ASHESI

ASHESI UNIVERSITY COLLEGE

BILLING AND LOCATION TRACKING APP FOR TRANSPORTATION

CAPSTONE PROJECT

B.Sc. Computer Engineering

Stephen Kwadwo Yiadom Owusu

2021

ASHESI UNIVERSITY COLLEGE

BILLING AND LOCATION TRACKING APP FOR TRANSPORTATION

CAPSTONE PROJECT

Capstone Project submitted to the Department of Engineering, Ashesi University College in partial fulfillment of the requirements for the award of Bachelor of Science degree in Computer Engineering.

Stephen Owusu

2021

DECLARATION

I hereby declare that this capstone is the result of my own original work and that no part of it has been presented for another degree in this university or elsewhere.

Candidate's Signature:

Candidate's Name: Stephen Kwadwo Yiadom Owusu

Date: Tuesday, April 27, 2021

I hereby declare that preparation and presentation of this capstone were supervised in accordance with the guidelines on supervision of capstone laid down by Ashesi University College.

Supervisor's Signature:

…………………………………………………………………………………………

Supervisor's Name:

…………………………………………………………………………………………

Date:

…………………………………………………………………………………………

Acknowledgments

I wish to thank my supervisor Dr. Nathan Amanquah, whose continuous supervision and encouragement helped me complete this project. I also appreciate my mentor Theodore Sagoe for helping me think deeper in security and risk management.

Abstract

Most companies in Ghana provide their staff with buses to make transportation more accessible, but this transportation medium is not tracked, making its operations difficult for staff. Means of payment become a problem when the staff has a higher denomination.

This project aims to solve these problems by adapting the tracking technology based on GPS and the cashless payment system based on RFID. Using these technologies, this project will make tracking and payment on these company buses easy. This project goes beyond tracking to estimating the arrival time of the bus and beyond RFID payment to tracking user pickup and endpoints. This system is flexible and allows users to make payments with cash and make payments by scanning a QR code.

Contents

Chapter 1 - **Introduction**

1.1 Introduction

Most big institutions in Ghana have buses for their staff, and this is to make movement from home to work easy for those without cars of their own. Being a company bus, it has a fixed route with an expected time of departure and arrival, and though this system works, it is sometimes, if not often, inconvenient for passengers. That is, the time wasted waiting for the bus's arrival, time wasted waiting for the bus when it is already gone, the bus arrives but it is already full, and making the payment with a high denomination.

1.2 Problem Definition

The two main problems identified in this area are bus tracking and payment of trips. Company buses are usually not tracked, making it difficult for the staff members to know the exact position of the bus at any given time.

The other problem is the payment system which is via cash, and this system though it works, comes with issues like payment with a high denomination. This is a problem for both the driver and the passenger, especially when the driver does not have change for the staff member. Also, staff members find themselves in a tight spot on days they do not have cash for transportation.

1.3 Proposed Solution

These problems can be solved with the two proposed systems, GPS-based tracking system, and RFID-based payment system. The GPS-based tracking system helps passengers know the exact position of the bus and the estimated time of arrival. The RFID-based payment system promotes a cashless payment system that will eliminate issues with a high denomination.

In this project, a prototype system and a mobile application will be built. The mobile application will serve as a platform for the users to interact with the system on the field. It will also update the user on information like available seats and the estimated time of arrival.

Chapter 2 - **Related Work**

The objective of this paper is to design a GPS tracking and cashless payment system for transportation. The objectives of this project can be achieved by understanding the concepts in Vehicle tracking, the concepts in cashless payment systems, and the technologies needed to achieve this objective. This chapter reviews similar work that has been done, the concepts used, and the considerations for designing the proposed solution.

2.1 **Vehicle Tracking Overview**

This section will be looking at the different techniques used for vehicle tracking, GPS tracking, and the level of accuracy. Techniques like image processing-based tracking, signpost technology, RFID-based tracking, and GPS-based tracking are used in vehicle tracking, but GPS tracking is the most used. This is because it is affordable and easy to use. Most transportation companies have adopted GPS-based tracking like uber, taxify which uses the GPS component in mobile applications [1].

GPS tracking is a technology that can be used in different areas in this modern world. This technology can be used for child tracking, asset tracking, and vehicle tracking, including buses, cars, and shipping. In 2014, Hind Abdalsalam Abdallah Dafallah researched real-time tracking and its accuracy using GPS technology [2]. In areas that this technology is applied, accuracy plays a huge role. An example is the research paper by Fleischer, Nelson, and Sowah on vehicle tracking and alert system against accident and armed robbery cases [3]. This severity and sensitivity of the project make the accuracy in GPS tracking very important. In this research, the GPS receiver receives the location coordinates from the satellite, a microcontroller process the data received, and a GSM modem sends out the coordinates as SMS. The test concluded that the accuracy of the

GPS coordinates from the GPS receiver compared to the coordinates from Google maps ranges from 1 to 2.5 meters, which tallies with the GPS datasheet [3]. The only problem was the real-time tracking because of the delay in the GSM module connecting to the network before transmitting the data. All vehicle tracking-related work came across used GPS technology. The GPS module needs to have the differential GPS mechanism not to have problems tracking the vehicle’s position when in a tunnel because the GPS rely on a clear line of sight with the satellite to properly function [4][5].

2.2 **Cashless Payment Overview**

The most used payment system in Ghana is the cash-based system. There are a few digital payment methods in the country, but these methods are not used on a large public scale. This is mainly because most Ghanaians do not trust the system except the mobile money system, which has gradually earned the people’s trust and is being used in most transactions in the country [6]. Cashless payment has been implemented in different ways like the contactless payment system like Apple pay, Google Pay, the NFC payment method, Atm cards – debit or credit, biometric authentication. The main goal for going digital is to make payment easy and also trackable.

The RFID technology is a type of contactless payment system which relies on radio waves to make payment. This technology is not complicated to equip; hence would be easy to set up and use. [5] shows the improvement of the payment system in London buses precisely the one operated by the Transport for London from a bus transit card to the Oyster system [7]. The Oyster system is a new payment platform adopted that allows passengers to make payments with a contactless card. This platform links the card to the user’s account [7]. This type of payment system saves the user the time spent queuing for a ticket or the time spent to top up their bus transit card.

2.3 **Relevant Ideas for the proposed solution**

From previous sections in this chapter, it is clear that much work has been done in vehicle tracking and the cashless payment system. The focus of vehicle tracking in previous work was on accuracy and real-time tracking. Also, the focus of the cashless payment system was to create a reliable and easier way of making payments from one account for their public transportation. However, this paper aims to design and build a system that allows tracking an organization bus that uses a cashless payment system.

The use of GPS technology and RFID technology will be used in building the proposed solution. In the related literature used, a web-based application was used in monitoring the location of the bus. However, for this project, a mobile application will be used to make accessibility easy.

Chapter 3 - **Requirement and Architecture**

3.1 **Project Purpose**

The system to be built is aimed at making transportation and payment easy for company buses. The purpose of the software requirements specification is to layout the system's functions, look at the targeted user, plan how to implement these functions, make the system user-friendly, and any other preparation measure to make the system complete and useable.

3.2 **Project Scope**

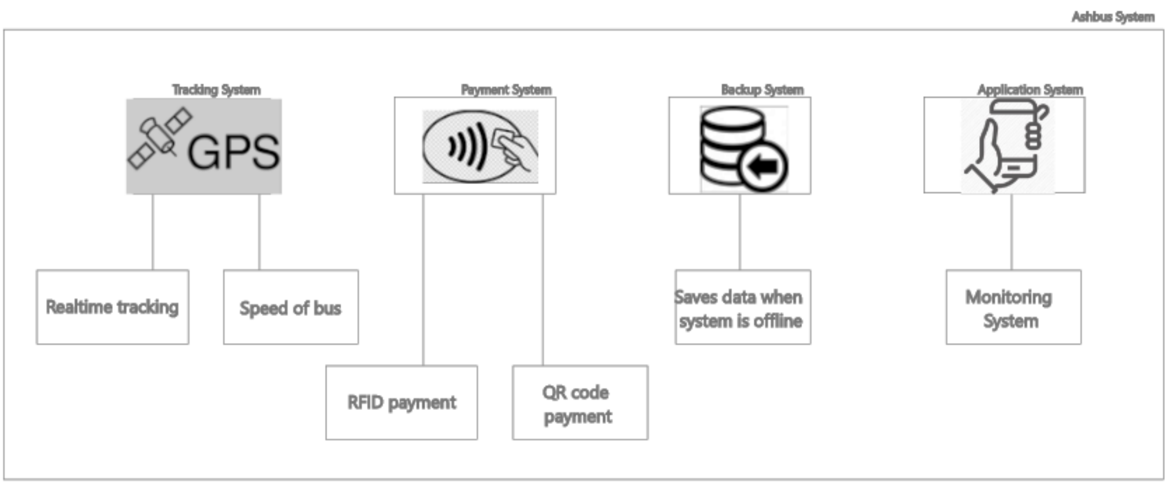


Figure 3.1 – System Modules

This system from Figure 3.1 comprises four main modules, the tracking system, the payment system, the backup system, and the application system. The Tracking system and the payment system are integrated using a microcontroller. Using an API, data from both tracking and payment system is transmitted to the database and accessed by the Application system.

3.2.1 Tracking system

This system is for tracking the bus's current position, which is achieved using the GPS module attached to the bus. This module will be used to get coordinates, and the data collected will be transmitted to the cloud, which is then accessed by the mobile platform.

