

# **BUS MANAGEMENT SYSTEM**

**MUSSA SARKAWT ABDUL**

**QAIWAN INTERNATIONAL UNIVERSITY**

**UNIVERSITI TEKNOLOGI MALAYSIA****DECLARATION OF THESIS / UNDERGRADUATE PROJECT REPORT AND  
COPYRIGHT**

Author's full name : MUSSA SARKAWT ABDUL

Date of Birth : 4/9/1999

Title : BUS MANAGEMENT SYSTEM

Academic Session : : Semester 1 2022/2023

I declare that this thesis is classified as:

☐**CONFIDENTIAL**

(Contains confidential information under the Official Secret Act 1972)\*

☐**RESTRICTED**

(Contains restricted information as specified by the organization where research was done)\*

☒**OPEN ACCESS**

I agree that my thesis to be published as online open access (full text)

1. I acknowledged that Universiti Teknologi Malaysia reserves the right as follows:
2. The thesis is the property of Universiti Teknologi Malaysia
3. The Library of Universiti Teknologi Malaysia has the right to make copies for the purpose of research only.
4. The Library has the right to make copies of the thesis for academic exchange.

Certified by:



SIGNATURE OF STUDENT



SIGNATURE OF SUPERVISOR

QU192SCSJ012

MATRIC NUMBER

Mr. AbdulSalam Abdullah

NAME OF SUPERVISOR

NOTES : If the thesis is CONFIDENTIAL or RESTRICTED, please attach with the letter from the organization with period and reasons for confidentiality or restriction

“I hereby declare that I have read this thesis and in my  
opinion this thesis is sufficient in term of scope and quality for the  
award of the degree of Bachelor of Computer Science (Software Engineering)

Signature

:



Name of Supervisor

: Mr. AbdulSalam Abdullah

Date

: 5 JANUARY 2023

# BUS MANAGEMENT SYSTEM

MUSSA SARKAWT ABDUL

A thesis submitted in fulfilment of the  
requirements for the award of the degree of  
Bachelor of Computer Science (Software Engineering)

School of Computing  
Faculty of Engineering and Science  
Qaiwan International University

JANUARY 2023

## DECLARATION

I declare that this thesis entitled “*BUS MANAGEMENT SYSTEM*” is the result of my own research except as cited in the references. The thesis has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

Signature :



Name : MUSSA SARKAWT ABDUL

Date : 5 JANUARY 2023

## **DEDICATION**

I want to thank everyone involved in this initiative. I'd like to thank my Supervisor (Mr. AbdulSalam Abdullah) who helped me learn a lot about this project. His ideas and comments aided in the completion of this project. I am grateful to the college administration for providing me with such a significant chance. I believe I will participate in more such activities in the future. I guarantee that this project was created entirely by me and is not a forgery. Finally, I'd like to express my gratitude to my parents and friends for their excellent comments and guidance during the completion of this project.

## **ACKNOWLEDGEMENT**

In preparing this thesis, I was in contact with many people, researchers, academicians, and practitioners. They have contributed towards my understanding and thoughts. In particular, I wish to express my sincere appreciation to my main thesis supervisor, Mr. AbdulSalam, for encouragement, guidance, critics and friendship, advices and motivation. Without his continued support and interest, this thesis would not have been the same as presented here.

My fellow postgraduate student should also be recognised for their support. My sincere appreciation also extends to all my colleagues and others who have provided assistance at various occasions. Their views and tips are useful indeed. Unfortunately, it is not possible to list all of them in this limited space. I am grateful to all my family member.

## **ABSTRACT**

The goal of this project idea is to assist the community of people who take buses on a daily basis due to the aforementioned and an easy transportation method to be used. The project will be developed to help people who have a problem getting to the bus in time to go to their appointments, whether they are for work or school. As you know, in our region, buses are unorganized and do not belong to any companies. As a result, people are unsure of how to get to the bus station; some arrive early, while others arrive late, which may result in them missing the bus because it is full or does not have enough seats. This application will help to solve these issues by allowing users to book a ticket ahead of time so they can travel whenever they want, and it will be a convenient way for the community and the transportation industry.

## TABLE OF CONTENTS

	<b>TITLE</b>	<b>PAGE</b>
	<b>DECLARATION</b>	<b>iii</b>
	<b>DEDICATION</b>	<b>iv</b>
	<b>ACKNOWLEDGEMENT</b>	<b>v</b>
	<b>ABSTRACT</b>	<b>vi</b>
	<b>TABLE OF CONTENTS</b>	<b>vii</b>
	<b>LIST OF TABLES</b>	<b>x</b>
	<b>LIST OF FIGURES</b>	<b>xi</b>
	<b>LIST OF ABBREVIATIONS</b>	<b>xii</b>
	<b>LIST OF APPENDICES</b>	<b>xiii</b>
<b>CHAPTER 1</b>	<b>INTRODUCTION</b>	<b>1</b>
1.1	Introduction	1
1.2	Problem Background and Proposed Solution	1
1.3	Project Aim	2
1.4	Research Objectives	2
1.5	Scope	2
1.6	Project Importance	3
<b>CHAPTER 2</b>	<b>LITERATURE REVIEW</b>	<b>5</b>
2.1	Introduction	5
2.2	Current Systems Analysis in General	5
2.3	Reviewing Similar System	6
2.3.1	TheRide.org	6
2.3.2	WeGo Public Transit	7
2.4	Compare Between Existing Systems	7
2.5	Literature Review of Technology Used	9
2.5.1	Front End Technologies	9
2.5.2	Xampp	9

2.5.3	Vs Code	9
2.6	Backend Technologies	10
2.7	The Methods Used	10
2.8	Chapter Summary	12
<b>CHAPTER 3</b>	<b>SYSTEM DEVELOPMENT METHODOLOGY</b>	<b>13</b>
3.1	Introduction	13
3.2	Analysis Process and Planning	14
3.3	Design	14
3.4	Development	15
3.5	Testing	15
3.6	Maintenance	15
3.7	Technology Used Description	16
3.8	HTML	16
3.9	FLUTTER	16
3.10	CSS	17
3.11	MYSQL	17
3.12	Hardware and Software Requirement Analysis	17
<b>CHAPTER 4</b>	<b>REQUIREMENT ANALYSIS AND DESIGN</b>	<b>19</b>
4.1	Introduction	19
4.2	Requirement Analysis	19
4.3	Functional Requirement	20
4.4	Use Case Diagram	20
4.4.1	Register Sequence Diagram	22
4.4.2	Login Sequence Diagram	23
4.4.3	Booking Tickets Sequence Diagram	24
4.4.4	Bus Route Sequence Diagram	25
4.4.5	Feedback Sequence Diagram	26
4.4.6	Payment Sequence Diagram	27
4.4.7	Logout Sequence Diagram	28
4.5	Activity Diagram	28
4.5.1	Booking Ticket Activity Diagram	29

4.5.2	Bus Route Activity Diagram	30
4.6	Data Dictionary	31
4.7	Interface Design	32
4.8	Chapter Summery	37
<b>CHAPTER 5</b>	<b>IMPLEMENTATION AND TESTING</b>	<b>39</b>
5.1	Introduction	39
5.2	Coding of System Main Function	39
5.2.1	Register	39
5.2.2	Login	39
5.2.3	Booking Ticket	40
5.2.4	Apply	40
5.3	Interfaces of System Main Function	40
5.4	Testing	40
5.4.1	Black Box Testing	40
5.4.2	White Box Testing	41
<b>CHAPTER 6</b>	<b>CONCLUSION</b>	<b>42</b>
6.1	Introduction	42
6.2	Achievement of Project Objectives	42
6.3	Suggestions For Future Improvement	42
<b>REFERENCES</b>		<b>44</b>

## LIST OF TABLES

TABLE NO.	TITLE	PAGE
Table 2.1	Comparison Between Existing System	7
Table 4.1	Functional Requirement	20
Table 4.2	User Case Description	21
Table 0.1	Definitions, Acronyms and Abbreviation	46
Table 0.2	Use Case Description	49
Table 0.3	User Characteristic	50
Table 0.4	Use Case for Register	51
Table 0.5	Use Case Description for Login	52
Table 0.6	Use Case Description for Booking Ticket	53
Table 0.7	Use Case Description for Bus route	54
Table 0.8	Use Case Description for feedback	55
Table 0.9	Definitions, Acronyms and Abbreviation	56
Table 0.10	Data Dictionary	58
Table 0.11	Abbreviation and Definitions	64
Table 0.12	Use Case Description for Register	65
Table 0.13	Use Case Description for Login	66
Table 0.14	Use Case Description for Booking Ticket	67
Table 0.15	Use Case Description for Bus route	68
Table 0.16	Use Case Description for feedback	69

## LIST OF FIGURES

<b>FIGURE NO.</b>	<b>TITLE</b>	<b>PAGE</b>
Figure 3.1	The Software Development Cycle	13
Figure 4.1	Use case diagram	20
Figure 4.2	Register Sequence Diagram	22
Figure 4.3	Login Sequence Diagram	23
Figure 4.4	Booking Ticket Sequence Diagram	24
Figure 4.5	Bus Route Sequence Diagram	25
Figure 4.6	Feedback Sequence Diagram	26
Figure 4.7	Payment Sequence Diagram	27
Figure 4.8	Logout Sequence Diagram	28
Figure 4.9	Booking Ticket Activity Diagram	29
Figure 4.10	Bus Route Activity Diagram	30
Figure 4.11	Login or Sign-up Page	32
Figure 4.12	Sign up page	33
Figure 4.13	Home Page	34
Figure 4.14	Payment Page	35
Figure 4.15	Payment Via PayPal page	36
Figure 0.1	Use Case	47
Figure 0.2	Entity Relationship Diagram	57
Figure 0.3	Register Interface	59
Figure 0.4	Login Interface	60
Figure 0.5	Home Page interface	61
Figure 0.6	Payment interface	62
Figure 0.7	Payment via paypal Interface	63

## **LIST OF ABBREVIATIONS**

SRS	-	Software Requirement Specification
SDD	-	Software Design Description
STD	-	Software Testing Document

## **LIST OF APPENDICES**

<b>APPENDIX</b>	<b>TITLE</b>	<b>PAGE</b>
Appendix A	Software Requirement Specification	45
Appendix B	Software Design Description	56
Appendix C	Software Testing Document	64

# **CHAPTER 1**

## **INTRODUCTION**

### **1.1 Introduction**

As a citizen of a big country like Iraq, we seek for a cheap and easy to get transportation to go and do our daily work. This application helps us to reach this aim. Bus management system is a helpful application for people because it includes a GPS that can determine where the bus go and where it comes from. Also, it contains a map shows the passengers their destination and where the buses stop in every station. In this application stakeholders can know exactly their benefits by using bus management system. First, this system provides an affordable transportation easily because people can know which time the buses arrive in this way, they can easily manage their time.

### **1.2 Problem Background and Proposed Solution**

Transportation has been a problem that is facing the Iraqi citizens for many years. Currently, in our daily life we are using public transportation widely, one of the main public transportations is buses, but we still facing many difficulties, regarding the time management, scheduling and availability of bus. Base on most people experiences, people will be waiting for buses at the bus station, curelessly knowing and uncertain about the bus arrival time. Additionally, passengers also do not know the seats availability especially during the peak hours. Besides, passengers also might face the uncertainty regarding the headed destination by the busses. So, in order to solve this problem that has been an obstacle for the Iraqi people for ages, a digital application is needed so that people no longer have to wait curelessly without knowing that they can get on the ride in the first place.

### **1.3 Project Aim**

The aim of this project is to establish an application its core is to manage people's time, every day people waste a lot of time waiting for affordable transportation to do their daily routine this application provides the location and time to its user and let them know how many seats are available. With this features the aim of this application is to make transportation easier and affordable for people.

### **1.4 Research Objectives**

1. To collect information from the public in order to get the necessary requirements.
2. To develop an application that will assist the public in keeping track of their appointments.
3. To test the application to avoid any future problems.

### **1.5 Scope**

When using public transit in Iraq, no one can predict when the bus will come or how long it would take to go to your destination. If there was a system where citizens could book their tickets, the process would be quicker. Since there is currently no bus system, it will allow customers to reserve tickets and view the bus schedule, among other functions. Also, "the benefits of public transportation are amplified when the systems are planned and engineered for interconnectivity, high-quality user experience, and efficiency." This application provides cheap and affordable transportation to people.

We can summarize the scope into the following points:

- Affordable
- Knowing the direction
- Easy booking

## **1.6 Project Importance**

As we know that public transportation is very important and it is most common way of transportation in the world it reduces congestion in towns and cities. using public transport is cheaper than owning and operating a car. no more sitting in traffic jams in rush hour thanks to bus lanes and other bus priority measures. it reduces your carbon footprint. This application is working to make people use public transportation more especially buses by make it easier to use and provide every information about it like time, seats and its destination.

