Tribhuvan University

Faculty of Humanities and Social Sciences

A Project Report on

"Vehicle Rental App"

In partial fulfillment of the requirement for the degree of Bachelor in Computer Application

(BCA)

Submitted to:

Department of Computer Application

Kathmandu College of Technology

Submitted by:

Suresh Thapa (11051200)

Sworup Khatri (11051201)

2079/03/10

Under the Supervision of

Deepesh Ghimire

Tribhuvan University

Faculty of Humanities and Social Sciences

Kathmandu College of Technology

SUPERVISOR'S RECOMMENDATION

I hereby recommend that this project prepared under my supervision by "**Sworup Khatri** and **Suresh Thapa**" entitled "**Vehicle Rental App**" in partial fulfillment of the requirements for the degree of Bachelor of Computer Application is recommended for the final evaluation.

…….

SIGNATURE

Deepesh Ghimire

SUPERVISOR

Kathmandu College of Technology

Lokanthali Bhaktapur, 44600

Tribhuvan University

Faculty of Humanities and Social Sciences

Kathmandu College of Technology

LETTER OF APPROVAL

This is to certify that this project is prepared by "**Sworup Khatri** and **Suresh Thapa**" entitled "**Vehicle Rental App**" in partial fulfillment of the requirements for the degree of Bachelor in Computer Application has been evaluated. In our opinion it is satisfactory in the scope and quality as a project for the required degree.

……. **Mr. Deepesh Ghimire** **Supervisor** **Kathmandu College of Technology** **Lokanthali Bhaktapur, 44600**	**…….** **Mr. Prashant Gautam** **Head of Department** **Kathmandu College of Technology** **Lokanthali Bhaktapur, 44600**
……. **Internal Examiner**	**…….** **External Examiner**

STUDENT'S DECLERATION

We hereby declare that we are the only author of this work and that no sources other than those listed have been used in this work.

……..	……..
Sworup Khatri	Suresh Thapa
Date: 2079/4/26	Date: 2079/4/26

ACKNOWLEDGEMENT

We are very grateful to the department of computer application, Kathmandu College of Technology for providing us an opportunity to work on a major project as part of our Third year project. We are delighted to express our deep sense of gratitude and indebtedness to our learned supervisor Mr. Deepesh Ghimire, for his invaluable guidance, encouragement and even monitoring to spare time despite her busy schedule for project's progress reviews.

Our special thanks go to our colleagues and everyone who directly and indirectly extended their hands in making this project success.

Suresh Thapa

TU Exam Roll no: 11051200

Sworup Khatri

TU Exam Roll no: 11051201

ABSTRACT

A Vehicle rental, hire vehicle or vehicle hire agency is a company that rents automobiles for short period of time to the public, generally ranging from few hours to few weeks. Vehicle rental has been introduced to avoid traditional method of getting vehicle problem such as waiting for long time to get vehicle. It performs tasks such as booking a vehicle, adding a vehicle; maintain user's profile. This App has been developed using open source and free software which include Apache, PHP, MySQL, JAVA, XML, Firebase etc. This App is supposed to be beneficial for those people with less IT knowledge with the use of internet access because it is simple to use. With this App users were able to register and login into the app. The admin panel was used to add the vehicle. Each user has provided unique firebase ID. In this way, Vehicle rental allow users to request vehicle with the help of their Smartphone easily and efficiently.

Keywords: Vehicle, Entities, Users, Admin panel, firebase etc.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

DFD	Data Flow Diagram
ER Diagram	Entity Relationship Diagram
GUI	Graphical User Interface
JAVA	JAVA
MySQL	Structured Query Language
PHP	Hypertext preprocessor
FIREBASE	FIREBASE

CHAPTER 1
INTRODUCTION

1.1 Introduction

A Vehicle rental, hire vehicle or vehicle hire agency is a company that rents automobiles for short period of time to the public, generally ranging from few hours to few weeks. Traditionally people have to wait for the vehicle in the bus stop to get the vehicle facility and sometimes it could take more time so in such case people cannot perform their other important work.

The uses of internet have been widely used in various sectors which motivate us to use the internet and its technology in the field of vehicle rental sector. Hence to overcome such problem that leads to waiting or any inconvenience to the people we developed an android application for providing the service of rent a vehicle to the registered user in the application and this application is specially developed for long tour travels.

An application we have built is made by using free technology available on the internet and these technologies include apache server, MySQL, XML, Java, Firebase and PHP which performed operations like Creation of users, add a vehicle, book a vehicle, approved a vehicle easily and efficiently.

1.2 Problem Statement

Traditionally, people have to wait for vehicle in the bus stop and which did not provide a proper service to the people because of the over crowd in the public bus. The problem we been facing while using the traditional system is that these procedure has increased the waiting and the idle time of the people. This has made the people time to be unused. Vehicle rental application is developed in response to the problem.

1.3 Objective

The main objective was to develop an application is to rent a vehicle to the people in the accessible price. The design of Vehicle rental application performed following activities and function:

- To create a simple application that helps to rent a vehicle.
- To request for vehicle for rent easy and efficiently.
- To reduce time and cost of people.
- To provide better service to people.

1.4 Scope and Limitations

An application was designed in order to provide automation to the Vehicle rental system. Many people are backward toward the technology and still use traditional method to get the service of vehicle. This application emphasizes on providing facilities that are easily accessible to the users.

1.4.1 Scopes

Applications which have been built are able to perform various tasks such as user creation, add vehicle, request vehicle etc. and it scopes are listed below:

- Users can register or create new account through our application.
- Application can provide user friendly interface for the users such that any type of user can create an account and login through their credentials.
- User can feel easy, fast and reliable to use the Application.

The Vehicle rental application meets the above mentioned objectives and features in the earlier stages however, no any system is perfect. It was quite difficult to design a system with full accuracy and efficiency.

1.4.2 Limitations

The project allows creation of users, add a vehicle, and request a vehicle and reject a vehicle. However, there are some limitations which are listed below:

- Online Application: Application is only supposed through internet access. Without the access of the internet user cannot use the Application.
- User can only get a vehicle that has been added by admin.
- User has to wait for few minutes until requested vehicle has been approved by admin.

1.5 Report Organization

Chapter 1: puts brief emphasis on background and overview, problem statement, Objectives, Scope and Limitation of the project. Problem statement highlights on the problem for which the project was developed; the section is theoretical and research based. The project objectives, scope and limitation are then discussed.

Chapter 2: Defines and describes Background Study and literature review about existing system along with their advantages and disadvantages.

Chapter 3: Presents the System Analysis and Design including Requirement Analysis and Feasibility Analysis. Requirement Analysis explains Functional and Non-Functional Requirements of the project and Feasibility Analysis explains why/how the project is implemented.

Chapter 4: Presents the Implementation and Testing, it clarifies the system workflow and provides an indication of how the system is implemented, what tools/ platform have been used.

Chapter 5: Presents the Conclusion and Recommendation, it marks an end to the documents by summing up the entire project and by opening a door for future researches mentioning future enhancements that can be carried out on the system.

CHAPTER 2
BACKGROUND STUDY AND LITERATURE REVIEW

2.1 Background Study

Smartphone users are increasing daily. Since most of the people use Smartphone and user can easily rent a vehicle through internet access. This makes users to rent a vehicle anytime easily and efficiently. General concept and terminologies are mentioned below:

1. **Add vehicle:** Admin can add a vehicle after logged in into the system.
2. **Request Vehicle:** users can request a vehicle added by admin.
3. **Approve vehicle:** Admin can approve a vehicle requested by user.
4. **Deny vehicle:** Admin can reject a vehicle requested by user.

2.2 Literature Review

As our structure relies upon the useful Car Renting System which is an authentic application we inspected the present working circumstance of the renting technique. At present renting, organizations are dependent on manual work which consolidates packages of work area work similarly as a human resource. To date we find Cab Services incredibly easy to book, pay, or drop as they have formed their structures into helpful applications similarly as locales. So there is a need to change the arrangement of the Car Renting Service. But, Car rental business, notwithstanding everything, uses the central methodology for Renting a vehicle to a customer asThe customer ought to go genuinely at center, the owner will similarly be accessible there and the owner will permit the vehicle with his/her own supported driver [1].

