CAIRDE

**Car Sharing App**

Functional Specification



18th April 2018

BSc (Hons) Software Development

**Name:** Dylan Lawlor

**Year:** 4th year

**Student ID:** C00197013

**Supervisor:** Lei Shi

# Abstract

The purpose of this document is to give an overview of the expected functionality, non-functional requirements and of the design of this application as well as give an iterative plan to detail the implementation of these functionalities.

Table of Contents

[Abstract 1](#_Toc511737523)

[Vision 3](#_Toc511737524)

[Concept 3](#_Toc511737525)

[Business Case Summary 3](#_Toc511737526)

[Measure of Success 3](#_Toc511737527)

[Target Market 4](#_Toc511737528)

[Use Cases 5](#_Toc511737529)

[Use Case Diagram 5](#_Toc511737530)

[Brief Use Cases 5](#_Toc511737531)

[Login Use Case 5](#_Toc511737532)

[Register Use Case 5](#_Toc511737533)

[Set up Profile Use Case 6](#_Toc511737534)

[Create a Journey Use Case 6](#_Toc511737535)

[Search for Journey Use Case 6](#_Toc511737536)

[View Journeys Use Case 6](#_Toc511737537)

[Accept Applicant Use Case 6](#_Toc511737538)

[Project Work Plan 7](#_Toc511737539)

[Iterations 7](#_Toc511737540)

[Iteration One 7](#_Toc511737541)

[Iteration Two 7](#_Toc511737542)

[Iteration Three 8](#_Toc511737543)

[Non-Functional Requirements 9](#_Toc511737544)

[Design 9](#_Toc511737545)

[User Experience (UX) 9](#_Toc511737546)

[User Interface (UI) 9](#_Toc511737547)

[Bibliography 10](#_Toc511737548)

# Vision

## Concept

CAIRDE is an app that facilitates the process of car sharing. The concept of car sharing is that people from a similar location, travelling to a mutual destination, share the journey in one car as opposed to two or more, sharing costs and reducing pollution as well as congestion. In this regard the app will allow these people to locate each other and request or offer a lift on the journey. In this way a user may be a Passenger or a Driver. The user will be able to use one account to access both sides of the app, depending on their needs.

The driver route will allow the user to create journeys, advertising a number of spaces in the vehicle for each particular journey, as well as the date and departure time for the trip. They can also view passenger applications for lifts and accept or reject applications.

The passenger route will allow the user to search for lifts, or create a request for a lift. They can also view details about their upcoming journeys. They can apply for seats on a journey, and they can review completed journeys.

## Business Case Summary

There is a need for such an application in Ireland today in order to offer customers a modern convenient secure mobile platform for setting up a car-sharing relation. While car-sharing is often promoted in a positive way as a viable means to undertake repetitive journeys, such as to work, or college. Traditionally this kind of arrangement was made face to face through word of mouth and pre-existing friendships. But with the advent of social media and a move towards online communication, there has not been much significant headway into bringing car-sharing into the modern world. While currently in the Irish market there are a handful of websites offering a similar service, they are not hugely popular or well know, and have little to no presence on the prevalent tech of today, the ever-present smartphone. In the past few years there has been a small upsurge in apps that offer lifts to users, but in a more taxi-like sense like Uber and Halo. This is not car-sharing, its car hiring, and so this application will aim primarily to promote and facilitate car-sharing. The issue of payment will be left up to the individuals involved, with an emphasis on fuel cost-sharing This way the application will maintain its car-sharing ethos, but hopefully create a full population of drivers and passengers.

## Measure of Success

I will consider this project a success when the below conditions are met:

* The Android app is completed and fully usable for car-sharing functionality
* The app backend is up and running
* Users can find and create journeys, and receive notifications of updates
* Users can view historical records

## Target Market

“2,962,550 - The number of persons commuting to work, school and college in April 2016”

Above is a statistic from the Central Statistic Office of Ireland (CSO) stating the number of commuters travelling on the road daily in Ireland in April of 2016. That number has only risen. These commuters are the target market, with the app designed to bring commuters on the same route together in order to reduce the number of cars driven by just one or two commuters.

It is not just drivers commuting to work that are targeted, but students too, particularily third-level students who may commute long distances daily to college.



(CSO, 2016)

This offers a large target market base and so this apps main target market is aimed at regular smart phone users who commute to work or college, either by car or public transport. This is likely the group that will benefit most from this app.

# Use Cases

## Use Case Diagram



## Brief Use Cases

This section will look at the brief use cases derived from the above use case diagram. First a brief description of terms:

* User – A user of the Mobile app CAIRDE
* Azure – The cloud backend, Azure App Services and SQL Database
* App – The CAIRDE Mobile App

### Login Use Case

**Actors**: User, Azure

**Brief description**: This use case begins when the user starts the app. The user is presented with a form prompting for two inputs, username and password. The user enters their details and clicks a button to login. The app communicates with Azure to determine the details provided are correct, and when successful the user is logged in.

### Register Use Case

**Actors:** User, Azure

**Brief description:** This use case begins when the user wants to register with the app. The user is presented with the register screen and is asked to choose a username and password, and confirm the password. The user submits the details and the app sends the data to Azure which stores the information. The user is now registered.

### Set up Profile Use Case

**Actors:** User, Azure

**Brief description:**  This use case begins the first time the user logs into the App. The user is presented with a screen requiring several details about the user to be filled in. The user fills in the details and clicks Set up Profile. The app sends the data to Azure where it is stored. The user is show the home screen.

