Technical Report

For

MobiGarage

Prepared by

Group Name: 11

20/U/5742/PS	**2000705742**	**MOOLI BRIAN**
20/U/5779/PS	**2000705779**	**YIGA GILBERT**
20/U/5656/EVE	**2000705656**	**KAYANJA EMMY WILLIAM**
20/U/0186/PS	**2000700186**	**KIBIRANGO KENNAN**
20/U/5680/EVE	**2000705680**	**KYAGULANYI DEO**

Mentor: Mr. Asiimwe Paddy Junior

Course: CSC 1304 Practical Skills Development

Date: 14/02/2022

Contents

Abstract

Locating the nearest garage, spare part centers, and mechanics in case of emergencies has been one of the greatest challenges over decades in case of a mechanical breakdown.

According to a survey carried out by Group 11 members across the country, most drivers get stuck on the way to their destinations due to failure or delay to get mechanical services. This leads to several inconveniences such as missing very important appointments, increased damage in case one tries to forcefully move the vehicle to reach the nearest garage, accidents where people lose their dear lives, to mention but a few.

Viable solution

In order to address the above problems, the team has come up with MobiGarage. This application will give room for interaction between drivers, mechanics and the garage management. MobiGarage will enable drivers to easily locate Oriental Motors Garage or contact a mechanic that can work on their specific problem in case of emergencies or when in transit.

MobiGarage will also enable drivers to rate the mechanics, raise complaints and also give comments depending on how best the problem the problem has been addressed.

We strongly believe that MobiGarage will be a very useful and easy-to-use tool for mechanics at Oriental Motors Garage and drivers as explained in this document.

1 Introduction (max. 2 pages)

This Report introduces MobiGarage, how it has been made and how it works. It also reviews the team's development process, successes and failures. The report is in three phases: Introduction, Project Results and then Limitations and Next Steps.

1.1 User Challenge

Most drivers always get stuck on the way to their destinations due to failure or delay in getting mechanical services in case of emergencies on the road since locating the nearest garage, spare part centres and mechanics is a very hard pin more so if someone is in a remote area, a highway or a place they are not familiar with. Sometimes it requires that someone have a personal mechanic they can call who in most cases is very far and cannot reach them urgently or even does not have the items to fix some issues. In the end they find themselves missing very important appointments, having increased damage in case they try to forcefully move the vehicle or motorcycle to reach the nearest garage or even lead to worse accidents where people lose their lives.

In such a dilemma, being able to find a credible mechanic who is in your proximity, has the tools to work on your issue and has been rated to be the best for working on that particular issue could be the most relieving and refreshing message that can hit anyone's ear. The only question is how? This is where MobiGarage comes in to make sure the problem is fixed with possible inconveniences expelled at the basic comfort of your Mobile Phone.

1.2 Project Goals

MobiGarage was invented by group 11 members. Their vision was to build an application that allows drivers to easily locate Oriental Motors garage, contact the garage in case of an emergency that calls for mechanical attention. It also to enables drivers buy spare parts at their convenience just using their mobile phones.

MobiGarage has been built with several features that ease access to mechanics from Oriental Motors, book car services, FAQ, and spare parts in case one is in need. In case one has gotten an accident or an emergency on the road and is in a remote area or a place they are no familiar with, the application provides a list of the problems that are common with different vehicles and helps to specify the different mechanics who can handle the problems. This helps the driver to contact the mechanic, explain their issue, determine its gravity and decide whether the mechanic should come to them or tow the broken down vehicle to the garage.

 The application also helps the drivers to avoid conmen who sell substandard spare parts from non credible sources at exaggerating prices.

Functional Requirements
Below are the Functional requirements of the MobiGarage Application.
1. Location: Users of this application are able to locate Oriental motors garage to access mechanical services the time they need attention.
2. Technical advice through FAQ: The users can be able to get technical advice through the FAQs provided in the application.

3. Booking Car services online: Any user should be able to book a service for their car, input car details and personal contact details. Though services that can be accessed are limited to three categories: Critical, One-time Service and Periodic Service.
4. Order for Spare-parts: Any user should be able to view and order for spare-parts from the garage.

1.2.1 **Functional Requirement 1: Location.**

Purpose: All users of the application especially drivers is able to view the contact address of Oriental motors garage and reach its location to access the services required.

Process: The user of the application acquires the address Oriental motors garage and gets to its location. If the user is not certain of its location, he/she can use google maps to locate the garage and reach there conveniently.

1.2.2 **Function Requirement 2: Technical advice through FAQ.**

Purpose: The application provides technical advice to the users which consists of the frequently asked questions that give general guidance to the users in case they are stuck.