## **CHAPTER 2**

### **LITERATURE REVIEW**

#### **2.1 Introduction**

Automating the bus management system cannot go unnoticed given the customer service sector's growing emphasis on patient safety. The Bus Management System is a complete dispensing workflow management tool developed to improve customer efficiency, safety, and accuracy. The systems currently in use and recent works that are being used to address the problems with this system are explained and discussed in this chapter. Additionally, it will explain the case studies, which will be explained in great detail. We also list the works' advantages and disadvantages. The most important details have also been highlighted, and the concepts, tools, system comparison, and methodologies used in this project have all been briefly explained.

#### **2.2 Current Systems Analysis in General**

There are many features in the systems of today, as well as a manual system from the past that is still used in offices. People in our region, for instance, are unable to make appointments, reserve tickets, or get information about bus arrival times so they can decide whether to leave early or late and avoid being late for work or school. As a result, they may be forced to miss important events. Which is an issue that our region needs to have resolved as soon as possible.

## **2.3 Reviewing Similar System**

The systems that are currently in use and have been researched or developed in the past that are comparable to the bus management system will be discussed in this section. The discussion will include the name of the article, a brief explanation of their weaknesses and strengths, and then we will compare them:

### **2.3.1 TheRide.org**

The greater Ann Arbor-Ypsilanti region's public transportation system is run by TheRide organization. TheRide offers dependable, secure, and reasonably priced transportation services to ensure customer satisfaction. Fixed route buses, paratransit services, commuter services (Park & Ride, vanpools), demand-responsive services, event services, and airport services are just a few of the service types offered on this website.

The Pros of the system:

- It provides a method for tracking buses.
- Contains a schedule by tracking a route.
- You can plan your trip through the website

The Cons of the system:

- Doesn't show how many seats are available.
- Doesn't show how many busses are available for a trip.

### 2.3.2 WeGo Public Transit

The WeGo brand is a collective dedication to providing first-rate customer service, friendliness, honesty, and integrity as we link riders to their community and one another. The goal of creating this website was to connect people, among other things. For instance, if a person is single and without a partner to travel with, they can meet new people.

The Pros of the system:

- Provides ticket types.
- Has more than bus transportation.

The Cons of the system:

- Has too many advertisements.
- The planning trip is too expensive.

## 2.4 Compare Between Existing Systems

Table 2.1 Comparison Between Existing System

Features	The Ride	WeGo Transit	Bus Management System
(you can modify add or delete)			

<b>Deployment</b>	Website	Website	Website and Desktop
<b>Specific Functions</b>	1. Allow user to login. 2. Allow user to schedule. 3. Allow user to plan trip.	1. Allow user to login. 2. Allow user to schedule. 3. Allow user to reserve tickets.	1. Allow user to login. 2. Allow user to reserve tickets. 3. Allow user to plan for trips and see bus tracker.
<b>Technology</b>	Web Technology	Web Technology	Mobile Application Technology
<b>Platform</b>	Google Chrome	Google Chrome	Android & IOS
<b>Connectivity</b>	Online	Online	Online
<b>Security features</b>	OTP Verification	Normal password	OTP Verification
<b>User Friendly</b>	Yes	yes	yes

## **2.5 Literature Review of Technology Used**

The technologies used to create the Delivery and Tracking Online System are covered in this section. A few of these technologies are as follows:

### **2.5.1 Front End Technologies**

The majority of the programming languages used in this project are Dart (flutter). We chose flutter because it is a multi-platform framework that can be used to create both mobile and web applications and because it is multi-platform, it works on both Android and IOS, making it simpler for customers to access our system without any problems.

### **2.5.2 Xampp**

XAMPP was created by Apache Friends and is open-source software. Apache distributions for Apache server, Maria DB, PHP, and Perl are all included in the XAMPP software package. Additionally, it is essentially a local host or server. Your personal desktop or laptop computer serves as the local server. Before uploading your website to the remote web server, use XAMPP to test the clients or your website. You can test MYSQL, PHP, Apache, and Perl projects on your local computer using the XAMPP server software.

### **2.5.3 Vs Code**

Visual Studio Code, a free open-source text editor made by Microsoft (also referred to as VS Code). VS Code works with Windows, macOS, and Linux. Despite its relatively light weight and robust features, VS Code has recently risen to the top of the list of development environment tools. In addition to Java, C++, and Python, VS

Code supports a wide variety of programming languages, including CSS, Go, and Docker files. You can also add new extensions to VS Code, such as code linters, debuggers, and support for cloud and web development.

## **2.6 Backend Technologies**

Built on Structured Query Language, MySQL is an open-source relational database management system (RDBMS) that is supported by Oracle (SQL). Almost every operating system, including Linux, UNIX, and Windows, supports MySQL. Websites and online publications are where MySQL is most frequently used, despite the fact that it has many other uses as well. A key element of the LAMP open-source enterprise stack is MySQL. The LAMP web development platform is made up of the relational database management system MySQL, the object-oriented scripting language PHP, the Linux operating system, and the web server Apache.

## **2.7 The Methods Used**

There are two methods for gathering data: qualitative and quantitative. Graphs and numbers are used in quantitative research. It is used to verify or validate theories and presumptions. Quantitative research collects data on people's behaviors, beliefs, attitudes, and other traits in order to make generalizations about a larger group of people. In quantitative research, facts and trends are expressed using quantifiable data. The outcomes of this kind of research are derived using statistical and mathematical tools. In an effort to quantify a problem and determine its prevalence, quantitative data will be used to look for effects that can be applied to a larger population.

Online surveys, print surveys, mobile surveys, kiosk surveys, online polls, systematic observations, and other data collection methods are included. telephone interviews, in-person interviews, and so forth. The results of qualitative research are written up. It helps people comprehend concepts, ideas, and emotions. Qualitative research focuses on words and their meanings. You can learn more about topics you

don't know a lot about by conducting this kind of research. I decided on quantitative research because a sample of people in a survey are given a list of closed or multiple-choice questions (online, in person, or over the phone). Quantitative research includes surveys. Data for the Volunteering Opportunities System were gathered through a survey.

We also prepared a number of important interview questions that centered on the advantages of volunteer work for students' future careers as well as all of the challenges that Kurdistan will face. Most importantly, we focused on how the system might help students. Because it makes it possible to spot flaws and problems, the interview process is also designed with a funnel structure. Because surveys are a particular kind of data collection tool used to gather information about people, that is the reason I selected them. a survey may be created to collect data on individuals or to elicit opinions from survey participants.

One of the most popular methods for conducting surveys today is online. Additionally, I selected Google Forms because it is one of the most widely used platforms for online surveys. The website has a huge selection of templates and options that make it very user-friendly and speed up the survey creation process. A statistical inference and estimation technique called sampling involves choosing individuals or a subset of the population. It is a crucial component of any research strategy because it also saves time and money. Software for research surveys can use sampling approaches for the best derivation. This approach will enable quicker data collection while saving money. Social media sites like Facebook, Instagram, and WhatsApp all shared the survey.

This study will employ a method that makes the facilities simpler. The two types of sampling available are probability sampling and non-probability sampling. The non-probability approach is a sampling technique that relies on a researcher's or instead of relying on a predetermined selection process, statisticians can choose a representative sample on their own. A non-probable sample usually produces biased results that may not accurately represent the target audience. Non-probability sampling, however, can be significantly more useful in some circumstances than the

other type, such as the preliminary stages of a study or when conducting research on a tight budget. Non-probability sampling is a type of convince sampling that enables us to choose users in accordance with our prior knowledge. Both organization staff and recently graduated students made up the population I focused on for my survey.

## **2.8 Chapter Summary**

This chapter covered an overview of the issue and a potential solution, which was followed by a thorough discussion of the issue, a case study of this project, a thorough discussion of the systems currently in use in the area that will be compared to the bus management system and other similar applications, and a review of pertinent technical literature. Last but not least, the survey results and analysis from the Google form will be used to address the technologies for the System's development. The System Development Methodology is covered in detail in the next chapter along with how it will be applied to this system.

## CHAPTER 3

### SYSTEM DEVELOPMENT METHODOLOGY

#### 3.1 Introduction

A planned and effective growth process must be ensured, and methodology is a crucial and important factor. It is a technique for developing software. It also acts as a roadmap to make sure the development of the system continues in the right direction. With a methodology, building a system will be simpler to organize, reducing system development costs, lengthening development times, and achieving goals through time management. This section will go over the phases of the life cycle development process and how it supports the bus management system.

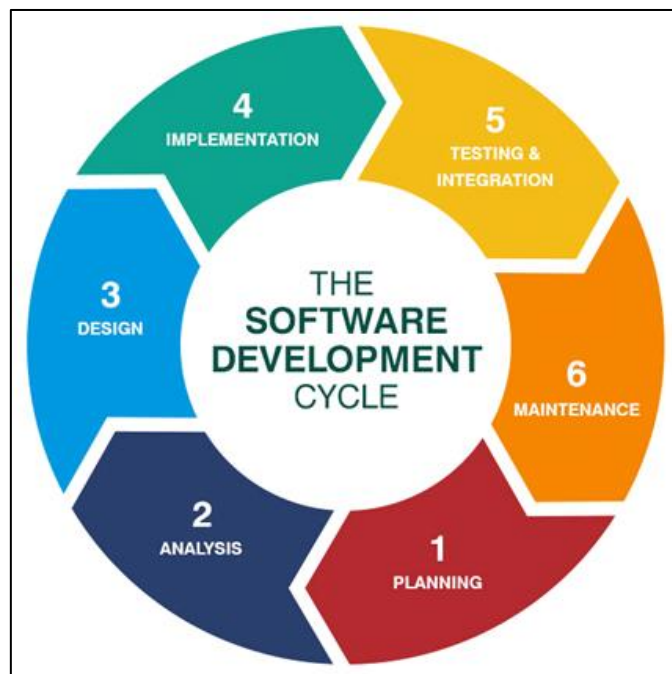


Figure 3.1 The Software Development Cycle

### **3.2 Analysis Process and Planning**

The requirements phase of the SDLC is the first step. The senior team members gather the feedback from all the stakeholders and industry subject matter experts. The need for quality assurance and the hazards involved are considered in the planning phase. At this point, the project's general scope, potential hurdles, and first directives are all more clearly defined. Next, the document software requirements are defined using the requirement obtained in the analysis phase.

Analysis and planning pros regarding the system:

- Describe any system documents for the interface.
- Designing interface in a friendly way.

### **3.3 Design**

- In the third phase, the requirement definition document is used to guide the creation of system and software design papers. The whole system's architecture is now more easily understood. The design phase is essential prior to the primary development phase. The inner workings and special characteristics of the program, including the software's user interface, its source code, and its data storage system. In addition, there are two distinct types of designs seen in design documents.
- High-Level Design.
- Low-Level Design.

### **3.4 Development**

The next step after the system design phase is the coding phase. The first step in developing a system is for programmers to start writing code in their preferred language. During the coding phase, tasks are divided into smaller components called "units" or "modules" and given to specific programmers to work on. It's the longest part of the SDLC, which stands for "software development life cycle." During this stage, the developer must adhere to established coding practices. They must also utilize programming tools to design and construct the interface and system code that will be released and completed for the users.

### **3.5 Testing**

The program is deployed to the testing environment after it is complete. As a first step, the testing team ensures the system is fully operational. This is carried out to ensure that the application as a whole performs as expected per the client's specifications. During the testing phase, developers examine their code in depth, keeping note of any errors or defects that need to be rectified and then retested. The software's final release must meet all of the quality standards specified in the SRS document.

### **3.6 Maintenance**

The System Development Life Cycle continues after software is put on the market for purchase. It is now necessary for developers to go into maintenance mode and begin doing any duties necessary to fix the problems that end users have reported. This stage includes the following actions:

**Bug Fixing** - Due to some functions not being tested at all, users have reported bugs and issues that they found in the system.

**Update** – updating the application to add new features that the users might need.

### **3.7 Technology Used Description**

This section will go deeply explain research on technology and software that will be used in the bus management system and the tools that will be used like. HTML, Flutter, CSS and MySQL.

### **3.8 HTML**

A formatting language called hypertext language is a way to show methods downloaded from the Internet. All units is known as a Web, html is the markup language that used to show Web pages. It is often stored in files with the .htm or .html extension. These HTML files, which might number in the hundreds or even thousands for a website, are kept in several folders. Every time you view a web page, the server sends the html files to the user's browser. Your browser then displays the html after reading it from the files. Instead of utilizing static html, some web applications build html in response to certain servers events.

### **3.9 FLUTTER**

Flutter is a multi-platform scripting language that can be used to create mobile and web applications. Since Flutter supports both Android and iOS, we don't need to write different code for each platform; instead, we can use the same code for both. Google developed the open-source Flutter language, which is mostly used to construct apps using widgets. The benefits of flutter are listed below to help you understand it better:

- Superb Productivity.
- Easy To Use.
- Free.
- Better Documentation website.

### **3.10 CSS**

The coding CSS, also is known as cascading style sheets, is used to style components in mark-up coding languages like HTML. It divides the website's content from its aesthetic appearance. The two are closely connected since HTML provides the framework for a website and CSS controls all of its features.