With the high speed development of car rental market, the traditional manual rental management is not enough for the various business information processes. Therefore, this study proposed an enterprise-class development program of car rental management system based on AJAX+SSH and this program will be applied in this development of the car rental management system. This article introduced the accomplishment of the functional module and the design of the whole structure. Practice proved that this program can not only greatly increase the developing

efficiency of this system but also reduce the difficulty; meanwhile, it can raise the response efficiency of Web application system and the effect of user experience [2].

An industry adage held that "there are two types of rental car companies: those that lose money and Enterprise." The company that would become Enterprise Rent-A-Car was started in 1957 in St. Louis, Missouri, by Jack Taylor. Taylor set up Enterprise offices in neighborhoods rather than at airports because he believed that Americans would welcome a local option for renting cars when their own vehicles were being repaired. In 2010 Enterprise had more than 6,000 rental locations in the United States and a fleet of 850,000 cars in service. Its parent, Enterprise Holdings (comprising Enterprise, National, and Alamo brands) accounted for nearly half of the car rental market and was more than twice the size of Hertz, the number two competitor. Enterprise's competitive advantage was the result of the combination of its practices in hiring, training, compensation, organization, customer service, IT, and fleet management, among others [3].

In this twenty first century car rental system it are famous using to tourism Web-based system understands as server components of distributed applications which use the HTTP protocol to exchange data between servers and clients(browser). By this definition, the principal problem of Web-based system development becomes apparent from business perspectives [4].

The car rental company EU-Rent has 1000 renting branches and 200,000 cars in towns all over Europe. At each branch cars, classified by car group, are available for rental. Each branch has a manager and booking clerks who arrange rentals and reservations for future rentals. Cars rented from one branch of EU-Rent may be returned to any other branch. The rental period and the car group are specified at the time of reservation. The renting branch must ensure that the car has been returned to some branch at the end of the rental period. If a car is returned to a branch other than the one that rented it, the car is assigned to the new branch. At the end of each day cars are assigned to reservations for the following day. If no car of a requested group is available, a branch manager may ask other nearby branches if they have cars they can transfer to him [5].

CHAPTER 3
SYSTEM ANALYSIS AND DESIGN

3.1 System Analysis

Considering the fact that this application involves design and implementation of a software system regardless that is android-based, it was necessary to mention and consider certain models used in software development and deployment, including the following generic software development models:

i. The Waterfall approach:

It represents activities in requirements, specifications, design, implementation and testing. All these as separate processes.

ii. Incremental / Evolutionary development:

It involves a rapid development of the specifications and then refined later for the customer.

iii. Formal transformation:

This approach is based on setting and producing a mathematical specification for the system to meet.

iv. System assembly from reusable components:

This method is based on the assumption that a portion of the system already exists. This model emphasizes integration.

After going over all of these models, I've come to a conclusion. The incremental/evolutionary model is appropriate for this android app development. It has had extremely clear iterations, especially considering that we are just two developers working on the project. As a result, we can concentrate on each component of the model during development and return to it if necessary. Based on this concept, the project may be readily broken down into distinct pieces.

The Evolutionary / Incremental Development model for software development

This is the model through which the vehicle rental app was created. During the software development process, however, feedback loops were there.

Software development should be interleaved, according to the evolutionary / incremental approach. The user's feedback was incorporated throughout the entire process. Several versions of the software package were developed. It necessitates software development follows the following stages:

- Analyses of the problem is made and requirements to be meet are proposed,
- The specifications of the project are put in place after the problem has been analyzed and the requirement to be meet are made clear,
- A design for the software system is made,
- After the design is completed the process of implementation and debugging starts,
- The program is tested and integration of the system is made,
- The developed software is t in its final stage and operation and maintenance can then be made continuously

This process has been illustrated below on Figure showing top-down development. The main stage description is in the boxes. The model looks like a waterfall model. For this model to work efficiently feedback loops are added to create a coordination of the whole process development.

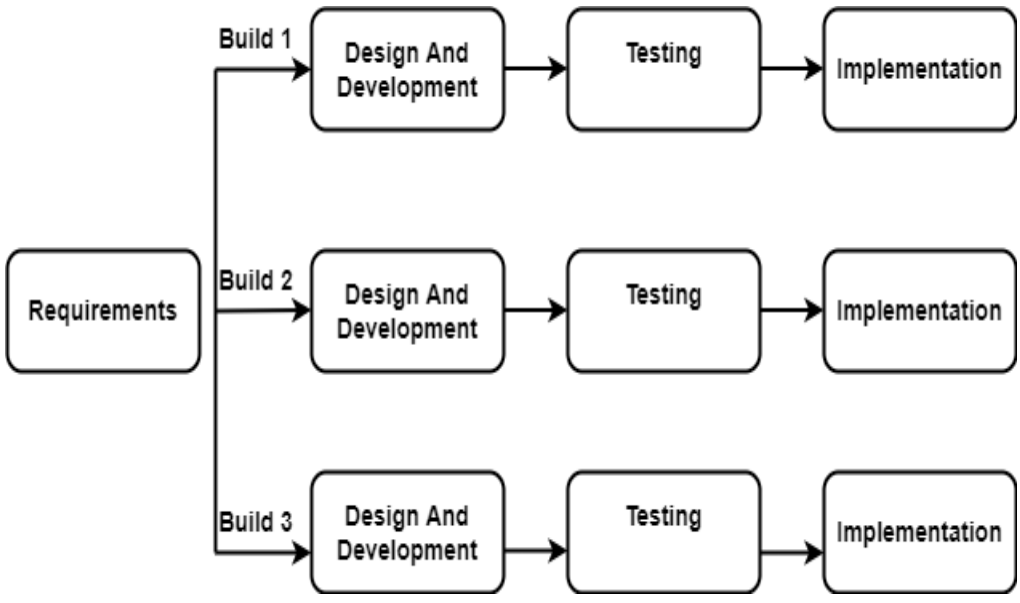

Figure3. 1 Incremental Development Model

3.1.1 Requirement Analysis

Requirements analysis is a crucial step for determining the success of a system or software project. Requirements are generally split into two types:

i. Functional requirements

ii. Non-functional requirements

i. Functional Requirement

This section provides the requirement overview of the application. Various modules implemented by the system are:

- **User Module**

➢ Users can register and login the system.

➢ Users can request a vehicle available in the application.

➢ Users can see the vehicle request after approved by admin

➢ Users can see the profile.

➢ Users can logout from the system after completion of transaction.

- **Admin Module**

➢ Admin can login the system.

➢ Admin can see the registered users.

➢ Admin can add the vehicle.

➢ Admin can logout from the system.

- **Login module**

➢ Only registered user can login the system.

➢ It ensures security to the system.

➢ It helps to authenticate the users through firebase.

➢ Only validate email and password is used to login the system.

ii. **Non –Functional Requirement**

Non-functional requirements of the system are identified as Efficiency Requirements, Reliability Requirements, Usability Requirements, and Implementation Requirements. The non-functional requirements included in the project are:

- **Efficiency Requirement**

The efficiency of a software system refers to how well it handles capacity, throughput, and response time. With the deployment of the vehicle rental app, both users and administrators may easily access the app and users were able to perform request a vehicle.

- **Reliability Requirement**

The degree to which the software system regularly executes the stated functions without failure is referred to as reliability. User registration, login, transaction detail, and balance inquiry were all performed accurately by the system.

- **Usability Requirement**

The system's usability criteria state how simple it must be to use. The system was created in a user-friendly environment so that users and administrators could easily and successfully complete various activities in the system.

- **Implementation Requirement**

The process of turning strategies and plans into actions in order to achieve strategic objectives and goals is known as implementation. The frontend was created using XML, with PHP and Java serving as the server-side programming language for database connectivity at the backend, i.e., MYSQL and Firebase was utilized to develop the database.

3.1.2 Feasibility Analysis

i. Technical Feasibility

These include hardware, software and technologies. The suggested system is technically possible because it requires access to the use of a Smartphone and the internet. The system's user interface is also quite simple.

ii. Operational Feasibility

Reliability, maintainability, usability, and supportability are among them. The suggested system is operationally practical since it is reliable for all types of users, regardless of whether or not they are computer literate. For a small to large-scale organization, the proposed system is supported. It is simple and straightforward to use.

iii. Economic Feasibility

The project was developed within the organization's budgetary constraints. The project was resource was freely available, and no additional obligations are required. The creation of the system does not necessitate the use of expensive hardware or software. The platform is open sources and the resources required for the project are also open source.

iv. Schedule Feasibility

Among various phases of the project data collecting took longer time as data were collected from different rental system and its working mechanism. After the data is acquired, the next development phase can be completed in as little as a month.

