .CommandText = "INSERT INTO " & Table_Name & " (Name, ID,
Address, Acc_No, Curr_Bal, Images) VALUES (@name, @ID, @address, @accno,
@currbal, @images)"
                    .Connection = Connection
                    .Parameters.AddWithValue("@name", TextBoxName.Text)
                    .Parameters.AddWithValue("@id", LabelGetID.Text)
                    .Parameters.AddWithValue("@address", TextBoxAddress.Text)
                    .Parameters.AddWithValue("@accno", TextBoxAccNo.Text)
                    .Parameters.AddWithValue("@currbal", TextBoxCurrBal.Text)
```

```
                .Parameters.AddWithValue("@images", arrImage)
                .ExecuteNonQuery()
            End With
            MsgBox("Data saved successfully", MsgBoxStyle.Information, "Information")
            IMG_FileNameInput = ""
            ClearInputUpdateData()
        Catch ex As Exception
            MsgBox("Data failed to save !!!" & vbCr & ex.Message,
MsgBoxStyle.Critical, "Error Message")
            Connection.Close()
            Return
        End Try
        Connection.Close()
    Else
        If IMG_FileNameInput <> "" Then
            PictureBoxImageInput.Image.Save(mstream,
System.Drawing.Imaging.ImageFormat.Jpeg)
            arrImage = mstream.GetBuffer()
            Try
                Connection.Open()
            Catch ex As Exception
                MessageBox.Show("Connection failed !!!" & vbCrLf & "Please check that
the server is ready !!!", "Error Message", MessageBoxButtons.OK,
MessageBoxIcon.Error)
                Return
            End Try
            Try
                MySQLCMD = New MySqlCommand
                With MySQLCMD
                    .CommandText = "UPDATE " & Table_Name & " SET
Name=@name,ID=@id,Address=@address,Acc_No=@accno,Curr_Bal=@currbal,Imag
es=@images WHERE ID=@id "
                    .Connection = Connection
                    .Parameters.AddWithValue("@name", TextBoxName.Text)
                    .Parameters.AddWithValue("@id", LabelGetID.Text)
                    .Parameters.AddWithValue("@address", TextBoxAddress.Text)
                    .Parameters.AddWithValue("@accno", TextBoxAccNo.Text)
                    .Parameters.AddWithValue("@currbal", TextBoxCurrBal.Text)
                    .Parameters.AddWithValue("@images", arrImage)
                    .ExecuteNonQuery()
                End With
                MsgBox("Data updated successfully", MsgBoxStyle.Information,
"Information")
                IMG_FileNameInput = ""
                ButtonSave.Text = "Save"
                ClearInputUpdateData()
            Catch ex As Exception
                MsgBox("Data failed to Update !!!" & vbCr & ex.Message,
MsgBoxStyle.Critical, "Error Message")
                Connection.Close()
```

```
                Return
            End Try
            Connection.Close()
        Else
            Try
                Connection.Open()
            Catch ex As Exception
                MessageBox.Show("Connection failed !!!" & vbCrLf & "Please check that
the server is ready !!!", "Error Message", MessageBoxButtons.OK,
MessageBoxIcon.Error)
                Return
            End Try
            Try
                MySQLCMD = New MySqlCommand
                With MySQLCMD
                    .CommandText = "UPDATE " & Table_Name & " SET
Name=@name,ID=@id,Address=@address,Acc_No=@accno,Curr_Bal=@currbal
WHERE ID=@id "
                    .Connection = Connection
                    .Parameters.AddWithValue("@name", TextBoxName.Text)
                    .Parameters.AddWithValue("@id", LabelGetID.Text)
                    .Parameters.AddWithValue("@address", TextBoxAddress.Text)
                    .Parameters.AddWithValue("@accno", TextBoxAccNo.Text)
                    .Parameters.AddWithValue("@currbal", TextBoxCurrBal.Text)
                    .ExecuteNonQuery()
                End With
                MsgBox("Data updated successfully", MsgBoxStyle.Information,
"Information")
                ButtonSave.Text = "Save"
                ClearInputUpdateData()
            Catch ex As Exception
                MsgBox("Data failed to Update !!!" & vbCr & ex.Message,
MsgBoxStyle.Critical, "Error Message")
                Connection.Close()
                Return
            End Try
            Connection.Close()
        End If
        StatusInput = "Save"
    End If
    PictureBoxImagePreview.Image = Nothing
    ShowData()
End Sub
Private Sub ButtonSave_MouseHover(sender As Object, e As EventArgs)
    ButtonSave.ForeColor = Color.White
End Sub
Private Sub ButtonSave_MouseLeave(sender As Object, e As EventArgs)
    ButtonSave.ForeColor = Color.FromArgb(6, 71, 165)
End Sub
```