### **3.11 MYSQL**

MYSQL, a relational database management system, was developed by the Swedish company MySQL AB (RDBMS). This program's principal function is to store and retrieve data that other applications have requested. Additionally, it includes tools like Transform, Extract, and Load and reporting services that let users query, edit, and modify data using SQL (Structured Query Language). The PHP framework-based systems typically utilize it. How does it function? As it gets a data to store and manage the data, MySQL defines the connections between each table.

### **3.12 Hardware and Software Requirement Analysis**

Both hardware and software are needed for this project's system. Tasks include computation, processing, and storing, as well as input and output. Any physical objects

utilized to carry out various tasks, such as sensors or computer hardware, are referred to as hardware. On the other hand, software is a set of instructions that tell a computer how to perform specific activities, such creating code and building a system. Both elements must be present to create. Moreover, A program, for instance, is a piece of installed software that has its own code.

#### Hardware Requirement:

- RAM 3GB
- Storage 16GB
- Low Version

## **CHAPTER 4**

### **REQUIREMENT ANALYSIS AND DESIGN**

#### **4.1 Introduction**

Chapter 4 which is the requirement analysis and design, this chapter will take a part to discuss the diagrams that will be used in this project which are the use case diagram describing the actors of the system and what they can do and how they will use the functions in the application. Then, we will explain the sequence diagram for the use case functions and what it does with the database system, as well as the activity diagram, which describes when the function starts and ends. After, this chapter also will discuss the data dictionary which describes the database of the functions and the keys for it like foreign key, prime key...etc. Lastly, it will show the interface design of the application before it is released.

#### **4.2 Requirement Analysis**

Requirements analysis is known as the process of Identification, validation, and documentation of requirements for projects and other change activities. Typically, stakeholders including owners, users, and subject matter experts help gather requirements. These undeveloped needs are then changed to make them realistic, implementable, tested, measurable, and sufficiently detailed in their documentation. This project includes 3 actors which are User, admin and bus driver and each actor has its own role that can be used.

### 4.3 Functional Requirement

Table 4.1 Functional Requirement

Actor	Role
User	The user is the most important actor in the system and has the most roles that can be used like register, login, booking tickets, watching bus routes, giving feedback, paying for tickets and logout.
Admin	The admin is the actor who can take the feedback and send the payment to the bank system.
Bus Driver	The bus driver can also login from accounts that have been given to from the system so that they can see the ticket number, station number and give feedback.

### 4.4 Use Case Diagram

Use case testing is a method of software testing that aids in locating test cases that cover every aspect of the system from action to itself. Use case testing, in another meaning, helps in outlining the customer path with its product.

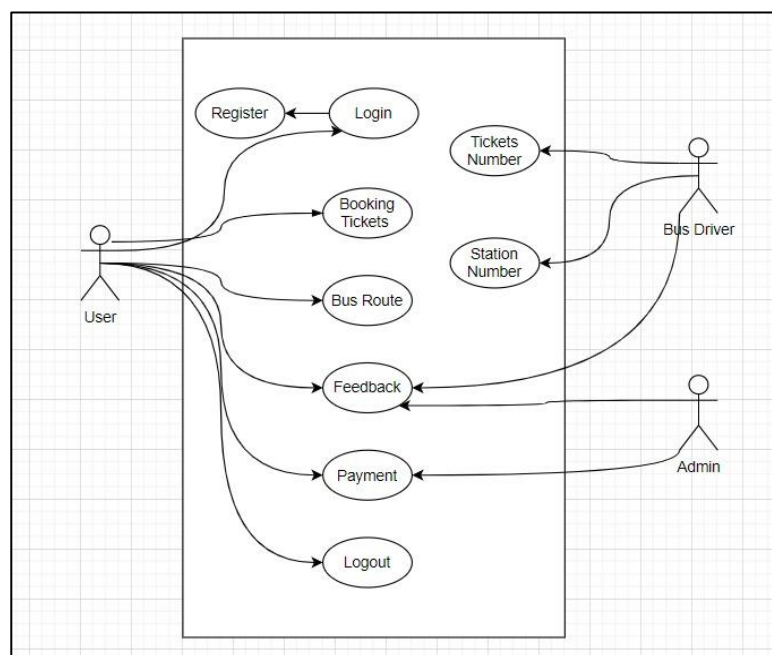


Figure 4.1 Use case diagram

Table 4.2 User Case Description

Use Case	Description
Register	Register is the first step that the user needs to do before they use any other function.
Login	The user needs to login to the application using the register form that has been sent to the database.
Booking tickets	One of the main functions that the user can do is booking a ticket, first the system shows the ticket types then the user books a ticket in their mind.
Bus route	In this function, the user can see where the bus is heading online using GPS.
Feedback	Feedback is a way that the user and the bus driver can contact the admins through the system. And only the admin sees the text.
Payment	After the user books a ticket they need to pay for the ticket through the system then the admin sends it to the bank to send the receipt to the user.
Logout	The user can log out if they want to.

#### 4.4.1 Register Sequence Diagram

In the register diagram the user sends a request to the system then the system sends it back to the user. After that the user needs to fill out a form and it will send it to the database to be saved.

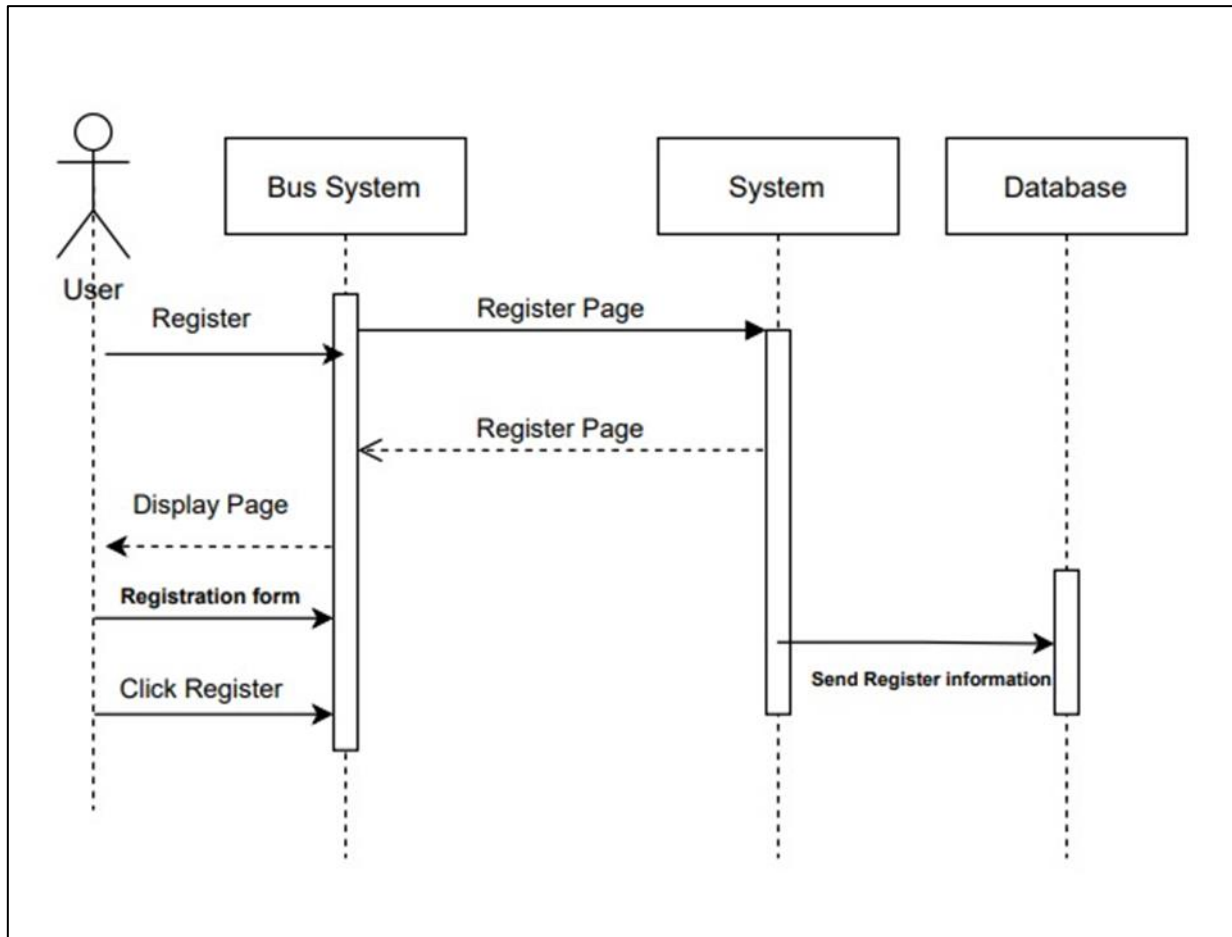


Figure 4.2 Register Sequence Diagram

#### 4.4.2 Login Sequence Diagram

In the login sequence diagram, the user sends a request to the system and receive it back then they have to fill the login form from the register page and send it back to the database.

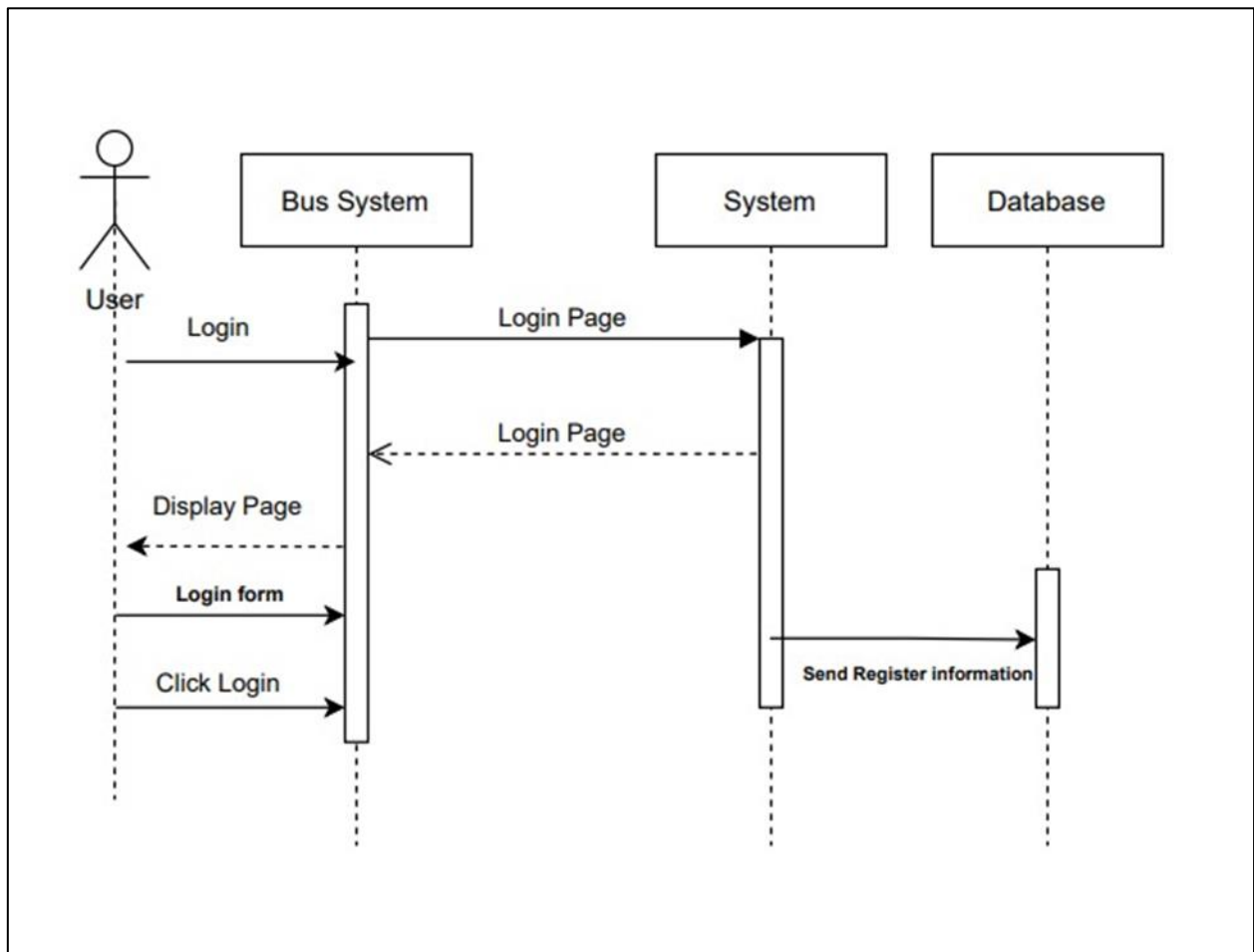


Figure 4.3 Login Sequence Diagram

#### 4.4.3 Booking Tickets Sequence Diagram

In booking tickets the user requests a ticket page and receives it back so the user can choose a ticket type then clicks on a book to be sent to the database and to be saved.

