CAIRDE

**Car Sharing App**

Research Document



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 provide a written account of the research carried out by the author – Dylan Lawlor - for the development of the carsharing application CAIRDE. This research will encompass investigating similar service providers, development technologies including coding languages, IDE’s, backends, API’s and design. Each topic will be investigated, and the findings presented here in individual sections, with a conclusion for each detailing the choices made by the author.

Contents

[Abstract 1](#_Toc511655566)

[Introduction 3](#_Toc511655567)

[Similar Services 4](#_Toc511655568)

[Iway Car Sharing 4](#_Toc511655569)

[Findings 5](#_Toc511655570)

[GetThere.ie 5](#_Toc511655571)

[Findings 6](#_Toc511655572)

[Carpoolworld 6](#_Toc511655573)

[Findings 6](#_Toc511655574)

[Conclusion 7](#_Toc511655575)

[Technologies 8](#_Toc511655576)

[Target Platforms 8](#_Toc511655577)

[Android 8](#_Toc511655578)

[IOS 9](#_Toc511655579)

[Conclusion 10](#_Toc511655580)

[Programming Languages & Development Environment 10](#_Toc511655581)

[Java & Android Studio 10](#_Toc511655582)

[Ionic & HTML 11](#_Toc511655583)

[Xamarin for Visual Studio & .NET 12](#_Toc511655584)

[Conclusion 12](#_Toc511655585)

[Backend Hosting 13](#_Toc511655586)

[Microsoft Azure 13](#_Toc511655587)

[Final Word 14](#_Toc511655588)

[Bibliography 15](#_Toc511655589)

# Introduction

CAIRDE is an application designed to facilitate and encourage car sharing. Car sharing is the act of several people travelling together in one vehicle to a mutual destination. The benefits of car sharing are multiple and include being environmentally cleaner by reducing the number of cars on the road, increased safety for the same reason, and greater social inclusion by encouraging people to communicate and share long commutes with other members of society, as well as providing an opportunity to those who don’t or can’t drive to have the same opportunities as those who do. The application achieves this by allowing registered users to create journeys for trips they will be undertaking and allowing other users to apply for a spot on that journey. Users will be able to search based on area and time, with the app showing similar posted journeys based on the searchers requirements. Upon finding a desirable journey, the user can apply for a lift, which will notify the driver about the applicant, which they can then accept or refuse. Once accepted the original user is then notified in return.

This document will examine the various technologies that could be used to develop this application, comparing their features and selecting one to use. The technologies that need investigating are the development environment (IDE), source control, development language/s, database hosting, API’s, and UX/UI design. To begin the research the author will look into similar services on the market and examine their approach, highlighting pro’s and cons of each.

# Similar Services

With the rise in environmental awareness and rising costs of owning a car, many people are looking for new ways to complete their daily commute, be it to work, college or other. There are several services out their that aim to connect users in this regard with the majority in Ireland being web based. Here are a few of the services the author found that operate in Ireland currently.

## Iway Car Sharing



Figure 1 – Iway Car Sharing

Link : <https://play.google.com/store/apps/details?id=com.iqmsoft.rideshare&hl=us>

Iway Car Sharing is an android app currently available on the play store that is similar in concept to CAIRDE. It offers much of the same functionality (create a ride, search a ride) in a fairly clean layout. The author registered for an account, and set up a profile by entering standard details such as name email phone etc, and could then use the app. The UI was easy to use, but search a lift relied on typing the address of both the source and destination with no map feedback, and the author did not find any journeys for any of the searches he performed.



Figure 2 – Landing screen after Login

 Creating a journey, or offering a ride in this app, also required typing the locations, which then brought the author to a new screen which he could set the fare, date and time as well as the number of seats available. It also had a map view but this seemed to be bugged as it failed to show the correct location.

The author created the journey and then attempted to view the journey they just created through the main menu, which resulted in a crash. At this point the author stopped using the app and moved on.

### Findings

Pros

* Clean uncluttered UI
* Available worldwide, not specific to one country

Cons

* Poor search and create functionality
* Buggy
* Apparent lack of users

## GetThere.ie



Figure 3 – GetThere.ie

Link: [GetThere.ie](https://play.google.com/store/apps/details?id=com.carpoolworld.www.carpoolworld)

This is an independent site that aims to “provide a single place to search for transport in Ireland”. It primarily offers a comprehensive journey finder by utilising public and private transport services like buses and trains, displaying departure times and routes for these services. It does however have a car sharing functionality which allows registered users to post their journeys to advertise spaces, and search for lifts. Yet again the function for creating or searching a lift is lacking, requiring the author to type in the locations, although a usefull feature was that the site can email you when a match has been found. The website itself has a slightly dated look that is nonetheless functional. It also reports that at least 70 journeys have been posted in the last 30 days.



