Masters Project Final Report

December 2017

Project Title	**Open Source Vehicle Reservation System for Company vehicle pool**			
Student Name	**Ms.R.M Vidanapathirana**			
Registration No. & Index No.	**2014/MIT/065** **14550656**			
Supervisor's Name	**Mr. G.K.A Dias**			
Please Circle the appropriate	**Master's Program**		**Type**	
	MIT	**MCS**	**Research**	**Implementation**

For Office Use ONLY

Open Source Vehicle Reservation System for a company vehicle pool

R.M Vidanapathirana

2017

Open Source Vehicle Reservation System for a company vehicle pool

Dissertation submitted for the Degree of Master of Information Technology

R.M Vidanapathirana

University of Colombo School of Computing

2017

Declaration

The thesis is my original work and has not been submitted previously for a degree at this or any other university/institute.

To the best of my knowledge it does not contain any material published or written by another person, except as acknowledged in the text.

Student Name: R.M Vidanapathirana

Registration Number: 2014/MIT/065

Index Number: 14550656

_____

Signature: Date: 14/08/2017

This is to certify that this thesis is based on the work of

Ms. R.M Vidanapathirana

Under my supervision, the thesis has been prepared according to the format stipulated and is of acceptable standard.

Certified by:

Supervisor Name: Mr.G.K.A Dias

_____

Signature: Date: 14/08/2017

Abstract

Each and every mechanism is automated in this new technological era. Many companies transfer their manual processes into automated computerized processes because of the easiness and the reliable use of them.

The main problem addresses in this project is to give an automated solution for a company who is having a large vehicle pool and a manual process to allocate those vehicles. When this task is done manually, employees of the company need to submit a requesting form manually or need to contact the vehicle pool for allocations .This is not an easy task, because the employees need to fill a form and fax it or go and give it to the vehicle pool. On the other hand, the admin of the vehicle pool has to collect the forms manually and need to allocate them manually and the records should be kept manually. Storage of these manual records is another massive problem and it is hard to recheck of these records in future.

This project introduced a sophisticated system for requesting vehicles, scheduling vehicle reservations, maintaining driver details, maintaining vehicle details and of course keeping records of the history of reservations. In brief, an employee can request a vehicle by filling form through the system and when the request is sent to the admin, employee gets an email with the allocation details. Admin can receive requests and he can allocate pool vehicles and also outside vehicles if pool vehicles are not available .Admin can manage driver details and also vehicle details and he can get reports if need. Drivers also get a login to this system where they can see the reservation available for them. They can accept the request and complete them after the ride is finish.

This system can be modified and changed according to any company requirement. So any company (using vehicles for their day today activities) can use this system. The most important requirements are built up in this system and also it is a user friendly system.

This system uses HTML, JAVA, PHP for the web portal and used Android Studio to create the mobile app.

Acknowledgement

I would like to express my gratitude to my supervisor Mr. G.K.A Dias for the useful comments, remarks and engagement through the learning process of this master thesis. Furthermore I would like to thank my office colleagues for introducing me to the topic as well for the support on the way. Also, I would like to thank my Family, who has supported me throughout entire process, both by keeping me harmonious and helping me putting pieces together. I will be grateful forever for your love.

Table of Content

Declaration…….. i

Abstract…… ii

Acknowledgement…….. iii

Table of Content……. iv-vi

List of Figures……. vii -viii

List of Tables…… ix

CHAPTER 1: Introduction ……..1

1.1 Motivation……. 1
1.2 Statement of the Problem…… 1
1.3 Aims and Objectives……. 2
1.4 Scope…….. 2 - 3

CHAPTER 2: Background……..4

2.1 Similar Systems……4

CHAPTER 3: Analysis and Design……..5

3.1 Software Requirement Specification……..5

3.1.1 Introduction……...5 -6

3.1.2 General Description…….. 7 -8

3.1.3 Specific Requirements…… 9-14

3.2 Design …….15

3.2.1 Software Architecture……. 15

3.2.2 Database Design……... 16-17

3.2.3 Class Diagram…….. 18

CHAPTER 4: Implementation……19

4.1 Module Structure of the system……19

4.1.1 Interface Layer……19

4.1.2 Business Logic Layer……..19

4.1.3 Data Layer……20

4.2 Technology Aspects…….21

CHAPTER 5: User Evaluation and Testing…….22

5.1 User Evaluation …….22

5.1.1 Sampling of Subjects…….. 22

5.1.2 Evaluation Scenarios…….. 23

5.2 Testing…….24

5.2.1 Test Plan……. 24

5.2.2 Unit Testing…… 25-26

5.2.3 Integration Testing…… 27-28

5.2.4 System Testing…… 29

5.2.5 Acceptance Testing……. 29-31

CHAPTER 6: Conclusion and Future Work……..32

6.1 Conclusion…….32

6.1.1 Major Findings……...32
6.1.2 Lessons Learnt…….....32
6.1.3 Achievement of objects……..32
6.1.4 How to extend…….33

6.2 Future Work……. 33

References……... 34

Appendix A- System documentation …… 35- 36

Appendix B- User Documentation……..... 37- 50

Appendix C-Design Documentation……51- 67

Appendix D –Test Results …….. 68-79

List of Figures

Figure 1: overall use case diagram……11

Figure 2: Descriptive use case……13

Figure 3: Three tier- architecture……15

Figure 4: ER Diagram…….

Figure 5: Class Diagram…….18

Figure 6: Interface Layer……19

Figure 7: Data Layer……20

Figure 8 : Software Testing……..24

Appendix B

Figure 9: User Registration Page……38

Figure 10: User Login page……...39

Figure 11: User Reservation Form……..40

Figure 12: Request details page……..41

Figure 13: Reservation History page……..42

Figure 14: User Profile page……...43

Figure 15: Admin Registration page……...44

Figure 16: Pending Requests page……..45

Figure 17: Vehicle Allocation page……46

Figure 18: Allocated list page…….47

Figure 19: Reports page……..48

Figure 20: Query page…….48

Figure 21: Admin profile page……49

Figure 22: Driver status page……..49

Figure 23: Driver Journey Status page……50

Appendix C

Figure 24: Client Reservation through website process flow diagram……52

Figure 25: Client Reservation through mobile app process flow diagram……..54

Figure 26: Admin side reservation acceptance through website……55

Figure 27: Admin side reservation acceptance through mobile app……...56

Figure 28: Main interface page……57

Figure 29: User registration page……57

Figure 30: Login Page……..58

Figure 31: Vehicle reservation page……58

Figure 32: Reservation history page…… 59

Figure 33: Pending list page……59

Figure 34: User profile page……60

Figure 35: Main interface page……61

Figure 36: Admin login page……...61

Figure 37: Admin pending list page……62

Figure 38: Admin allocated list……...62

Figure 39: Admin profile…….63

Figure 40: Registration page (mobile app) …….64

Figure 41: Login page (mobile app) ……...64

Figure 42: Reservation page (mobile app) ……..65

Figure 43: Options available for the user (mobile app) ……..65

Figure 44: Reservation approval page (mobile app) ……..66

Figure 45: Driver status (mobile app) ……67

Figure 46: Journey Status page (mobile app) …….67

Appendix D

Figure 47: Acceptance Testing result……..79

List of Tables

Table 1: Product functions ……7

Table 2: Unit testing (User login) ……25

Table 3: Unit testing (User Registration) ……..26

Table 4: Integration testing (User) ……27

Table 5: Integration testing (Admin) ……28

Table 6: Integration testing (Driver) ……28

Appendix D

Table 7: Unit testing (reservation) ……69

Table 8: Unit testing (History)……...69

Table 9: Unit testing (Profile) ……69

Table 10: Unit testing (Admin side login) ……70

Table 11: Unit testing (Admin side pending list) ……70

Table 12: Unit testing (Admin side allocated list) ……71

Table 13: Unit testing (Admin side vehicle list) ……...71

Table 14: Unit testing (Admin side query) ……...71

Table 15: Unit testing (Admin side reports) ……71

Table 16: Unit testing (Driver side driver status)……..72

Table 17: Unit testing (Driver side journey status) ……..72

CHAPTER 1: Introduction

1.1 Motivation

A vehicle reservation system is a main system in a large business company which has large number of vehicles for their day today business. Practically I already used this kind of system in my company .As SLT is a leading company in sri lanka they maintained a vehicle reservation system for the use of employees for their day today works. This system motivated to do a more effective , more efficient , more reliable Vehicle reservation system to be used by any company. Having a vehicle reservation system makes the company more easy to use their vehicle in day to day operations.

1.2 Statement of the problem

A large company which is having a large vehicle pool may face to following problems while they reserve vehicles manually for their day to day work.

- Employee need to get a manual form and fill it and submit. Employee has to submit the form to the pool himself. This is a time consuming process.
- When admin gets forms, he needs to manually allocate vehicles by searching the vehicle availability through Files. He has to maintain many files like Reservations, History, Driver attendance, Vehicle availability. This is a time consuming process and the accuracy of the details are doubtable.
- Drivers have a manual attendance marking process. When a reservation made admin has to inform it to the driver manually.
- Admin plays a very stressful role in this manual process.

1.3 Aims and Objectives

The ultimate objective of this project is to provide cost effective, reliable, user friendly computerized system for a vehicle pool of a large company. This system can be modified and changed according to new technologies and for new requirements. So any company (using vehicles for their day today activities) can use this system with their requirements. The basic and main requirements and features are added to this system.

- Objectives of this Vehicle reservation system are as follow:
  - To automate the manual vehicle requesting process.
  - To automate the manual vehicle reservation process.
  - To automate the manual Driver management process.
  - To provide an efficient, reliable, cost effective system.

1.4 Scope

A web based system which a client can reserve a vehicle from the company and also from outside the company. A client can get a login as soon as he fills the login form. To reserve a vehicle, he can fill a request form and he will receive an email saying that his request was sent to the vehicle pool of the company. When the request is accepted, again an email sent to the requested client saying that his request has been accepted and later he will receive an email and a SMS with the details regarding the reservation (driver name, Vehicle No, driver's contact number).

From the Administrator view, when a request made, he need to check whether enough vehicles are free to allocate. If it is free he will accept the request. If not, he will arrange an outside vehicle (Taxi/Cab) manually. This process is not automated and admin need have a connection or an agreement signed with an outside taxi service company. Admin arrange the requirement and send a mail and SMS to the client with the driver's Details. Admin can generate reports regarding vehicle details, driver details and reservation details.

Admin can enter insurance/licence details of each vehicle and if the licences are about to expire, system will pop up an alert one month before the expiration date. So the admin can take necessary actions for that.

Also the authorized people can access the site using mobile through the mobile app made for this vehicle reservation system. They can accept the request and approve them, even if they are not in office. Drivers can mark their availability through mobile and also they can complete the journey after they came back to the office.

Client side and Administrator side features are as follow:

Client side

- Login to the website.
- Request a vehicle through a requesting form.
- Can check for previous reservation details.
- Can check for their profile details.
- Can view details about the vehicle pool.
- Can use the Mobile app for the above purposes.

Administrator side

- Can accept requests sent from the user.
- Can check the availability of vehicles.
- Can reserve vehicles for requests.
- Can generate reports (daily reservations/driver attendance/vehicle availability/Licence, insurance details).

Chapter 2: Background

2.1 Similar systems

Vehicle reservation systems are getting popular in the world these days. In the term of vehicle reservation, there can be many reservation types such as company vehicle pool management and reservation systems, Vehicle rental company reservation systems e.t .c.
This Project is mostly aimed at Company vehicle pool management and reservation system, because it is critical to manage a company with a large amount of vehicles for their day to day works and it will be easy if there is a system to manage the vehicle pool.

There are many vehicle pool management systems all over the world and they are created for various reasons.

For an example,

- **City of Kelowna, decided to change the way they do things when it comes to their pool vehicle operations. After years of struggling with underutilized vehicles and a cumbersome management system, the city of Kelowna became the first city in North America to adopt an efficient and unique pool vehicle management technology.**
  COCOS, an acronym for Company Car Organization system is the technology supplied by INVERS mobility solutions.[5]

- **Honeypot IT consulting private Limited is an internationally established software development company with offices USA and India. Transport Portal is an online software application which fulfills the requirement of a typical management of details of vehicles, Transport Categories, Routes and the requests for bus or cab from the employees in a company. It provides the interface to users in a graphical way to manage the daily transactions. Also provides the management reports like Bus Request details, Bus occupancy, Cab Request details, Cab Request Status etc.** [3]

Chapter 3: Analysis and Design

3.1 Software Requirement Specification

3.1.1 Introduction

Purpose

This SRS section describes the functionality and the performance goals for the software project “Vehicle Reservation”. This Vehicle reservation system intended to help the employees/clients who need to allocate vehicles for the use of company’s day today performances.
This SRS delivers the features of the Vehicle reservation system to serve as a guide to the developers and a software validation document to the client.

Scope

In this section we describe the features that are in the scope and that are not in the scope of the software to be developed.

Functionalities of the Client are categorized below:

- Clients can register through the System.
- Clients can change their details.
- Clients can request a vehicle.
- Clients can cancel a request.
- Clients can check the reservation history.
- Clients can do changes to their profile.
- Clients can do the above tasks using the mobile app too.

Functionalities of the Administrator are categorized below:

- Administrator can check for the vehicle availability.
- Administrator can accept a reservation request.
- Administrator can allocate an outside vehicle if the pool vehicles are already allocated.
- Administrator can add/delete/update vehicle details in the database
- Administrator can get reports.
- Administrator can query the database.
- Administrator can do the above tasks using the mobile app too.
- Administrator gets an alert if the license/insurance are about to expire.

Functionalities of the Driver are categorized below:

- Drivers can mark their attendance through mobile app.
- Drivers can complete the journey through mobile app.

Features outside the Scope

- Vehicle damages, Fuel consumptions are out of the reservation process and should handle separately.
- GPS tracking should handle separately if needed.

Overview of Document

Section 3.1.1 provides a brief introduction of the software requirements of the system. The 3.1.2 section provides a description of the overall system, some user requirements and the interaction between the user and the system. The 3.1.3 section describes the functional and interface requirements in detail.

3.1.2 General Description

Product Perspective

This Open source vehicle reservation system is mainly based on Companies who are having a large Vehicle pool and who are willing to have a system to manage this vehicle pool. The primary Users will be the registered employees of the company and the administrator who organize everything in the system. This system should be user friendly, quick to access and reliable software to use. Following Products perspectives are important:

- This system must be a stand- alone system which is not depend on another subsystem.
- This system will require a web browser and should support to any browser.
- This system needs not any special hardware components. Basic needs are a computer, working web browser and an internet connection.
- This system should work as a mobile app too. To use the mobile app Clients need a smart Phone.

Product Functions

Vehicle reservation system support the use cases defined in Table 1.

Use case	Description
Client Details	An Employee can register.
Vehicle details	Details of the Vehicles and drivers.
Reservations	Details of reservation made.
Alerts	Alerts on when a insurances/ license expired

Table 1: Product functions

Actors

- Client/Employee
- Admin
- System

Assumptions and dependencies

- Full working of Vehicle Reservation system is based on the availability of the internet connection to the desktops.
- Full working of Vehicle Reservation system is based on the availability of the internet connection to the Mobile.

3.1.3 Specific Requirements

Functional Requirements of this Vehicle reservation system are as follow:

Client side functional requirements

- Login to the website.
- Request a vehicle through a requesting form.
- Can check for previous reservation details.
- Can check for their profile details.
- Can view details about the vehicle pool.
- Can use the Mobile app for the above purposes.

Administrator side functional requirements

- Can accept requests sent from the user.
- Can check the availability of vehicles.
- Can reserve vehicles for requests.
- Can generate reports (daily reservations/driver attendance/vehicle availability/Licence, insurance details).

This section describes the functional requirements by giving various use cases. Figure 1 shows the overall use case diagram.

Use case related to Client Details:

Use case 1: Client

Primary Actor: Client/Employee

Pre -condition: Internet Connection

Main scenario:

1. User log to the intranet of the Company.
2. System asks the user for a login. If the user is still not a member of the system user need to register to the system.
3. User will get a form to enter specific details of the user.
4. If the user already signed in, they can give the relevant user name and password.

Use case related to Vehicle Details:

Use case 2: Vehicle

Primary Actor: Admin

Pre -condition: Internet Connection

Main scenario:

1. Admin Can enter the vehicle details
2. Admin can enter Driver details according to each vehicle
3. Admin can enter details of insurance details of the vehicles

Use case related to Reservation Details:

Use case 3: Reservation

Primary Actor: Client/Employee

Pre -condition: internet connection and login

Main scenario:

1. Client enter with the username and password
2. Client fill the reservation form
3. Send to the admin for approval.

Use case related to Alerts Details:

Use case 5: Alerts

Primary Actor: Administrator

Pre -condition: Admin need to log to the system backend.

Main scenario:

1. Admin set alert for the insurance details

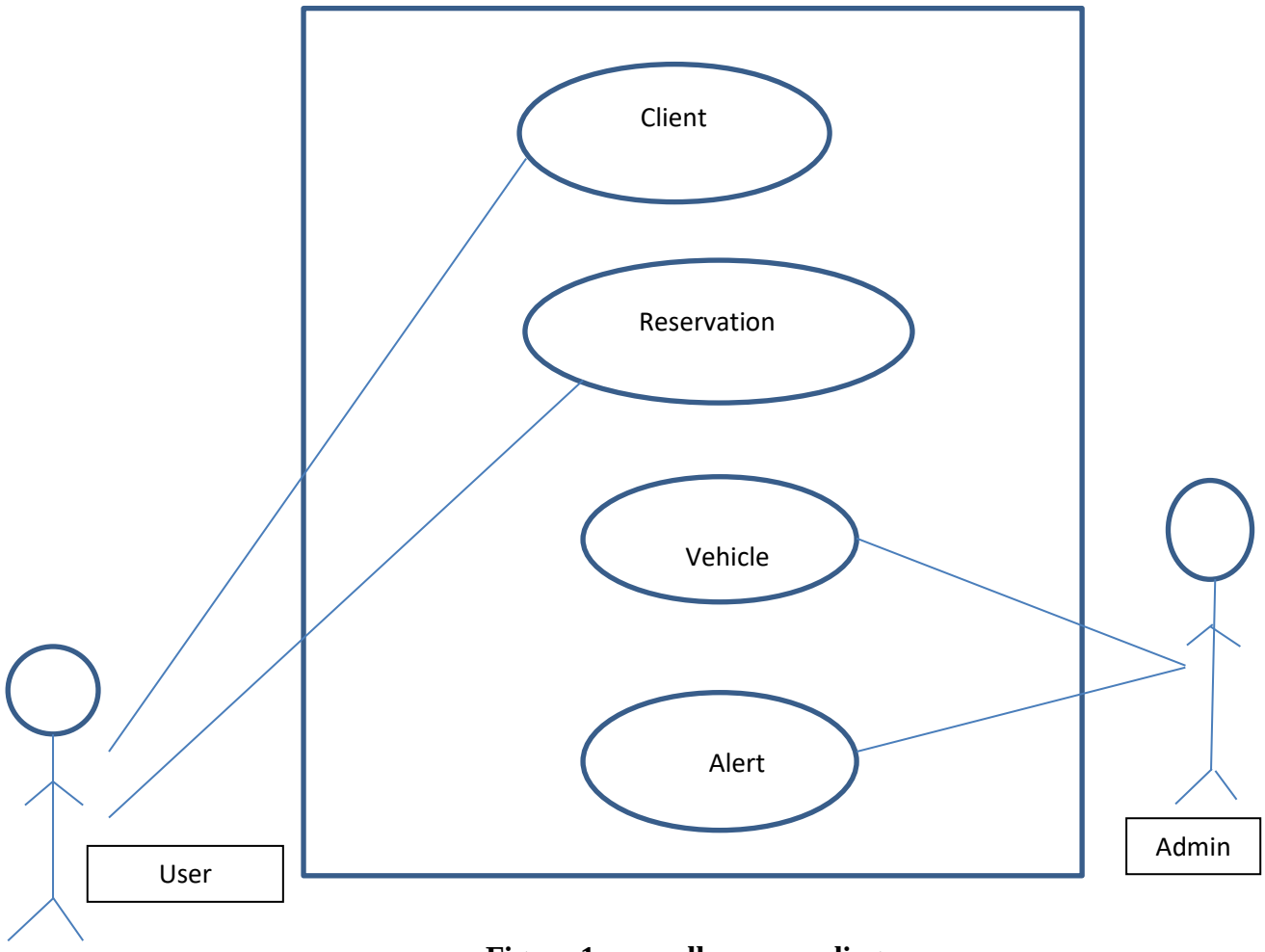


Figure 1: overall use case diagram

More descriptive use cases:

Use cases for Client

- Get a login
- Make an vehicle request

Use cases for Admin

- Check the availability of vehicles
- Accept the request
- Allocate vehicles
- Send details to clients through sms and emails
- Generate Reports
- Enter Insurance/license details

Use cases for System

- Pop up an alert for insurance/license

More Descriptive use cases are shown in Figure 2

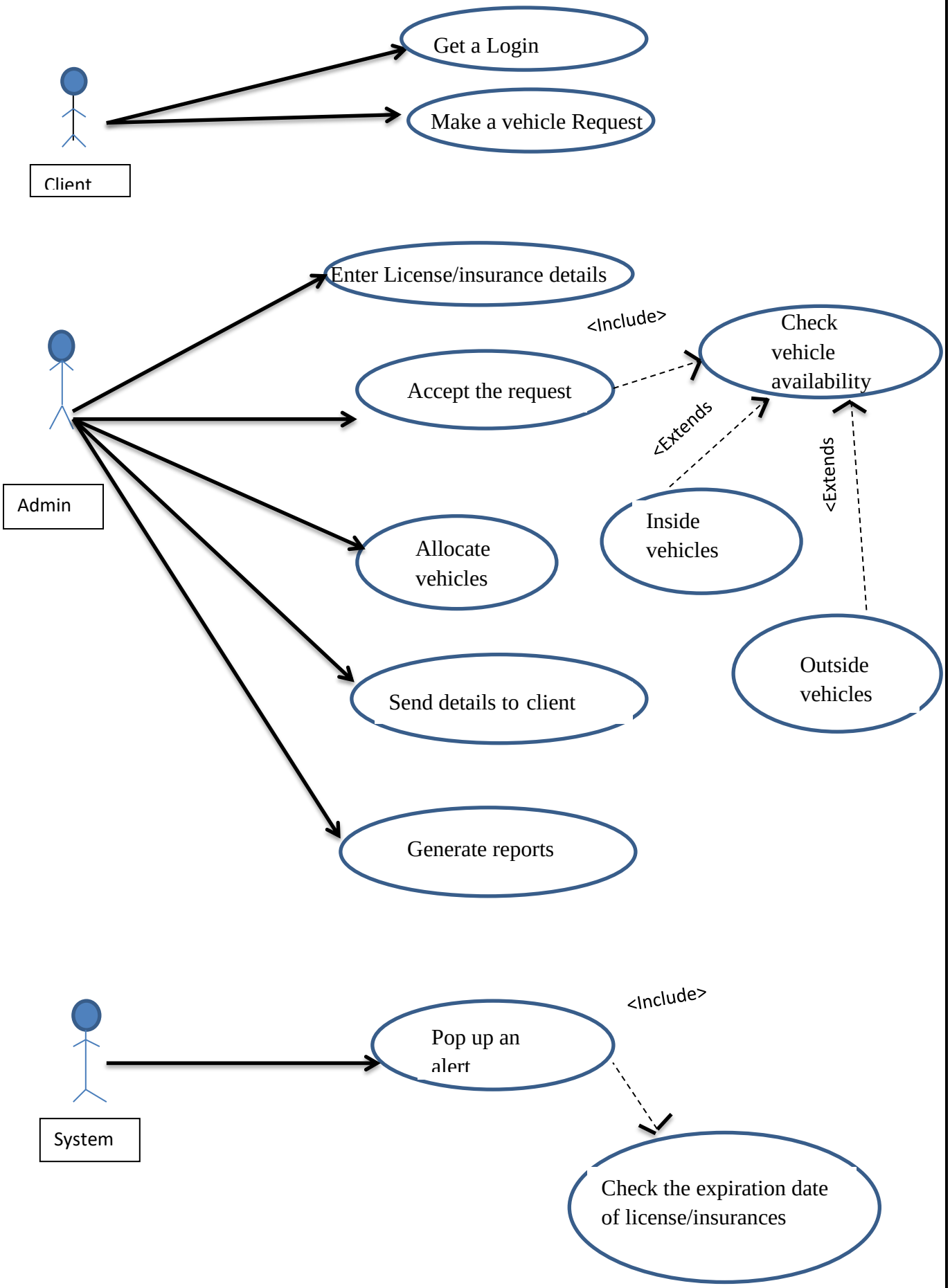


Figure 2: Descriptive use case

Performance Requirements:

1. Should have a good internet connection.
2. Compatible machine to log the site or a mobile device to log the site.
3. Mobile device to get alert quickly.

Design Constraints:

1. *Security:* The details about the clients and vehicle should be secured.
2. *Fault tolerance:* Data should not become corrupted in case of system crash or power failure.

Software Requirement Specification

Programming Languages: Java (jdk1.5),PHP
Web Technologies : Html, JavaScript.
Database : Mysql
Server : Apache

Hardware Requirements Specification

Processor : PIII
Hard Disk: 40 GB
Ram : 256 mb

3.2 Design

System design is mainly focus on the design of the system which includes the ER diagrams, Process flow diagrams, user interface design etc. Designing the front end and the back end includes under this section.

3.2.1 Software architecture

Software application architecture is the process of defining a structured solution that meets all of the technical and operational requirements, while optimizing common quality attributes such as performance, security, and manageability. It involves a series of decisions based on a wide range of factors, and each of these decisions can have considerable impact on the quality, performance, maintainability, and overall success of the application. The 3 tier-architecture for the reservation system is as follows (Shown in figure 3)

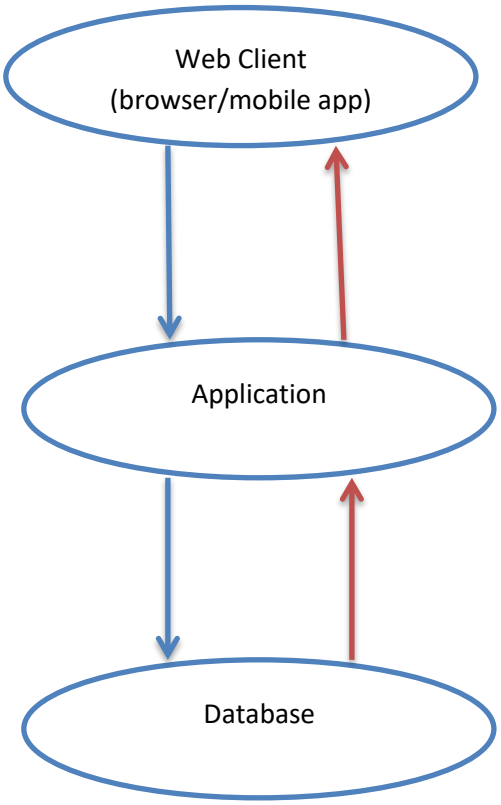


Figure 3: Three tier- architecture

3.2.2 Database Design

An Entity Relationship (ER) Diagram is a type of flowchart that illustrates how “entities” such as people, objects or concepts relate to each other within a system. ER Diagrams are most often used to design or debug relational databases in the fields of software engineering, business information systems, education and research.

ER diagram for the vehicle reservation system is shown in Figure 4.

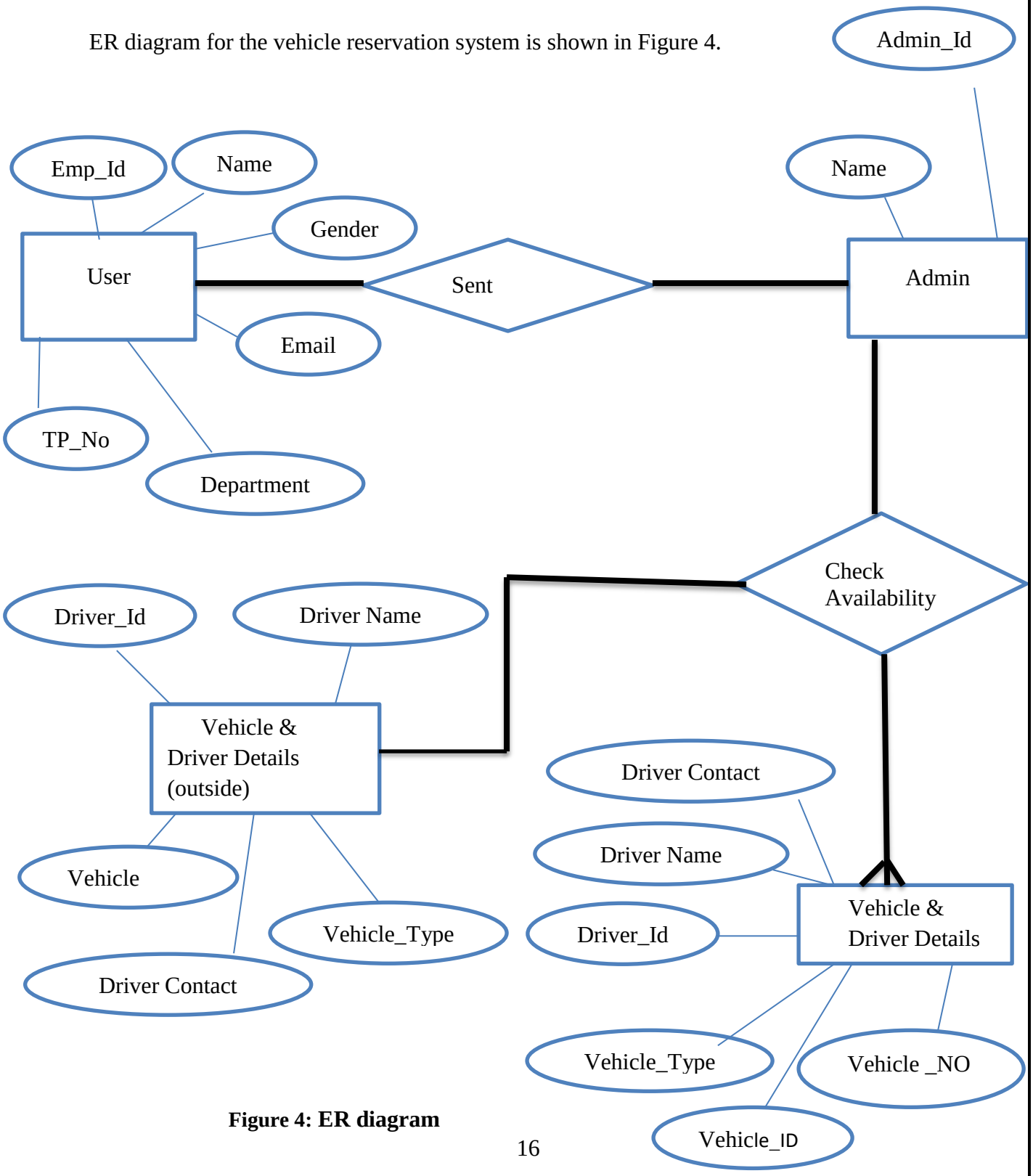


Figure 4: ER diagram

Entities

- Client- Properties of the client.
- Admin- Properties of the Admin.
- Vehicle & Driver Details (inside pool)- Properties of the vehicle pool vehicles and drivers.
- Vehicle & Driver Details (outside) - Properties of the outside vehicles which are allocated if the vehicles in the pool are not sufficient and properties of drivers.

3.2.3 Class diagram

The class diagram of the system is shown in figure 5.

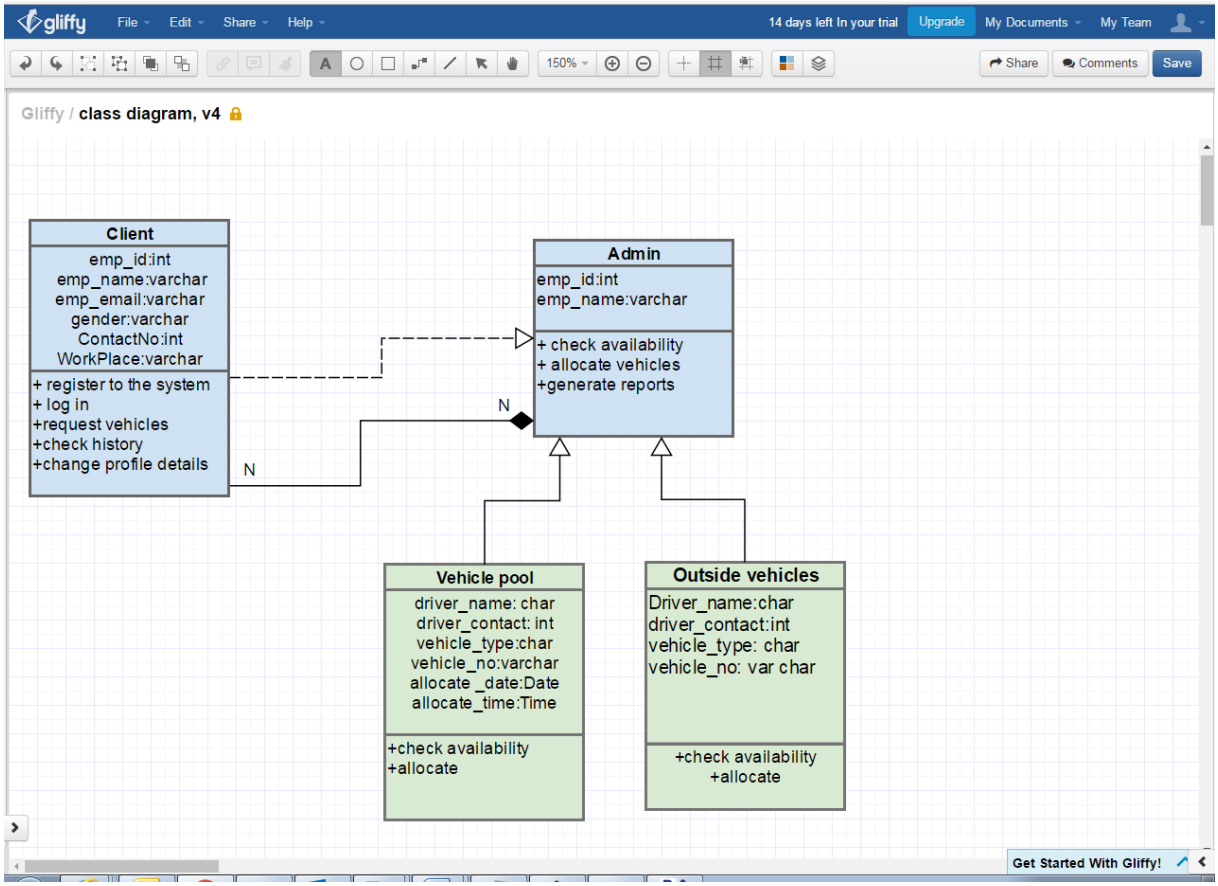


Figure 5: Class Diagram

Refer Appendix C for more diagrams.

Chapter 4: Implementation

4.1 Module Structure of the system

4.1.1 Interface Layer

The interface layer of this system is built using HTML and CSS style sheets. The main menu is available in the left hand side is for easy navigation throughout the system. The menu structure displayed for each user will be based on the user role of the logged in user.
Some of the interface controls will also be enabled and disabled according to the user (shown in figure 6).

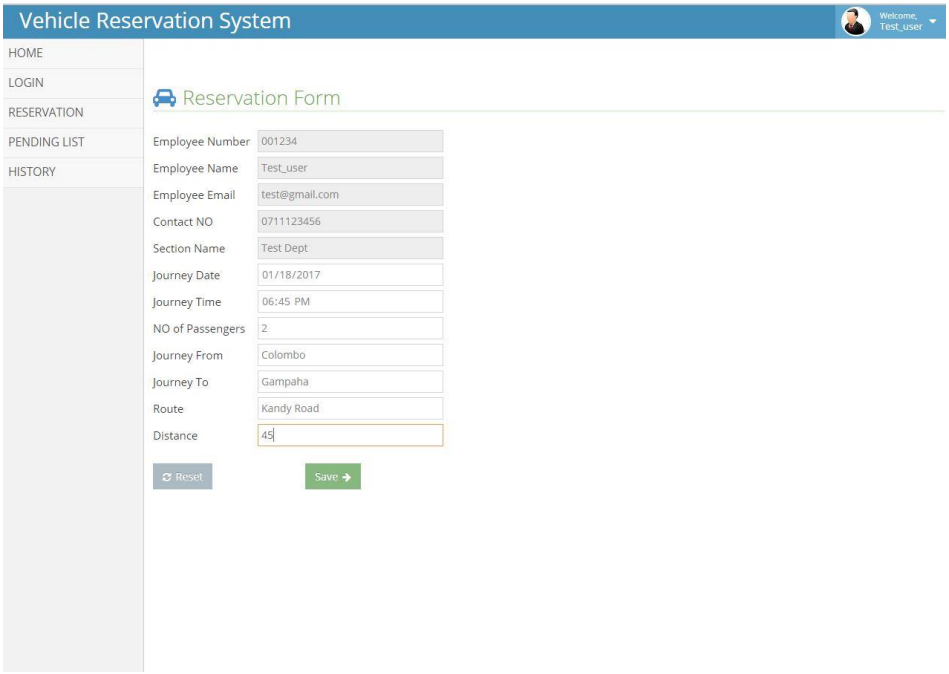


Figure 6: Interface Layer

4.1.2 Business Logic Layer

The business logic layer is created using javascript.

4.1.3 Data Layer

The data layer of this system is created using SQL server. The database has been modeled using an entity relationship diagram. This ER diagram used to create the initial tables (shown in figure 7).

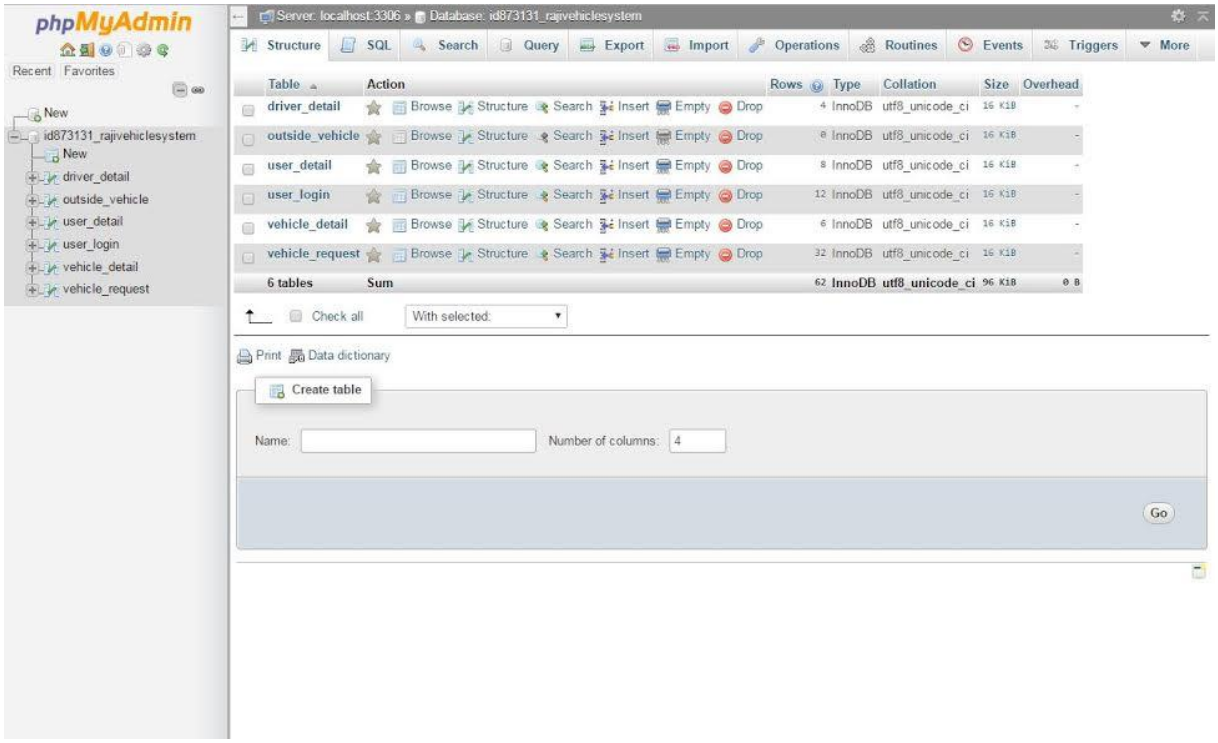


Figure 7: Data Layer

4.2 Technology Aspects

The actual codlings of the system started just after the design of the system as mentioned in the previous chapter. Following Techniques used:

PC Platform: - Windows OS, Google chrome, Mozilla

Mobile platform: - Android studio.