GANTT CHARTS

A Gantt chart is a form of bar chart that shows the progress of a project. A Gantt chart, which is widely used in project management, is one of the most popular and useful methods for displaying activities against time. It can also be used to examine a project's start and finish dates in a single graph. Gantt charts were created in our project using Microsoft Excel, as seen in the picture below.

Task	Start Date	Days to finish	End Date
Requirement Analysis	15-May-22	7	22-May-22
Design	22-May-22	6	28-May-22
Development	28-May-22	14	11-Jun-22
Implementation	11-Jun-22	6	17-Jun-22
Maintainance	17-Jun-22	4	21-Jun-22

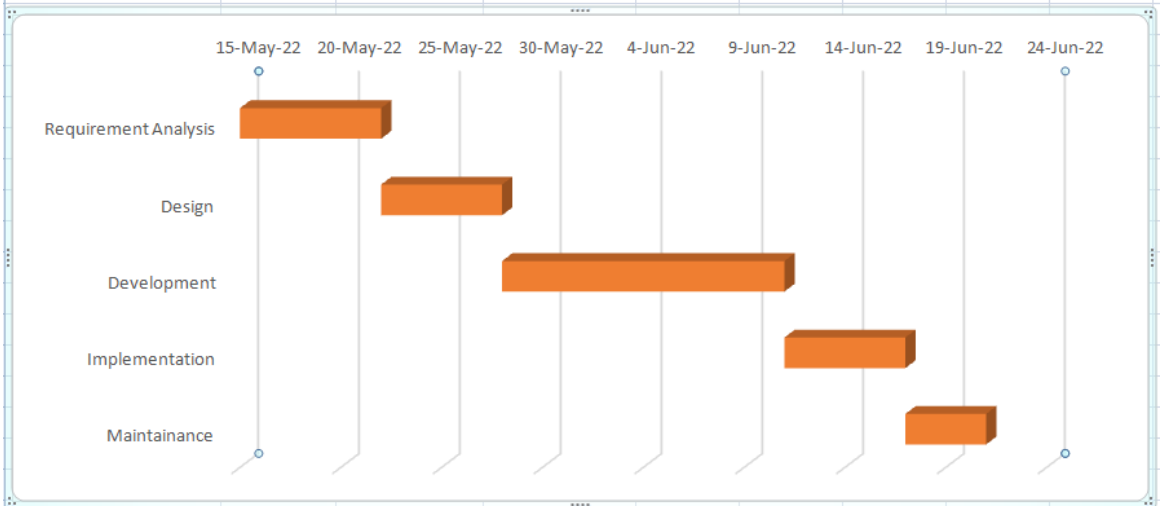

Figure3. 2 Gantt chart of Vehicle Rental App

3.1.3 Data Modeling(ER-Diagram)

This ER (Entity Relationship) Diagram represents the model of "Vehicle rental Application". The entity-relationship diagram of "Vehicle rental Application" shows as all the visual instrument of database tables and the relations between Admin, Users, and list of vehicle. It used structure data and to define the relationships between structured data groups of "Vehicle rental Application" functionalities. Database system contains users which contain a primary key as a unique identifier for each entity and other attribute to show the properties of these entities. ER diagram of Vehicle rental contains the three entities such as user, vehicle, and admin where user entity have different attribute such as user-id, full name, image, email, password and another vehicle entity have vehicle name, vehicle rating, vehicle image, vehicle detail and vehicle category and another entity admin have user-id, full name, image, email, password.

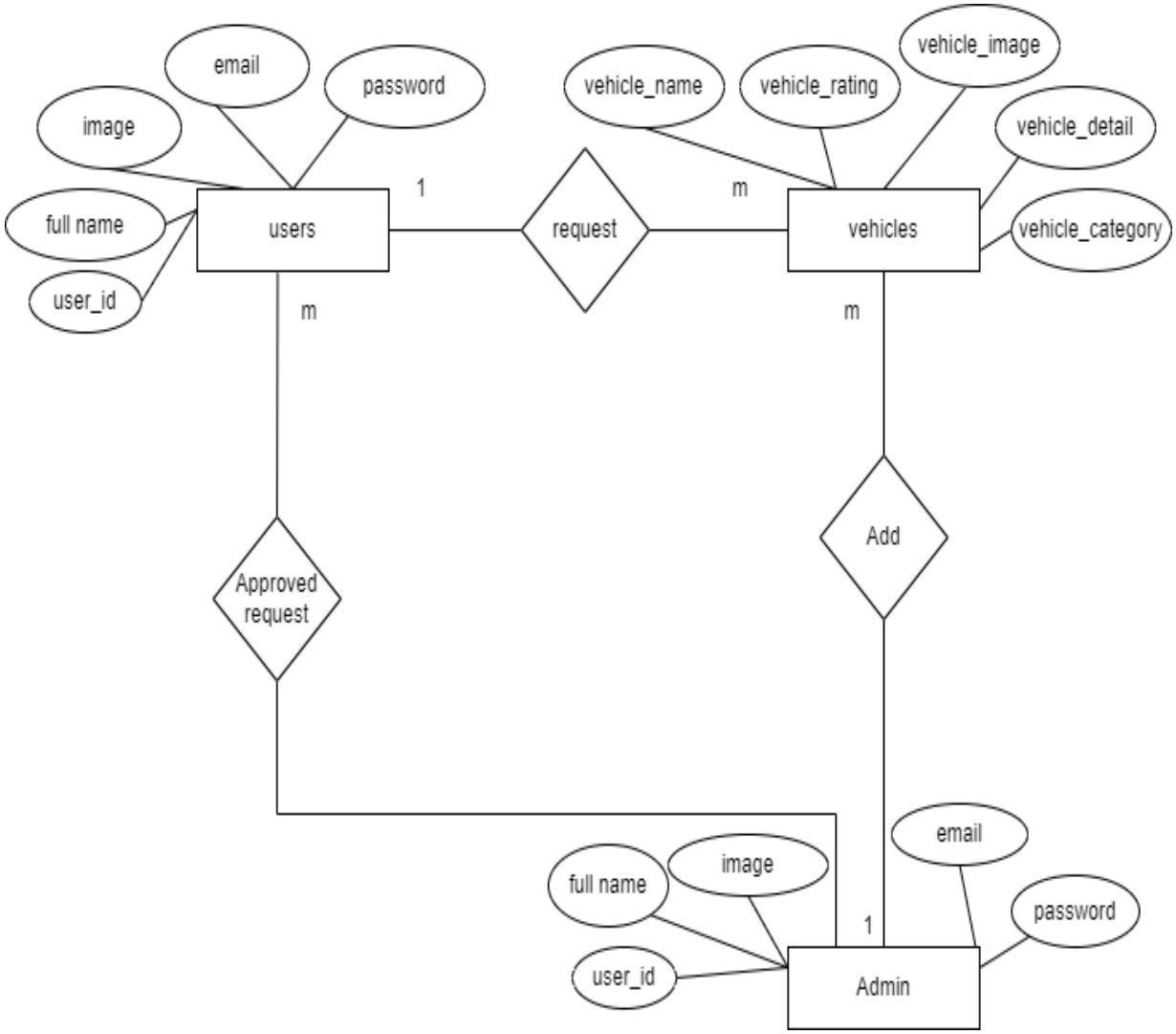

Figure3. 3 ER Diagram (Vehicle Rental App)

3.1.4 Use Case Diagram

This Use Case Diagram is a graphic depiction of the interactions among the elements of Vehicle Rental App. It represents the methodology used in system analysis to identify, clarify and organize system requirements of Vehicle Rental App. There is admin and users where admin perform various tasks like adding new vehicle approving and rejecting a vehicle and record all the transaction and on the other hand users perform various task like register and login in the system and can request for the available vehicle, users also can approve and reject a vehicle.

