

# **INTELLIGENT COMPLEMENTARY RIDE-SHARING SYSTEM**

## **(User Profile Management)**

**Project ID: CDAP 19-055**

### **Software Requirements Specification**

**B.Sc. Special (Honors) Degree in Information Technology**  
**Specializing in Software Engineering**

**Submitted on: 29th of April 2019**

## Declaration

I hereby declare that this is my work and this document does not incorporate without acknowledgment any material previously submitted for a degree or diploma in any other university or Institute of higher learning and to the best of our knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgment is made in the text.

Name	Registration Number	Signature
V.A.Wickramasinghe	IT16030190	

The above candidates are carrying out research for the undergraduate Dissertation under my supervision.

Supervisor

Date

.....

Dr. Janaka Wijekoon

.....

# Table of Contents

Declaration.....	2
LIST OF FIGURES .....	5
LIST OF TABLES.....	5
1. Introduction .....	6
1.1 Purpose.....	6
1.2 Scope .....	6
1.3 Definitions and Abbreviations .....	7
1.4 Overview .....	7
2. Overall Descriptions .....	8
2.1 Product Perspective .....	8
2.1.1 System Interfaces.....	9
2.1.2 User Interfaces .....	9
2.1.3 Hardware Interfaces .....	12
2.1.4 Software Interfaces.....	12
2.1.5 Communication Interfaces.....	13
2.1.6 Memory Constraints .....	13
2.1.7 Operations .....	13
2.1.8 Site Adaptation Requirements.....	13
2.2 Product Functions.....	14
2.2.1 Use Case Scenario .....	14
2.2.2 Use Case Diagram .....	19
2.3 User Characteristics .....	19
2.4 Constraints .....	20
2.5 Assumptions and Dependencies .....	20
2.6 Apportioning of Requirements .....	20
3. Specific requirements.....	21
3.1 External interface requirements .....	21
3.1.1 User Interfaces .....	21
3.1.2 Hardware Interfaces .....	25
3.1.3 Software Interfaces.....	26
3.1.4. Communication Interfaces.....	26

3.2. Classes/Objects.....	27
3.3 Performance requirements .....	27
3.4 Design constraints .....	27
3.5 Software system attributes .....	28
3.5.1 Reliability.....	28
3.5.2 Availability.....	28
3.5.3 Security .....	28
3.5.4 Maintainability .....	28
3.6. Other Requirements .....	29
4. Supporting Information .....	30
4.1 Appendices.....	30
4.2. References .....	30

## LIST OF FIGURES

*Figure 2.1 Home Screen of ICRSS*

*Figure 2.2 Phone Number Verification*

*Figure 2.3 User Profile*

*Figure 2.4 Adding Bank Details*

*Figure 2.5 Adding Vehicle Details*

*Figure 2.5 Suggested Driver List*

*Figure 2.2.1 Use case Diagram*

*Figure 3.2 Class diagram of UPM*

## LIST OF TABLES

*Table 2.2.1.1 Register to the system Scenario*

*Table 2.2.1.2 Log in to the system Scenario*

*Table 2.2.1.3 Search for drivers Scenario*

*Table 2.2.1.4 Report drivers Scenario*

*Table 2.2.1.5 Remove Report drivers Scenario*

*Table 2.2.1.6 Get the current location of the drivers Scenario*

*Table 2.2.1.7 Update account Scenario*

*Table 2.2.1.8 Delete account Scenario*

*Table 3.1.1 Send Verification Code Interface*

*Table 3.1.2 Login Interface*

*Table 3.1.3 Verify the code Interface*

*Table 3.1.4 Add details to profile Interface*

*Table 3.1.5 Add details to profile Interface*

*Table 3.1.6 Add payment details Interface*

*Table 3.1.7 Add vehicle details Interface*

# 1. Introduction

## 1.1 Purpose

The main purpose of this document is to give an overall overview of the research component “User Profiling Management”. Intelligent Complementary Ride-Sharing System which was proposed as a solution to reduce traffic congestion in urban areas [1], User Profiling Management plays a major role in selecting the most suitable driver for the passenger. This SRS document gives a detailed explanation of how user profiling is utilized in our research. It describes the scope of the component and the effect it has on every stakeholder of the system. This document can be used as a reference by both the client and developers for any clarification on the end product.

