# CARKILA

Project Documentation Submitted

To the Faculty of School of

Computing and Information Technologies Of

Asia Pacific College

In Partial Fulfillment of the Requirements for the subject

Applied Projects 2

By

Martha Angela A. Salamat

April 25, 2018

**ASIA PACIFIC COLLEGE**

# Approval Sheet

Carkila

Prepared and Submitted by:

Martha Angela A. Salamat

In Partial Fulfilment of the Requirements for the Degree of

Bachelor of Science in

Examined and Recommended for Acceptance and Approval for Research/Capstone Presentation

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Adviser

Panel of Judges

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Chairperson

Panel Members

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Panel Member Panel Member

Acceptance and Approved in partial fulfillment of the requirements for the degree of Bachelor of Science

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Executive Director

School of Computing and Information Technologies

# Abstract

Table of Contents

[Approval Sheet 2](#_Toc512488313)

[Abstract 3](#_Toc512488314)

[List of Figures, List of Tables, List of Notations 6](#_Toc512488315)

[I. Introduction 8](#_Toc512488316)

[1.1 Project Context 8](#_Toc512488317)

[1.2 Purpose and Description 8](#_Toc512488318)

[1.3 Objectives 10](#_Toc512488319)

[1.4 Scope and Limitation 10](#_Toc512488320)

[II. Review of Related Literature and Systems 10](#_Toc512488321)

[2.1 Traditional Car Rental Systems 10](#_Toc512488322)

[2.1.1 Hertz Car Rental 10](#_Toc512488323)

[2.1.2 Viking Car Rental 12](#_Toc512488324)

[2.2 Similar Systems in the Philippines 13](#_Toc512488325)

[2.3 Peer-to-Peer Rentals 14](#_Toc512488326)

[2.4 Peer-to-Peer Car Sharing 14](#_Toc512488327)

[III. Technical Background 15](#_Toc512488328)

[3.1 Development Tools 15](#_Toc512488329)

[3.1.1 PHP 15](#_Toc512488330)

[3.1.2 Laravel Framework 15](#_Toc512488331)

[3.1.3 Integrated Development Environment: PhpStorm 15](#_Toc512488332)

[3.1.4 Database: MYSQL 15](#_Toc512488333)

[3.1.5 Composer 16](#_Toc512488334)

[IV. Technical Background 16](#_Toc512488335)

[4.1 Requirements Analysis 16](#_Toc512488336)

[4.2 Requirements Documentation 17](#_Toc512488337)

[4.3 Gap Analysis 23](#_Toc512488338)

[4.4 Design of Software, Systems, Product, and/or Processes 24](#_Toc512488339)

[4.4.1 Functional Decomposition Diagram 24](#_Toc512488340)

[4.4.2 Data Flow Diagram 25](#_Toc512488341)

[4.4.3 Entity Relationship Diagram 27](#_Toc512488342)

[4.4.4 Class Diagram 28](#_Toc512488343)

[4.4.5 Object Diagram 29](#_Toc512488344)

[4.4.6 Event Table 30](#_Toc512488345)

[4.4.7 Use Case Diagram 31](#_Toc512488346)

[4.4.8 Use Case Full Description 32](#_Toc512488347)

[4.4.9 Activity Diagram 38](#_Toc512488348)

[4.4.10 Sequence Diagram 45](#_Toc512488349)

[4.4.11 State Diagram 51](#_Toc512488350)

[4.4.12 Timing Diagram 54](#_Toc512488354)

[4.4.13 Package Diagram 57](#_Toc512488356)

[4.4.14 Communication Diagram 58](#_Toc512488357)

[4.4.15 Component Diagram 59](#_Toc512488358)

[4.4.16 Composite Structure Diagram 60](#_Toc512488359)

[4.4.17 Deployment Diagram 61](#_Toc512488360)

[4.4.18 Interaction Overview Diagram 62](#_Toc512488361)

[V. Bibliography 63](#_Toc512488362)

[VI. Appendices 64](#_Toc512488363)

[6.1 Survey on Renters 64](#_Toc512488364)

[6.2 Survey on Drivers 65](#_Toc512488365)

[6.3 Work Breakdown Structure 66](#_Toc512488366)

[6.4 Project Schedule 67](#_Toc512488367)

[6.5 Activity List 68](#_Toc512488368)

[6.6 Estimates 70](#_Toc512488369)

[6.7 Vision and Scope Document 71](#_Toc512488370)

[6.8 Software Requirements Specification 74](#_Toc512488371)

[6.9 Risk Management Plan 78](#_Toc512488372)

[6.10 Change Management Plan 82](#_Toc512488373)

[6.11 Quality Management Plan 85](#_Toc512488374)

[6.12 Curriculum Vitae 90](#_Toc512488375)

# List of Figures, List of Tables, List of Notations

[Figure 1: Source of rentable vehicles 17](file:///C:\Users\Martha\Desktop\carkila\CARKILA.docx#_Toc512487498)

[Figure 2: Difficulty of finding rentable vehicles 18](file:///C:\Users\Martha\Desktop\carkila\CARKILA.docx#_Toc512487499)

[Figure 3: Time it takes to find rentable vehicles 19](file:///C:\Users\Martha\Desktop\carkila\CARKILA.docx#_Toc512487500)

[Figure 4: Usefulness of the proposed project for renters 20](file:///C:\Users\Martha\Desktop\carkila\CARKILA.docx#_Toc512487501)

[Figure 5: Ways of advertising rentable vehicles 20](file:///C:\Users\Martha\Desktop\carkila\CARKILA.docx#_Toc512487502)

[Figure 6: Pricing reference in renting vehicles 21](file:///C:\Users\Martha\Desktop\carkila\CARKILA.docx#_Toc512487503)

[Figure 7: Usefulness of the proposed project for vehicle owners 22](file:///C:\Users\Martha\Desktop\carkila\CARKILA.docx#_Toc512487504)

[Figure 8: Functional decomposition diagram 24](file:///C:\Users\Martha\Desktop\carkila\CARKILA.docx#_Toc512487505)

[Figure 9: Context diagram 25](file:///C:\Users\Martha\Desktop\carkila\CARKILA.docx#_Toc512487506)

[Figure 10: Diagram 0 26](file:///C:\Users\Martha\Desktop\carkila\CARKILA.docx#_Toc512487507)

[Figure 11: Entity relationship diagram 27](file:///C:\Users\Martha\Desktop\carkila\CARKILA.docx#_Toc512487508)

[Figure 12: Class diagram 28](file:///C:\Users\Martha\Desktop\carkila\CARKILA.docx#_Toc512487509)

[Figure 13: Object diagram 29](file:///C:\Users\Martha\Desktop\carkila\CARKILA.docx#_Toc512487510)

[Figure 14: Use case diagram 31](file:///C:\Users\Martha\Desktop\carkila\CARKILA.docx#_Toc512487511)

[Figure 15: Create account activity diagram 38](file:///C:\Users\Martha\Desktop\carkila\CARKILA.docx#_Toc512487512)

[Figure 16: Rent vehicle activity diagram 39](file:///C:\Users\Martha\Desktop\carkila\CARKILA.docx#_Toc512487513)

[Figure 17: Manage listing activity diagram 40](file:///C:\Users\Martha\Desktop\carkila\CARKILA.docx#_Toc512487514)

[Figure 18: Approve request activity diagram 41](file:///C:\Users\Martha\Desktop\carkila\CARKILA.docx#_Toc512487515)

[Figure 19: View profile activity diagram 42](file:///C:\Users\Martha\Desktop\carkila\CARKILA.docx#_Toc512487516)

[Figure 20: View feedback activity diagram 43](file:///C:\Users\Martha\Desktop\carkila\CARKILA.docx#_Toc512487517)

[Figure 21: Generate reports activity diagram 44](file:///C:\Users\Martha\Desktop\carkila\CARKILA.docx#_Toc512487518)

[Figure 22: Create account sequence diagram 45](file:///C:\Users\Martha\Desktop\carkila\CARKILA.docx#_Toc512487519)

[Figure 23: Rent vehicle sequence diagram 46](file:///C:\Users\Martha\Desktop\carkila\CARKILA.docx#_Toc512487520)

[Figure 24: Manage listing sequence diagram 47](file:///C:\Users\Martha\Desktop\carkila\CARKILA.docx#_Toc512487521)

[Figure 25: Approve request sequence diagram 48](file:///C:\Users\Martha\Desktop\carkila\CARKILA.docx#_Toc512487522)

[Figure 26: View profile sequence diagram 49](file:///C:\Users\Martha\Desktop\carkila\CARKILA.docx#_Toc512487523)

[Figure 27: View feedback sequence diagram 49](file:///C:\Users\Martha\Desktop\carkila\CARKILA.docx#_Toc512487524)

[Figure 28: Generate reports sequence diagram 50](file:///C:\Users\Martha\Desktop\carkila\CARKILA.docx#_Toc512487525)

[Figure 29: State diagram of create account use case 51](file:///C:\Users\Martha\Desktop\carkila\CARKILA.docx#_Toc512487526)

[Figure 30: State diagram of rent vehicle use case 51](file:///C:\Users\Martha\Desktop\carkila\CARKILA.docx#_Toc512487527)

[Figure 31: State diagram of manage listing use case 52](file:///C:\Users\Martha\Desktop\carkila\CARKILA.docx#_Toc512487528)

[Figure 32: State diagram of approve request use case 52](file:///C:\Users\Martha\Desktop\carkila\CARKILA.docx#_Toc512487529)

[Figure 33: State diagram of view profile and feedback use case 53](#_Toc512487530)

[Figure 34: State diagram of generate reports 53](file:///C:\Users\Martha\Desktop\carkila\CARKILA.docx#_Toc512487531)

[Figure 35: Timing diagram of create account use case 54](file:///C:\Users\Martha\Desktop\carkila\CARKILA.docx#_Toc512487532)

[Figure 36: Timing diagram of rent vehicle use case 54](file:///C:\Users\Martha\Desktop\carkila\CARKILA.docx#_Toc512487533)

[Figure 37: Timing diagram of manage listing use case 55](file:///C:\Users\Martha\Desktop\carkila\CARKILA.docx#_Toc512487534)

[Figure 38: Timing diagram of approve request use case 55](file:///C:\Users\Martha\Desktop\carkila\CARKILA.docx#_Toc512487535)

[Figure 39: Timing diagram of view profile and feedback use case 56](file:///C:\Users\Martha\Desktop\carkila\CARKILA.docx#_Toc512487536)

[Figure 40: Timing diagram of generate reports use case 56](file:///C:\Users\Martha\Desktop\carkila\CARKILA.docx#_Toc512487537)

[Figure 41: Package diagram 57](file:///C:\Users\Martha\Desktop\carkila\CARKILA.docx#_Toc512487538)

[Figure 42: Communication diagram 58](file:///C:\Users\Martha\Desktop\carkila\CARKILA.docx#_Toc512487539)

[Figure 43: Component diagram 59](file:///C:\Users\Martha\Desktop\carkila\CARKILA.docx#_Toc512487540)

[Figure 44: Composite structure diagram 60](file:///C:\Users\Martha\Desktop\carkila\CARKILA.docx#_Toc512487541)

[Figure 45: Deployment diagram 61](file:///C:\Users\Martha\Desktop\carkila\CARKILA.docx#_Toc512487542)

[Figure 46: Interaction overview diagram 62](file:///C:\Users\Martha\Desktop\carkila\CARKILA.docx#_Toc512487543)

[Table 1: Carkila vs. TNVS apps 9](#_Toc512487544)

[Table 2: Carkila vs. Existing Car Rental Systems 9](#_Toc512487545)

[Table 3: Gap Analysis 23](#_Toc512487546)

[Table 4: Event Table 30](#_Toc512487547)

[Table 5: Create Account Use Case Description 32](#_Toc512487548)

[Table 6: Rent Vehicle Use Case Description 33](#_Toc512487549)

[Table 7: Manage Listing Use Case Description 34](#_Toc512487550)

[Table 8: Approve Request Use Case Description 35](#_Toc512487551)

[Table 9: View Profile Use Case Description 36](#_Toc512487552)

[Table 10: Write Feedback Use Case Description 37](#_Toc512487553)

# Introduction

## Project Context

Currently, social media, word of mouth, flyers and/or posters are the common means of how DIY (Do-It-Yourself) travelers, small production teams and families acquire vehicle rental services. This principle also applies to people who want their vehicles to be rented, all of this can be said in the grey economy. Finding rentable vehicles takes a lot of time according to the result of the survey conducted by the proponents. In order to make it easier for people to find rentable vehicles, the proponents proposed to develop an Android-based mobile application called Carkila that will provide a venue for users –vehicle owners and renters – to interact with each other.

This mobile application will allow users –vehicle owners and renters – to either list a vehicle for rent or rent one. This application will allow renters to look for a rentable vehicle according to their preferences like location, price range, seating capacity, etc. There will also be a geolocation feature that will help them find rentable vehicles near their location. Another feature will allow renters to make an offer or negotiate the renting price of the vehicle. There will also be a private chat feature where they can further discuss the important details of their transaction.

Thus, if everything goes well, this mobile application will alleviate the concerns of the target users that were found on the survey conducted.

## Purpose and Description

People look for rentable vehicles manually. A survey was conducted by the group with 50 respondents and the information that the group gathered from the survey shows that there is a limitation as to where people can find vehicles for rent; the majority being from referrals from other parties or by searching through social media. Also based on the data gathered, 44% of the respondents described that finding rentable vehicles was quite on the difficult side.

Another question on the survey asked the respondents if having a mobile app would be useful for finding private rentable vehicles and 98% of the respondents answered yes. Now with the data that the survey provided, the proponents have deduced that a mobile app would indeed be useful for people who need to find rentable vehicles.

Moreover, the group also conducted a survey for the drivers. Based the results of the survey, there is also a limitation as to where drivers can advertise their vehicles to get clients. Majority of the respondents (64 out of 80) answered that if they were to rent out their vehicle, it would be through social media. 70 out of 80 respondents think that an app would be useful for advertising their vehicles and getting more passengers. With the data, the proponents deduced that a mobile application would be helpful for both vehicle owners and renters.

Carkila is an Android-based mobile application that connects people who need to find rentable vehicles and people who want their cars to be rented. It includes a chat feature wherein both parties may offer a reasonable price. The owner can choose to accept the offer or decline it. This mobile application would also ensure that the transaction is secure for both end-users as both parties will have each other’s personal information.

There are already existing vehicle rental websites and mobile applications such as Hertz, Viking Vehicle Rentals, and Manila Rent-A-Vehicle. All these systems offer the same type of services; they have options like self-drive and chauffeured drive. The difference of these systems from the proposed project is that, they are all owned by a company; meaning they own fleet vehicles, unlike in Carkila where the vehicles are owned by local and private vehicle owners. Table 1 shows the difference between TNVS (Transport Network Vehicle Services) apps and the proposed project. Table 2 on the other hand, shows the difference between existing car rental systems and Carkila.

Table 1: Carkila vs. TNVS apps

|  |  |  |  |
| --- | --- | --- | --- |
| **Features** | **Grab** | **Uber** | **Carkila** |
| Out-of-town transport | ✕ | ✕ | ✓ |
| Long-term rental | ✕ | ✕ | ✓ |

Table 2: Carkila vs. Existing Car Rental Systems

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Features** | **Viking** | **Hertz** | **Manila Rent-a Car** | **Avis** | **Carkila** |
| Mobile-based | ✕ | ✓ | ✕ | ✓ | ✓ |
| Peer-to-Peer | ✕ | ✕ | ✕ | ✕ | ✓ |
| Offers chauffeured drive | ✓ | ✓ | ✓ | ✓ | ✓ |
| Shows driver information | ✕ | ✕ | ✕ | ✕ | ✓ |
| Locates nearest vehicle available | ✕ | ✕ | ✕ | ✕ | ✓ |

## Objectives

* To develop a peer-to-peer mobile application for car rental services
* To make give vehicle owners a new source of income by allowing them to post their vehicles
* To make it easier for users to find rentable vehicles by providing them a list
* To allow users to find retable vehicles near them

## Scope and Limitation

The project is a vehicle rental mobile application that will run in Android. The scope of the mobile application is from the registration up until a transaction has been made between end-users.

The mobile application would need to internet connection for it to be updated and stored to the server. The app will record transactions done by the vehicle owner and renter. The app will not be involved if both parties agree to a “self-drive” transaction; users must be aware of the risk of renting a vehicle. Therefore, they must comply with what is stated in the terms and conditions that will be provided, since the main goal of the app is to provide an environment for vehicle owners and renters who need to find rentable vehicles in order to make it easier to find rentable vehicles. The payment method will be cash only as the system does not handle the payment and invoice.

# Review of Related Literature and Systems

## Traditional Car Rental Systems

### Hertz Car Rental

Hertz car rental system provides customers a list of available cars that can be rented. These cars are from the company itself and not from other customers. What makes Hertz Car Rental System effective is also their own rental qualifications and requirements that should be taken into consideration before anything else to make more secured and efficient transactions. Here are some of their important qualifications and requirements:

* Filters – When reserving a car for rent, users are prompted to select/input filters such as pick up locations, pick up date and time, return date and time, age, car type, and then asks if a user is a guest or a member.
* Driver’s License – In order for a customer to rent a car, one should have a driver’s license and is valid for the duration of the desired rental period. Moreover, if the customer’s driver’s license is registered in a foreign country, but wants to rent a car for example, in the United States, the customer may need to issue an International Driver’s Permit to be qualified in processing the rental request.

:

* Insurances – Accidents may happen during the renting period. Therefore, different sets of insurances are also viable. For example, Personal Accident Insurance (PAI) can be applied when an accidental death and accidental medical expense happens. For an accidental death, the immediate family receives $175,000, while each passenger receives $17,500. For an accidental medical expense, the renter receives $2,500 and each passenger receives $2500. There are also a lot of additional insurances that can be applied depending on the country and the customer affected as well.
* Age restrictions and exceptions – the minimum age for a customer to be a candidate for the services of Hertz is 18. Of course the driver’s license requirement is also applied. However, there are also some things to be noted such as the type of car to be rented by the customer. For the Adrenaline, Dream and Prestige Collections, the minimum age is 25. All other cars can be rented for age lower than 25. Also, corporate accounts also state that young renters can also be allowed to rent the 3 collections for age 25. However, agreements should be made between the two parties.

Now, taking all these sample rental qualifications and requirements, the proponents can adapt these things in making policies for the application. For example, filters. Filters are recommended, so that the system knows what to process from the vast amount of data in itself. Filters help so it will be easier for the system and for the customers to find their desired car. The most important thing of all requirements is the driver’s license. In developing policies in the app, driver’s license should be the most significant requirement for a user to be registered in the system.

Complexities such as International Driver’s Permit should also be a requirement if a renter is foreign. This will ensure that the user is capable of driving even in foreign locations. In making features and policies in the system, insurances are a must also. And because it deals with cars and trips, insurances should be accounted by the system.

Lastly, age restrictions should also be studied very well. Not all young renters can drive all types of vehicles. There should be some type of vehicles that should be restricted for some certain reasons just like in Hertz Car Rental System (Hertz, n.d.).

### Viking Car Rental

Viking Car Rental is another existing company that lends cars to the public. It from different kind of cars up to vans, and even buses. Viking Car Rental System also offers some of the same features of Hertz Car Rental System. However, Viking offers some features that Hertz do not, such as rental on buses and vans, tips and tricks on renting, detailed car specifications, and many more. Viking Car Rental System directly stated options that users can pick according to their own preferences:

* Self-Drive – Viking gives the customers an option of self-drive; meaning they rent and drive the rented car. They will also be the ones to submit the car in their location after usage.
* Chauffeured Drive – The customer rents the vehicle, but is accompanied by a driver. This option is more expensive than self-drive. It’ll be safe for the customers because the drivers are trained and licensed by the company.
* Leasing/Long-Term rental – Viking recommends this option for users who still cannot decide whether to buy a certain vehicle or not. This can help them decide. Also, corporate use belongs here.
* Special Events: Conventions, Summit, Trade Shows etc. – Users can avail bigger discount or special rates if a large number of vehicles are rented for special occasions.
* One-way Rental Pick Up or Drop Off – shares the same method with taxis, Uber, and Grab
* Promos – Viking Car Rental System also offers marketing promotions which the users can avail.
* Detailed Specifications of Available Cars – All available cars are rendered in a user-friendly interface, where users can see the detailed description of vehicles such as number of doors, number of persons that can fit in the car, number of luggage that can fit in the car, and if the car is air conditioned or not. A picture of the car is also included (Viking Rent-A-Car, n.d.).