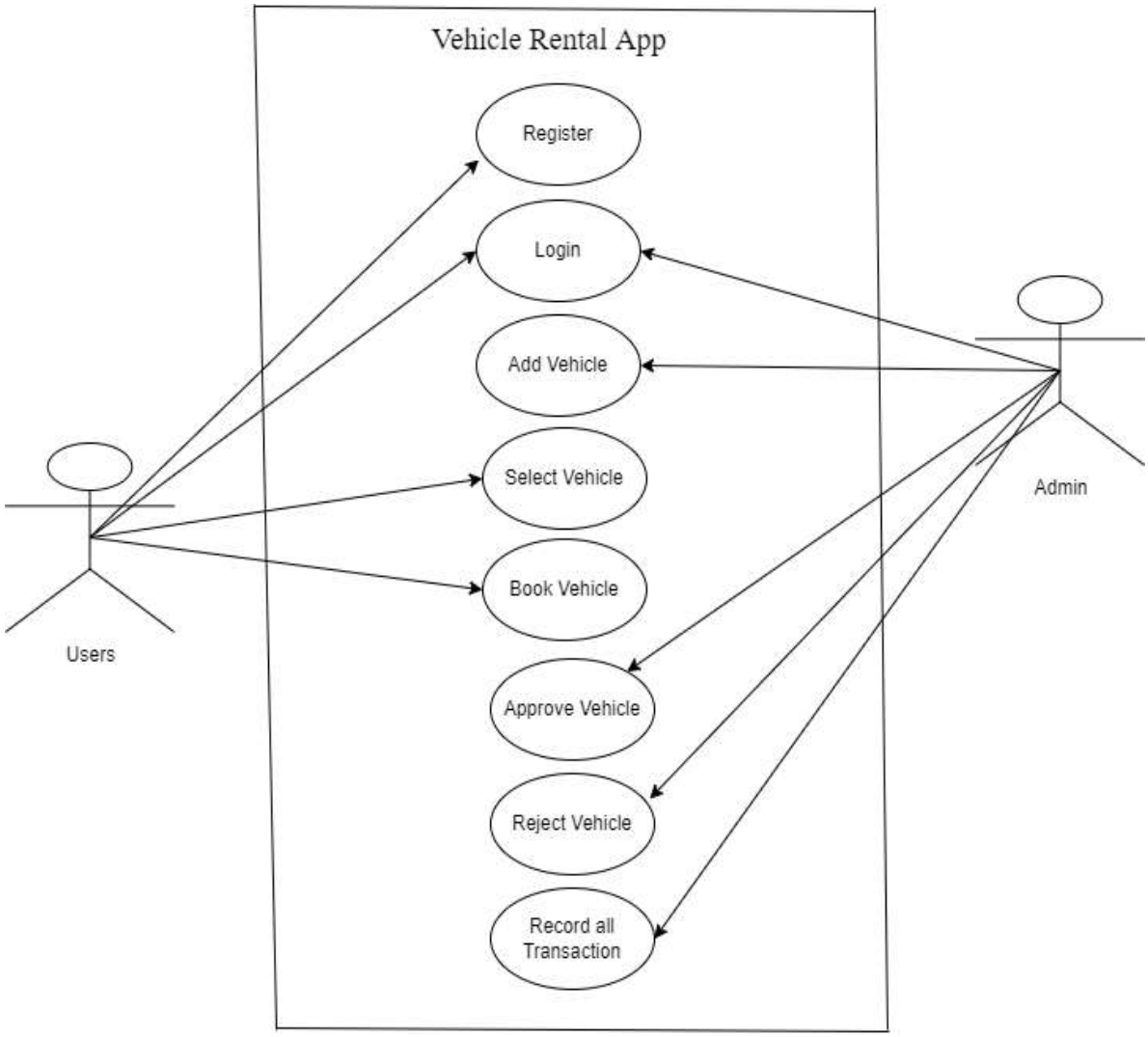

Figure3. 4 Use Case Diagram of vehicle Rental App

3.1.5 Process Modeling (DFD)

A process model describes the flow of work or activities, usually in a graphic format, that contribute to accomplishing a specific goal. Process models are typically used to represent and analyze a series of activities that occur repeatedly and on a regular basis.

A data flow diagram (DFD) is a graphical or visual representation using a standardized set of symbols and notations to describe a business's operations through data movement.

Context Level DFD (0 Level)

The 0 Level DFD shows flow of data of application. DFD Level 0 is also called a Context Diagram. It's a basic overview of the whole system or process being analyzed or modeled.

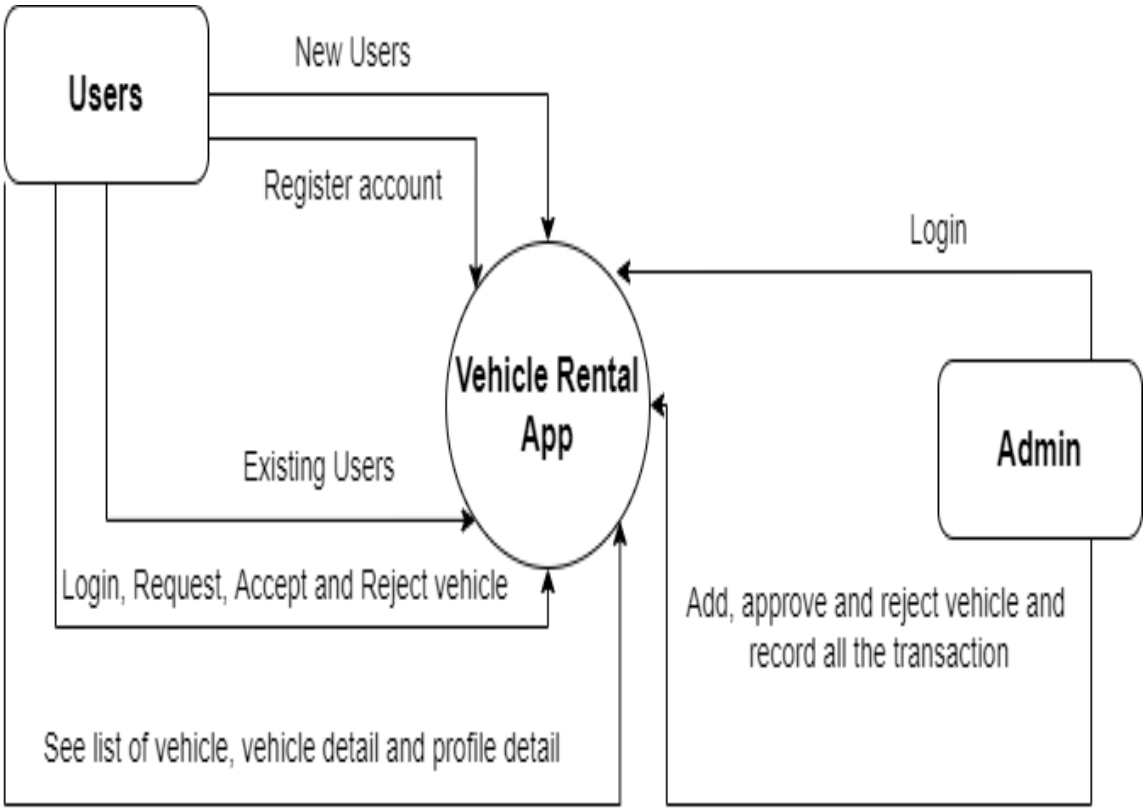

Figure3. 5 context diagram of Vehicle Rental App

DFD Level One

DFD Level one provides a more detailed breakout of pieces of the Context Level Diagram. This DFD describes main functions carried out by the system, as we break down the high- level process of the context Diagram into its sub-processes.

