Name: Stephen Cassedy

Project Title: College Carpool

Document: Functional Specification

Student Number: 13391736

Date Finished: 25/11/2016

Index

1. Introduction

1.1 Overview

The application I have chosen to develop is called CollegeCarpool. It is an android application that will act as a carpooling service to all Dublin City University students, both past and present. The application will allow users to either act as a passenger or a driver for the service. The aim of the application is to try and reduce traffic in the Glasnevin/Beaumont/Whitehall area by encouraging people to car share. This would theoretically lead to safer roads and less pollution also. I felt that due to the huge amount of traffic in the area, this application could greatly improve the travel style of both drivers and pedestrians alike.

1.2 Document Conventions

This application is intended as a mobile application developed using the Android operating system. Any future references to the application will assume this.

For the sake of convenience, application will also be shortened to app. in this document.

Dublin City University will be referenced in the future as DCU.

1.3 Glossary

Any terms defined in *italics* in this document will be defined in here.

- Near Frequency Controller (NFC) - A small chip able to be stored in a device such as a phone that uses radio waves over lengths of 4 inches to communicate with another NFC chip and pass information.
- Firebase – A real time cloud service for syncing data across multiple clients. Contains an API to allow developers use it for mobile app development.
- Splash Screen – A view containing an image or logo pertaining to the program being launched.

2. General Description

2.1 Product/System Functions

As I have stated previously, the app will function as a carpooling service for DCU students. The aim of targeting DCU students is to enhance the current sense of community between the students who attend the university. Many people commute from the countryside and could easily pick up people on their way into the city; the only problem being that they do not know who to pick up. I believe that given the chance (and incentive such as payment to cover petrol costs), students who drive would be a lot more willing to help those who don't. Targeting DCU students, both as drivers and passengers, is also safer. As a student myself, I would feel a lot more comfortable transporting another student who I share a university in common with instead of a stranger. All information on a person is available through their e-mail if an incident were to occur. This is why users will only be able to log in to the app with a "mail.dcu.ie" e-mail address. I have also chosen to create this app with the intention of reducing traffic congestion and pollution. Less cars on the road would also lead to safer roads around DCU and possibly reduce rates in traffic accidents.

The main functions in the app will be either to act as a driver or as a passenger. If driver is chosen, the user will be able to add pickup locations either manually or from a map. They will then be able to choose to either plan a journey in either real-time or for the future. If passenger is chosen, users will be able to set a pickup location either manually or through GPS by tracking where their phone is. Once again they can request a car ride or plan a journey for the future.

I hope to incorporate a payment system implemented with *Near Frequency Controller (NFC)* cards. This will allow users to make payment without having to use cash. I would like to stress at this point that I don't intend in this application being used for commercial reasons. Any payment to be made I would like to be for covering costs of travel. Drivers can make a small profit but the price to be paid for the ride will be decided between passenger and driver.

The complexity behind the app is to use traffic prediction to tell drivers which routes are best to take at any given time. For real time data, I intend on using the Google Maps API. For planning routes, I plan on using historical data from both the National Transport Authority and Dublinked.

The system will support a user signup, login and forgot password function for user accounts.

An function I wish to implement is private and group messaging between all users but at the moment, this is a 'wishlist' item that I may not incorporate.

2.2 User Characteristics and Objectives

The main user community for this app will be DCU students in the age range 18 – 25 years old. Users will not be expected to have high expertise using apps as the front end is based off of a very simple premise. All that is required of them is to have an android device capable of running the app.

When using the app, a user's objective should be the task of getting or giving transport to a given location. A passenger should be able to set where they want to get picked up from to obtain a lift while a driver should be able to set a location where they wish to go and be presented with passengers who wish to go that direction too.

2.3 Operational Scenarios

This section will describe the steps both a passenger and a driver will take upon entering the app. Both cases assume that the user has a DCU e-mail address and has already registered.

Actions marked with "*" are optional and are not a necessary step but are still features of the app.