3.2.2 Payment System

Users of the system are supposed to pay for every trip made. This module handles all payment transactions and makes payment possible. The RFID module based on radio waves will be used to implement the cashless payment system. The user ID and company ID of the user will be stored on the card, and upon scan, that information is read and processed to make payment.

3.2.3 Backup System

This module is triggered when the system is in offline mode. This ensures that there is no transaction loss or a halt to the system's operation when it is offline. The SD card module is used in implementing this module where all transaction data is saved on the sd card.

3.2.4 Mobile application

This module makes interacting with the system possible. For instance, user can take their mobile phone and check the exact position of the bus. The tracking system got the bus coordinate and the mobile application retrieved and displayed to the user in the background.

3.3 **Overview**

This section of the document is descriptive information about the project, use cases, design requirements, and implementation.

3.4 **Product Function**

The objective of the solution is to make transportation and payment easy. The company's staff can check the bus's current location and estimated arrival time using GPS tracking. The system will use the RFID scans of the users to log the start and endpoints of the staff members. The RFID scans will also be used for payment.

3.5 **User Classes and Characteristics**

The product will be used frequently by the workers of a registered company with their buses. The system will only work for the employees of companies that are signed up on the system. It will be a flexible and friendly system that would not require an advanced educational background to use.

3.6 **Operating Environment**

The mobile application will be developed for both Android and IOS devices. This will remove any form of device restriction; hence anyone can access the system.

3.7 **Design and Implementation Constraints**

One of the main constraints is that not all Android devices support NFC; hence the NFC payment option will be optional. Also, since AshBus is for company buses, each company will be responsible for managing the individuals signed under the company's name. This is to prevent individuals outside the company from registering under the company.

3.8 **User Interface**

- A constraint layout will be used for a standard UI across all platforms.
- Map interface for bus tracking.
- A standard Error Message UI.
- A standard Permission request UI.

- Fixed dimensions on every screen.
- User friendly.
- Workable font size.

3.9 **Hardware interfaces**

Hardware interfaces needed to complete the system,

- The phones should support NFC (optional).
- RFID reader and card
- GPS module.
- Microcontroller Unit
- GSM module
- SD Card module
- Display module

3.10 **Software interface**

The software interface is designed to be multiplatform, meaning it can operate on Android and IOS devices. Other implementations in this interface,

- Database implementation.
- API Implementation.
- Google Map integration.
- Built-in QR code scanner for payment purposes.
- Authorization and permission level checks.

3.11 **Use Cases**

The system features are captured using a use case diagram based on the interaction between the staff and the application. There are three actors in this system, the staff with admin privileges, the driver, and the other staff as passengers. This is also shown in Figure 3.2. Each actor has permission to certain features based on their position. For instance, the staff with admin privileges can top up the accounts of other staff members while the other actors cannot.

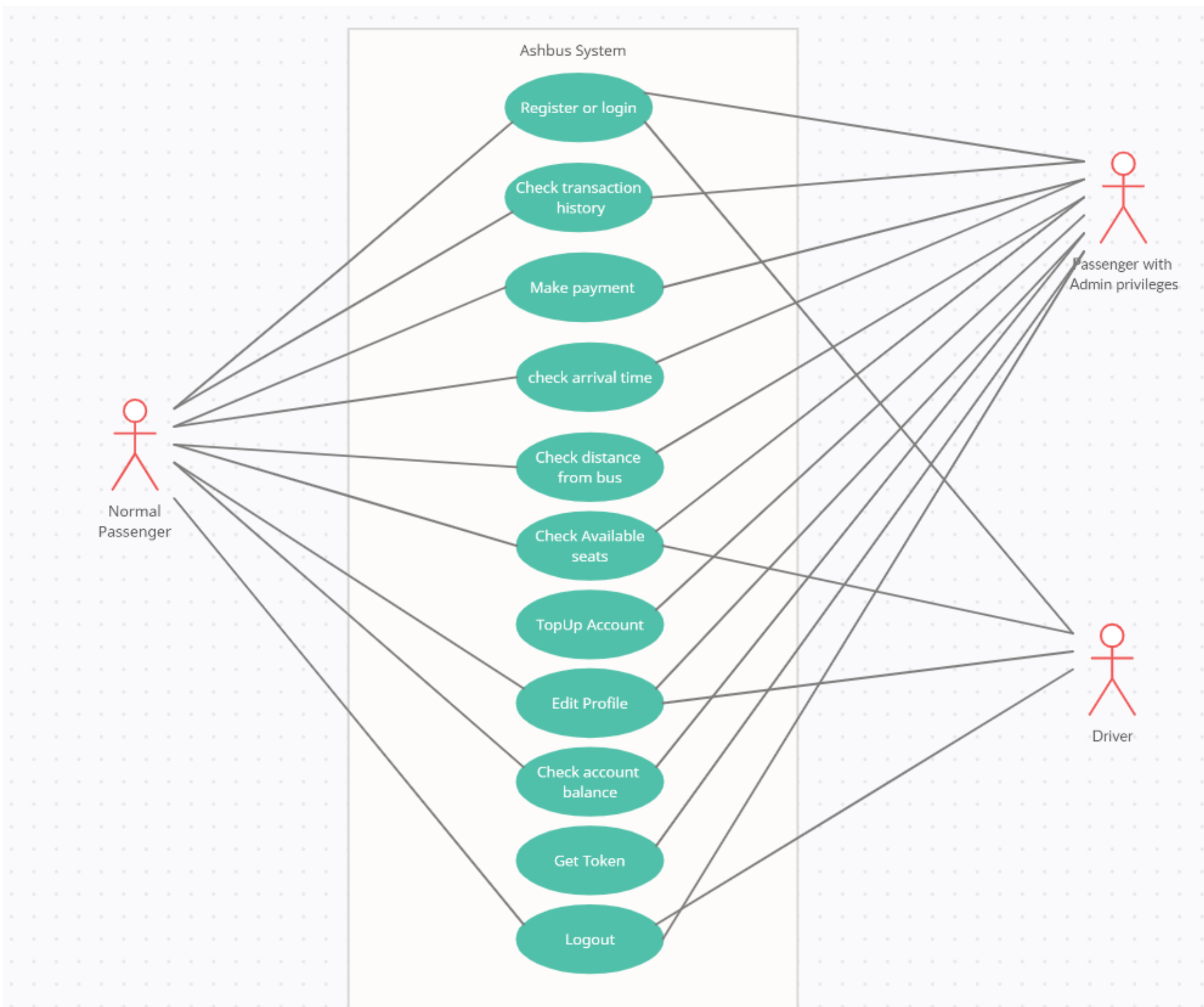


Figure 3.2 – Use case diagram

3.12 Requirement Specification

3.12.1 User Requirements

3.12.1.1Functional Requirement

The user should be able to,

- **REQ-1:** Create an account.
- **REQ-2:** Check transaction history.
- **REQ-3:** Make payments by scanning the QR code.
- **REQ-4:** Make payments by scanning RFID cards.
- **REQ-5:** The user has the option to pay with cash.
- **REQ-6:** See the estimated time for bus arrival.
- **REQ-7:** Make payment even when the system is offline.
- **REQ-8:** See the available seats on the bus.
- **REQ-9**: See bus location.

The driver should be able to,

- **REQ-10:** Check transaction history.

The system administrator should be able to,

- **REQ-11:** Approve registration request from workers.
- **REQ-12:** Update transportation charges.
- **REQ-13**: Can assign authorization level based on position.
- **REQ**-14: Can top up a user's account.

3.12.1.2Non-functional Requirement

- Performance Requirements
  - The mobile application should be stable.
  - The mobile application should be responsive.
  - The RFID reader should be responsive.
  - The system should be able to operate even when offline.
  - The system should not have any form of glitch.
- Safety and Security Requirements
  - The system should sanitize data before sending it to the database.
  - There should be a security policy.
  - The system should not allow default passwords like "admin" and "password."
  - Account should be suspended after three failed attempts.
- Software Quality Attributes
  - The mobile application should be reliable.
  - The mobile application should be robust.
  - The mobile application should be maintainable.
  - The system should be flexible and adaptive.
  - The system should be reusable.
  - The system should be friendly.

3.12.2 System Requirements

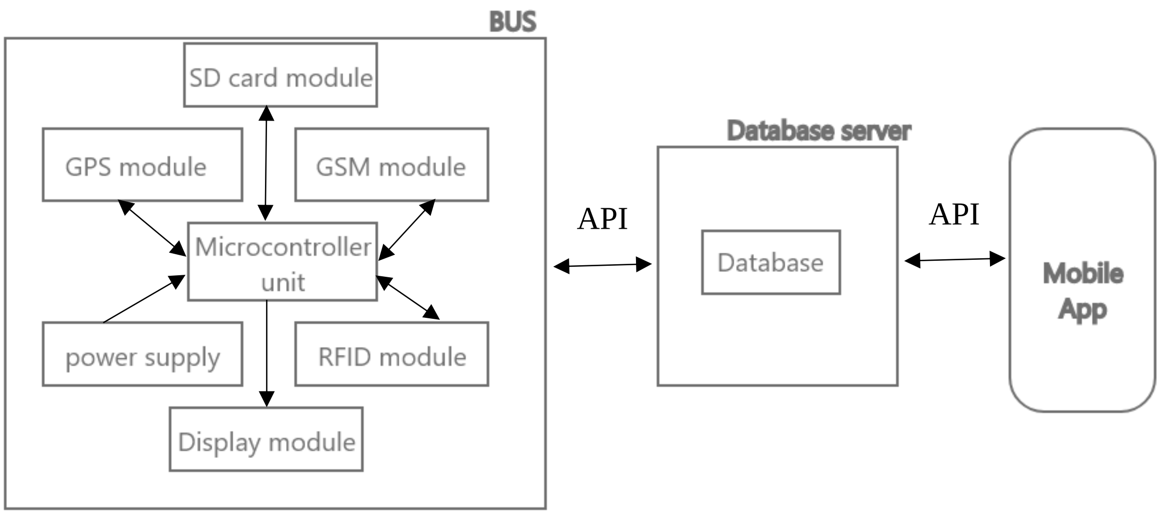
- The system will have a user from each institution with admin privileges.
- The system will have fixed charges irrespective of destination.
- The system should work both offline and online.