## Similar Systems in the Philippines

In order to confirm the need for a mobile app, the proponents researched about peer-to-peer car rental in the Philippines and they have only found one similar system called Arkila.ph. According to the owner, Arkila.ph removes the inconvenience of searching the internet and rentable vehicles and then contacting the drivers if they are available. With this information, the researchers concluded that the proposed project would indeed be useful.

In this website, a user has to post the trip details in the website and then the system will automatically send an SMS notification to all drivers that are registered in the system. After posting, a bidding process will come next, wherein the drivers will post the rates of their service. The user can choose among the bids posted by the drivers by clicking on the Book button (Arkila, n.d.).

## Peer-to-Peer Rentals

According to investopedia.com, Peer-to-Peer or P2P service is a decentralized platform where two individuals interact directly with each other, without the intermediation by a third-party, or without the use of a company of business selling a product or service. The past years, peer-to-peer e-commerce have been emerging. According to a study, peer-to-peer marketplaces have now expanded to provide short-term rental of products. There are different categories of peer-to-peer rental marketplaces. There are such marketplaces that offer car rentals, home/apartment rentals, and clothing rentals. These marketplaces differ from the usual e-commerce apps that have B2C or Business-to-Consumer models. In peer-to-peer rentals, the transaction is between two individuals rather than between a firm/company and an individual (Fraiberger & Sundararajan, 2016). Examples of these includes Airbnb, an online marketplace where one (guests) can rent hotel rooms, apartments, etc. from hosts (owners). Another example or P2P rental is *StyleLend*, which allows users to rent clothing items from other people.

According to an article written by Lydia Dishman, the founder of Rentalic.com – a website where people can rent to/from each other – said that “the whole idea is to build a community-based rental market place where individuals or businesses in a community can share goods and services by renting to/from each other”. Based on this article, there are a few number of growing peer-to-peer rental sites such as *Rentalic, Zilok,* and *Rent-Instead.*

## Peer-to-Peer Car Sharing

According to a study conducted by Ingrid Ballus-Armet, Susan Shaheen, Kelly Clonts, and David Weinzimmer (2014), peer-to-peer car sharing is a new way of car rentals wherein vehicle owners temporarily rent their personal vehicle to other people. It is considered as a part of the “sharing economy”. Sharing economy is a model wherein people shares assets with each other instead of owning them. P2P car sharing consists of privately owned vehicles that are made temporarily available for rent in a short period of time with pickup and drop-off locations agreed upon by the two individuals. The owners of the vehicle gets profit from the transaction with customers/renters

.

It was also mentioned in the study that trust is an issue when it comes to car sharing. However, it was stated that some platforms try to limit car sharing to a closed environment and others rely solely on user ratings and feedback systems and by integrating social media into the platform for credibility. Operator screening and selection was also considered as a mechanism that could ensure trust among users.

Example of P2P car sharing is *Car Next Door*, a P2P car sharing app that is based in Australia. It is a mobile app where car owners can list their cars and where customers can find rentable vehicles. Another example of P2P car sharing is *Hubber,* a mobile app that matches travelers who need cars by matching them with cars that are rented out by car owners when they go out of town (Peltier, 2015). Other examples of peer-to-peer car rentals include *Turo* (previously known as *RelayRides*), *Getaround,* and *Drivy* (Fraiberger & Sundararajan, 2016).

# Technical Background

## Development Tools

### PHP

PHP or *PHP: Hypertext Preprocessor* is a scripting language used to develop interactive web pages. It can be used with HTML codes, frameworks, content management systems, etc.

### Laravel Framework

Laravel is an open-source framework used to develop web applications using PHP. It uses the MVC or Model-View-Controller pattern to speed up the development of websites. However, it can also be used to create hybrid mobile applications; meaning, a Laravel project can run as a mobile application.

### Integrated Development Environment: PhpStorm

PhpStorm is an IDE (Integrated Development Environment) for PHP built on IntelliJ platform by Jetbrains. It includes Laravel plugins – the framework used for the development. A control version is also integrated with the IDE, which makes it easier for changes to be tracked.

### Database: MYSQL

The proponent plan to use MYSQL as the main database for the project. It is a relational database system that can be used for free. It uses SQL (Structured Query Language) and it can be used with PHP.

### Composer

Composer is a dependency manager that is used with PHP. It is used to install dependencies or packages in Laravel.

# Technical Background

## Requirements Analysis

Vehicle rental or “arkila” is known to be one of the common means of transportation especially for out-of-town or vacation trips here in the Philippines. The proponents used Facebook and observed that many people are posting about car rentals, specifically about where to find one or who knows anyone who offers car rentals. With this observation, the proponents thought that it is a problem as there is no mobile application for it yet.

The proponents researched about vehicle rentals here in the Philippines and they have found an existing website called Arkila.ph, which addresses the same issue that the proponents would like to target. The proponents considered this as a proof that there is indeed a problem regarding vehicle rentals.

In order to confirm the need for the mobile application, the proponents conducted an online survey that consisted of (4) four questions. The survey (See Appendix A) was posted in travel groups on social media and it was also answered by those who have experience in renting a vehicle. Moreover, the proponents also conducted surveys to ask for the opinions of drivers. The survey (See Appendix B) was also posted in car rental groups on social media and it was also given out to drivers as anyone who owns a car can be a potential user of the mobile application.

## Requirements Documentation

The following data support the proponents’ claim about the problems that the proposed application might mitigate. The respondents usually look for rentable vehicles through social media or through other people. Therefore, the mobile application will serve as a platform that will connect renters to vehicle owners who rents out their cars. According to the respondents, finding rentable vehicles can take up to more than an hour. This will be resolved by the mobile application for it will provide the users a list or choices of rentable vehicles. The proponents were also able to find a system/website that addresses the same problem that the group claims. Furthermore, the proponents also conducted a survey for potential car owners who may want to rent out their car in the future. This is to determine whether an app could also be useful for the drivers and also to determine whether to opt for a variable pricing or put bidding system feature in the app. Based on the results, the majority of the respondents are open to negotiation of prices and therefore, a negotiation feature could be implemented in the mobile application.

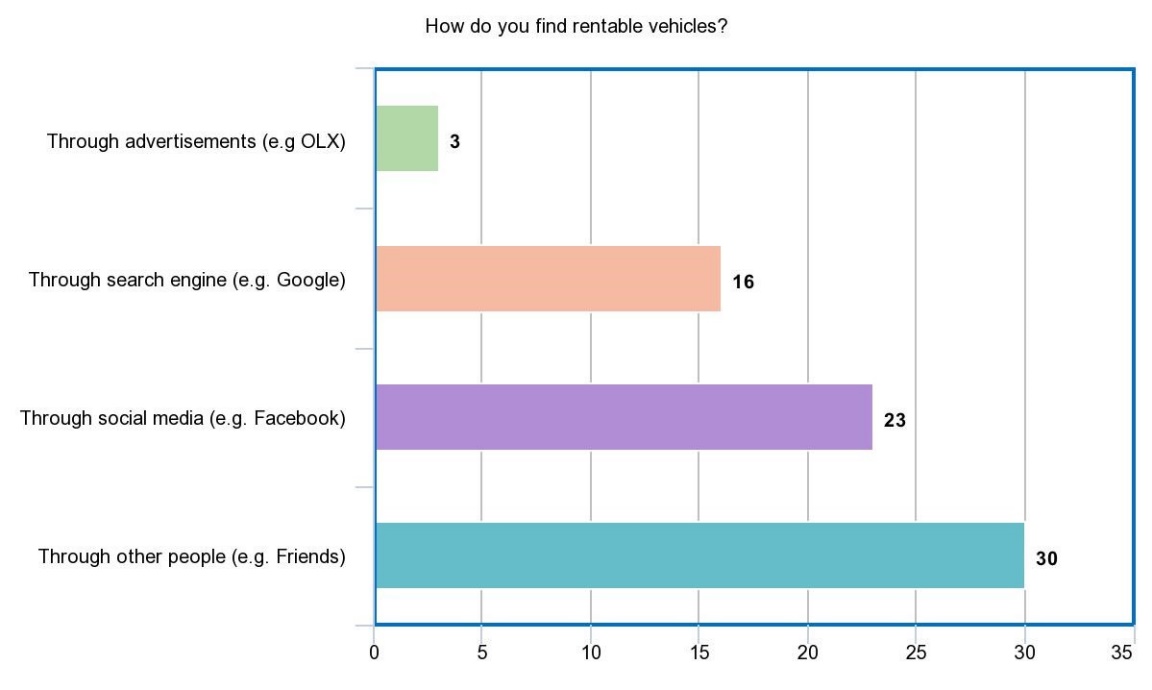


Figure 1: Source of rentable vehicles

Figure 1 shows that most of the respondents (30 out 50) search for rentable vehicles through other people. With this data, the proponents can foresee that not one of them had used a mobile application that is specifically for renting vehicles.

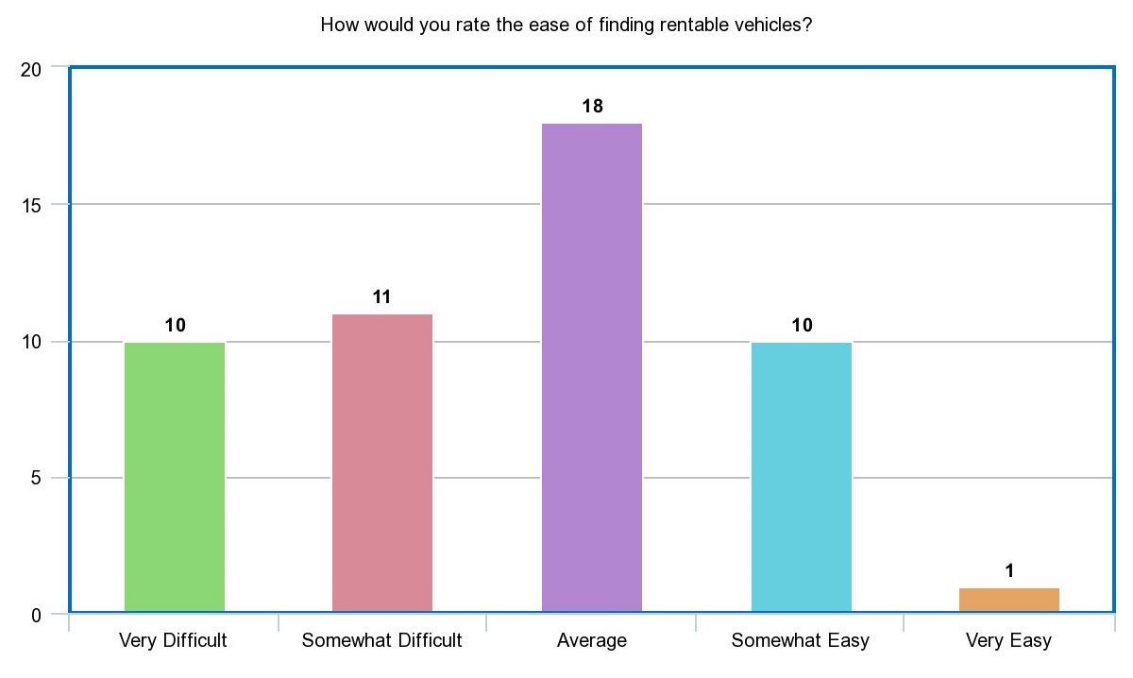


Figure 2: Difficulty of finding rentable vehicles

The proponents want to create a vehicle rental mobile application to make it easier for the people to look for rentable vehicles. Figure 2 shows that 36% (18 out 50) of the respondents rate the ease of finding rentable vehicles as average. Although it got the highest number of respondents, the majority of the result (21 out of 50) was quite on the difficult side. This data confirmed that there is indeed difficulty when finding rentable vehicles.

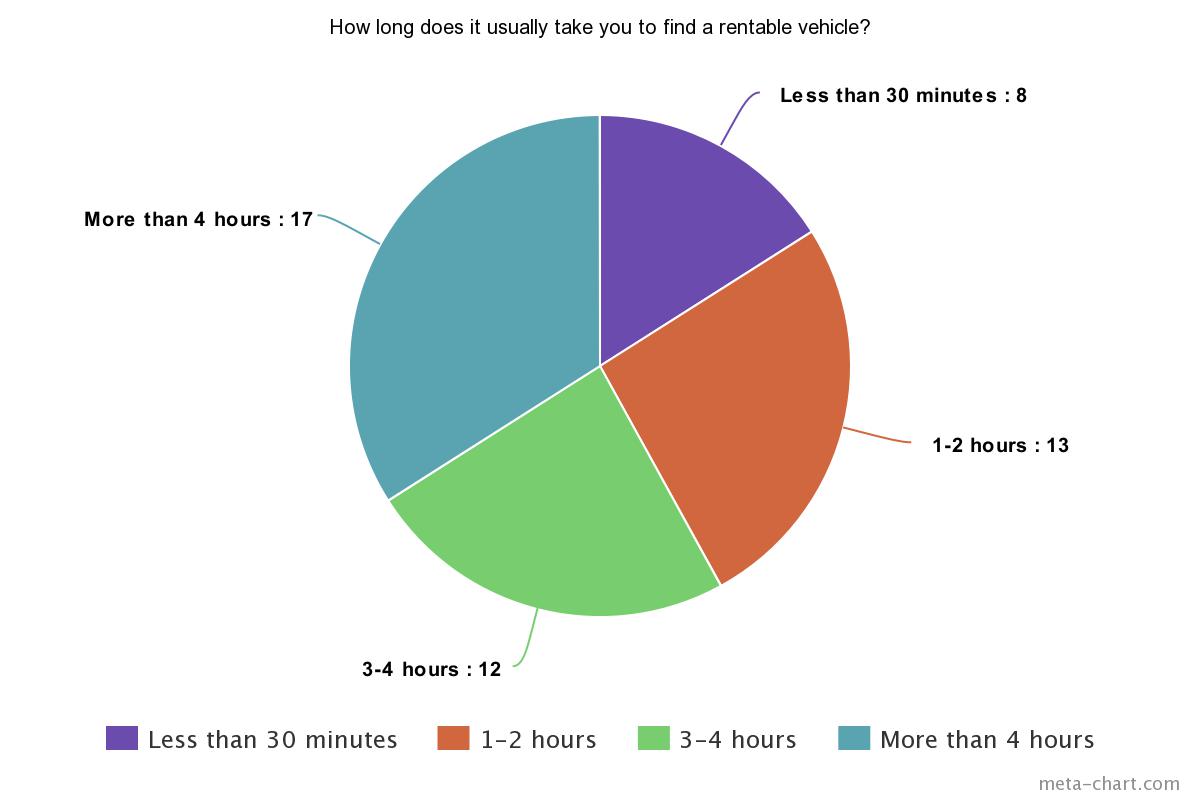


Figure 3: Time it takes to find rentable vehicles

Figure 3 shows the usual time it takes when finding rentable vehicles. Based on the results, 84% (42 out of 50) of the respondents took couple of hours to find rentable vehicles, and only 16% of them claim to have found a rentable vehicle in less than 30 minutes. As shown above, 34% (17 out of 50) respondents claim that it took them more than 4 hours to find a rentable vehicle. This is another issue that the proponents want to mitigate. The proponents would like to make it faster and easier for people to find rentable vehicles with the help of a mobile application.

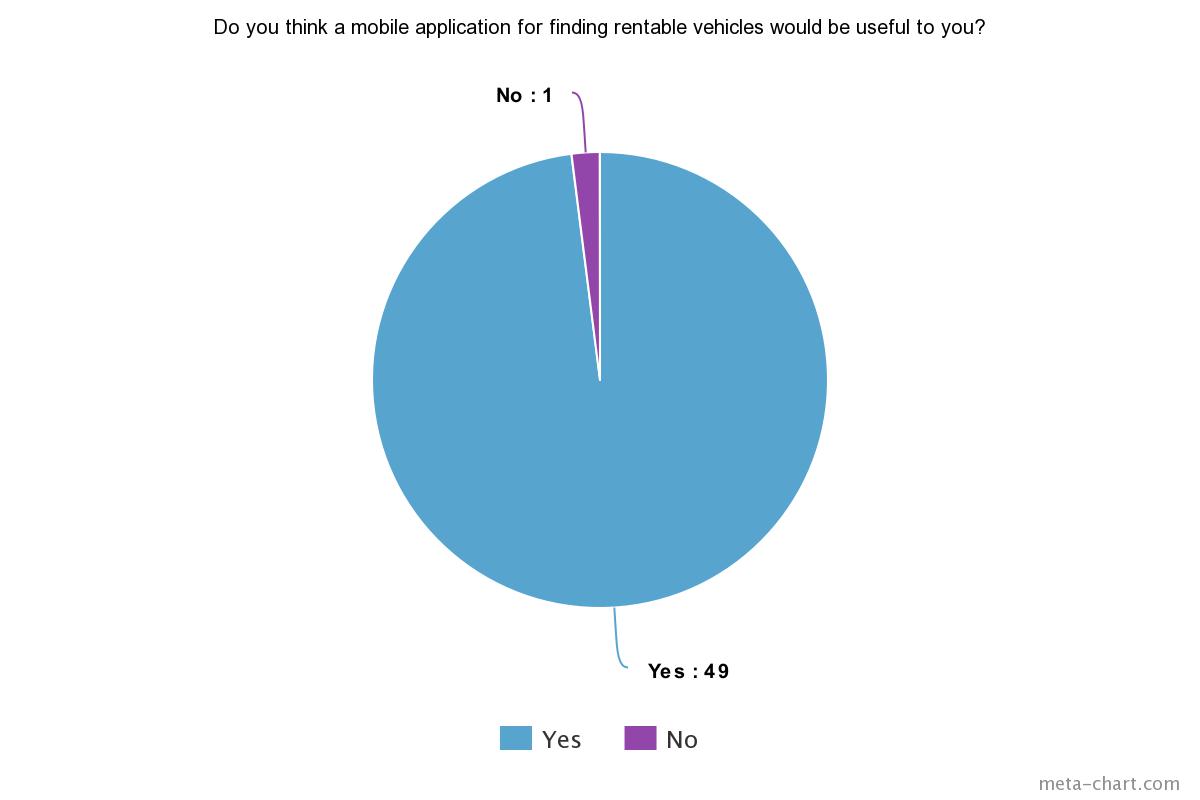


Figure 4: Usefulness of the proposed project for renters

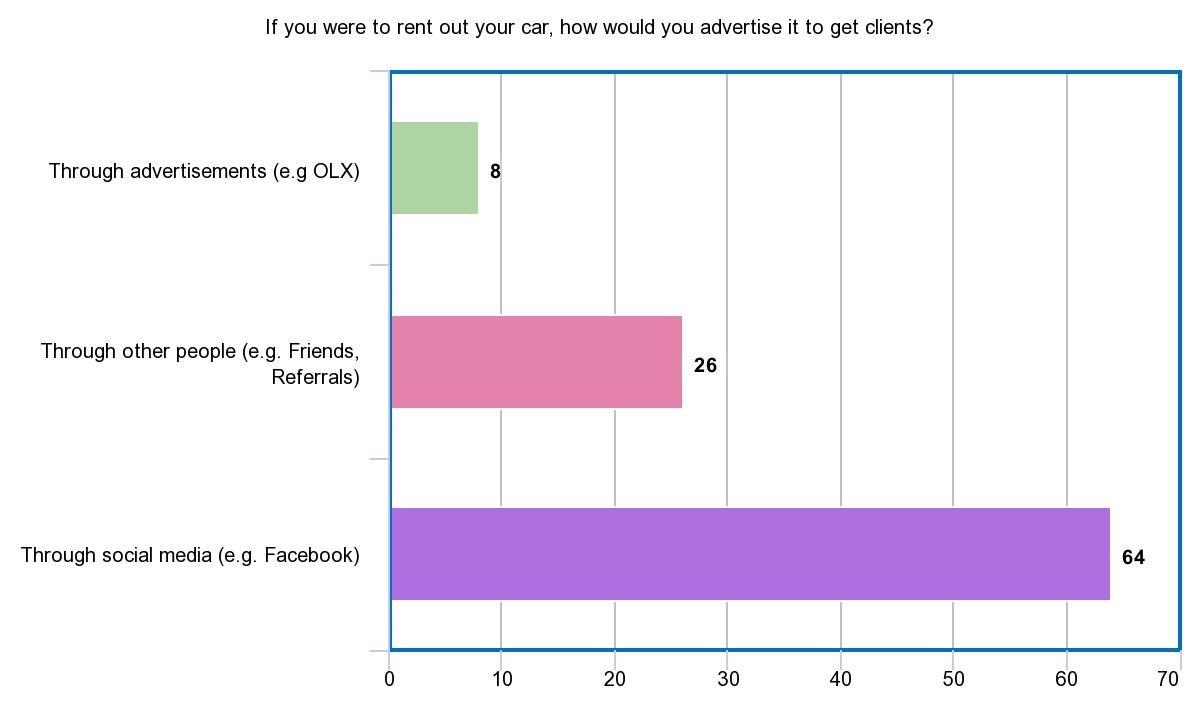
The proponents asked if a mobile application would be useful. Based on the results, 98% think it would indeed be useful. Therefore, with this data, the proponents can foresee that a mobile application would indeed aid the problem.