### Create a Journey Use Case

**Actors:** User, Azure

**Brief description:** This use case begins when the user selects the Create Journey option from the home menu. The user is shown a screen where they can set a from and to location, as well as other details of the journey, and then submit the journey, which will be stored on Azure.

### Search for Journey Use Case

**Actors:** User, Azure

**Brief description:** This use case begins when the user clicks the search journey button on the home screen. The user is taken to a screen where they can enter in details about the journey they are looking for: location to and from, date and time and range within which to search. The user submits the search and the app requests matching journeys from Azure. The user is presented with a list of matching journeys.

### View Journeys Use Case

**Actors:** User, Azure

**Brief description:** This use case begins when the user selects one of the 4 options to view journeys (Driver upcoming, Driver old, Passenger Upcoming, Passenger Old). The App sends the request along with the users ID to Azure, and Azure returns a list of journeys that match the query.

### Accept Applicant Use Case

**Actors:** Azure, User

**Brief description:** This use case begins when a user receives a notification from the app that another user has applied for a lift on one of this users journeys. The user logs into his app and checks the journey. The user clicks Check Applicants and a list of applicants is displayed. The user clicks on an applicants name and selects accept applicant from the popup. The user is removed from applicants and added as a passenger to the journey, and a notification is sent to the passenger. This change is stored on Azure.

# Project Work Plan

## Iterations

The following sections describe planned workload for each iteration. Each Iteration will have 1 or more general tasks associated with it, which themselves may be broken down into more specific tasks at the time of execution.

### Iteration One

In iteration one, it is planned to achieve the following functionalities. This Iteration will mainly be about set-up and research.

#### Development Setup

This involves setting up the IDE and plugins so that development can commence. It also involves learning how to use the tools available and gaining a basic familiarity with the environment.

#### Backend Setup

This involves setting up the Azure account to use the App service and SQL database, as well as integrate it into the project environment. Also in this step is the setting up of any API keys/ Accounts required, ex Google Maps.

#### App Development

This step involves creating the main project, and testing its connectivity to the backend. UI screens should be created at this point

#### Iteration One Results

The objectives that are mentioned above were achieved. However not much of the actual app was developed at this point, just a simple register login system with secure password encryption, so the author had some ground to make up in Iteration two.

### Iteration Two

In iteration the focus was on getting the core function of the app down. This meant integrating the Google Maps layout, and creating and storing Journey details. Searching and applying for a lift is also central to this iteration.

#### Maps Integration

This involves integrating with the Google API to create a map view in the app, which will allow the user to choose locations for departure and destination. These locations will need to be visually shown on the map, and a method or retrieval of these details and markers is required.

#### Create Journey

This will be the logic behind the Create Journey use case. This step will implement the functionality to create and save a journey.

#### Search Journey

This will be the logic behind the Search Journey use case. This step will implement the functionality to search for matching journeys

#### Apply Journey

This will be the logic behind the Apply for a Journey use case. This step will implement the functionality to allow a user to apply for a seat on a journey. This will notify the driver of the journey.

#### Accept Applicant

This step will implement the functionality to accept those applicants who have applied for a journey and notify them of such.

#### Iteration Two Results

This Iteration was less successful than the first, with outside projects and work interfering in the available time. Maps were fully integrated, and the creation and searching of a journey was also done. However search needed to be refined to work within a range as this was a major selling point in the apps functionality. The Apply and accept tasks were pushed to iteration 3.

### Iteration Three

This iteration is planned for polishing the implemented features, and finishing the app to the highest standard possible. It also allows for tasks from previous iterations to be completed, as there is little core functionality required in this iteration.

#### View Journeys

This involves implementing a mechanism that will allow users to view specific journeys depending on certain factors. It will also include being able to see details about chosen journeys.

#### Code Improvement

This is a placeholder task to use the remaining time to make small meaningful updates to the app. By reserving this space for non-vital work, delays in earlier Iterations can be handled easier.

#### Iteration Three Results

At the end of this iteration the author had completed all the functionality of the app and although there were a few features he would have liked to implement; the time just wasn’t there. The core app functionality however was working smoothly, and the app could be described as a finished product, just very much a version 1.0

# Non-Functional Requirements

Non-functional requirements for this project would be:

1. Registration should take no more than a minute.
2. The process of creating a journey should take no more than a minute
	1. This process should work 99% of the time
3. The process of searching for a journey should take no more than a minute and a half
	1. This process should work 99% of the time
4. The process of accepting an applicant should take less than a minute

# Design

## User Experience (UX)

This application will aim to have a fluid and simple UX that will allow users to quickly and easily access all the functionality of the app within moments of starting it. By ensuring simplicity and ease of use, we can speed up the users interaction with the app, wasting less of the users time on unnecessary tasks, and provide a positive experience for the user.

## User Interface (UI)

The design of the UI is vital in the probable success of the app. Cluttering or unnecessary complication of the UI can have a degrading impact on the user, which may lead them to uninstalling the app, not due to functionality but form. By ensure a clean simple UI with a sombre colour palette we can make the application easy on the eye, and a pleasure to use. This will lead to better reviews from users and potentially more downloads. Screenshots of the potential UI design will be documented in the design document.

# Bibliography

Cso.ie. (2018). Number of Commuters - CSO - Central Statistics Office. [online] Available at: http://www.cso.ie/en/releasesandpublications/ep/p-cp6ci/p6cii/p6noc/ [Accessed 17 Apr. 2018].