Final Year Project:

Clever Carpooling

User Manual

Authors:

Shaun Carey *16450454*

Nigel Guven *14493422*

Supervisor:

Ray Walshe

clever carpool

Table of Contents

1. Introduction

1.1 Overview

Clever Carpooling is a ridesharing planner which aims to make transport options more widely available for the public. The application primarily focuses on the personal automobile market such as cars and motorbikes but also may be extendible to private vans, buses and more. Users may register as a driver or a passenger and be able to post a route of their journey for a certain date and time. Other users may be able to view that post and then contact the owner of that post, with an option to pick up or request a lift. The users can communicate within the application messaging system where they can send text messages, images, and PDF documents. There are also public forums which any user can set up with ease so that multiple users can organise a carpool.

1.2 Future Work

As of the 12th of May 2020, we have completed all functionality set out by our manifesto in the Functional Specification which can also be viewed in the same repository as this document. Further plans may be extended to this application given the ability to bring this application to market. The Google Play Store is the most likely place to platform this application. Currently, the team is using Firebase Blaze plan for extensive application features but may reconsider this given financial circumstances.

Future features may include the implementation of a payment system for users to donate as much money as they would like to a driver, and this would require a revision of Ethics and a secure system for payments. There is also the greater possibility of adding weather features to the application so that users may know of what the weather will be like for a certain day.

1.3 Glossary

Android – Android is a mobile operating system designed for highly interactive touch screen mobile devices. Clever Carpooling is supported on Android devices.

Android Studio – Android Studio is an IDE specifically designed for developing new and innovative Android applications. Clever Carpooling was built using Android Studio

GUI – The graphical user interface is a form of user interface which allows users to interact with the device functionality.

Operating System – An operation system is a piece of software which manages computer hardware, software and provides underlying services for computer applications.

Navigation Bar - The bright blue toolbar located at the bottom of the GUI, used to navigate through the various sections of the application.

Passenger Icon - The blue and white 'thumbs up' symbol which can be seen in the application's logo as well as in various sections of the application

Driver Icon - The blue car symbol which can be seen in the application's logo as well as in various sections of the application

Firebase Blaze – Blaze is an upgraded Software-as-a-Service version of Firebase which provides access to the latest versions of Firebase tools.

ADB – Android Debug Bridge is a versatile command-line tool which lets you communicate with a device and provides access to a UNIX shell.

Gitlab – Gitlab is a DevOps platform which provides version control, CI/CD and issue tracking.

IDE - An Integrated Development Environment is a software application that provides facilities for development of software. Android Studio is an example of an IDE

Dashboard - The Carpool Dashboard is the central system of the device which holds a list of routes which can be further expanded by a user. The dashboard contains functionality to add new posts and delete them also. Furthermore, there is the option to filter list items using a dashboard search bar.

Firebase Cloud Messaging – Firebase Cloud Messaging is a cloud solution for adding notifications to applications. Notifications are stored on a Node.js server and are triggered when some event occurs within the application database.