Figure 5: Ways of advertising rentable vehicles

The drivers were asked how they would advertise their cars if ever they were to rent it out. Figure 5 shows that 80% (64 out of 80) of the respondents consider using social media to advertise their cars. 32.5% (26 out of 80) the respondents considered advertising through other people. Meanwhile, 10% (8 out of 80) considered using advertisements such as OLX, flyers, etc.

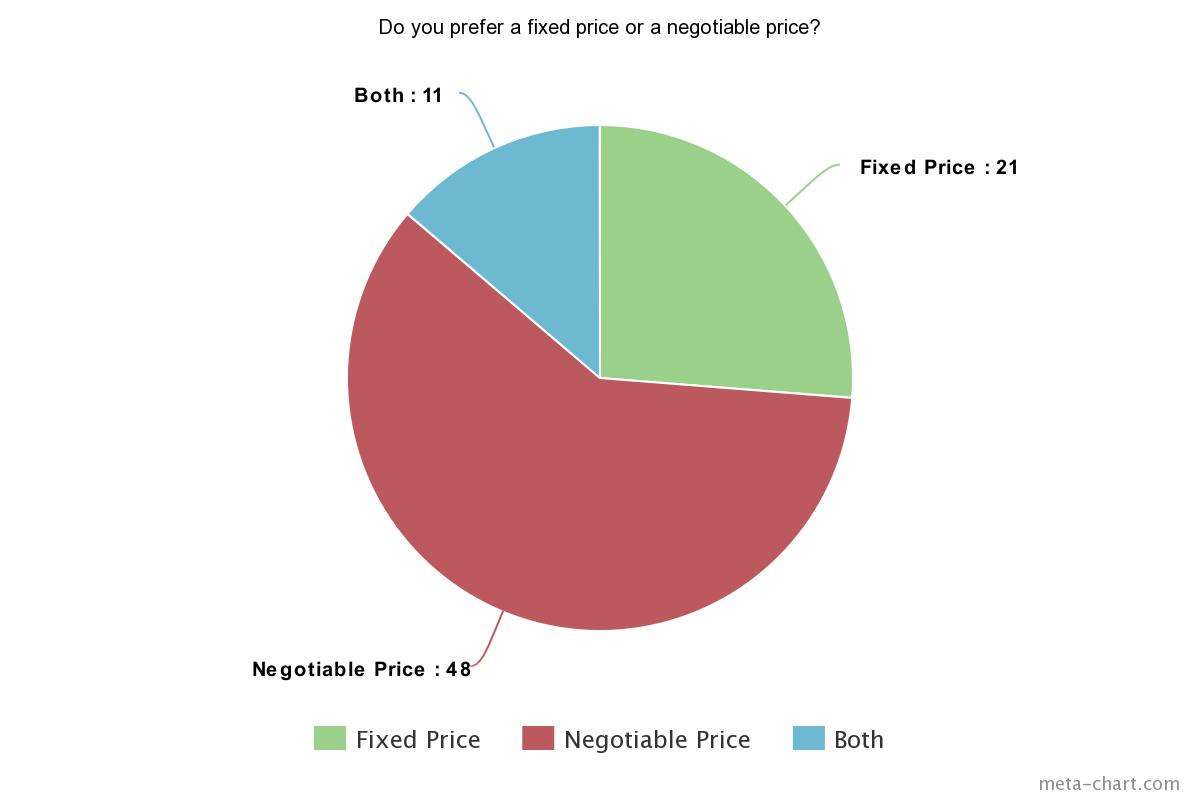


Figure 6: Pricing reference in renting vehicles

The proponents would like to determine whether car owners would agree with a bidding system wherein car renters could make offers or negotiate the renting price of a vehicle. Therefore, the proponents asked whether they prefer a fixed price or a negotiable price. Figure 6 shows that 60% (48 out of 80) of the respondents prefer a negotiable price, while 26.3% (21 out of 80) of them prefer a fixed price. Meanwhile, 13.7% (11 out of 80) of the respondents are okay with both. Based on the data gathered, the proponents concluded that majority of the respondents are open to negotiation of prices. Thus, it would be ideal to apply a feature wherein the users can negotiate the renting price of a vehicle.

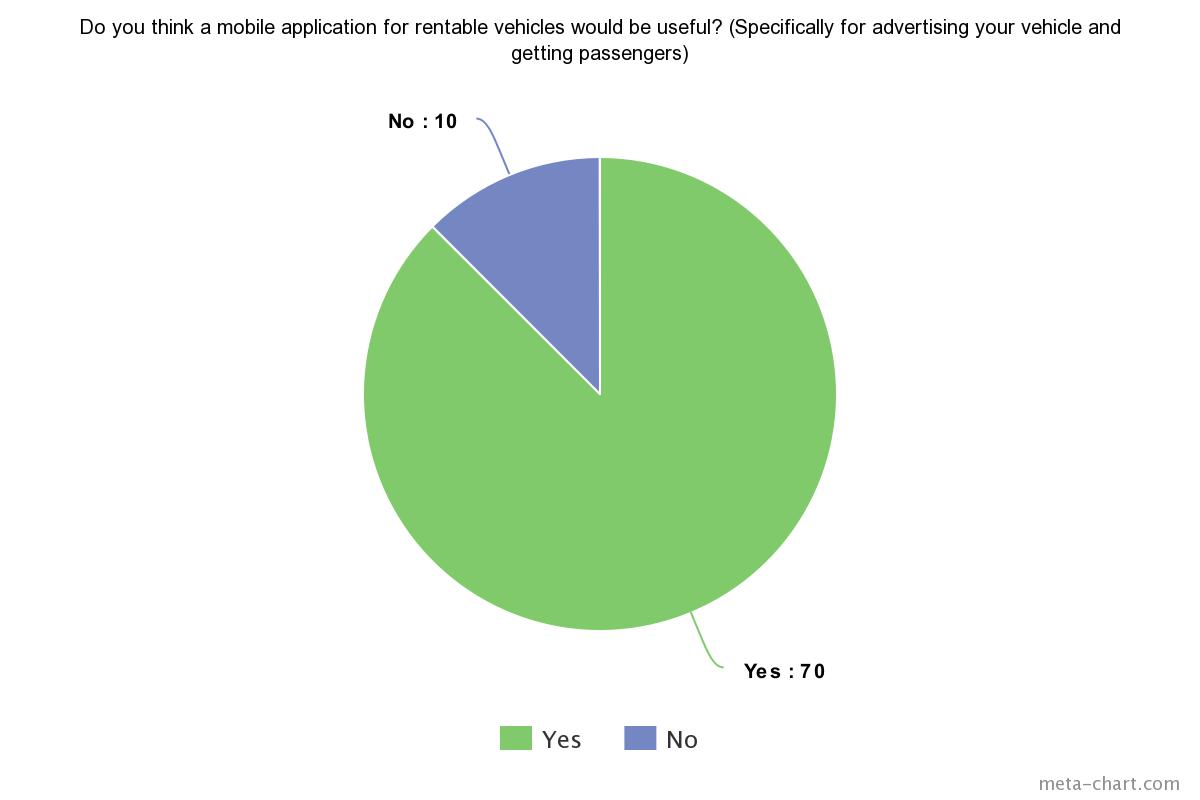


Figure 7: Usefulness of the proposed project for vehicle owners

The proponents also asked if a mobile application could also be useful for advertising their vehicles 87.5% (70 out of 80) respondents answered yes while 12.5% (10 out of 80) answered no. Therefore, the proponents concluded that a mobile would be useful for the vehicle owners.

## Gap Analysis

Table 3: Gap Analysis

|  |  |  |
| --- | --- | --- |
| User Requirements | Current Systems | Proposed Changes |
| To be able to immediately find out the rental rate of a vehicle | Existing systems such as Viking still use the manual process where one still has to contact the company for the rental rates. | Display the rental rate of a vehicle in the mobile application |
| To be able to have an information about the driver of rented vehicle | Existing rental systems do not give out information about their drivers. | Add a feature that shows the needed information of the driver for security purposes. |
| To be able to find the nearest available vehicle to the user | There is a limitation of branches. Rental companies such as Hertz only has 7 available pick up locations which are mostly airports. | Add a feature that will allow people to find the nearest available vehicle and rent whenever and wherever they are. |

## Design of Software, Systems, Product, and/or Processes

### Functional Decomposition Diagram

Figure 8 shows the broken-down process and functions of the proposed system.

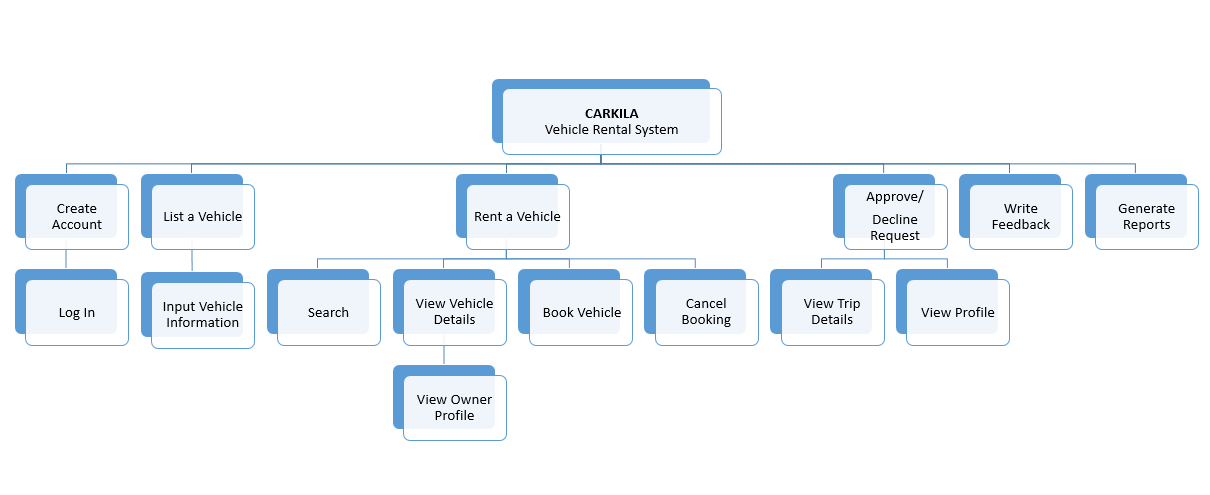


Figure 8: Functional decomposition diagram

### Data Flow Diagram

Figure 9 and 10 show the graphical representation of the proposed project without going over in detail of how the system works, but instead, it focuses on how all the data/information will move through the system.

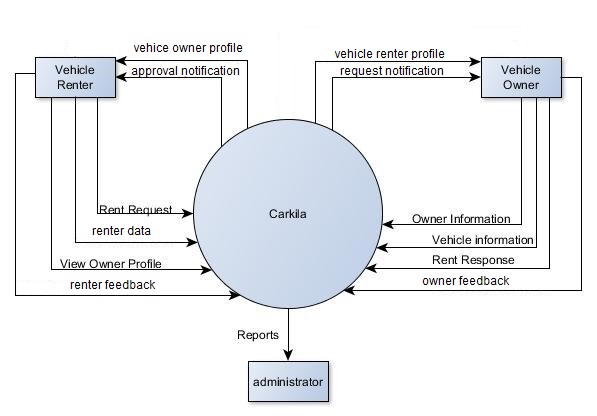


Figure 9: Context diagram

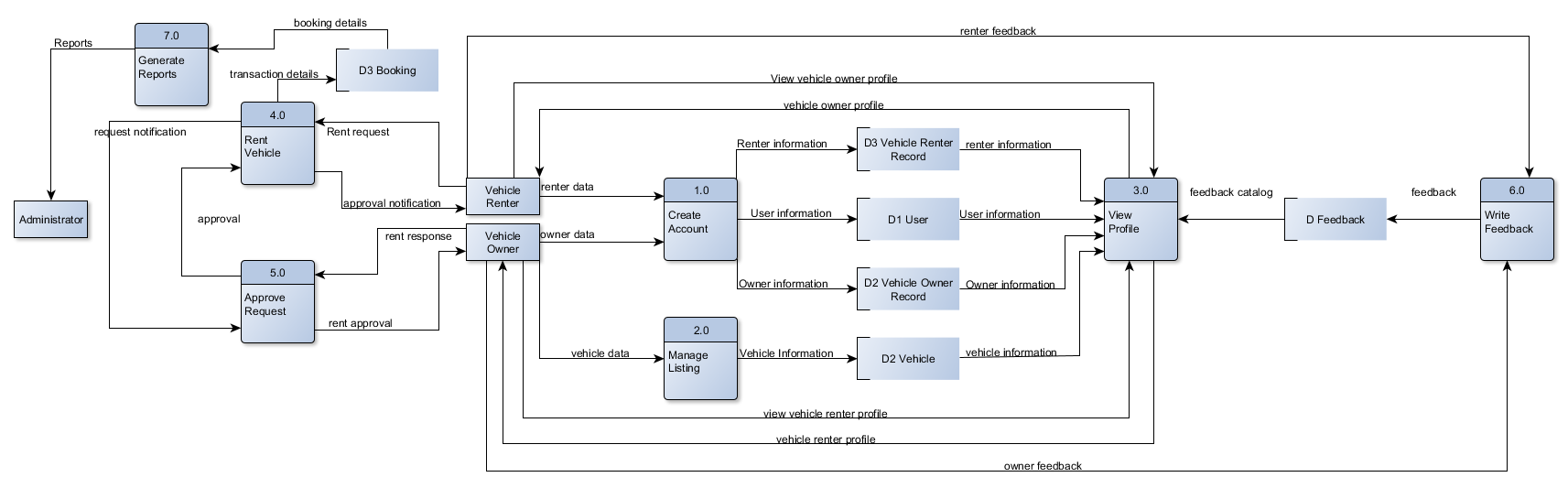


Figure 10: Diagram 0

### Entity Relationship Diagram

Figure 11 shows the proposed database design of the system. It shows the relationship between entities and where all the incoming data will be stored.

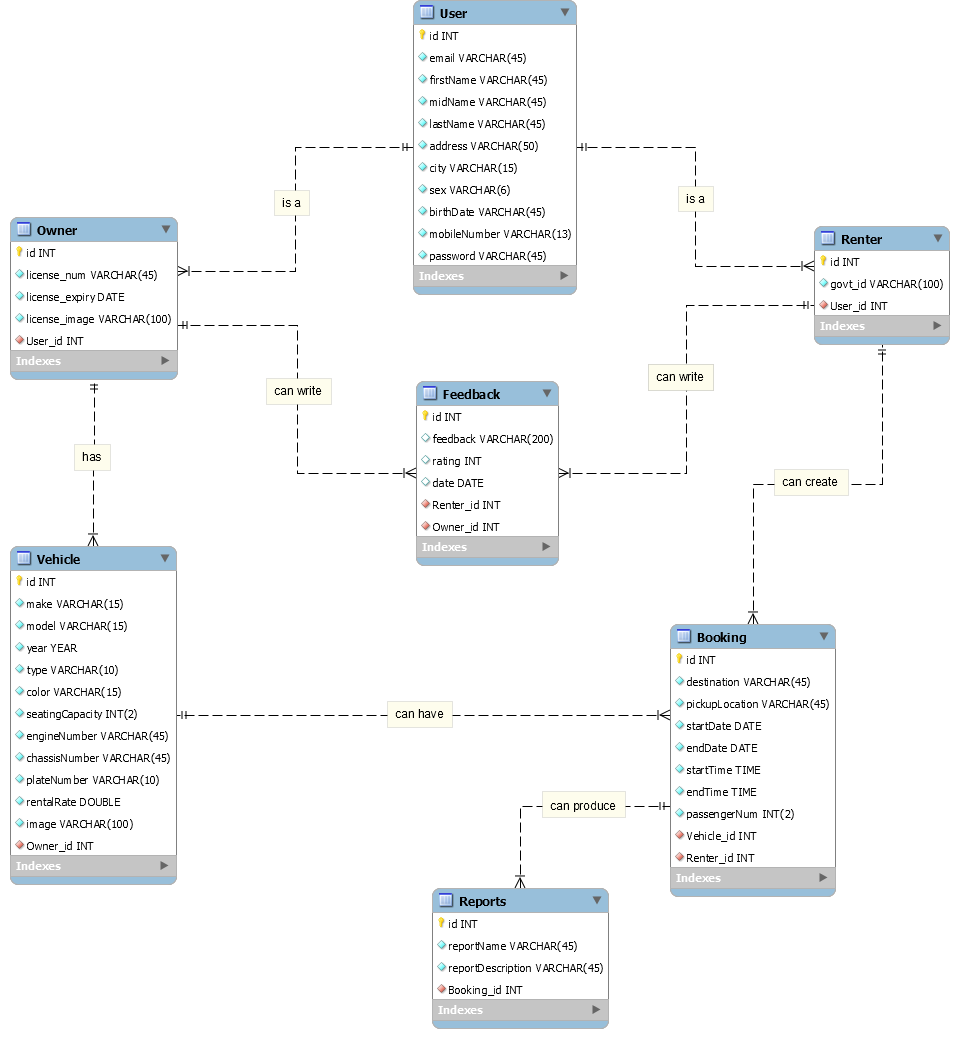


Figure 11: Entity relationship diagram

### Class Diagram

Figure 12 shows the classes, their attributes, methods, and their relationship with one another which will later be used in developing the system.

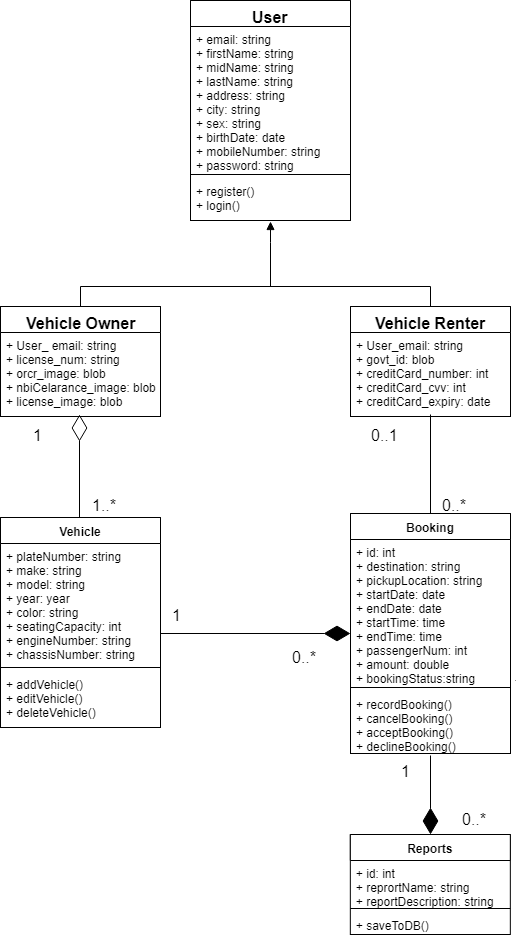


Figure 12: Class diagram

### Object Diagram

Figure 13 shows the relationship between objects. This diagram will also help in understanding the relationship between class instances.

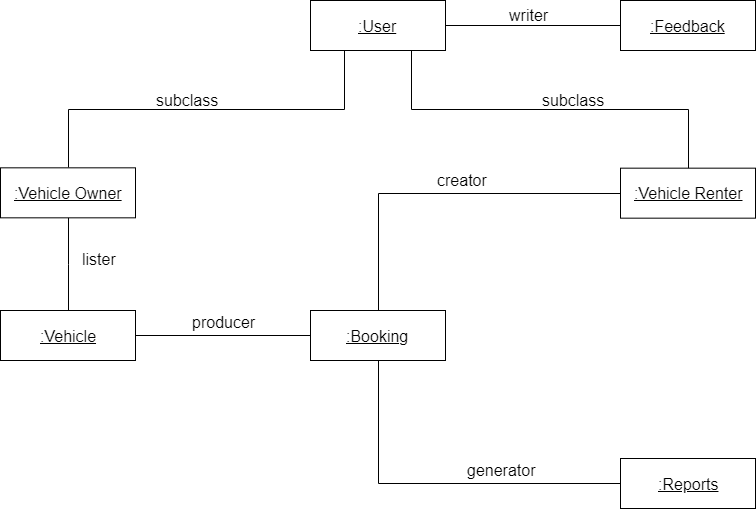


Figure 13: Object diagram

### Event Table

Table 6 shows the events where the system must respond and how it is supposed to respond. This table will also help in identifying and creating use cases for the proposed project.