- The system supports multiple payments by a user.
- The system will provide uses with the location of the bus.

3.13 **Security Policy Statement**

- Confidentiality of information is assured.
- Protection of user information against any unauthorized access.
- The integrity of user information will be maintained.
- User information will be made available whenever needed.
- Provide employees with some level of security training.

3.14 **System Architecture**



The main hardware components of the system are the GPS module, RFID module, GSM module, SD card module, display module, and the microcontroller located on the bus. There is the database hosted on an online server. There are also the APIs and the mobile application.

The GPS module is for reading the bus's current location through coordinates, the RFID module for payment transactions, the GSM module for transmitting data from the field to the online database server, and the storage module for system backup to manage offline mode. The microcontroller is the processing unit of the system. AshBus will be designed to operate online and offline to make payment possible under both conditions. This system will be managed and accessed using the mobile application. The user will check bus location, time for the bus to reach them, make payments and even check the available seats.

3.15 **Component Specification**

- Database: The system uses a MySQL database because it is flexible, easy to use, and quick integration with applications.
- API: The API was implemented using Python Flask RESTFUL API. All APIs built with this platform handle all forms of communication with the database. There is also the Google Maps API which handles map-related features.
- Android Studio: This is the platform for building the prototype version of the mobile application. Flutter will be used to build the final product, and this is because it is cross-platform and has the option to build a web application with the same code.
- GPS Module: A module with low power consumption and a strong satellite search capability, and this module handles receiving coordinates from the satellites.
- GSM Module: A GSM module with a GPRS feature so data can be transmitted via the internet.
- Microcontroller: Has an I2C, SPI, and UART communication interface. Has about 23 or more programmable I/O pins and low power consumption. This is so that multiple components with different communication interfaces can be connected.
- Display Unit: This module should have the ability to display Text.
- RFID Module: This module will handle payment and passenger tracking.

Chapter 4 - **Design Process and Implementation**

4.1 **Overview**

In this section of the document, detailed information on how the tracking and billing system was implemented. Under Figure 4.1 are the hardware design and component selection, and embedded programming, database implementation, API implementation, and mobile application development.

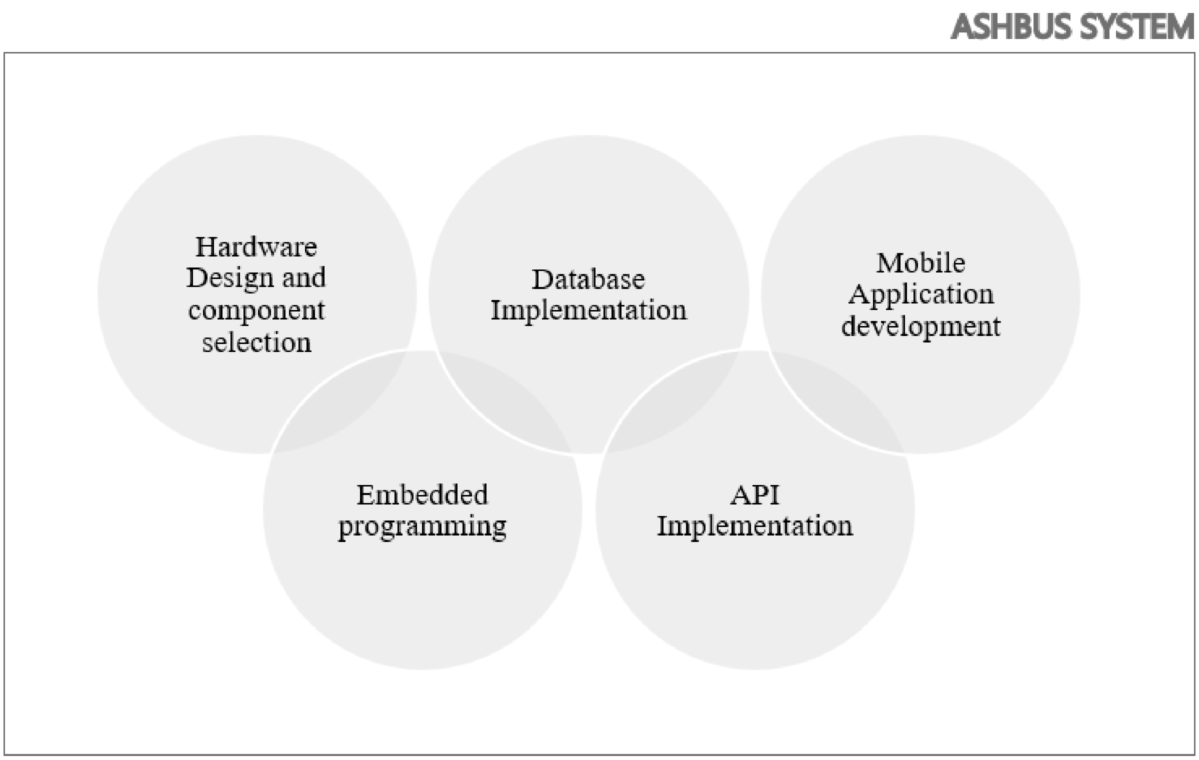


Figure 4.1 – Main implementations of the system

4.2 **Hardware Design and Component Selection**

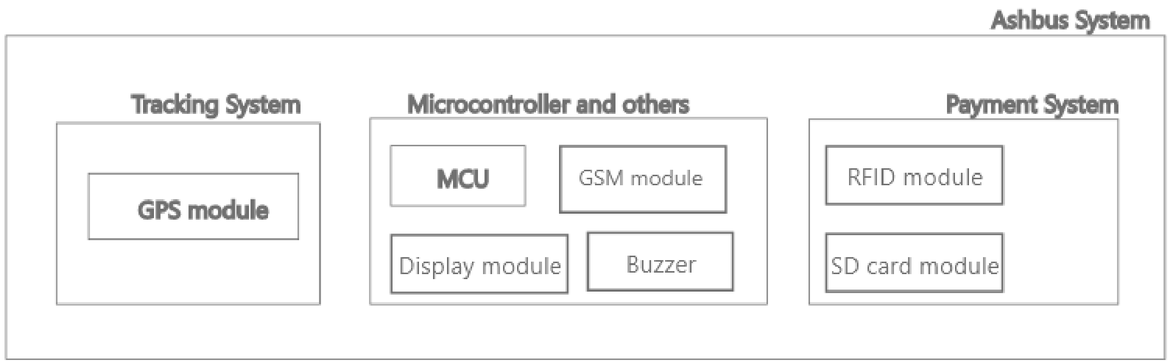


Figure 4.2 – component selection category

The Ashbus system is built to handle transportation tracking and cashless payment of trips for company buses. Figure 4.2 shows the category considered in the selection of hardware components. There is the tracking system, the payment system, and the category that handles data processing, data transmission, and others. The GPS-based tracking system used the GPS module connected to the system to collect the bus coordinates. The RFID MFRC522 module was used to read user data from the card for the payment. The RFID will also be used to track by taking note of the user's onboard and offboard status. Upon their first scan, the system marks them as onboard. The number of seats available on the bus reduces by one when this condition is met. Therefore, the number of onboard statuses recorded on the bus will determine the number of available seats. Under the payment category, there is the SD card module to help make offline payment possible. This satisfies the system requirement that "the system should operate both online and offline." A local storage is needed to record all transactions made when the system is offline. There is also the display module to show the user what is going on in the system. That is, if the payment made was successful or if it failed. A buzzer gives off a particular tone when payment failed and another when payment is successful.

There are sensors and actuators, and these devices do nothing on their own unless they are connected to a microcontroller; the Atmega328P-Au is that component. This component communicates with all the devices connected to it and uses the data collected by the sensors to trigger a particular action. This is done by programming the atmega328P-au and telling it what to do with the data collected. Its primary purpose is to integrate the different components as one to get the expected results. For instance, when the RFID card is scanned, the RFID module communicates with the atmega328p updating it on changes that have taken place like “isCardpresent” and sending the data read from the card to the atmega328p. The atmega328p checks and analyzes the data, then it triggers the buzzer and displays the result on the display module letting the user know if their payment was successful or not. The difference between the atmega328p-AU and the atmega328p-PU is that the atmega328p-AU is 32-pins and surface-mount friendly, and the atmega328p-PU is 28-pins and breadboard friendly. The atmega328p-PU was mounted on a breadboard for prototype purposes, but the final product will have the atmega328p-AU mounted on a PCB.

The GSM module will handle data transmission using the GPRS feature to transmit data over the internet using the Rest-based communication API built for the system.

4.2.1 Power Requirement

The device would not work without power, and not just any kind of power can be connected to the system because of the 100% chance of destroying one or more of the modules in the tracking and payment device. Some of the components take in 5V, 3.3V, and others take from 3.4V to 4.4V. There must be a voltage regulator to make the input to the different devices suitable, reducing the chances of damage. There are different voltage regulators, and each has its output current, and since the GSM module can draw up to 2A of current, the voltage regulator must give an output of at least 3A of current hence LM350. A heat sink was added to the voltage regulator to help regulate the device's temperature since the LM350 generates much heat.