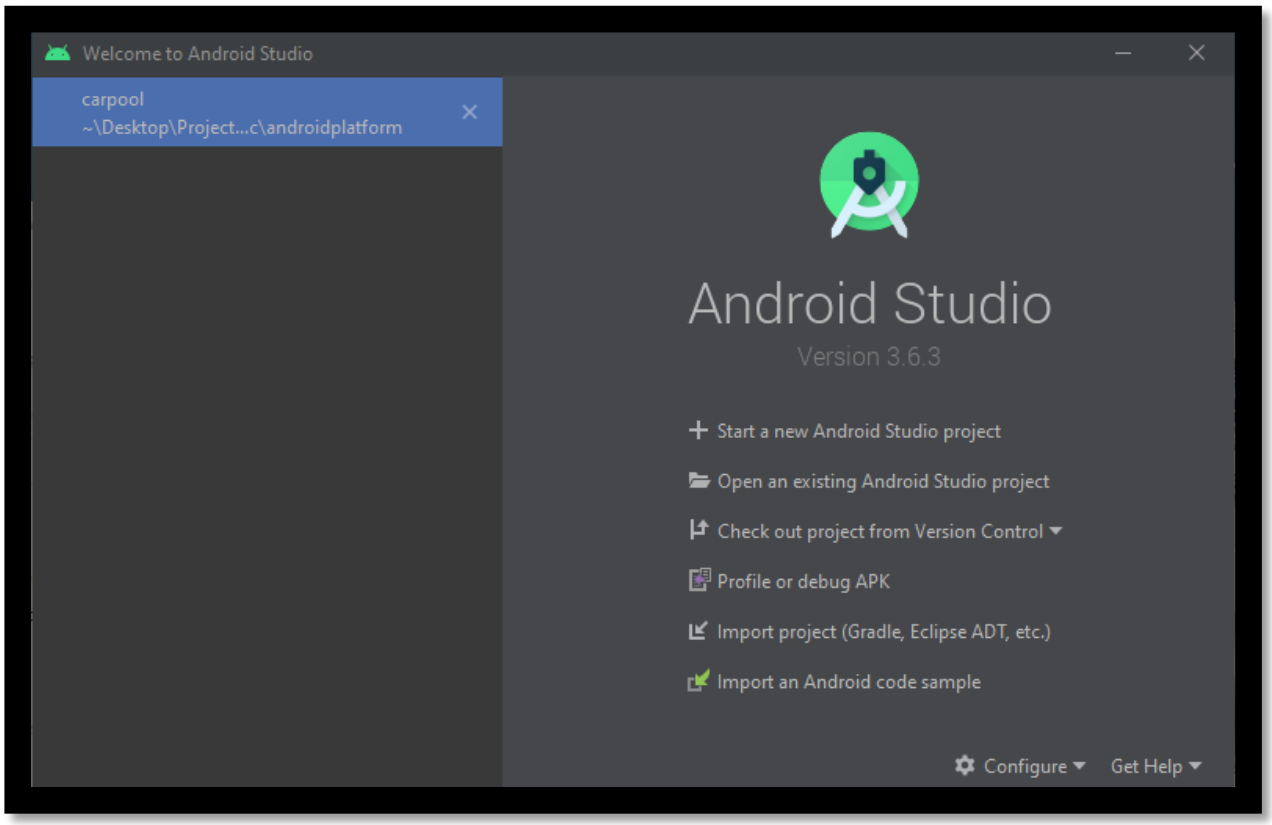
2. Installation and Prerequisites

These installation instructions are described for those who wish to use the application. For those who want to know how to view the source code and add features to the application, please refer to the installation instructions in the Technical Manual contained within the same repository branch.

1. Prerequisites:
    a. Android Studio is installed on the user's computer, operating system being Windows or OSX.
    b. [OPTIONAL] ADB is installed on the user's device.

2. Download the project folder from its repository located on Gitlab at the following link: https://gitlab.computing.dcu.ie/guvenn2/2020-ca400-nguven-scarey

3. From the Android Studio home interface, click "*Open an existing Android Studio project*". Open the application root app folder in Android Studio IDE. This root app folder can be found by looking for the Android Studio icon in the folder.



4. Wait for the program to compile the gradle files and build the system.

5. Once the green tick symbol has been displayed, a run option should become available at the top of the screen.

6. If no virtual device has been downloaded or the ADB has not been configured, please download a virtual phone, or connect a physical device by USB or Wi-Fi. Click the Run button to the right of the Emulator drop-down menu and the application should run.

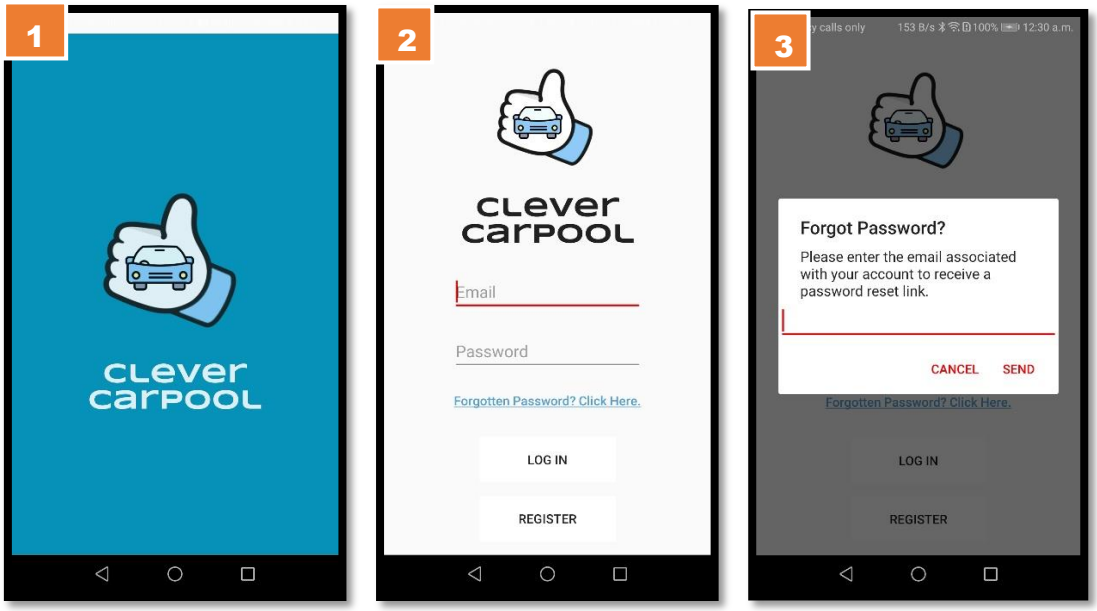
7. The application splash screen should display as below indicating that the phone has been connected to Android Studio with the application installed. The application will run automatically but if this is not the case, please refer to errors that may display in Logcat.