Table 4: Event Table

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Event** | **Trigger** | **Source** | **Use Case** | **Response** | **Destination** |
| 1. Vehicle renter wants to create an account. | New account request | Vehicle renter | Create Account | New Account/  Account details | System  Vehicle renter |
| 1. Vehicle renter wants to rent a vehicle. | Request a vehicle | Vehicle renter | Rent Vehicle | Trip details | Vehicle owner |
| 1. Vehicle owner wants to create, view, update, or delete listing. | Request to manage vehicle listing | Vehicle owner | Manage Listing | Vehicle listing  details | System Vehicle owner |
| 1. Vehicle owner wants to accept a rent request. | Rent request notification | Vehicle owner | Approve Request | Approval | Vehicle renter |
| 1. Vehicle renter wants to view profile | Profile request | Vehicle renter | View Profile | Vehicle Owner Profile | Vehicle renter |
| 1. Vehicle renter wants to give feedback. | Request to give user feedback | Vehicle renter | Give Feedback | Feedback form | Vehicle renter |
| 1. Admin requests the system to generate reports. | Report request | Admin | Generate Reports | Report details | System  Admin |

### Use Case Diagram

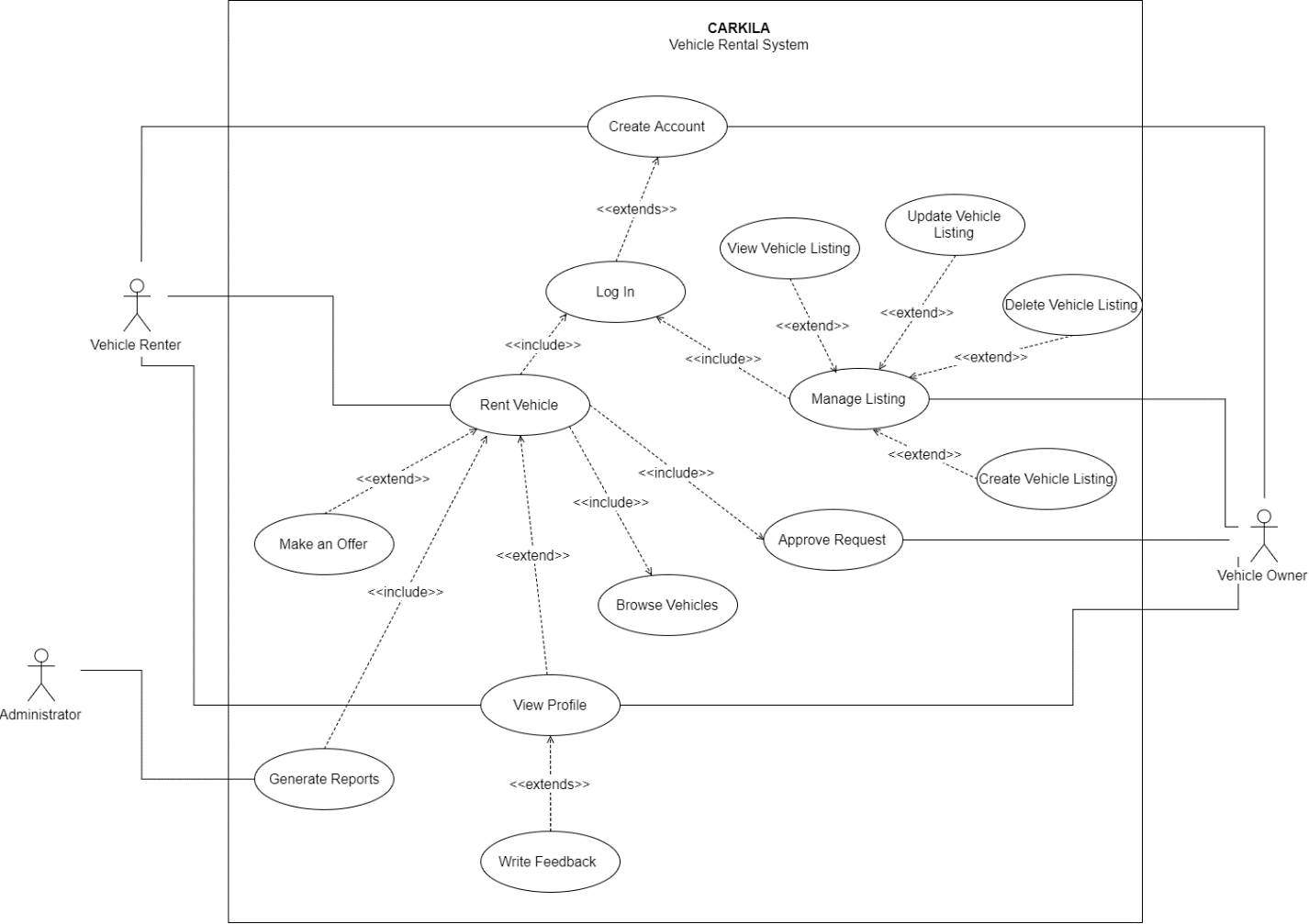
Figure 14 shows how the users can interact with the system. This diagram also contains the step-by-step activities that occur in the system.

Figure 14: Use case diagram

### Use Case Full Description

The following tables shows the description of each use cases.

Table 5: Create Account Use Case Description

|  |  |  |
| --- | --- | --- |
| **Use Case Name:** | Create Account | |
| **Scenario:** | User wants to create an account. | |
| **Triggering Event:** | New account request | |
| **Brief Description:** | This will allow the user to create an account. | |
| **Actors:** | Vehicle owner, vehicle renter | |
| **Related Use Case:** | Rent Vehicle, Manage Listing | |
| **Stakeholders:** | Vehicle owner, vehicle renter | |
| **Preconditions:** | The user must have the mobile application installed. | |
| **Post conditions:** | The user must be able to access the app features. | |
| **Flow of Activities:** | **Actor** | **System** |
| 1. The user opens the mobile application. 2. The user taps **Create Account**. 3. The user enters the required information. 4. The user enters the verification code. | * 1. The system displays the create account/log in page.   2. The system prompts the user to enter the required information.   3.1 The system validates the data entered.  3.2 The system sends a verification code.  3.3 The system displays the verification page.  4.1 The system directs the user to the home page. |
| **Exception Conditions:** | 1. If the user entered invalid data. 2. If the user entered invalid verification code. 3. If the user cancels the creation of account. | |

Table 6: Rent Vehicle Use Case Description

|  |  |  |
| --- | --- | --- |
| **Use Case Name:** | Rent Vehicle | |
| **Scenario:** | Vehicle renter wants to rent a vehicle. | |
| **Triggering Event:** | Request a vehicle | |
| **Brief Description:** | This will allow the user to rent a vehicle. | |
| **Actors:** | Vehicle renter | |
| **Related Use Case:** | Create Account | |
| **Stakeholders:** | Vehicle renter | |
| **Preconditions:** | The vehicle renter must be logged in. | |
| **Post conditions:** | The vehicle must be able to book a vehicle.  A notification must be sent to the vehicle owner. | |
| **Assumptions:** | The vehicle renter is already logged in. | |
| **Flow of Activities:** | **Actor** | **System** |
| 1. The renter taps **Browse Vehicles.** 2. The renter clicks on a vehicle listing. 3. The renter clicks **Book Now**. 4. The renter enters the trip details. 5. The renter taps **Send Request**. | * 1. The system displays the list of vehicles.   2.1 The system displays the vehicle information page.  3.1 The system prompts the renter to enter trip details.  5.1 The system sends a rent request notification to the vehicle owner. |
| **Exception Conditions:** | 1. If required data is missing. 2. If booking is cancelled. | |

|  |  |  |
| --- | --- | --- |
| **Use Case Name:** | Manage Listing | |
| **Scenario:** | Vehicle owner wants to create, view, update, or delete a listing. | |
| **Triggering Event:** | Request to add, edit, or delete a listing | |
| **Brief Description:** | This will allow the vehicle owner to create, view, update, or delete a vehicle listing. | |
| **Actors:** | Vehicle owner | |
| **Related Use Case:** | Create Account | |
| **Stakeholders:** | Vehicle owner | |
| **Preconditions:** | The vehicle owner must be logged in. | |
| **Post conditions:** | The listing must be published in the mobile application.  The vehicle owner must be able to create, view, update, or delete a listing. | |
| **Assumptions:** | The vehicle owner is already logged in. | |
| **Flow of Activities:** | **Actor** | **System** |
| 1. The vehicle owner taps onthe **Listing** tab**.**   2.A. The vehicle owner taps **Create Vehicle Listing.**  2.B. The vehicle owner enters all required information.  3. The vehicle owner views an existing listing.  4. The vehicle owner edits the vehicle listing  5. The vehicle owner deletes an existing listing. | * 1. The system displays the listing page.   2.A.1 The system prompts the user to enter required vehicle information and upload required documents.  2.B.1 The system will store the information and publish the vehicle listing in the app.  3.1 The system displays the vehicle listing information.  4.1 The system displays the vehicle listing in edit mode.  5.1 The system will remove the listing from the app. |
| **Exception Conditions:** | 1. If the user entered an invalid data. 2. If there is no existing listing, user must create one first to view, update, or delete. | |

Table 7: Manage Listing Use Case Description

Table 8: Approve Request Use Case Description

|  |  |  |
| --- | --- | --- |
| **Use Case Name:** | Approve Request | |
| **Scenario:** | Vehicle owner wants to accept a rent request. | |
| **Triggering Event:** | Request to manage listing | |
| **Brief Description:** | This will allow the vehicle owner to accept a rent request. | |
| **Actors:** | Vehicle owner | |
| **Related Use Case:** | Create Account, Rent Vehicle | |
| **Stakeholders:** | Vehicle owner | |
| **Preconditions:** | The vehicle owner must be logged in.  The vehicle owner must receive a rent request notification. | |
| **Post conditions:** | The vehicle owner must be able to accept a request.  The vehicle renter must receive a rent request. | |
| **Assumptions:** | The vehicle owner is already logged in. | |
| **Flow of Activities:** | **Actor** | **System** |
| 1. The vehicle owner taps on **Notifications**. 2. The vehicle owner taps on a request. 3. The vehicle renter taps **Accept.** | * 1. The system displays the notification page.   2.1 The system displays the trip details sent by the renter.  3.1 The system notifies the renter that the rent request has been accepted. |
| **Exception Conditions:** | 1. If the owner declines the rent request. | |

|  |  |  |
| --- | --- | --- |
| **Use Case Name:** | View Profile | |
| **Scenario:** | The vehicle renter wants to view the profile of the vehicle owner. | |
| **Triggering Event:** | Profile request | |
| **Brief Description:** | This will allow the vehicle renter to view the profile of the vehicle owner or vice versa. | |
| **Actors:** | Vehicle renter | |
| **Related Use Case:** | Rent a Vehicle | |
| **Stakeholders:** | Vehicle renter | |
| **Preconditions:** | The vehicle renter must select a vehicle listing. | |
| **Post conditions:** | The vehicle renter must be able to see the vehicle owner’s profile. | |
| **Assumptions:** | The vehicle renter is already logged in.  The vehicle renter is already on a vehicle listing page. | |
| **Flow of Activities:** | **Actor** | **System** |
| 1. The vehicle renter clicks on the icon of the vehicle owner. | 1.1 The system displays the vehicle information.  2.1 The system displays the profile of the owner. |
| **Exception Conditions:** | 2. If the vehicle renter wants to view the feedbacks on the owner, initiate View Feedback use case. | |

Table 9: View Profile Use Case Description

Table 10: Write Feedback Use Case Description

|  |  |  |
| --- | --- | --- |
| **Use Case Name:** | Write Feedback | |
| **Scenario:** | The vehicle renter wants to give feedback on the owner. | |
| **Triggering Event:** | Request to give user feedback | |
| **Brief Description:** | This will allow the vehicle renter to give feedback on the owner. | |
| **Actors:** | Vehicle renter | |
| **Related Use Case:** | View Profile, Rent Vehicle | |
| **Stakeholders:** | Vehicle renter | |
| **Preconditions:** | The vehicle renter must select a vehicle listing. | |
| **Post conditions:** | The vehicle renter must be able to write feedback. | |
| **Assumptions:** | The vehicle renter is already on the profile of the vehicle owner. | |
| **Flow of Activities:** | **Actor** | **System** |
| 1. The vehicle renter wants to give feedback**.** 2. The renter will give feedback. | * 1. The system displays the feedback form.   2.1 The system will save the feedback. |
| **Exception Conditions:** | - | |

### Activity Diagram

The following diagrams graphically represent the flow of activities of each use cases of the system.

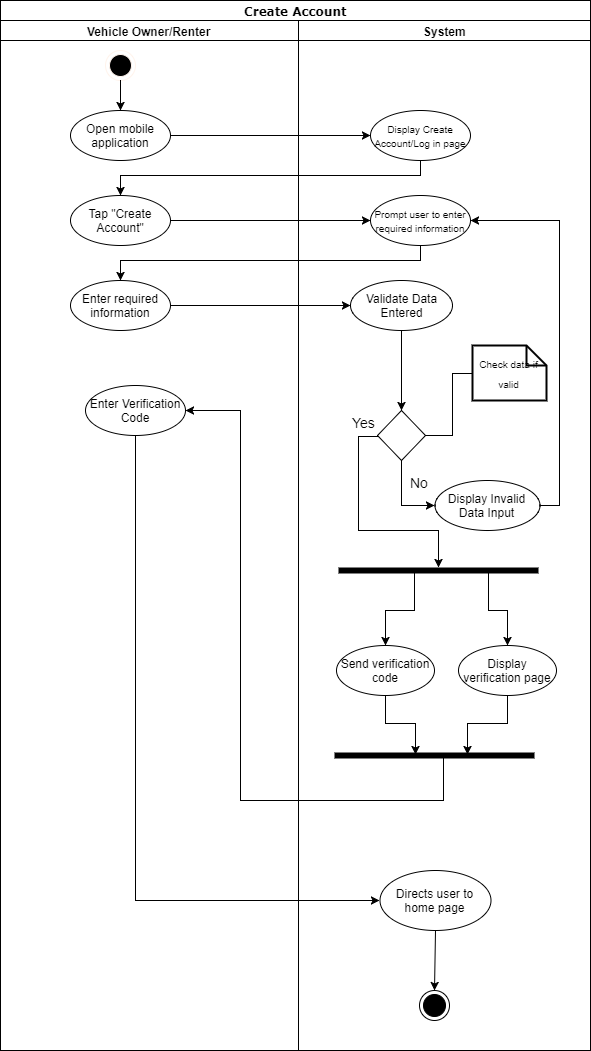
****

Figure 15: Create account activity diagram

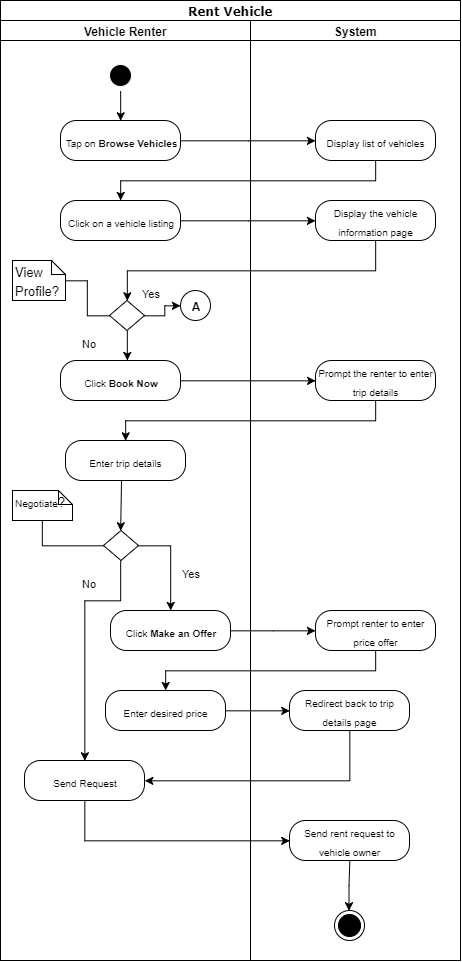
****

Figure 16: Rent vehicle activity diagram

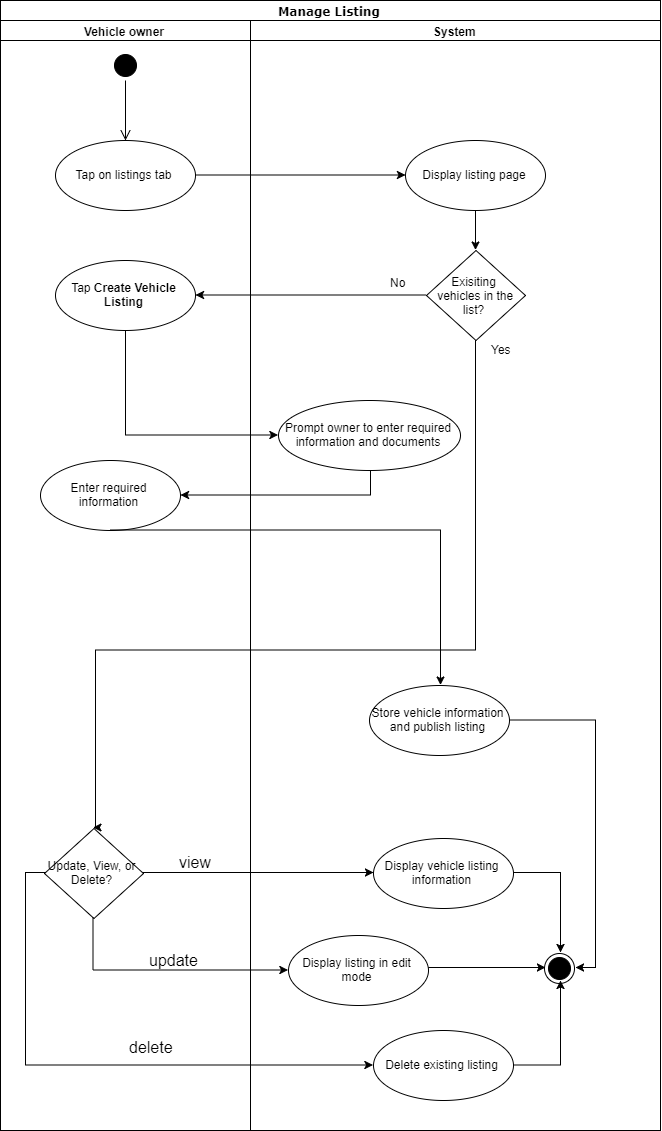


Figure 17: Manage listing activity diagram

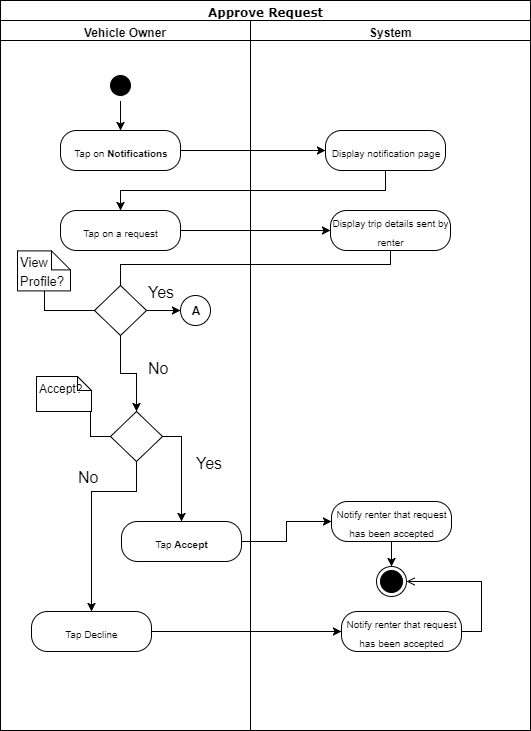
****

Figure 18: Approve request activity diagram

****

Figure 19: View profile activity diagram

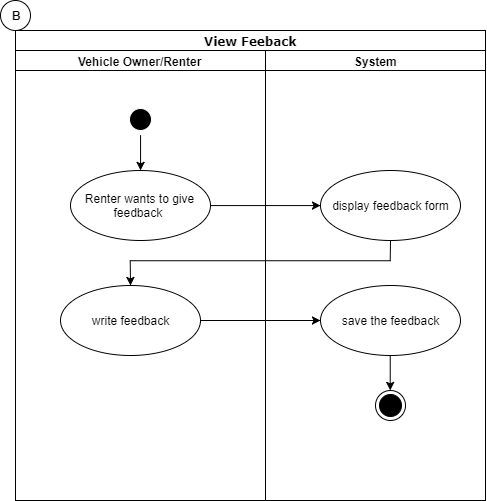
****

Figure 20: View feedback activity diagram

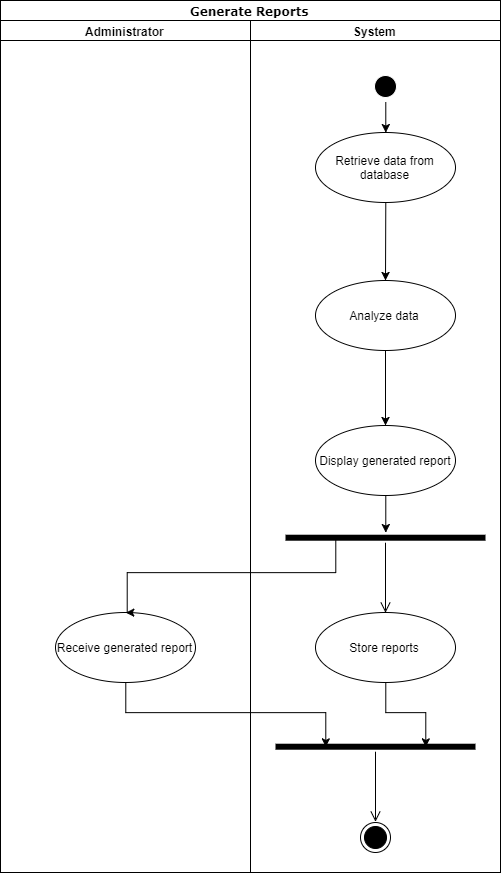
****

Figure 21: Generate reports activity diagram

### Sequence Diagram

The following diagrams shows a graphical representation of how the objects of the system interact with each other