User Input: FAQ

Process: The user of this application logs in and open the home page.
Look for the FAQ button on the home page and click in it.
A list of frequently asked questions and their corresponding responses is displayed.
The user reads through the list and identifies the question corresponding to his/her challenge.
The user eventually reads the attached response which provides the solution to the problem faced.

1.2.3 **Functional Requirement 3: Booking Services**

Service Booking is categorized into two users; those that are using the app for the first time and have an emergency and those that have been customers before and have accounts as to be shown below:

Purpose: To enable a customer book a service but also help the garage schedule and know when to handle which customers and understand which cases to consider priorities.

User Input: Signing in, car details, servicing dates and time periods.

Process:
1) The user taps the booking button on the home page which bring them to the booking page.
2) The booking page displays three options: Critical Service, One-time Service and Periodic Service.
3) Critical Service can be accessed by anyone even without an account on their first time to use the app.
4) On tapping the Critical Service, the user is taken to a page with contact details that the garage team responds to urgently.

5) If the user chooses the One-time service, they are required sign in and if they have no account, they are required to create one.
6) They are then required to enter their car details which are studied by the garage team
7) They are then given an appointment which is scheduled for the service of the car.
8) Choosing Periodic Service is similar to the One-time Service since it requires signing in and entering car details.
9) The difference is that this time the user is allowed to schedule and set periods of time when they need their car serviced.
10) The app then sets an alarm that notifies the user of when to take their car for servicing.

Output:
- Emergency contact details for the Critical Service option
- Appointment details including date and time that the car can be serviced for One-time Service Option.
- Organized Schedule of when the car is to be serviced. Notification when car servicing is due. This is for the Periodic Service option.

1.2.4   **Functional Requirement 4: Booking Spare parts**

Purpose: Enable users with accounts to book spare parts
**User Input:** Sign in details and spare part name

Process:
1) The user is required to go to either the spare part page or tap a spare part image on the home page.
2) They can also search for the needed spare part in the provided search text field.
3) On tapping on the needed spare part, the user is required to have an account and therefore sign in if they hadn't.
4) The user is then allowed to choose a spare part and his order is stored in the database. They are also given details for pick-up of the product.