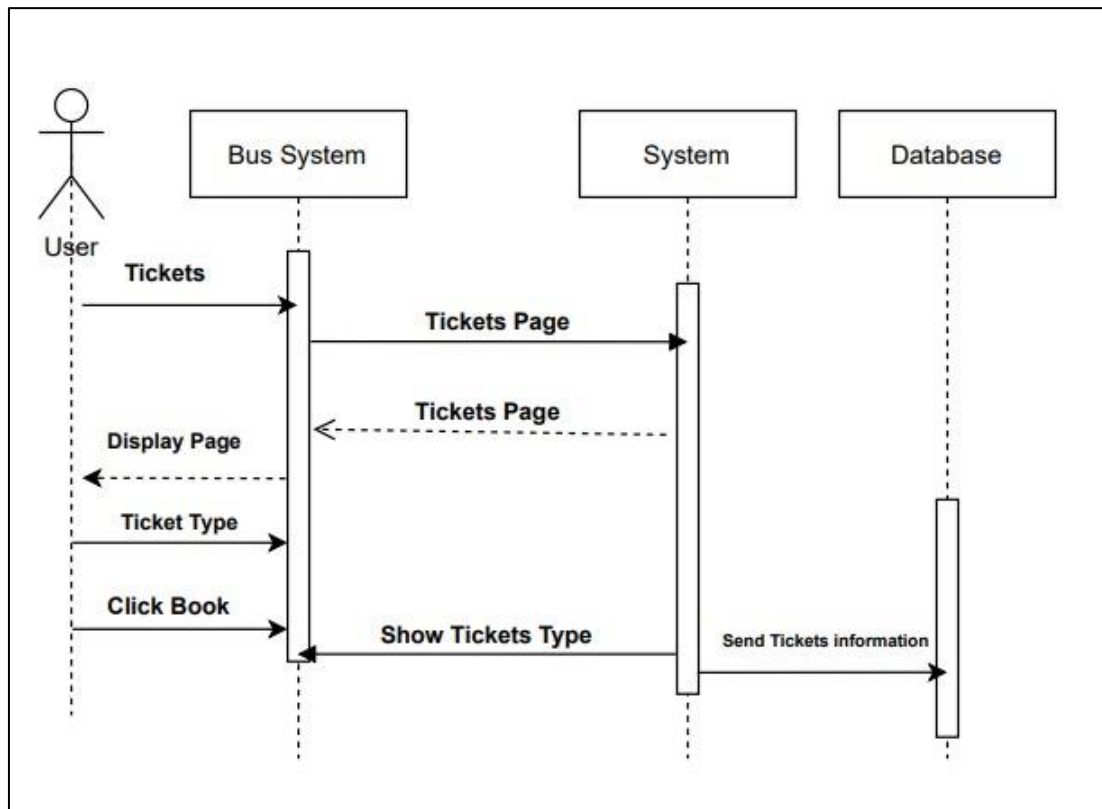


Figure 4.4 Booking Ticket Sequence Diagram

#### 4.4.4 Bus Route Sequence Diagram

In the bus route diagram the user requests the bus route to the system then they receive it back so the user can open the location to receive the GPS location from the system.

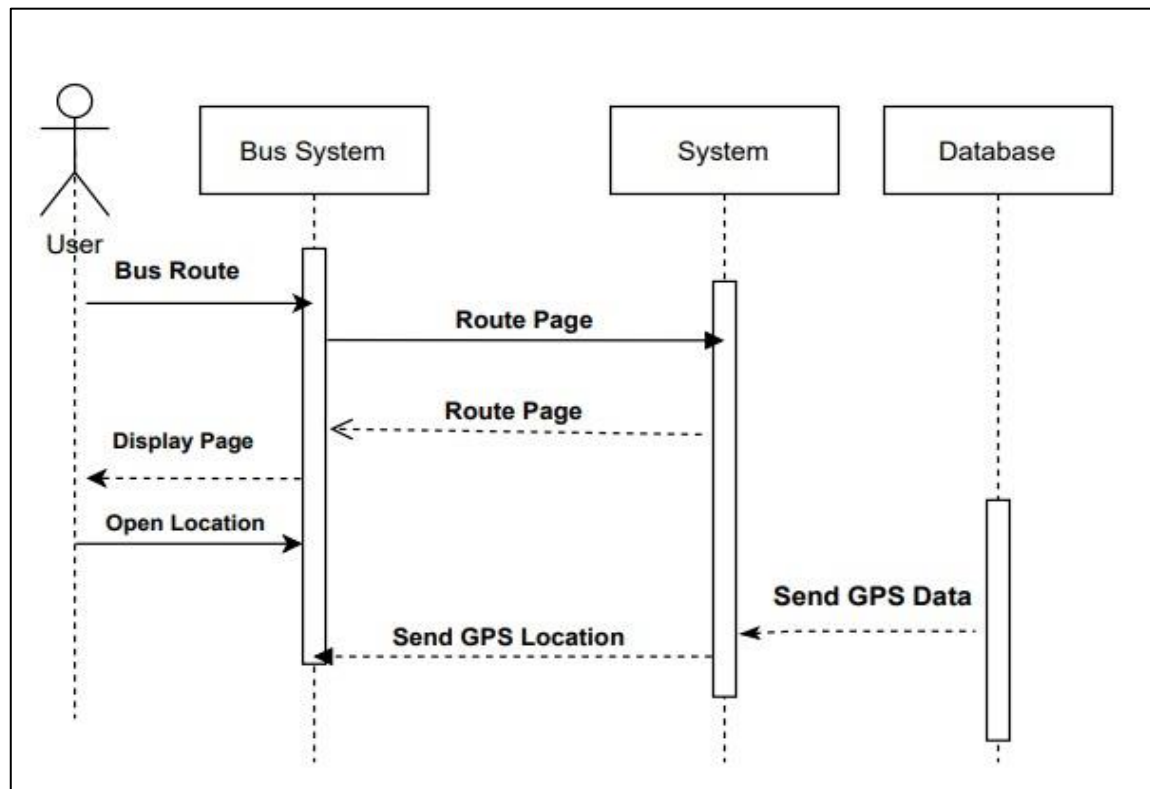


Figure 4.5 Bus Route Sequence Diagram

#### 4.4.5 Feedback Sequence Diagram

In the feedback sequence diagram the user requests for the feedback page from the system to receive it back and enters the text to be sent to the database.

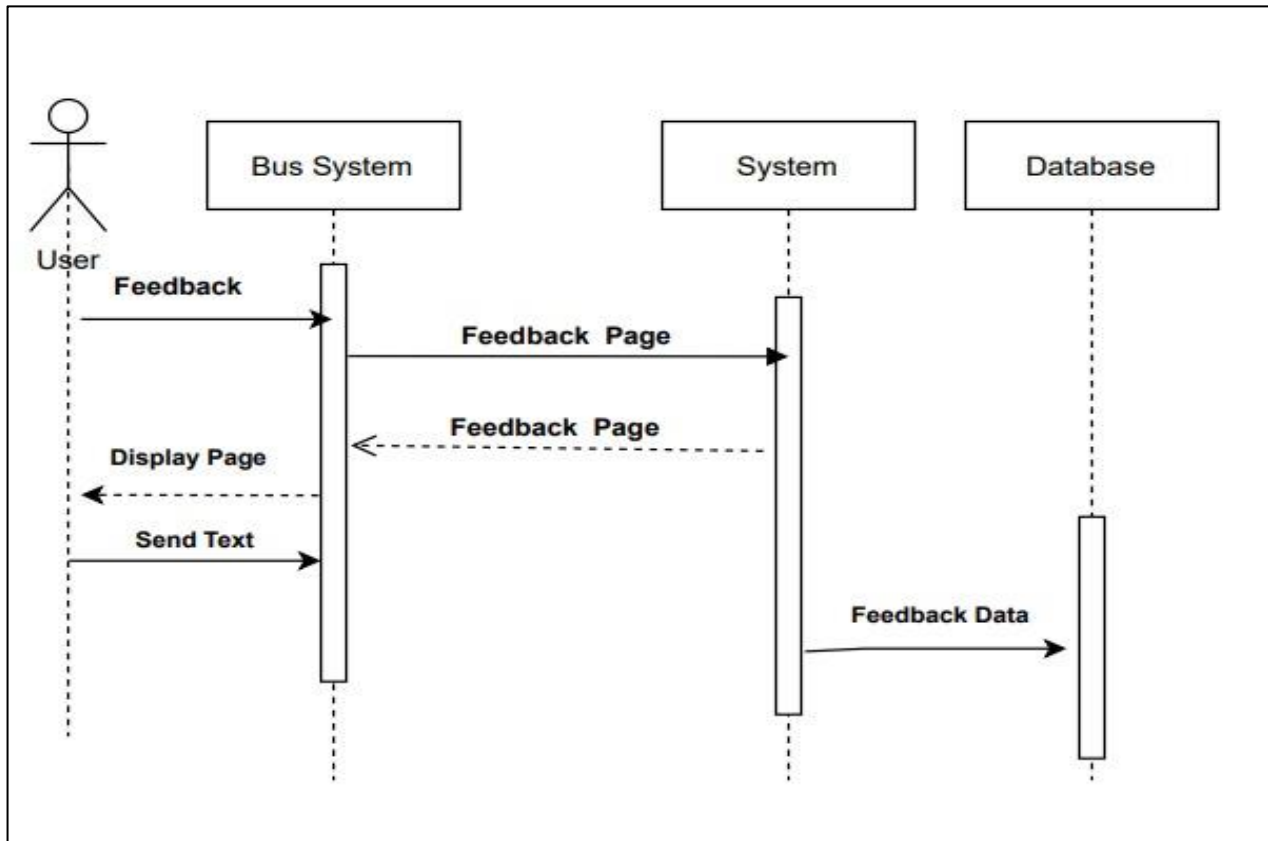


Figure 4.6 Feedback Sequence Diagram

#### 4.4.6 Payment Sequence Diagram

In the payment sequence diagram the user requests for the payment page to receive it again the the system sends payment form to the user to fill the information then the bank sends a receipt to the system to be sent to the user.

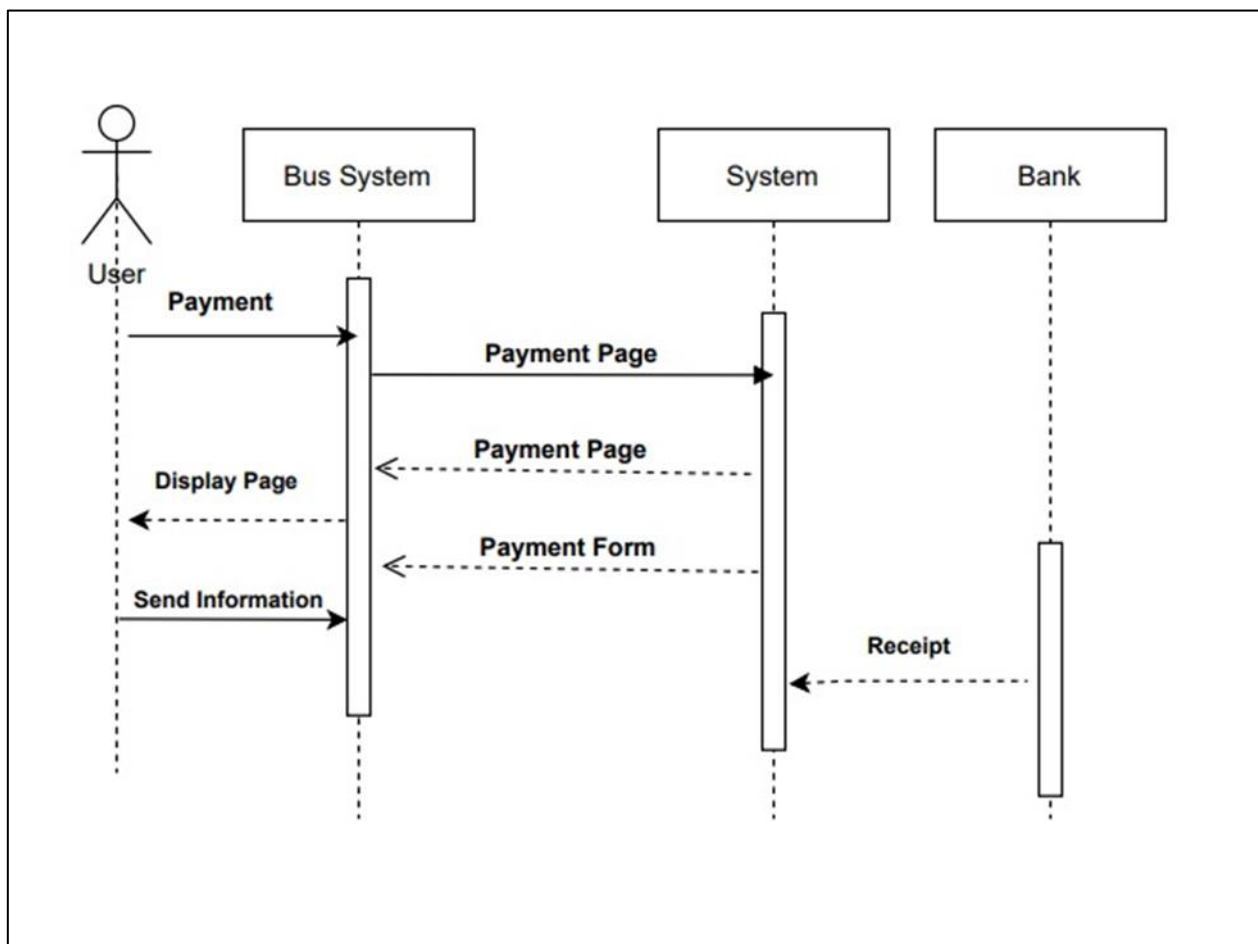


Figure 4.7 Payment Sequence Diagram

#### 4.4.7 Logout Sequence Diagram

The logout sequence diagram shows how the user can log out though the system.