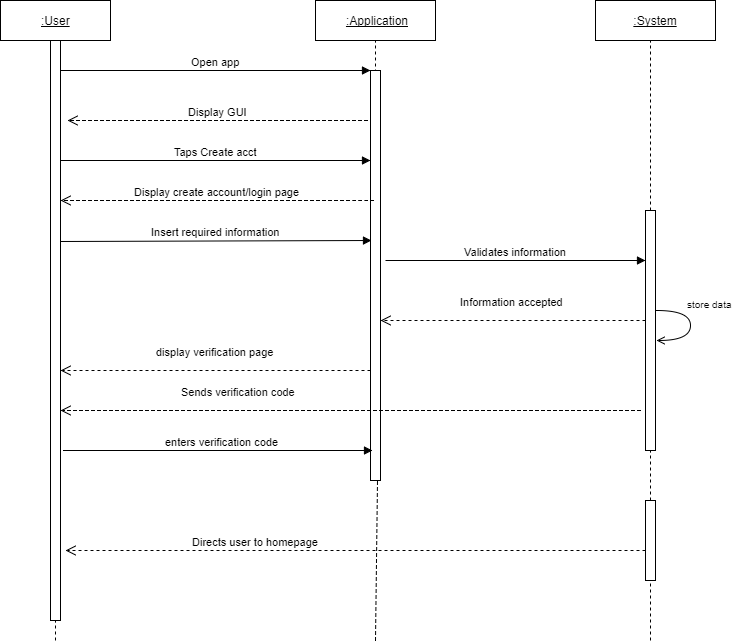
****

Figure 22: Create account sequence diagram

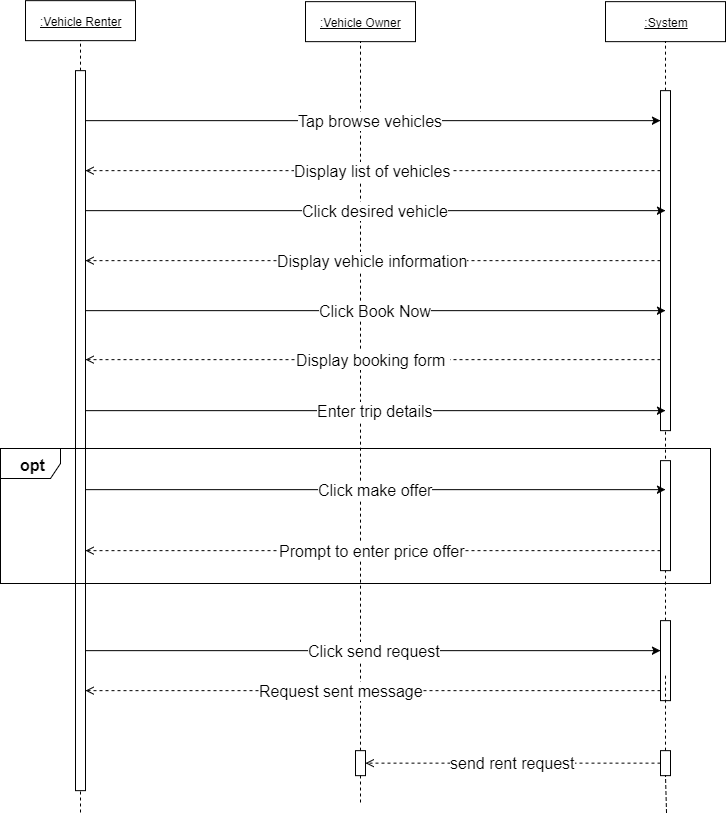


Figure 23: Rent vehicle sequence diagram

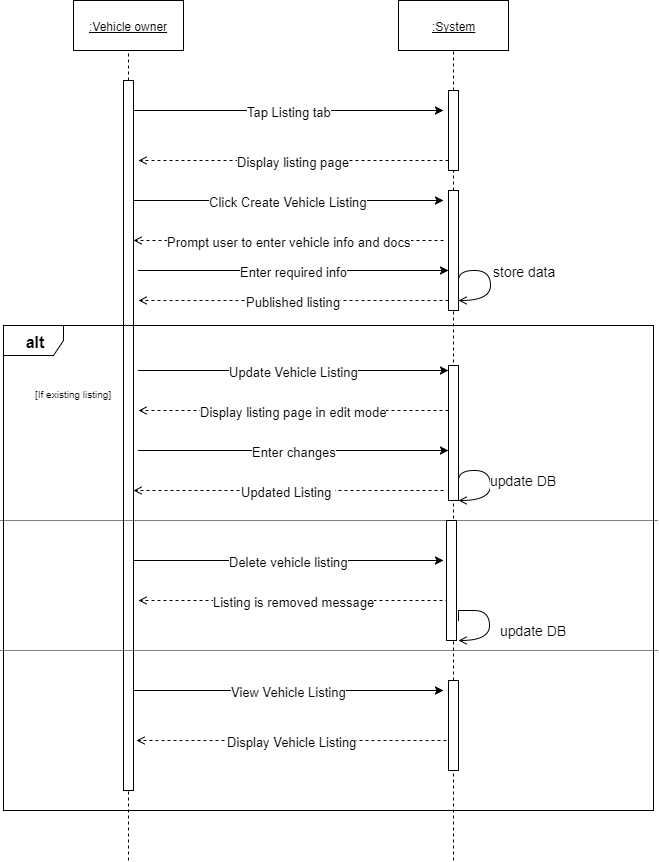


Figure 24: Manage listing sequence diagram



Figure 25: Approve request sequence diagram

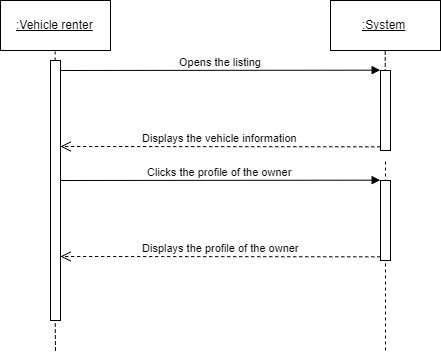


Figure 26: View profile sequence diagram

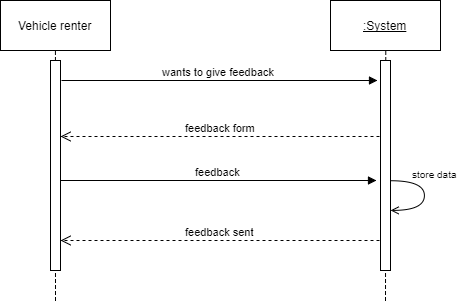


Figure 27: View feedback sequence diagram

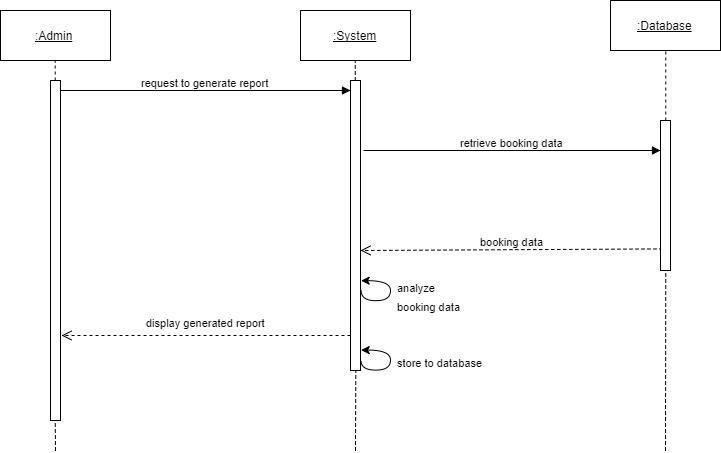


Figure 28: Generate reports sequence diagram

### State Diagram

The following diagrams show the state or behavior of the system in response to the action executed.

### C:\Users\Martha\Downloads\SD-Create (1).png

Figure 29: State diagram of create account use case

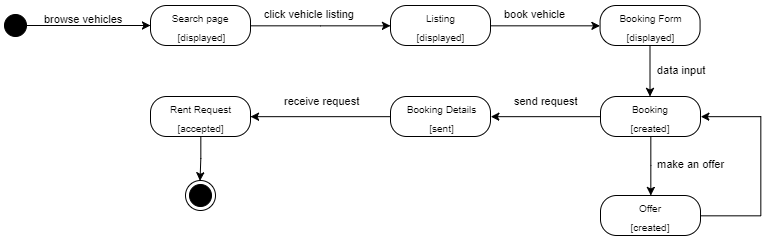


Figure 30: State diagram of rent vehicle use case

### C:\Users\Martha\Downloads\SD-manage.png

Figure 31: State diagram of manage listing use case

### C:\Users\Martha\Downloads\SD-approve.png

Figure 32: State diagram of approve request use case

C:\Users\Martha\Downloads\SD-viewprof (1).png

C:\Users\Martha\Downloads\SD-writefeed.png

Figure 33: State diagram of view profile and feedback use case



Figure 34: State diagram of generate reports

### Timing Diagram

Figures 36 to 40 shows the graphical representation of conditions that are changing within lifelines of the system,

## C:\Users\Martha\Downloads\TD-Create.UPDATED.png

Figure 35: Timing diagram of create account use case

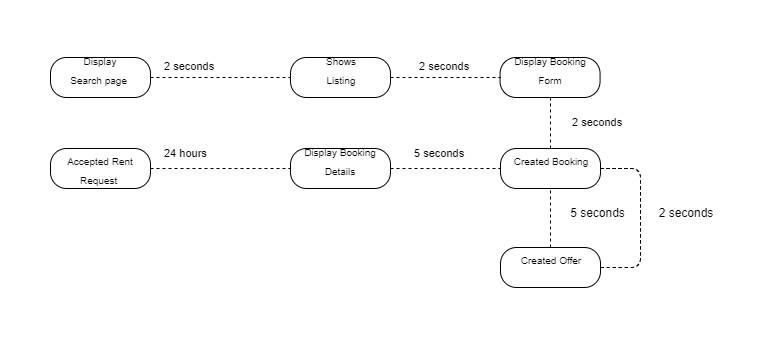


Figure 36: Timing diagram of rent vehicle use case

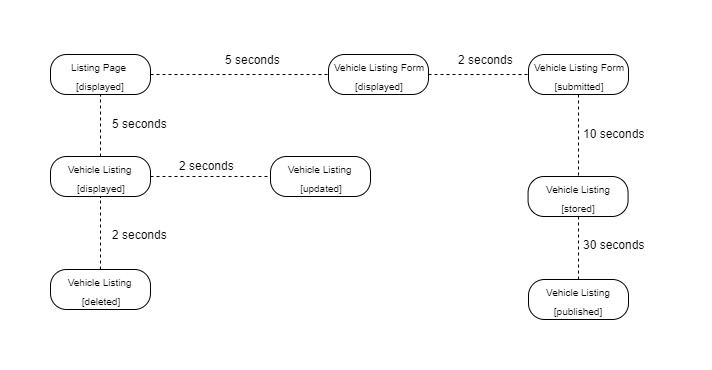


Figure 37: Timing diagram of manage listing use case

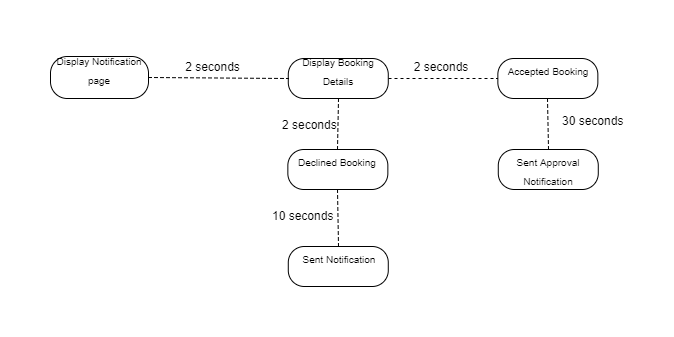
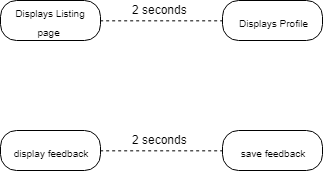


Figure 38: Timing diagram of approve request use case

Figure 39: Timing diagram of view profile and feedback use case



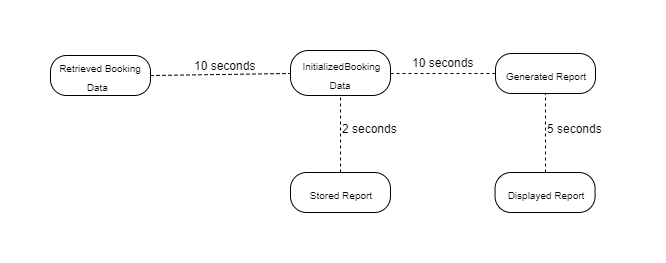


Figure 40: Timing diagram of generate reports use case

### Package Diagram

Figure 41 shows the grouped elements of the system and the dependencies between them.

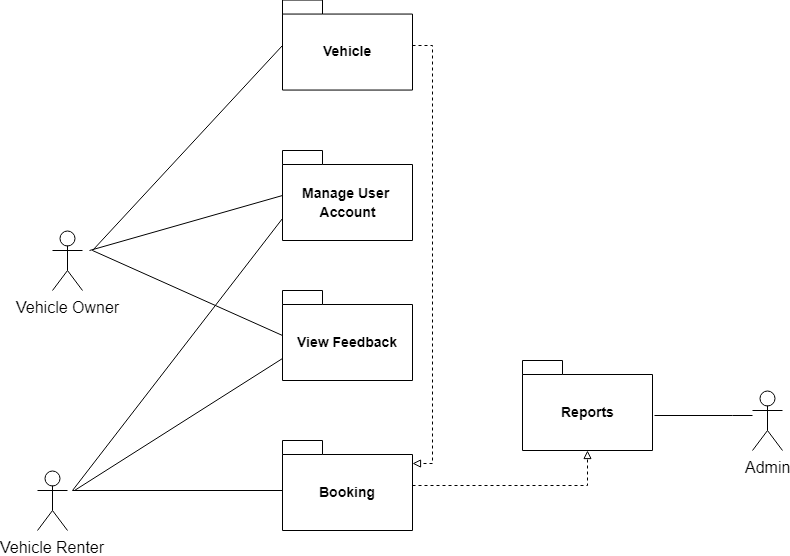


Figure 41: Package diagram

### Communication Diagram

Figure 42 shows the interaction between components that includes the message or data that travels from one to another element.



Figure 42: Communication diagram

### Component Diagram

Figure 43 shows the relationship between components in the system and how they are connected together.

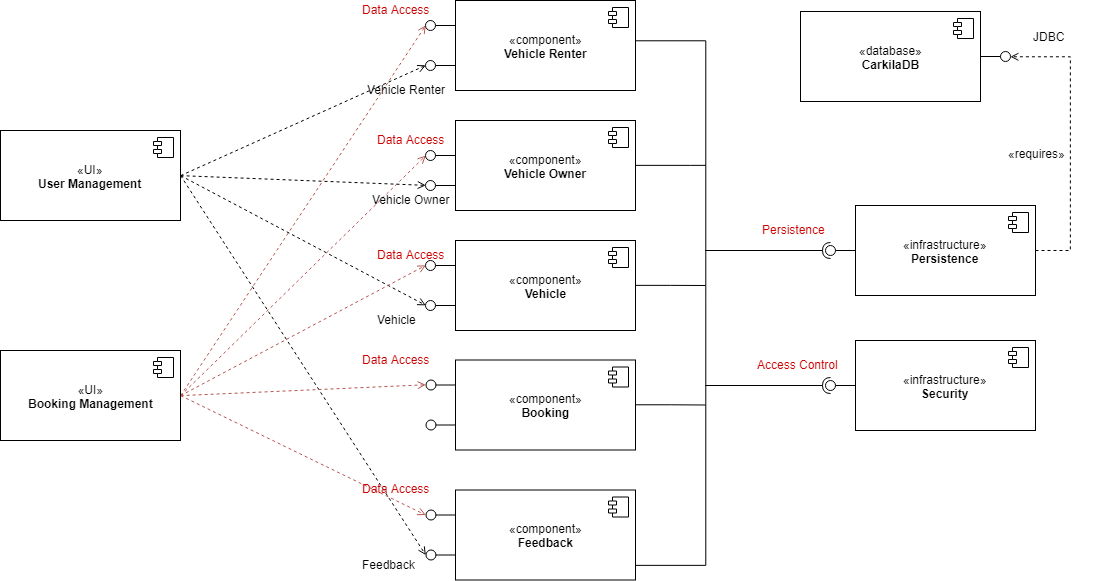


Figure 43: Component diagram

### Composite Structure Diagram

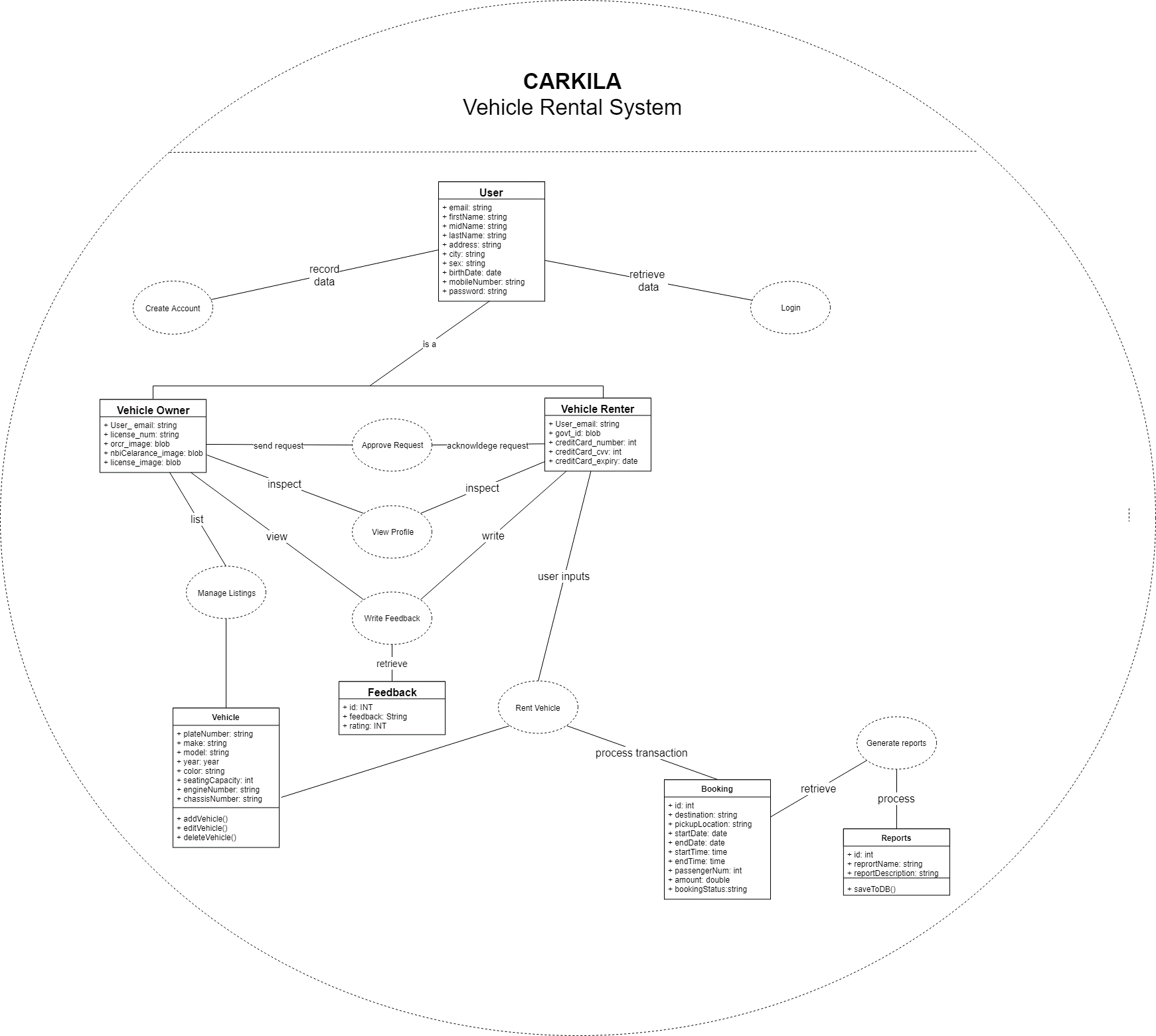
Figure 44 show the internal structure of classes in the system; their relationship and how they are connected.