Driver Scenario

- Login with user credentials.
- Select driver.
- *Enter e-mail addresses of pickups manually in highlighted entry fields.
- Enter the map section indicated by a 'View Map Button'.
- *Select pickups by clicking on passengers who appear on the map. They will be marked by highlighted dots.
- *Select 'Go Now' which will allow the driver to make the journey using real-time information.
- *Select 'Plan Ahead' which will allow the driver to plan the journey by entering dates and times they wish to travel on. This travel route will be based off of historical data.
- Begin journey and collect passengers.
- *Deliver passenger to desired location and enter 'Payment' Screen.
- *Enter number for amount to be paid, agreed by driver and passenger.
- *Hold phones together and complete payment via NFC card.

Passenger Scenario

- Login with user credentials.
- Select Passenger.
- *Enter a pickup location in the highlighted field/Use GPS to set pickup location to current location.
- Enter the map section indicated by a 'View Map Button'.
- *Select 'Request Ride Now' which will immediately broadcast you for pickup on the map.
- *Select 'Plan Ahead' which will allow you a plan a trip in the future. (Side note – a messaging function in the app would make planning journeys much easier as you could be in contact with the driver and confirm the dates and times).
- Await collection.
- Get picked up and brought to your desired location.
- *Upon arrival enter the payment screen.
- *Enter number for amount to be paid, agreed by driver and passenger.
- *Hold phones together and complete payment via NFC card.

2.4 Constraints

Time

Due to the nature of my project, it has many parts that can be added and modified. The app itself is likely to grow and evolve as time goes by so implementing all of the features I wish to desire could be a challenge given the time frame that I have to do it in. I can only begin to add the features that make it a fully rounded app once I have the core system working.

Server Account Limitations

As of the time of writing, user accounts for the app are being managed by *Firebase*. This only allows 50 user accounts to be active until they require payment so if the app does end up being used, I may have to consider where account information is stored.

NFC cards

NFC cards are still considered a new technology even though they work off basic radio waves. Not all phones are shipped with them so the payment system in the app will be limited to phones that have a card in them. I hope to test the app on phones that have an NFC card and on phones that don't have one.

3. Functional Requirements

3.1 Registration and Login

Description

This function is the first step that a user will take to use the app. When they open the app and get by the *splash screen*, they will be presented with a login screen. Users must use a DCU e-mail address and create a password to login. A confirmation e-mail will be sent to the address they enter. A registration and forgot password option will also be available for first time users and users who have forgotten their password respectively.

Criticality

This function is an essential part of the entire app. It limits users to those with a DCU e-mail and is the only way to gain access to the app. It allows for the recovery of lost passwords too which is quite a useful feature when a person has many different passwords in the modern age.

Technical Issues

Having a user login involves storing information in a database. At the moment it is being handled by Firebase but I may switch that to a different database in the future.

Dependencies

I plan on drawing usernames from the DCU email address. This name will be displayed as the name for both the passenger and driver. This information will be used through the entire app.

3.2 Allow User to Act as a Driver

Description

A user must be able to designate themselves as a driver if they desire. Being a driver involves being able to choose pickup addresses and to enter addresses manually or choose them from the map. A driver must also be able to both plan a trip or select to depart immediately. Finally, they must be able to accept payment via NFC card.

Criticality

Without the ability to act as a driver, there will be no journeys being made. The app would be pointless.

Technical Issues

Google Maps has native functions for planning routes but I would like to use my own historical data to use for planning trips in the future. I hope this will provide more accurate information in traffic in localized areas.

Dependencies

Becoming a driver depends on the user login being successful. Passenger pickup also depends on a driver being there to pick them up.

3.3 Allow User to Act as a Passenger

Description

Users must be able to designate themselves passengers. They must be able to select a destination to be picked up from either manually or through GPS. Once again, they must be able to put themselves up for collection immediately or plan a journey in advance. Payment via NFC card is also required for the passenger function to work correctly.

Criticality

Without this aspect of the app, the point of carpooling is defeated as drivers will not be ride sharing.

Technical Issues

The user will have to be able to place themselves on the map once they request a ride. How I'll differentiate between users who want a lift immediately and those who want a lift in the future is yet unclear to me. I plan on having the users marked as dots in the map so a possible way to solve my problem might be to have different colours per user depending on how far away their journeys are.