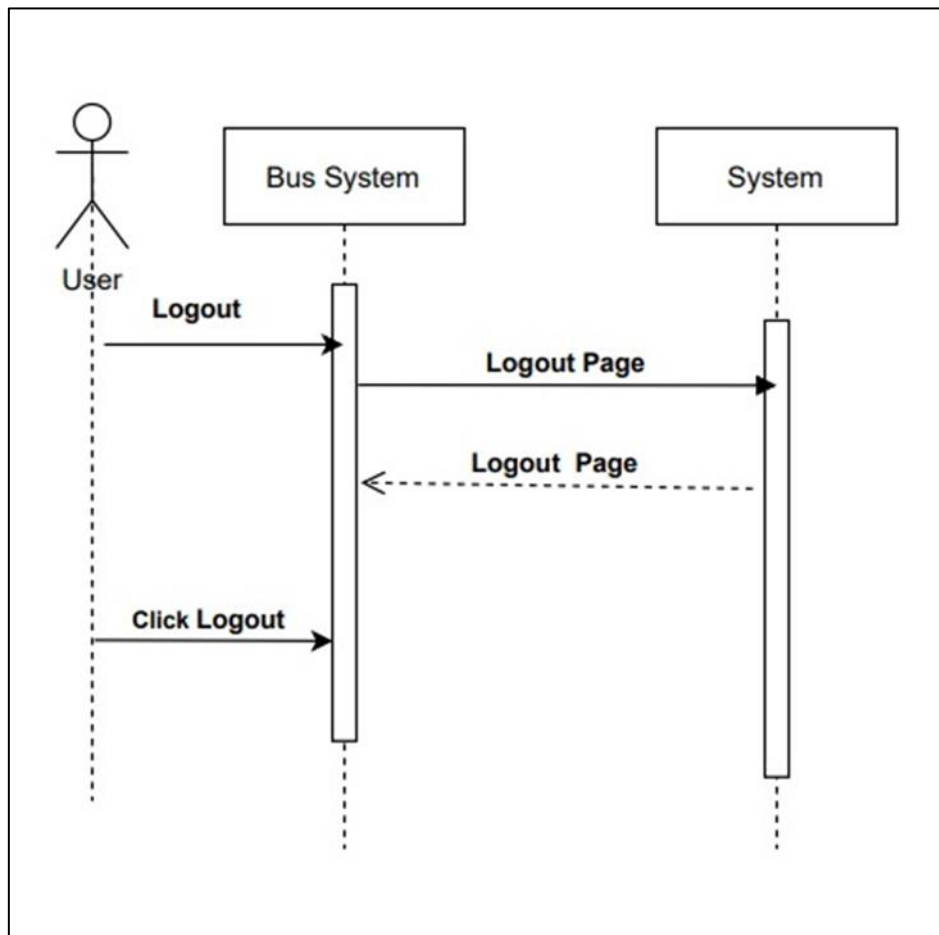


Figure 4.8 Logout Sequence Diagram

#### 4.5 Activity Diagram

An activity diagram is a complicated process that expresses the phase steps of the required requirements. Although regarded to be a more advanced version of the traditional flow chart, the activity diagram also depicts the flow of control. The sequence diagram we saw in our prior lesson is comparable to the activity diagram. However, in contrast to a sequence diagram, an activity diagram focuses on the tasks that must be completed by the items rather than the things themselves.

#### 4.5.1 Booking Ticket Activity Diagram

In the activity diagram of the booking ticket, the user starts with the home page, then selects a ticket that the database will show with the type. After that, an option will appear to either reserve a ticket or book a ticket. When the user selects, the database will show the ticket, and the diagram will end.

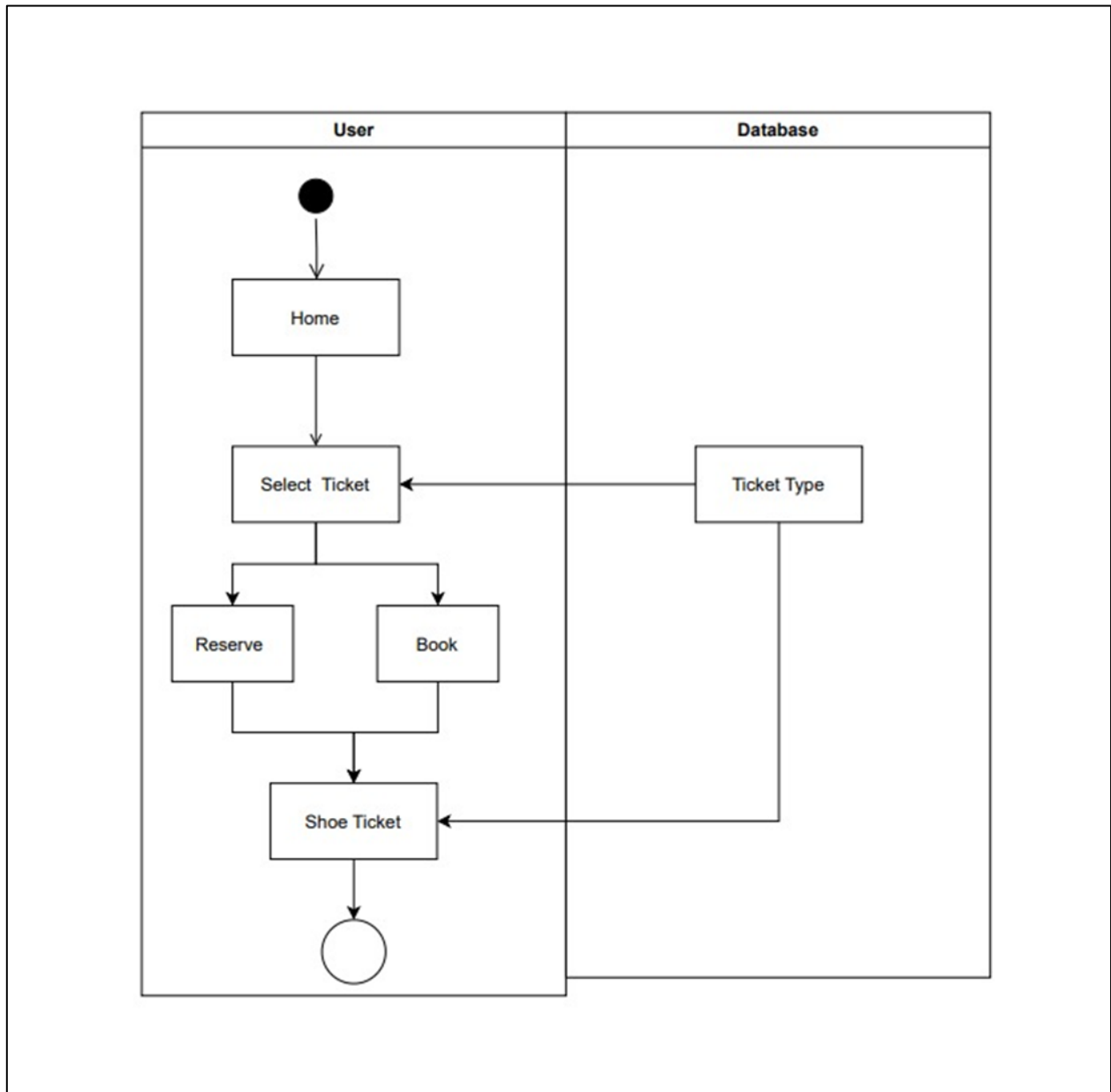


Figure 4.9 Booking Ticket Activity Diagram

### 4.5.2 Bus Route Activity Diagram

In the activity diagram of the bus route, the user starts with the home page, then selects a route that the database will show its location, then the user opens their GPS, and the database will show the route and where it ends.

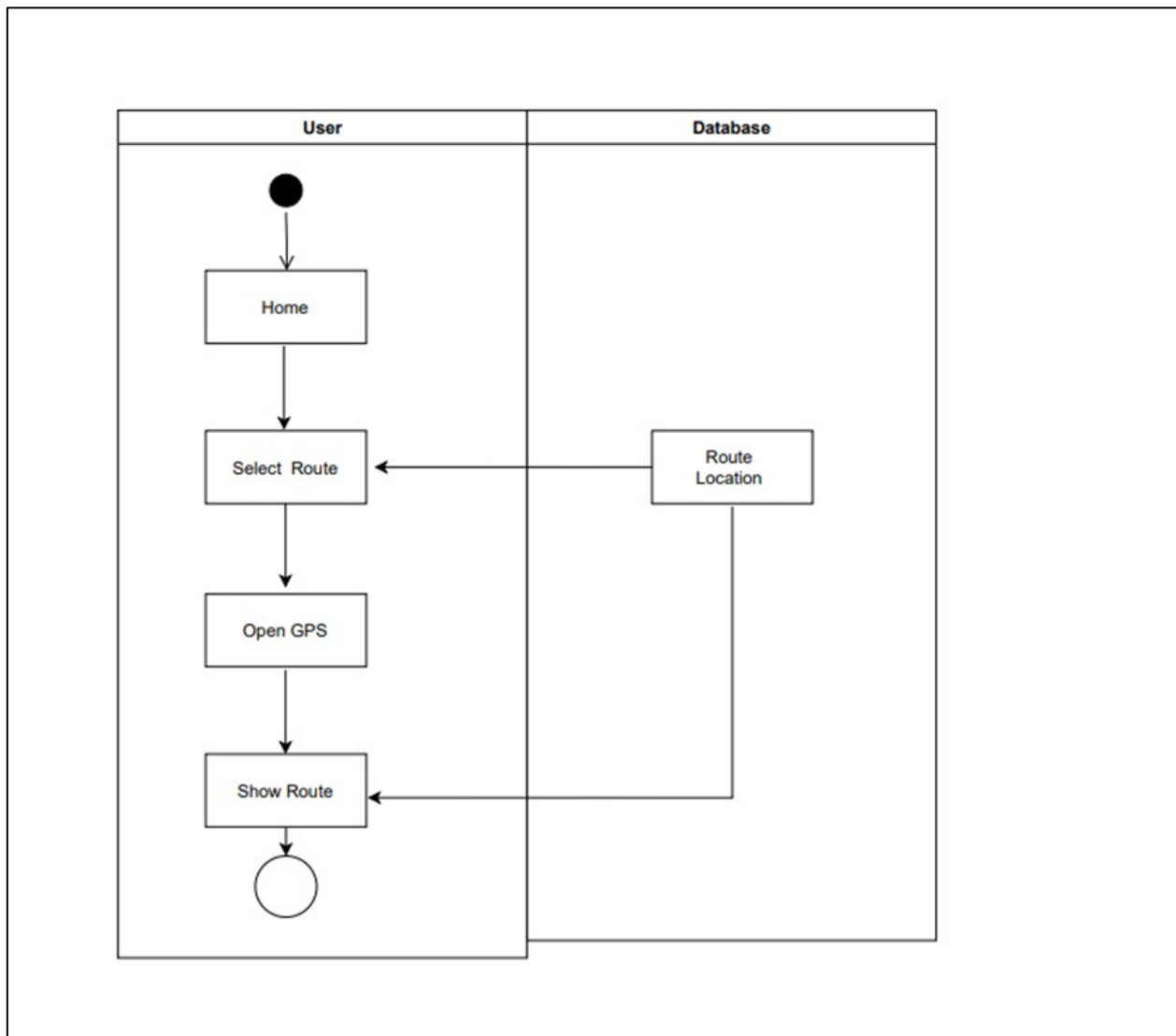



Figure 4.10 Bus Route Activity Diagram

## 4.6 Data Dictionary

Table	Attribute	Type	Length	PK/FK	Null
User	u_id	int	20	PK	No
	u_name	varchar	250		No
	u_email	varchar	250		No
	u_password	varchar	250		No
Admin	a_id	int	20	PK	No
	a_email	varchar	250		No
	a_role	varchar	250		No
Bus driver	d_id	int	20	PK	No
	d_code	int	20	FK	No
Route	r_id	int	20	PK	No
	r_road	varchar	250		No
	r_location	varchar	250		No
	r_date	date	20		No
Feedback	f_id	int	20		No
	f_text	varchar	250		No
Payment	p_id	int	20	PK	No
	p_date	date	20		No