The LM350 also supports adjustable voltage output, and this is achieved by changing the resistance of the two main resistors connected. Figure 4.3 shows the main resistors (R2 and R3) that affect the voltage output.

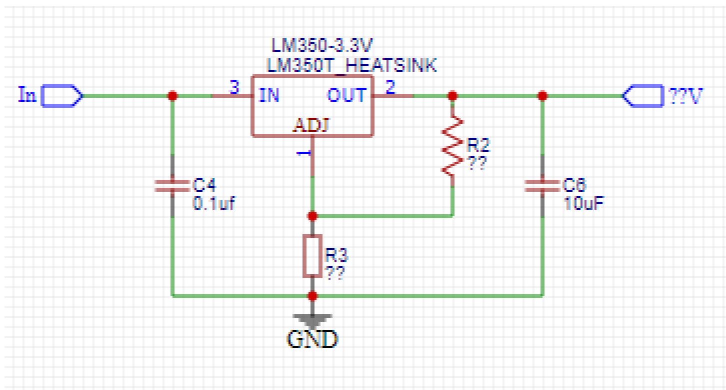


Figure 4.3 – a circuit of voltage regulator

The output voltage is calculated using,

$$Vout = VREF * \left(1 + \frac{R3}{R2}\right),$$

Where VREF is 1.25.

Output Voltage	R2	R3
5V	3.3KΩ	10KΩ
4.4V	270Ω	680Ω
3.3V	2kΩ	3.3kΩ

Table 4.1 – Required resistors for specific output voltage.

The device will be connected to the bus's power outlet, which would provide the system with an input voltage of 12V. In Table 4.1, the exact resistance required to get a separate voltage output is shown.

After component selection, these components were selected, as shown in Table 4.2.

Type	Component	Design requirement	Function
Microcontroller Unit	Atmega328p-AU	• High-performance Microchip • 32KB (SP flash memory with read-write capabilities • Three flexible timer/counters with compare modes, internal and external interrupts. • SPI serial port • 32 general-purpose I/0 pins	Acts as the brain of the system
GSM Module	SIM800l	• Can transmit data over the internet using GPRS	Data transmission through the internet
RFID Module	RFID-RC522	• SPI communication • Low power consumption	
GPS Module	Ublox Neo 6m	UART communication	For GPS tracking
Voltage Regulator	LM350 with heatsink	• Output current minimum of 2A • Output adjustable voltage	Voltage regulation
Power Adapter	12V adapter	12V Adapter for the car's power outlet	Tap power from a vehicle

Table 4.2 – selected components

A schematic design was made for the selected components. This helped to show how the connection of each of the components to each other.

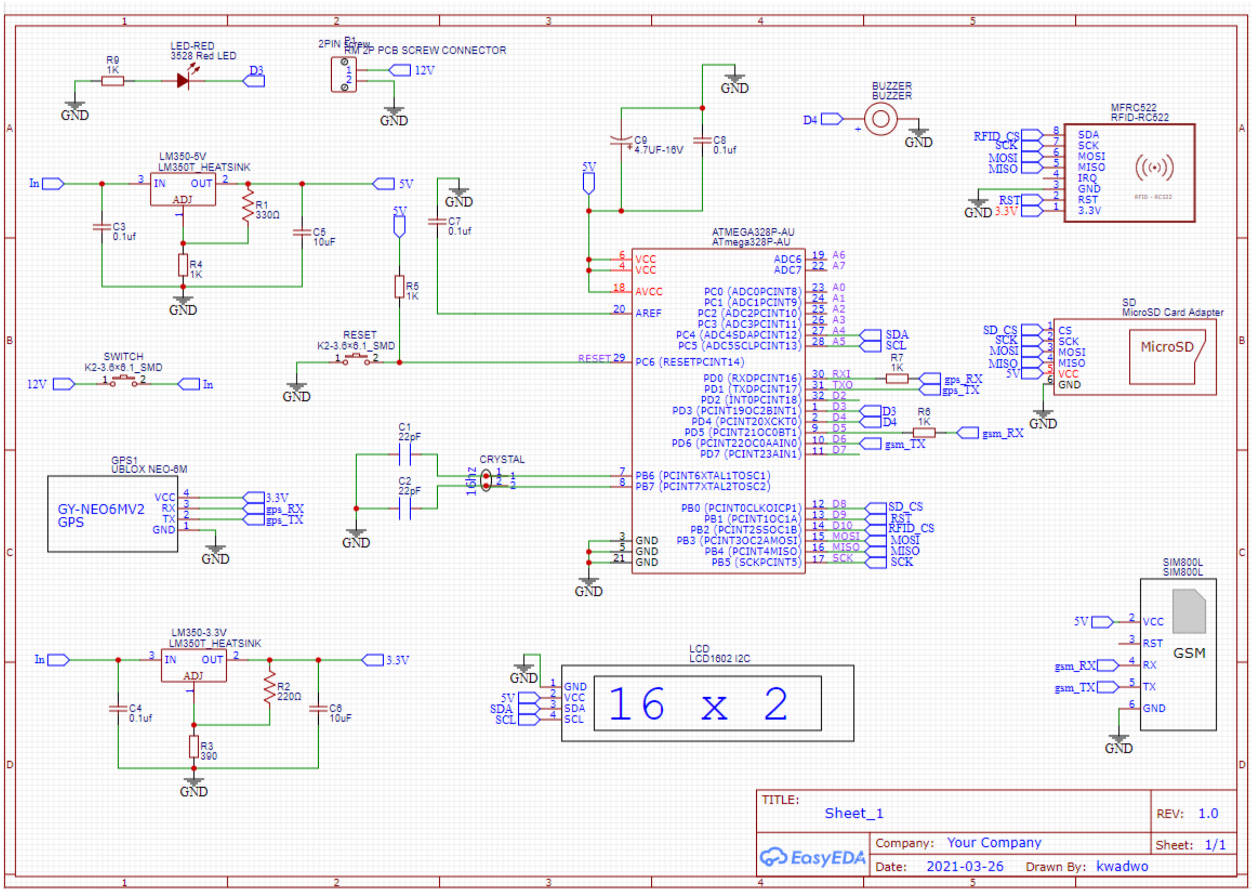


Figure 4.4 – Schematic is showing a connection between components.

EasyEDA is the tool used to design this schematic. There is a detailed connection of the components to each other and the extra connections required to get the components fully operational on this schematic. A PCB design was made from the schematic design, and this was also done using EasyEDA. The purpose of the PCB design is to generate a circuit board with the components connected to form one complete working system based on the schematic design.

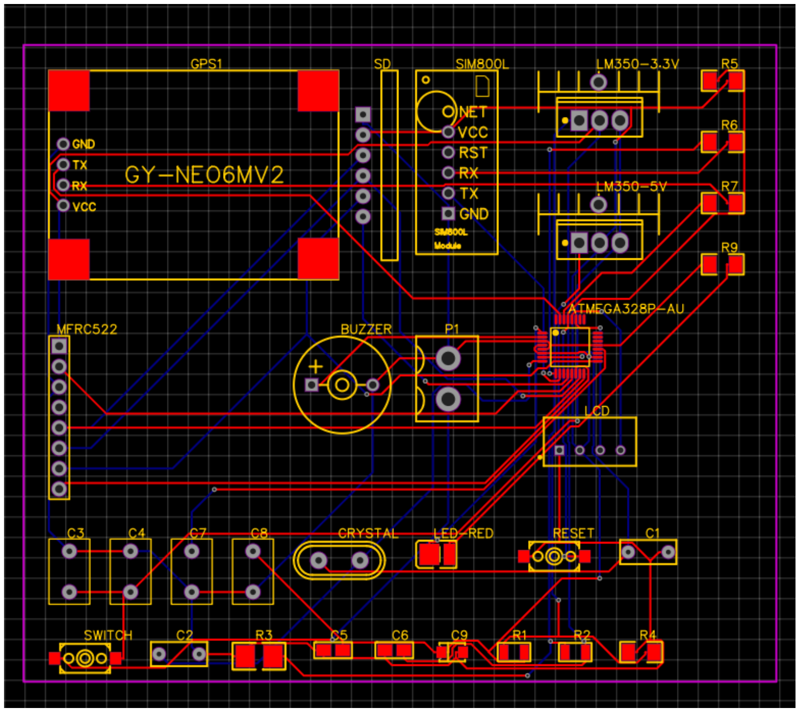


Figure 4.5 – PCB design front view

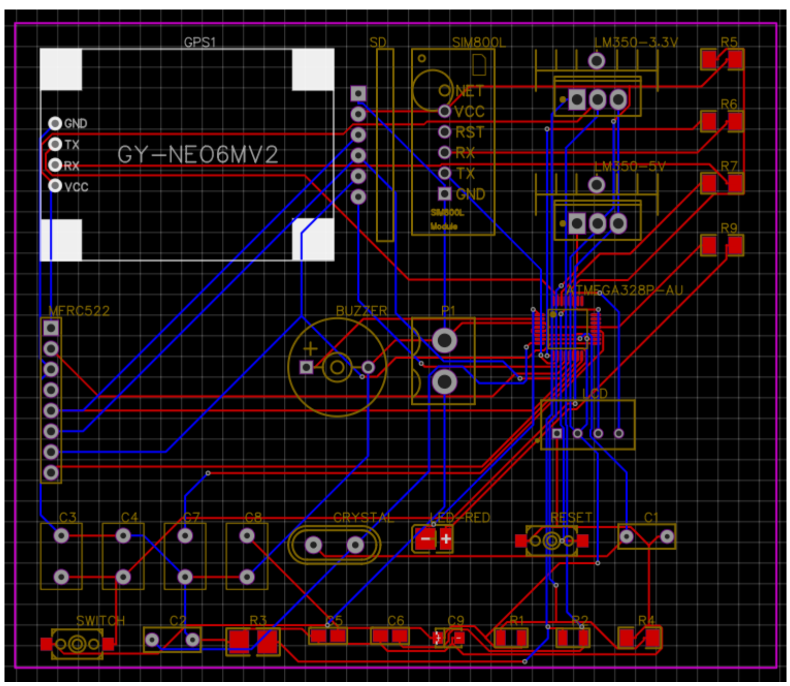


Figure 4.6 – PCB design rear view

The PCB has a width of 0.254mm, and the average Via diameter is 0.610mm with a drill diameter of 0.305mm. The finished product for this system will use the SMD components. SMD components because PCB with SMD components is easy to make, fast, and less expensive. The SMD components' sizes also make it possible to have a small PCB with enough space for routing.