Dependencies

For the passenger function to work correctly, driver must be implemented. Else no rides would be given. Preferably, passengers would also have driver information so that they know the name of the person that they are getting picked up by. Once again a messaging system might help to make the experience a lot smoother here.

4. System Architecture

4.1 System Architecture Diagram

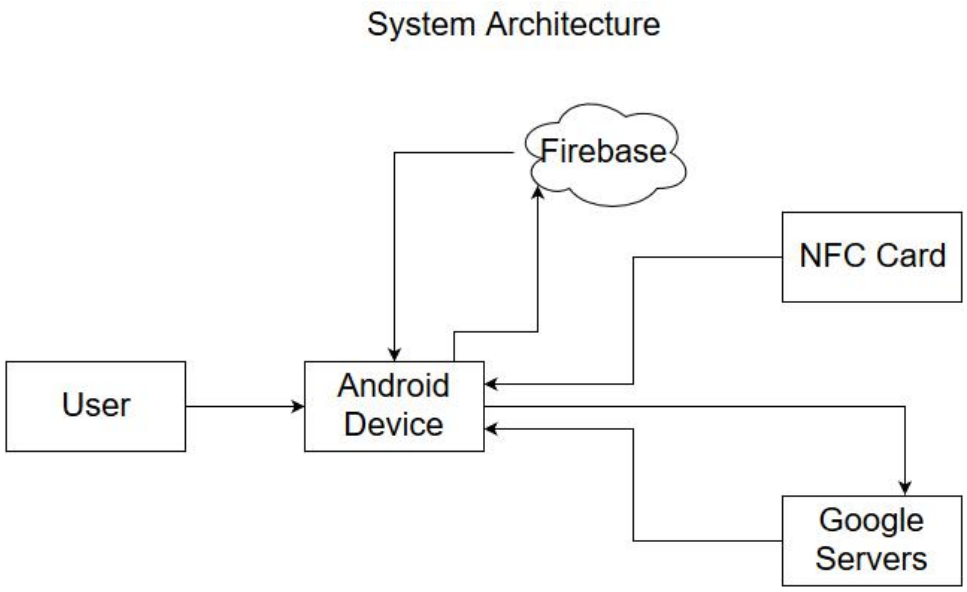

Fig 4.1: The above figure illustrates the planned architecture of the system. The user will generally only see what's displayed on the android device while it relays information between the cloud services and awaits a request from a NFC card.

4.2 User

The user will be any given user who has signed up to use the service and has an active account. They will navigate the User Interface (UI) to interact with the other elements of the app.

4.3 Android Device

The android device is any device running the Android operating system that is capable of using the app. NFC features depend on phone make and age.

4.4 Firebase

Firebase is the cloud service that I intend on using as a database. User will not see interactions with this service. This will store information gathered from analysis of historical data.

4.5 NFC Card

The near frequency controller card is what I intend on using to sup[port a payment system in the app. Phones must be held together within a distance of 4 inches to allow communication between 2 cards.

4.6 Google Servers

Google Servers will communicate data to me about real time information on traffic.