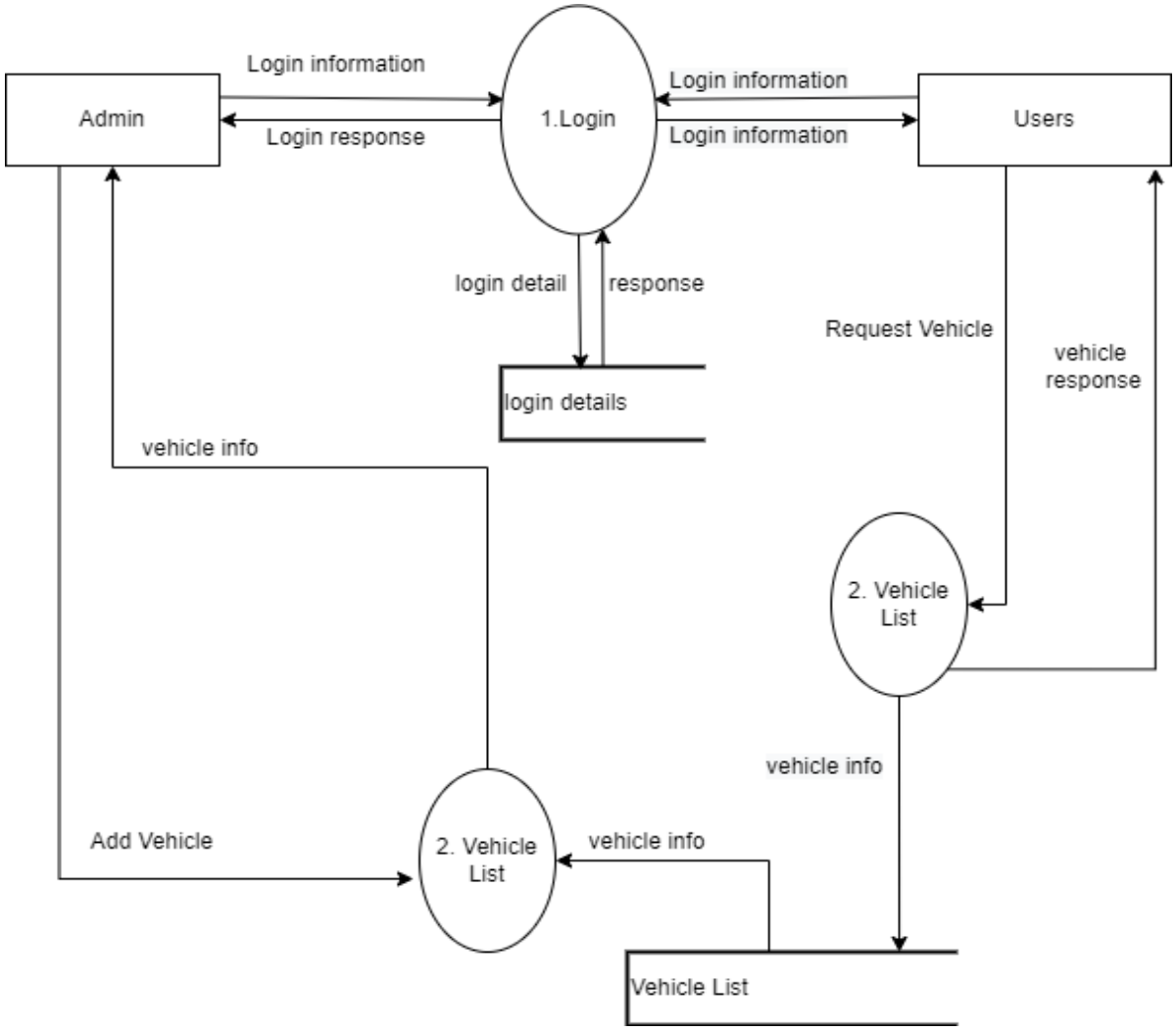

Figure3. 6 level One DFD (Vehicle Rental App)

3.2 System Design

3.2.1 Architectural Design

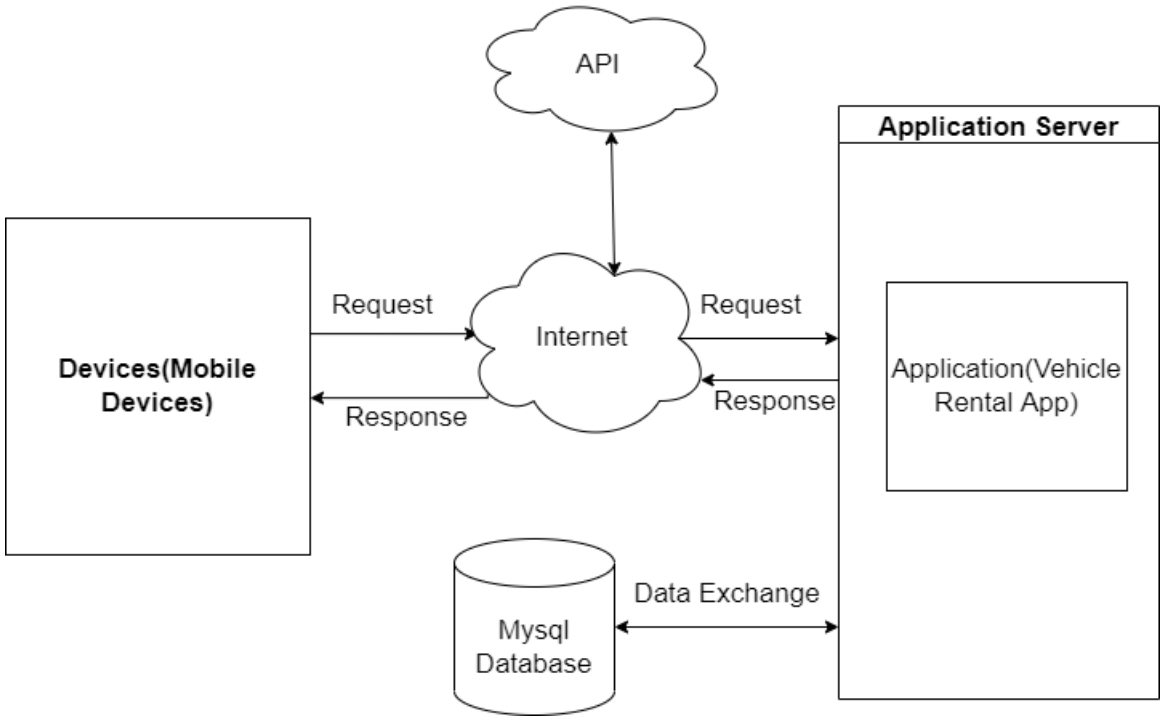

Figure3. 7 Architecture of Vehicle Rental App

3.2.2 Interface Design (UI Interface / Interface Structure Diagrams)

A few user interface designs were created before the actual design of the project is implemented to visualize the user interaction with the system as they explore registration, login, and add vehicles services. Our Functional Decomposition Diagram, which shows the early designs of the activity, would be closely followed by the user interface design. Some of the user interface designs are done in figma and they are shown below

Figure3. 8 Register page Design (Vehicle Rental App)

Figure3. 9 Login page Design (Vehicle Rental App)

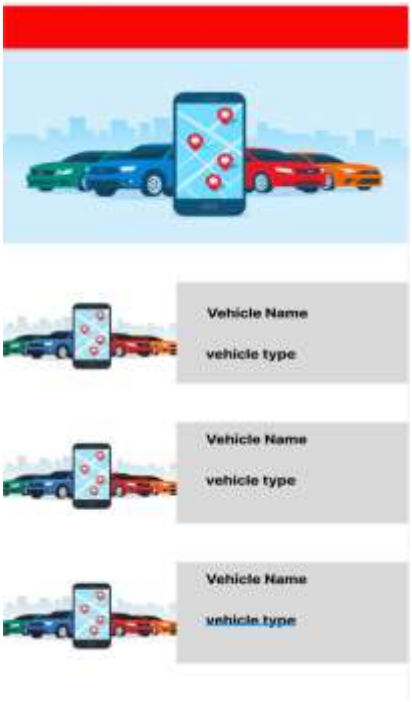

Figure3. 10 User home page Design (Vehicle Rental App)

Figure3. 11 Vehicle Detail Page design (Vehicle Rental App)

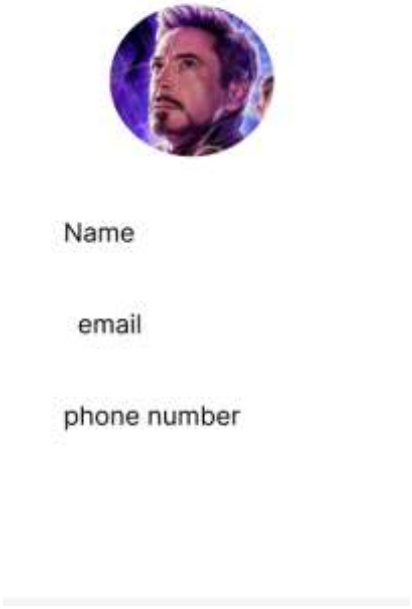

Figure3. 12 Profile page Design (Vehicle Rental App)

Figure3. 13 user Approved page design (Vehicle Rental App)

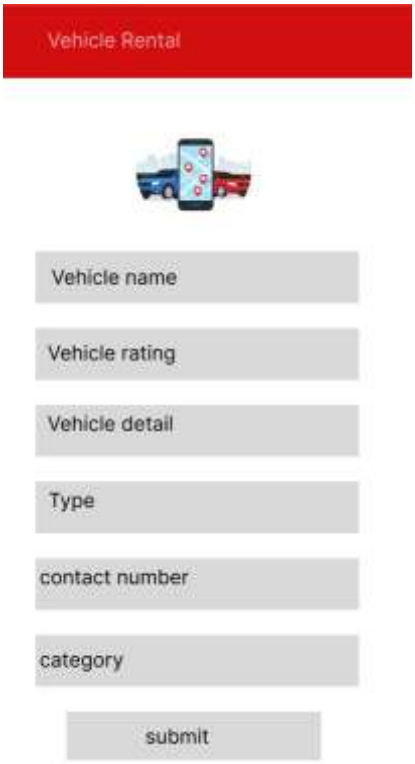

Figure3. 14 Admin home page Design (Vehicle Rental App)

3.2.4 Physical DFD

Physical DFD goes one step deeper than level one DFD where user account is created when user enter all details correctly and user data is stored in the firestore database. User could have logged in to the app after successfully creation of an account and could have requested the vehicle. Likewise, admin could have logged to the app by entering admin detail correctly and can add a vehicle and approve and reject the vehicle requested by users.