```
    Private Sub ButtonClearForm_Click_1(sender As Object, e As EventArgs) Handles
ButtonClearForm.Click
        ClearInputUpdateData()
    End Sub
    Private Sub ButtonClearForm_MouseHover(sender As Object, e As EventArgs)
Handles ButtonClearForm.MouseHover
        ButtonClearForm.ForeColor = Color.White
    End Sub
    Private Sub ButtonClearForm_MouseLeave(sender As Object, e As EventArgs)
Handles ButtonClearForm.MouseLeave
        ButtonClearForm.ForeColor = Color.FromArgb(6, 71, 165)
    End Sub
    Private Sub ButtonScanID_Click_1(sender As Object, e As EventArgs) Handles
ButtonScanID.Click
        If TimerSerialIn.Enabled = True Then
            PanelReadingTagProcess.Visible = True
            GetID = True
            ButtonScanID.Enabled = False
        Else
            MsgBox("Failed to open User Data !!!" & vbCr & "Click the Connection menu
then click the Connect button.", MsgBoxStyle.Critical, "Error Message")
        End If
    End Sub
    Private Sub ButtonScanID_MouseHover(sender As Object, e As EventArgs)
        ButtonScanID.ForeColor = Color.White
    End Sub
    Private Sub ButtonScanID_MouseLeave(sender As Object, e As EventArgs)
        ButtonScanID.ForeColor = Color.FromArgb(6, 71, 165)
    End Sub
    Private Sub PictureBoxImageInput_Click_1(sender As Object, e As EventArgs)
Handles PictureBoxImageInput.Click
        OpenFileDialog1.FileName = ""
        OpenFileDialog1.Filter = "JPEG (*.jpeg;*.jpg)|*.jpeg;*.jpg"
        If (OpenFileDialog1.ShowDialog(Me) =
System.Windows.Forms.DialogResult.OK) Then
            IMG_FileNameInput = OpenFileDialog1.FileName
            PictureBoxImageInput.ImageLocation = IMG_FileNameInput
        End If
    End Sub
    Private Sub CheckBoxByName_CheckedChanged(sender As Object, e As EventArgs)
        If CheckBoxByName.Checked = True Then
            CheckBoxByID.Checked = False
        End If
        If CheckBoxByName.Checked = False Then
            CheckBoxByID.Checked = True
        End If
    End Sub
    Private Sub CheckBoxByID_CheckedChanged(sender As Object, e As EventArgs)
        If CheckBoxByID.Checked = True Then
            CheckBoxByName.Checked = False
```