5. High Level Design

5.1 Context Diagram

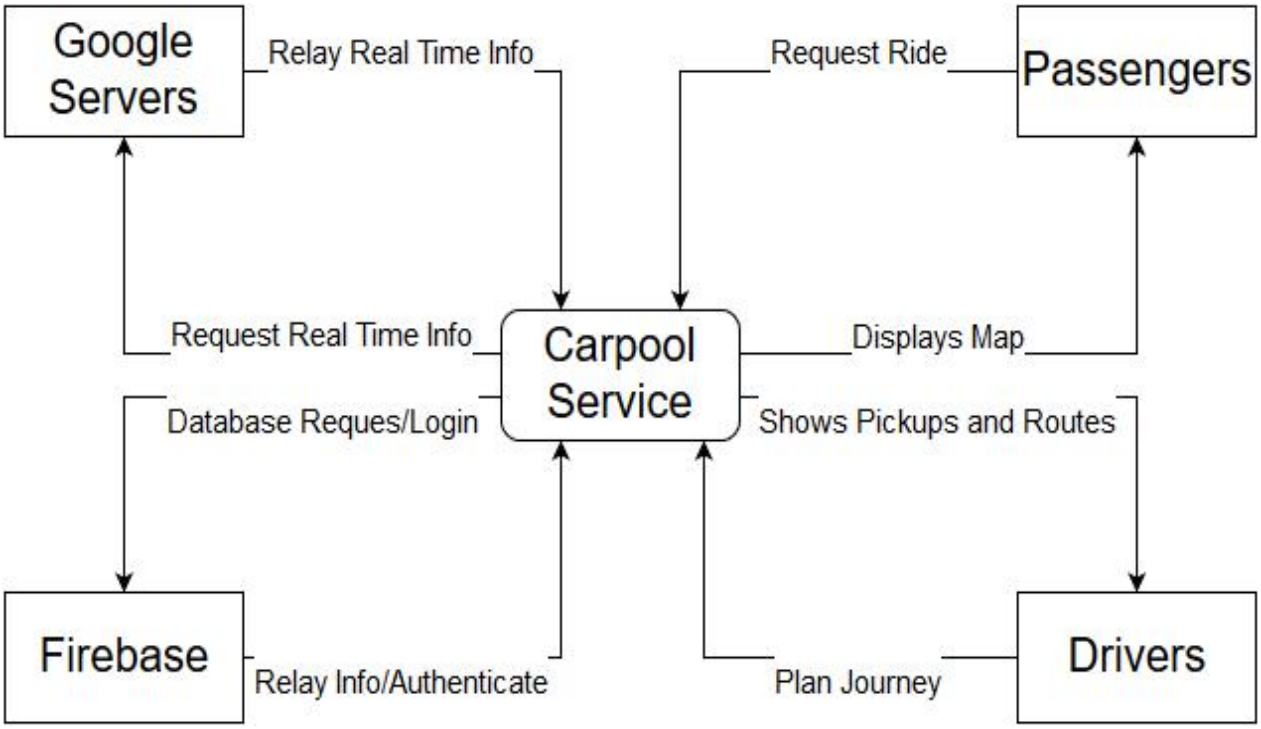

Fig 5.1: The context diagram in figure 5.1 shows how external entities interact with the system and what sorts of communications are happening. In this case it shows the users it will interact with (passengers and drivers) as well as the services (Google Servers and Firebase).