4.2.2 Programming Atmega328P chip

After component selection and circuit and PCB design, the next step is the embedded code to give the entire system purpose. This code is uploaded to the microcontroller, the system's central unit with every other component connected. The Atmega328p chip acts as the system's brain with the computation power to process basic instructions from the code to operate the components connected. The atmega328p used for this system has the Arduino bootloader on it, and this chip can be programmed using a USB to Serial adapters like the CH340 or the USBasp or other components that perform the same function. In this project, the Arduino Nano was used as an ISP to upload to the atmega328p chip. This was done by uploading the ArduinoISP code onto the Arduino Nano. With the connection in Table 4.3, code upload was made possible.

Atmega328p-au pins	Arduino Nano pins
MOSI on pin 17	MOSI on Digital Pin 11
MISO on pin 18	MISO on Digital Pin 12
SCK on pin 19	SCK on Digital Pin 13
VCC and AVCC	5V
GND	GND
Reset on pin 1	Digital pin 10

Table 4.3 – Arduino and Atmega328p programming connection

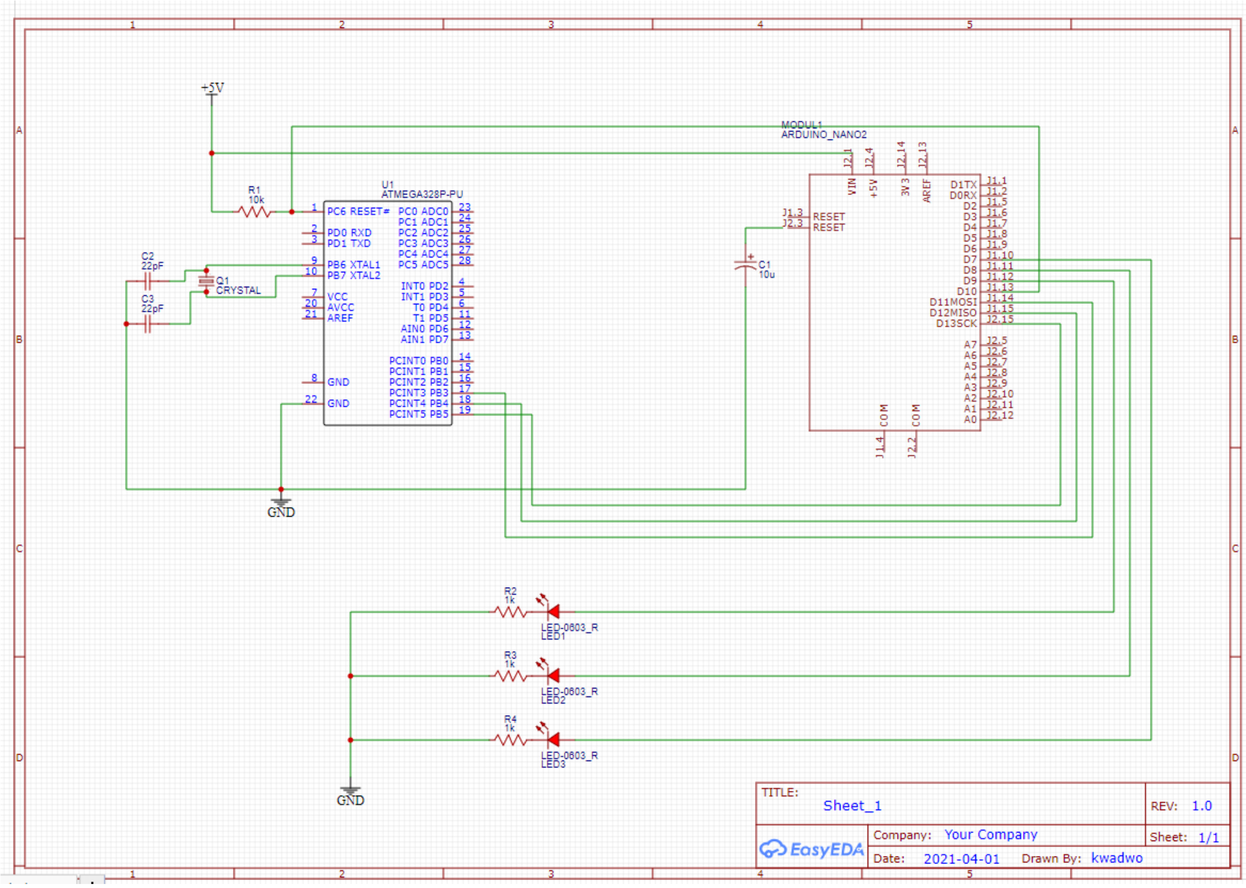


Figure 4.7 – Schematic of Arduino and atmega328p connection for code upload

The schematic in Figure 4.7 has LEDs connected, and this acts as an indicator showing what is happening in the system. Table 4.4 shows what the LEDs mean when they light up.

LED digital pin.	Indicator.
Digital pin 7.	The program is running.
Digital pin 8.	There is an error.
Digital pin 9.	Arduino is communicating with the slave.

Table 4.4 – LED digital pins and what they are indicating.

After the connection, Arduino uploads the code to the atmega328p as a programmer.

4.3 **Database and API Implementation**

In order to keep records of data on the users of the system, a database is required. The database will store user personal information and user transactions. The database was implemented using MySQL. Different tables were created in the database based on the data collected from both the users and the system. The database was hosted on Heroku using the addon ClearDB. The APIs to send and retrieve data to and from the database were implemented using Python Flask Restful API and hosted in Heroku. Figure 4.8 shows the different tables and columns needed for this system.