**Output:** Pick-up details of the booked spare part.

2 Project Results (max. 3 pages)

2.1 Project Design

2.1.1 The application has been designed to have two end users. That is the driver and the Oriental motors manager. To fully use the app, one ought to have installed the app and must have access to the internet. This links them to the management system which then gives them access to the cloud-based server. This is better explained by the architectural design diagram below.
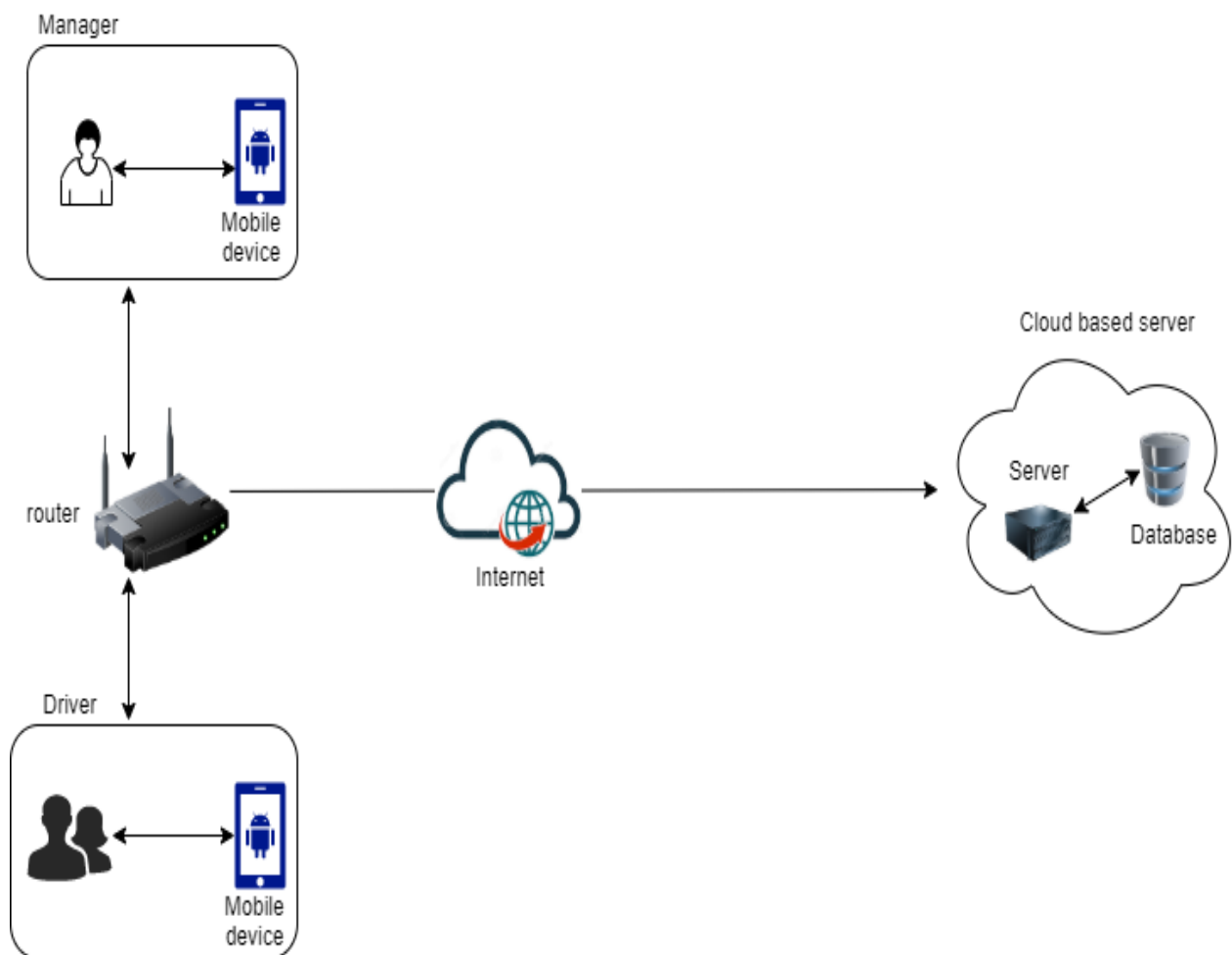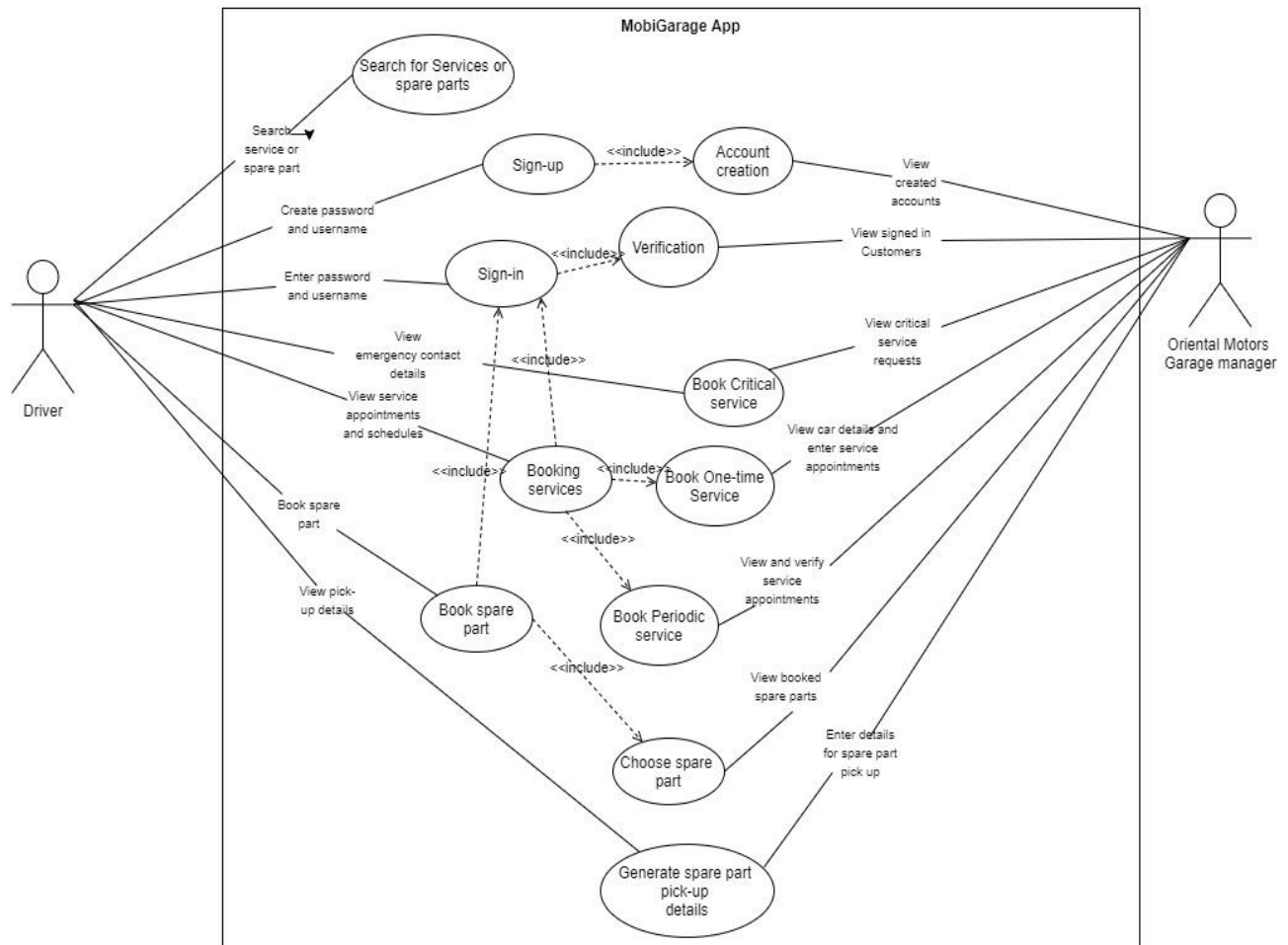
2.1.2   The app deals with mainly two entities and below is a context diagram to describe how data flows between the app and these external entities.
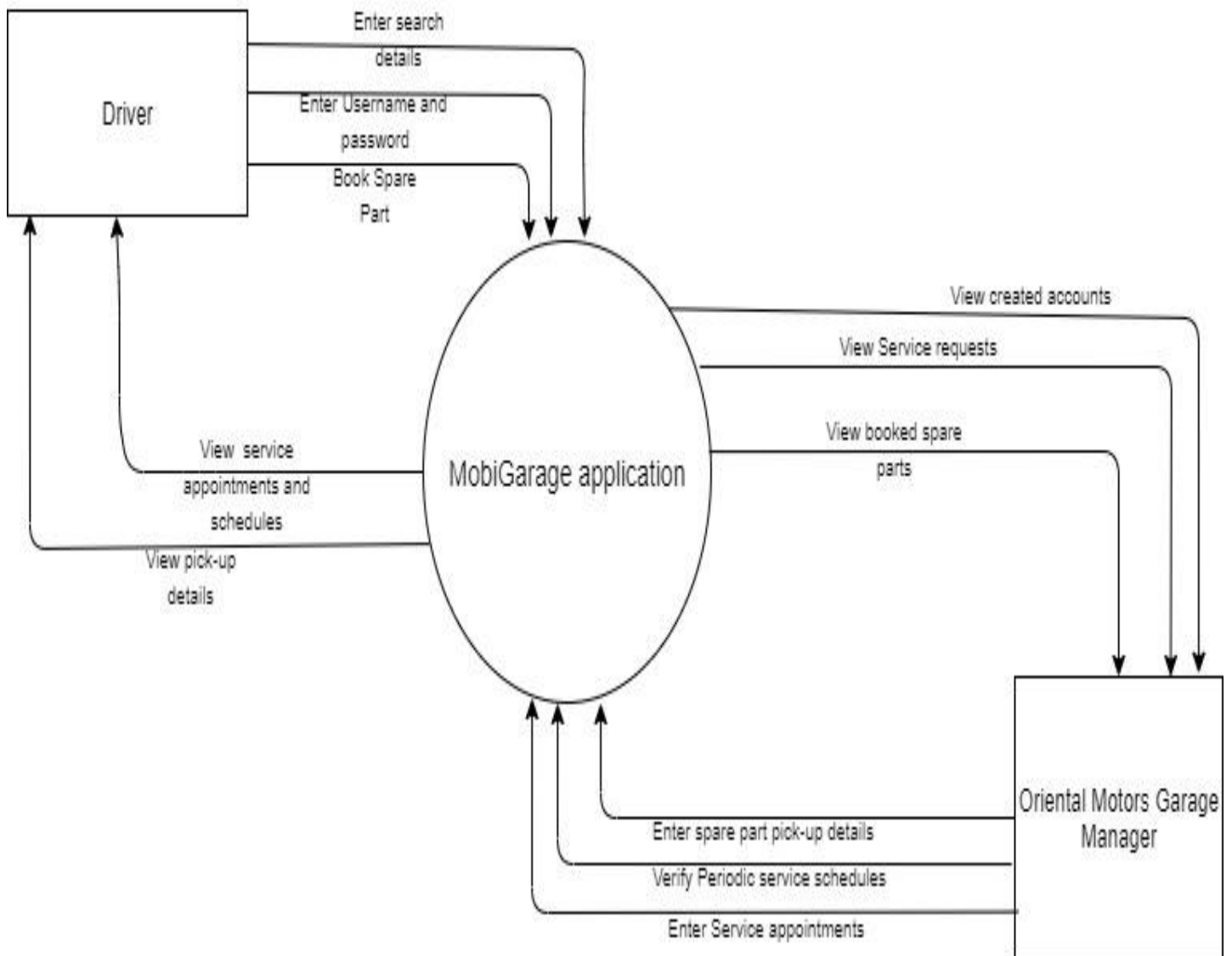
2.1.3   The App mainly has two end users, that is: the Driver and the Oriental Motors Garage
manager as shown in the Use Case diagram below with the different functionalities they
engage in.