Software framework: - Oracle Database

Languages used: - PHP, JAVA, HTML

These techniques are used because these are the well-known, easy to work with and also as the project title says ”Open source”, any company could get this and can be used with their knowledge and tools.

Chapter 5: User Evaluation and Testing

5.1 User Evaluation

5.1.1 Sampling of Subjects

Sampling is a Statistic and quantitative research methodology. Since it is generally impossible to study an entire population (every individual in a country, all college students, every geographic area, etc.), researchers typically rely on sampling to acquire a section of the population to perform an experiment or observational study. It is important that the group selected be representative of the population, and not biased in a systematic manner.

There are three main types of Sampling.

1. Simple Random Sampling: - Simple random sampling is the basic sampling technique where we select a group of subjects (a sample) for study from a larger group (a population).

2. Stratified Random Sampling: - A stratified sample is obtained by taking samples from each stratum or sub-group of a population.

3. Multistage Random Sampling: - A multistage random sample is constructed by taking a series of simple random samples in stages

According to this system, it is better to select Stratified Random sampling, because this system may be used in a transportation section on a Company, and that means this system can be handle with a normal IT knowledge person to all the employees in a company, because this is client based system. Selecting different people in different IT knowledge levels is a best way to check the system in various levels of knowledge.

This Sample contain System administrators, normal clients, Engineers; supervisor's e.t.c. A different set of clients will give the best result regarding the process of the system some They can checked the system, the algorithms, the flow, and the features.

5.1.2 Evaluation Scenarios

A scenario is a description of a person's interaction with a system. This interaction is based on different scenarios with the system. These Scenarios describe the process of the system in various levels, in various user angles.

User side Scenarios

- Login- If the system login with no issues meets the relevant requirements and if the system pops ups the needed error messages at the right time.
- Registration- If the user is new to the system, the have to register first. There is a Registration form, which should be filled by the user. Check when the relevant fields in the form pop up the correct messages and when submit it added to the database.
- Reservation – when user login they can see the reservation form to fill. Check when the relevant fields in the form pop up the correct messages and when submit it added to the database and send it to admin panel for reservation and an email received for the client about the reservation made.
- Profile- User can make changes to their profile.
- History- User can check their previous reservations.

Admin Side Scenarios

- Login- Admin can login with their username and password.
- Requests - Check whether the request count increased when a user made a request. Check whether the admin can check the request and assign the available drivers to the request.
- Vehicle Availability – admin can check the unassigned vehicles and can assign for the coming requests.
- Request Acceptance – Check when Admin accept the request an email and a sms sent to client.

5.2 Testing

Software testing is a process of executing a program or application with the intent of finding the software bugs. Also, it can be defined as the process of validating and verifying that a software program or application meets the business and technical requirements that guided its design and development. Software testing can be as follows.(shown in figure 7)

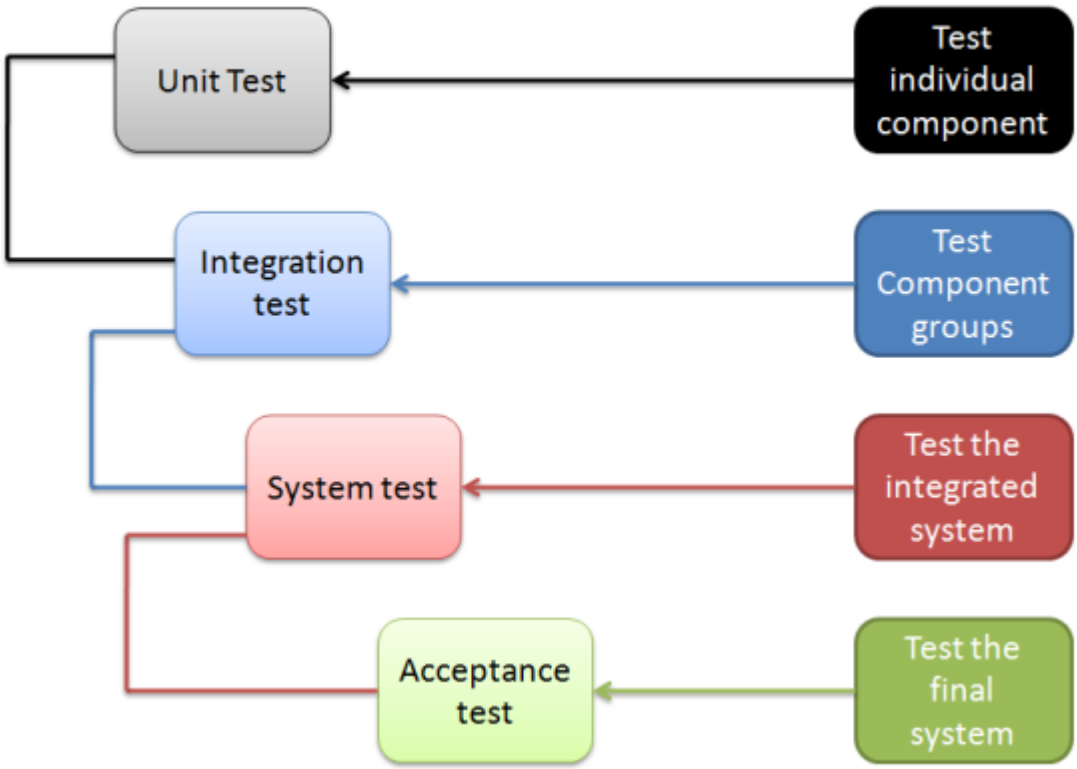


Figure 8: Software Testing

5.2.1 Test Plan

The test plan is implemented to plan how to test the system. This system should test unit by unit to check for any bugs or errors. Also with the result of the Unit Testing, then the Integration test can be implemented or can correct any errors found from Unit testing. Integration test allowed testing the units together. When the integration testing is successful, System test will be implemented to test the complete system. Finally the Acceptance testing is done by the internal and external volunteers to check whether the system gives the needed requirements to the client.

5.2.2 Unit Testing

According to the above scenarios following test cases are implemented and following results gained.

User side Scenarios

- **Login**

Test cases(input data)	**Expected result**	**Actual result**
Click login button without entering username or password.	Error occurrence: “Pls enter a valid username or password” and user cannot log on to the system	Error occurrence: “Pls enter a valid username or password” and user cannot log on to the system
Click login button only entering the username.	Error occurrence: “Wrong username or password.” and user cannot log on to the system.	Error occurrence: “Wrong username or password.” and user cannot log on to the system.
Click login button only entering the password.	Error occurrence: “Wrong username or password.” and user cannot log on to the system.	Error occurrence: “Wrong username or password.” and user cannot log on to the system.
Click login button entering wrong username and password.	Error occurrence: “Wrong username or password.” and user cannot log on to the system.	Error occurrence: “Wrong username or password.” and user cannot log on to the system.
Click login button entering correct username and password.	Successfully login to the system.	Successfully login to the system.

Table 2: Unit testing (User login)

- **Registration**

Test cases(input data)	Expected result	Actual result
Click Register button without entering the details	Error occurrence: “Pls fill out the fields”. and user cannot register.	Error occurrence: “Pls fill out the fields”. and user cannot register.
Entered an email address with wrong validations. Ex: ruwan#.bbh@gmail com	Error occurrence and user cannot register.	Error occurrence and user cannot register.
Enter mismatch passwords to Password field and Confirmed password field	Error occurrence: “Passwords do not match” and user cannot register.	Error occurrence: “Passwords do not match” and user cannot register.
Enter all the details and Click Register Button.	Successfully registered.	Successfully registered.
Click on Reset button	Successfully reset the details and erase the fields.	Successfully reset the details and erase the fields.

Table 3: Unit testing (User Registration)

Refer Appendix D for the rest of the test results.

5.2.3 Integration Testing

System is tested with the database. The details entered to the system and details retrieved from the database to the system are tested top to bottom and bottom to top.

User scenario

Test cases	Expected result	Actual result
Click on Register button after filling the form.	Database updated with a new user each time.	Database updated with a new user each time.
Login with the username and password.	Check with the user details and if correct logged in otherwise pop up an error.	Check with the user details and if correct logged in otherwise pop up an error.
Reservation form	User details filled automatically and need to enter the rest of the details and reserve.	User details filled automatically and need to enter the rest of the details and reserve.
History	Data retrieved from the database and shows in the system.	Data retrieved from the database and shows in the system.
Pending list	Data retrieved from the database and shows in the system.	Data retrieved from the database and shows in the system.

Table 4: Integration testing (User)

Admin Scenario

Test cases	Expected result	Actual result
Click on pending allocation.	Data retrieved from the database and shows in the system	Data retrieved from the database and shows in the system
Admin select a pool vehicle or outside vehicle	Database updated with a new allocation.	Database updated with a new allocation.
Reservation form	User details filled automatically and need to enter the rest of the details and reserve.	User details filled automatically and need to enter the rest of the details and reserve.
Allocated List	Data retrieved from the database and shows in the system.	Data retrieved from the database and shows in the system.
Reports	Data retrieved from the database and shows in the system.	Data retrieved from the database and shows in the system.
Query	Data retrieved from the database and shows in the system.	Data retrieved from the database and shows in the system.

Table 5: Integration testing (Admin)

Driver scenario

Test cases	Expected result	Actual result
Driver Status(ON/OFF)	Database updated with the users status.	Database updated with the users status.
Journey Ending	Database updated and the reservation goes to completed list and the driver again available.	Database updated and the reservation goes to completed list and the driver again available.

Table 6: Integration testing (Driver)

5.2.4 System Testing

After successfully finishing the unit and integration testing the system is all together tested. This testing will be done by the developers their selves to check the system, if it meets the requirements. As the unit testing and integration testing was successful, the system testing also successful.

5.2.5 Acceptance Testing

Formal testing with respect to user needs, requirements, and business processes conducted to determine whether or not a system satisfies the acceptance criteria and to enable the user, customers or other authorized entity to determine whether or not to accept the system.
Acceptance testing is done through two stages:

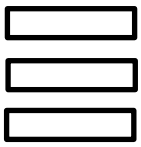
- **Internal Acceptance Testing-**Done by the members of the Organization who developed the system, but not who involved in the development.

- **External Acceptance Testing-**Done by the outside users.

This system is given to some internal and external users for acceptance testing. Following Questionnaire is distributed and collected from them. (Test Results are stated Appendix D)

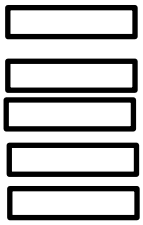
1) How likely is it that you would recommend this system to a friend or company?

- Not at all
- Recommend
- Highly Recommend



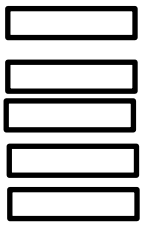
2) How satisfied are you with the reliability of this system?

- Extremely satisfied
- Very Satisfied
- Somewhat Satisfied
- Not so Satisfied
- Not at all satisfied



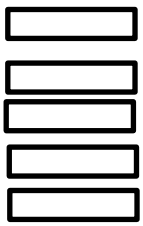
3) How satisfied are you with the security of this system?

- Extremely satisfied
- Very Satisfied
- Somewhat Satisfied
- Not so Satisfied
- Not at all satisfied



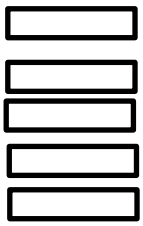
4) How satisfied are you with this system's ease of use?

- Extremely satisfied
- Very Satisfied
- Somewhat Satisfied
- Not so Satisfied
- Not at all satisfied



5) How satisfied are you with the look and feel of this system?

- Extremely satisfied
- Very Satisfied
- Somewhat Satisfied
- Not so Satisfied
- Not at all satisfied



6) Is it easy to learn to use this system?

- Yes
- No

7) Overall, how would you rate?

- Excellent
- Good
- Average
- Poor

8) Do you have any Thought on how to improve this system? State Below.

…….
…….
…….

Chapter 6: Conclusion and Future Work

6.1 Conclusion

The ultimate objective of this project is to provide cost effective, reliable, user friendly computerized system for a vehicle pool of a large company. When implementing the below aspects were achieved.

6.1.1 Major Findings

Major findings gathered after the above stated questionnaire is distributed and the results are added in appendix D.

6.1.2 Lessons Learnt

- Self-studied new html features.
- Self-studied new java features.
- Self-studied new php features.
- Studied how to plan the work of the system
- Time Management
- Learn how to change the system according to others comments

6.1.3 Achievement of Objectives

- Website for the Vehicle reservation client side built.
- Website for the Vehicle reservation Admin side built.
- Mobile app for the Client side built.
- Mobile app for the admin side built
- Client can login using Website and mobile app.
- Client can reserve a vehicle using Website and mobile app.
- Admin can check the newly created requests through Website and mobile.
- Admin can generate reports
- Admin can query the database.
- Admin get an alert one month prior to the licenses/insurance expiration date.
- Driver can mark their attendance through mobile
- Driver can end the journey through mobile.

6.1.4 How to extend the work

- GPS tracker can be added to track the vehicle.
- A way to track when an outside vehicle finished the journey.

6.2 Future Work

As a Future work I recommend that it is better to use a GPRS tracker to find out where the vehicle is in while on a ride. So that will be easier to the admin for their future reservations.

References

ONLINE SOURCES

[1] 1.264J / ESD.264J Database, Internet, and Systems Integration Technologies Fall 2013

[2] PHP Tutorials: https://vineetsaini.wordpress.com/tutorials/

REPORTS

[3] SHADAN COLLEGE OF ENGINEERING & TECHNOLOGY, 2014 Apr 16.[Accessed 21 September 2016] .

[4] Northern territory Government: Vehicle Booking System User Guide – Managed Pool Bookings, 26 June 2015, version 1.0

[5]City of kelwona- A Vehicle Management System, California Fleet News,2005 Nov/Dec.[Accessed 22 March 2016].

System documentation

Appendix A

Installation of the web portal

This Vehicle reservation system is currently running on a free webhosting site. A company can get the link of the website and they can post in their intranet as another button for vehicle reservation.

The link is as below.

System link: **https://abcvehicle.000webhostapp.com/**

Installation of the mobile app

Any user can request the mobile app through the email of the vehicle reservation system. When they request we can send a link containing the download file to them.

The can run it and download it to their mobile easily.

User Documentation

Appendix B

User

1) First of all user need to get registered to the system through mobile app or web portal. (Shown in figure 8.)

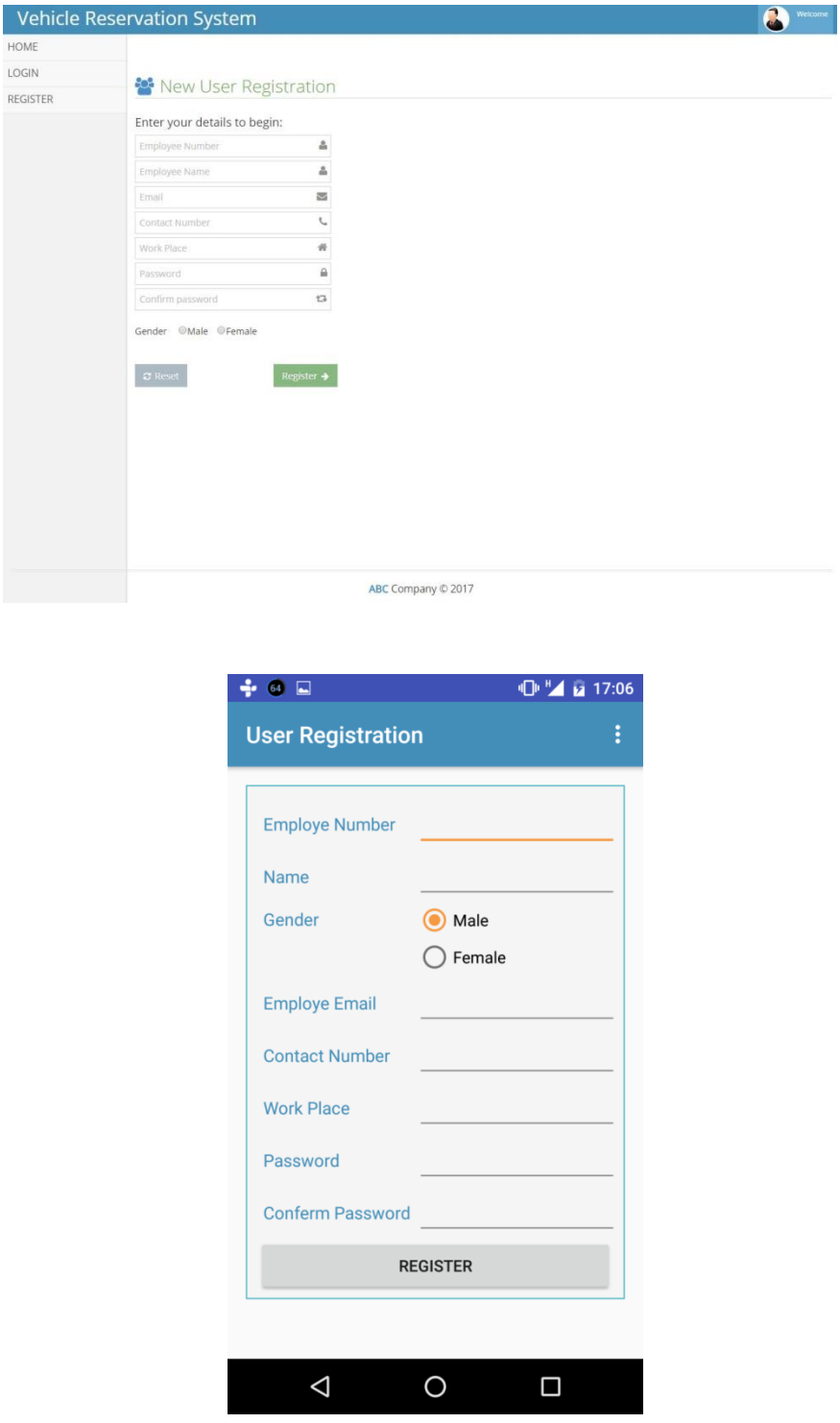
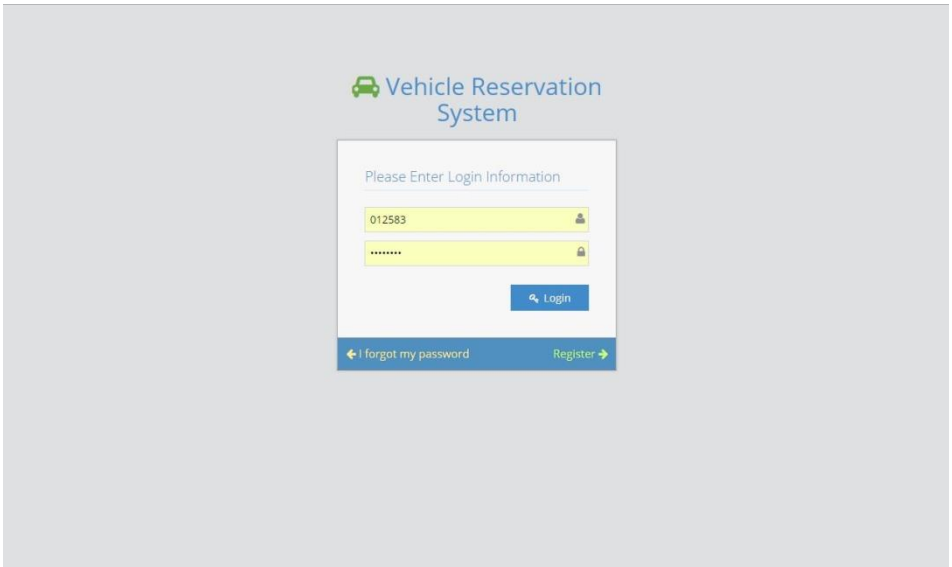


Figure 9: User Registration Page

2) After filling the above form user can get a username and password and can log in to the system. (Shown in figure 9.)



