UNIVERSITY OF DAR ES SALAAM

COLLEGE OF INFORMATION AND COMMUNICATION TECHNOLOGIES

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

IS335/CS498: FINAL YEAR PROJECT REPORT

END OF SEMESTER TWO

2022/2023

PROJECT TITLE: SCHOOL BUS TRACKING MANAGEMENT SYSTEM (MOBILE APP)

STUDENT'S NAME:	Mwaisela, Peter Abel
REGISTRATION NUMBER:	2020-04-08600
PROGRAM:	BSc In Business Information Technology
SUPERVISOR'S NAME:	Mr. Marco Masembo

DECLARATION

I certify this report and the work it describes are my original creations, with any explicit acknowledgment or citation of the contributions of others.

I further declare that the work contained in this report was completed in accordance with University of Dar es Salaam regulations and that it was not sent to another university for review, either in Tanzania or abroad.

This report may be submitted for consideration for the University of Dar es Salaam's award of a BSc in Business Information Technology.

Student's signature:

……

Mwaisela, Peter Abel

Supervisor's signature

……

Mr. Marco Masembo

ABSTRACT

This paper presents a system to monitor the pick-up/drop-off school children to enhance the safety of children during the daily transportation from and to school. It intends to reduce the waiting time of parents and their children waiting for school bus, thereby to stimulate the sharing of real time information about the bus between the bus driver and parent.

The system is a mobile based and web based application. The mobile version of the system is designed for use by the parents and drivers, while the web version of the system is designed for use by the school admin.

This paper consists of six chapters, introduction, literature review, methodology, system analysis and design, implementation, recommendation and conclusion. The general methodology used in this project is waterfall methodology and for data collection unstructured interviews was used.

ACKNOWLEDGEMENT

Since authoring this work has not been simple, I would like to take this opportunity to express my gratitude to the Almighty God.

Additionally, we would like to express our sincere gratitude to our supervisor, Mr. MARCO MASEMBO, who worked tirelessly with us to guide and direct us in the proper course of action. And all of that has been sufficient, for which I am sincerely grateful.

Additionally, I want to thank Dr. Joseph Cosmas, the coordinator of our senior project, for his support and involvement in the writing of this report. Along with that, I'd like to express my gratitude to the University of Dar es Salaam's administration and the College of Information and Communication Technology (CoICT). That gave us the opportunity to complete this project, which will be extremely helpful to us as University of Dar es Salaam students. I'm very happy about that.

TABLE OF CONTENTS

LIST OF ABBRIVIATION.

API	Application Programming Interface
COICT	College of information and communication technology.
DBMS	Database Management System
GPS	Global positioning system.
SDLC	System development life cycle
UDSM	University of Dar es Salaam
NODE MCU	NODE microcontroller Unit

LIST OF FIGURES AND CODE SNIPPETS

LIST OF TABLES

CHAPTER ONE
INTRODUCTION

1.1 General introduction.

When it comes to the issue of transportation, time and patience are more essential. In other words many people especially those who use the public transport buses have experienced time loss because of various delays that happen. Millions of children need to travel from home to school every day. Thus most parents are striving to get a safe transport for their children because there have been an increase in crimes against children.

The safety of children has become a major concern for many parents in recent years, most parents are working,this becomes a challenge to take their children to school and later to bring them home. Here the bus transport provided by the school comes into picture, due to safety reasons, this has caused them to worry much about their children's safety on school buses. They are constantly concerned about the arrival time of their children's bus for pick-up and drop-off. Have children boarded the bus? Where is the bus taking their kids? Is the arrival of the kids delayed in any way?

The increased use of school buses has posed a great pressure to the school administration to come up with better ways to manage students when they use school transport. Since these school buses are used by students of very young age, there might be some risks that may happen having no one to blame.

In some situations, school managers employ new drivers who are not familiar with the students' pick-up and drop-off points thus they end up dropping students at points which are far from their homes or even going to routes that they were not supposed to go causing huge fuel consumption.

1.2 Statement of the Problem

The school buses take long to arrive at students' pick-up point causing parents to wait longer with their kids but this situation may be caused by various reasons such as break-down of the school bus or sometimes may be due to new drivers who are not familiar with the routes. Sometimes the delay can be caused by parents' negligence to reach at pickup point in right time, this can lead driver to wait for them and delay to reach to the pick up points of other students in right time.

1.3 Objective

1.3.1 Main Objective.

To develop School Bus Tracking Management System that keeps parents directly informed and notified easy about the movements of their children from home to school and way back home, whilst allowing schools to manage and control emergency situations efficiently. Allow a driver to privately notify school administrator and parents of emergency events directly through customized notifications such as car breakdown emergency.

1.3.2 Specific Objectives.

i. Conducting a survey to identify the need for School Bus Tracking Management System.
ii. Establishing requirements for the School Bus Tracking Management System from stake holders
iii. Designing School Bus Tracking Management System from the gathered requirements.
iv. Implementing the design of the system.

1.4 Significance of the project

i. The system helps to reduce the waiting time of parents and students waiting for the school bus.
ii. The system helps the school administration to monitor the route of the school buses
iii. The system acts as a directive to new drivers to know pickup points of the students
iv. The system gives notification to the parents and school administration about any emergency that has occurred.

1.5 Project scope

The scope of this project is based on nursery and primary school students, parents, school administration and school bus drivers in tracking the school buses and getting real-time information about them. The administrator module is concerned with all administration details, The parent module is concerned with the management of the student details in regards to the school bus assigned to his child, while the driver module is based on the details regarding the real-time information about the vehicle. The system database stores the school Admin details, student details, parents details driver details, routes' details, pickup points of students and the vehicle details.

1.6 Organization of the project

Table 1.1: Organization of Report the Project

Number	Content
1	Cover page
2	Table of contents
3	List of abbreviations and figures
4	Chapter one: Provides a deep introduction about the project by explaining what is about to be done and what are the steps and procedures to be followed so as to implement the project successfully.
5	Chapter two: Provides literature review details, gives the overview of the given topic by considering other related publications regarding that topic.
6	Chapter three: Provides project methodology, where in this chapter the methodology that has been used to implement the project has been discussed in depth by describing it, its significance and the phases associated with it at large.
7	Chapter four: Provides System designs and analysis, where it shows methods used to gather requirements, core functionalities of the system. Requirement analysis which explains about functional and non-functional requirements. Also, it contains system design phase which consist of system architecture design, sequence diagram and entity relation diagram.
8	Chapter five: Provide System implementation of project based on Parent and Driver module (Mobile application) by considering the goals, functional and non-functional outlined in previous sections, along with the system design. Its purpose is to achieve the primary objective and successfully complete the project
9	Conclusion and Recommendation

CHAPTER TWO
LITERATURE REVIEW

2.1 Overview

A literature review involves the process of survey of the major writings and other sources that provides an overview of a selected topic. This chapter consists of different reviews on the concept of this project topic.

2.2. Existing solutions

2.2.1 Bustracker

Bustracker is a mobile and web-based application which focuses on providing parents/guardians with information about the school buses boarded by their children. The literature review was undertaken how this system works. This system helps to track school buses from the parent's mobile phones and receive all real-time information about the school bus, it also offers additional functionalities such as emergency management, and push notifications.

(https://bustracker.co.tz/)

2.2.2 Schoolbus

Schoolbus is a mobile based application where parents and guardians track their children's school buses via SMS and Mobile application. This application has two modules which are driver module and the parent module, it mainly facilitates the communication between the parent and the bus driver. (https://multics.co.tz/products/schoolbus)

2.3 Project Gap

Some core and basic functionalities such as GPS tracking of school buses, Vehicle information management, Mobile alerts, Route management that exist in the current systems will be reused and modified in our system.

Our system will also add other functionalities such as emergency management. Improve user experience as some existing systems are not user friendly, by improving user experience it helps to fulfill the user's need in a smooth way.