8. You are now ready to use the application outside of Android Studio.

3. User Guide

3.1 Account Creation

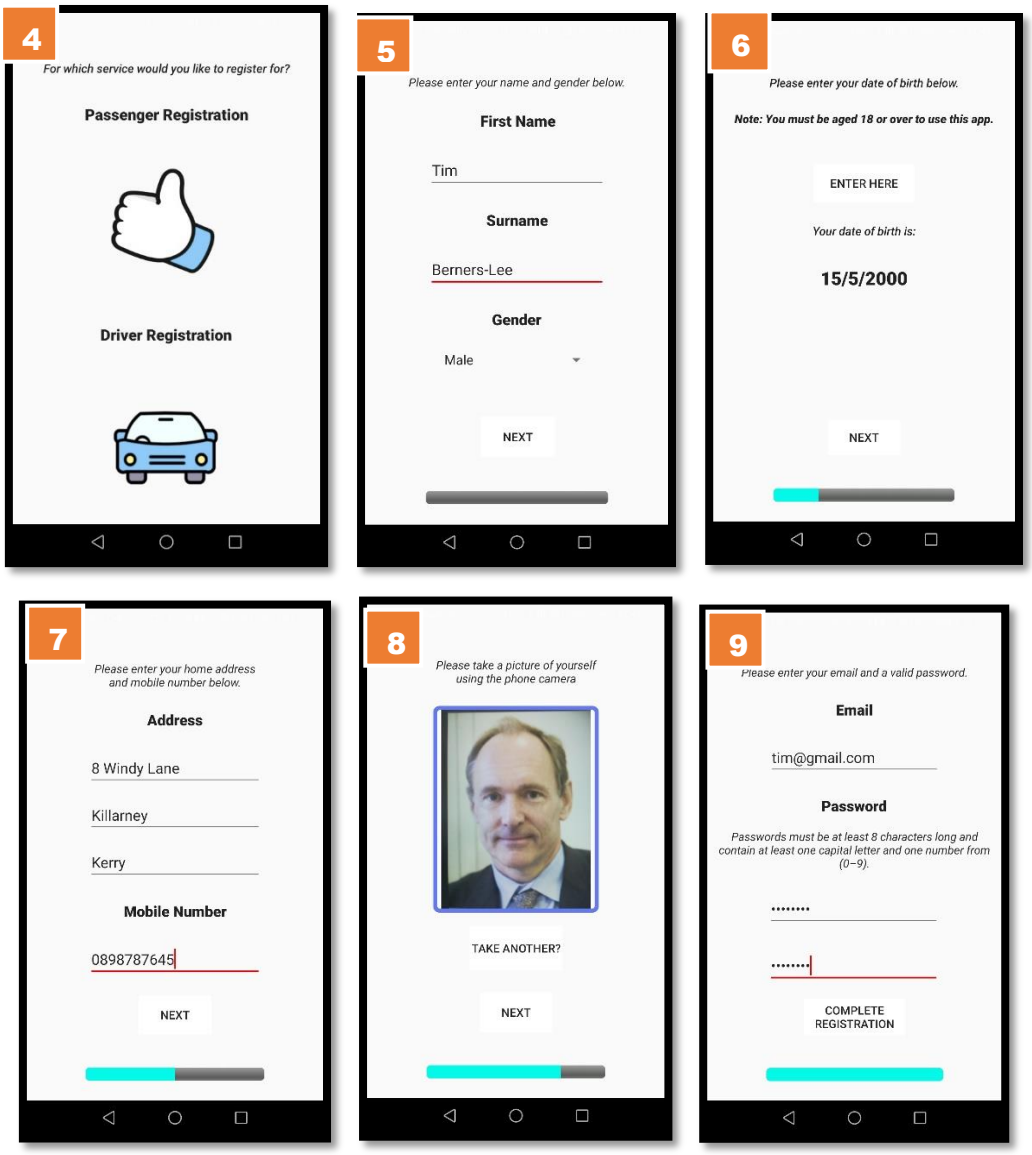
3.1.1 Login Screen



From starting the application, the user views a splash screen and is then presented with the login screen. If a user has already created their account, they may login to the home screen of the application. A forgotten password can be resolved by typing in the user's email, and a password change link is sent to the email of that user. Firebase handles the generation of forgotten password links. First-time users will be required to click the 'Register' button in order to use the application any further.

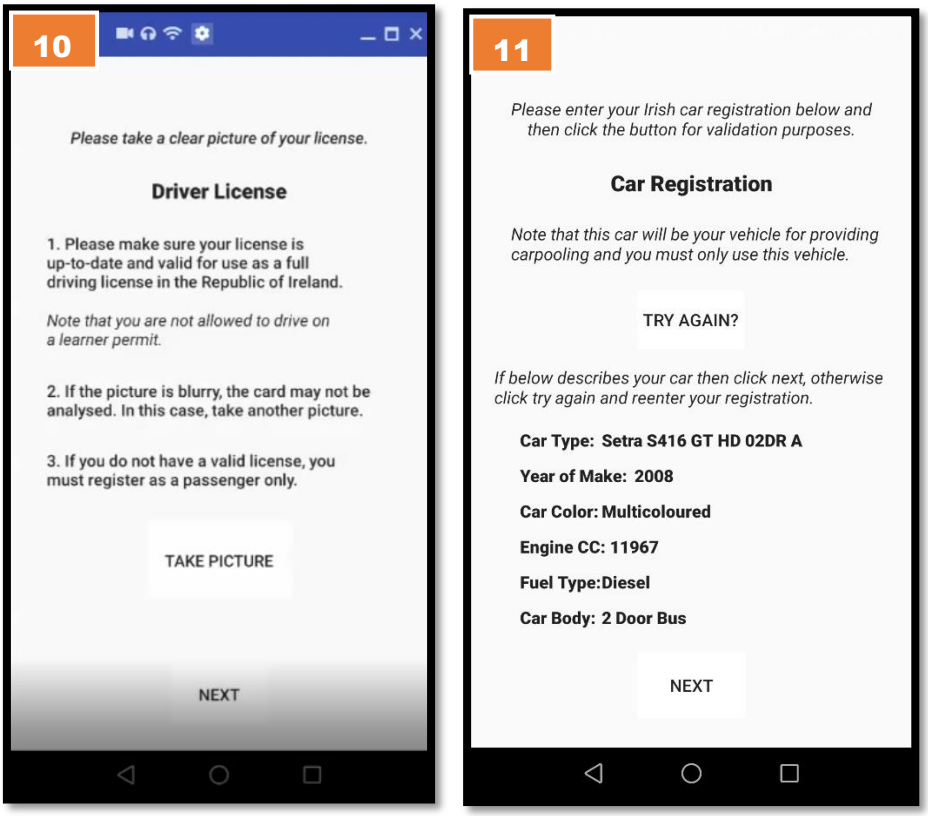
Image 1 shows the splash screen. Image 2 shows the login screen. Image 3 shows the forgot password test box which is where a user enters their email to send a forgot password link to their account.