## 1.2 Scope

As mentioned earlier, this document covers the specifications brought out by the “User Profiling Management”. This component is subdivided into;

- User Profiling
- Ride-Matching Algorithm

In the User Profiling subcomponent, the personal information of the user, personal interests and bank details are collected upon the registration. To be a valid user, first the mobile number is verified to avoid creating spam/fake accounts in the system. For every individual, the profile is maintained with personal information and relevant rating. In the Ride-Matching Algorithm, the most suitable driver list is selected for the passenger upon searching using the passenger's interests. The benefit of this is to provide the passenger with the best-suited driver for the preference. The “+Go” application will achieve all these mentioned goals and features in the implementation.

### 1.3 Definitions and Abbreviations

<b>SRS</b>	<b>Software Requirements Specification</b>
<b>ICRSS</b>	<b>Intelligent Complementary Ride Sharing System</b>
<b>UPM</b>	<b>User Profiling Management</b>
<b>UI</b>	<b>User Interface</b>
<b>OS</b>	<b>Operating System</b>
<b>API</b>	<b>Application Programming Interface</b>

### 1.4 Overview

The proposed application is to overcome the failures and loopholes in the existing ride-sharing applications prevailing in Sri Lanka [2]. In the initial phase, we have targeted to cover the Colombo district and later on moving onto the other areas.

The first part of the document brings you comprehensive detail about the UPM component of the proposed application. In the second section of the document, it describes every individual component used in the app and their relevant interfaces. Both the functional and non-functional perspectives are thoroughly covered in this document. At the latter half of the document, some non-functional attributes like reliability, maintainability, security, and availability of the software solution are further described.

## 2. Overall Descriptions

The main idea behind the introduction of ICRSS is to reduce traffic congestion, especially in urban areas. As this is targeted on the office crowd, the time wasted on roads will decrease due to the less number of vehicles entering the city limits due to the use of our app [3]. UPM plays a key role in achieving our main goal. This whole document describes the main goals and how they are achieved, the connectivity between other modules, all the interfaces required during the implementation, etc. are comprehensively discussed in this document.

### 2.1 Product Perspective

As per our market research, we identified many existing solutions to the domain that we are addressing. Out of them, many solutions are a failure and does not address the need of people. Therefore, we did a deep study on each of the solutions to identify their features and loopholes, so that we can make our ICRSS a better one for the user. This ICRSS is a combination of features of the existing solutions as well as we have added many new features for the beneficiary of the user and to stand out from the existing solutions. Out of the several research components, UPM has been a revolution with some newly added features which does not see in any other application in Sri Lanka. Below table describes the features which newly added to the solution [2, 4, 5].

Features	UDIO	Carpooling.lk	RideShare.lk	Proposed Solution (ICRSS)
The system mainly focused on office staff	X	X	X	✓
Matching the passengers' profile with suitable drivers	X	X	X	✓
Allow the spouse/guardian to check the passenger's trip details	X	X	X	✓
The suggestion of drivers per passenger interests and preferences	X	X	X	✓
Consider gender preference when registering to provide high security	✓	X	X	✓

*Table 2.1 - Feature Comparison Table*



This application is only focused on office staff to maintain professionalism which is a new concept introduced to the market. Also, in the existing solutions, any driver is suggested to the passenger, and we have taken a step forward in making a new advanced method to suggest drivers who are matching with the interests, job profile, and preferences of the passengers. This reduces the number of dissatisfactions from the point of both the driver and the passenger. Also, we have added a new feature for the spouse or the guardian to check the trip history of the fellow passenger. UDIO too has the feature to enable and disable the gender preference [2], and we have taken it as an added feature to ensure more security on the point of the passenger. As a conclusion, ICRSS has taken the ridesharing to a new step forward in society.

### **2.1.1 System Interfaces**

As the initial step, ICRSS is run on the Android platform, and it is done using the Android Studio. For the backend of the application, the node server is deployed on the cloud storage, and firebase is used for some parts of the application to take the real-time data. We have designed several system interfaces about the UPM to have synchronization between the back end data and front end data.