CHAPTER THREE
METHODOLOGY

3.1 Overview

Methodology is a formalized approach to implementing the system development life cycle (SLDC) i.e., is the list of steps and deliverables. In this project the software development methodology that is going to be used is Waterfall Development. (Allan Dennis , Barbara Haley WIxom & David Targarden, 2015)

3.2 Waterfall Development

Waterfall Development is a system development methodology in which analysts and users proceed sequentially from one phase to the next. (See Figure 1) The key deliverables for each phase are typically voluminous (often, hundreds of pages) and are presented to the approval committee and project sponsor for approval as the project moves from phase to phase. Once the work produced in one phase is approved, the phase ends and the next phase begins. As the project progresses from phase to phase, it moves forward in the same manner as a waterfall. While it is possible to go backward through the phases (e.g., from design back to analysis), it is quite difficult. (Allan Dennis , Barbara Haley WIxom & David Targarden, 2015)

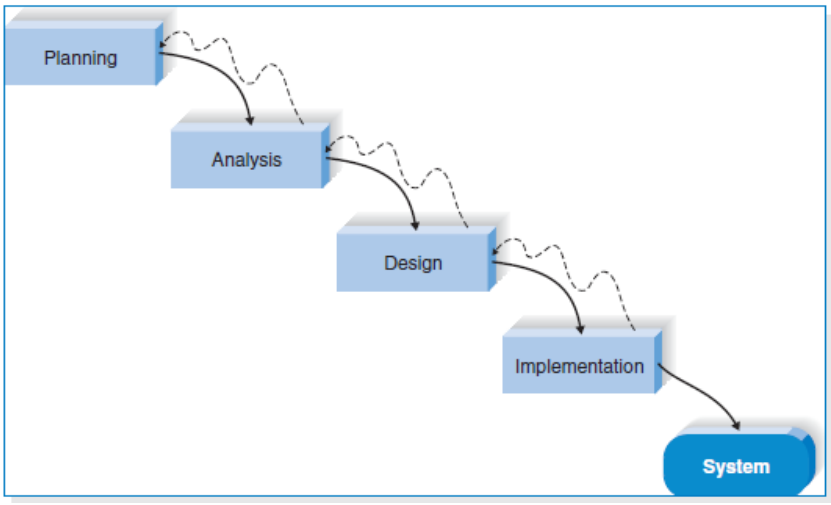

Figure 3.1: Methodology Model

3.3 Significance of Waterfall Development

Waterfall development methodologies have the advantages of identifying requirements long before programming begins this well help to clear understanding of requirements of clients, limiting changes to the requirements as the project proceeds.

Furthermore, in waterfall methodology the process phases distinction is clear since the process proceed sequentially from one phase to the next. Once the work produced in one phase is approved, the phase ends and the next phase begins no interaction between phrases.

3.4 Stages of Waterfall Development

The following tasks are accomplished throughout SDLC of the project;

Waterfall Development breaks the product into small incremental builds.

3.4.1 Requirements gathering and Analysis

Requirements were established from experts and the stakeholders such as parents, school administration and drivers. The requirements were gathered through interviews (open ended questions).

In interviews, we organized some interviews with some parents, school heads and drivers. This method can be used accordingly depending on the type of project you have and the type of information you want to collect.

The team analyzed the data collected during the requirements gathering stage by using object-oriented approach and identified the key challenges and opportunities facing the stakeholders. This focused in identifying functional requirements of system, based on findings from our stake holders

In addition to that our team used the collected data to brainstorm non-functional requirements of the system since it would be difficult for our stakeholders to talk about them.

3.4.2 Design

After clearly understanding the requirements, we planned how to build requirements into a product. That is creating and designing of the system. We produced design diagrams such as the use case diagram, database design, sequence diagram and architecture design of the system. The use case diagram includes the actors and the functionalities that are triggered by the specific user. The database design shows how the data of the different entities of the system is organized and how these entities are related to each other. The sequence diagram shows how the process interaction is arranged in time sequence. Lastly the architecture design shows how the hardware and software components interact and their interfaces. The tools used are draw.io.

3.4.3 Development

In this phase we implemented the design of the system. The implementation of the system involved linking the GPS tracker with a software part of the system. In software part of the system we used flutter framework for the frontend and Firebase for the backend. Moreover, there are some APIs that were used in fetching the data from the GPS tracker as well as the database.

Also, in this phase our project used git as version control for our system. It helped us to keep track of various changes that we made and also organizing our project and team as well. The code editor used is Visual studio, and all of its supported extensions to ensure good quality code.

CHAPTER FOUR
SYSTEM ANALYSIS AN DESIGN.

4.1 Introduction.

This chapter shows the summary of data collected from stakeholders and analyses the collected data which lead to the development of system's requirements and the overall design of various system parts. It provides complete description of all functionalities and specification of School Bus Tracking Management System

4.2 Requirement gathering

This stage involves collection of data from our stakeholders i.e., the school Admin, parents and drivers. The techniques used in gathering data were as follows;

4.2.1 Interview

We prepared open-ended questions that was used to interview the school Admin, drivers and parents. From the collected data we were able to generate a list of requirements such as functional and non-functional requirements.

The following are interview questions that we used in gathering of requirements of the system from different stakeholders which are parents, School Admin and school bus driver;

School Admin

- How do you manage bus routes and schedules?
- How do you communicate delays and changes in bus routes to parents and guardians?
- How do you ensure that students are dropped off/picked up at their correct points?
- How do you help new drivers to know the pickup points of the students?
- How do you ensure that drivers pass to the allocated route?
- Generally, what are the challenges faced by the school administration in managing school buses?

Drivers

- How do you handle delays caused by parents?
- How do you know the pickup points of the newly admitted students?
- How do you communicate emergencies to parents and guardians?
- As a driver, what challenges do you face?

Parent

- How do you ensure that your kids were picked up and dropped off at the correct point?
- As a parent or guardian what challenges do you face concerning school buses?

4.3 Core functionalities.

i. Management of users

ii. Global position System (GPS) tracking

iii. Routes Management.

iv. Emergence Managements.

And the activities under each core functionality are as follows;

i. Management of users

- User registration
- Update users' information
- Deletes users' information

ii. Routes Management

- Registering of buses and assign buses to drivers.
- Buses route planning.

iii. Global position System (GPS) tracking

- Get real time information of the bus
- View bus details

iv. Emergence managements.

- Declaration of bus emergence. Example bus breakdown
- Declaration of student's emergence example sickness of students.

4.3.1 Functional requirements.

These relate directly to the functioning of the system and aspects of the system the client is likely to recognize. Functional Requirements for each core functionality are described in the table below

Table 4.1: Functional Requirement

Ref. No.		FUNCTION DESCRIPTION	CATEGORY
F1	**Users Management**		
	F1.1	The system should allow users to be registered in the system.	Evident
	F1.2	The system should be able to send notification about Successful registered or not successful registered users.	Evident
	F1.3	The system should authenticate users and redirect them to their specified profiles.	Hidden
	F1.4	The system should allow users to log out from the system	Evident
F2	**Routes Management**		
	F2.1	System should allow School Admin register buses and assign them to drivers.	Evident
	F2.2	System should allow School Admin to plan bus routes	Evident
F3	**Global position System (GPS) tracking**		
	F3.1	System should allow user to view real time information of the bus	Evident
F4	**Emergence managements.**		
	F4.1	System should allow a driver to declare emergence example car breakdown.	Evident
	F4.2	System should allow parent to declare student's emergence	Hidden

4.3.2 Non-functional requirements

Non-functional requirements are the requirements that specifies how a system performs its operations. These are the constraints by which the system is under, they are the quality or standards that the system should adhere to

summarized in the table below as follows;

Table 4.2: Non-functionalities of the system

Attribute	**Constraints**
Usability	The system must be easy to use and understand to both Administrator and parents.
Scalability	The system must be able to handle an increase number of users,buses and students.
Security requirement	The system must be secure and protect sensitive information such as students' personal details and bus routes.
Compatibility	The system must be compatible with variety of devices include Computers, smart phone and tablets
Maintainability	The system shall be easy to maintain, repair and undergo some improvement since we are localizing the system will be easy to Maintain, upgrade and update its functionality so as to improve its efficiency and effectiveness of its functionalities.
Performance	The system must provide real time update about bus location and arrival time and respond quickly to the user request.
Interoperability	The system must be able to integrate with other system.
Reliability	The system must be dependable and consistently and deliver an accuracy information about bus location and arrival time

4.4 Requirement analysis

Requirements analysis is a set of operations that helps define users' expectations of the application that is being built or modified.

4.4.1 System actors

System actors will interact, either directly or indirectly, with the following significant roles;

i. **System administrator -** Deals with overall management of system
ii. **Parents –** Deals with tracking the bus and sending emergencies of the child.
iii. **Bus driver-** Deals with viewing route and sending notifications once an emergence occurs.