Figure 44: Composite structure diagram

### Deployment Diagram

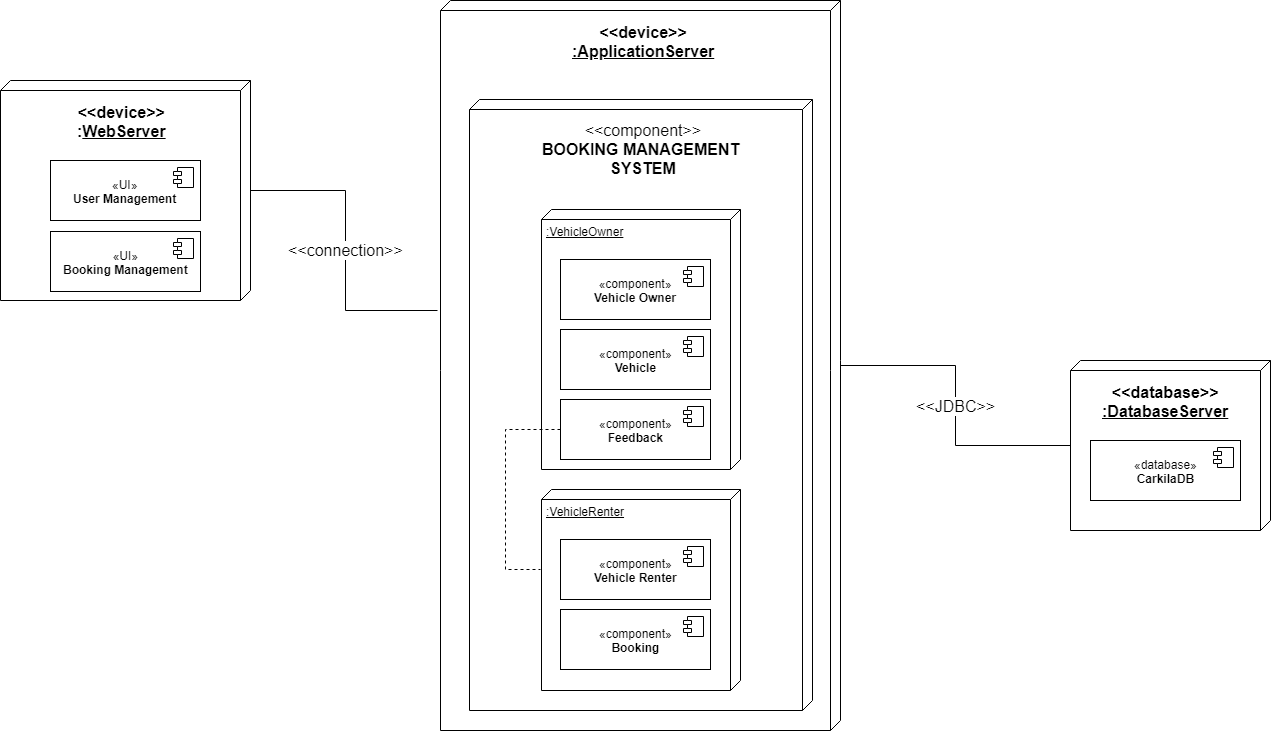
****

Figure 45: Deployment diagram

### Interaction Overview Diagram

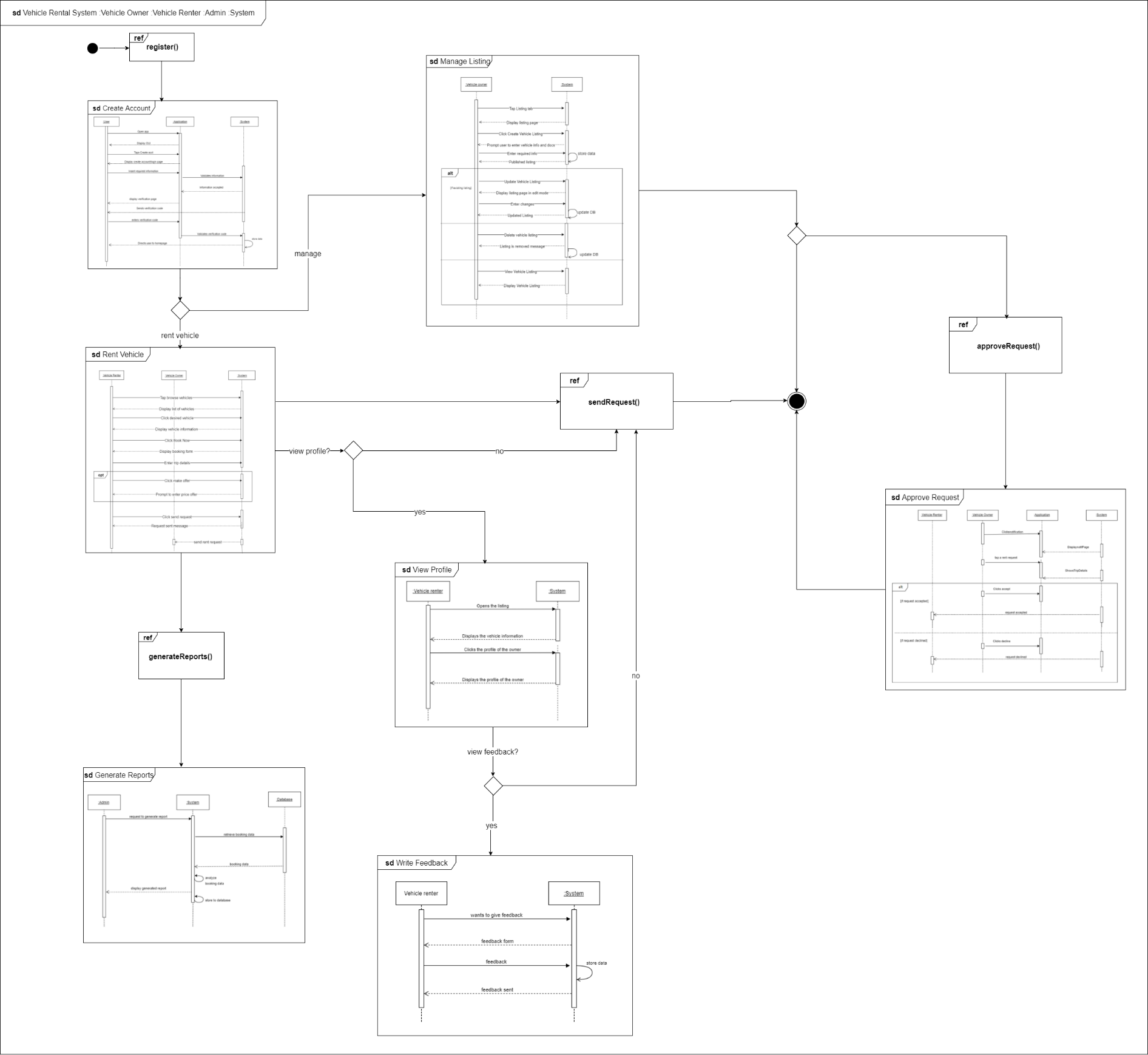


Figure 46: Interaction overview diagram

# Bibliography

A*rkila*. (n.d.). Retrieved from Arkila.ph: http://arkila.ph/

Arkila.ph. (2015, February 25). *Arkila.PH: The fastest and easiest way to rent a vehicle in the Philippines*. Retrieved from IdeaSquare: https://ideasquare.io/t/arkila-ph-the-fastest-and-easiest-way-to-rent-a-vehicle-in-the-philippines/83

Ballus-Armet, I., Shaheen, S. A., Clonts, K., & Weinzimmer, D. (2014). *Peer-to-Peer Carsharing*. Retrieved from http://www.carsharing.org/research/TSRC/P2Pcarsharing\_TRR2014.pdf

Covington, P., Adams, J., & Sagrin, E. (2016). Deep Neural Networks for YouTube Recommendations.

Deepak. (2015, September 25). In-depth Feature Analysis of P2P Car Renting Marketplace. *Takeover the Peer-to-Peer Car Sharing Marketplace with These Essential Website Features*. Retrieved from https://www.fatbit.com/fab/peer-to-peer-car-sharing-marketplace-features-business-model/

Dishman, L. (2010, June 21). *Brother, Can You Spare a Blender? The Scoop on Peer-to-Peer Equipment Rentals*. Retrieved from Fast Company: https://www.fastcompany.com/1662305/brother-can-you-spare-blender-scoop-peer-peer-equipment-rentals

Einav, L., Farronato, C., & Levin, J. (2015, August). *Peer-to-Peer Markets*. Retrieved from http://www.nber.org/papers/w21496.pdf

FATbit Technologies. (2015, July 29). Facilitating Car Renting-Sharing with Internet: Business Mdel & Site Feature. *Planning to Launch Car Renting Marketplace? Script Features to Succeed!* Retrieved from https://www.fatbit.com/fab/start-peer-to-peer-car-renting-portal-with-these-advanced-features/

Fraiberger, S., & Sundararajan, A. (2016, October 6). *Peer-to-Peer Rental Markets in the Sharing Economy*. Retrieved from https://papers.ssrn.com/sol3/papers.cfm?abstract\_id=2574337

Greenrobot. (n.d.). Mobile Databases. *SQLite and SQLite alternatives for Android and iOS*. Retrieved from http://greenrobot.org/news/mobile-databases-sqlite-alternatives-and-nosql-for-android-and-ios/

Hertz. (n.d.). Retrieved from https://www.hertz.com

McCarthy, A. (2017, March 1). *Geolocating Carmen Sandiego*. Retrieved from https://www.gravitatedesign.com/blog/what-is-geolocation/

Peebles, C. (2013, December 8). *Android Becomes Top Mobile Operating System in the Philippines*. Retrieved from http://mobileadvertisingwatch.com/android-becomes-top-mobile-operating-system-philippines-2246

Peltier, D. (2015, November 2). *5 New Travel Startups Redefining the Model for Car Rentals*. Retrieved from https://skift.com/2015/11/02/5-new-travel-startups-redefining-the-model-for-car-rentals/#1

Techopedia. (n.d.). *Geolocation*. Retrieved from https://www.techopedia.com/definition/1935/geolocation

Thomas, T. (2016, September 15). *How Artificial Intelligence Recommendation engines will change Online Publishing*. Retrieved from https://boomtrain.com/artificial-intelligence-recommendation-engines-publishing/

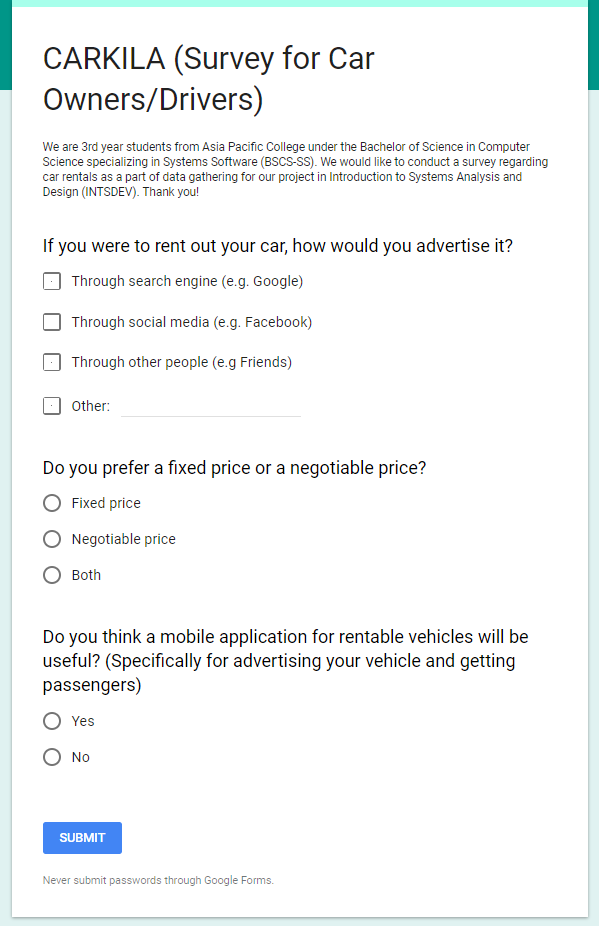
Vdovychenko, H. (2017, July 31). *How To Build a Peer to Peer Car Rental App Like Turo.* Retrieved from JSSolutions: https://jssolutionsdev.com/blog/how-to-build-a-car-rental-app-like-turo/