3.1.2 User Information



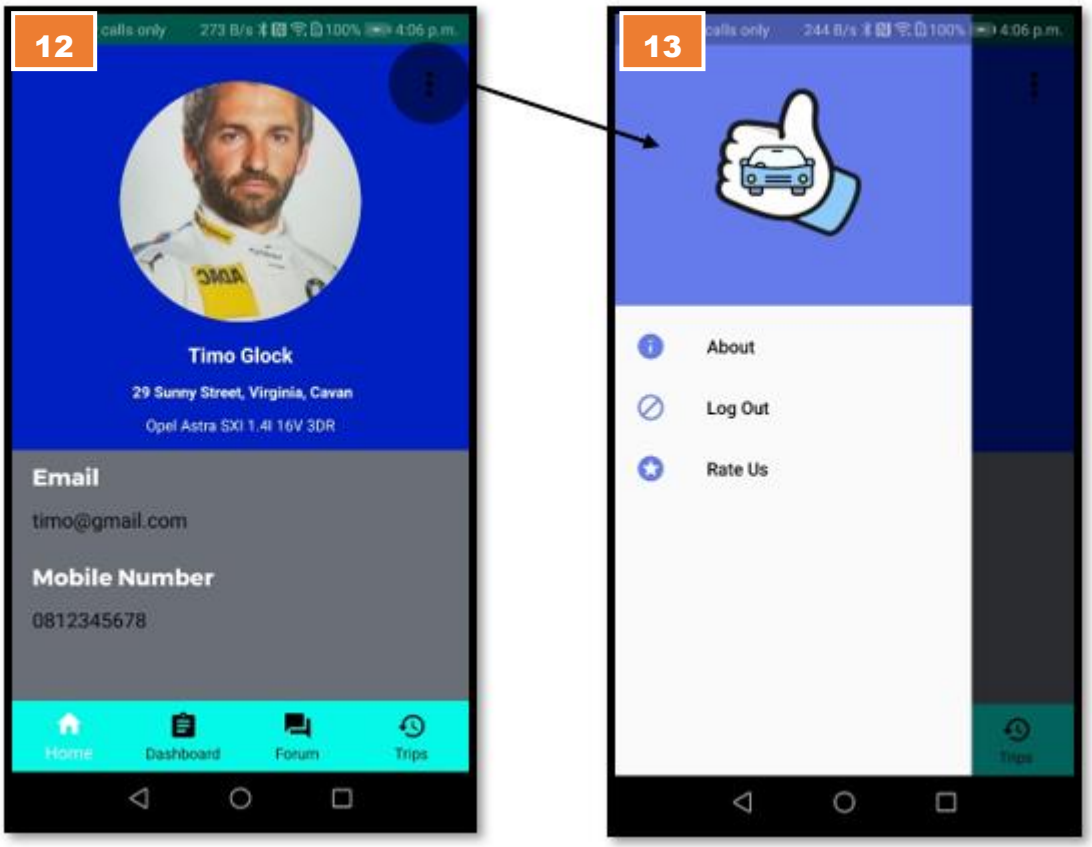
When a user clicks on the register button, they can choose to register as a driver or a passenger.Users will then be asked for usual registration details, such as name, date of birth (must be 18 or older to use the application), and email address – which must be valid. Should a user want to register as a driver, they will be asked for their car registration plate and a photograph of their driver's licence. Image 4 shows the selection page to register as passenger or driver. Image 5 shows the registering of a name and surname. Image 6 shows the selection of user's date of birth who must be over 18. Image 7 shows the input of the address and mobile number of the user. Image 8 shows the taking of the profile picture of the user which is displayed on the screen and can be retaken. Image 9 shows the entry of an email and password which will be the login details of the user

3.1.3 Driver Information



Drivers are required to provide a valid licence. If a learner permit is detected or there is no valid licence detected, then the user may not progress. If the user has their licence validated, then they must provide their car registration which will be determined from the input text, and the driver may confirm those details. Once registered, users are free to use the rest of the application. Image 10 shows the rules and regulation which the user must abide by when taking a picture of their driver's licence. Image 11 shows a car registration which has been retrieved from the user input.