5.2 Data Flow Diagram

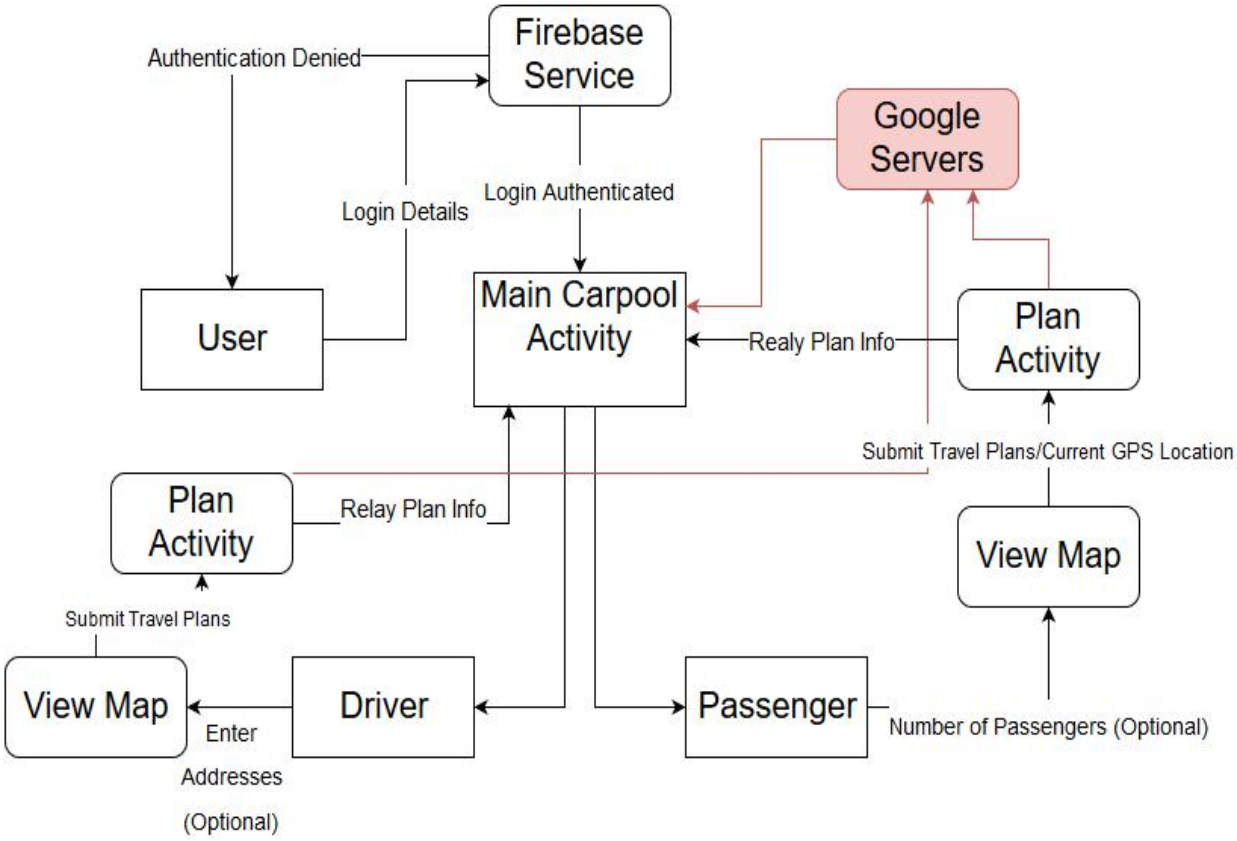

Fig 5.2: This diagram represents a data flow diagram. It indicates where information is travelling through the system. Services/Functions are represented by rounded boxes while entities are represented by square boxes. Boxes are connected by arrows which indicate the path to take. The red line indicates the connection with the Google Servers. This is in red to highlight it because as of yet, how the system interacts with the servers exactly and what output I'll be getting from Google Maps is unclear.

6 Preliminary Schedule

6.1 Tasks

There is a lot of work to be done to get the project off the ground. The task list is as follows:

- Establish Login/Registration/Forgot Password System.
- Enable Choice Between Passenger and Driver.
- Make Input fields optional for Driver and Passenger.
- Establish how to obtain GPS location for passenger and implement.
- Create map view to place routes/passengers on.
- Analyse dataset containing traffic details for certain points around DCU.
- Establish connection between dataset analysis and Firebase to store results.
- Enable 'Go Now' features for Driver which will show real time/ predicted routes to take.
- Enable 'Plan Ahead' feature for Driver and Passenger.
- Enable a wallet feature to contain 'money'/bitcoin.
- Enable the NFC feature for payment.

6.2 Gantt Chart

ID	Task Name	Start	Finish	Duration
1	Login/Registration/Forgot Password	25/11/2016	30/11/2016	4d
2	Passenger/Driver Choices	30/11/2016	01/12/2016	2d
3	Address Input Fields	05/12/2016	07/12/2016	3d
4	GPS Location	07/12/2016	09/12/2016	3d
5	Create Map for Routes/Passengers	09/12/2016	31/01/2017	38d
6	Analyse Dataset for Predictions	01/02/2017	07/02/2017	5d
7	Establish Connection Between Dataset and Firebase	07/02/2017	27/02/2017	15d
8	Enable Go Now Feature for Driver	27/02/2017	10/03/2017	10d
9	Enable Plan Ahead Feature for Passenger and Driver	27/02/2017	10/03/2017	10d
10	Enable Wallet with NFC Feature	10/03/2017	23/03/2017	10d
11	System Testing	23/03/2017	29/03/2017	5d
12	Write Documentation	23/03/2017	27/03/2017	3d
13	What can I improve?	27/03/2017	07/04/2017	10d