### **2.1.2 User Interfaces**

Given below are UI used in the UPM.



*Figure 2.1 Home Screen of ICRSS*

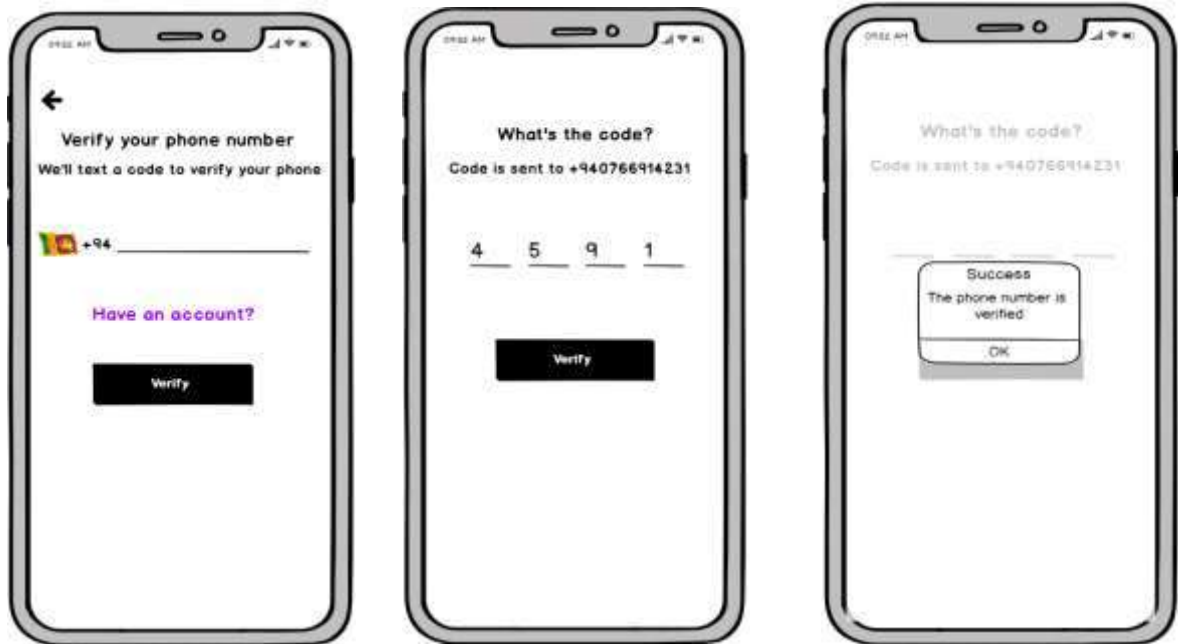


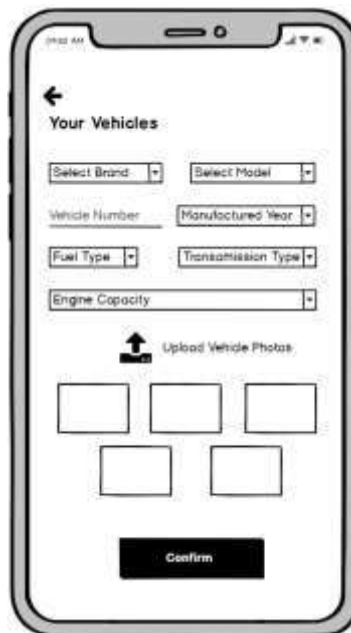
Figure 2.2 Phone Number Verification



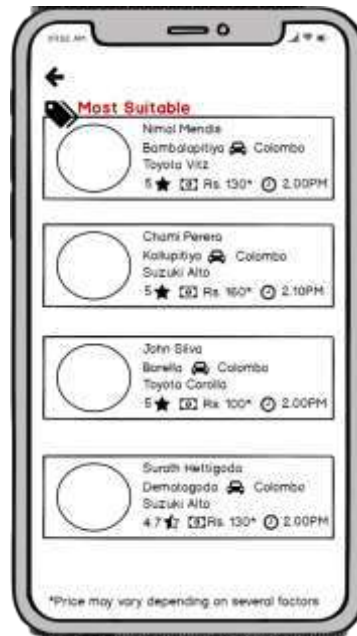
Figure 2.3 User Profile



*Figure 2.4 Adding Bank Details*



*Figure 2.5 Adding Vehicle Details*



*Figure 2.5 Suggested Driver List*

### 2.1.3 Hardware Interfaces

As this is a mobile platform based application, less amount of hardware is needed. For the application to run, internet connectivity is a must. For the UPM, the camera component and GPS module of the phone is used as the hardware components.