3.2 Home Interface



Upon tapping on 'Home' on the navigation bar, users will be brought to their homepage which simply contains their personal information: a profile picture, name, address, car details (if registered as a driver), email address, and phone number. Should a user wish to log out, view the 'About Us' section, or rate the application, they may tap on the dotted lines in the top right corner. Image 12 shows the home profile which shows the majority of the current user's details, including email, mobile number, home address and car model if the user is a driver. Image 13 shows the side navigation bar which shows a list of three items containing different sections of the application,

3.2.1 Navigation Side Bar

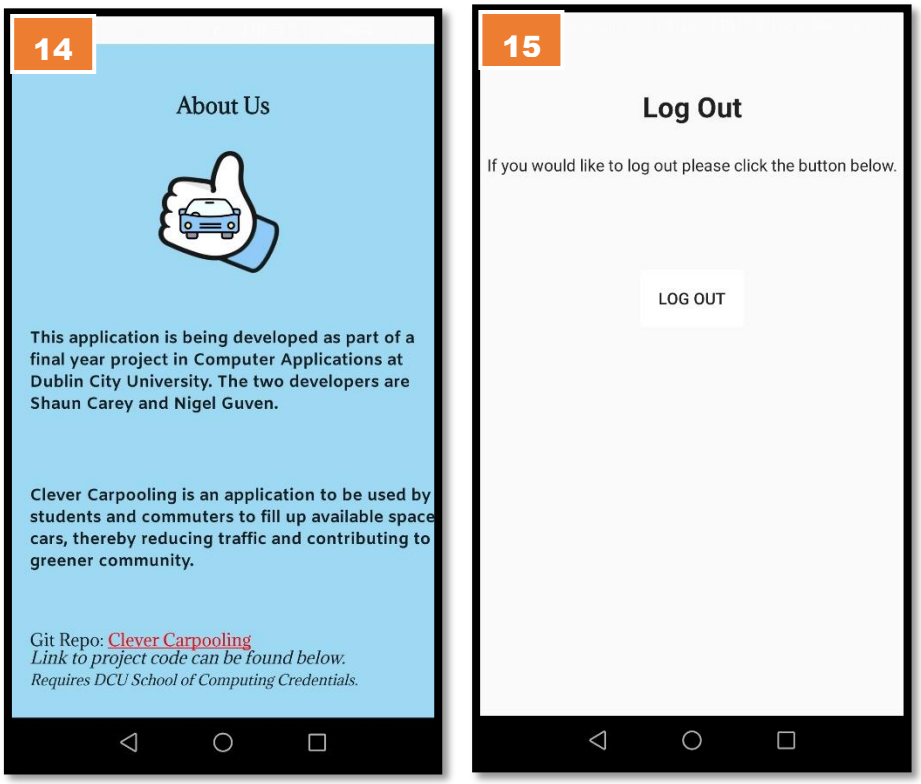
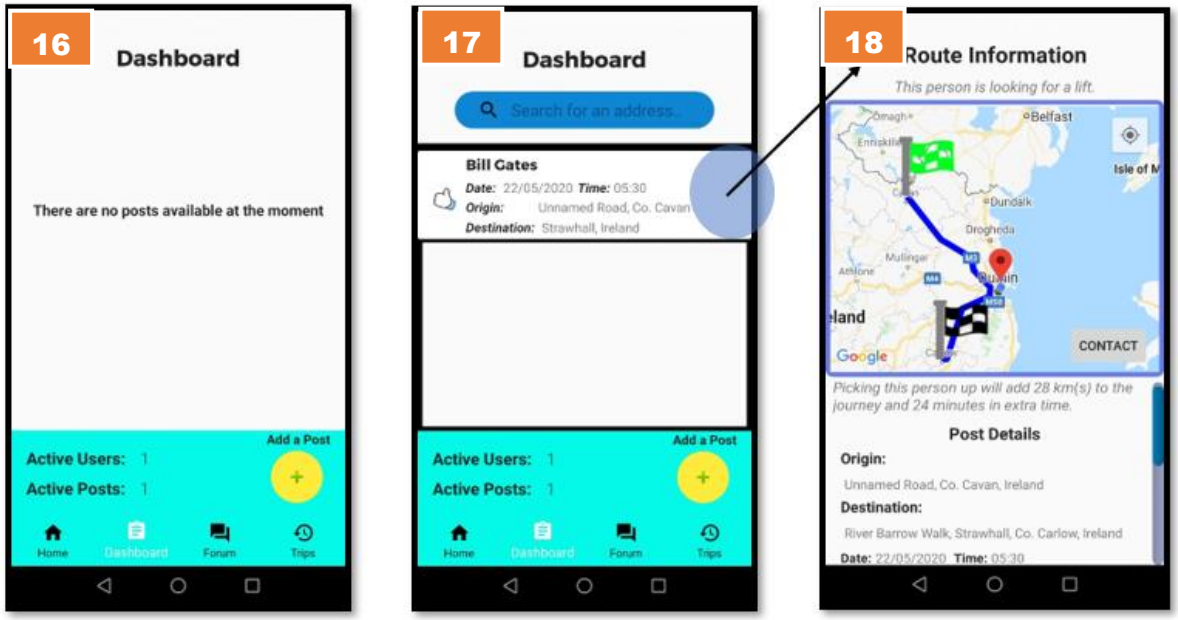


Image 14, the About Activity contains information about the project, the repository location and the development team. Image 15, the Log Out activity shows a single button which the user may click to end their current instance on the application. They will be brought back to the login screen. Note that the 'Rate Us' section is missing as it is still under development and will be completed when the application is added to the Google Play store.