## 4.7 Interface Design



USERNAME


PASSWORD

LOGIN

[Forget Password?](#)

SIGN UP

Figure 4.11 Login or Sign-up Page



Full Name

Email

Phone Number

Password

Confirm Password

**SIGN UP**

Figure 4.12 Sign up page

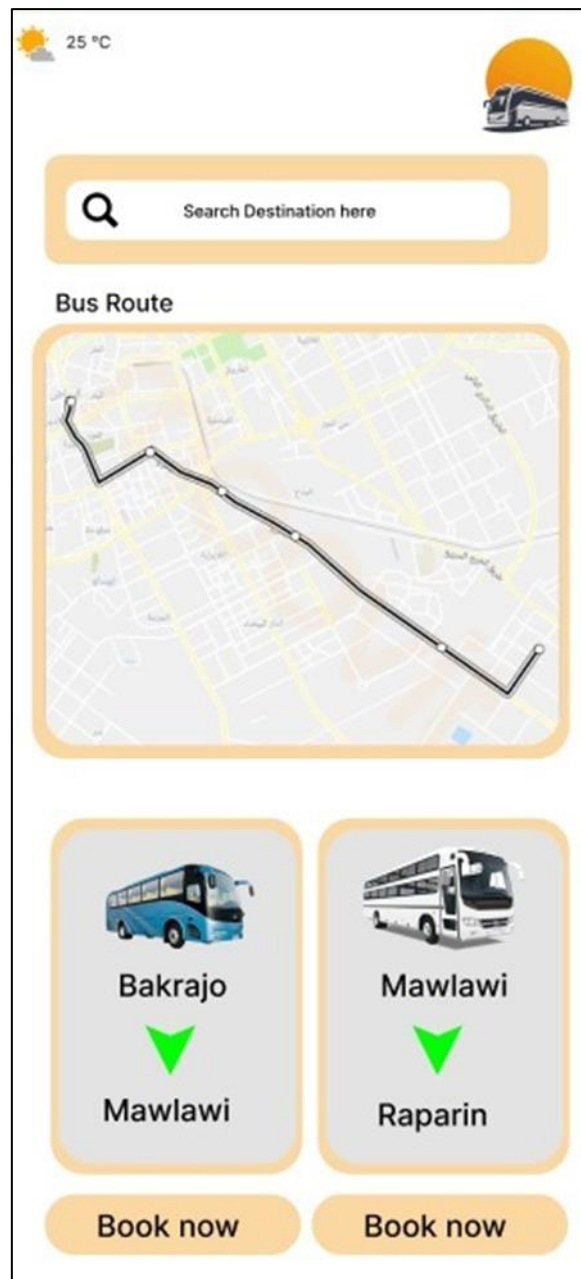





Figure 4.13 Home Page






Figure 4.14 Payment Page



Payment








 €375.00 EUR 


English 

### PayPal Guest Checkout


We don't share your financial details with the merchant.

Country/Region  
United States 



Card number 

Expires

CSC 

First name

Last name

**Billing address**

Street address

Apt., ste., bldg.

City

Figure 4.15 Payment Via PayPal page

36

## **4.8 Chapter Summery**

This chapter discussed the diagrams, which are use-case diagrams that describe how the actors connect with each other in order to work together. After that, a sequence diagram is presented, which explains the object of the user and how it sends requests to the system. Then there is the activity diagram, which explains when the function starts and when it ends. The data dictionary was also explained, along with which attributes have a primary key that cannot be changed. Lastly, the interface design shows how the user will use the application before it is developed.

## **CHAPTER 5**

### **IMPLEMENTATION AND TESTING**

#### **5.1 Introduction**

In this chapter, we will go over the procedures for implementing and testing the University Finder Application, and we will also demonstrate some instances of the primary website code components. This chapter will cover not just the development and testing processes, but also the testing and coding processes.

#### **5.2 Coding of System Main Function**

The flutter dart programming language was used to develop this application. Including both the front-end and the back-end code together with the Android Studio tools and Firebase as the database platform.

##### **5.2.1 Register**

Register is the first step that the user needs to do before they use any other function.

##### **5.2.2 Login**

The user needs to login to the application using the register form that has been sent to the database.

### **5.2.3 Booking Ticket**

One of the main functions that the user can do is booking a ticket, first the system shows the ticket types then the user books a ticket in their mind.

### **5.2.4 Apply**

In this function, the user can see where the bus is heading online using GPS.

## **5.3 Interfaces of System Main Function**

This system's user interface is built to function exactly like the prototype did, complete with a straightforward organization that is simple enough for anyone to comprehend in a short amount of time. Everything, from the color to the content, appears to be an exact replica of the prototype.

## **5.4 Testing**

### **5.4.1 Black Box Testing**

In the method of software testing known as "black box testing," the tester does not have access to any of the internal details of the object that has to be tested, such as its structure, design, or implementation. The analysis of software functioning, as opposed to the internal workings of the system, is the primary focus of the "black box testing" technique, which is a software testing method.

A black-box tester has no understanding of how a software system actually functions on the inside. Testing that takes place at a high level and focuses on the

operation of the software is known as black box testing. It includes testing conducted from the perspective of an outsider or an end-user.

#### **5.4.2 White Box Testing**

A type of testing known as white-box testing examines the operations that take place behind the scenes within a system. This type of testing determines whether or not certain code statements, branches, pathways, or circumstances have been tested. Testing at a low level is referred to as white-box testing.

## **CHAPTER 6**

### **CONCLUSION**

#### **6.1 Introduction**

This project explores the development of the bus management system, which was inspired by the requirements of commuters who rely on buses and the many transportation problems they face. Their biggest issue has been that individuals consistently come late for work or school because there are either no buses available, all of the tickets have been purchased, or the bus may arrive at the bus terminal after it has already left. But after examining the aforementioned problems, our application will address their issue by offering features like purchasing or reserving a ticket so they may board the bus as soon as they come to the terminal. When the application is finished, it will have access to all of these features.

#### **6.2 Achievement of Project Objectives**

The biggest achievement that we have made so far for this project is that we had the chance to resolve all the requirements that had been gathered from the community that we are targeting and to analyze the issues to remake it into an application that the user could easily use.

#### **6.3 Suggestions For Future Improvement**

People want the program to include a function for traveling, according to the ideas we are receiving from the needs that have been gathered; this request has been made frequently by both domestic and international users. We thus want to offer buses

in the future so that users may travel solely through applications rather than by filling out any paperwork by hand.

## REFERENCES

“Bus Management System Project Idea for Final Year.” *LovelyCoding.org*,  
[www.lovelycoding.org/bus-management-system](http://www.lovelycoding.org/bus-management-system)

. Accessed 14 Jan. 2023.

“AusRail 2022 Highlights From Trapeze.” *Bus Transport Management Software Solutions L Trapeze Group*, [www.trapezegroup.com.au/bus](http://www.trapezegroup.com.au/bus)

. Accessed 14 Jan. 2023.

“BusLogic - Leading Bus Ticket Booking Solution.” *BusLogic*, [bus-ticketing-system.com/en/index.php](http://bus-ticketing-system.com/en/index.php)

. Accessed 14 Jan. 2023.

“Bus Ticketing Software | Bus Booking System.” *Bus Ticketing Software / Bus Booking System*, [www.codewareltd.com/productlink/bus-ticket-booking-system](http://www.codewareltd.com/productlink/bus-ticket-booking-system).

Accessed 14 Jan. 2023.

“Bus Ticketing Software | Bus Booking System.” *Bus Ticketing Software / Bus Booking System*, [www.codewareltd.com/productlink/bus-ticket-booking-system](http://www.codewareltd.com/productlink/bus-ticket-booking-system)

. Accessed 14 Jan. 2023.

## **APPENDIX A**

### **SOFTWARE REQUIREMENT SPECIFICATION (SRS)**

#### **1.1 Introduction**

The process of writing out all the system and user needs in the form of a document is known as requirement specification, also known as documentation. These standards must be unambiguous, exhaustive, thorough, and consistent. We collect all the needs during the capturing action from multiple sources. We examine and comprehend those criteria throughout the analytical and negotiating processes. We must now create a formal paper outlining those specifications. The required specification is just that. To be more explicit, it is the process of clearly and accurately describing all of the requirements and limitations of the user and the system. The bus management system will define all the use cases and mark each move the users make to avoid any future issues when the application is published.

#### **1.2 Scope**

The bus management system is a system that will concentrate on the people who take buses on a daily basis. This project will help both citizens and bus employees get things done easily with the help of our application. This system can be summarized into some points which are:

1. This system will be available free.
2. This system will allow users to register and login.
3. This system will allow users to book or reserve tickets.
4. This system will allow users to see the bus route.

5. This system will allow users to send feedback.

### **1.3 Definitions, Acronyms and Abbreviation**

Table 0.1 Definitions, Acronyms and Abbreviation

SRS	Software Requirement Specification
-----	------------------------------------

### **1.4 Overview**

This overview of the SRS will be divided into two parts. The first part is an introduction that contains the purpose, scope, definitions, acronyms, and abbreviations, as well as an overview. The second part talks about the overall description, which is the use case diagrams, system interfaces that contain hardware and software interfaces, and so on.

## 2. Overall Description

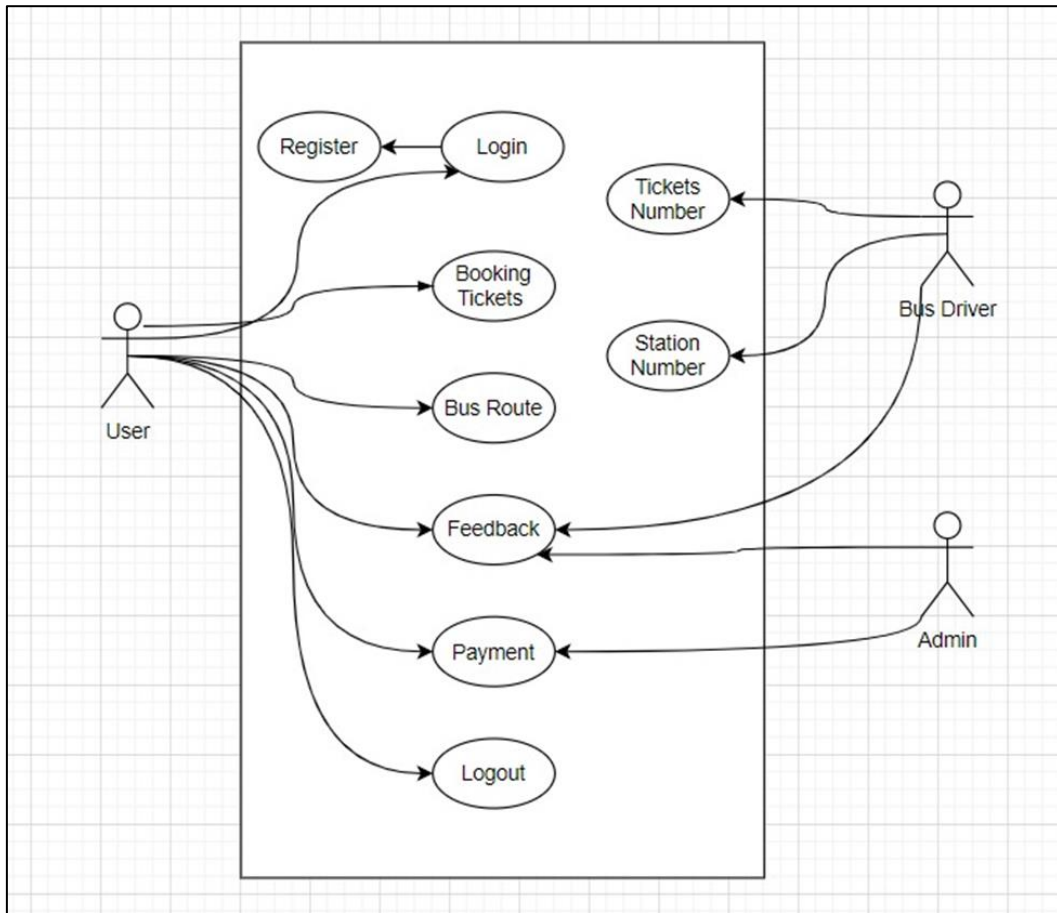


Figure 0.1 Use Case

### 2.1 Product Perspective

This project is helpful for city people who travel with buses on a regular basis, particularly those who use these transportation methods to get to work and often arrive late. This application will assist users in purchasing tickets or learning when to expect them, allowing them to come on time and without wasting any time.