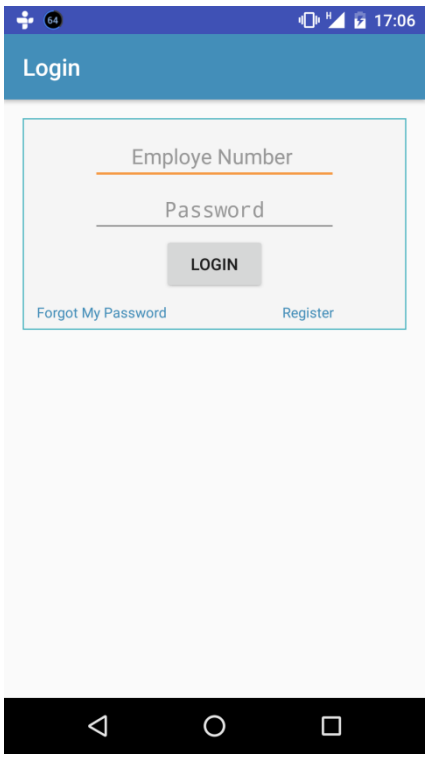


Figure 10: User Login page

3) When logged in, reservation form will be appeared as the default page. User can fill the relevant fields and submit. (Shown in figure 10.)

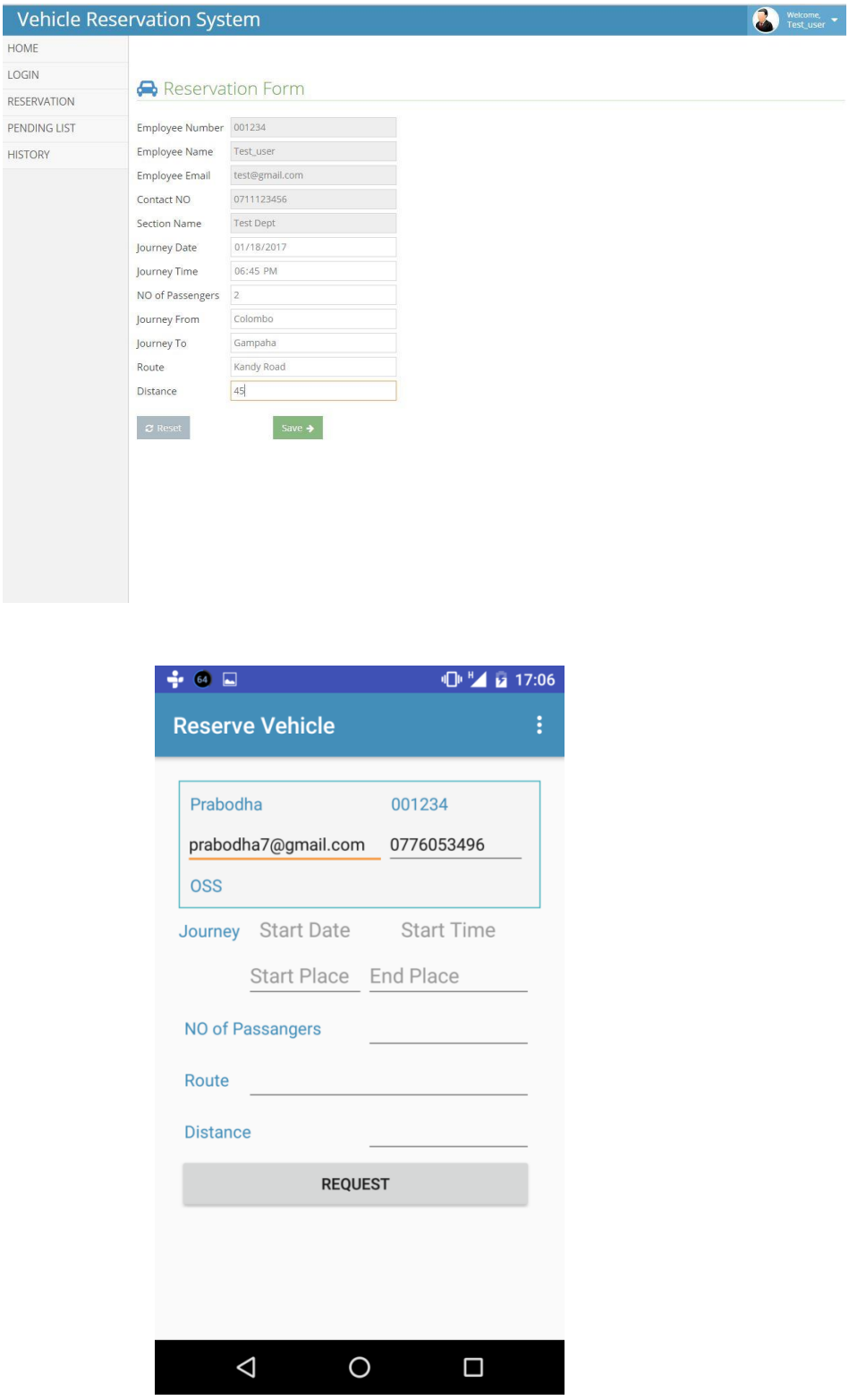


Figure 11: User Reservation Form

4) When the reservation submitted, user will get and email for the submission. The reservation appears in the pending list. User can check the status of the reservation. There are three categories of the status: Pending, Allocated and completed.(Shown in figure 11.)

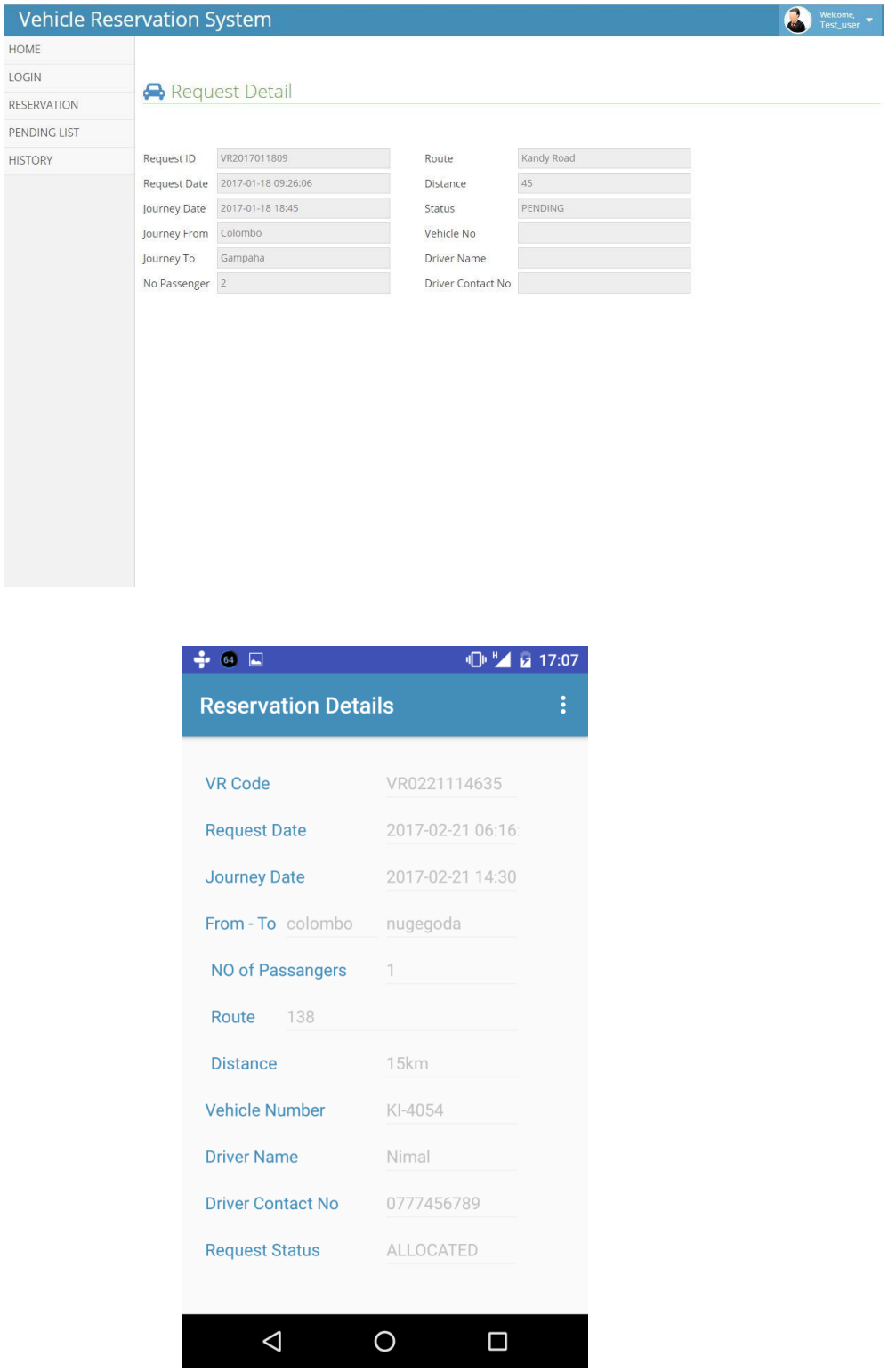


Figure 12: Request details page

5) User can check for the Reservation History.(Shown in figure 12.)

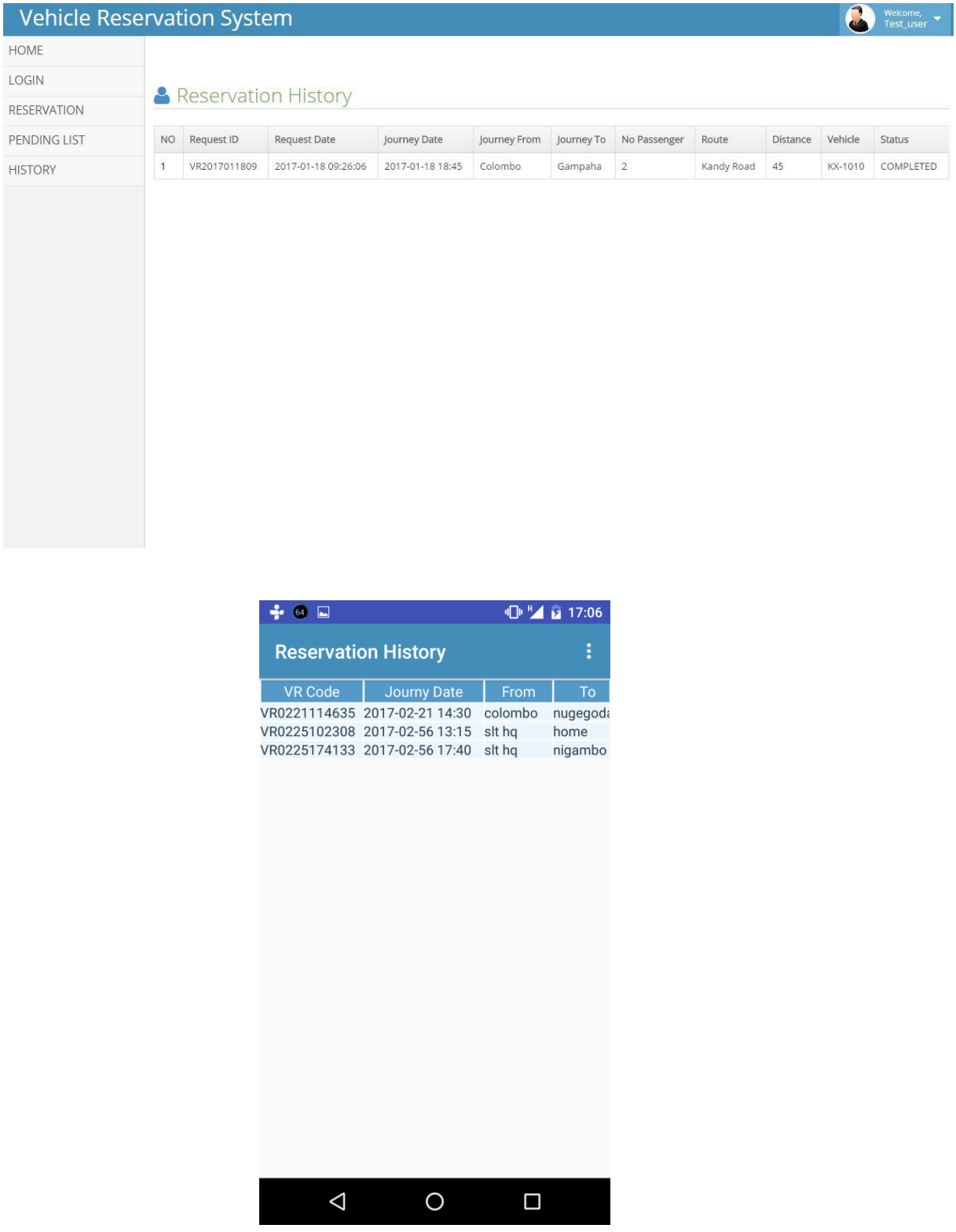


Figure 13: Reservation History page

6) User can check their profile and can do changes to it.(shown in figure 13)

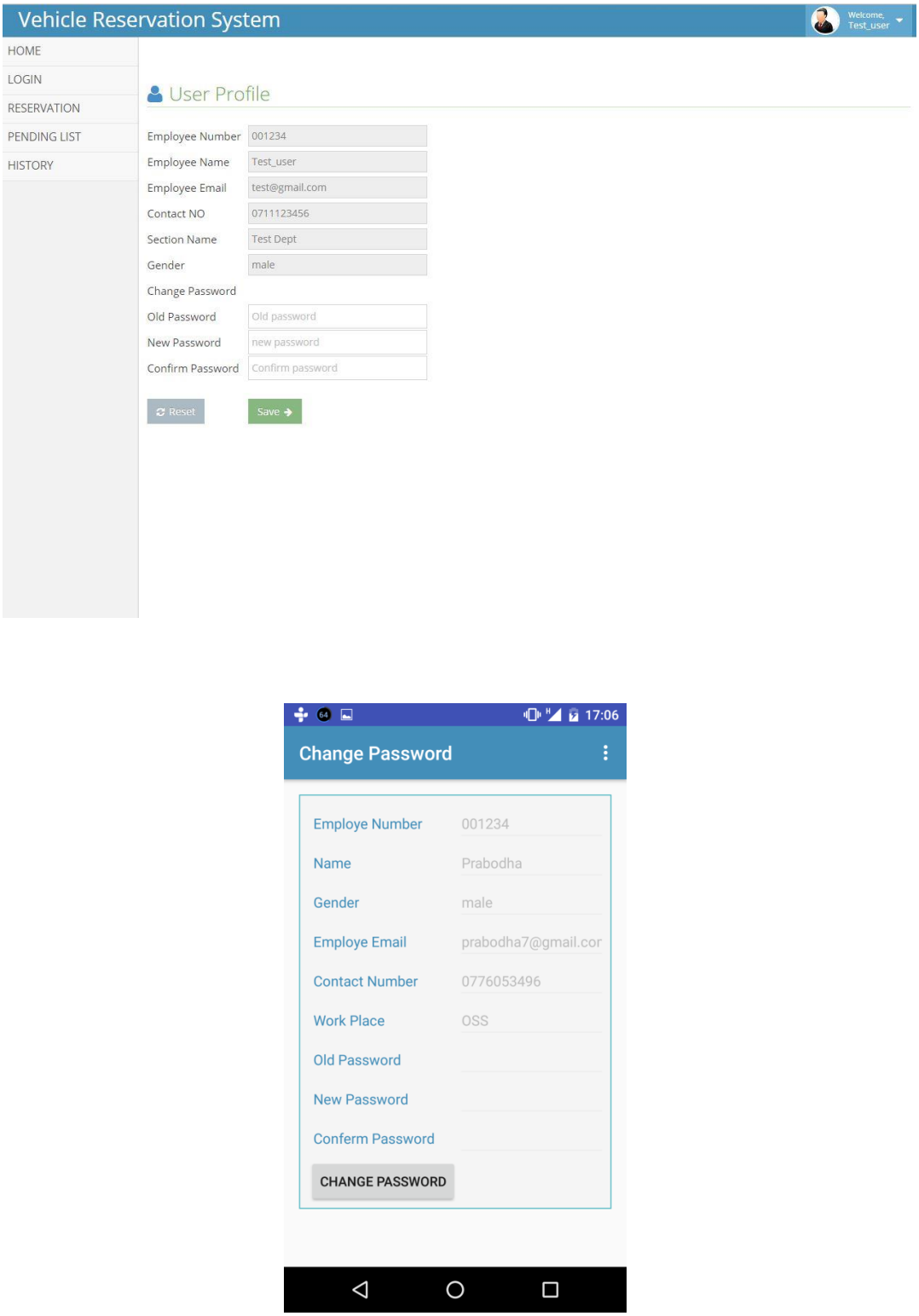


Figure 14: User Profile page

Administrator

1) The admin can register to the system with the same registration form.(Shown in figure 14.)

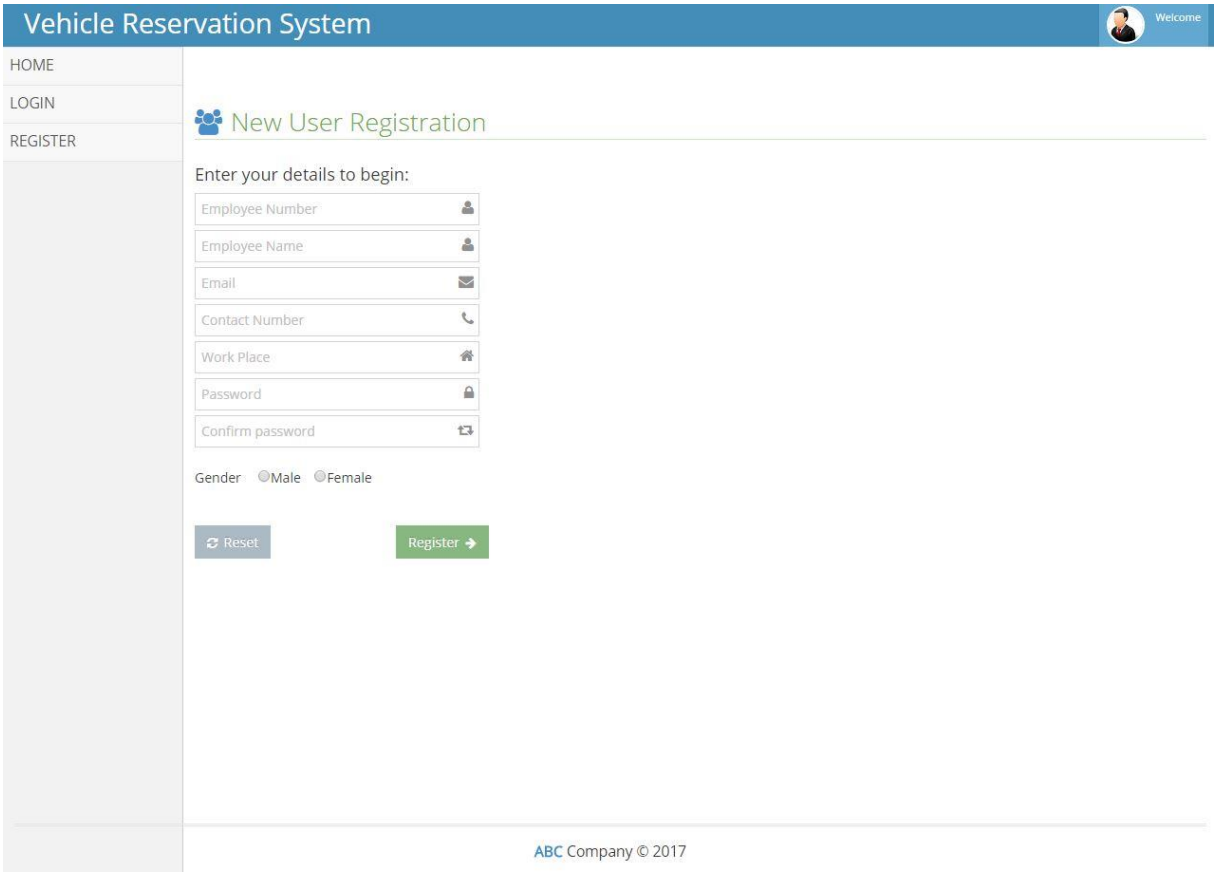


Figure 15: Admin Registration page

2) Admin can see the pending requests as the default page.(Shown in figure 15.)