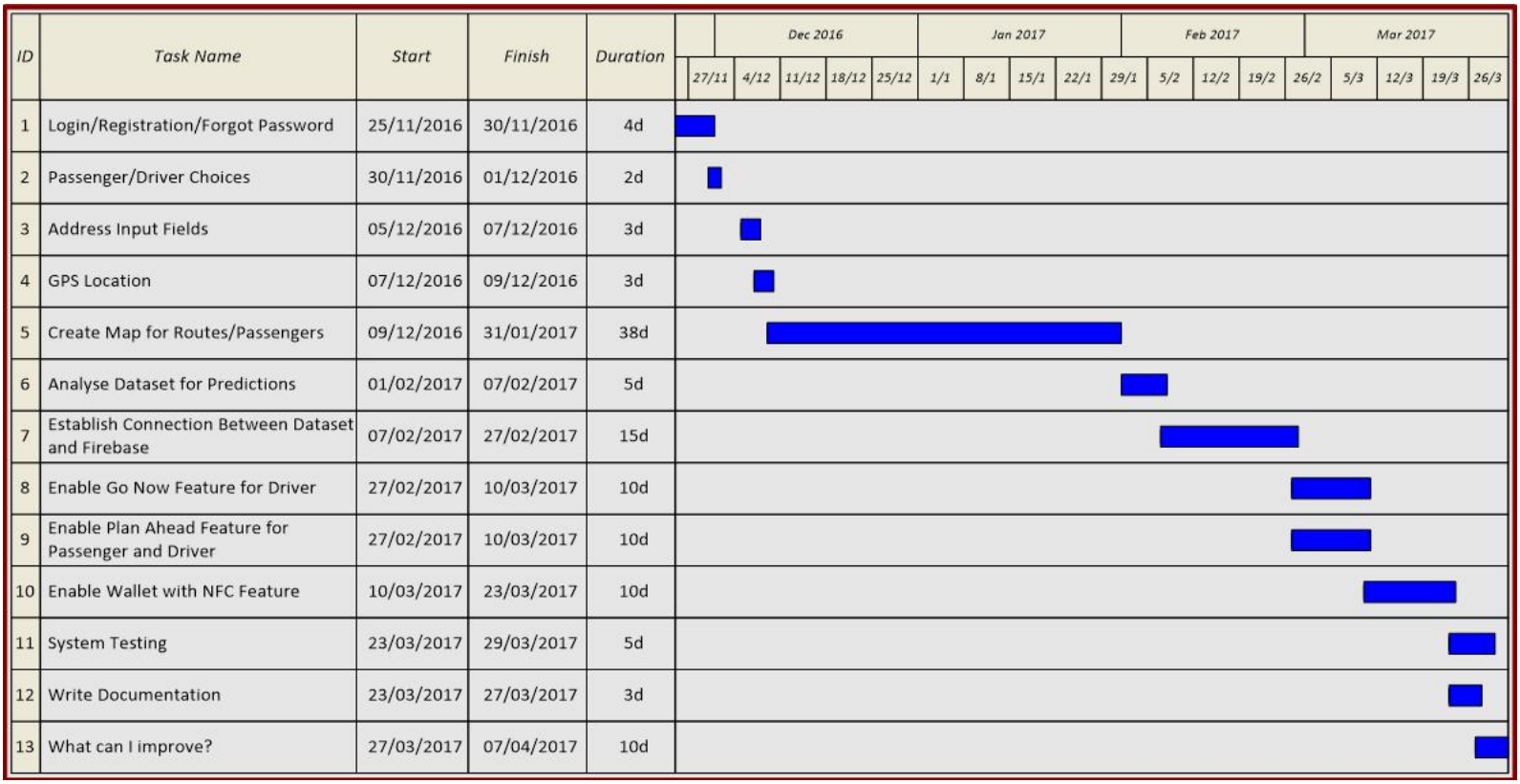

Fig 6.2: This Gannt chart shows the progress I wish to take. It is a rough outline of what I hope to achieve in given times. These are subject to change as the project evolves but it is a rough indication of how I will be spending my time.

7. Appendices

7.1 Wish List

I'm including a wish list just to state one extra thing that I would like to implement if I have any time left at the end. I would like to introduce an investigation into the psychology behind getting stuck in traffic. My investigation would lead me to question what affect would getting stuck on a particular route have on a driver. By this I mean that if a driver took one route when it was suggested as the best route to take and continually got stuck in traffic, would they continue taking it or would they begin to switch routes. I think this would be an interesting research topic to do provided I have a small test group to work with, i.e. App users, and have enough time to begin the research.