### 2.1.4 Software Interfaces

The main software interfaces used in UPM component are,

- For the application development: Android Studio
- Backend Database Server: MySQL
- Real-time data synchronization: Firebase
- For the Inbuilt data storage: SQLite
- For the implementation of algorithms: Python
- For the implementation of web API: Express.js

### **2.1.5 Communication Interfaces**

Internet Connectivity of the phone is the medium used for the synchronization between the databases and preferably 3G connection or higher is essential for the smooth data transaction.

### **2.1.6 Memory Constraints**

The Android mobile application is required,

- Android version should be 6.0 or higher
- 2 GB RAM(Minimum) and 4GB RAM is Recommended
- 100 MB Memory space

### **2.1.7 Operations**

Most of the operations are done by the Passenger in the UPM and registration process is shared between both the passenger and the driver. Below are user operations are done by Passenger, Driver and additionally Spouse/Guardian.

- User should sign up to our application
- User can update/delete an existing account
- User should verify the phone number
- User should provide personal details
- User should provide personal interests and preferences
- User should add bank details
- User should add vehicle details
- Passenger should search for a driver
- Passenger should select a suitable driver from the suggested list
- Spouse or Guardian can report suspected drivers

### **2.1.8 Site Adaptation Requirements**

English is used as the common medium to represent the application interfaces. For the application to be used, the mobile phone should have internet connectivity and access to the camera. Unless the mobile phone number is verified, no user is allowed to create an account at the first instance.

This application only works in the OS above API 23. Older OS will not be compatible with the advanced features of our application.

## 2.2 Product Functions

### 2.2.1 Use Case Scenario

<b>Use Case ID</b>	UPM001
<b>Use Case Name</b>	Register to the system
<b>Preconditions</b>	User should have a valid mobile phone number
<b>Actor</b>	User
<b>Main Success Scenario</b>	<ol style="list-style-type: none"><li>1. Verify the phone number</li><li>2. Provide Personal Details</li><li>3. Provide Personal Interests/Preferences</li><li>4. Provide Bank details</li><li>5. Provide Vehicle Details</li></ol>
<b>Extensions</b>	<ol style="list-style-type: none"><li>1.a. If the phone is already registered to the system, prompt an error message to the user.</li><li>4.a. If invalid bank details are provided, prompt an error message to the user.</li><li>5.a. If vehicle images are not uploaded, prompt an error message to the user.</li></ol>

*Table 2.2.1.1 Register to the system Scenario*

<b>Use Case ID</b>	UPM002
<b>Use Case Name</b>	Log in to the system
<b>Preconditions</b>	The user should be registered to the application.
<b>Actor</b>	User
<b>Main Success Scenario</b>	<ol style="list-style-type: none"> <li>1. Provide the username</li> <li>2. Provide the password</li> <li>3. Click to login</li> </ol>
<b>Extensions</b>	<ol style="list-style-type: none"> <li>1.a. If the Username is invalid, the system will prompt an error message.</li> <li>2.a.1. If the Password is invalid, the system will prompt an error message.</li> <li>2.a.2. If the Password is forgotten, the reset link is sent to the email.</li> </ol>

*Table 2.2.1.2 Log in to the system Scenario*

<b>Use Case ID</b>	UPM003
<b>Use Case Name</b>	Search for drivers
<b>Preconditions</b>	The user should be registered to the application and have a valid account.
<b>Actor</b>	Passenger

<b>Main Success Scenario</b>	<ol style="list-style-type: none"> <li>1. Include (Login to the system)</li> <li>2. Search for a destination</li> <li>3. Check the driver profile from the suggested list</li> <li>4. Select the most suitable driver</li> </ol>
<b>Extensions</b>	4.a. If the suggested driver does not suit the passenger, the passenger can select the driver that he/she wants.