Viking Rent-A-Car. (n.d.). Retrieved from http://vikingcarrentals.com.ph/

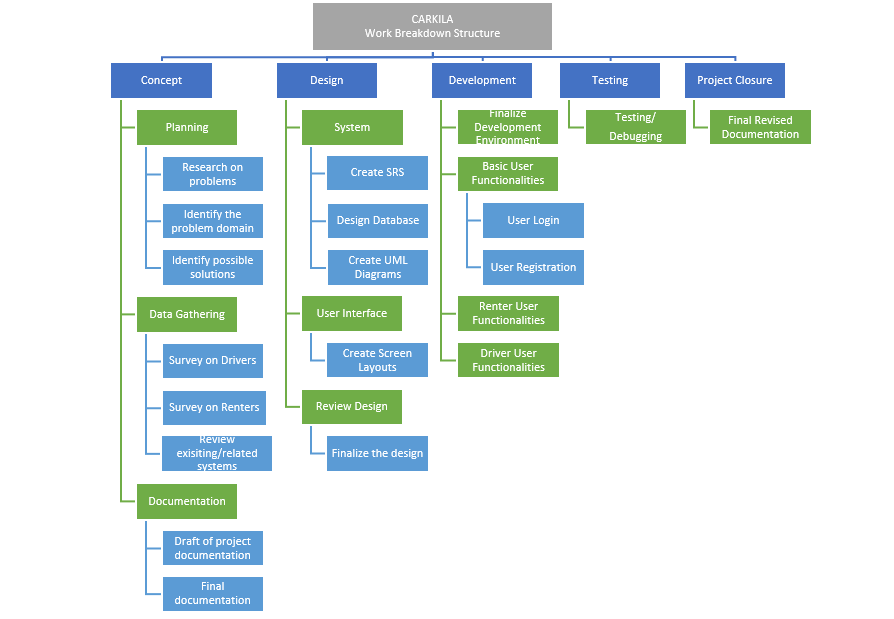
# Appendices

## Survey on Renters

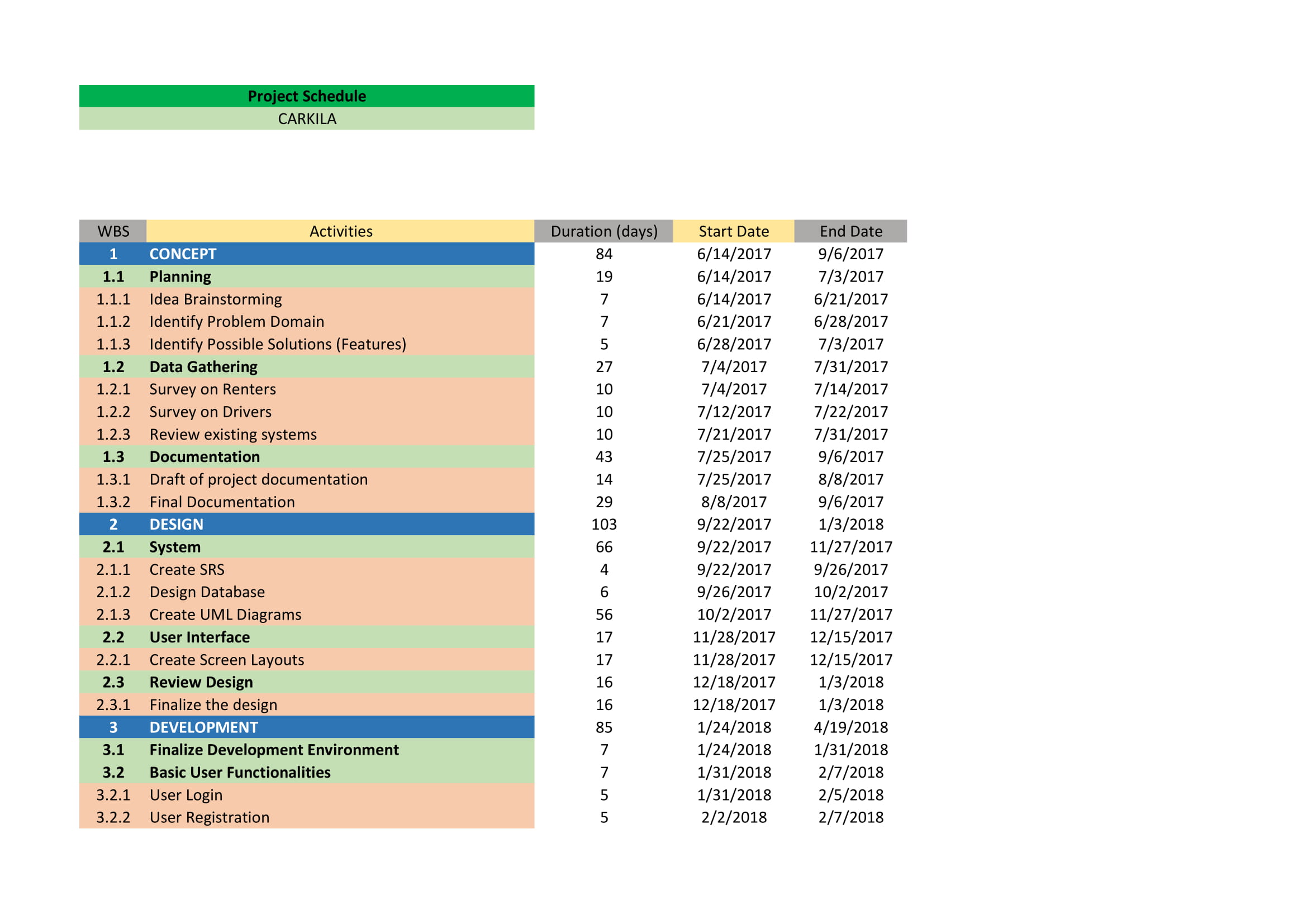
## Survey on Drivers



## Work Breakdown Structure



## Project Schedule

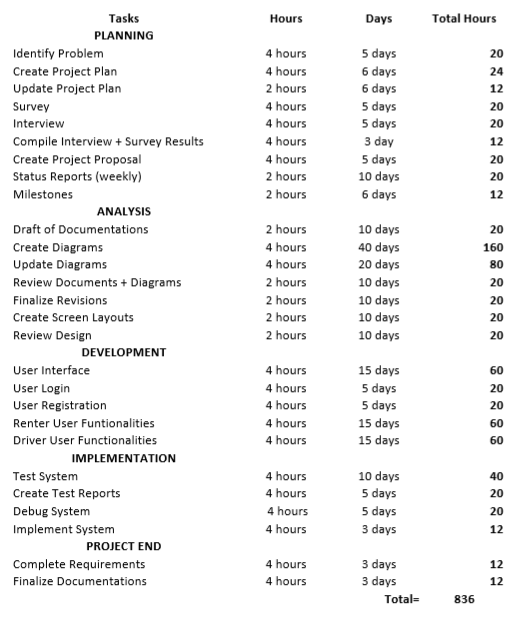


## Activity List

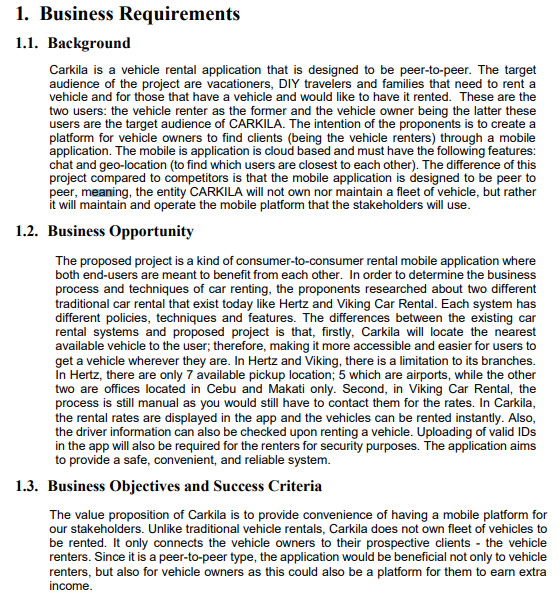


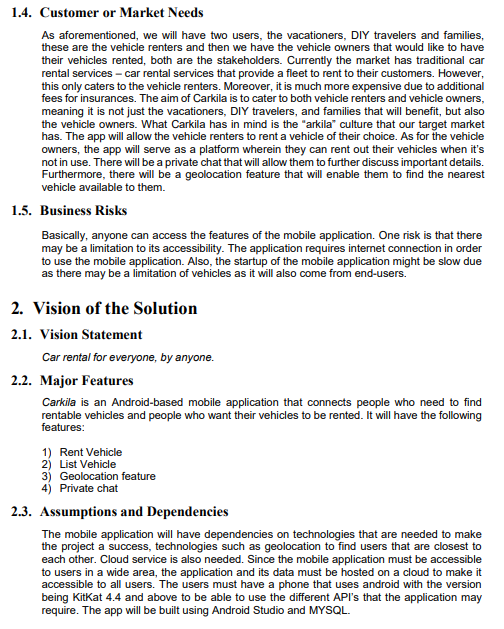


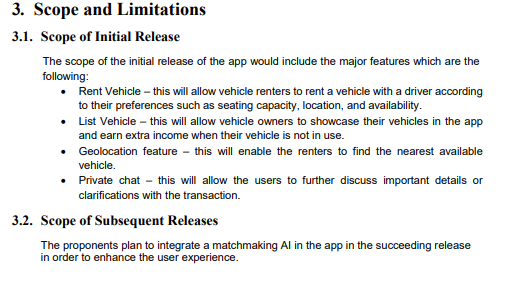
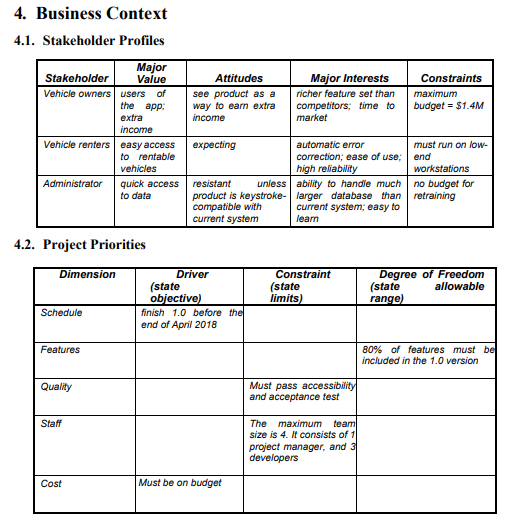
## Estimates

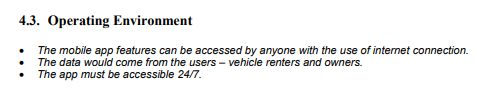


## Vision and Scope Document

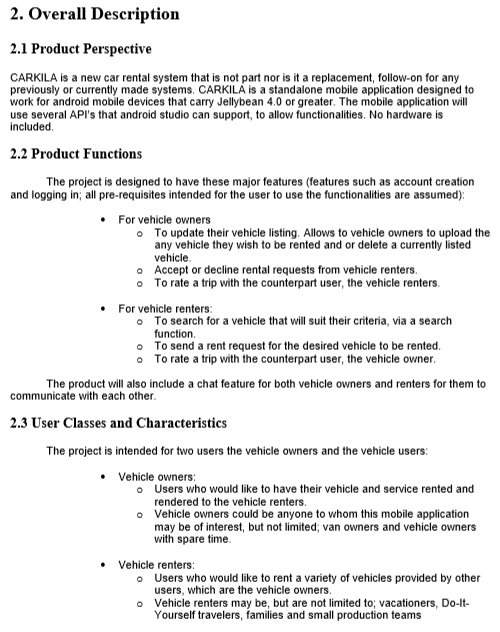




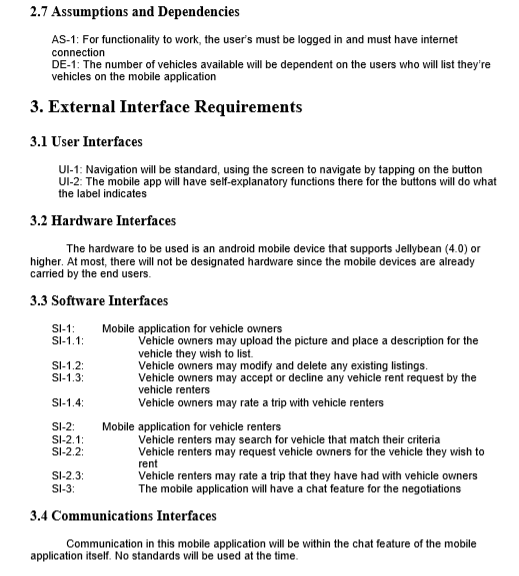


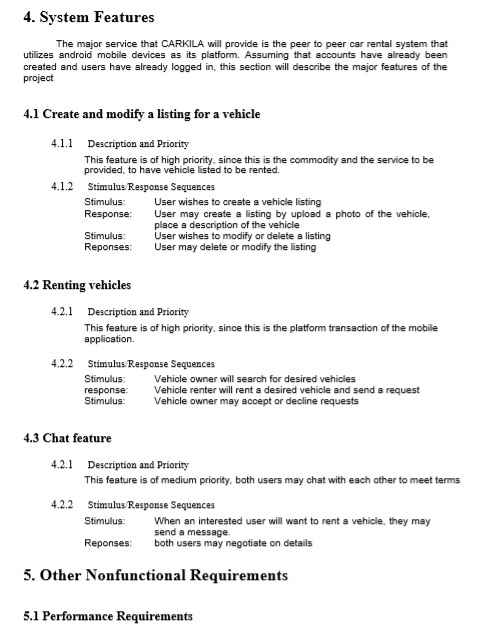
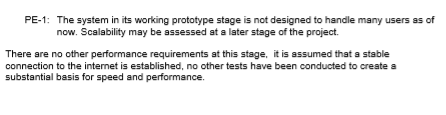


## Software Requirements Specification

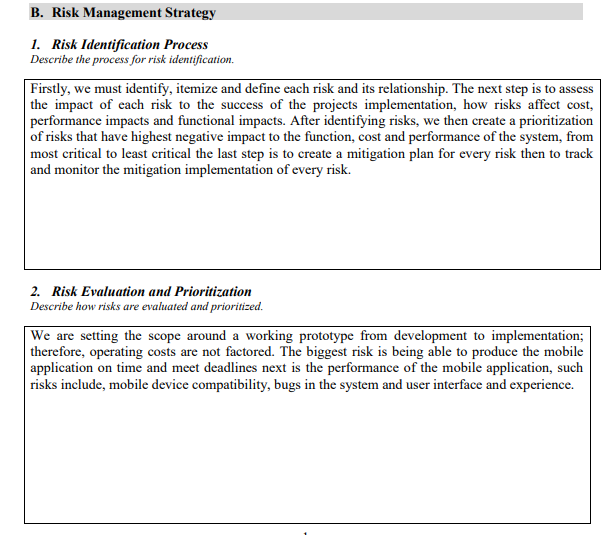


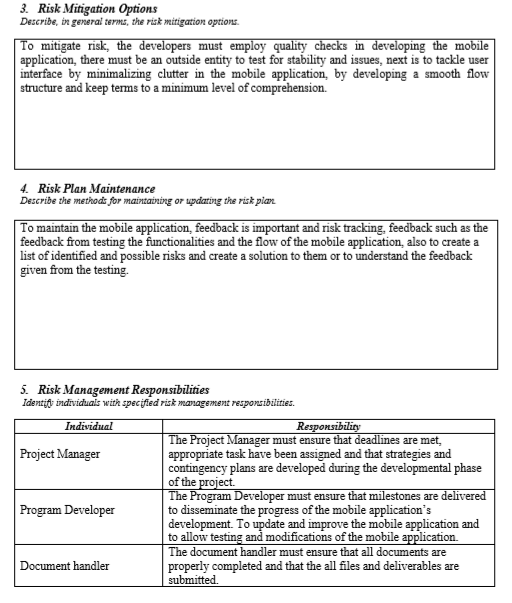


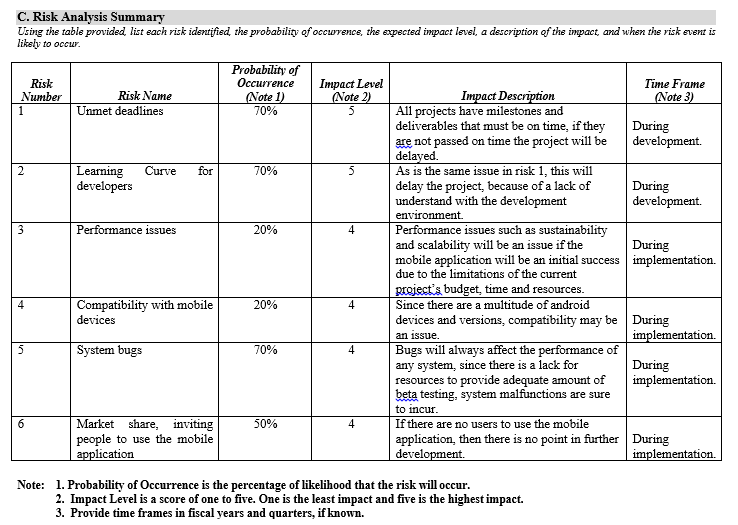


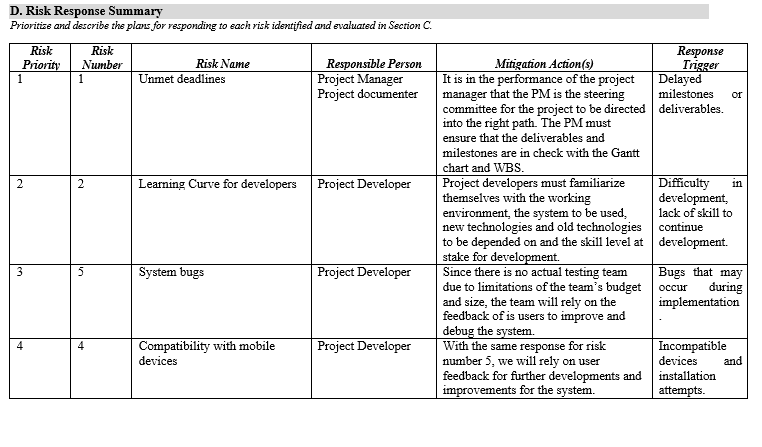


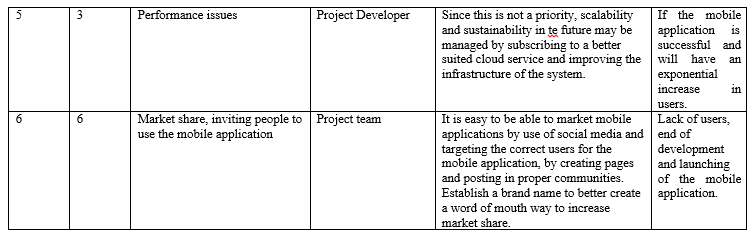
## Risk Management Plan





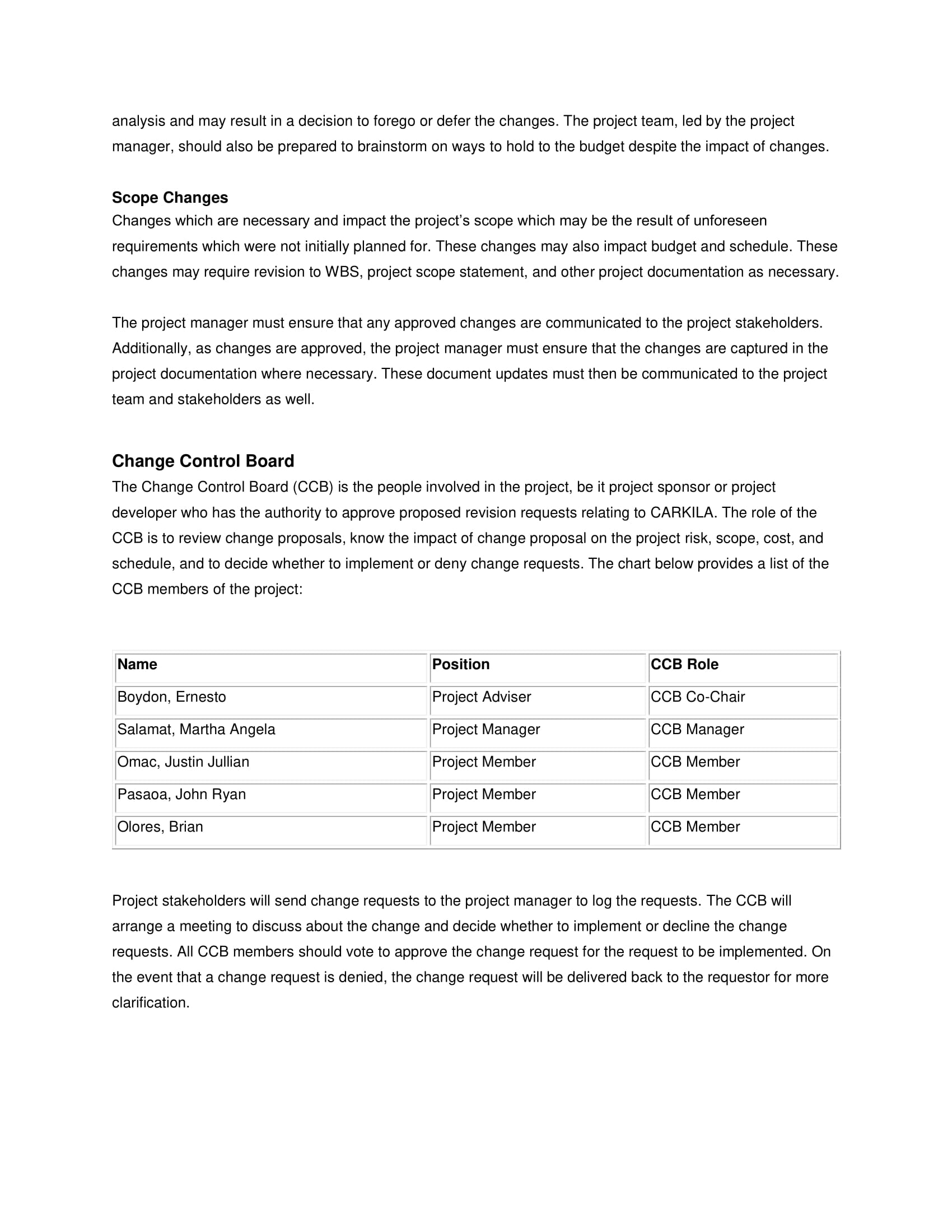


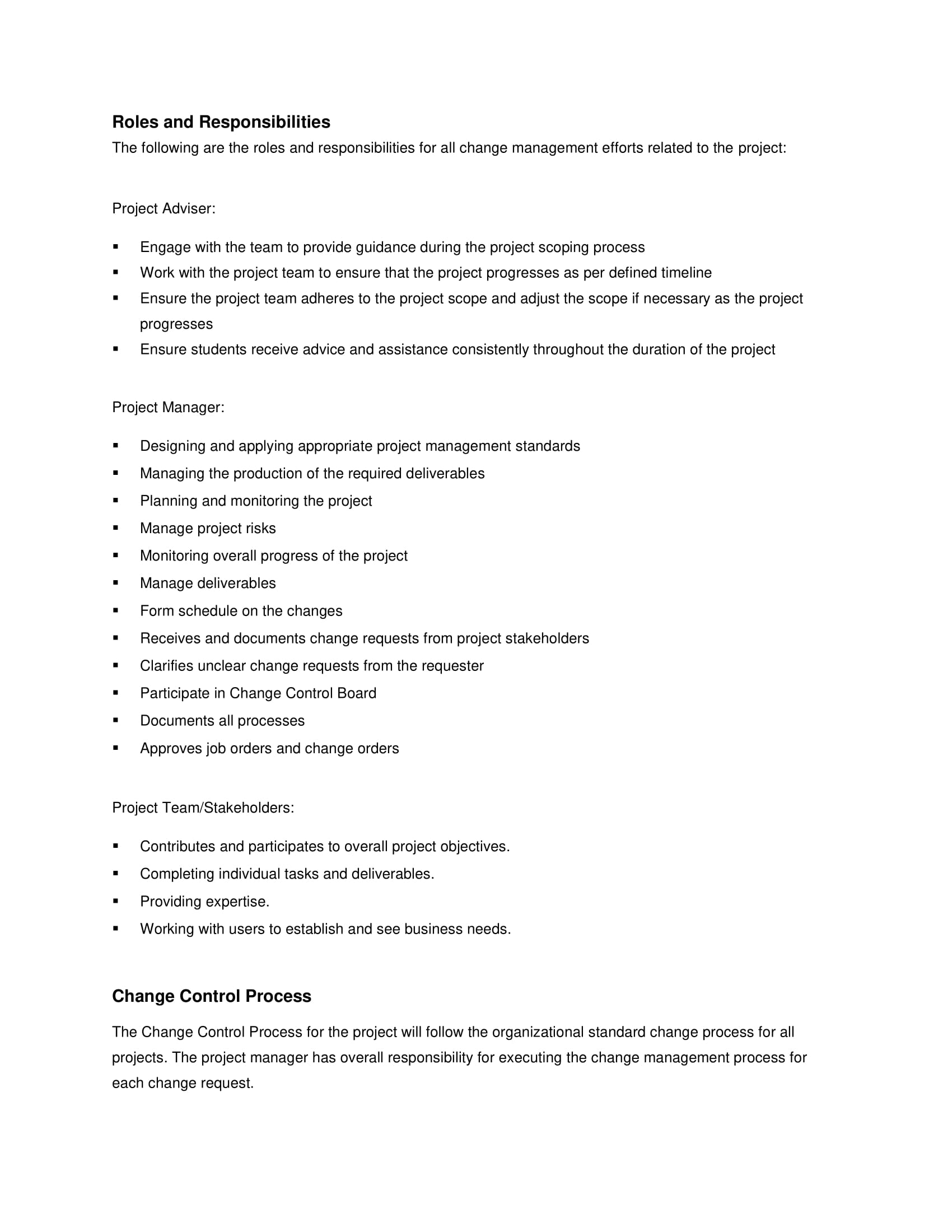
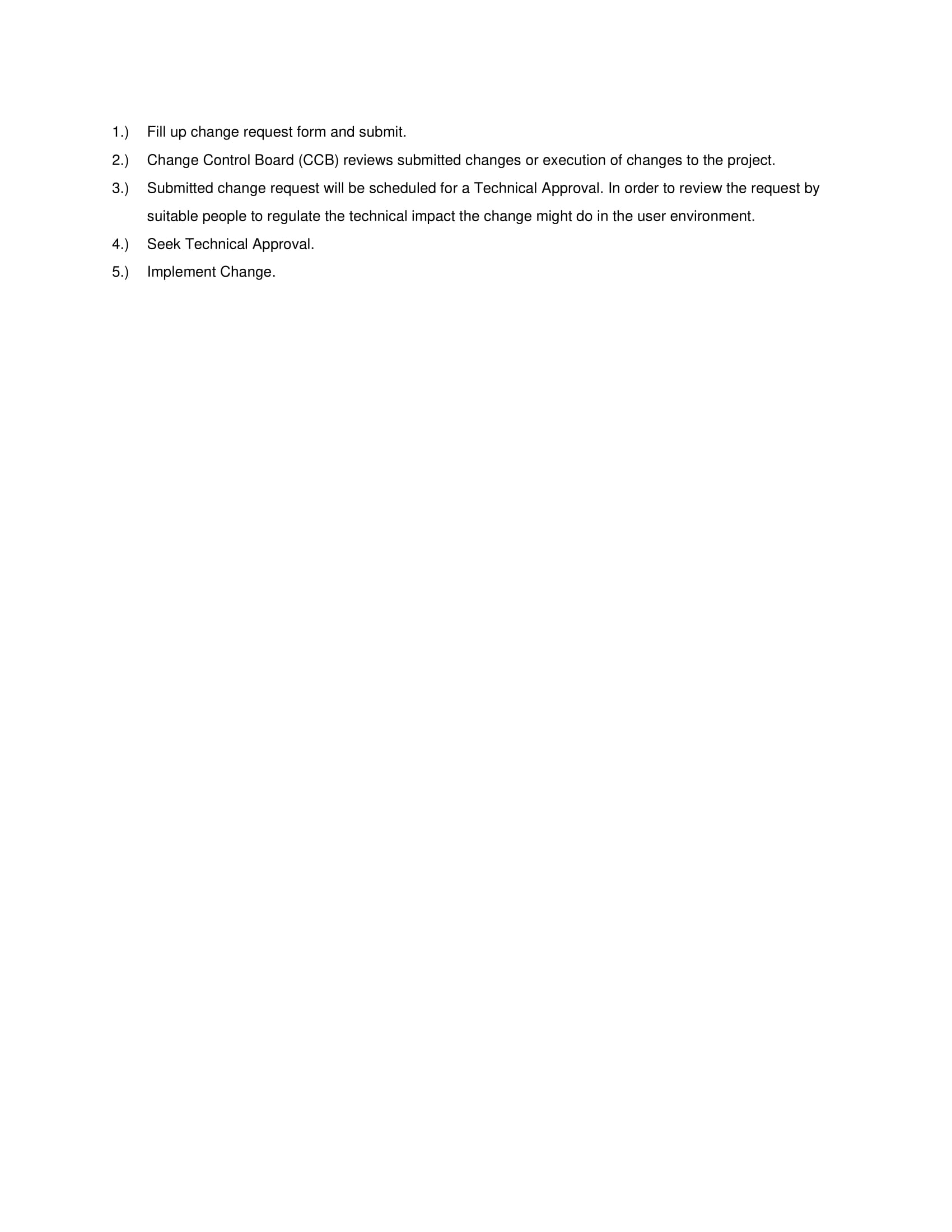