4.4.2 Use case diagram

The use case diagram presented in the figures below uses actors and use cases to represent the functionality of the suggested system. It demonstrates how, after the data is transferred to the database, the users of the proposed system can communicate with it using the system's user interface.

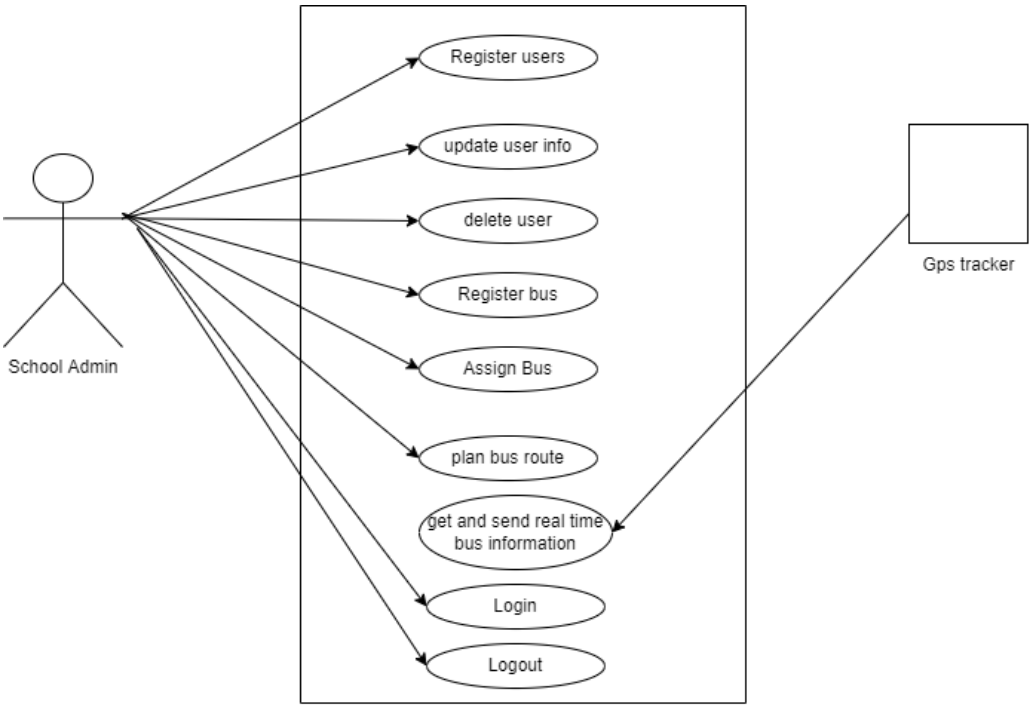

Figure 4.1: Admin and GPS tracker Use Case Diagram.

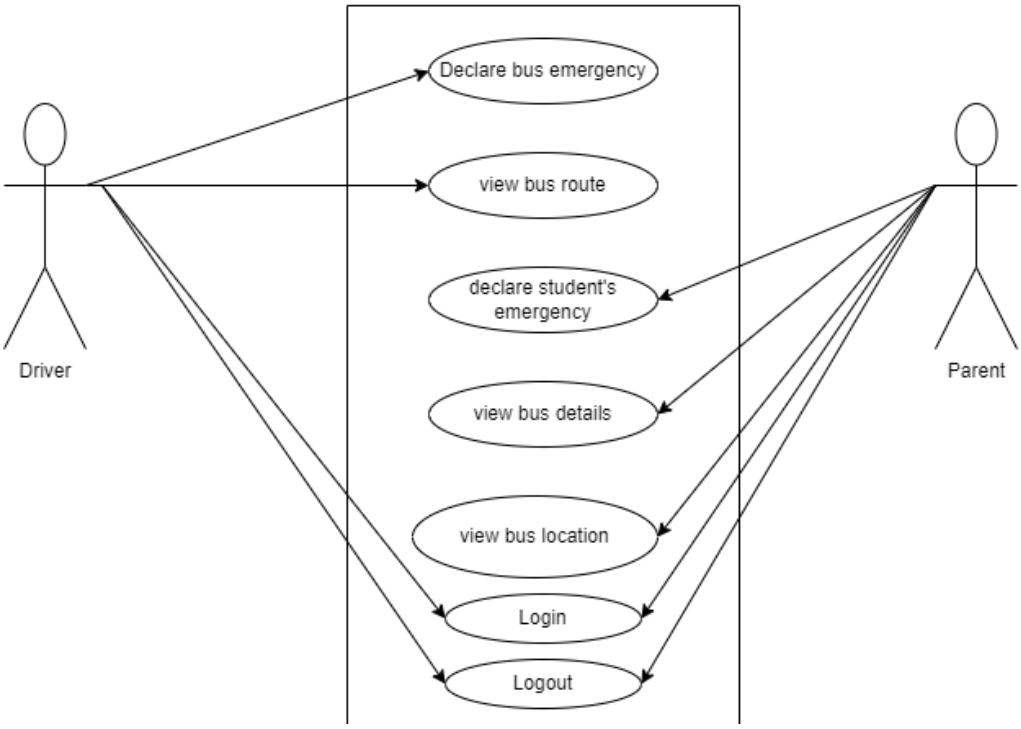

Figure 4.2: Parent and Driver Use Case Diagram.

4.4.3 Use Case Descriptions

Use case description describes the process in use case between the actor and the system and the use case goals. Below are the use case descriptions.

Table 4.3: Register users use case

Field	**Description**
Use case:	Register Users
Actors:	School Administrator
Short description:	The school administrator registers the users and actors of the system.
Pre-condition:	The school administrator must login to the system.
Main flow:	1. The system displays the page for the specific user to be registered. 2. The admin fills in the details of the specific user to be registered. 3. The admin submits a completely filled form of the user to be registered
Exception flow:	If the user's details are incomplete, an error is displayed and the form is not submitted.

Table 4.4: Plan Bus Route use case

Field	**Description**
Use case:	Plan Bus route
Actors:	School Administrator
Short description:	The school admin assigns the buses to the drivers , and their respective routes
Pre-condition:	The buses and routes must be registered
Main flow:	1. The system displays the form with dropdown lists of routes and buses to be assigned. 2. The admin assigns the buses to the respective routes.

	3. The admin submits a complete assignment of the routes to the buses
Exception flow:	If the admin tries to assign a single bus to multiple routes , an error is displayed

Table 4.5: Declare emergency use case

Field	**Description**
Use case:	Declare bus emergency
Actors:	Driver
Short description:	The driver declares an emergency to notify the parents and school administrator
Pre-condition:	The driver must be logged in to the system
Main flow:	1. The driver navigates to the declare emergency section 2. The driver presses the button to send notifications to parents and school admin

Table 4.6: View Bus Details use case

Field	**Description**
Use case:	View Bus Route
Actors:	Parent
Short description:	The parents view the bus details such as the current location
Pre-condition:	The parent must be logged in to the system
Main flow:	1. The system displays the parent view 2.The parent navigates to the view bus details section
Exception-flow:	Failure for bus location to be retrieved and displayed to the system

4.4.4 Class diagram

This particular static structure diagram illustrates a system's classes, their properties, actions (or methods), and relationships between objects in order to define the structure of the system.