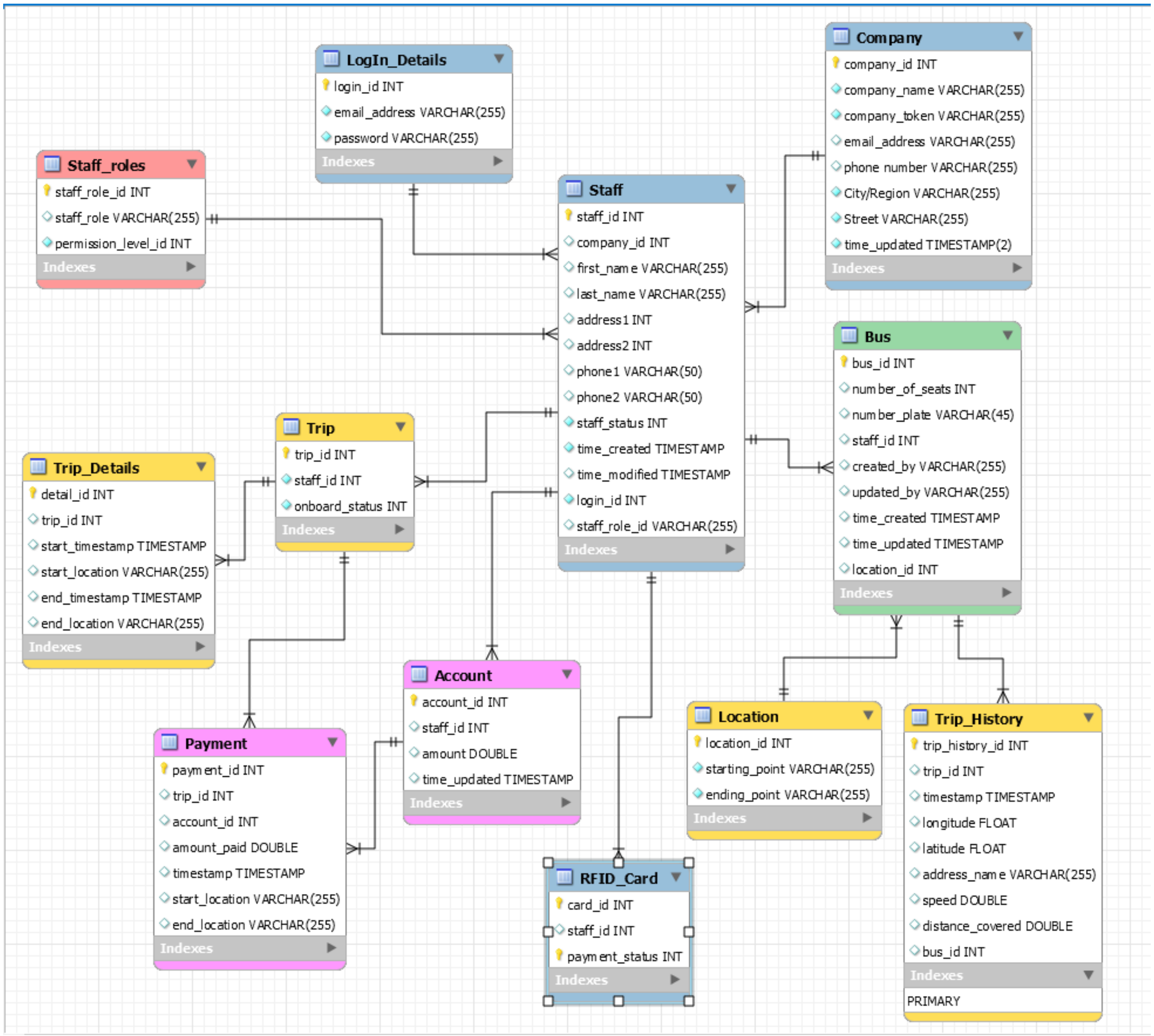


Figure 4.8- Database model

When a company signs up, a random token is generated for that company. As seen in the "company" table, this token is what the company's staff need to identify themselves as a member of that institution. During the signing up process of the staff members, they are required to provide this token then a request will be sent to any user with administration privilege to approve their request to confirm they are of that institution. The "Login_Details" table contains the information the user needs to log in to the application, and the staff roles are where the staff is assigned a role to determine their privilege level in the company. The system has three roles: the administrator role, the driver role, and the passenger role and privileges given in that order from the highest to the lowest. The staff member with driver privileges besides the administrator accesses the Bus table containing information on the bus. Connected to the "Bus" table is the "Location" table, which contains information on the starting and ending point of the bus for any trip made, and the "Trip_History" table, which contains information on the current position of the bus at a particular time.

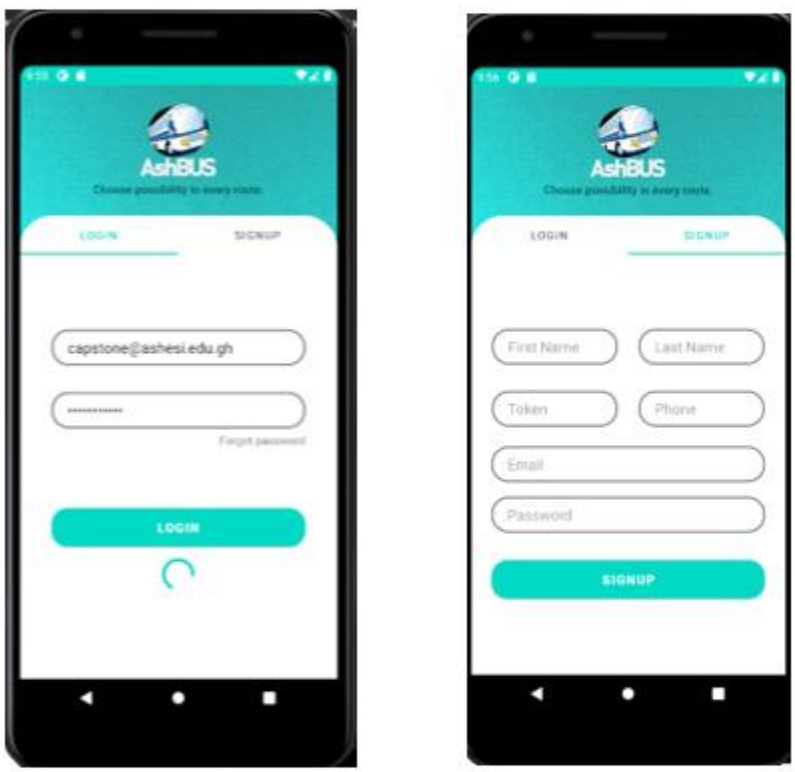
The system was designed to have all operations revolving around the staff members of each institution, so from the database model in Figure 4.9, you can see that connected to the "staff" table are "Account," which contains information on the user's account. There are the "Trip" and "Trip_Details," which contain information on the user's trip history from the point they board the bus to the point they got off the bus. The last table is the "RFID_Card" table containing information on the user's RFID card used for payment.

This "Trip History" table allows the system's admin to view all the places the bus has been to. A new column is added whenever there is a change in coordinates. This is to prevent the repetition of coordinates, especially during traffic. All information recorded would not be deleted from the database, including the bus trip history and user payment history, because Ashbus is a commercial system.

4.4 **Mobile Application Implementation**

The mobile application will be the platform the users will use to interact with the system. The application was built in Android Studio using Java as the backend language and XML as the front-end language. The application makes it easier for the users to check the bus's position at any given time using an integrated map. Upon logging into the application, the user can check their current position and a company bus around them. The user is also able to view how long it will take the bus to reach their position.

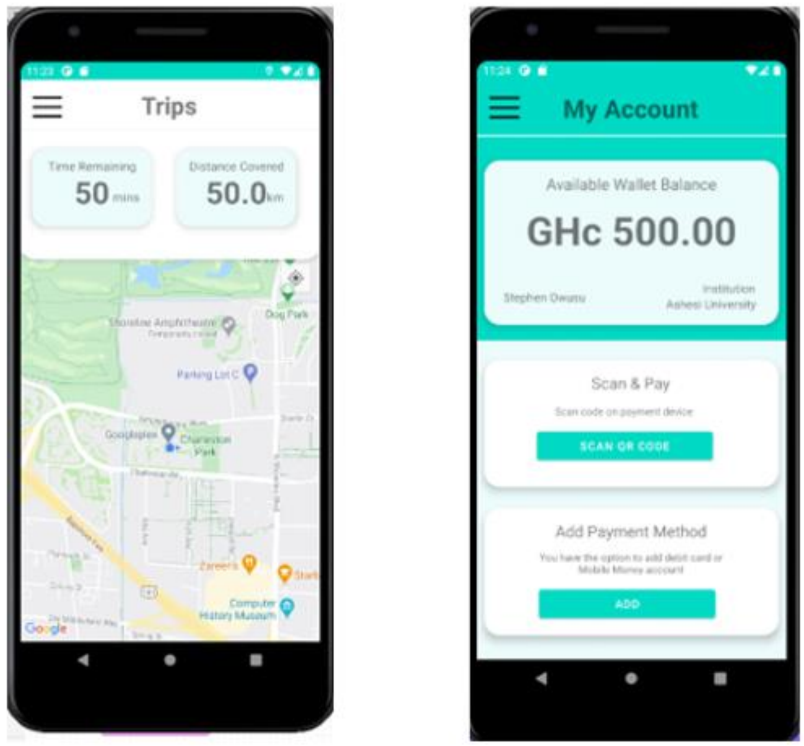
Under the payment UI, the user gets to view the amount left in their account and can make payment with their phones by scanning the QR code on the payment device on the bus.



(a) (b)

Figure 4.9 – Mobile Interface for login and signup

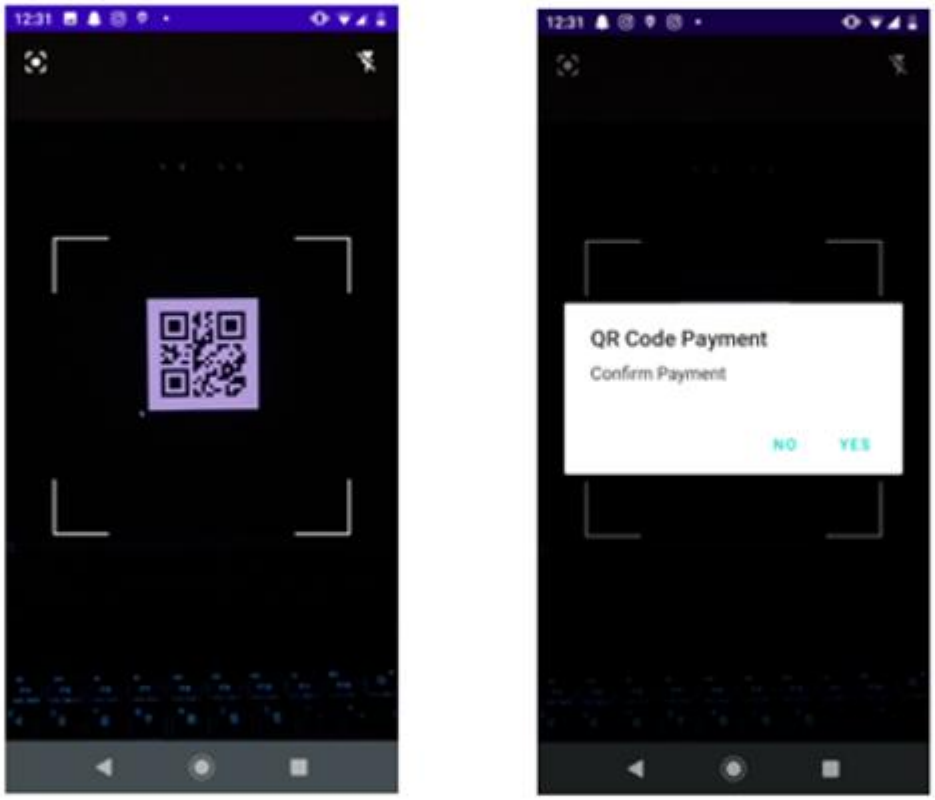
Figure 4.9a is the login UI, and the progress bar shows the logging-in attempt after the username and password were provided. Figure 4.9b is the signup UI which takes in the necessary data to create a new account.