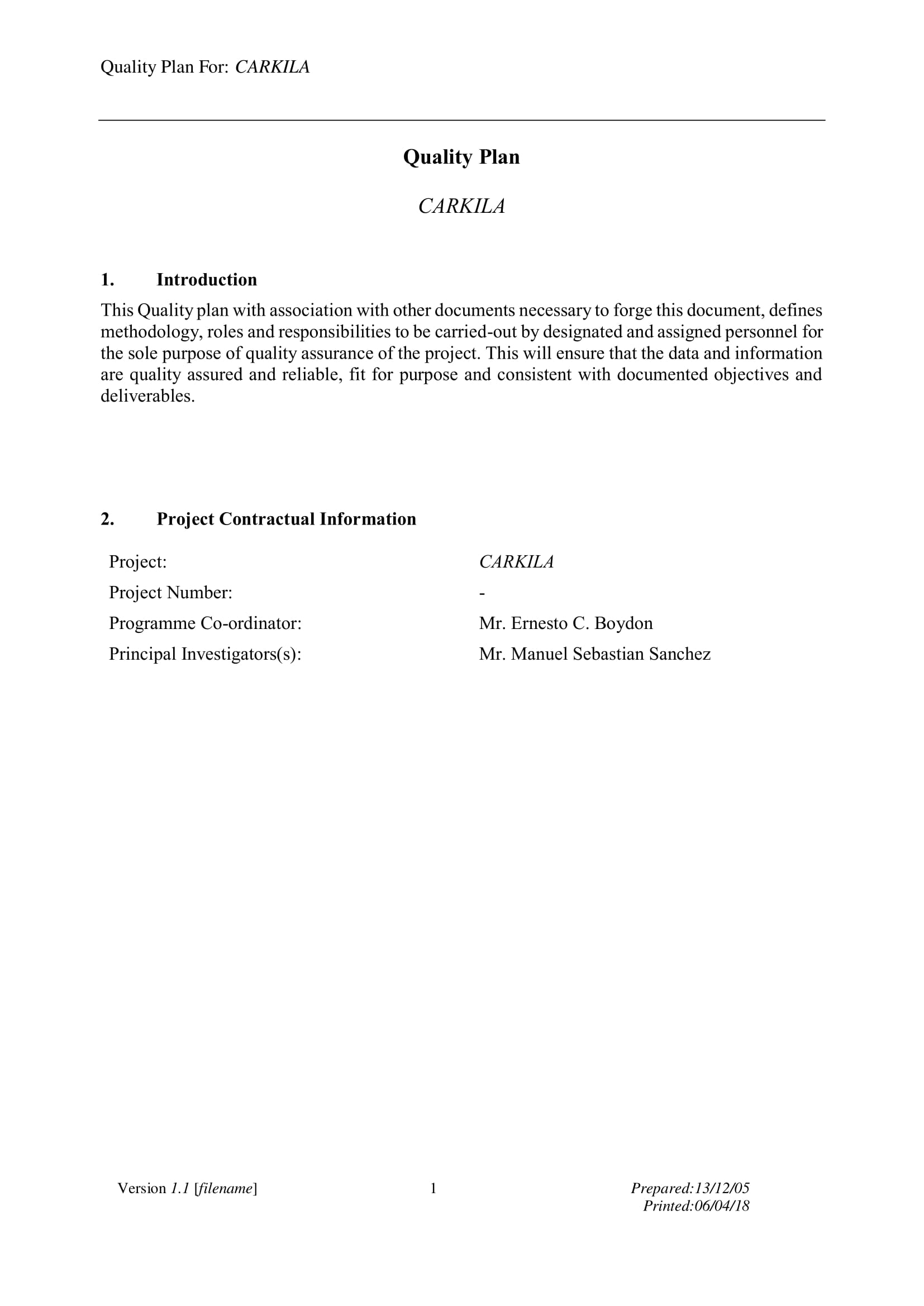
## Change Management Plan

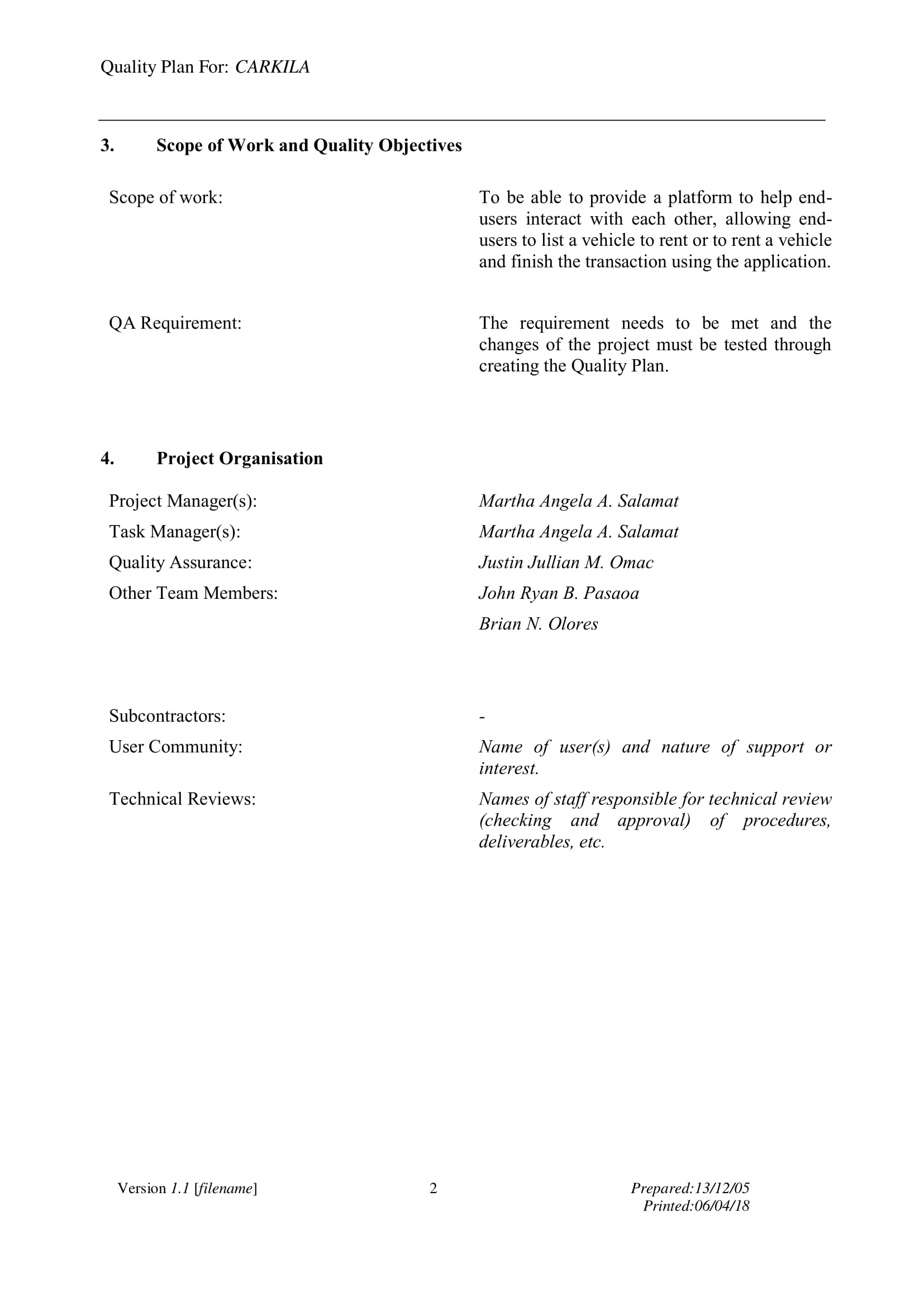




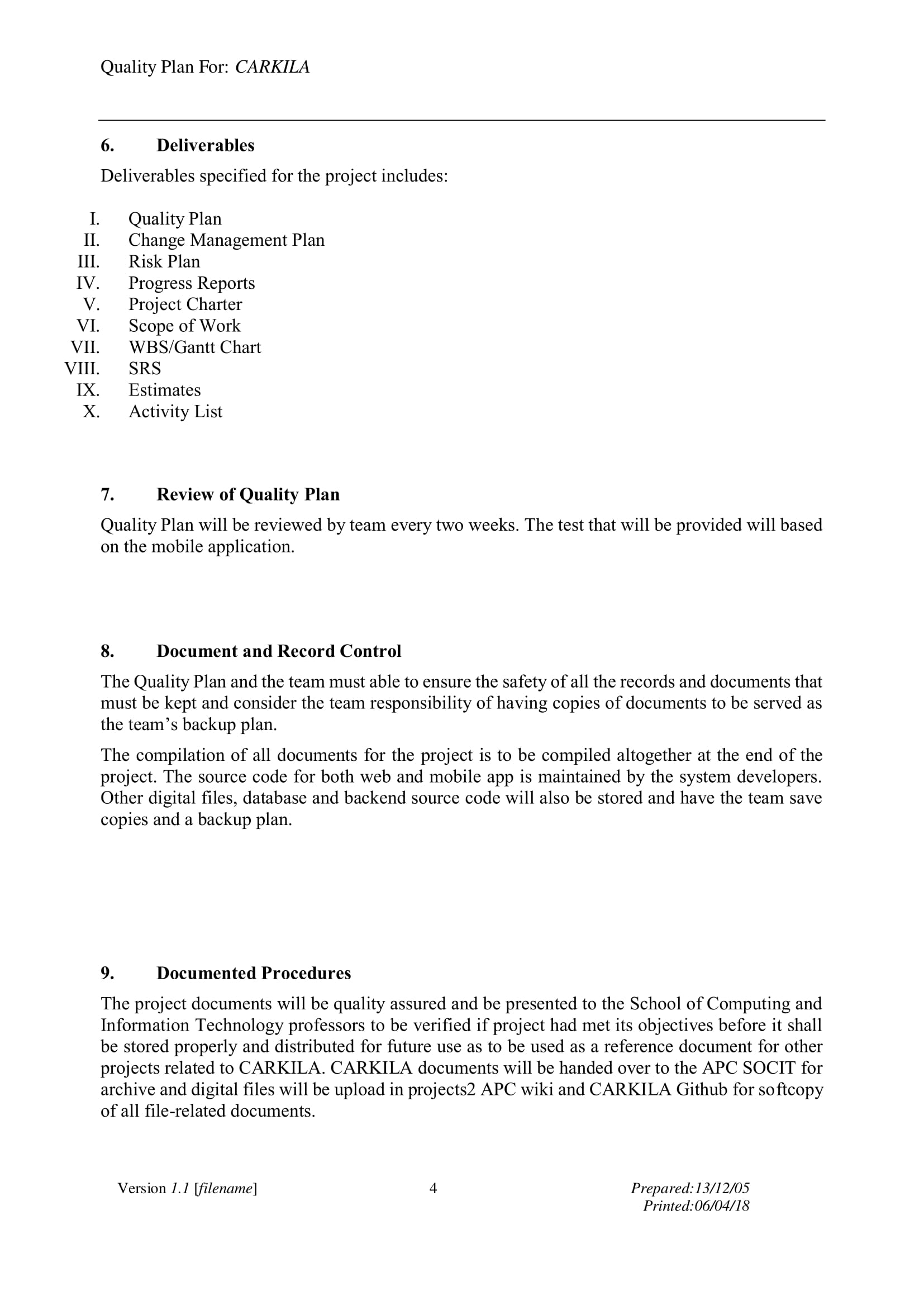


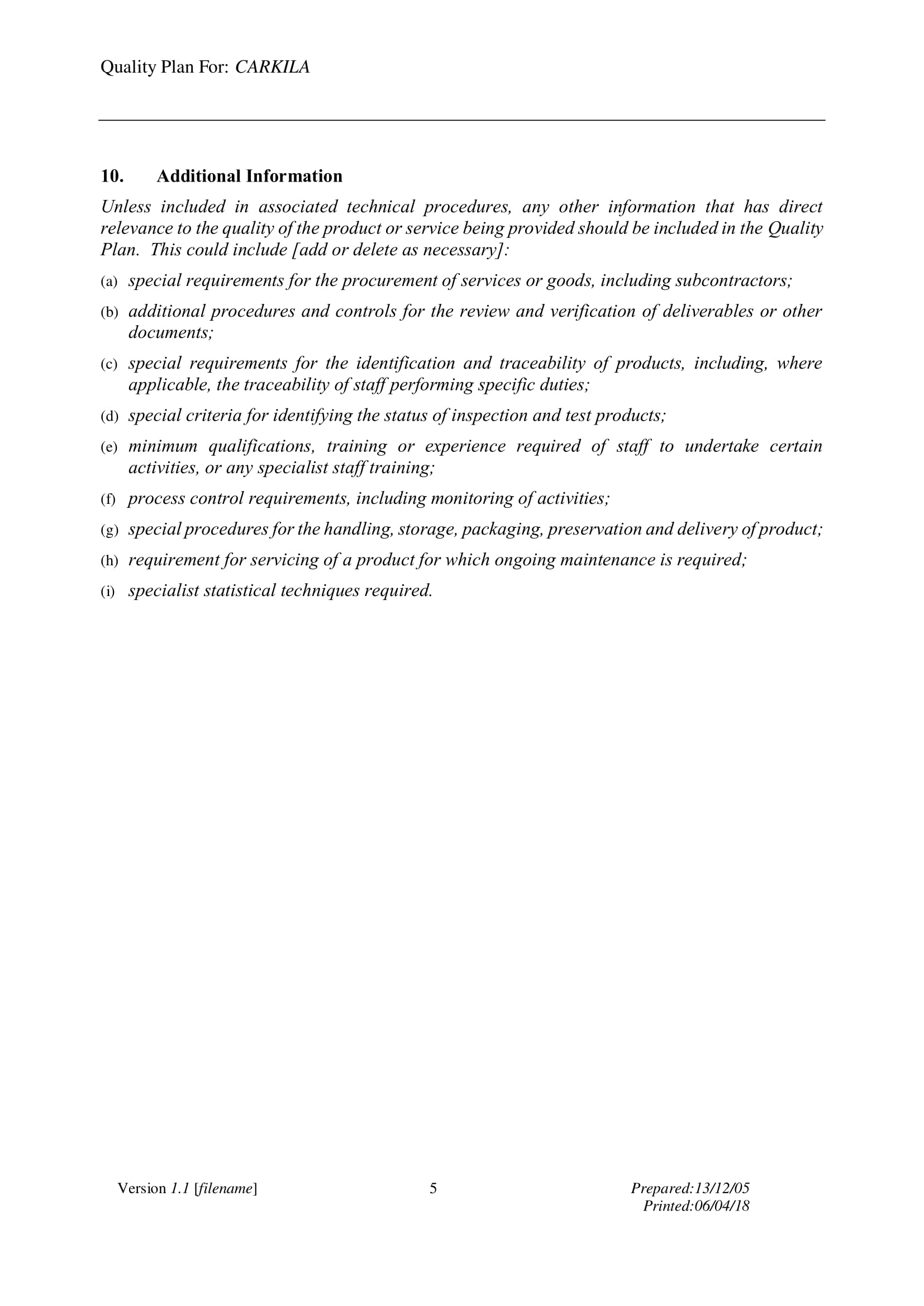
## Quality Management Plan











## Curriculum Vitae

|  |  |  |
| --- | --- | --- |
| **Name:** | **Martha Angela Salamat** | **A person smiling for the camera  Description generated with very high confidence** |
| **Complete Address:** | Block 1 Lot 10, Jamaica Street, Betterliving Subdivision |
| Paranaque City |
| **Contact Number/s:** | Landline: 822-15-91 |
| Mobile No.: (0995) 977-1609 |
| **Email Address:** | [masalamat@student.apc.edu.ph](mailto:masalamat@student.apc.edu.ph)  [salamat.martha@gmail.com](mailto:salamat.martha@gmail.com) |
| **Website:** | <https://www.linkedin.com/in/marthasalamat/> |  |

|  |  |
| --- | --- |
| **Personal Statement** | A hardworking individual who is willing to learn further and enhance my skills in programming. |
| **Education** | **Asia Pacific College**, Magallanes, Makati City  **Bachelor of Science in Computer Science with specialization in Systems Software (BSCS-SS)**  June 2015-present |
| **Work-Related Courses** | * Web Development * Software Quality Management * System Analysis and Design |
| **Academic Projects** | **CARKILA**   * Vehicle Rental Mobile Application * June 2017 – present * Project Manager/System Analyst/System Developer   **PONDO**   * Crowdfunding Investment Framework * January – April 2018 * System Analyst/Web developer   **Prison Art Pinoy Website**   * E-commerce website for prison art and paintings * September – January 2018 * Project Manager |
| **Technical Skills** | * Competent in PHP, HTML and CSS * Skilled in Java programming language * Have basic knowledge in Python and Ruby * Proficient in MS Office: Word, Excel, PowerPoint |
| **Awards & Recognitions** | * Cheers & Chants, 1st Runner Up   Asia Pacific College, 2015   * Speech Choir, 1st Runner Up   Asia Pacific College, 2015 |
| **Seminars & Trainings Attended** | * The Fight Against Counterfeiting   Asia Pacific College, April 2018   * Sans Impunity: Forum   Asia Pacific College, December 2017   * Introduction to Ethical Hacking   Asia Pacific College, July 2016   * Microsoft Azure   Asia Pacific College, 2017 |
| **Extra-Curricular Activities** | * Junior Philippine Computer Society, Member, SY 2015-2016 * Explorer’s Club, Member, SY 2015-2016 |