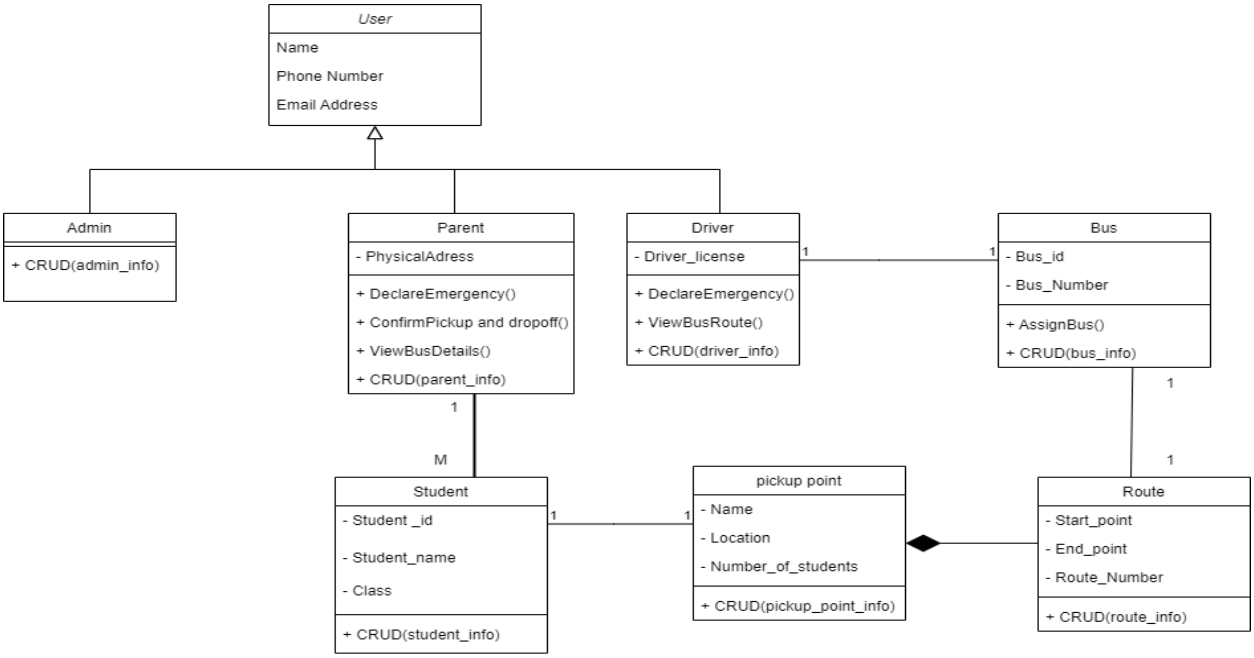

Figure 4.3: Class Diagram for the System.

4.5 System design.

This defines a general system architecture and serves as a meaningful representation of the system that will be created. It entails locating and outlining the underlying software system abstractions and their connections.

4.5.1 Architecture design

This is a representation of the data and software structures needed to construct a computer-based system. It concerns with the process of defining a collection of hardware and software components and their interfaces to establish the framework for the development of a computer system.

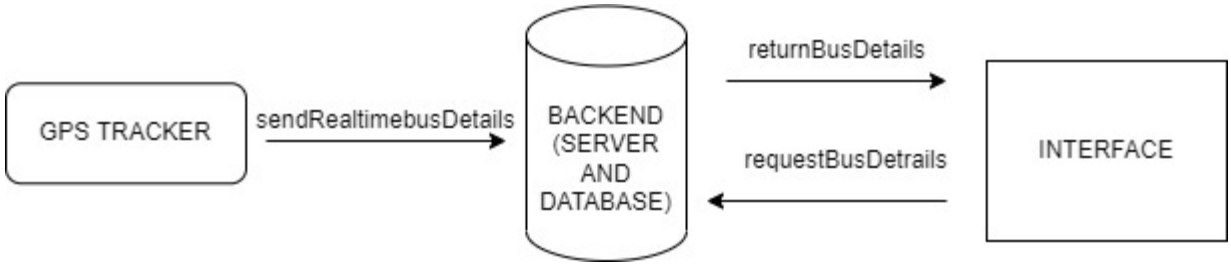

Figure 4.4: Architecture Design of System.

4.5.2 Sequence diagram.

A sequence diagram is used to explain how system items interact throughout time with respect to the order in which they are executed. The sequence diagram provides a clear understanding of the process by demonstrating how the system will behave and respond.

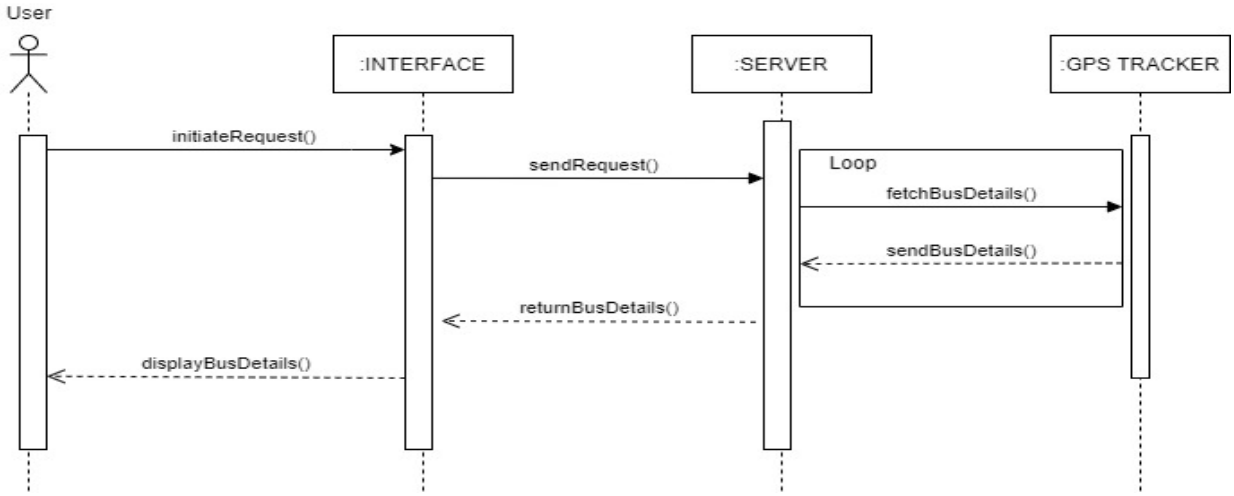

Figure 4.5: Sequence Diagram of System User.

4.5.3 Entity relationship diagram

An entity relational diagram is a highly conceptual diagram that displays the relationships between entities that are stored in the database. An entity can be an object having physical presence or mental existence, and these entities have characteristics that specify their properties. The entity relational diagram is then displayed in the picture below.