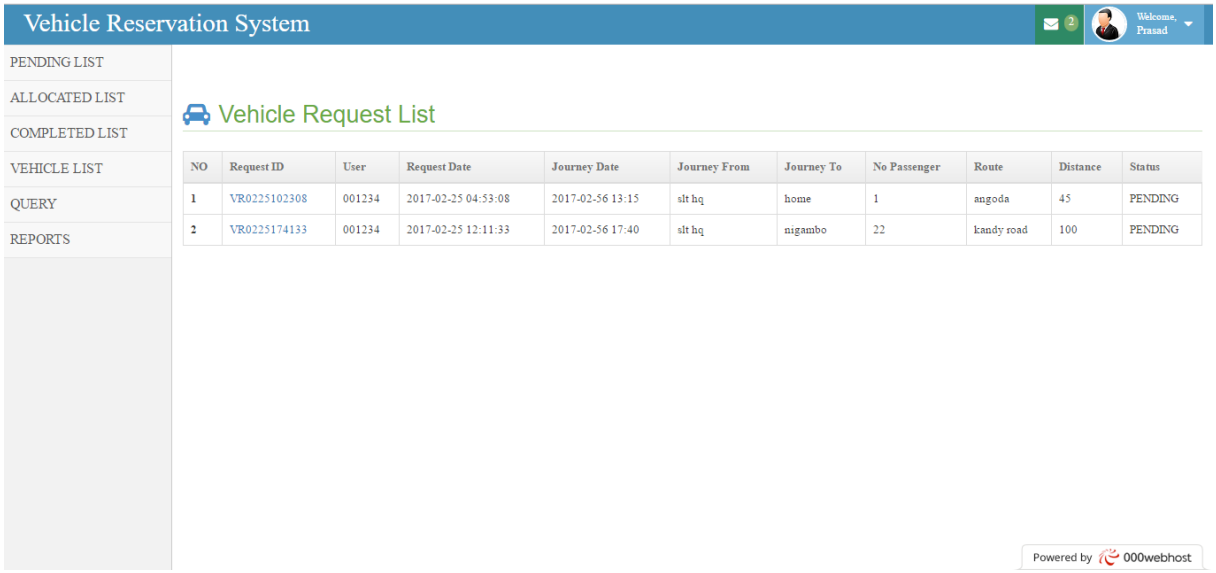


Figure 16: Pending Requests page

3) When he click on a Request ID , admin can allocate a pool vehicle or he can reserve an outside vehicle. Following Message will appear if the allocation is successful.(shown in figure 16.)

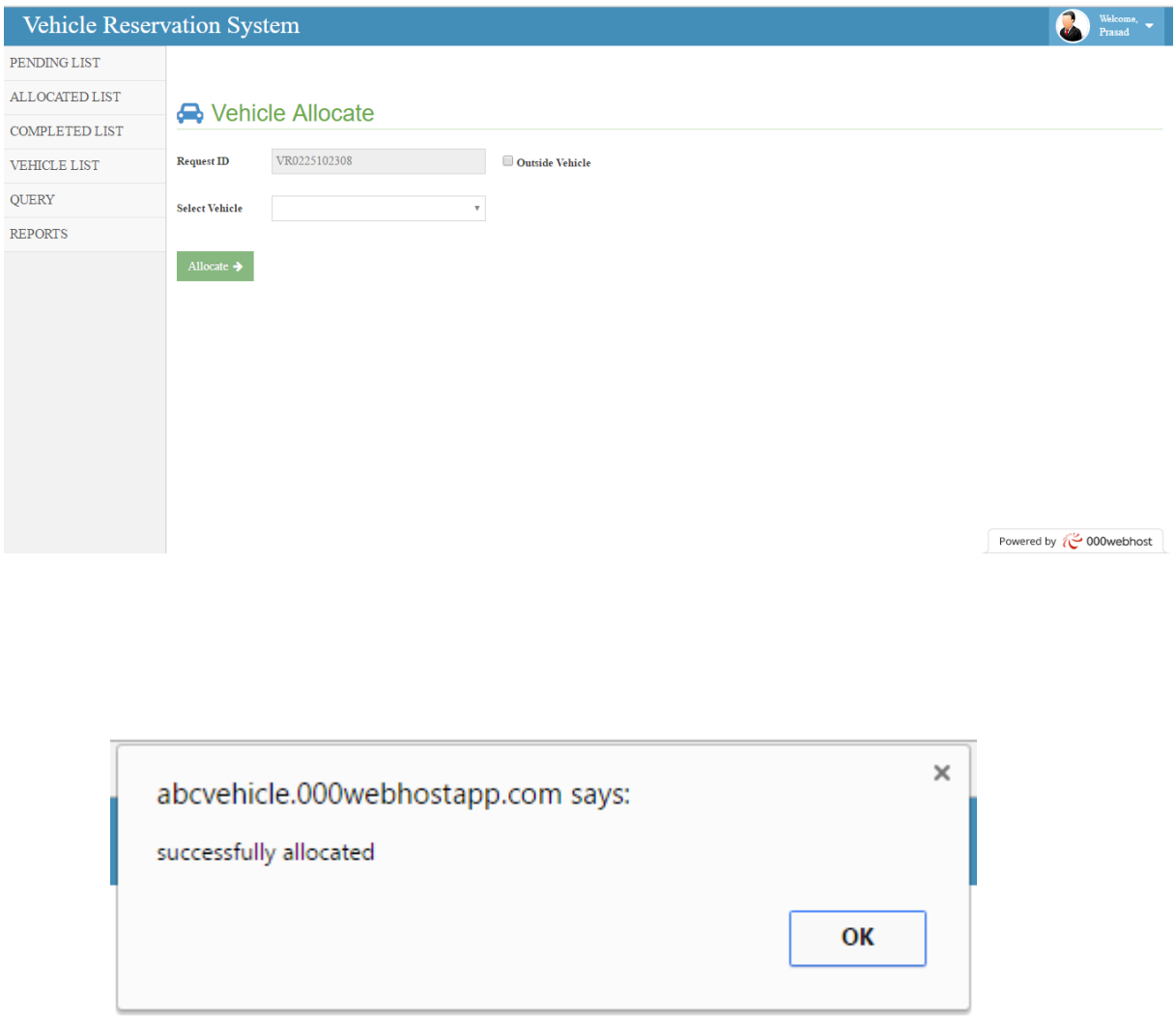
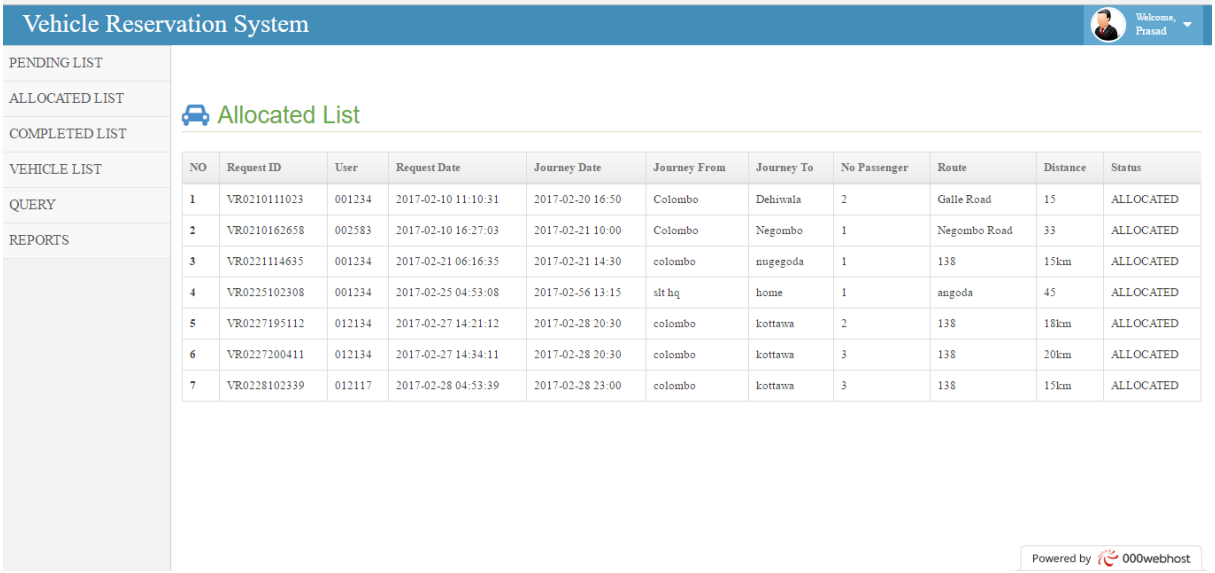


Figure 17: Vehicle Allocation page

4) Admin can check for the allocated list.(shown in figure 17.)



NO	Request ID	User	Request Date	Journey Date	Journey From	Journey To	No Passenger	Route	Distance	Status
1	VR0210111023	001234	2017-02-10 11:10:31	2017-02-20 16:50	Colombo	Dehiwala	2	Galle Road	15	ALLOCATED
2	VR0210162658	002583	2017-02-10 16:27:03	2017-02-21 10:00	Colombo	Negombo	1	Negombo Road	33	ALLOCATED
3	VR0221114635	001234	2017-02-21 06:16:35	2017-02-21 14:30	colombo	nugegoda	1	138	15km	ALLOCATED
4	VR0225102308	001234	2017-02-25 04:53:08	2017-02-56 13:15	slt hq	home	1	angoda	45	ALLOCATED
5	VR0227195112	012134	2017-02-27 14:21:12	2017-02-28 20:30	colombo	kottawa	2	138	18km	ALLOCATED
6	VR0227200411	012134	2017-02-27 14:34:11	2017-02-28 20:30	colombo	kottawa	3	138	20km	ALLOCATED
7	VR0228102339	012117	2017-02-28 04:53:39	2017-02-28 23:00	colombo	kottawa	3	138	15km	ALLOCATED

Figure 18: Allocated list page

5) Admin can generate reports. (Shown in figure 19.)

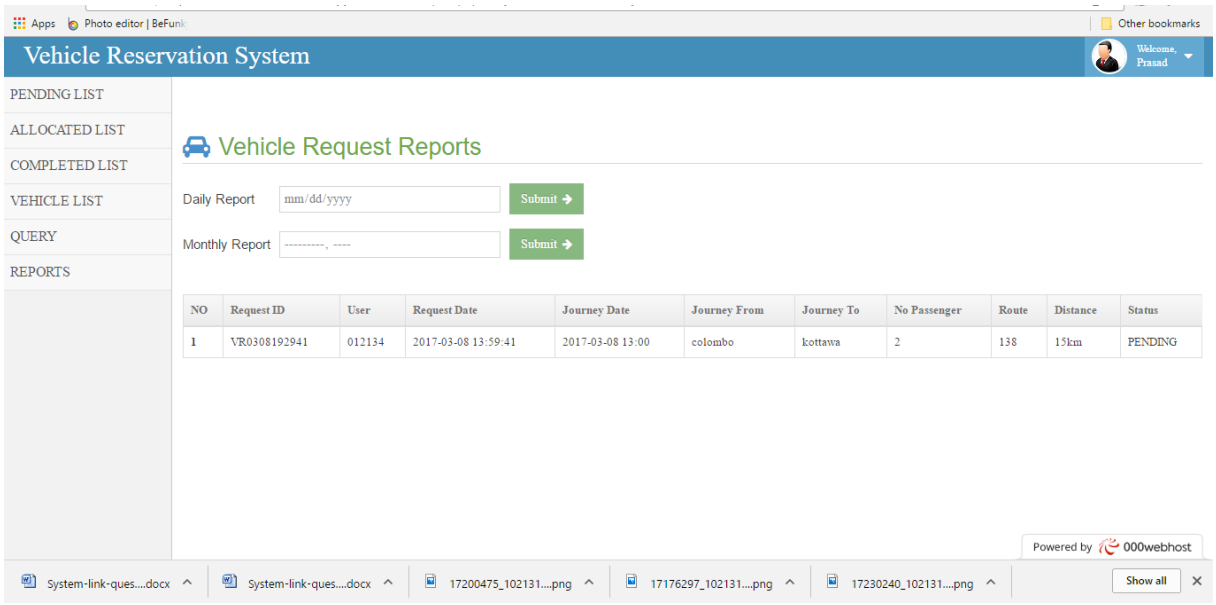


Figure 19: Reports page

6) Admin can query the database. (Shown in Figure 20.)

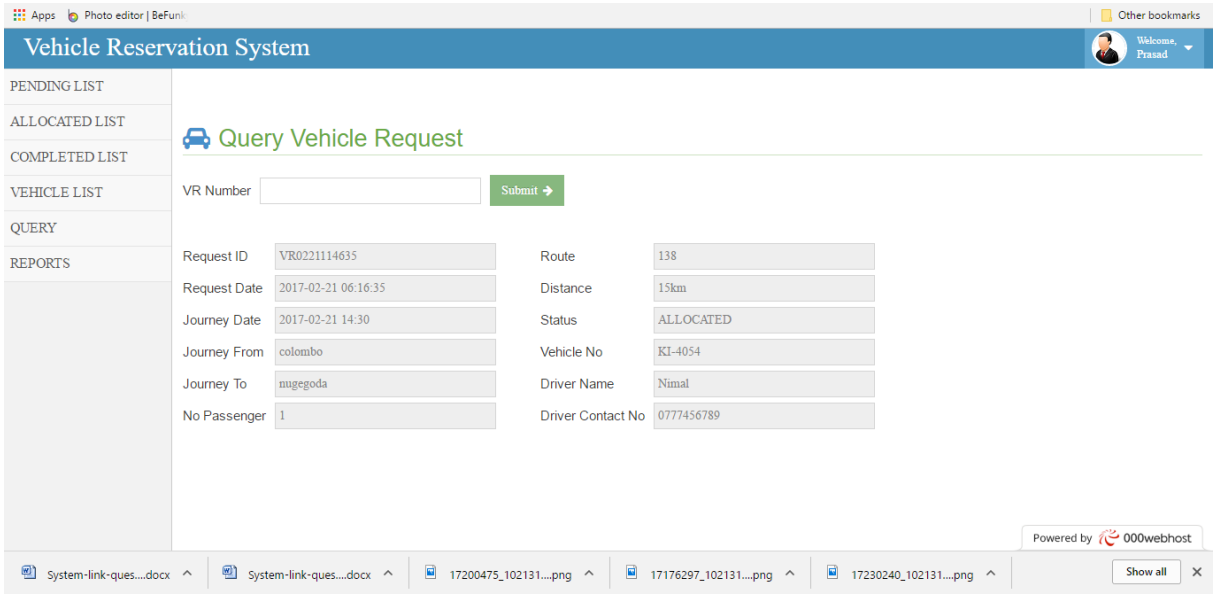


Figure 20: Query page

7) Admin can check for his own profile and do changes.(Shown in Figure 21.)

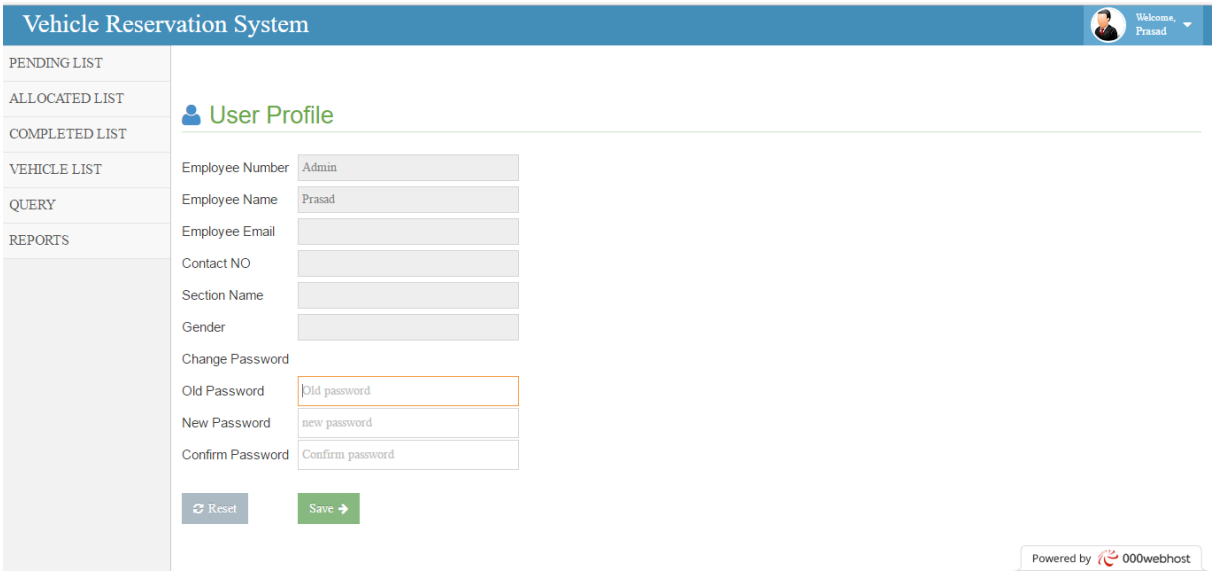


Figure 21: Admin profile page

Driver

1) Driver can change their status only through mobile app.(Shown in figure 22.)

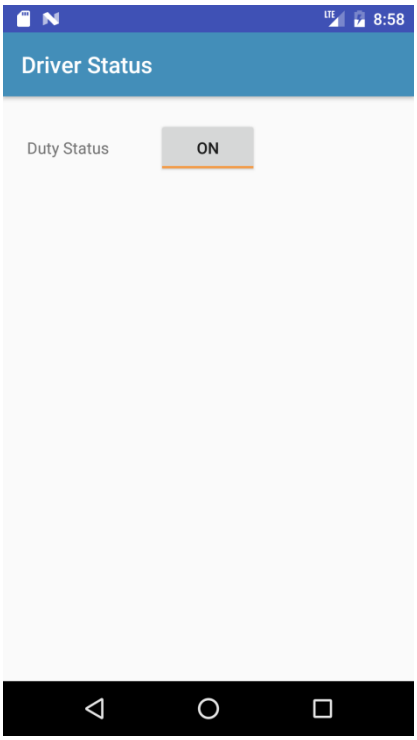


Figure 22: Driver status page

2) Driver can complete the journey through mobile app after they finished the journey.(Shown in figure 23.)

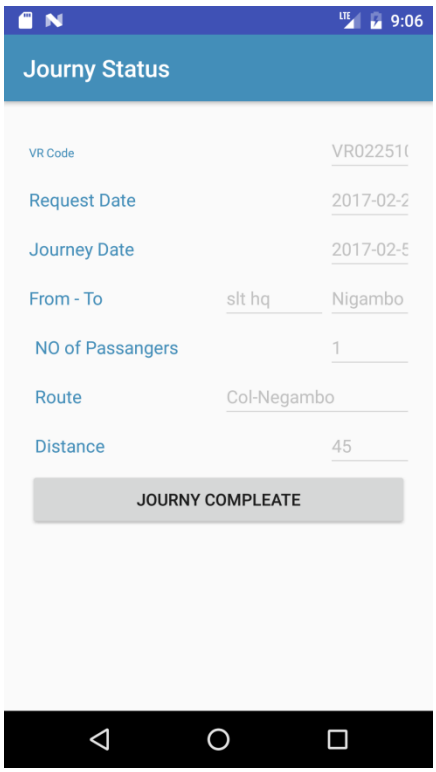


Figure 23: Driver Journey Status page

Design documentation

Appendix C

Process Flow Diagram

A process flow diagram (PFD) is a diagram commonly used in chemical and process engineering to indicate the general flow of plant processes and equipment. The PFD displays the relationship between major equipment of a plant facility and does not show minor details such as piping details and designations.