*Table 2.2.1.3 Search for drivers Scenario*

<b>Use Case ID</b>	UPM004
<b>Use Case Name</b>	Report drivers
<b>Preconditions</b>	The spouse/guardian should have a valid account associated with the passenger.
<b>Actor</b>	Spouse/Guardian
<b>Main Success Scenario</b>	<ol style="list-style-type: none"> <li>1. Include (Login to the system)</li> <li>2. Check the trips traveled by the associated passenger.</li> <li>3. Report any drivers who traveled more frequently with an associated passenger.</li> </ol>

*Table 2.2.1.4 Report drivers Scenario*



<b>Use Case ID</b>	UPM005
<b>Use Case Name</b>	Remove reported drivers from the search list
<b>Actor</b>	System
<b>Main Success Scenario</b>	<ol style="list-style-type: none"> <li>1. Include(Report drivers)</li> <li>2. Checks the database with the passenger's unique ID</li> <li>3. Retrieve any reported driver IDs'</li> <li>4. Remove those IDs' from the suggested list</li> </ol>
<b>Extensions</b>	1.a. If database connectivity is not established, reconnect with the server.

*Table 2.2.1.5 Remove Report drivers Scenario*

<b>Use Case ID</b>	UPM006
<b>Use Case Name</b>	Get the current location of the drivers
<b>Actor</b>	System
<b>Main Success Scenario</b>	<ol style="list-style-type: none"> <li>1. Send a request to the database in every 5 seconds</li> <li>2. Get the current location of the drivers</li> <li>3. Update the current location of the MySQL database</li> </ol>

<b>Extensions</b>	1.a. If polling doesn't work, prompt an error message to the user to reconnect internet in the mobile phone.
-------------------	--

*Table 2.2.1.6 Get the current location of the drivers Scenario*

<b>Use Case ID</b>	UPM007
<b>Use Case Name</b>	Update account
<b>Actor</b>	User
<b>Main Success Scenario</b>	<ol style="list-style-type: none"> <li>1. Include (Login to the system)</li> <li>2. Go to profile</li> <li>3. Update the details</li> <li>4. Click update button</li> </ol>
<b>Extensions</b>	<p>1.a. If login fails, re-prompt to enter the credentials</p> <p>3.a. If all the details are not entered, prompt an error message to the user.</p>

*Table 2.2.1.7 Update account Scenario*

<b>Use Case ID</b>	UPM008
<b>Use Case Name</b>	Delete account
<b>Actor</b>	User



categorized under the user category. So nobody from outside who are not registered to the ICRSS can't use this application.

## **2.4 Constraints**

As this is the initial phase of the project, we have only targeted android, mobile phone users. Also, our application is compatible with the phones which has API 23 or higher OS. For the smooth execution of the application, stable internet connectivity is a must, and without the connectivity, some functions may not produce accurate results. Availability of a camera with higher resolution (at least 8MP) should be present to capture the images.

## **2.5 Assumptions and Dependencies**

- Smartphone with stable internet connectivity and considerable battery power is essential
- Registering user should be having a vehicle to register to the system
- GPS module is enabled in the mobile phone
- Obtaining the current location of the drivers is heavily dependent on the firebase synchronization
- Frequently used data is stored in the in-built database to avoid overhead in connecting to the servers
- User will not add vague information to create fake accounts
- This application only covers Colombo District

## **2.6 Apportioning of Requirements**

In section 1 and section 2 of the SRS, we mainly focused on the primary functions associated with the UPM. In the first section of the document, we briefly introduce with the system functions. In the next section, a detailed explanation of the requirements is provided. In section 3, specific requirements are presented with the OOP methodology. Due to the dependency with OOP, the implementation may differ from normal implementation.

From the first two sections, the methodology, interfaces, requirements, etc. are discussed. In the 3rd section, they are elaborated in detail. All these requirements are expected to be implemented in the initial phase of the implementation. Additional requirements solely depend on the time constraint and will not be committed by developers unless there is a special requirement.

### 3. Specific requirements

#### 3.1 External interface requirements

This section provides a detailed description of all the interfaces discussed in the earlier sections.

Finer details of each software, hardware and communication interfaces elaborated.