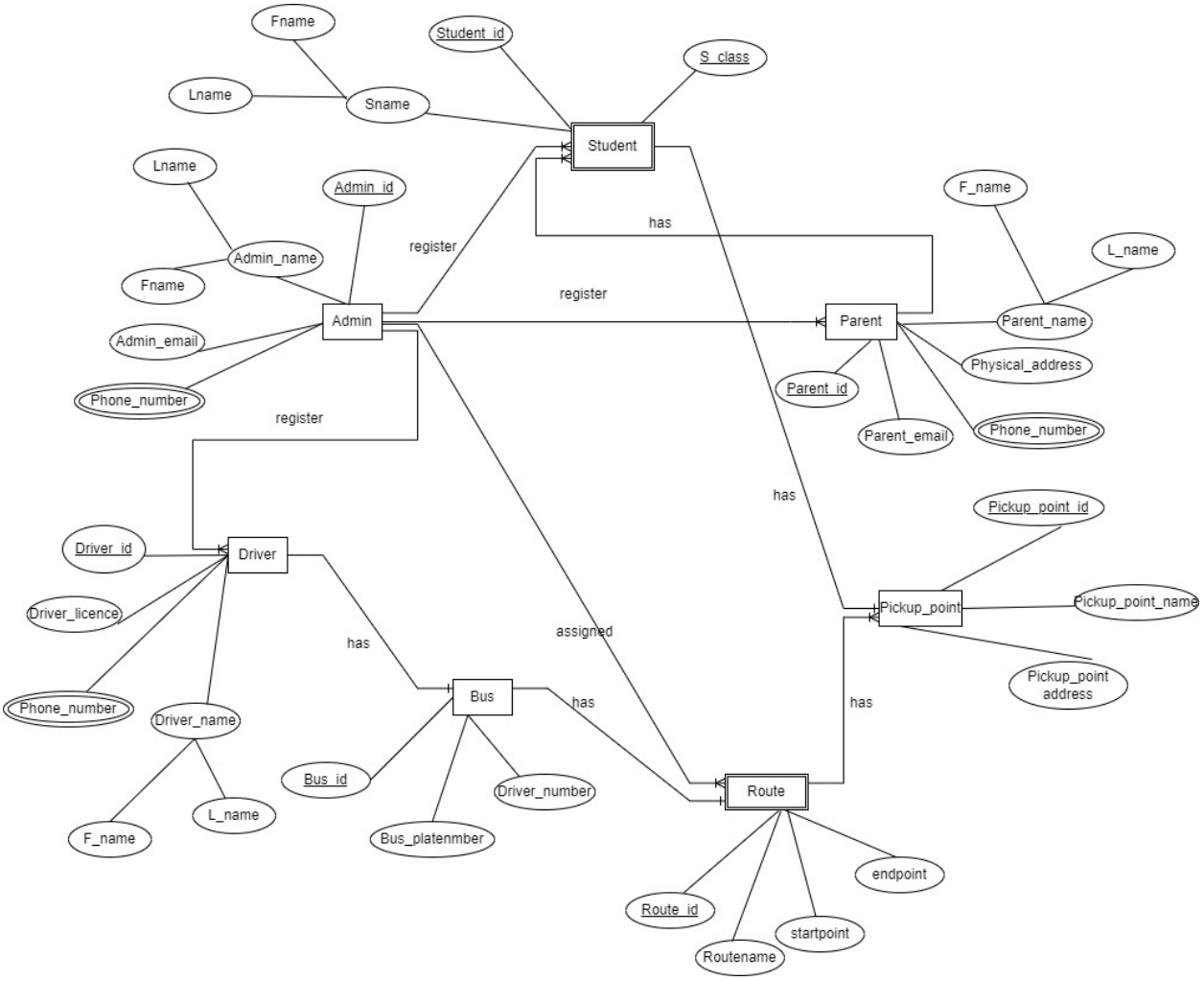

Figure 4.6: Entity Relationship Diagram for the System

CHAPTER FIVE
IMPLEMENTATION

5.1 Introduction

This chapter focuses on the implementation of the project taking into account the objectives, functional and non-functional requirements that were gathered and analysed, as well as the system design, in order to reach the main objective and accomplish the project work. The implementation of the project is divided into three phases that is; Database and backend, Frontend and hardware implementation.

5.2 Database and Backend Implementation

The database and backend for the project is implemented using Firebase cloud services, the firebase services used in the project are authentication, firebase database and real-time database.

The authentication service handles all the authentication requests for the users such as registration, login, logout and password recoveries.

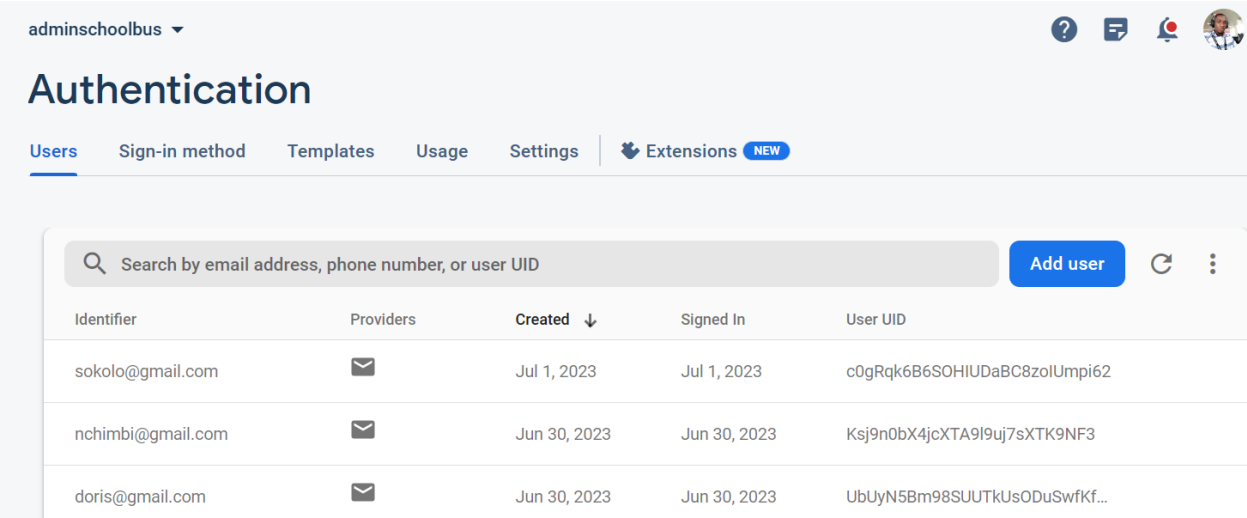

Figure 5.2.1: Registered users managed by authentication service.

The firebase database is a Nosql database that organizes data in form of collections and documents. The firebase Database service handles the storage of the captured information and the retrieval of the stored information using various http requests like post, delete, update and get. The database created was named as adminschoolbus, having three collections named userRecords, bus and route collection. Also there are some sub-collections that exist under the user's document, which are messages and students.

- The userRecords collection stores the information about the users registered into the system such as parents with their children and drivers.
- The bus collection stores the information about the buses that are registered by the school admin, it includes information like the bus number, plate number, Real time location of the bus and route assigned for that bus.
- The route collection stores the information about the routes registered by the admin, it includes information like route name, start point of the route and end point of the route.

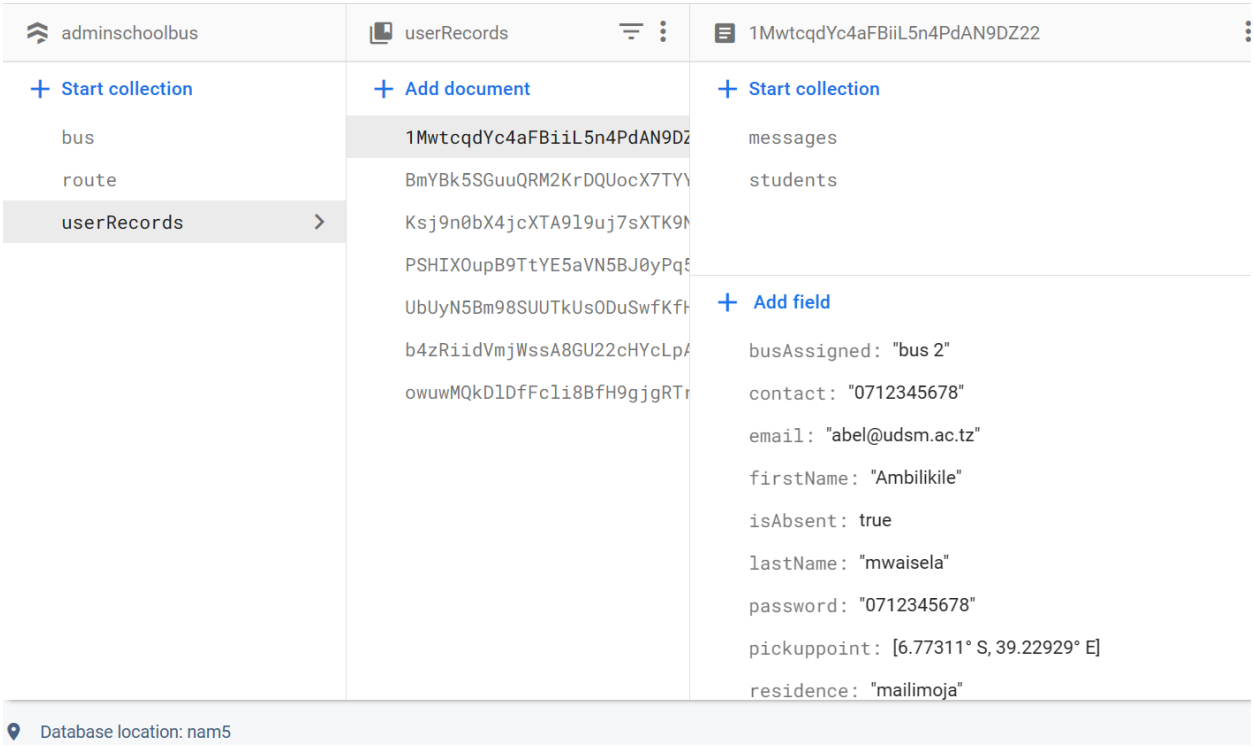

Figure 5.2.2: Admin database with three collections

```
CollectionReference messagesCollectionRef =
    parentDocRef.collection('messages');

// Create a new message document
Map<String, dynamic> newMessage = {
  'messageBody': description,
  'Title': dropdownValue,
  'sender': uid,
  'timestamp': FieldValue.serverTimestamp(),
};

messagesCollectionRef.add(newMessage);
```

Figure 5.2.3: Code snippet to Send Data to messages collection

```
DocumentSnapshot snapshot = await FirebaseFirestore.instance
    .collection('userRecords')
    .doc(uid)
    .get();
firstName = snapshot['firstName'];
lastName = snapshot['lastName'];
busNo = snapshot['busAssigned'];
email=snapshot['email'];
notifyListeners();
}
```

Figure 5.2.4: Code snippet to Get user's data

Lastly, the real-time database service offered by firebase was used to handle the storage and retrieval of the changing bus location geopoints.