(Reservation/client side/Through Website)

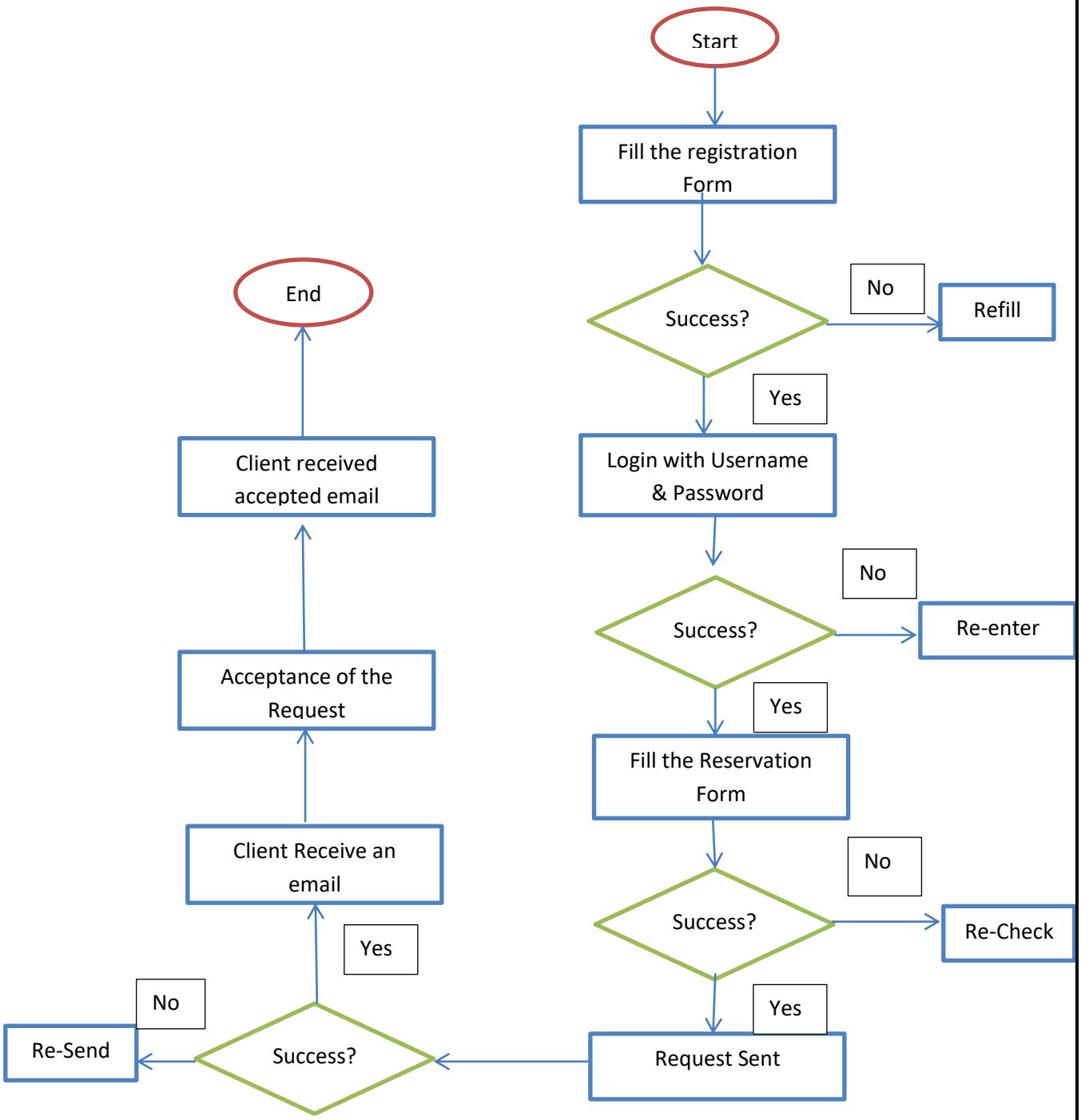


Figure 24: Client Reservation through website process flow digaram

Figure 24 shows the client side reservation through website. Client fills the registration form first to get a login. Then he/she can log on to the system using the username and password. When the log in is success it auto redirect to the reservation form. User then fills the reservation form and sent it to admin for allocation. User gets an email if the request sent successfully. An also User get an email with the allocation details when the admin allocate a vehicle.

(Reservation/client side/Through mobile app)

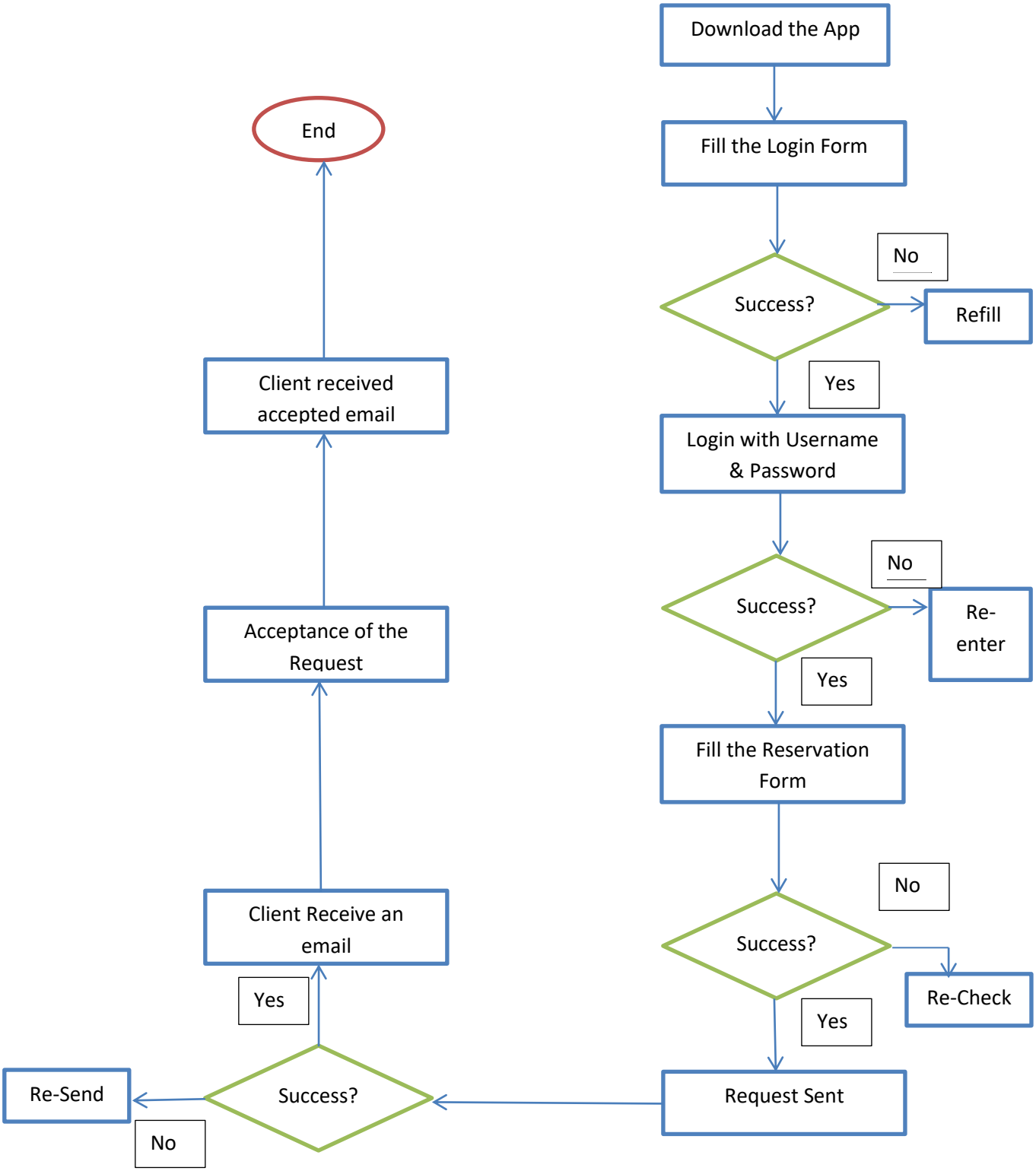


Figure 25: Client Reservation through mobile app process flow digaram

Figure 25 shows the same procedure stated above in figure 24 but with the mobile app.

(Reservation acceptance/Admin side/Through PC)

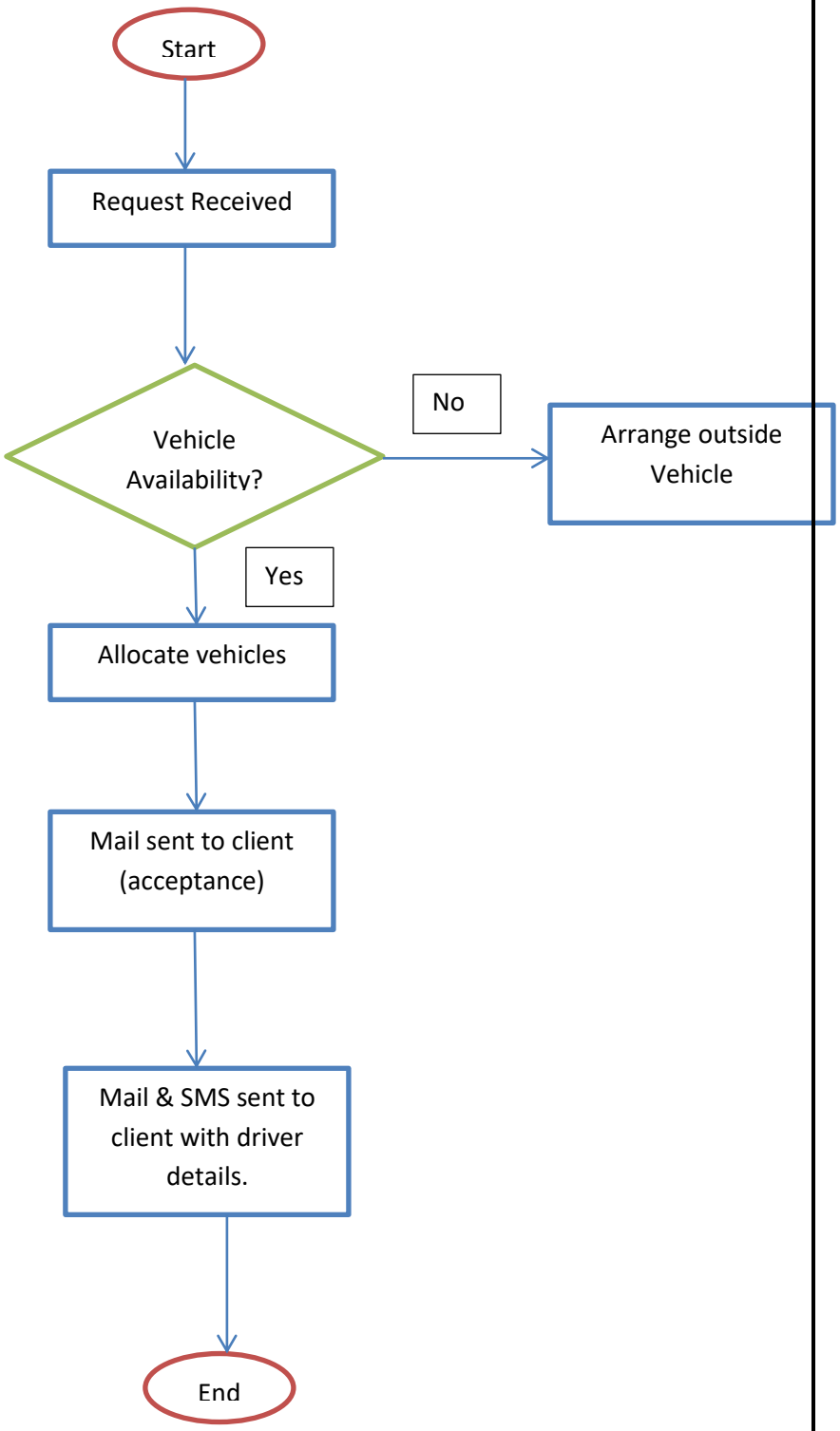


Figure 26: Admin side reservation acceptance through website

Figure 26 shows the admin procedure of the system. When a reservation made, admin can check that reservation and can check availability of the vehicles and then if a vehicle is available he can allocate it and the allocation is successful. User received an email with the allocation details. If a vehicle is not available in the pool, admin can call a outside vehicle company and reserve a vehicle and admin can add those details to the system and allocate the vehicle.

(Reservation acceptance/Admin side/Through Mobile App)

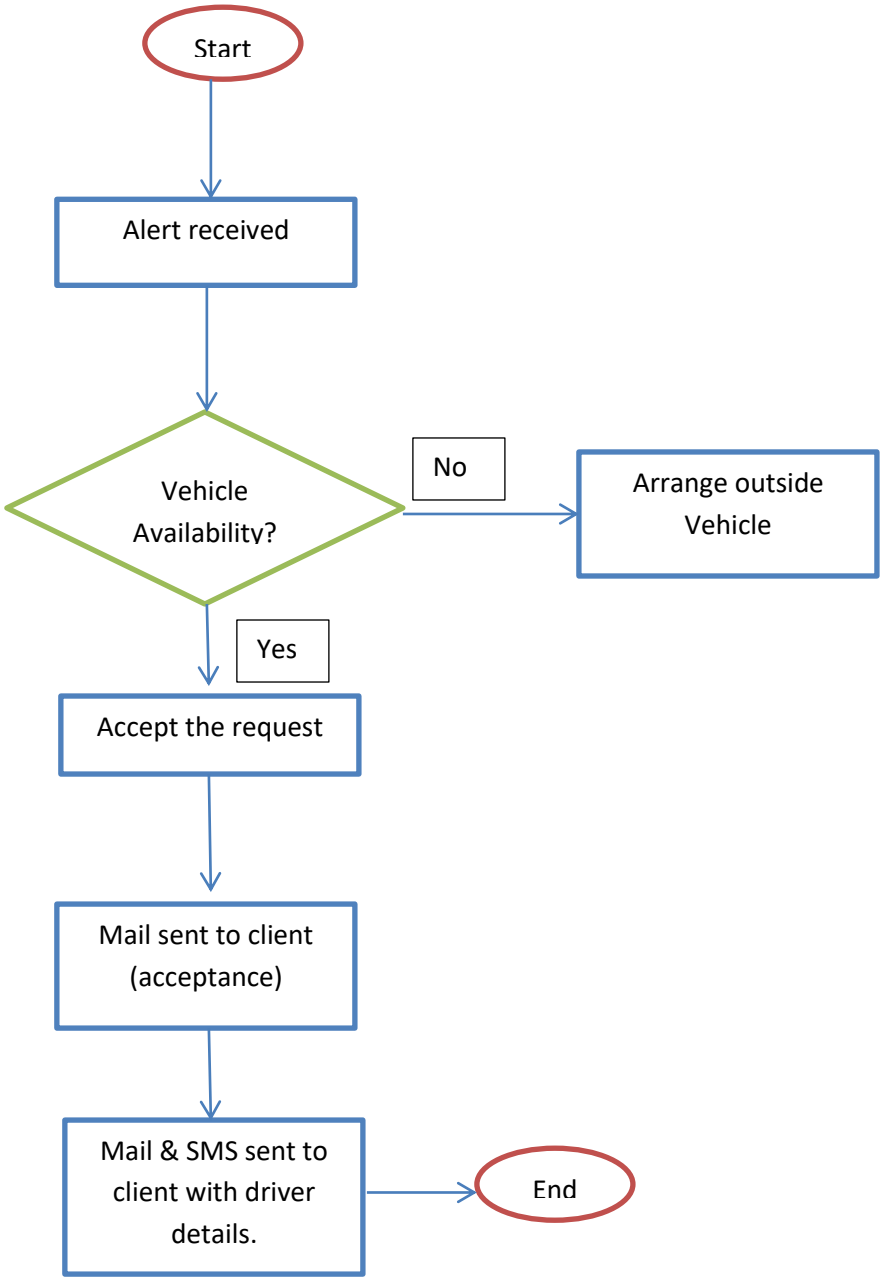


Figure 27: Admin side reservation acceptance through mobile app

Figure 27 shows the same procedure stated above in figure 27 but with the mobile app.