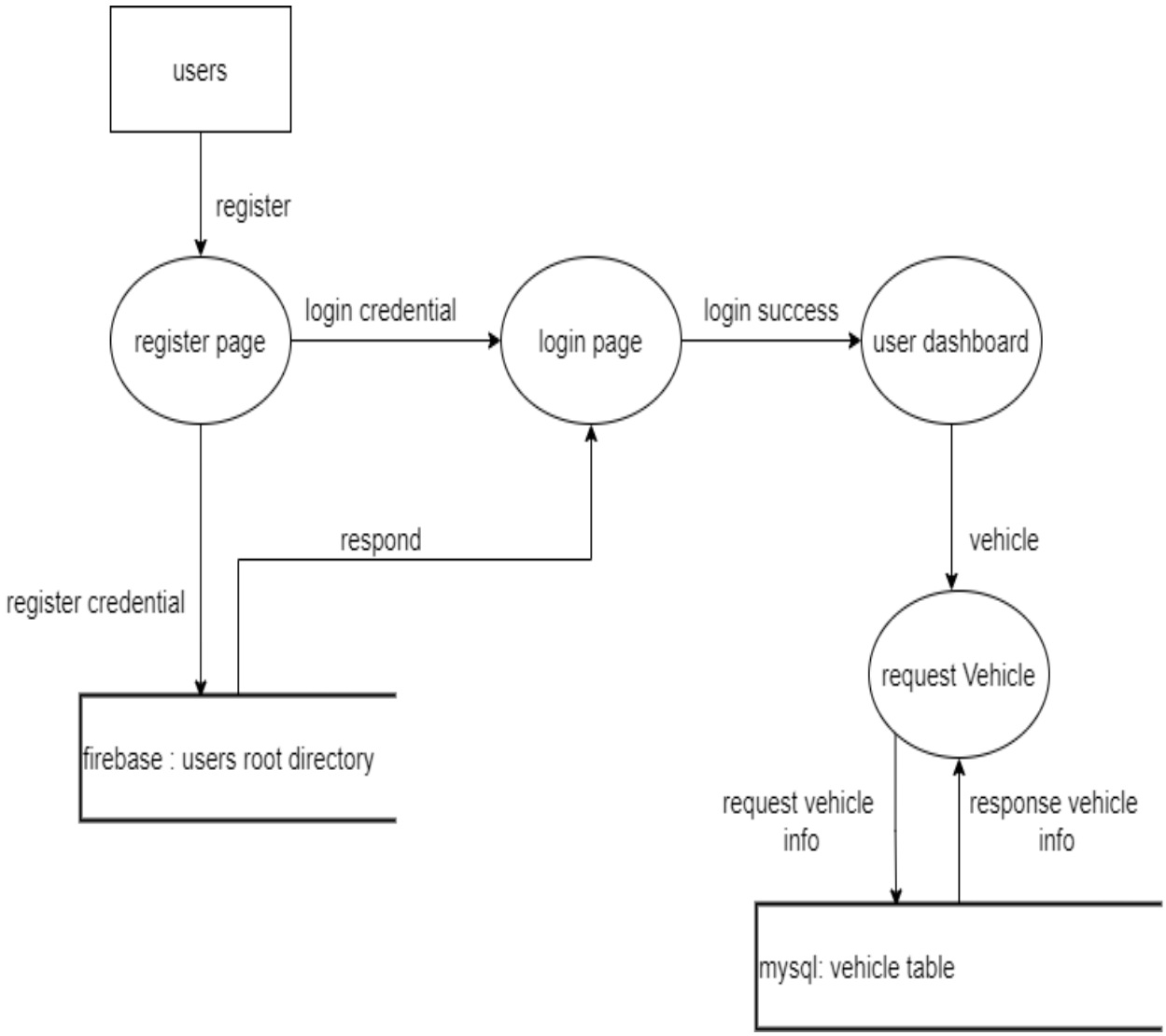

Figure3. 15 Physical DFD (Vehicle Rental App)

CHAPTER 4

IMPLEMENTATION AND TESTING

4.1 Implementation

In first phase, data were collected. Data collection took longer time than other phase. It was the critical stage in project's development. All the physical design of the project is turned into working Android code. Many tools and technologies that were utilized to develop the system were discussed in the preceding chapter.

4.1.1 Tools Used

The various system tools that have been used in developing both the front-end and back-end of the project are being discussed in this chapter.

CASE tools:

a) Draw.io
b) LucidChart.com

Client side:

XML is used for developing the front-end.

XML (Extensible Markup Language)

XML tags define the data and used to store and organize data. It is used to implement UI-related data in android and it doesn't make layout heavy.

Server Side:

The backend is implemented using Java, PHP, MySQL and firebase. MySQL and firebase is used to design the database.

PHP

It is used to make an API request.

JAVA

It is used to handle android data and UI related components programmatically.

MySQL and Firebase

It is mainly used for the purpose of database.

XAMPP

XAAMP is used for local server and database to fulfill the need of the project and Apache and MySQL is used as local server and database.

4.1.2 Algorithms

Admin Login Algorithm

Start

Enter admin login details

IF(details valid)

    Allow login and redirect to Home

ELSE IF(details not valid)

    Check if username and password exist in database

IF(detail do not exist)

    Return a Toast message

User Login Algorithm

Start

Enter user login details

IF(details valid)

    Allow login and redirect to Home

ELSE IF(details not valid)

    Check if username and password exist in database

IF(detail do not exist)

    Return a Toast message

Greedy Algorithm

Greedy algorithm is used to accept and Reject a vehicle by users in the system. Greedy algorithm is an algorithmic strategy that makes the best optimal choice at each small stage with the goal of eventually leading to a globally optimum solution. This means that the algorithm picks the best solution at the moment without regard for consequences. It picks the best immediate output, but does not consider the big picture, hence it is considered greedy.

Working Principle:

- A greedy algorithm is an approach for solving a problem by selecting the best option available at the moment.
- The algorithm never reverses the earlier decision even if the choice is wrong. It works in a top-down approach.
- This algorithm is straightforward and can perform better than other algorithm but, not in all cases.

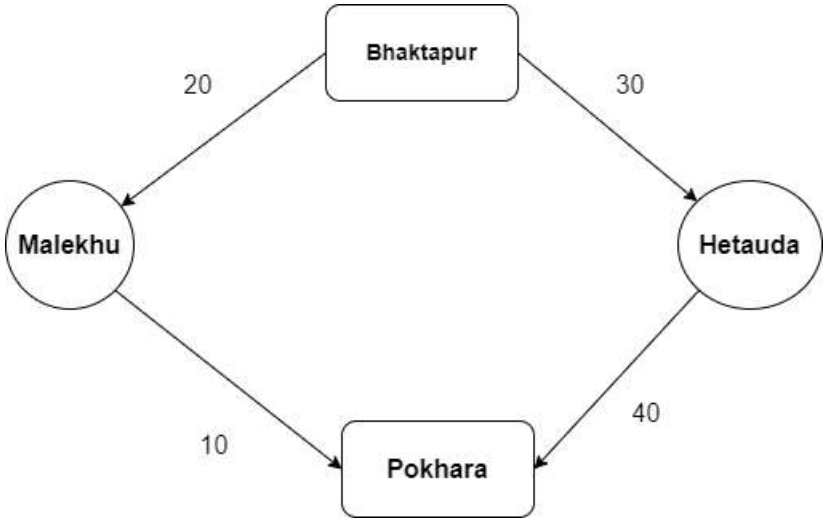

Figure 4. 1 Greedy Algorithm

Algorithm:

- Start from a minimum distance to reach the destination.
- This algorithm chooses the best optimal choice at the moment.
- While going to the destination if user didn't like the price added by an admin then user can reject the vehicle and request for another vehicle.