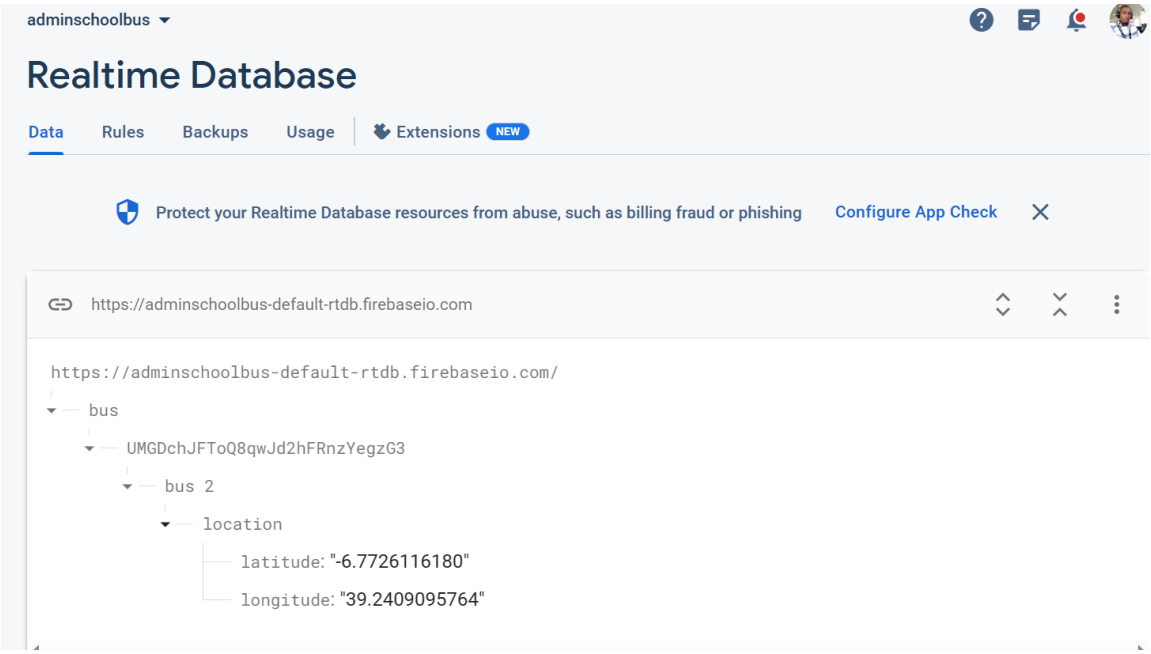

Figure 5.2.5:Real-time database to handle bus location geopoints.

5.3 Frontend Implementation

The frontend of the mobile application is developed using the flutter framework and dart library to ensure the user gets a smooth and good experience while using the application. The frontend implementation based on providing a user with an easier way to perform various functions in the application such as authentication, emergency management, notifications management, view route and track bus.

5.3.1 Authentication

Once the user is registered by the admin, he is able to login to the system using a default password that is his phone number that was used during registration. On successful login, the user gets redirected to his specific page basing on his role.

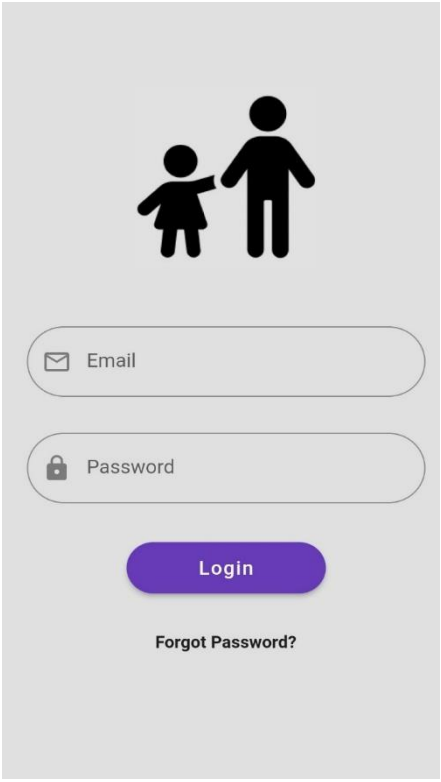

Figure 5.3.1: Authentication page for both parents and drivers

5.3.2 Parent Module

5.3.2.1 Track Bus page

This page allows the parent to view the real-time location of the bus by showing the user's pickup point and the bus movement towards the pickup point on the map. The marker with the green colour indicates the current bus location and the marker with the blue colour indicates the student's pickup point connected with a red line indicating the bus route.

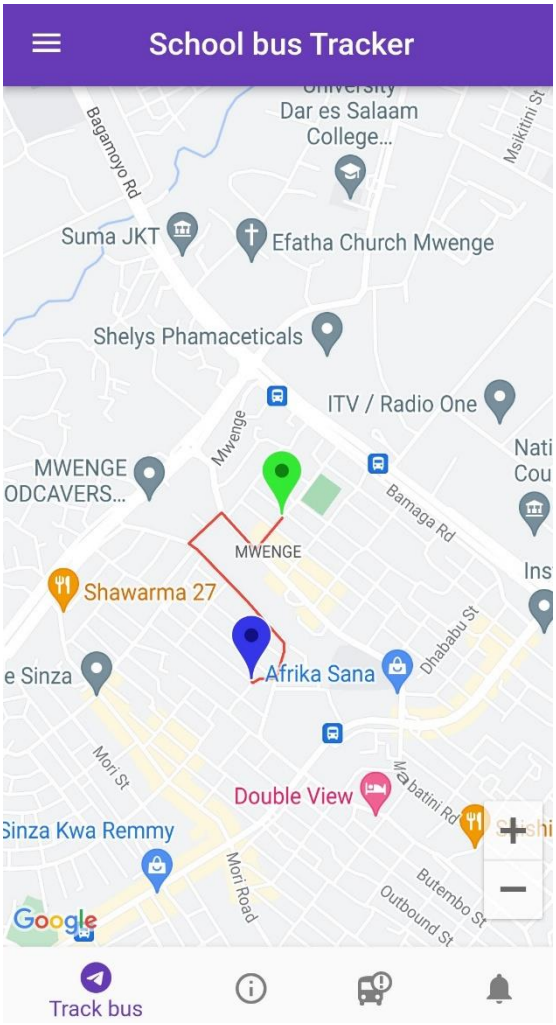

Figure 5.3.2: Parent's page to track bus

5.3.2.2 Bus info page

This page allows the parent to view the details about the bus that is assigned to pick up his child. Some of the information available on this page are Bus number, Plate number, Driver's name and Driver's contacts.

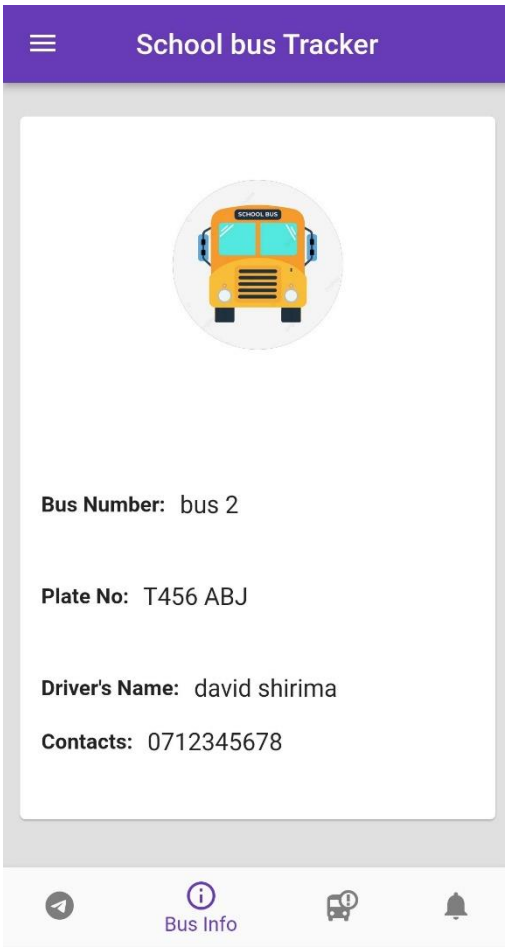

Figure 5.3.3: Bus Info page

5.3.2.3 Emergency page.

This page enables the parent to declare any emergency related to the child's attendance to school that may affect the bus route. The emergencies are labelled as "child won't attend" and "child will delay", once the emergencies are declared they are sent to the driver at real-time and the pickup point gets marked with a red color automatically, indicating that there is an emergency on that pickup point.