2.2 **Product** Functionality and Screenshots

2.2.1   **Home Page with Search field main application menu and Booking button**

On this page, any user should be able to search for any spare part, service or mechanic. User should enter the required service or spare parts in the search box. As result, the user is able to reach the desired page whether service page or spare part ordering page. The home page is displayed below.
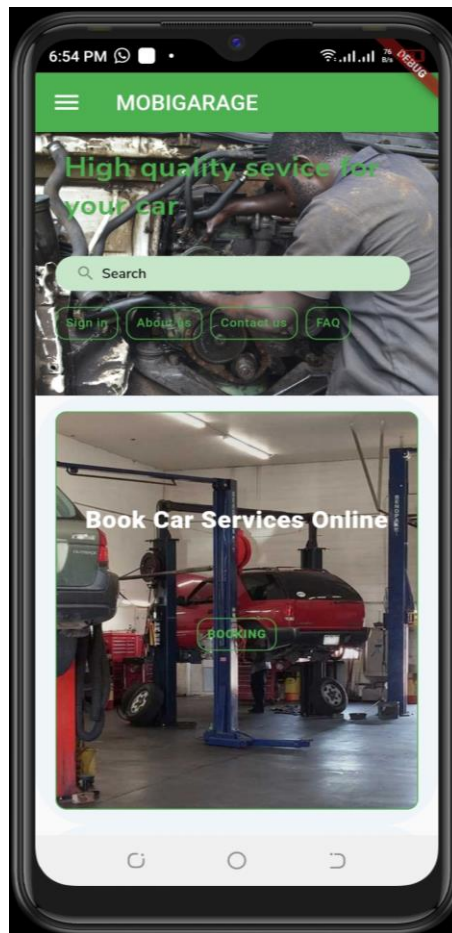

2.2.2   **Sign in**

The sign in page provides a field for the user to input their username and password if they have an account. If they don't have it provides an opportunity for a driver to create an account with the app by creating a new username and password. This is shown below in the sign-in page.
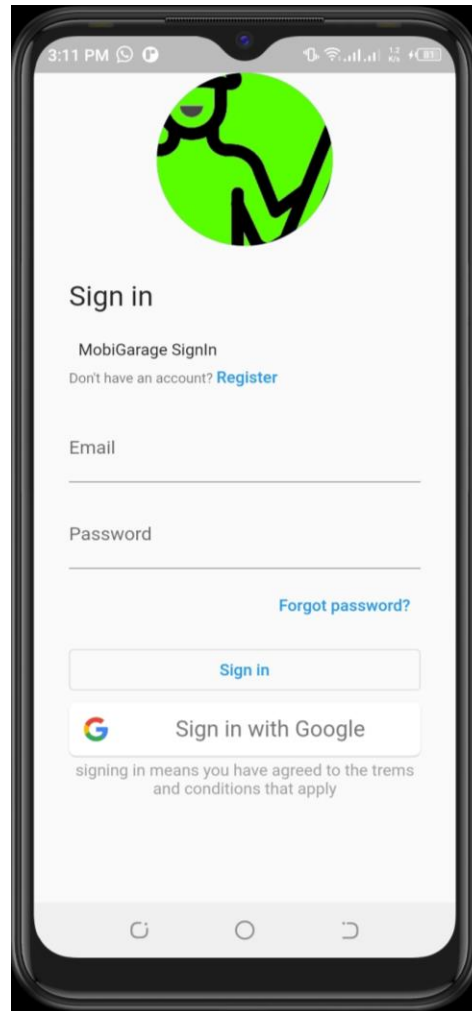