#### 2.1.1 System Interfaces

All types of actors in the system have their own unique features and their own interfaces for their modules, and the bus management system's user interface has been designed to please all users. While the UI appears straightforward, anybody may use

it. The four modules available to users of the customer type are ticket booking, bus route, payment, and feedback. While for the bus driver they can control Login, feedback, ticket number, and station numbers. The final two admin modules are payment and login.

### **2.1.2 Hardware Interfaces**

- Processor: intel(R), Core (TM), i5-8350U @ 1.70GHz, 1.90 GHz
- Random Access Memory: 3 GB
- Hard Drive Capacity: 255 GB
- Operating system architecture: 36-bit
- Input device: Touch Screen
- Output device: Mobile screens

### **2.1.3 Software Interfaces**

- Operating System: Windows and IOS
- Integrated Development Environment: Microsoft Visual Studio
- Database Management System: XAMPP Sql
- Web Browser: Google Chrome, Microsoft Edge
- Visual Modelling & Design Tool: Draw.io, Creately
- High Fidelity Prototype: Figma

## 2.2 Product Functions

Table 0.2 Use Case Description

Use Case	Description
Register	Register is the first step that the user needs to do before they use any other function.
Login	The user needs to login to the application using the register form that has been sent to the database.
Booking tickets	One of the main functions that the user can do is booking a ticket, first the system shows the ticket types then the user books a ticket in their mind.
Bus route	In this function, the user can see where the bus is heading online using GPS.
Feedback	Feedback is a way that the user and the bus driver can contact the admins through the system. And only the admin sees the text.
Payment	After the user books a ticket they need to pay for the ticket through the system then the admin sends it to the bank to send the receipt to the user.
Logout	The user can log out if they want to.

## 2.3 User Characteristic

Table 0.3 User Characteristic

Actor	Role
User	The user is the most important actor in the system and has the most roles that can be used like register, login, booking tickets, watching bus routes, giving feedback, paying for tickets and logout.
Admin	The admin is the actor who can take the feedback and send the payment to the bank system.
Bus Driver	The bus driver can also login from accounts that have been given to from the system so that they can see the ticket number, station number and give feedback.

## 2.4 Constraints

- **Performance:** The performance of the application needs to be smooth.
- **Usability:** The system needs to have an interface that everyone can use.
- **Availability:** The system should be available for free.

### 3. Specific Requirement

#### 3.2.1 Use Case for Register

Table 0.4 Use Case for Register

Use Case Name	Register
Use Case ID	UC001
Actors	User
Description	This use case tells how the user registers in the bus management system.
Pre-Condition	<ol style="list-style-type: none"><li>1. The user enters the application.</li><li>2. Fill in email and password.</li><li>3. Click Register.</li></ol>
Normal Flow	<ol style="list-style-type: none"><li>1. Enter the application.</li><li>2. Go to the Register page.</li><li>3. Enter email and password.</li><li>4. Click the Register button.</li></ol>
Exception	Error message is shown to the user.
Post-Condition	<ol style="list-style-type: none"><li>1. Users successfully Register into the application.</li><li>2. Directed to the first page.</li></ol>

### 3.2.2 Use Case for Login

Table 0.5 Use Case Description for Login

Use Case Name	Login
Use Case ID	UC002
Actors	User
Description	This use case tells how the user log in into the bus management system.
Pre-Condition	<ol style="list-style-type: none"><li>1. The user enters the application.</li><li>2. Fill in email and password.</li><li>3. Click Login.</li></ol>
Normal Flow	<ol style="list-style-type: none"><li>1. Enter the application.</li><li>2. Go to the Login page.</li><li>3. Enter email and password.</li><li>4. Click the Login button.</li></ol>
Exception	Error message is shown to the user.
Post-Condition	<ol style="list-style-type: none"><li>1. Users successfully Login into the application.</li><li>2. Directed to the first page.</li></ol>

### 3.2.3 Use Case for Booking Ticket

Table 0.6 Use Case Description for Booking Ticket

Use Case Name	Booking Ticket
Use Case ID	UC003
Actors	User
Description	This use case tells how the user books a ticket in the bus management system.
Pre-Condition	<ol style="list-style-type: none"><li>1. The user enters the application.</li><li>2. Fill in email and password.</li><li>3. Click Register.</li></ol>
Normal Flow	<ol style="list-style-type: none"><li>1. Enter the application.</li><li>2. Go to the Book ticket page.</li><li>3. Choose a ticket type.</li><li>4. Click the Book button.</li></ol>
Exception	Error message is shown to the user.
Post-Condition	<ol style="list-style-type: none"><li>1. Users successfully Book tickets into the application.</li><li>2. Directed to the first page.</li></ol>

### 3.2.4 Use Case for Bus Route

Table 0.7 Use Case Description for Bus route

Use Case Name	Bus Route
Use Case ID	UC004
Actors	User
Description	This use case tells how the user sees the bus route in the bus management system.
Pre-Condition	<ol style="list-style-type: none"><li>1. The user enters the application.</li><li>2. Fill in email and password.</li><li>3. Click Login.</li></ol>
Normal Flow	<ol style="list-style-type: none"><li>1. Enter the application.</li><li>2. Go to the bus route.</li><li>3. Open GPS location.</li></ol>
Exception	Error message is shown to the user.
Post-Condition	<ol style="list-style-type: none"><li>1. Users successfully Open GPS in the application.</li><li>2. Directed to the first page.</li></ol>

### 3.2.5 Use Case for Feedback

Table 0.8 Use Case Description for feedback

Use Case Name	Feedback
Use Case ID	UC005
Actors	User and bus driver
Description	This use case tells how the user and bus driver send feedback in the bus management system.
Pre-Condition	<ol style="list-style-type: none"><li>1. The user enters the application.</li><li>2. Fill in email and password.</li><li>3. Click Login.</li></ol>
Normal Flow	<ol style="list-style-type: none"><li>1. Enter the application.</li><li>2. Go to the feedback section.</li><li>3. Send text.</li></ol>
Exception	Error message is shown to the user.
Post-Condition	<ol style="list-style-type: none"><li>1. Users successfully Send feedback in the application.</li><li>2. Directed to the first page.</li></ol>

## **APPENDIX B**

### **SOFTWARE DESIGN DESCRIPTION (SDD)**

#### **1. Introduction**

A software design document, sometimes referred to as a software design specification or technical specification document, is a report that details the overall architecture of a software product. These design documents are often created by project managers or software designers and delivered to the software development team to give them a general understanding of what needs to be done and how. The Software Design Document (SDD) is a document that helps the bus management system to clarify and helps in understanding the software design.

##### **1.1 Scope**

The Software Design Description chapter will talk about the introduction, scope, definitions, acronyms, and abbreviations with the overview of the chapter. Also, it will talk about the entity relationship diagram (ERD) and the data dictionary, which explain the attributes of the table and the user interface that shows how the system was before its last launch.

##### **1.2 Definitions, Acronyms and Abbreviation**

Table 0.9 Definitions, Acronyms and Abbreviation

SDD	Software Design Description
-----	-----------------------------

## 1.3 Overview

With an overview of the chapter, the introduction, scope, definitions, acronyms, and abbreviations are discussed in the chapter on software design descriptions. Additionally, it will discuss the entity relationship diagram (ERD), the data dictionary, and the user interface, which describe the characteristics of the table and illustrate how the system was before its most recent introduction.

## 2. Entity Relationship Diagram

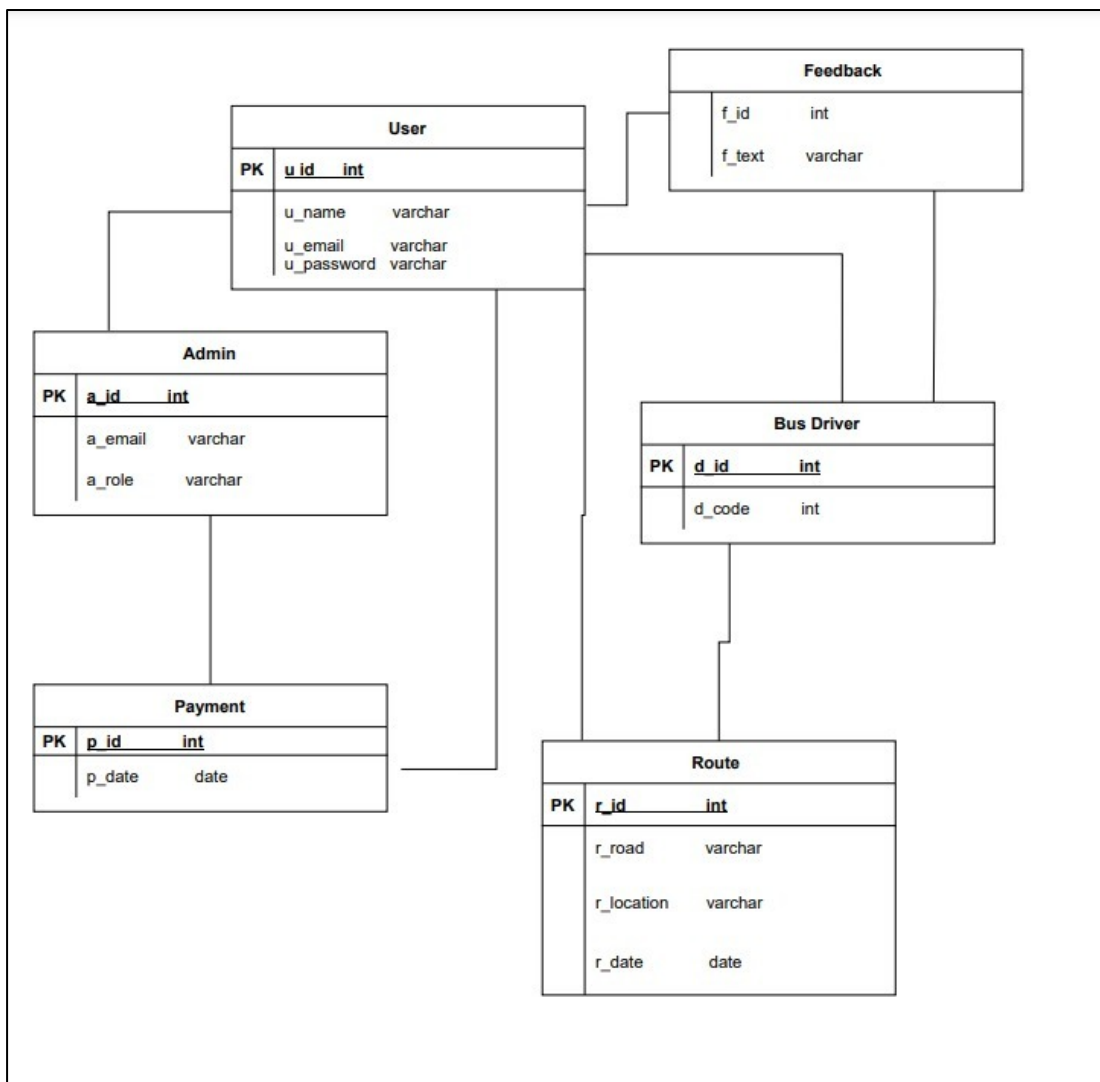



Figure 0.2 Entity Relationship Diagram

### 3. Data Dictionary

Table 0.10 Data Dictionary

Table	Attribute	Type	Length	PK/FK	Null
User	u_id	int	20	PK	No
	u_name	varchar	250		No
	u_email	varchar	250		No
	u_password	varchar	250		No
Admin	a_id	int	20	PK	No
	a_email	varchar	250		No
	a_role	varchar	250		No
Bus driver	d_id	int	20	PK	No
	d_code	int	20	FK	No
Route	r_id	int	20	PK	No
	r_road	varchar	250		No
	r_location	varchar	250		No
	r_date	date	20		No
Feedback	f_id	int	20		No
	f_text	varchar	250		No
Payment	p_id	int	20	PK	No
	p_date	date	20		No

#### 4. Interface Design



Full Name

Email


Phone Number

Password

Confirm Password

**SIGN UP**

Figure 0.3 Register Interface



USERNAME

PASSWORD

LOGIN

[Forget Password?](#)