Figure 4 – GetThere.ie Homepage with lift sharing on the right of page.

### Findings

Pros

* Relativly active userbase
* Provides information about other forms of transport
* Email notifications

Cons

* Dated appearance
* Poor interface for searching and creating journeys

## Carpoolworld



Link: <https://play.google.com/store/apps/details?id=com.carpoolworld.www.carpoolworld> CarpoolWorld is another app available on the play store which operates in Ireland. The author downloaded the app, which dissapointingly only serves as a link to a mobile site, from which the service is run. Upon signing up, which could integrate with facebook and Google+ the author had access to the service, but found the interface so cluttered and convoluted that he was initially put off. For the sake of research he continued, and while the site offers some interesting choices (designate smoker/non smoker, cost sharing) overall the process felt cumbersome and took too long.

### Findings

Pros

* Worldwide app
* Interesting additional filters

Cons

* App is just a weblink
* Cumbersome, unfriendly interface

## Conclusion

From the above and a handful of other similar services that were sampled the author found that there was a clear need for a carsharing app in the Irish market. While there were some offerings already available, most were web based or very poor implementations of an app. The general experience was similar to the authors initial vision – register, enter journey details and match up with other users. In the authors opinion none of the offerings made it particularly easy or attractive to use the service, which may discourage people from carsharing. This was the area the author would like to improve on the other services, the core process of searching for and creating a journey. The next section will discuss the possible technologies to make this vision possible.

# Technologies

This section will detail the research the author carried out in relation to development languages, target platforms, IDE’s and backend hosting. Each section will detail the authors findings as well as chosen technologies. Given that a choice in one of these categories will by its nature define the possibilities in another, each area will be investigated in turn – Platform, Language then Backend, with the choice from the previous section influencing the research of the next, in this way there will be no time lost investigating a language and framework for a platform that is ultimately not developed for.

## Target Platforms

### Android



Figure 5 – Android OS

Android the Google owned and developed operating system and is one of the most popular mobile OS in the world powering devices in over 190 countries. Its built upon the open-source OS Linux. Due to its open-source nature it is possible for developers to create highly customisable versions of the OS for their devices, and gives app developers a highly versatile and powerful platform on which to create apps. Android OS is found in numerous technologies in its current form, including but not limited to smartphones, smart watches, tablets, laptops, TVs and cars.

Android OS is provided in several different versions depending on the age of the devices hardware and manufacturers decisions. The most recent version of Android is Oreo, Android version 8.0 & 8.1, although it is only available on the very latest devices. For this reason developers generally target a lower form of android to maintain compatibility with the majority of users devices. The following table contains a breakdown of the current usage share of the various Android OS’s.

Current Android Usage Share – October 2017

|  |  |  |
| --- | --- | --- |
| **Android Name** | **Android Version** | **Usage Share** |
| Marshmallow | 6.0 | 32% |
| Lollipop | 5.0, 5.1 | 27.7% |
| Nougat | 7.0, 7.1 | 17.8% |
| KitKat | 4.4 | 14.5% |
| Jelly Bean | 4.1.x, 4.2.x, 4.3.x | 6.6% |
| Gingerbread | 2.3.3 to 2.3.7 | 0.6% |
| Ice-Cream Sandwich | 4.0.3, 4.0.4 | 0.6% |
| Oreo | 8.0 | 0.2% |

From - https://fossbytes.com/most-popular-android-versions-always-updated/

From this table its clear that only targeting the newest version of android – Oreo – would result in a tiny slice of potential users, and so development on this platform would ideally target a lower version number for increased compatibility.

Android is a very popular development platform and developers can choose to develop in many different languages, not just Java. This will be explored in a later section in more detail – see Development Languages.

### IOS



Figure 6 – Apple & IOS

Apple shook up the mobile market in 2007 with its release of the inaugural IPhone. It was the first true smartphone that got rid of the old method of using a keypad to navigate a phone’s menus, instead opting for a full screen touch screen. Multiple versions of the phone released in subsequent years building on the existing platform and innovating the market for many years. However the rise of Googles Android began to erode Apples market dominance due to the wider array of choices the open source OS offered, as well as the comparatively high pricing that came with Apples own devices. While Android is undoubtly the bigger of the two in regards to market share (>90% of smartphones run a version of Android), IOS remains a popular platform with a dedicated user base.