4.1.3 Implementation Details of Modules

The proposed system is composed of different module such as user module, admin module, and login module. In user module user can register and login the system and can make vehicle request and can get logout from the system likewise admin also can login the system and approve the vehicle request made by user and can logout from the system and in login module only registered user can login the system.

User Module:

In user module account is created by filling the register form detail which includes the field like name, email address, phone number and password and user image. While filling the input field user must fill the all data in the input field so that it would not throw an error message. User data is stored in the firestore root directory after filling correct details in the registration form while creating an account. After successfully creating an account user can login to the system.

Admin Module:

In Admin Module authentication is done using email and password from firebase authentication given to the admin if admin enter correct email and password then admin can access to his dashboard. Admin can add vehicle in the system and can approve and reject the vehicle request made by users and admin can logout of the system by clicking logout button.

Login Module:

In Login Module user can login to the system after successfully creating an account. Login module consists of two field such as email field and password field. User is only logged in to the system when email and password entered by user is matched with firebase authentication email and password. In this module user can login through email and password. User must enter correct email and password to login into the system. If user enters wrong email and password, then it throws an error message and in order to login in the system user must enter correct email and password.

4.2 Testing

On the basis of the software requirement specification document, testing was performed to investigate and validate the behavior of a fully integrated software product. Before deploying an application or website, it must be thoroughly tested. As a result, this application's test cases were written. Some of the types of testing that we did are described below.

4.2.1 Test Cases for Unit Testing

Table 4.2. 1 Admin Login with Valid data

ID	Test Case Description	Test Data	Expected Result	Actual Result	Pass/Fail
T01	Check Admin Login with Valid data	email: subhadra@mail.com password: subhadra123	Logged into admin page	As expected,	Pass

Table 4.2. 2 Admin Login with Invalid data

ID	Test Case Description	Test Data	Expected Result	Actual Result	Pass/ Fail
T02	Check Admin Login with Invalid data	email: asdf@mail.com password: krishnaji123	Display Toast Message *Email or password is incorrect*	As expected	Pass

Table 4.2. 3 User Login with Valid data

ID	Test Case Description	Test Data	Expected Result	Actual Result	Pass/Fail
T03	Check user Login with Valid data	email: tony@mail.com password: ironman	Logged into users page	As expected,	Pass

Table 4.2. 4 User Login with Invalid data

ID	Test Case Description	Test Data	Expected Result	Actual Result	Pass/Fail
T04	Check user Login with Invalid data	email: abc@mail.com password: shreekrishnaji	Logged into users page	As expected,	Pass

Table 4.2. 5 User Register with complete Data

ID	Test Case Description	Test Data	Expected Result	Actual Result	Pass/Fail
T05	User enters all the details correctly and successfully	Image: image url Name: yushika sigdel Email: yushika@mail.com Phone: 9823803531 Password:yushika123	Display Toast message Users Created	As expected	Pass

Table 4.2. 6 User Register with Incomplete Data

ID	Test Case Description	Test Data	Expected Result	Actual Result	Pass/Fail
T05	User try to register into an application without an image	Image: Name: yushika sigdel Email: yushika@mail.com Phone: 9823803531 Password:yushika123	Display Toast message *please select user Image*	As expected	Pass

4.2.2 Test Case for System Testing

Check system behavior,

- If the Android activity launches properly with all the relevant pages, features and logo.
- If the user can register/login to the Android.
- If the main features, such as adding vehicle, requesting vehicle and approving requested vehicle and so forth, function as expected.
- If a user is satisfied with the android app after utilizing it, or if the user does not find it difficult to utilize it.

CHAPTER 5
CONCLUSION AND FUTURE RECOMMENDATION

5.1 Lesson Learnt / Outcome

With the completion of the project, it was possible to achieve the project's goal. After filling the register form, user can view and perform different task online through android application. In this way user can save time and perform task from this app.

5.2 Conclusion

A Vehicle rental, hire vehicle or vehicle hire agency is a company that rents automobiles for short period of time to the public, generally ranging from few hours to few weeks. Traditionally people have to wait for the vehicle in the bus stop to get the vehicle facility and sometimes it could take more time so in such case people cannot perform their other important work. Vehicle rental app is successfully implemented using XML, Java, PHP, MySQL, Firebase which are open source and freely available on internet and it successfully solve the problem of traditional method of getting vehicle. The proposed system is useful for long tour travel and speople with minimal IT knowledge with the use of internet. Towards the end of the project it was discovered that the application might benefit from a number of improvements. Some of these suggestions came from the app's testers, while others came from both of us. Any more enhancements to the application can be made during future development.

5.3 Future Recommendation

There are many things that can be added in future to improve this app such as user experience, and portability. There is more to be done, thus this application can be seen of as a launching pad for something bigger to come. All of them will need more time and resources to complete, but they are still highly realistic and achievable goals.

- Making a good user profile,
- Greater user experience,
- Making good UI design

REFERENCES

[1] C. D, "Car Rental System," https://ijirt.org/master/publishedpape, 2010.

[2] A. O. I. Onashoga, "Design and realization of a car rental management system based on AJAX+ SSH.," https://doi.org/10.1080/20421338.2016.1156837, 2016.

[3] M. B. M. S. J. S. J. Busse, "Developement of car Rental Management Information System," https://doi.org/10.1108/case.kellogg.2016.00011, 2017.

[4] Kaiser, "Car Rental System," epicessayhelp, 2000.

[5] kinoe, "Web Based car rental system," termpaperwarehouse, 2013.

APPENDIX

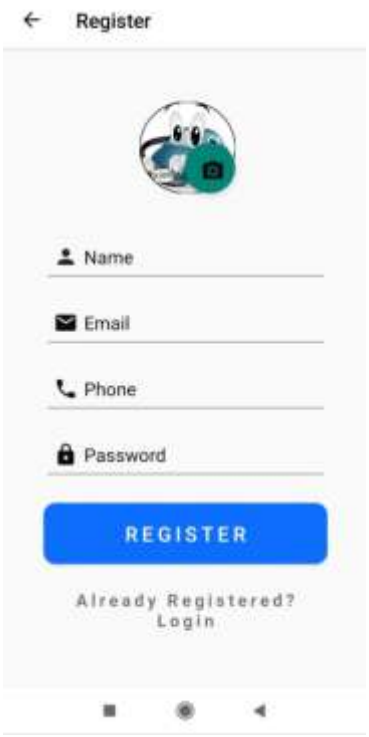

Figure A 1 Registration activity (Vehicle Rental App)

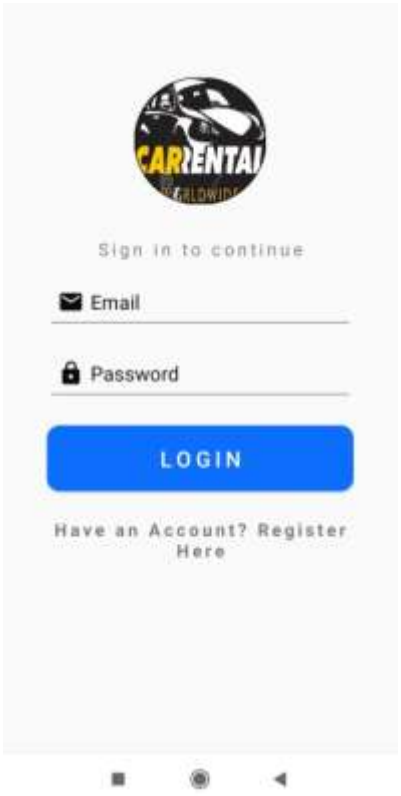

Figure A 2 Login activity (Vehicle Rental App)