User interfaces

❖ Client Side web portal

- ***The first page appears for the client***

Figure 28: Main interface page

- ***Registration page***

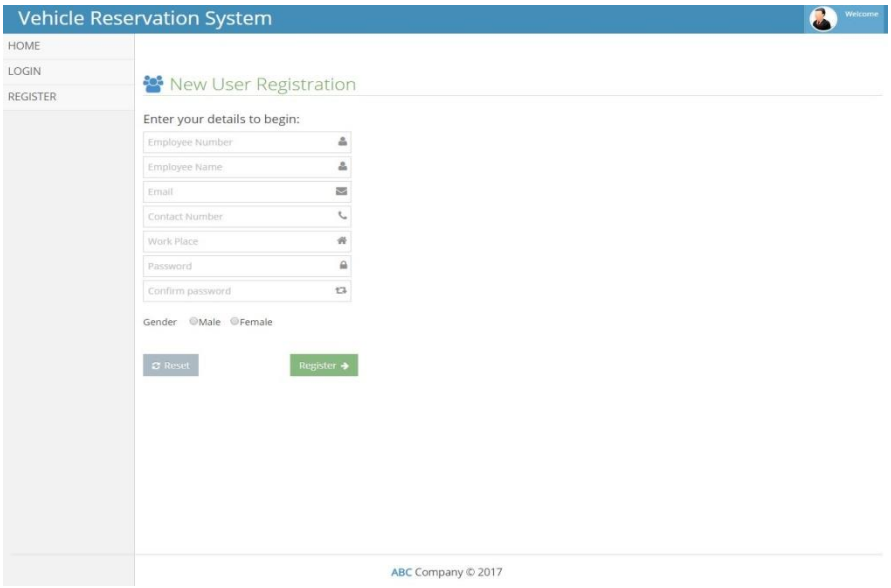


Figure 29: User registration page

- ***Login page***

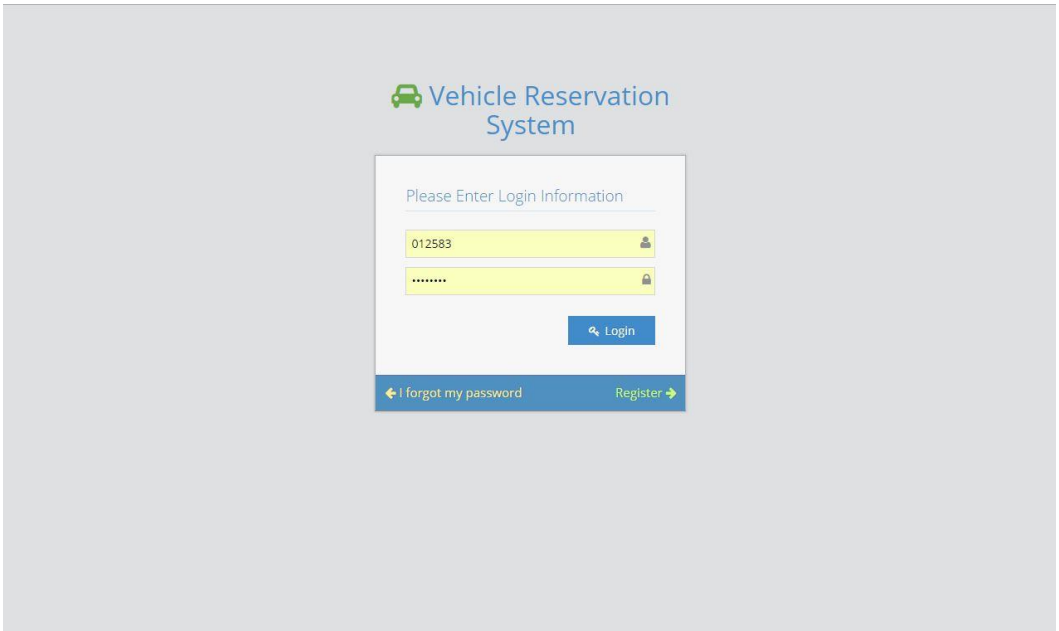


Figure 30: Login Page

- ***Reservation page***

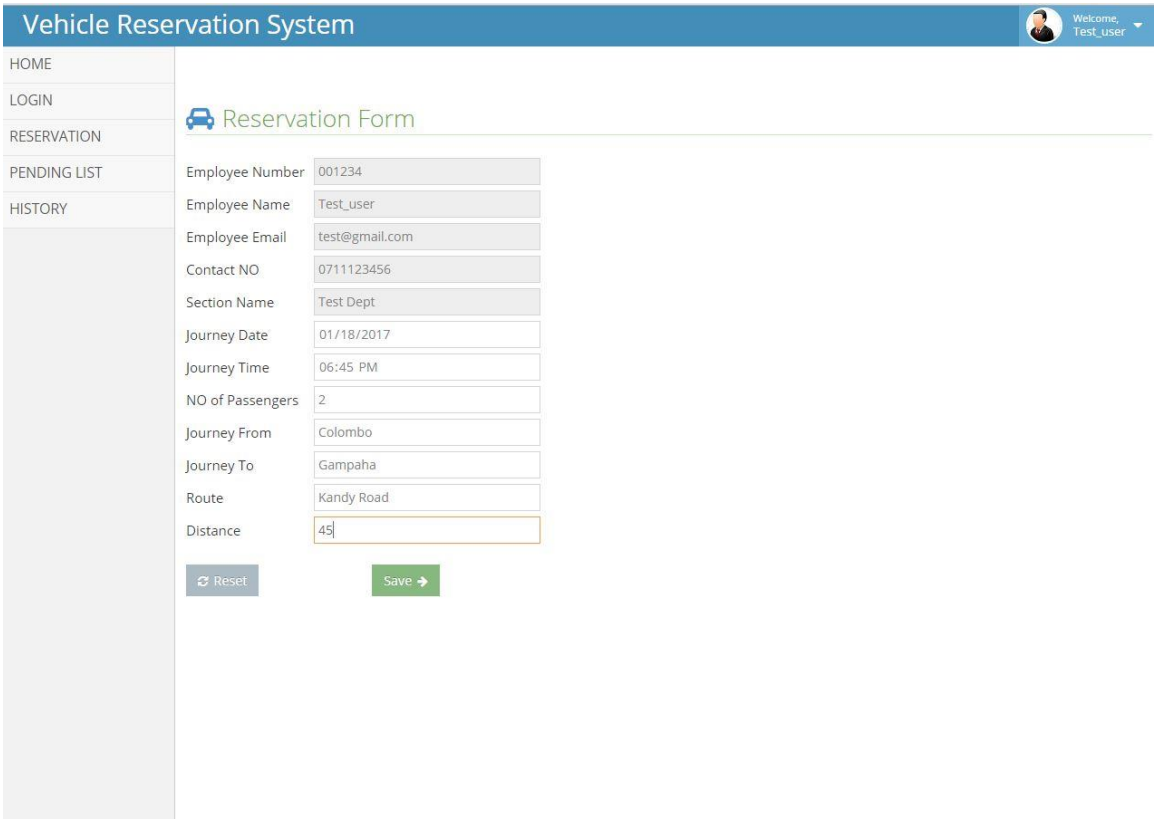


Figure 31: Vehicle reservation page

- ***History page***

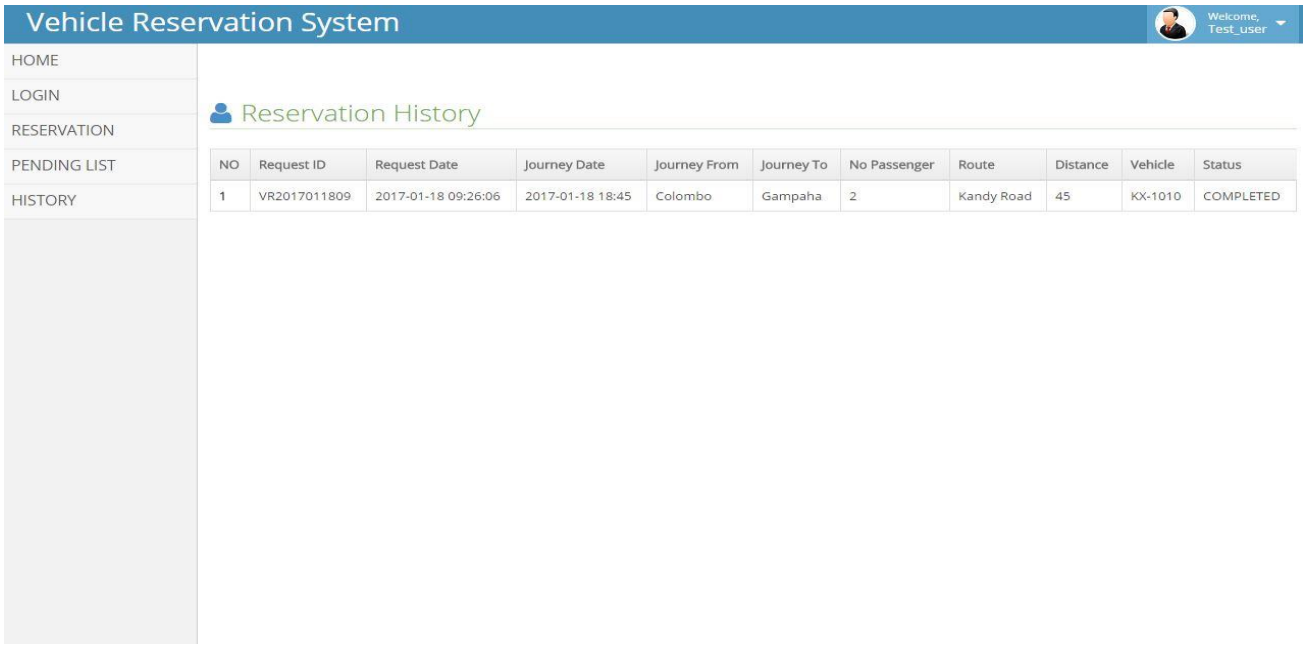


Figure 32: Reservation history page

- ***Pending page***

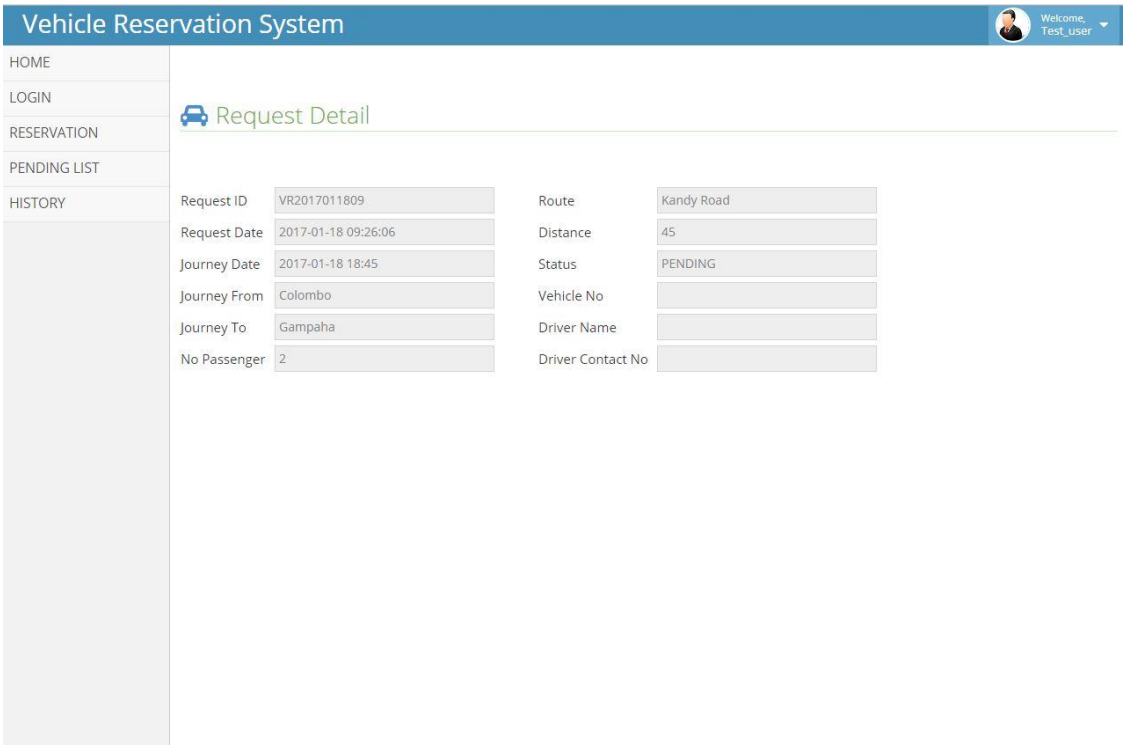


Figure 33: Pending list page

- ***Profile page***

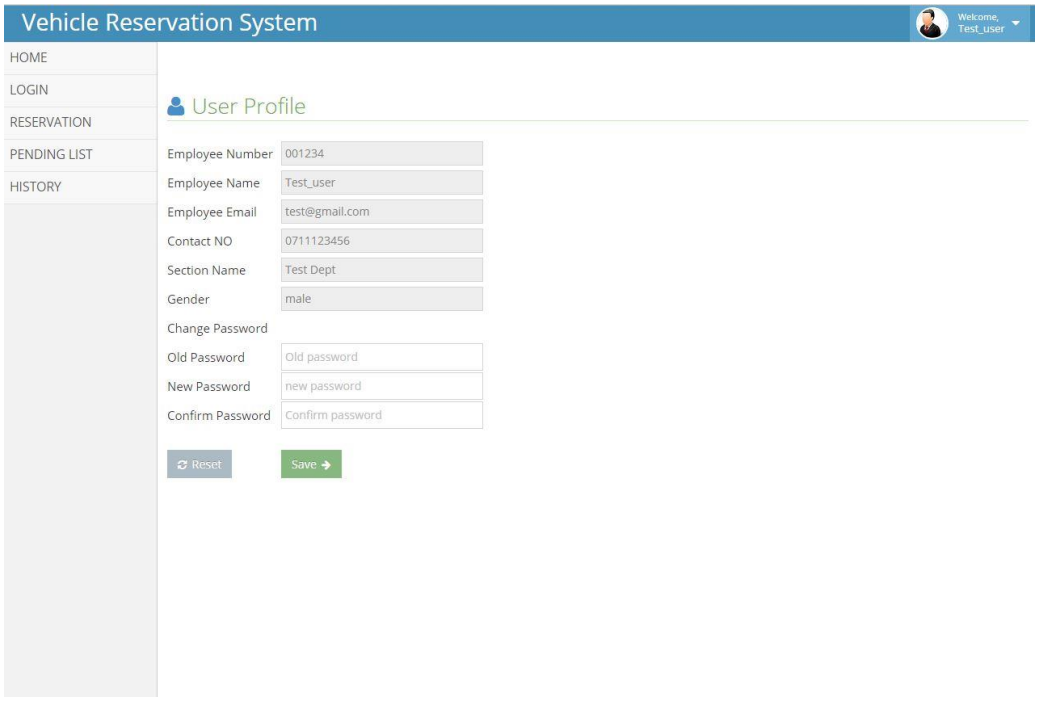


Figure 34: User profile page

❖ Administrator Side web portal

- ***First appearance***

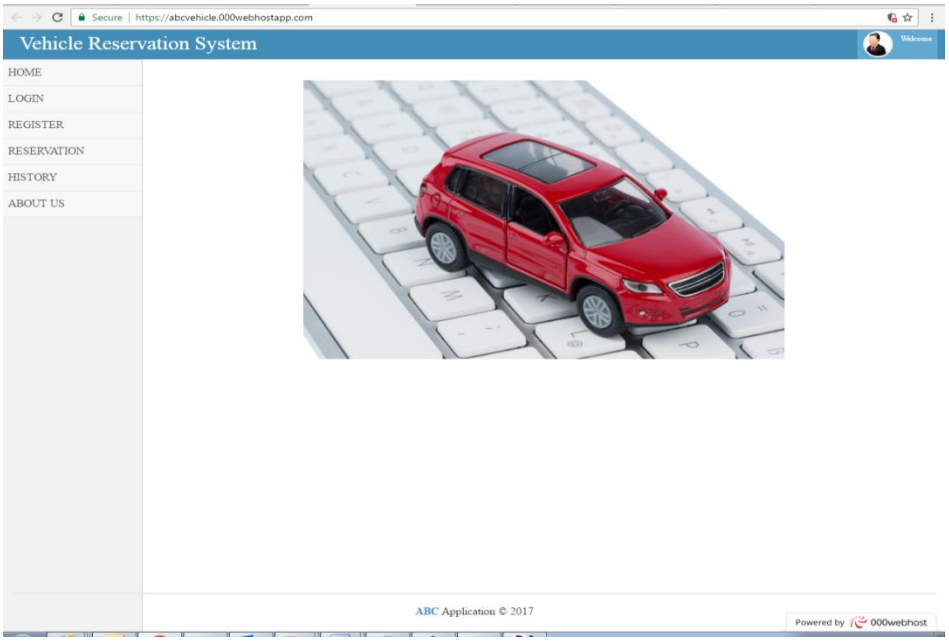


Figure 35: Main interface page

- ***Admin login***

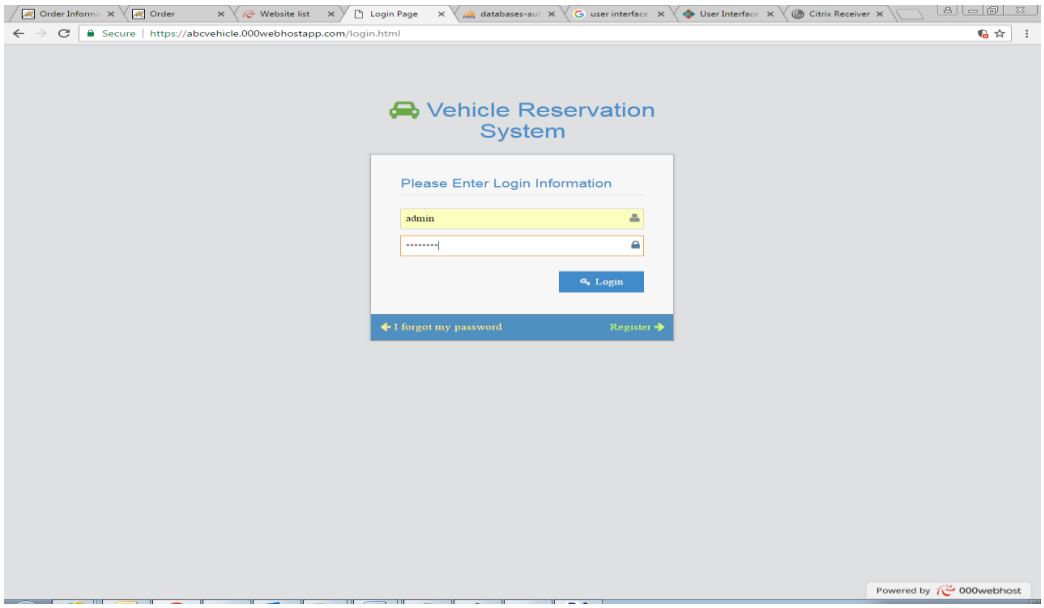


Figure 36: Admin login page

- ***Pending List***

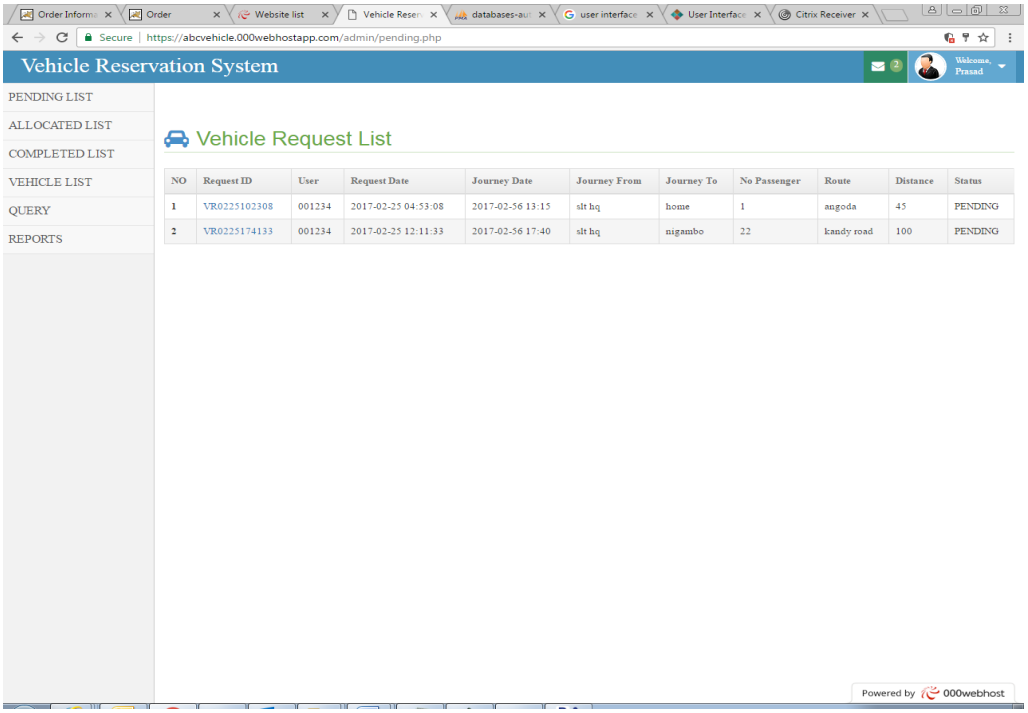


Figure 37: Admin pending list page

- ***Allocated list***

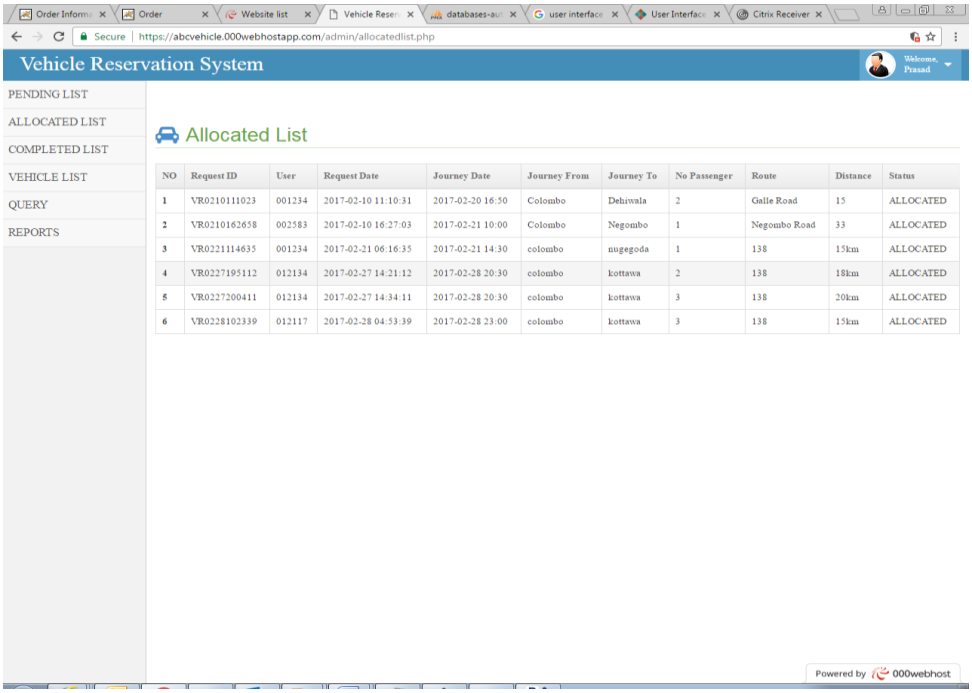


Figure 38: Admin allocated list

- ***Admin Profile***

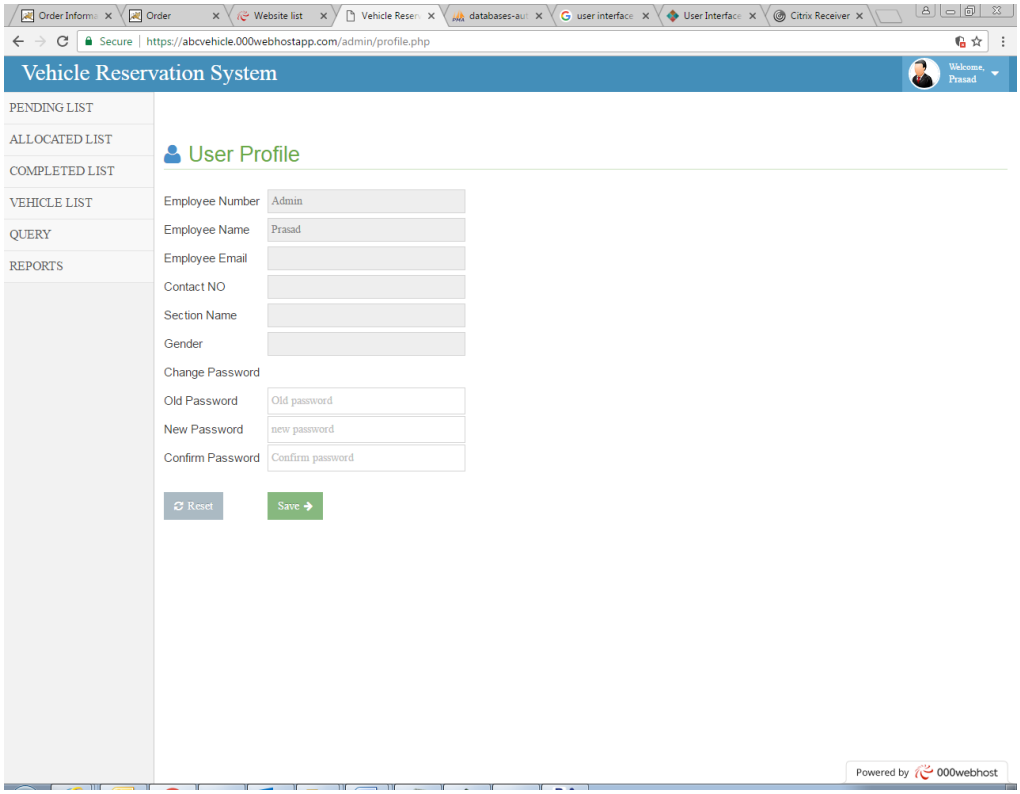


Figure 39: Admin profile

❖ Client Side mobile app

- ***Registration page***

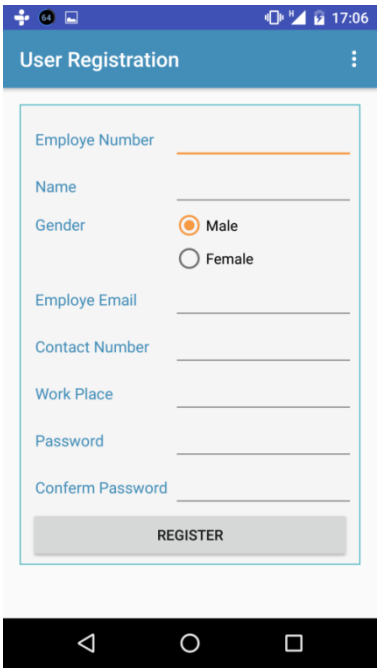


Figure 40: Registration page (mobile app)