2.2.3   Booking page

This is a functionality page that the user reaches after tapping the booking button on the home page. The page displays three booking options: Critical Service booking, the One-time Service booking and Periodic service booking. Critical Case works with any client with an emergency even if they have no account yet. Then the One-time and Periodic Service booking is restricted to only drivers with accounts. The application generates a unique identification number for each service booked. These are shown below.
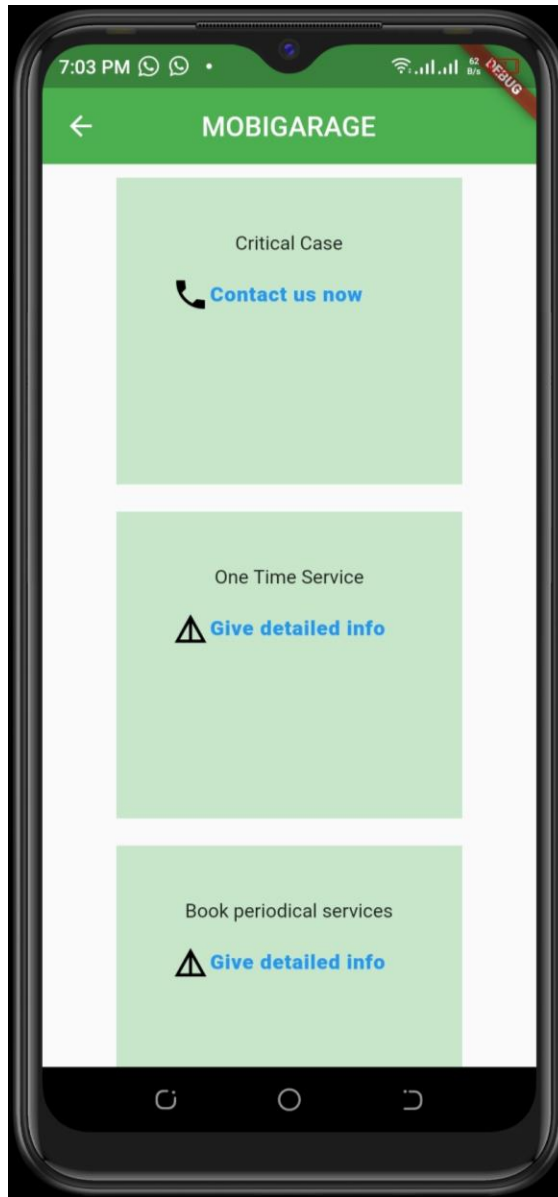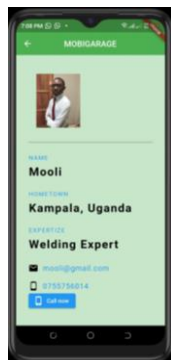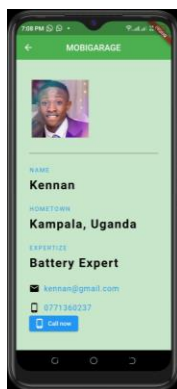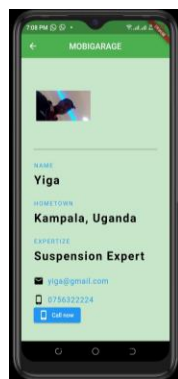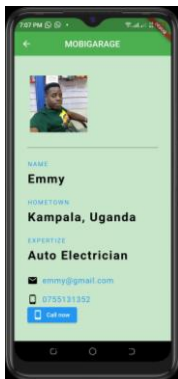
2.2.4 The Services

This mainly displays the different service options that a driver can request for. Each service option gives the driver access to the specific mechanic to work on the problem and also the booking details.
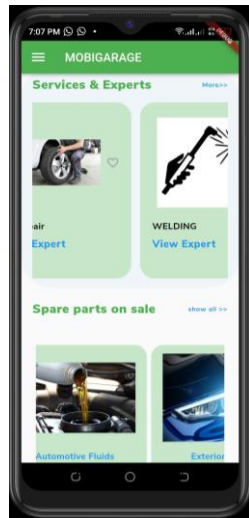
On Tapping More

On tapping view expert
displays mechanic suitable


2.2.5 Booking Spare-Parts

This enables the driver to search for the needed spare part in the search box, select and book the needed spare parts. On booking the spare part the user receives the delivery details of the booked spare part. The Application also allows the driver to add the booked spare parts to a cart. The application also generates a unique identification number for each order made or spare part booked. The Spare part display is shown on the page below.