##### 3.1.1 User Interfaces

Name of Item	Description of Purpose	Source of input or destination of output	Valid range accuracy and tolerance	Timing	Relationship to other input and output	Data formats
Phone number Field	Phone number is required	Phone keyboard	Any ten digits valid number	N/A	No	Number
Verify Button	Click to send the verification code	Phone keyboard	N/A	N/A	Redirect to Verify Code	N/A

*Table 3.1.1 Send Verification Code Interface*

Name of Item	Description of Purpose	Source of input or destination of output	Valid range accuracy and tolerance	Timing	Relationship to other input and output	Data formats
Email Field	Email is required	Phone keyboard	Any alphanumeric characters with special symbols	N/A	No	String
Password Field	Password is required	Phone keyboard	Any alphanumeric characters with special symbols	N/A	No	String
Login Button	Click to Login to the system	Phone keyboard	N/A	N/A	Redirect to user profile	N/A

*Table 3.1.2 Login Interface*

Name of Item	Description of Purpose	Source of input or destination of output	Valid range accuracy and tolerance	Timing	Relationship to other input and output	Data formats
Code Field	Code is required	Phone keyboard	Four digit code	N/A	No	Number
Verify Button	Click to Verify the phone number	Phone keyboard	N/A	N/A	Redirect to user profile	N/A

*Table 3.1.3 Verify the code Interface*

Name of Item	Description of Purpose	Source of input or destination of output	Valid range accuracy and tolerance	Timing	Relationship to other input and output	Data formats
Full Name Field	Full name is required	Phone keyboard	Any number of characters	N/A	No	String
Profession Field	Profession is required	Phone keyboard	Select from the given list	N/A	No	String
Email Field	Email is required	Phone keyboard	Any alphanumeric characters with special symbols	N/A	No	String
DOB Field	DOB is required	Phone keyboard	Any date selected from the calendar	N/A	No	Date
Gender Field	Gender is required	Phone keyboard	Select either one of radio buttons	N/A	No	Boolean
Guardian/Spouse Name Field	Name is required	Phone keyboard	Any number of characters	N/A	No	String
Guardian/Spouse phone number Field	Phone number is required	Phone keyboard	Any ten digits valid number	N/A	No	Number

Save Button	Click to Login to the system	Phone keyboard	N/A	N/A	Redirect to user preferences	N/A
-------------	------------------------------	----------------	-----	-----	------------------------------	-----

*Table 3.1.4 Add details to profile Interface*

<b>Name of Item</b>	<b>Description of Purpose</b>	<b>Source of input or destination of output</b>	<b>Valid range accuracy and tolerance</b>	<b>Timing</b>	<b>Relationship to other input and output</b>	<b>Data formats</b>
Gender Preference Field	Gender Preference flag is required	Phone keyboard	Select from 3 radio buttons	N/A	No	String
Language Spoken Field	Language Spoken flag is required	Phone keyboard	Select from 3 radio buttons	N/A	No	String
Smoking Field	A smoking flag is required	Phone keyboard	Select from 2 radio buttons	N/A	No	Boolean
Music Lover Field	Music Lover flag is required	Phone keyboard	Select from 2 radio buttons	N/A	No	Boolean
Motion Sickness Field	Motion Sickness flag is required	Phone keyboard	Select from 2 radio buttons	N/A	No	Boolean
Confirm Button	Click to Login to the system	Phone keyboard	N/A	N/A	Redirect to Add payment details	N/A

*Table 3.1.5 Add details to profile Interface*

Name of Item	Description of Purpose	Source of input or destination of output	Valid range accuracy and tolerance	Timing	Relationship to other input and output	Data formats
Credit Card Number Field	Credit Card Number is required	Phone keyboard	16 digit valid number	N/A	No	Number
Expiration Date Field	Expiration Date is required	Phone keyboard	Four-digit date with month and year	N/A	No	Date
CSV Field	CSV is required	Phone keyboard	Three digit number	N/A	No	Number
Login Button	Click to Login to the system	Phone keyboard	N/A	N/A	Redirect to add a vehicle	N/A