IOS is known for its strict development systems, with Apple having the majority of control on what can be done on the device, and the final say on what is allowed on the App store. App development requires an Apple device for testing (often enough of a price barrier to stop enthusiast developers) and can be developed for in several languages including Objective C, C, C++ and various other languages so long as they conform to apples rules regarding interpreters.

### Conclusion

The author decided from an early point that he would focus first on development of a native Android app, given Androids leading share in the market, and greater flexibility for developers. With regards the target version of Android, its is clear from the statistics above that in order to target the highest percentage of Andriod user, a developer should target version 4.4 (KitKat) and above, which would be compatible with 92% of the Android market. So the minimum version the CAIRDE app will support will be Android 4.4. No other platform (IOS, Windows) will be developed for in the scope of this project, although steps may be taken to allow this at a later stage.

## Programming Languages & Development Environment

With the decision to develop an Android only application made, the author next had to choose what programming language to use for development. There is quite a bit of overlap in choosing a programming language and a development environment, so the author will discuss both together in this section. The author was familiar in name with a few choices he could use, mainly from hearsay in work and a brief stint developing an android app in 3rd year in college. With this baseline he decided to investigate 3 potential candidates, Java in Android Studio, HTML, JS & CSS with Ionic, and C# using Xamarin for Visual Studio.

### Java & Android Studio



Figure 7 – Java logo

Figure 8 – Android Studio, based off IntelliJ IDEA

Android studio is the de-facto IDE for android app development, being the official IDE for Googles Operating System. It is built upon the very popular IntelliJ IDEA, and is available on Windows, Mac and Linux. Among other things it offers a very rich layout editor that allows users to drag-and-drop UI components and the option to preview layouts on multiple screen configurations. Aswell as supporting Java, the current version 3.0 and above support other languages such as Python and Kotlin. The author had previous experience with Android Studio and Java, haven built simple apps before with them. The author found Android studio ok to work with, but the emulator was very resource intensive at the time of testing, and Java not a preferred language to use.

### Ionic & HTML



Ionic is a complete open-source SDK for hybrid mobile app development. The original version was released in 2013 and built on top of AngularJS and Apache Cordova. The more recent releases, known as Ionic 3 or simply "Ionic", are built on Angular. Ionic provides tools and services for developing hybrid mobile apps using Web technologies like CSS, HTML5, and Sass. Ionic allows developers who are more familiar with web development to develop applications that will run on consumers devices, while allowing them to leverage their web based language skills such as HTML, CSS and JS. The author was suggested to look into Ionic by a collegue in work, and despite not having the best web development skills decided to investigate it. Setup was easy and starting a new project is a simple command - ionic start myApp tabs – with the last word tabs specifying the template to begin with, giving a quickstart to app development. However further development of more advance functions like maps integration was difficult for the author and while Ionic is a popular choice and easy to use, it presented a steep learning curve for the author.

### Xamarin for Visual Studio & .NET

Figure 10 – Xamarin – a C# based framework for Mobile Development

Figure 9 – Visual Studio – Microsoft’s IDE

Xamarin is a Microsoft owned subsidiary that allows for the development of Mobile apps from within Visual Studio, leveraging the huge library of PCL (Portable Class Libraries) available within the .NET Core framework. Xamarin provides a developer the tools to develop native apps for Android, IOS and Windows devices, as well as cross-platform apps by using Xamarin.forms, which allows the use of portable controls which are mapped to native controls for each device. Xamarin’s integration with Visual Studio also allows developers access to a drag and drop development platform for building the UI, similar to Android Studio. Coding is done in C# and users have access to a wide array of packages thanks to the Nuget package manager, allowing for a wide range of functionality relatively easily. The author found development of sample apps to be very easy with this platform, mostly due to an already high degree of familiarity with Visual Studio from his employment, as well as the familiarity of coding with C#. Visual Studio also provided a means ofsource control using VSTS – Visual Studio team Services – which is powered by Git and built into Visual Studio. This simplifies the act of dealing with external source control utilities.