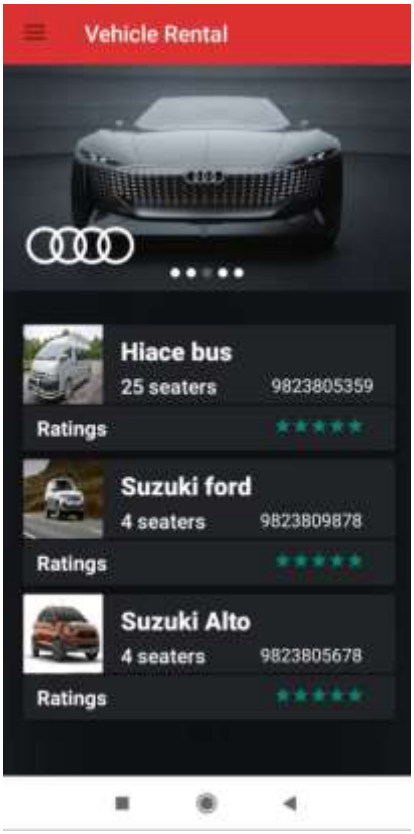

Figure A 3 User activity (Vehicle Rental App)

Figure A 4 Vehicle Detail activity (Vehicle Rental App)

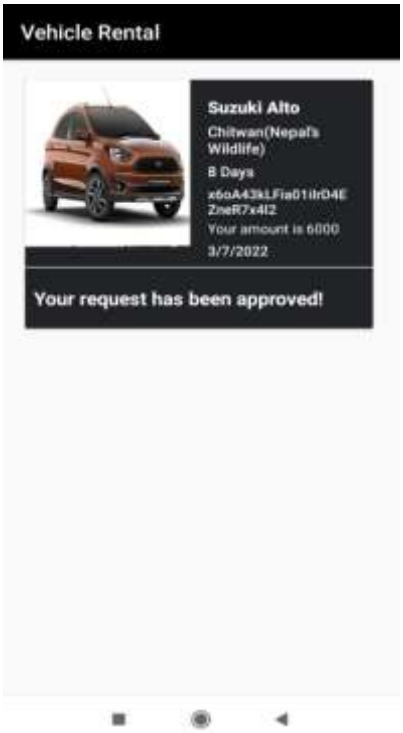

Figure A 5 user approved vehicle list activity (Vehicle Rental App)

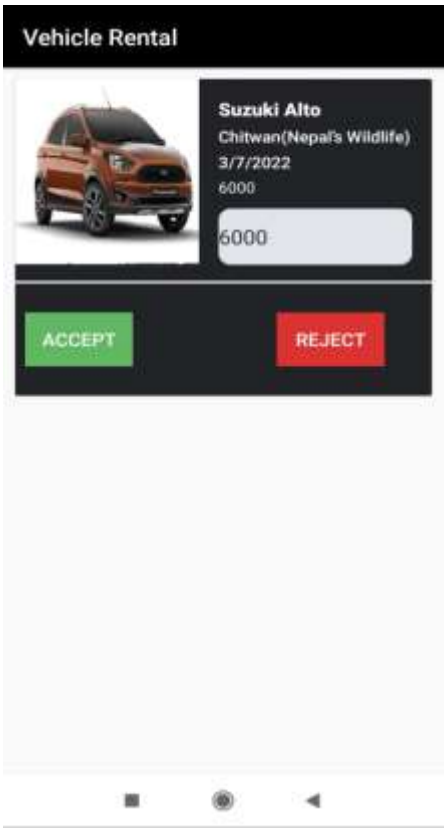

Figure A 6 user vehicle accepting and rejecting activity (Vehicle Rental App)

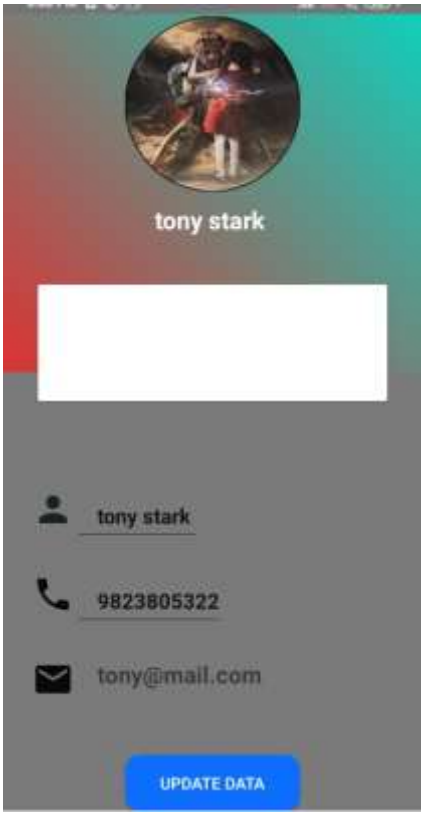

Figure A 7 User profile Activity (Vehicle Rental App)

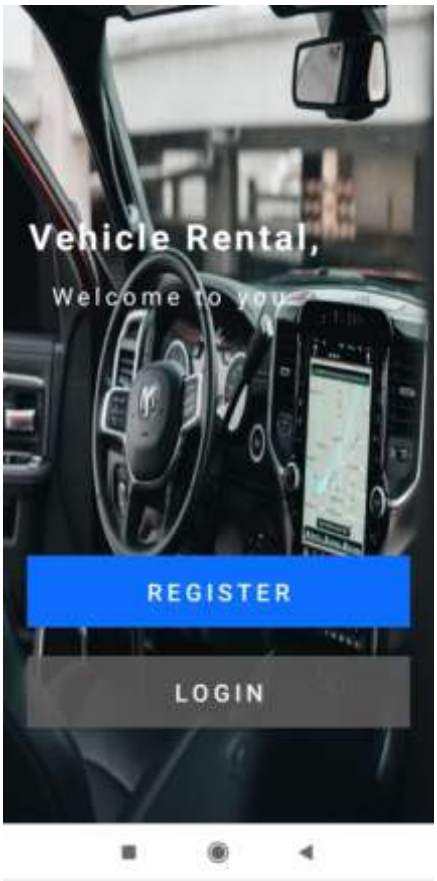

Figure A 8 Home Activity(Vehicle Rental App)

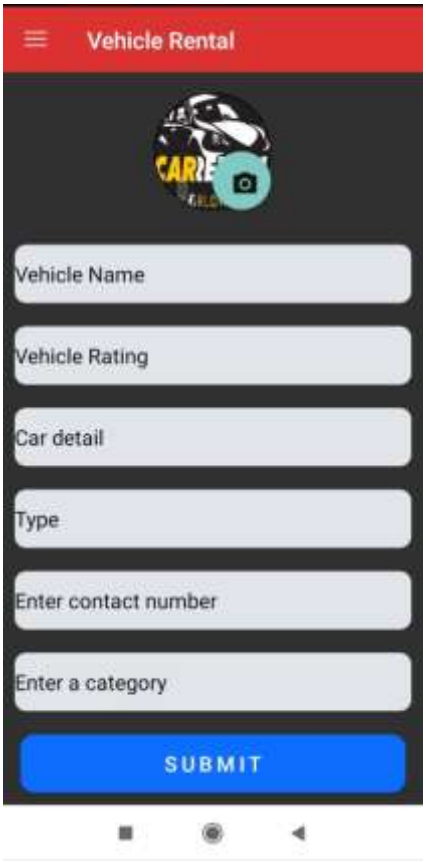

Figure A 9 Admin Activity (Vehicle Rental App)

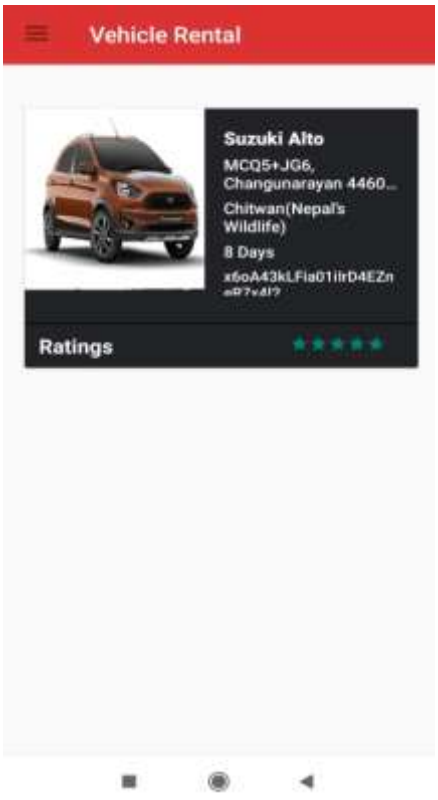

Figure A 10 Request vehicle list in Admin Activity (Vehicle Rental App)

Figure A 11 Approving and rejecting vehicle in Admin Activity (Vehicle Rental App)