*Table 3.1.6 Add payment details Interface*

Name of Item	Description of Purpose	Source of input or destination of output	Valid range accuracy and tolerance	Timing	Relationship to other input and output	Data formats
Select Brand Field	Vehicle Brand is required	Phone keyboard	Select any from the list	N/A	No	String
Select Model Field	Model is required	Phone keyboard	Select any from the list	N/A	Yes	String
Vehicle Number Field	Vehicle Number is required	Phone keyboard	Any alphanumeric characters	N/A	No	Number
Manufactured Year Field	Manufactured Year is required	Phone keyboard	Select any from the list	N/A	No	Number



Fuel Type Field	Fuel Type is required	Phone keyboard	Select any from the list	N/A	No	String
Transmission Type Field	Transmission Type is required	Phone keyboard	Select any from the list	N/A	No	String
Registered Year Field	Registered Year is required	Phone keyboard	Select any from the list	N/A	No	int
Engine Capacity Field	Engine Capacity is required	Phone keyboard	Select any from the list	N/A	No	String
Upload images	Click to upload a maximum of 6 images	Phone keyboard	Select any gallery	N/A	No	Blob
Login Button	Click to Login to the system	Phone keyboard	N/A	N/A	Redirect to add a vehicle	N/A

*Table 3.1.7 Add vehicle details Interface*

### 3.1.2 Hardware Interfaces

#### 1. Smart Phone

- a. CPU: 1.6GHz or higher
- b. Storage: 200MB or higher
- c. RAM: 2GB or higher
- d. OS: Android API level 23 - Lollipop
- e. Front Camera: 8 MP
- f. Back Camera: 13 MP
- g. GPS Module

#### 2. Server

- a. CPU: 2.6GHz Intel Core i7 (6th Gen or Higher)

- b. RAM: 16GB DDR3 or higher
- c. Storage: 1TB HDD
- d. OS: Linux (Preferably Ubuntu 16.04)

### **3.1.3 Software Interfaces**

- a. Android Studio 5.6
- b. Visual Studio Code
- c. Android Emulator
- d. Firebase Interface
- e. SQLite Browser
- f. Java SE 8

### **3.1.4. Communication Interfaces**

In UPM, the communication process happens through internet connectivity. To get the current location of the driver, firebase communication is established via internet connectivity. It is preferred to have at least a 3G connection for the best data synchronization.