Add to cart
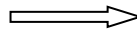
On booking, the spare parts booked by the driver are added to his cart on the app
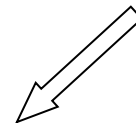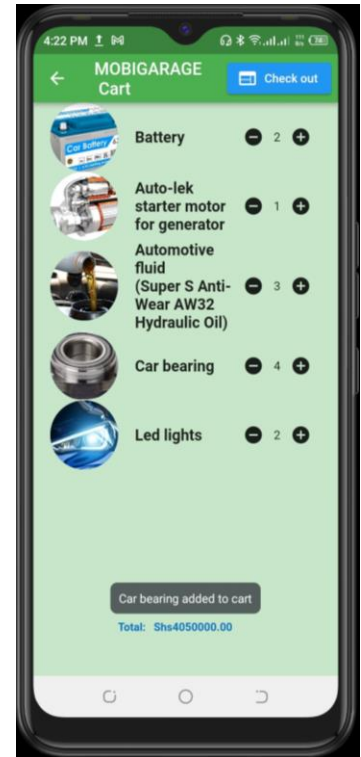
Confirming order

2.2.6   Contact Details

This simply takes to driver to the about us page where he is able to know more about it Oriental Motors garage. The application also provides contact details of the garage, location and also enables users make direct phone calls through the application by just tapping call now as displayed below:
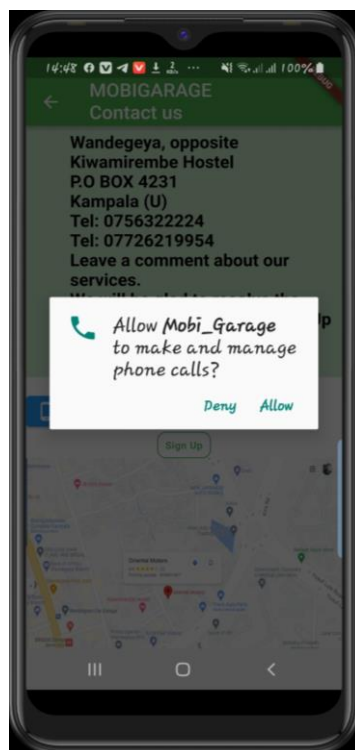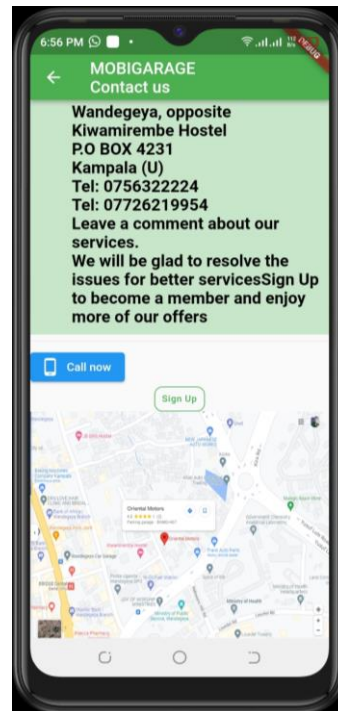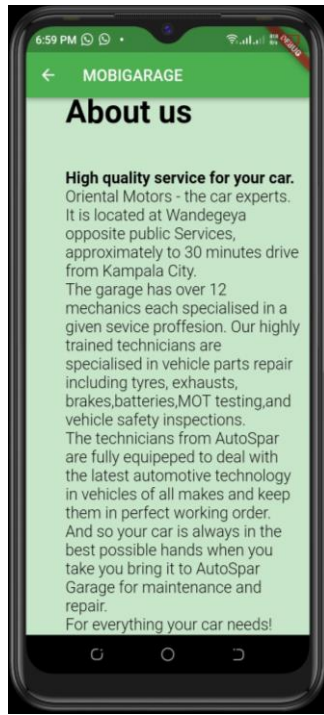
2.3 Project website and repository

2.3.1 **Project Website**

http://emmywilliams.github.io/

2.3.2 **Project Repository**

https://github.com/MobiGarage-Group-11/mobi_garage

3 Limitations and Next Steps (max. 1 page)

3.1 Limitations

We encountered a few challenges without which we the development process would have been somewhat less challenging. Here the limitations are categorised according to phases as to be explained below.

3.1.1 Requirements Collection stage

For Requirements Collection we were to visit and study Oriental Motors Garage. In this we were required to Interview the workers, look at the service and customer care of the garage and definitely have a clear picture of the Garage organisation and culture. Here is where we faced the most outstanding challenges as stated below.

- **Language Barrier:** The mechanics at Oriental Motors Garage were less acquainted with English -which was the language the team could express themselves easiest- and more acquainted with Luganda. This gave us a hard time in understanding the would-be practicability of the application.
- **Confidentiality Of the Garage manager:** the manager was not free to give us some of the information that would be profitable for development of the app. This we believe is because many of us were new to his eyes and he had met only one of us prior. This limited us from using actual details while developing the application and limited our data collection. The team was also not a loud to take pictures of the garage and the only record were writings and a video coverage of the interview process.
- **Limited Vehicles:** Oriental Motors Garage is one that works on only Toyota and Mercedes Benz vehicles which caused the application to be limited to service of only those vehicles.