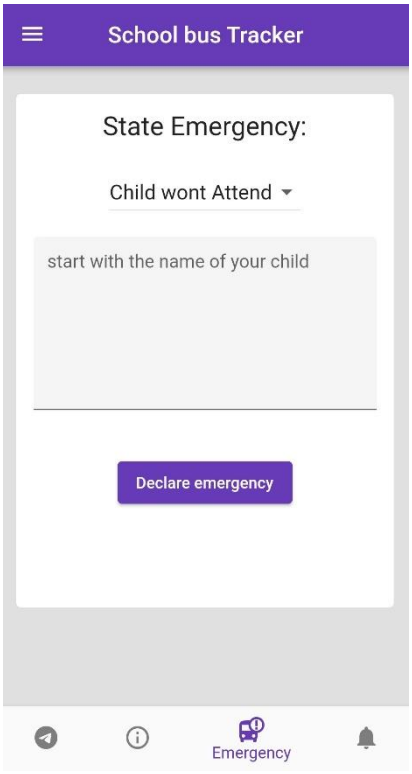

Figure 5.3.4: Parent Declare emergency page

5.3.2.4 Notifications Page

This page enables the parent to view any emergency regarding the bus that is assigned to pick up his child. The notification contains a title, body of the message and the time to indicate when that message was sent.

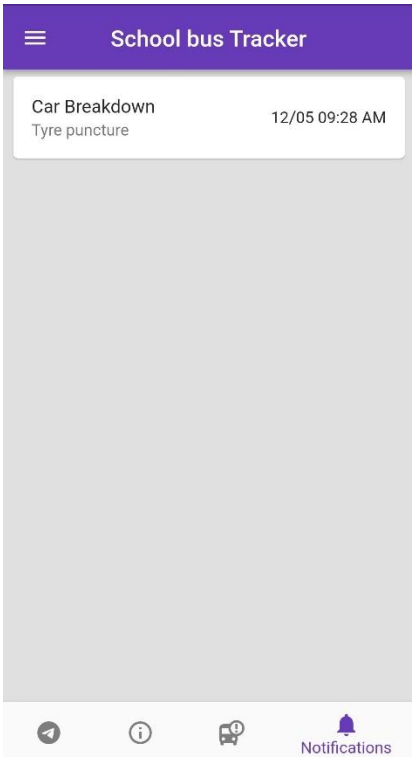

Figure 5.3.5: Parent Notifications page

5.3.2.5 Profile Settings page

This page allows the parent to update some of his information like email, contact and changing the password, also to set up the pickup point of his child.

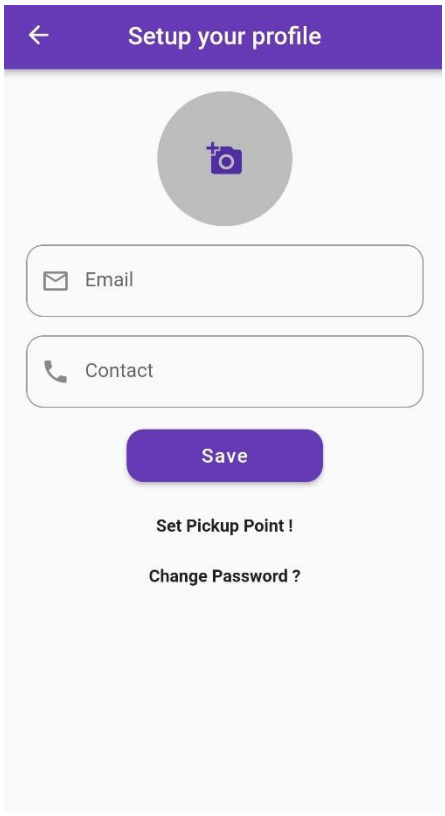

Figure 5.3.6: Parent Profile settings page

5.3.3 Driver Module.

5.3.3.1 View Route page

This page allows the driver to view the pickup points of the students that are assigned to that bus, the pickup points of the students are displayed on the map. The markers with red colour indicate that the child at that pickup point has an emergency, he will not attend school on that day and the markers with blue colour indicate that the child at that point has no emergency.

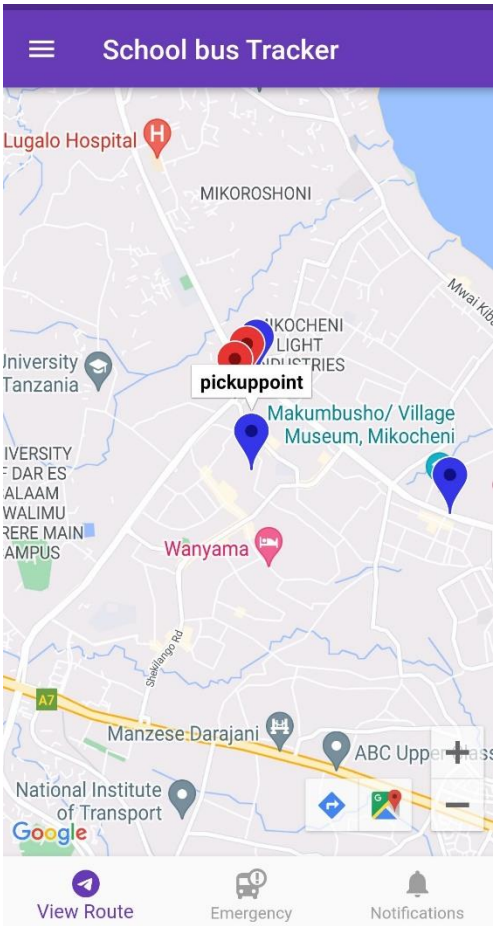

Figure 5.3.7: Driver's View Route page

5.3.3.2 Emergency page

This page allows the driver to send a notification to the parents and school admin regarding any emergency that has occurred during their route. This emergencies are labelled as “breakdown”, ”heavy traffic” or “delay”.

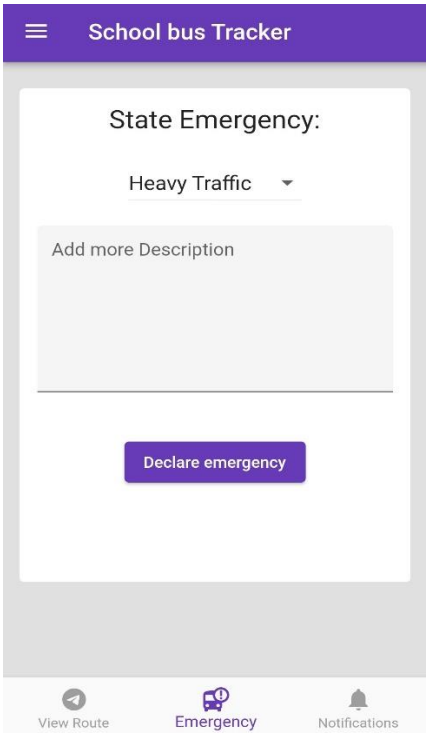

Figure 5.3.8: Drivers Page to declare emergency

5.3.3.3 Notifications page

This page allows the driver to view the notifications from parents regarding emergencies that are likely to cause delays to the bus route.

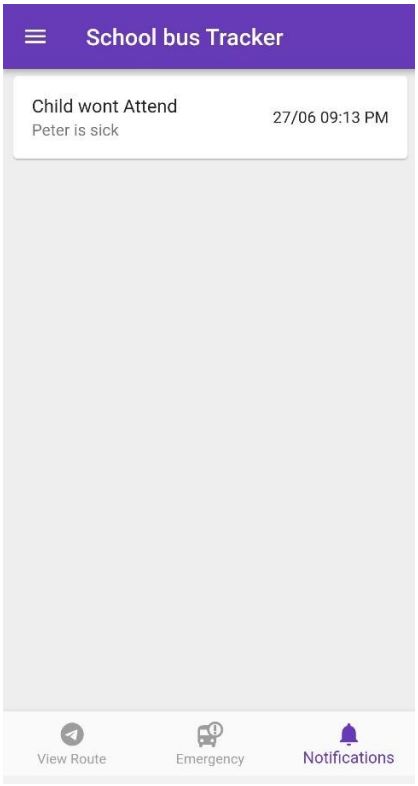

Figure 5.3.9: Driver's page to view Notifications

5.3.3.4 Profile Settings

This page allows the driver to provide additional information and update some of his information like email, contact and changing password.