## 3.2. Classes/Objects

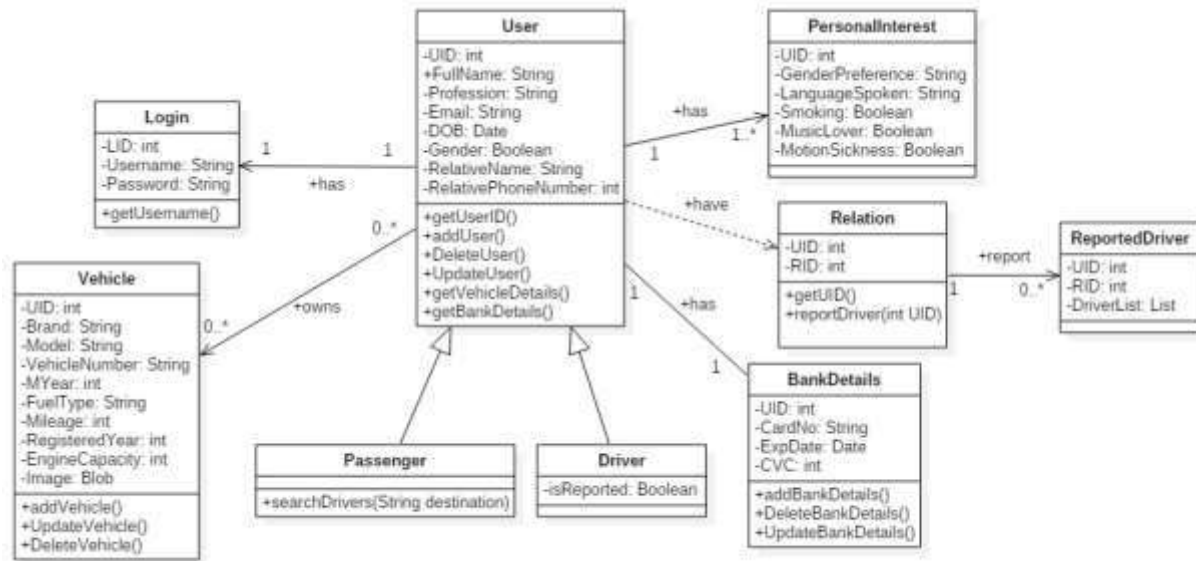


Figure 3.2 Class diagram of UPM

## 3.3 Performance requirements

As this is planned to be implemented only for the Android platform, it takes a considerable amount of RAM capacity of the mobile phone for the execution. So it is recommended to have at least 2GB RAM. As this application requires to gather some real-time data of the user, it is essential to have stable internet connectivity to maintain the quick response and stability of the application. Losing any one of the factors can cause the application to malfunction. Also, databases are stored in servers; anyone number of users can use the application at the same time without getting interrupted.

## 3.4 Design constraints

There are several design constraints that need to be considered in designing the “+Go” mobile application.

- Sessions need to be maintained for every individual user
- The loading time of the mobile application should be less than a few seconds
- Application and database should be available 24x7

- User Interfaces should be easy to use and understand

## **3.5 Software system attributes**

### **3.5.1 Reliability**

Reliability of the +Go application, high depending on the performance of the UPM component. After the destination is searched, it is the duty of the UPM component to get the drivers who are traveling to that particular location and display it to the passenger. Reliability heavily depends on the accuracy and availability of data and real-time data synchronization.

### **3.5.2 Availability**

The application should be available 24x7 and server should be up and running always. Users should be able to access the app from anywhere they go using their internet connection. Before the post-release, each component of the UPM is thoroughly tested individually as well system wise to make sure those executes without any bugs. System downtime is also taken into consideration while implementing.

### **3.5.3 Security**

As we are collecting some sensitive data of the user, we must secure them without exposing to external parties. During the registration process, the password is taken from the user and saved it in the database after hashing using the salt method. All the databases are created with admin privileges, and no other external party can access them. Confidentiality of the users are fully confirmed, and no fake accounts can be created in our system due to a fully functional verification system.

### **3.5.4 Maintainability**

Each algorithm is modularized into packages to reduce the coupling between them. Thereby cohesion is increased, and dependability of other modules is decreased as a step in easy maintainability. Even when there is any server failure, due to maintaining an in-built database, user's ability to use the app like the same way without getting interrupted until the server is

instantiated back again. Application is coded in a way that is using classes and object, which can be further modified or utilized in any other components.

### **3.6. Other Requirements**

This section elaborates the other optional requirements that can be seen in our application. As these requirements are optional, these may be implemented in the second phase of the implementation.

- Locking of the account, if three failed attempts taken to the login
- Feasibility of the system in desktop versions
- Open source code bases are used
- Reliable and powerful servers are used

## 4. Supporting Information

### 4.1 Appendices

Throughout the SRS, we have elaborated the solution that we presented to overcome the traffic congestion in morning hours. For that, UPM plays a major role in the implementation.

### 4.2. References

[1] Urban Urban Transport Master Plan for Colombo Metropolitan Region and Suburbs. (2014). [ebook] Colombo ,Sri Lanka: Ministry of Transport, pp.25,26. Available at: [http://www.transport.gov.lk/web/images/stories/CoMTrans%20Final%20Report\\_S.pdf](http://www.transport.gov.lk/web/images/stories/CoMTrans%20Final%20Report_S.pdf) [Accessed 13 Jan. 2019].

[2][online] Available at: <http://www.ft.lk/motor/Innovative-car-pooling-app-UDIO-goes-live/55-645554> [Accessed 18 Jan. 2019].

[3] [online] Available at: <https://www.newsfirst.lk/2017/03/16/rs-500m-loss-incurred-daily-traffic-congestion-transport-authorities-observe/> [Accessed 30 Dec. 2018].

[4]"Carpooling", Carpooling.lk, 2019. [Online]. Available: <http://carpooling.lk/>. [Accessed: 03- Feb- 2019].

[5]"Carpool for better commute and quality of life", RideShare.lk, 2019. [Online]. Available: <http://rideshare.lk/>. [Accessed: 22- Feb- 2019].