3.1.2 Development Phase

The most inconveniencing limitation at this stage was the Limitation in time. This limited the number of functionalities that could perfectly be added to the application. The team was given a maximum of two weeks to accomplish the planning, analysis, designing, development and documentation of the application. This was a challenge in putting up a functional application, but the team tried its best.

Other than that, the whole development phase carried on smoothly considering the coordination and teamwork.

3.2 Next Steps

Here are several features that we plan to add in the application in the future

- Making payments through the application by the help of a mobile money API.

- Connecting a number of garages to this application to scale the application to the whole world.

References

 [1] *https://study.com/academy/lesson/project-requirements-definition-types-process.html#*

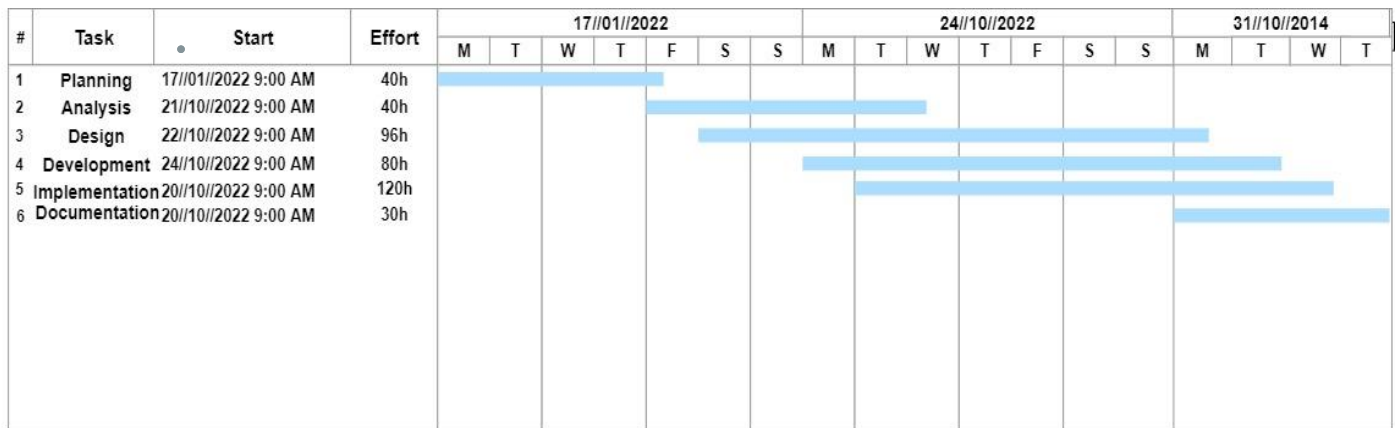[2] *https://www.slideteam.net/powerpoint/Application-Architecture*

[3] *https://www.jamasoftware.com/requirements-management-guide/writing-requirements/how-to-write-system-requirement-specification-srs-documents*

[4] *https://www.udacity.com/course/build-native-mobile-apps-with-flutter--ud905*

4 Appendix A – Project Work plan

4.1 A Gannt Chart to show the project Workplan that was followed by the team to develop the MobiGarage Application

#	Task	Start	Effort	17//01//2022							24//10//2022							31//10//2014			
				M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T
1	Planning	17//01//2022 9:00 AM	40h																		
2	Analysis	21//10//2022 9:00 AM	40h																		
3	Design	22//10//2022 9:00 AM	96h																		
4	Development	24//10//2022 9:00 AM	80h																		
5	Implementation	20//10//2022 9:00 AM	120h																		
6	Documentation	20//10//2022 9:00 AM	30h																		

Appendix B – Contribution by Team Members

No.	Team Member	Contribution
1.	Kibirango Kennan	• Technical report (24%) • Software requirements specification (19%) • Project Poster (19%) • Project website (19%) • Application Development (19%)
2.	Yiga Gilbert	• Technical report (19%) • Software requirements specification (19%) • Project Poster (19%) • Project website (19%) • Application Development (24%)
3.	Kayanja Emmy William	• Technical report (19%) • Software requirements specification (19%) • Project Poster (19%) • Project website (24%) • Application Development (19%)
4.	Mooli Brian	• Technical report (19%) • Software requirements specification (24%) • Project Poster (19%) • Project website (19%) • Application Development (19%)
5.	Kyagulanyi Deo	• Technical report (19%) • Software requirements specification (19%) • Project Poster (24%) • Project website (19%) • Application Development (19%)

End