- ***Login page***

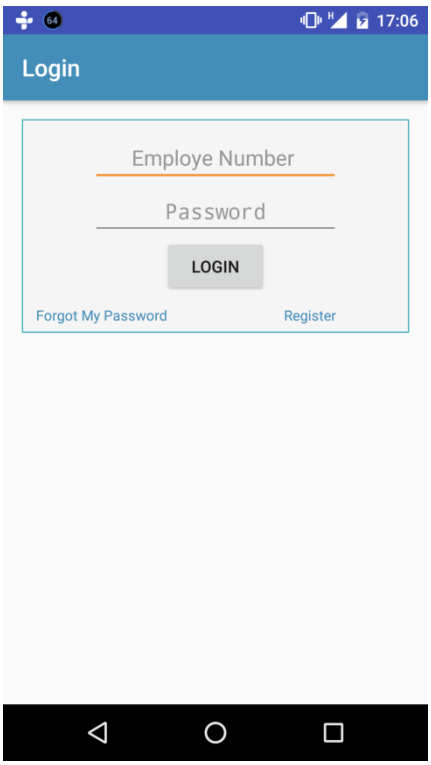


Figure 41: Login page (mobile app)

- ***Reservation page***

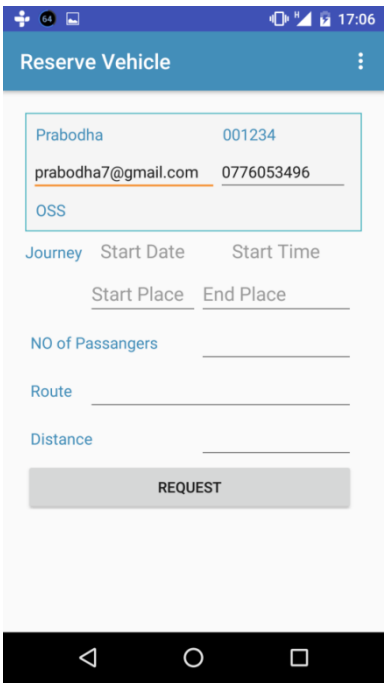


Figure 42: Reservation page (mobile app)

- ***Options available for the user***

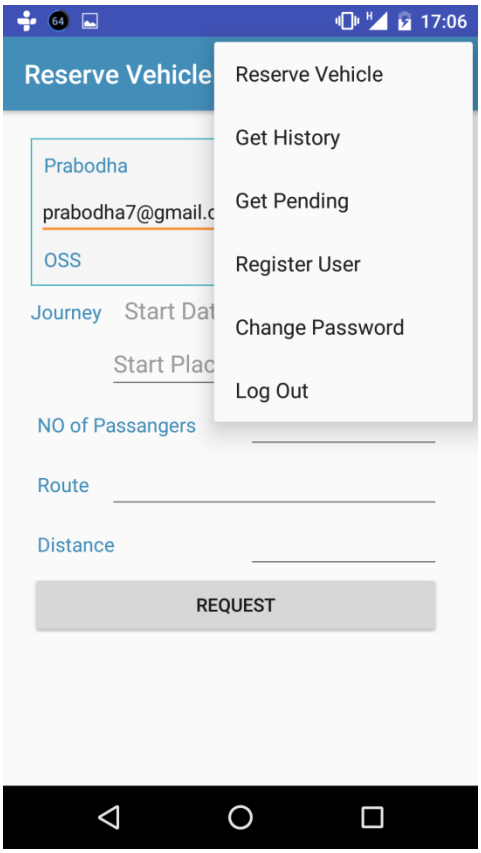


Figure 43: Options available for the user (mobile app)

❖ Admin side of the mobile app

- ***Approve reservation***

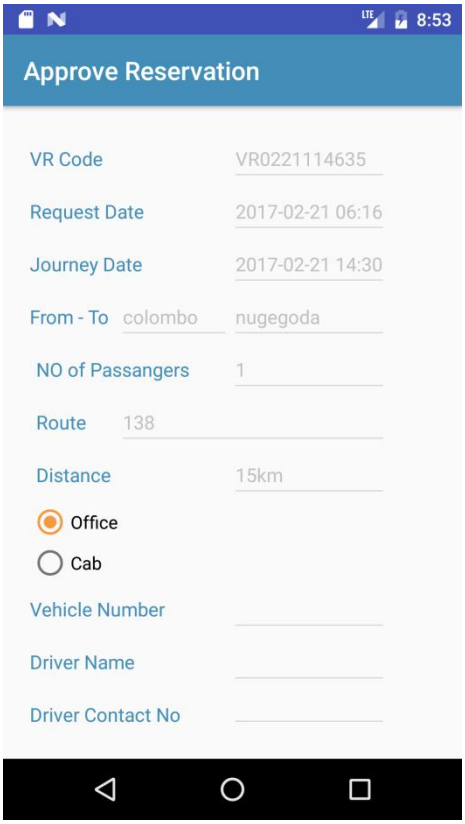


Figure 44: Reservation approval page (mobile app)

❖ Driver Side mobile app

- *Driver Status*

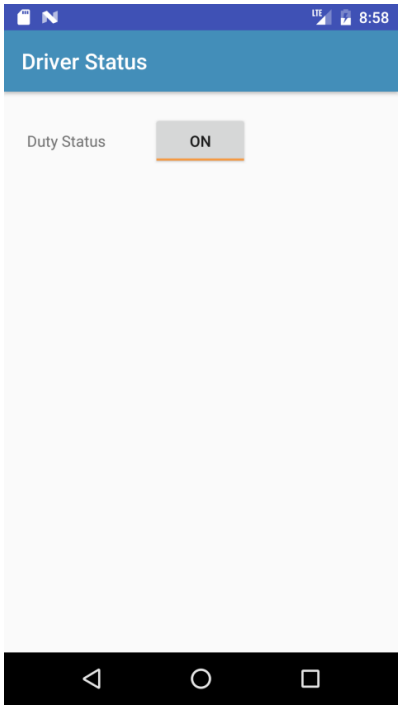


Figure 45: Driver status (mobile app)

- *Journey ending*

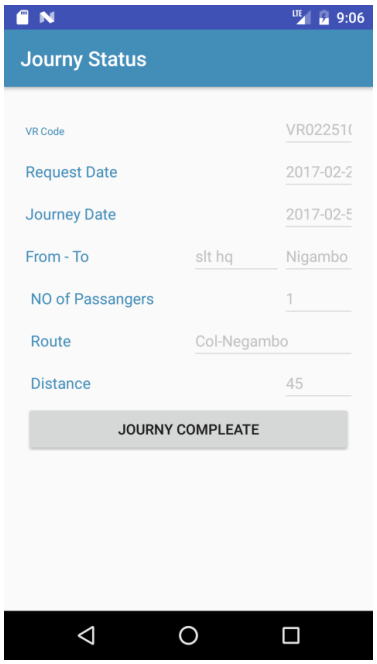


Figure 46: Journey Status page (mobile app)

Test Results

Appendix D

Unit Testing

User side Scenarios

- **Reservation**

Test cases(input data)	Expected result	Actual result
Click reservation button.	Redirect to reservation form.	Redirect to reservation form.
Click Save button without entering the details.	Error occurrence:"Pls fill out the fields". And user cannot save.	Error occurrence:"Pls fill out the fields". And user cannot save.
Click save button after entering the details.	Message pop up: "Successfully submitted" and reservation request goes to admin.	Message pop up: "Successfully submitted" and reservation request goes to admin.
Click Reset button	Entered fields reset.	Entered fields reset.

Table 7: Unit testing (Reservation)

- **History**

Test cases(input data)	Expected result	Actual result
Click History button.	Redirect to history of user reservation.	Redirect to history of user reservation.

Table 8: Unit testing (History)

- **Profile**

Test cases(input data)	Expected result	Actual result
Click Profile button.	Redirect to Profile form.	Redirect to reservation form.
Click Save button without entering the details.	Error occurrence:"Pls fill out the fields". And user cannot save.	Error occurrence:"Pls fill out the fields". And user cannot save.
Click save button after entering the details.	Message pop up: "Successfully Password changed".	Message pop up: "Successfully Password changed".
Click Reset button when fields entered.	Entered fields reset.	Entered fields reset.

Table 9: Unit testing (Profile)

Admin side scenarios

- **Login**

Test cases(input data)	Expected result	Actual result
Click login button without entering username or password.	Error occurrence:"Pls enter a valid username or password" and user cannot log on to the system	Error occurrence:"Pls enter a valid username or password" and user cannot log on to the system
Click login button only entering the username.	Error occurrence: "Wrong username or password." And user cannot log on to the system.	Error occurrence: "Wrong username or password." And user cannot log on to the system.
Click login button only entering the password.	Error occurrence: "Wrong username or password." And user cannot log on to the system.	Error occurrence: "Wrong username or password." And user cannot log on to the system.
Click login button entering wrong username and password.	Error occurrence: "Wrong username or password." And user cannot log on to the system.	Error occurrence: "Wrong username or password." And user cannot log on to the system.
Click login button entering correct username and password.	Successfully login to the system.	Successfully login to the system.

Table 10: Unit testing (Admin side login)

- **Pending List**

Test cases(input data)	Expected result	Actual result
Click on the Request ID.	Redirect to vehicle allocation form.	Redirect to vehicle allocation form.
Click allocate button without selecting a Vehicle.	Error occurrence: "Please select an item in the filed". And admin cannot allocate.	Error occurrence: "Please select an item in the filed". And admin cannot allocate.
Tick the "Outside" checkbox if pool vehicles are not available.	Redirect to Outside vehicle Allocation form.	Redirect to Outside vehicle Allocation form.
Click save button in the allocation form.	Error occurrence:"Pls fill out the fields". And admin cannot allocate.	Error occurrence:"Pls fill out the fields". And admin cannot allocate.
Enter details and click Save button.	Message pop up "Successfully allocated" and allocate to the user and user get the mail and db updated.	Message pop up "Successfully allocated" and allocate to the user and user get the mail and db. Updated.

Table 11: Unit testing (Admin side pending list)

- **Allocated List**

Test cases(input data)	Expected result	Actual result
Click allocated list button.	Redirect to already allocated requests .	Redirect to already allocated requests .

Table 12: Unit testing (Admin side allocated list)

- **Vehicle list**

Test cases(input data)	Expected result	Actual result
Click Vehicle list button.	Redirect to already all pool vehicles.	Redirect to already all pool vehicles.

Table 13: Unit testing (Admin side vehicle list)

- **Query**

Test cases(input data)	Expected result	Actual result
Click Query button.	Redirect to Query vehicle requests.	Redirect to Query vehicle requests.
Click without typing the VR number.	Error Occures;"Pls fill out the field."	Error Occures;"Pls fill out the field."
Enter the VR number and submit.	Show the Vehicle request details.	Show the Vehicle request details.

Table 14: Unit testing (Admin side query)

- **Reports**

Test cases(input data)	Expected result	Actual result
Click Reports button.	Redirect to a page where you can get various reports.	Redirect to a page where you can get various reports.
Submit without filling the fields.	Error occurrence:"Pls fill out the fields".	Error occurrence:"Pls fill out the fields".
Submit after filling the fields.	Data retrieved from the Database and shows.	Data retrieved from the Database and shows.

Table 15: Unit testing (Admin side reports)

Driver side scenario

- **Driver status**

Test cases(input data)	Expected result	Actual result
Driver can set their status to ON or OFF.	Admin can check if the driver is duty on or off.	Admin can check if the driver is duty on or off.

Table 16: Unit testing (Driver side driver status)

- **Journey status**

Test cases(input data)	Expected result	Actual result
Click Journey completed button.	Database updated and the reservation goes to completed list and the driver again available.	Database updated and the reservation goes to completed list and the driver again available.

Table 17: Unit testing (Driver side journey status)

Acceptance Testing

The result from the Questionnaire is evaluated. Following are sample Questionnaire result I got.

Designation: - IT & network officer

1) How likely is it that you would recommend this system to a friend or company?

- Not at all ☐
- Recommend ☐
- Highly Recommend ☒

2) How satisfied are you with the reliability of this system?

- Extremely satisfied ☐
- Very Satisfied ☒
- Somewhat Satisfied ☐
- Not so Satisfied ☐
- Not at all satisfied

3) How satisfied are you with the security of this system?

- Extremely satisfied ☐
- Very Satisfied ☒
- Somewhat Satisfied ☐
- Not so Satisfied ☐
- Not at all satisfied ☐

4) How satisfied are you with this system's ease of use?

- Extremely satisfied ☒
- Very Satisfied ☐
- Somewhat Satisfied ☐
- Not so Satisfied ☐
- Not at all satisfied ☐

5) How satisfied are you with the look and feel of this system?

- Extremely satisfied ☐
- Very Satisfied ☒
- Somewhat Satisfied ☐
- Not so Satisfied ☐
- Not at all satisfied ☐

6) Is it easy to learn to use this system?

- Yes ☒
- No ☐

7) Overall, how would you rate?

- Excellent ☒
- Good ☐
- Average ☐
- Poor ☐

8) Do you have any Thought on how to improve this system? State Below.

…….
…….
…….

Designation: - Engineer

9) How likely is it that you would recommend this system to a friend or company?

- Not at all ☒
- Recommend ☐
- Highly Recommend ☒

10) How satisfied are you with the reliability of this system?

- Extremely satisfied ☐
- Very Satisfied ☒
- Somewhat Satisfied ☐
- Not so Satisfied ☐
- Not at all satisfied ☐

11) How satisfied are you with the security of this system?

- Extremely satisfied ☐
- Very Satisfied ☒
- Somewhat Satisfied ☐
- Not so Satisfied ☐
- Not at all satisfied ☐

12) How satisfied are you with this system's ease of use?

- Extremely satisfied ☐
- Very Satisfied ☒
- Somewhat Satisfied ☐
- Not so Satisfied ☐
- Not at all satisfied ☐

13) How satisfied are you with the look and feel of this system?

- Extremely satisfied ☒
- Very Satisfied ☐
- Somewhat Satisfied ☐
- Not so Satisfied ☐
- Not at all satisfied ☐

14) Is it easy to learn to use this system?

- Yes ☒
- No ☐

15) Overall, how would you rate?

- Excellent ☒
- Good ☐
- Average ☐
- Poor ☐

16) Do you have any Thought on how to improve this system? State Below.

It is very much user friendly if it is possible to merge Reservation, Pending list and History to a single screen.

Designation: - Customer care officer

17) How likely is it that you would recommend this system to a friend or company?

- Not at all ☐
- Recommend ☐
- Highly Recommend ☒

18) How satisfied are you with the reliability of this system?

- Extremely satisfied ☒
- Very Satisfied ☐
- Somewhat Satisfied ☐
- Not so Satisfied ☐
- Not at all satisfied ☐

19) How satisfied are you with the security of this system?

- Extremely satisfied ☒
- Very Satisfied ☐
- Somewhat Satisfied ☐
- Not so Satisfied ☐
- Not at all satisfied ☐

20) How satisfied are you with this system's ease of use?

- Extremely satisfied ☒
- Very Satisfied ☐
- Somewhat Satisfied ☐
- Not so Satisfied ☐
- Not at all satisfied ☐

21) How satisfied are you with the look and feel of this system?

- Extremely satisfied ☒
- Very Satisfied ☐
- Somewhat Satisfied ☐
- Not so Satisfied ☐
- Not at all satisfied ☐

22) Is it easy to learn to use this system?

- Yes ☒
- No ☐

23) Overall, how would you rate?

- Excellent ☒
- Good ☐
- Average ☐
- Poor ☐

24) Do you have any Thought on how to improve this system? State Below.

The overall results of the Acceptance Test are as follow.

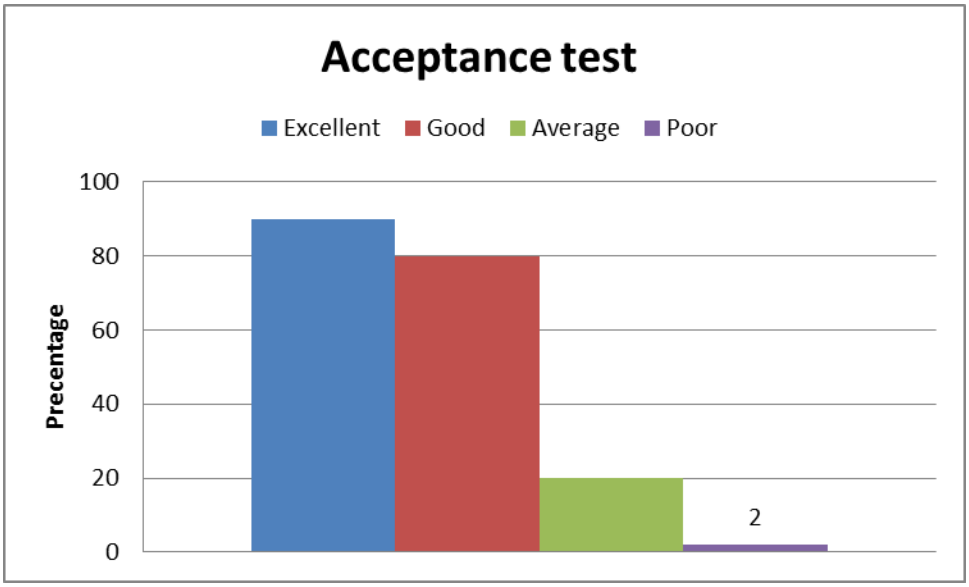


Figure 47: Acceptance Testing result