3.3 Dashboard

3.2.1 Viewing Posts

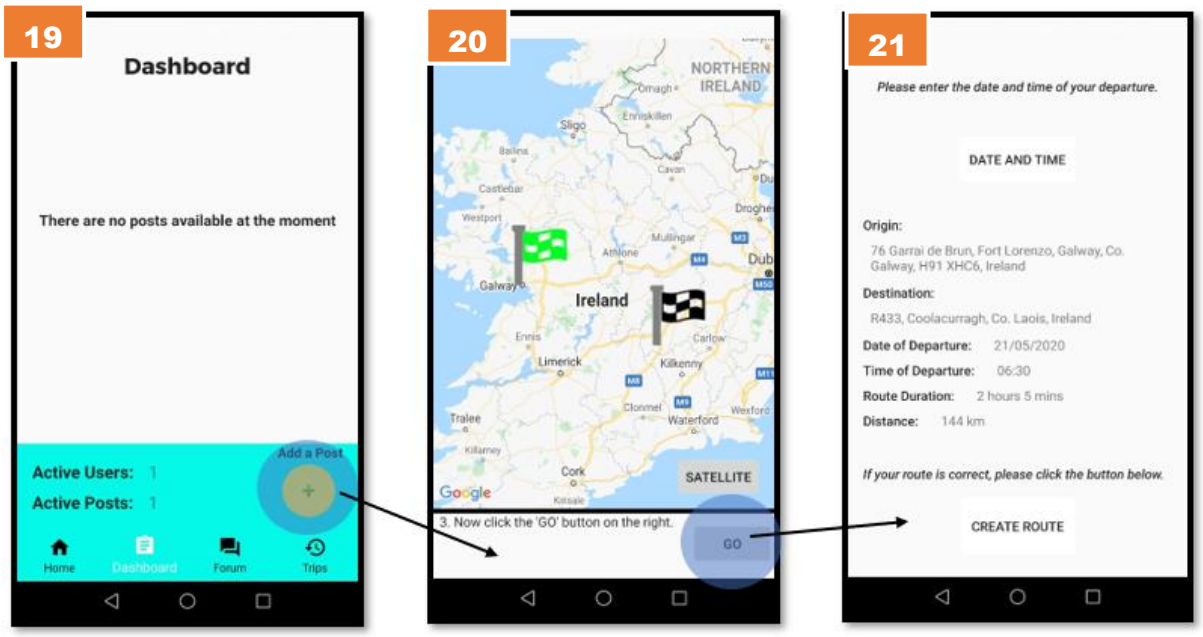


The user is sent to the dashboard once they have logged in to the application. Otherwise, to view the Dashboard, the main activity hub of the application, the user may tap on 'Dashboard' on the navigation bar. The user will then see a list of posts (if there are any) of other users either *requesting* a lift or *offering* a lift. When another user is requesting a lift, the post will be denoted by the passenger icon on the left-hand-side. When they are offering a lift, it will be denoted by the driver icon. The owner of an item on the dashboard can delete their own post.

The primary details of this trip will be displayed (Image 17). Upon tapping on this post, further trip information will be displayed (Image 18). In image 18, the user can see the full details, showing its origin and destination as well as the time stamp and duration of the trip. Depending on whether the user is a driver or passenger, or the owner of the post a helpful notice will indicate the extra distance required to add the current user to the journey.

The user may click the contact button in image 18 to send a request for a pick-up or collection of that user associated with the post. Passengers are only able contact drivers. In Image 17, the user may refine their search for an item using the search bar. If there are relevant results found, the user may sort them by their relevance or date.