(a) *(b)*

Figure 4.10 – Mobile Interface for both tracking and payment

Figure 4.10a, the integrated Google Maps indicating the user positioning, the bus position, time remaining, and distance covered. Figure 4.10b is the account UI which shows the amount in the user's account.



(a) *(b)*

Figure 4.11 – Mobile Interface for QR code payment

In Figure 4.11a, the QR code being scanned was generated with the company's ID number, and in Figure 4.11b, an alert to confirm payment. The system makes payment when "YES" is selected and cancels transactions when "NO" is selected.

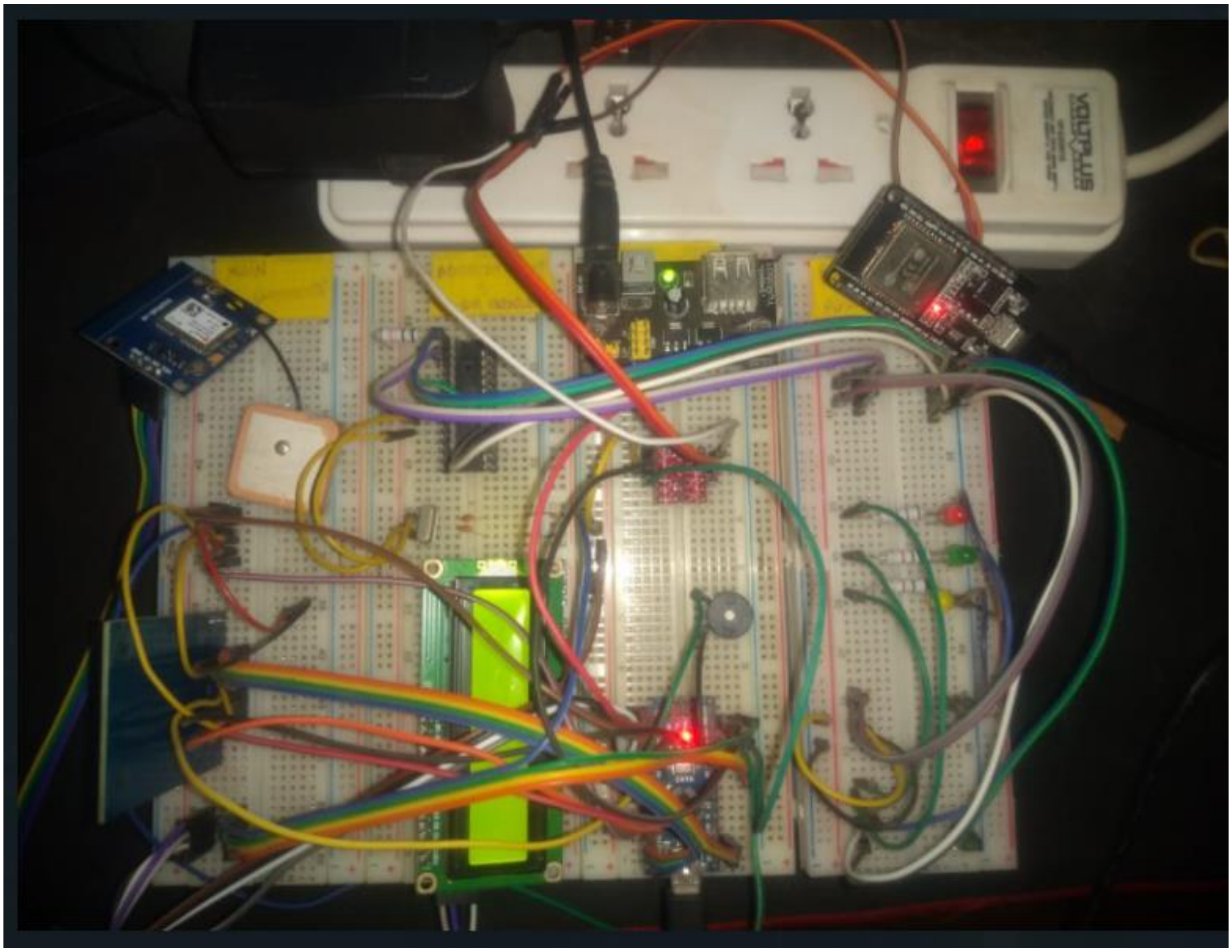


Figure 4.12 – Hardware prototype of the AshBus System

Chapter 5 - **Testing and Results**

In this chapter, the various tests that have been conducted and their results were recorded. The goal of these tests is to ensure that the system meets the requirement specification for the system.

5.1 **Test Description**

The test will be based on the requirement specifications of the system. There will be a test on the hardware device to check concurrency, consistency, accuracy, and performance. There will also be a regression test on the mobile application to check if the features meet the requirements.

5.2 **User requirement Testing and results.**

5.2.1 Functional Requirement testing and results.

Requirement number	Functional requirements	Testing	Results
REQ-1	Create an account.	Successfully signed up a new user with the mobile application	PASSED
REQ-2	Check transaction history.	PENDING	PENDING
REQ-3	Make payments by scanning the QR code.	Successfully made payment with the application by scanning a generated QR code with the company ID	PASSED
REQ-4	Make payments by scanning RFID cards.	Placed RFID card near RFID reader and data was read successfully.	PASSED
REQ-5	The user has the option to pay with cash.	Passed	Passed
REQ-6	See the estimated time for bus arrival.	Pending	Pending
REQ-7	Make payment even when the system is offline.	The RFID does not work when the SD card module is connected. Therefore, was not able to test for payment in	Partial

		offline mode. But without the RFID, the SD card module can save data successfully	
REQ-8	See the available seats on the bus.	Pending	Pending
REQ-9	See bus location.	Bus location can be seen on the map.	Passed
REQ-10	Check transaction history.	Pending. Transaction history is currently only possible on the API level.	Pending
REQ-11	Approve registration request from workers.	Pending	Pending
REQ-13	Can assign authorization level based on position.	Assigning authorization level is currently only possible on the database level and not the application level.	PENDING
REQ-14	Can top up a user's account.	Top-up user API is fully operational and is available for only users with admin privileges.	Passed

5.2.2 Non-Function Requirement testing and result.

Performance Requirement	Requirements	Testing	Results
	The mobile application should be stable.	The system is stable. The mobile application is not complete.	Partial
	The mobile application should be responsive.	All features respond, and interaction with the system is easy.	Passed
	The RFID reader should be responsive.	RFID card was scanned two times in a row, and the reader picked on both.	Passed
	The system should be able to operate even when offline.	Failed	Failed

	The system should not have any form of glitch.	No glitches so far	Passed
Safety and Security Requirement			
	The system should sanitize data before sending it to the database.	Passed	Passed
	There should be a security policy.	Passed	Passed
	The system should not allow default passwords like “admin” and “password.”	Passed. Created a new account with these default passwords, and account creation failed.	Passed
	Account should be suspended after three failed attempts.	Pending	Pending
Software Quality Attributes			
	The mobile application should be reliable.	Failed. The database runs on a free tier online server. Due to this, connection to the database is not stable because the API takes a long time to establish a connection with the database and times out as a result sometimes. This makes it not reliable. This problem has been checked by	Failed

		giving the user the chance to try again.	
	The mobile application should be robust.	Pending. The mobile application is not complete.	Pending
	The mobile application should be maintainable.	Pending. The mobile application is not complete.	Pending
	The system should be flexible and adaptive.	The system can easily switch in and out features based on the user permission level	Passed
	The system should be reusable.	The system is reusable.	Passed
	The system should be friendly.	Easy for the user to navigate.	Passed

5.3 **System Requirements**

Requirements	Testing	Results
The system will have a user from each institution with admin privileges.	Passed. This was checked in the database when a user was identified with admin privileges	Passed
The system will have fixed charges irrespective of destination.	Passed. The charge for all trips is stored in the database.	Passed
The system should work both offline and online.	Offline implementation works, but the RFID fails to work when the SD card is connected.	Failed

The system supports multiple payments by a user.	Multiple payments have been integrated into the payment API, but both hardware system and mobile application by default make only one payment.	Partial