SIGN UP

Figure 0.4 Login Interface

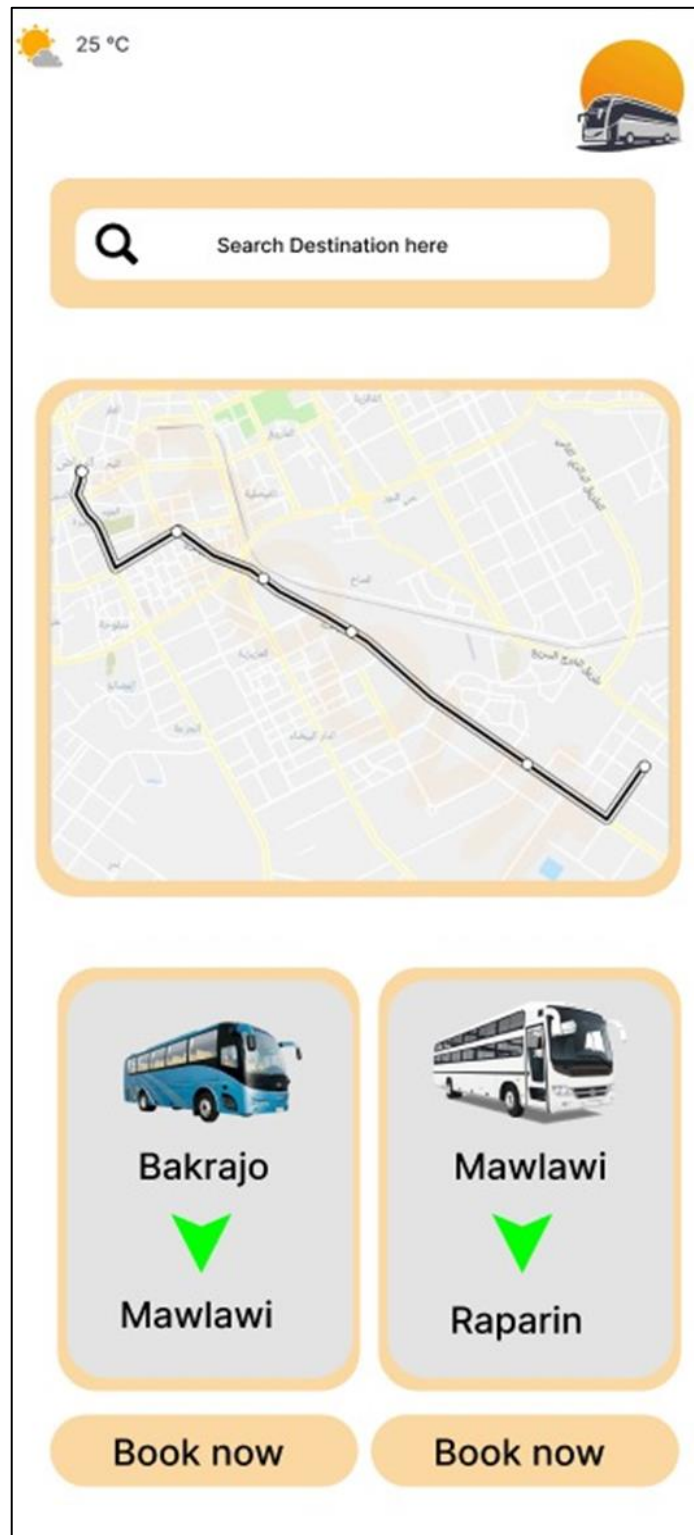


Figure 0.5 Home Page interface

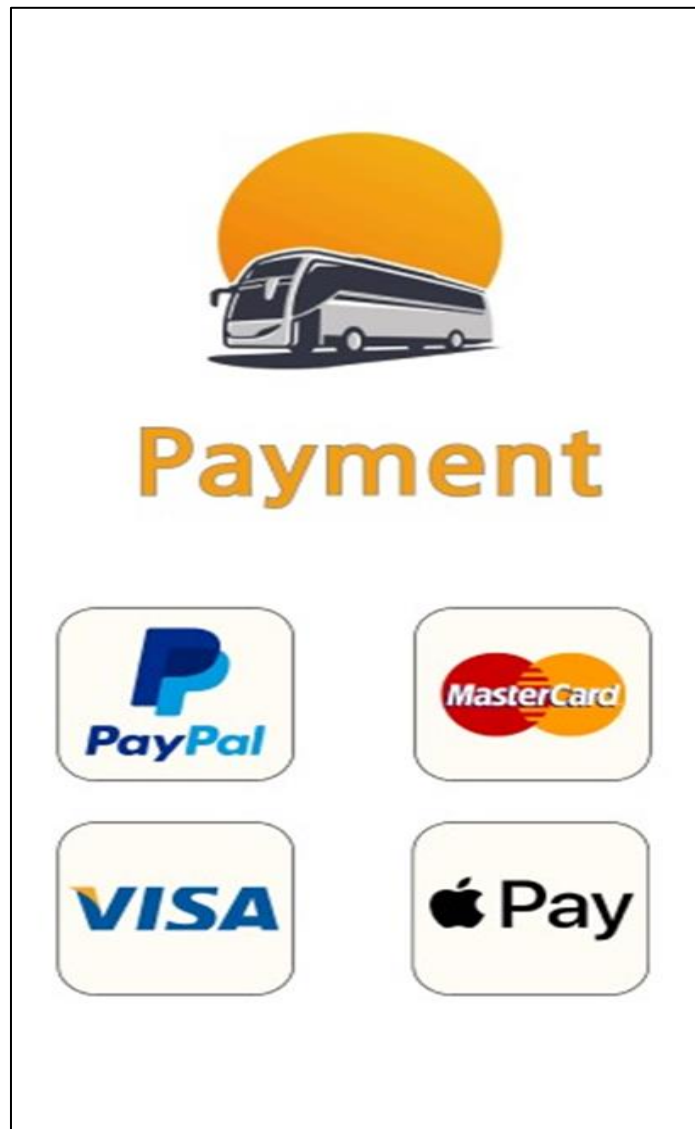







Figure 0.6 Payment interface




# Payment










 €375.00 EUR 


English 

## PayPal Guest Checkout


We don't share your financial details with the merchant.

Country/Region  
United States 



Card number 

Expires

CSC 

First name

Last name

**Billing address**

Street address

Apt., ste., bldg.

City

Figure 0.7 Payment via paypal Interface

## **APPENDIX C**

### **SOFTWARE TESTING DOCUMENT (STD)**

#### **1. Introduction**

Software testing is a procedure to assess a software application's functioning with the aim of determining if the generated software matches the required specifications or not and to uncover any flaws to guarantee that the product is free of errors and produces a high-quality result. Software testing will help the bus management system to test the functions of the system before launching the system to avoid any future errors.

##### **1.1 Purpose**

Each stage of the development of a software product, especially a website, is crucial, but system testing is extremely crucial. Before launching the website, it is crucial to test it to ensure that everything functions perfectly. We tested this website using three techniques during development: user acceptability testing, white box testing, and black box testing.

##### **1.2 Definitions, Acronyms and Abbreviation**

Table 0.11 Abbreviation and Definitions

STD	Software Testing Document
-----	---------------------------

## 1.3 Overview

The overview of the Software Testing Document chapter will describe the introduction in a way that clarifies how the system should be tested before launching it to the users. The methods that will be tested for the bus management system is use case.

## 2. Testing

### 2.1 Use Case Testing

Table 0.12 Use Case Description for Register

Use Case Name	Register
Use Case ID	UC001
Actors	User
Description	This use case tells how the user registers in the bus management system.
Pre-Condition	<ol style="list-style-type: none"><li>1. The user enters the application.</li><li>2. Fill in email and password.</li><li>3. Click Register.</li></ol>
Normal Flow	<ol style="list-style-type: none"><li>1. Enter the application.</li><li>2. Go to the Register page.</li><li>3. Enter email and password.</li><li>4. Click the Register button.</li></ol>
Exception	Error message is shown to the user.
Post-Condition	<ol style="list-style-type: none"><li>1. Users successfully Register into the application.</li><li>2. Directed to the first page.</li></ol>

### 2.2.2 Use Case for Login

Table 0.13 Use Case Description for Login

Use Case Name	Login
Use Case ID	UC002
Actors	User
Description	This use case tells how the user log in into the bus management system.
Pre-Condition	<ol style="list-style-type: none"><li>1. The user enters the application.</li><li>2. Fill in email and password.</li><li>3. Click Login.</li></ol>
Normal Flow	<ol style="list-style-type: none"><li>1. Enter the application.</li><li>2. Go to the Login page.</li><li>3. Enter email and password.</li><li>4. Click the Login button.</li></ol>
Exception	Error message is shown to the user.
Post-Condition	<ol style="list-style-type: none"><li>1. Users successfully Login into the application.</li><li>2. Directed to the first page.</li></ol>

### 2.2.3 Use Case for Booking Ticket

Table 0.14 Use Case Description for Booking Ticket

Use Case Name	Booking Ticket
Use Case ID	UC003
Actors	User
Description	This use case tells how the user books a ticket in the bus management system.
Pre-Condition	<ol style="list-style-type: none"><li>1. The user enters the application.</li><li>2. Fill in email and password.</li><li>3. Click Register.</li></ol>
Normal Flow	<ol style="list-style-type: none"><li>1. Enter the application.</li><li>2. Go to the Book ticket page.</li><li>3. Choose a ticket type.</li><li>4. Click the Book button.</li></ol>
Exception	Error message is shown to the user.
Post-Condition	<ol style="list-style-type: none"><li>1. Users successfully Book tickets into the application.</li><li>2. Directed to the first page.</li></ol>

#### 2.2.4 Use case for Bus Route


Table 0.15 Use Case Description for Bus route

Use Case Name	Bus Route
Use Case ID	UC004
Actors	User
Description	This use case tells how the user sees the bus route in the bus management system.
Pre-Condition	<ol style="list-style-type: none"><li>1. The user enters the application.</li><li>2. Fill in email and password.</li><li>3. Click Login.</li></ol>
Normal Flow	<ol style="list-style-type: none"><li>1. Enter the application.</li><li>2. Go to the bus route.</li><li>3. Open GPS location.</li></ol>
Exception	Error message is shown to the user.
Post-Condition	<ol style="list-style-type: none"><li>1. Users successfully Open GPS in the application.</li><li>2. Directed to the first page.</li></ol>

### 2.2.5 Use Case for Feedback

Table 0.16 Use Case Description for feedback

Use Case Name	Feedback
Use Case ID	UC005
Actors	User and bus driver
Description	This use case tells how the user and bus driver send feedback in the bus management system.
Pre-Condition	<ol style="list-style-type: none"><li>1. The user enters the application.</li><li>2. Fill in email and password.</li><li>3. Click Login.</li></ol>
Normal Flow	<ol style="list-style-type: none"><li>1. Enter the application.</li><li>2. Go to the feedback section.</li><li>3. Send text.</li></ol>
Exception	Error message is shown to the user.
Post-Condition	<ol style="list-style-type: none"><li>1. Users successfully Send feedback in the application.</li><li>2. Directed to the first page.</li></ol>

	QAIWAN INTERNATIONAL UNIVERSITY	Form No.: LIB003
	FINAL THESIS SUBMISSION FORM	Edition: 01 Effective Date: 01/ 06/ 2024 Page (s): 03

**Section 1 (To be completed by Student)**  
 (Please tick (✓) where applicable)

Library  
 Qaiwan International University

**Submission of Final Copies of Thesis**

I MUSSA SARKAWT ABDUL (name) hereby submit:  
 the final copies of my thesis for the degree of ☒ Bachelor / ☐ Master / ☐ Doctor of Philosophy

**The title of the thesis is:**

BUS Management System

Hence, I declare that:

- ☒ i) My thesis has been reviewed by my Main Supervisor / HoD / Dean / Faculty
- ☒ ii) I had submitted the CD of my softcopy of the thesis (the contents are similar with the hardcopy of the thesis) in a single PDF file.
- ☒ iii) I had submitted the Degree Scroll Information Form.

Student Name: MUSSA SARKAWT ABDUL

Department: Software Engineering

Signature: [Signature]

Date: 27 Aug - 2024

## ENDORSEMENT BY MAIN SUPERVISOR

Section 2 (To be completed by Main Supervisor)

Supervisor's Name: ..... Mh. Salam .....  
Department: ..... S-E .....  
I have examined the thesis of Mr. /Mrs. /Ms.: ..... Musa .....

I hereby confirm that all corrections and amendments made to the thesis have been rectified by the candidate.

(Main Supervisor's Signature & Stamp)

27/8/2024

(Date)

## ENDORSEMENT BY Librarian

Section 3 (To be completed by Librarian)

Librarian's Name: ..... Dilshwar Othman .....  
☒ Approved ☐ Not Approved

I hereby confirm that all corrections and amendments made to the thesis have been rectified by the candidate.

(Librarian's Signature & Stamp)

27/8/2024  
(Date)

**STUDENT'S DECLARATION**  
**Section 4 (To be completed by Student)**

**Student Thesis's Copyright and University's Intellectual Property**

*"The copyright to a thesis belongs to the student. However, as a condition of being awarded the degree, the student hereby grants to the University, a free, ongoing, non-exclusive right to use the relevant work and/or thesis for the University's teaching, research and promotional purposes as well as free and the non-exclusive right to retain, reproduce, display and distribute a limited number of copies of the thesis, together with the right to require its publication for further research and archival use."*

I declare that the contents presented in this thesis are my own which was done at Qaiwan International University unless stated otherwise. The thesis has not been previously submitted for any other degree.

I also declare that my thesis has been reviewed by the Main Supervisor / HoD / Dean / Faculty the comments are as stated in Section 2 (page 2).

Name of Student: ~~Abdul~~ MUSSA Sarkawt Abdul

Signature of Student: 

IC. No. / Passport No.: A17007890

Date: 27-Aug-2024