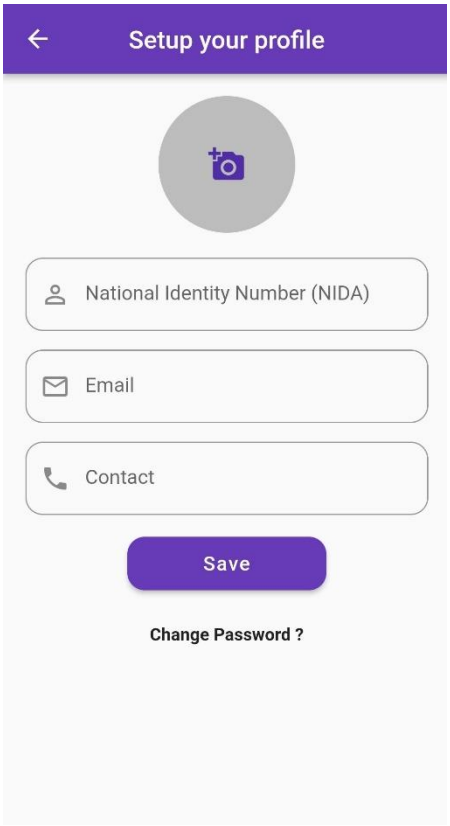

Figure 5.4: Driver's page to setup his profile

5.4 Hardware implementation

This part focuses on the implementation and integration of the hardware components that were used to implement the tracking functionality .These hardware components are;

- GPS NEO 6M,

  The gps is a satellite based navigation system that is used to capture the accurate time and location. In our project it was used as a means to capture the real-time location of the bus.

- NodeMCU ESP32,

  It is a powerful development board that offers enhanced capabilities such as Wi-Fi and Bluetooth connectivity. In our project the NodeMCU ESP32 is connected with the GPS NEO 6M to provide the tracking functionality and this is achieved as follows ; The GPS captures the time and location and the NodeMCU ESP32 receives the captured information , processes it and sends it to our real-time database after every 5 seconds.

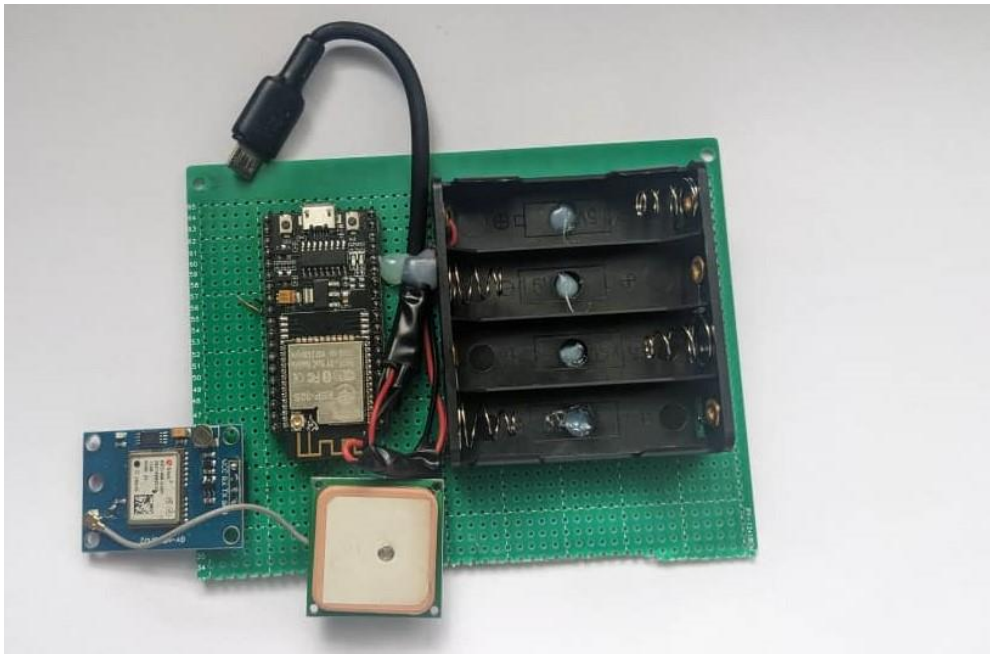

Figure 5.4.1: Gps neo 6M and NodeMCU ESP32.

```
  if (gps.location.isUpdated()) {

    latitude = gps.location.lat();
    longitude =  gps.location.lng();
    gps_t = millis();
    gps_st = true;
    digitalWrite(led, LOW);


void firebase_cloud() {

  if (Firebase.ready() && (millis() - t > 5000)) {

    parentPath = databasePath + "/" + "location";
    json.set(latPath.c_str(), String(latitude, 10));
    json.set(lonPath.c_str(), String(longitude, 10));

    Serial.printf("Set json... %s\n", Firebase.RTDB.setJSON(&fbdo, parentPath.c_str(), &json) ? "ok" : fbd
    t = millis();

  }

}
```

Figure 5.4.2: Code snippet to send bus location to firebase real-time database

CHAPTER SIX
CONCLUSION AND RECOMMENDATION

6.1 Conclusion

The development of the School Bus Tracking Management System has successfully addressed the crucial need for enhancing the safety and convenience of school transportation. By keeping parents directly informed and notified about the movements of their children, the system fosters peace of mind and ensures efficient communication. Additionally, the ability for drivers to privately notify school administrators and parents of emergency events through customized push notifications enables prompt response and effective emergency handling. Also, it acts as a directive to new drivers to familiarize with the pickup points of the students assigned to their bus. Overall, this system serves as a vital tool for schools to ensure the well-being of students and streamline their transportation operations.

6.2 Recommendation

Based on the implementation and evaluation of the School Bus Tracking Management System, several recommendations can be made for further improvement. Firstly, integrating real-time GPS tracking and route optimization algorithms can enhance the accuracy and efficiency of the system, allowing for better bus routing and minimizing delays. Additionally, incorporating features such as automated attendance management and student check-in/check-out systems would further streamline administrative tasks. Moreover, exploring the integration of smart devices and Internet of Things (IoT) technologies could provide additional functionalities, such as monitoring environmental conditions inside the bus or detecting unusual events. Lastly, it is essential to prioritize data security and privacy measures to ensure the protection of sensitive information. By considering these recommendations, the School Bus Tracking Management System can continue to evolve and meet the evolving needs of schools and parents alike.

REFERENCES

Adjunct, Carnegie Mellon University. (n.d.). *UML, Patterns, and Java, 3rd Edition.* Technical University of Munich. .

Allan Dennis , Barbara Haley WIxom & David Targarden. (2015). *System Analysis and Design: An Object-Oriented Approach with UML.* Wiley.

Arlow, J. a. (2005). *). Practical Object-Oriented Analysis and Design* . Addison-Wesley Professional.

Blanchard, B. S., and W. J. Fabrycky. (2010). *Systems Engineering and Analysis (5th Ed.).*

Sommerville, I. (2011). *Software Engineering.* New York San Francisco: Addison Wesley.

`

APPENDICES

APPENDIX A

Semester one project scheduling

ACTIVITY WEEK	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Performing project research															
Literature review															
Studying different methodologies															
Consultation with supervisor															
Preparing mid semester report															
Presentation of progress															
Project amendments															
Requirements gathering and analysis															
Creating design diagrams															
Consultation with supervisor															
Preparation and submission of report															
First semester presentation															

Figure 6.1: Semester one project scheduling

Semester two project scheduling

Activity week	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Database Redesign															
Database Implementation															
Web UI Implementation															
Parent module UI Implementation															
Driver module UI Implementation															
Backend Implementation															
Consultation with Supervisor															
Presentation of Progress															
Project Amendments															
Hardware Implementation															
Hardware Integration															
Consultation with Supervisor															
Preparation and Submission of report															
Second Semester Presentation															

Figure 6.2: Semester two project scheduling

APPENDIX B

Project budget.

Table 5.1: Project Budget

s/n	**Expenses**	**Costs (Tsh)**
1	GPS MODULE	50,000
2	MICROCONTROLLER	30,000
3	BATTERY	8000
4	JUMPER CABLES	2000
6	INTERNET	80,000
	TOTAL	170,000