```
        End If
        If CheckBoxByID.Checked = False Then
            CheckBoxByName.Checked = True
        End If
    End Sub
    Private Sub TextBoxSearch_TextChanged(sender As Object, e As EventArgs)
        If CheckBoxByID.Checked = True Then
            If TextBoxSearch.Text = Nothing Then
                SqlCmdSearchstr = "SELECT Name, ID, Address, Acc_No, Curr_Bal FROM
" & Table_Name & " ORDER BY Name"
            Else
                SqlCmdSearchstr = "SELECT Name, ID, Address, Acc_No, Curr_Bal FROM
" & Table_Name & " WHERE ID LIKE'" & TextBoxSearch.Text & "%'"
            End If
        End If
        If CheckBoxByName.Checked = True Then
            If TextBoxSearch.Text = Nothing Then
                SqlCmdSearchstr = "SELECT Name, ID, Address, Acc_No, Curr_Bal FROM
" & Table_Name & " ORDER BY Name"
            Else
                SqlCmdSearchstr = "SELECT Name, ID, Address, Acc_No, Curr_Bal FROM
" & Table_Name & " WHERE Name LIKE'" & TextBoxSearch.Text & "%'"
            End If
        End If
        Try
            Connection.Open()
        Catch ex As Exception
            MessageBox.Show("Connection failed !!!" & vbCrLf & "Please check that the
server is ready !!!", "Error Message", MessageBoxButtons.OK, MessageBoxIcon.Error)
            Return
        End Try
        Try
            MySQLDA = New MySqlDataAdapter(SqlCmdSearchstr, Connection)
            DT = New DataTable
            Data = MySQLDA.Fill(DT)
            If Data > 0 Then
                DataGridView1.DataSource = Nothing
                DataGridView1.DataSource = DT
                DataGridView1.DefaultCellStyle.ForeColor = Color.Black
                DataGridView1.ClearSelection()
            Else
                DataGridView1.DataSource = DT
            End If
        Catch ex As Exception
            MsgBox("Failed to search" & vbCr & ex.Message, MsgBoxStyle.Critical, "Error
Message")
            Connection.Close()
        End Try
        Connection.Close()
    End Sub
```

```
    Private Function AllCellsSelected(dgv As DataGridView) As Boolean
        AllCellsSelected = (DataGridView1.SelectedCells.Count =
(DataGridView1.RowCount *
DataGridView1.Columns.GetColumnCount(DataGridViewElementStates.Visible)))
    End Function
    Private Sub TimerTimeDate_Tick(sender As Object, e As EventArgs) Handles
TimerTimeDate.Tick
        LabelDateTime.Text = "Time " & DateTime.Now.ToString("HH:mm:ss") & "  Date
" & DateTime.Now.ToString("dd MMM, yyyy")
    End Sub
    Private Sub DeleteToolStripMenuItem_Click(sender As Object, e As EventArgs)
Handles DeleteToolStripMenuItem.Click
    End Sub
    Private Sub SelectAllToolStripMenuItem_Click(sender As Object, e As EventArgs)
Handles SelectAllToolStripMenuItem.Click
        DataGridView1.SelectAll()
    End Sub
    Private Sub ClearSelectionToolStripMenuItem_Click(sender As Object, e As
EventArgs) Handles ClearSelectionToolStripMenuItem.Click
        DataGridView1.ClearSelection()
        PictureBoxImagePreview.Image = Nothing
    End Sub
    Private Sub TimerSerialIn_Tick(sender As Object, e As EventArgs) Handles
TimerSerialIn.Tick
        Try
            StrSerialIn = SerialPort1.ReadExisting
            LabelConnectionStatus.Text = "Connection Status : Connected"
            If StrSerialIn <> "" Then
                If GetID = True Then
                    LabelGetID.Text = StrSerialIn
                    GetID = False
                    If LabelGetID.Text <> "________" Then
                        PanelReadingTagProcess.Visible = False
                        IDCheck()
                    End If
                End If
                If ViewUserData = True Then
                    ViewData()
                End If
            End If
        Catch ex As Exception
            TimerSerialIn.Stop()
            SerialPort1.Close()
            LabelConnectionStatus.Text = "Connection Status : Disconnect"
            PictureBoxStatusConnect.Image = My.Resources.disconnect
            MsgBox("Failed to connect !!!" & vbCr & "Arduino is not detected.",
MsgBoxStyle.Critical, "Error Message")
            ButtonConnect_Click_1(sender, e)
            Return
        End Try
```