### Conclusion

While there are many platforms and languages on the market today, each with their relative strengths and weaknesses, the author found Xamarin to be the obvious personal choice due to the familiarity with C# and .NET core, as well as the IDE Visual Studio. Having access to VSTS was an added bonus, reducing the number of additional programs needed compared to other choices – Xamarin required just Visual Studio with the Xamarin plugin installed, while Android Studio would require external source control, and Ionic any number of additional add-ons plus source control. In the end it was a case of playing to the authors strengths and building upon a solid foundation.

## Backend Hosting

The application will require some amount of backend hosting both for the database as well as the API to interact with the database, as it is very bad practice to communicate directly with the database through the app. Given the decision to develop the app using Xamarin and Visual Studio, there were several options when it came to backend hosting, but only one was investigated in any meaningful way. While MariaDB and MongoDB were considered, with a PHP API to interface with them, Microsoft’s Azure was the obvious choice for the author.

### Microsoft Azure

Microsoft Azure is Microsofts cloud computing service that provides a wide array of services and tools. It provides over 600 services including web hosting, database hosting, cloud storage, virtual machines, serverless computing and many more. They offer a range of pricing models for each service, with many services having a basic free tier. At the time of this research the author was in the process of investigating Azure for his employment, and so this fitted in well with this project. Azure provided a means to supply all the required backend functionality for the application in one place, almost for free, with the database the only service requiring an monthly fee of just over €4 to use. The database was a SQL database hosted in the cloud and always available. Creating a database is as easy as clicking a button and filling in some details thanks to Azures GUI Dashboard. In addition to the database Azure also provided a web app service which would integrate with the application and database providing the link between the two. While it took some configuring and a few attempts to set up correctly, once completed and running the web app made communicating with the database from the app safe, secure and easy. By posting C# objects to the service, the web app then translated the call into SQL for the database, reducing the programming load on the developer.

All of Azures functionality integrates seamlessly into Visual Studio, with tutorials and Documentation providing a solid base upon which to begin building a knowledgebase. It was the obvious choice for the author given that it ticked every box for required functionality, was always online, and cheap. It also had the added advantage that it was fully scalable, meaning a shift from development app to production app required no more than a scale up of resources (and cost), thereby future-proofing the approach.

# Final Word

This concludes the research document. Research on the various technologies and options was carried out and a suite of tools and development choices were made on the back of this research. The results of the research can be summarised as follows:

* Application will be developed for the Android platform.
* Minimum Android version supported will be 4.4
* Development will be done using Xamarin for Visual Studio
* Source Control will use Visual Studio team Services
* Backend Hosting will be done using Microsoft Azure

# Bibliography

Tiwari, A., 2018. Most Popular Android Versions 2018 (Always Updated List). Fossbytes. Available at: https://fossbytes.com/most-popular-android-versions-always-updated/ [Accessed April 16, 2018].

AndroidPub. (2018). Apple Vs Android — A comparative study 2017 – AndroidPub. [online] Available at: https://android.jlelse.eu/apple-vs-android-a-comparative-study-2017-c5799a0a1683 [Accessed 16 Apr. 2018].

Azure.microsoft.com. (2018). Microsoft Azure Cloud Computing Platform & Services. [online] Available at: https://azure.microsoft.com/en-us/ [Accessed 16 Apr. 2018].

Developer.android.com. (2018). Download Android Studio and SDK Tools | Android Studio. [online] Available at: https://developer.android.com/studio/index.html [Accessed 16 Apr. 2018].

Getthere.ie. (2018). Bus, Rail & Lift Sharing in Ireland - GetThere.ie. [online] Available at: http://getthere.ie/ [Accessed 16 Apr. 2018].

Ionic Framework. (2018). Ionic Framework - Ionic Pro. [online] Available at: https://ionicframework.com/pro [Accessed 16 Apr. 2018].

Logomakr.com. (2018). Free Logo Maker - Create your own logo in minutes!. [online] Available at: https://logomakr.com/ [Accessed 16 Apr. 2018].

Play.google.com. (2018). [online] Available at: https://play.google.com/store/apps/details?id=com.carpoolworld.www.carpoolworld [Accessed 16 Apr. 2018].

Play.google.com. (2018). [online] Available at: https://play.google.com/store/apps/details?id=com.iqmsoft.rideshare&hl=us [Accessed 16 Apr. 2018].

Xamarin.com. (2018). Mobile Application Development to Build Apps in C# - Xamarin. [online] Available at: https://www.xamarin.com/platform [Accessed 16 Apr. 2018].