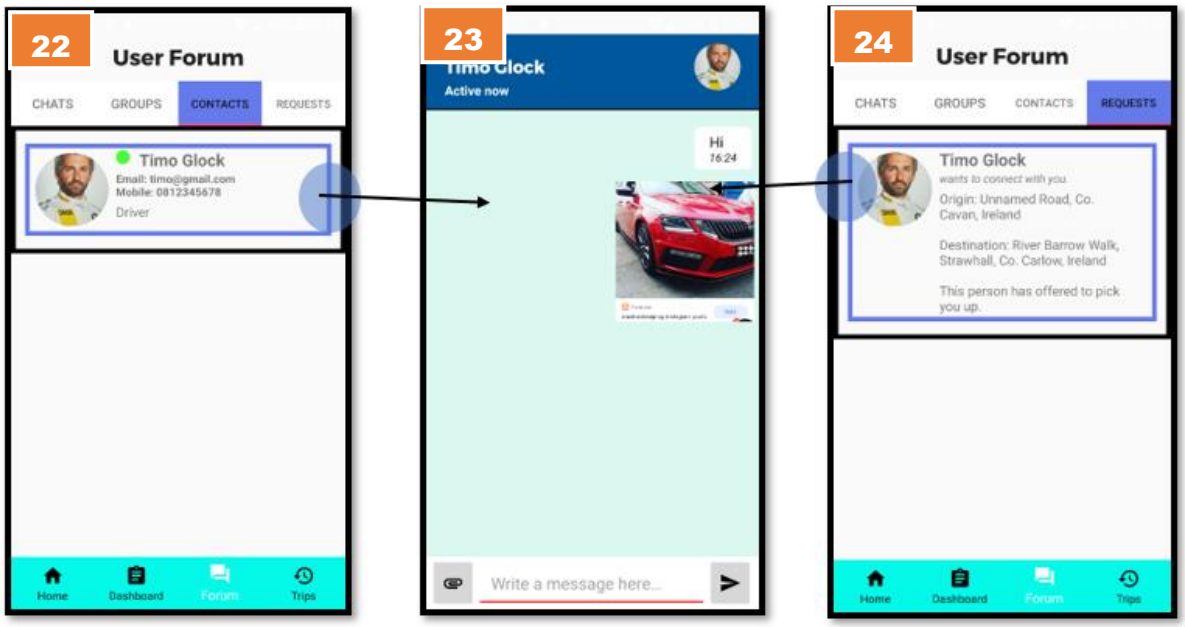
3.2.2 Adding and Removing Posts



A user may create their own request, either as a passenger or a driver, by tapping on the green 'Add a Post' button on the bottom right of the Dashboard (Image 17). The user will then be given instructions on how to set a start and end point for their journey. After doing so, they will be brought to a confirmation screen where they may enter a date and time for the journey, as well as being asked to confirm whether the rest of the details are correct or not. Upon tapping on 'Create Route', the post will be created and will now be visible to fellow users on the Dashboard.

3.4 Messaging Forum

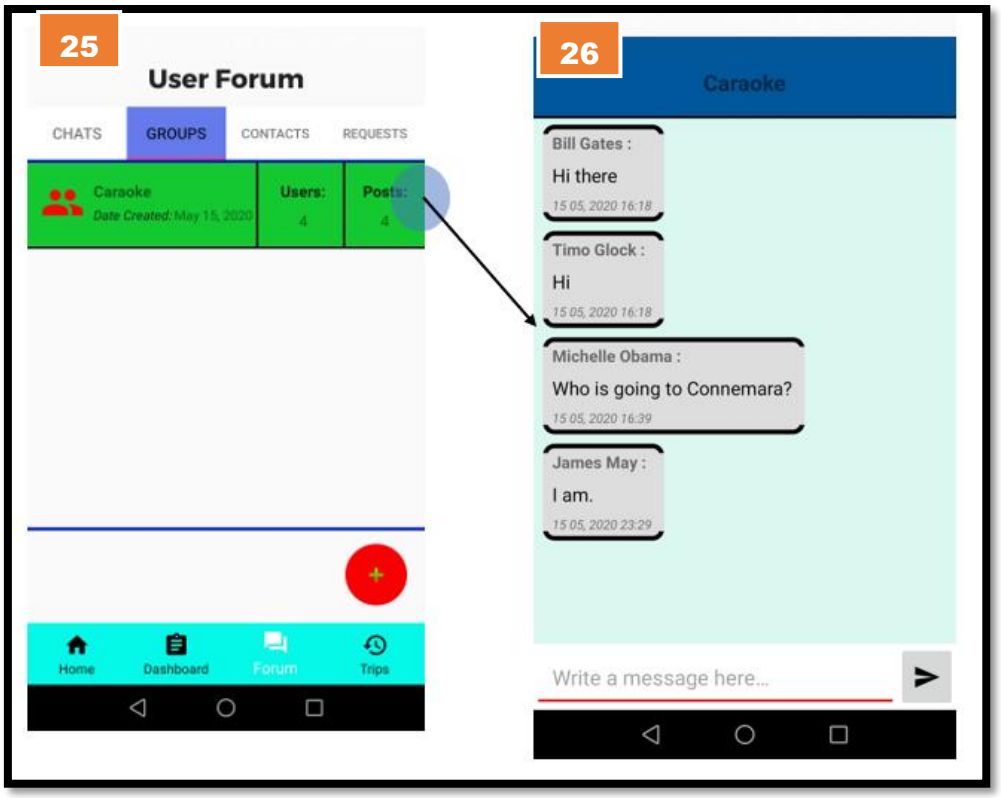
3.4.1 Peer-to-Peer Messaging



Tapping on the 'Forum' icon on the primary navigation bar will bring the user the Messaging Forum – a place where you can chat with saved contacts from a previous trip (Image 22), or view chat requests from an incoming user (Image 24), by utilizing a second navigation bar at the top of your screen. Tapping on either of these will bring you to a messenger system where you may write messages to that user. User are able to send text messages, images, and PDF documents from within this 1-1 chatroom.

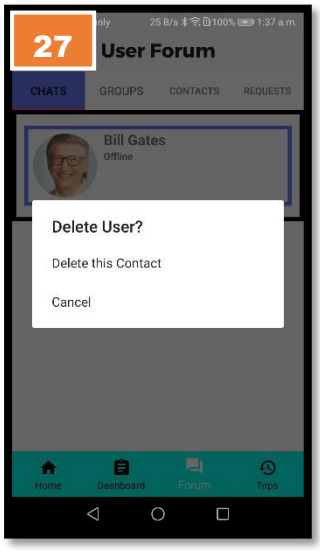
From the request section in image 24, the current user is able to view the request type from the other user. If the current user is a driver and the request is from a passenger, then text will be displayed indicating how long it will take in time and kilometres to pick up that person. From this, the user can make a judgement on whether or not to pick up the person involved in the request. In the chat room users are able to delete messages, PDF documents and images for themselves or for the two users in that group. A user can also see if another person is online by navigating to the Contacts section or viewing over all private chats which may display as Last Seen with a timestamp, Active Now or Offline if the user is full logged out.