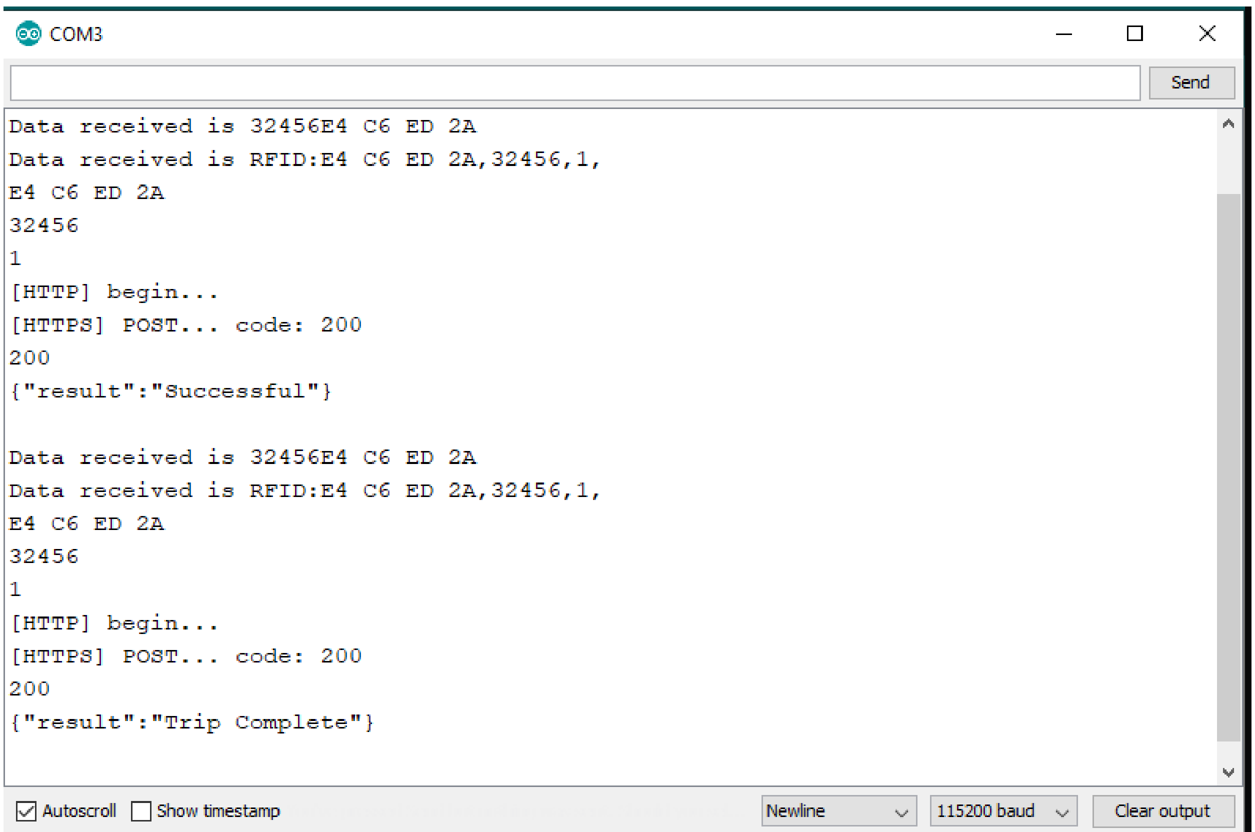


Figure 5.1 – Payment Made and Trip Complete

Figure 5.1 shows the RFID being scanned twice; the first one was when the user was getting on board the bus hence the HTTP response "Successful." This means payment was successful. The second scan, the user has reached their destination and is going offboard. The HTTP response this time is "Trip Complete." On the second scan, payment is not made.

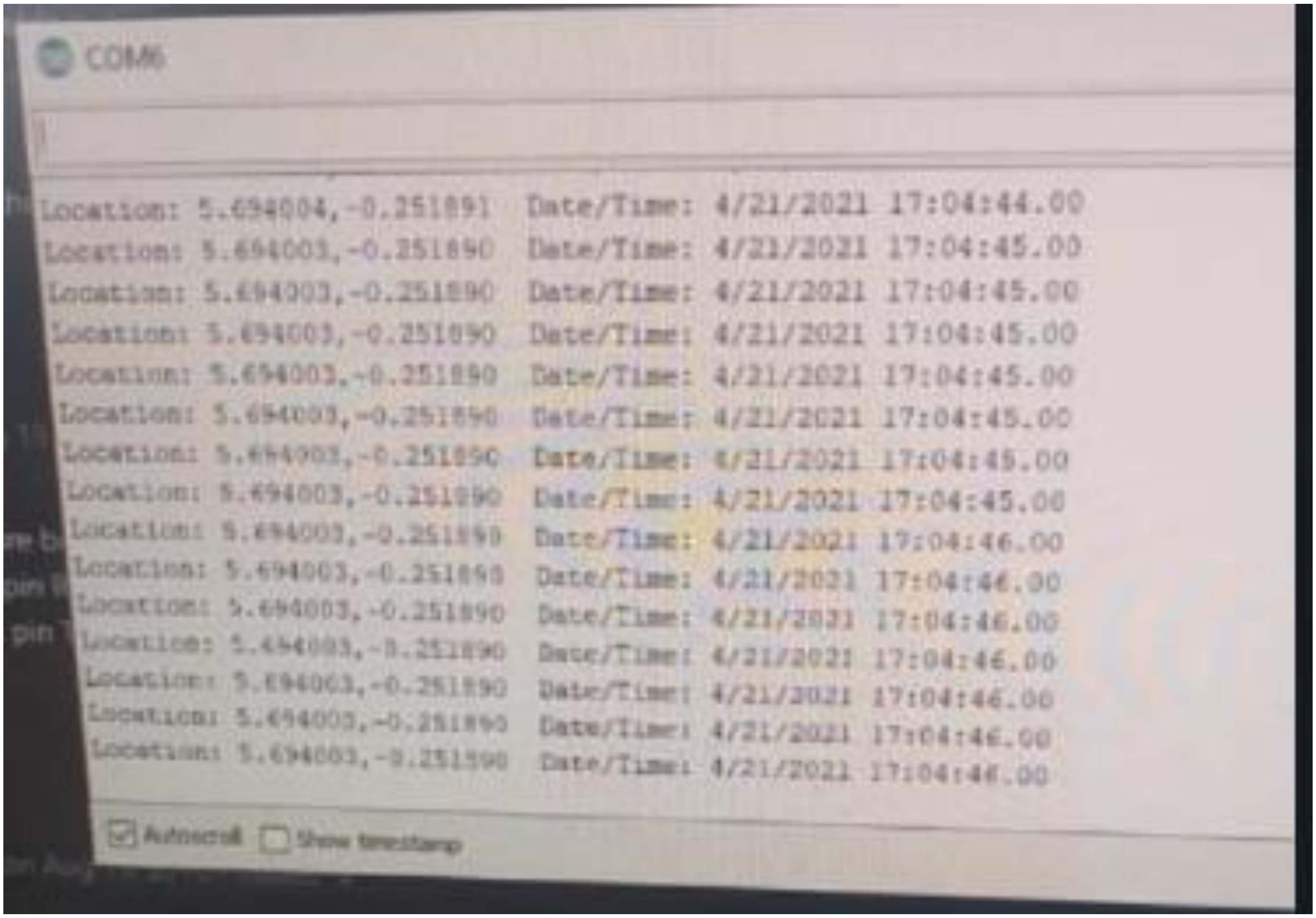


Figure 5.2 – GPS data coordinates received from GPS module.

Figure 5.2 shows prove of GPS working. The API to handle data transmission for the GPS module is not ready; hence the tracking system is partially complete.

Chapter 6 - **Conclusion**

The GPS-based tracking system will make tracking the location of the bus easy for staff members. This system will also make it possible for the staff members to know the estimated time of arrival of the bus to their current location. The RFID cashless system for transportation adopted in the Ashbus system will make payments very easy, and it also helps track the user's location from start to endpoint. These two solutions solve our problem and make transportation and payment easier for the staff members.

6.1 **Limitations**

Some constraints of this projects are:

- **a.** The GPS module works well when in an open space; hence, enclosed areas will affect the module's performance.
- **b.** The system does not currently transmit data using the GSM module but uses a Wi-Fi module due to a faulty GSM component.
- **c.** The mobile application interacts with the device on the company's buses; hence, users without a smartphone cannot take advantage of the system.
- **d.** There is no mobile application build for IOS yet.
- **e.** The current solution does not support multiple payments by a user.

6.2 **Future Work**

- The scope of the project can be expanded from only company buses to public transportation. The only problem is that public transport passengers are all from different places; hence, their information cannot be verified as 100% authentic. This could result in fraud. The system right now uses company ID to authenticate the cards on the bus, and the companies handle the user accounts. However, since public transports do not belong to an organization, the system would have to change card authentication and system management. The Oyster payment system idea can be used in this case.

- For account top-up or a different means of payment, the system will have a mobile money integration. The current system does not have this feature because staff can make payments to the accounting office and update their accounts. Using the Oyster payment system idea, top-up would not be handled by Ashbus.
- The system has fixed charges for all trips and does not have an option for different charges based on destination. This is another feature for future work. This feature makes it possible for public transport to adopt the payment system.

REFERENCE

[1] K. A. Mensah, “A tracking and billing system for commercial vehicles,” 2019, Accessed: Apr. 24, 2021. [Online]. Available: https://air.ashesi.edu.gh/handle/20.500.11988/540.

[2] H. A. Abdallah Dafallah, “Design and implementation of an accurate real-time GPS tracking system,” in *The Third International Conference on e-Technologies and Networks for Development (ICeND2014),* Apr. 2014, pp. 183–188, DOI: 10.1109/ICeND.2014.6991376.

[3] P. B. Fleischer, A. Y. Nelson, R. A. Sowah, and A. Bremang, “Design and development of GPS/GSM based vehicle tracking and alert system for commercial inter-city buses,” in *2012 IEEE 4th International Conference on Adaptive Science Technology (ICAST),* Oct. 2012, pp. 1–6, DOI: 10.1109/ICASTech.2012.6381056.

[4] R. Bajaj, S. L. Ranaweera, and D. P. Agrawal, “GPS: location-tracking technology,” *Computer*, vol. 35, no. 4, pp. 92–94, Mar. 2002, DOI: 10.1109/MC.2002.993780.

[5] K. M. McKay, “Integrated Automatic Vehicle Location Systems,” IEEE AES Systems Magazine, pp. 18-22, March 1997.

[6] W. Ametepe, D. Abdullai and N. K. Annan, “Examining Issues of electronic payment systems in Ghana,” Information and Knowledge Management, vol. 5, no. 5, 2015.

[7] H, Glickenstein, “Contactless Payment Debuts on London’s Buses, New and Improved

Railways around the world," IEEE Vehicular Technology Magazine, pp. 19-25, June 2013