```
    End Sub
    Private Sub IDCheck()
        Try
            Connection.Open()
        Catch ex As Exception
            MessageBox.Show("Connection failed !!!" & vbCrLf & "Please check that the
server is ready !!!", "Error Message", MessageBoxButtons.OK, MessageBoxIcon.Error)
            Return
        End Try
        Try
            MySQLCMD.CommandType = CommandType.Text
            MySQLCMD.CommandText = "SELECT * FROM " & Table_Name & "
WHERE ID LIKE '" & LabelGetID.Text & "'"
            MySQLDA = New MySqlDataAdapter(MySQLCMD.CommandText,
Connection)
            DT = New DataTable
            Data = MySQLDA.Fill(DT)
            If Data > 0 Then
                If MsgBox("ID registered !" & vbCr & "Do you want to edit the data ?",
MsgBoxStyle.Question + MsgBoxStyle.OkCancel, "Confirmation") =
MsgBoxResult.Cancel Then
                    DT = Nothing
                    Connection.Close()
                    ButtonScanID.Enabled = True
                    GetID = False
                    LabelGetID.Text = "_________"
                    Return
                Else
                    Dim ImgArray() As Byte = DT.Rows(0).Item("Images")
                    Dim lmgStr As New System.IO.MemoryStream(ImgArray)
                    PictureBoxImageInput.Image = Image.FromStream(lmgStr)
                    PictureBoxImageInput.SizeMode = PictureBoxSizeMode.Zoom
                    TextBoxName.Text = DT.Rows(0).Item("Name")
                    TextBoxAddress.Text = DT.Rows(0).Item("Address")
                    TextBoxAccNo.Text = DT.Rows(0).Item("Acc_No")
                    TextBoxCurrBal.Text = DT.Rows(0).Item("Curr_Bal")
                    StatusInput = "Update"
                End If
            End If
        Catch ex As Exception
            MsgBox("Failed to load Database !!!" & vbCr & ex.Message,
MsgBoxStyle.Critical, "Error Message")
            Connection.Close()
            Return
        End Try
        DT = Nothing
        Connection.Close()
        ButtonScanID.Enabled = True
        GetID = False
    End Sub
```

```
    Private Sub ViewData()
        LabelID.Text = "ID : " & StrSerialIn
        If LabelID.Text = "ID : _________" Then
            ViewData()
        Else
            ShowDataUser()
        End If
    End Sub
    Private Sub ButtonCloseReadingTag_Click(sender As Object, e As EventArgs)
        PanelReadingTagProcess.Visible = False
        ButtonScanID.Enabled = True
    End Sub
    Private Sub Form1_Resize(sender As Object, e As EventArgs) Handles
MyBase.Resize
        GroupBoxImage.Location = New Point((PanelUserData.Width / 2) -
(GroupBoxImage.Width / 2), GroupBoxImage.Top)
        PanelReadingTagProcess.Location = New
Point((PanelRegistrationandEditUserData.Width / 2) - (PanelReadingTagProcess.Width /
2), 106)
    End Sub
        Private Sub RefreshToolStripMenuItem_Click(sender As Object, e As EventArgs)
Handles RefreshToolStripMenuItem.Click
        ShowData()
    End Sub
    Private Sub ButtonCloseReadingTag_Click_1(sender As Object, e As EventArgs)
Handles ButtonCloseReadingTag.Click
        PanelReadingTagProcess.Visible = False
        ButtonScanID.Enabled = True
    End Sub
    Private Sub ButtonPay_Click(sender As Object, e As EventArgs) Handles
ButtonPay.Click
    End Sub
    Private Sub Button2Clear_Click(sender As Object, e As EventArgs) Handles
Button2Clear.Click
        Label1Show.Text = ""
        LabelRoute1.Text = ""
        Label1Distance.Text = ""
        Label1EstTime.Text = ""
    End Sub
End Class
```