3.4.2 Public Group Messaging



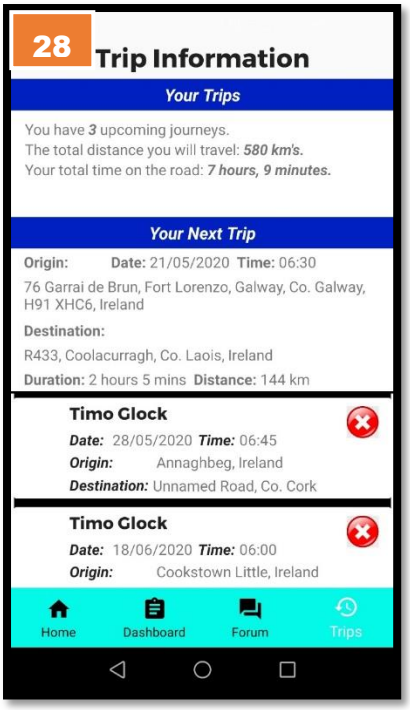
Also included in the User Forum is a Groups section, where users may create and join messaging groups in the case of multiple users sharing the same lift on a multiple basis. This can be accessed by tapping on the 'Groups' button on the upper navigation bar. Each group has displayed a count of the users in that group and also the number of posts in that group. Image 25 shows a group in the list with its count of users and posts. Image 26 shows a group chat room with more than 2 users writing text messages. In the group forum, users can only send text messages.

3.4.3 Deleting Contacts



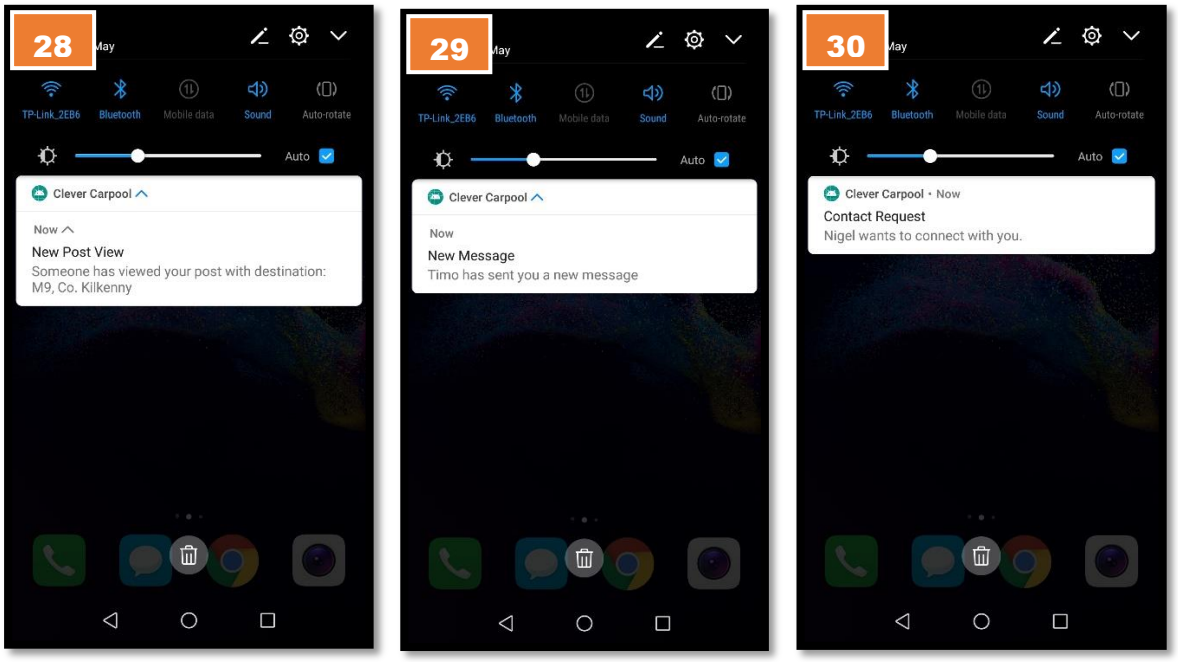
In the interest of user safety, users may be able to delete those users which are using the application for nefarious purposes. The development team will do its best to make sure that users are protected inside the application. Deleting a user is done by simply going to either the chat room or the contacts list and holding over that specific user. A dialog box will show, asking if the user would like to delete that person. Image 27 shows this action.

3.5 User Statistics



The final section accessible by the navigation bar is the Trip Information section. Here, users can see all information on any upcoming trips they may have and have quick access to cancel any of them if they so wish, by tapping on the X button on the top right of each trip. This is beneficial to a user since it is easier to delete routes inside this management section rather than having to manually browse the dashboard for any posts with the current user's ownership. Image 29 shows the total distance and time covered across a user's routes which makes it easier to plan trips ahead. The next route is displayed along with any upcoming trips if available or not.

3.6 Notifications



These notifications are sent by the Firebase Cloud Messaging service and activate when a user triggers an event inside the application. Image 28 is a notification sent to the owner of a post whereby another user viewed their post, which holds a destination. Image 29 is a message notification with the name of the sender attached. Image 30 shows a connection request which is triggered when a user sends a request for a carpool to the current user.

-End of Document-