COMBATTING THE DE-SKILLING OF MODERN DRIVERS THROUGH THE DEVELOPMENT OF AN IMPLICIT IN-CAR NAVIGATION SYSTEM

Charles Lonsdale (130177276)

Supervisor: Dr. Dave Kirk
Word count: 14996

Abstract

There can be no doubt as to the numerous benefits the increase in contemporary satellite navigation usage has brought; however, it's uptake has also led to a number of negative repercussions for the modern driver. Foremost, the average Sat Nav user fails to learn and develop basic navigational skills while following explicit turn-by-turn instructions. In an attempt to combat this de-skilling, this project involves the development of an alternative navigation system using implicit audio feedback. This feedback is intended as a supplementary navigational aid to guide the driver towards their destination, as opposed to explicitly telling them which route to take.

Upon conclusion of the development I test the system in a real-world scenario using 6 volunteer drivers, in a head-to-head test against a stock Sat Nav system. Qualitative and quantitative results are collected showing that the implicit system is better for exploring an area and developing navigation skills, although at the cost of being more stressful and more challenging to use. Despite further work being required in the area of implicit navigation in order to assess long-term benefits and safety concerns, I conclude that the logic behind the idea is sound and could be integrated into a current system to aid modern drivers in developing navigation skills.

Declaration

I declare that this dissertation represents my own work except where explicitly stated.

Acknowledgements

I would like to thank my dissertation supervisor Dr Dave Kirk for his support and input throughout the entire project.

I would also like to thank the volunteers that took time out of their schedules to help me with the initial and final user testing sections of the project.

Finally, I would like to thank my family and friends for their support during the year.

Contents

Introduction

Purpose

Prior to the advent of in-car GPS systems, journeys would often involve a passenger searching through the glovebox of the vehicle for an A-Z or a fold-out map of the area, in an attempt to find their current location and how to get to their destination. Drivers would rely on road signs, instincts and the knowledge of passers-by to get to where they wanted to go. Now though, all of these interactions and all of this knowledge has been replaced by the seemingly ubiquitous automotive satellite navigation system; indeed, data shows that the number of cars that contain some form of Sat Nav system (either built-in or plug-and-go) rose by 20% in England between 2009 and 2014 (Department for Transport, 2015).

With this change in the method of navigation while driving comes a change in the skills required on the part of the driver to get to a destination; a number of researchers argue that the use of GPS technology with explicit turn-by-turn based instructions leads to the de-skilling of drivers (Leshed, et al., 2008). Every once in a while, a story appears on the news of a driver blindly following their Sat Nav instructions and driving into a field, river or lake (Goessl, 2012). Whilst such cases are extreme examples, they serve to highlight the fact that modern day drivers can be too focussed on electronic guidance and not focussed enough on learning crucial navigational skills, skills which may come in handy down the line.

The focus of this dissertation is therefore to create a more implicit in-car satellite navigation system that serves to guide the driver towards their destination whilst still letting them develop and use navigational skills. My hope is that the system will allow the user to engage more with their surrounding environment without the fear that they may get completely lost and become unable to find their way back. It will hopefully therefore benefit people who find themselves in a situation where they are without a navigation system, as through using my application they will have learned not to rely on explicit instructions. The successful completion of a journey using my system will hopefully lead to *“a sense of fulfilment and accomplishment”* in the task, whereas *“GPS technology takes that experience away”* (Aporta & Higgs, 2005).

Aims & Objectives

At the beginning of my project, I broadly set out my aim as follows:

> *"To create a working in-car navigation system using implicit instructions to improve user experience."*

In order to achieve this aim, I set out six main objectives. The successful completion of these objectives would lead to me meeting my project aim. The objectives are:

1. *To complete a thorough literature review to provide a knowledge base for the design and implementation of my system.*
2. *To create and use a Wizard-of-Oz prototype in order to test basic user interaction concepts.*
3. *To use findings from the initial user tests to design a suitable application.*
4. *To create a series of working versions to evaluate the suitability for each to meet the user's needs.*
5. *To create a final working system for testing.*
6. *To test my final system with real users against a turn-by-turn system to compare the strengths of both implementations.*

Outline

This report begins with a section on background reading in which I study and evaluate scientific papers and other resources to gain a base knowledge of satellite navigation as a whole and of the specific problem I am tackling.

Next there is a section on the design process of my application. This includes primary research into the feasibility of my concept as well as the overall design and implementation of my final solution.

Afterwards, I complete a round of testing using real-life users, testing the extent to which the system I have created meets up to my specified aim, as well as an evaluation of my overall performance throughout the project.

Finally, I conclude my findings and discuss the skills I have developed throughout the project along with potential follow-on work in the area.

The final sections contain any literary references used as well as an appendix of items referred to directly or indirectly in the text.

Background

What is Satellite Navigation?

An automotive satellite navigation system is a system traditionally located in a car or other vehicle that gives the driver explicit directions in how to get to their destination. By entering a desired destination, they are asking the system to show them the quickest route between their current location and the place they want to go to.

Before the route can be calculated for the driver, a number of complex processes have to occur. Firstly, the system has to work out where on the planet it is currently located, something it does by communicating with the Global Positioning Service (GPS). The GPS is a network of 31 medium Earth orbit (MEO) satellites operated by the United States. By communicating with at least three of these satellites and finding its distance from each, the Sat Nav can use a process called trilateration to calculate its own position (Mio Technology, 2012).

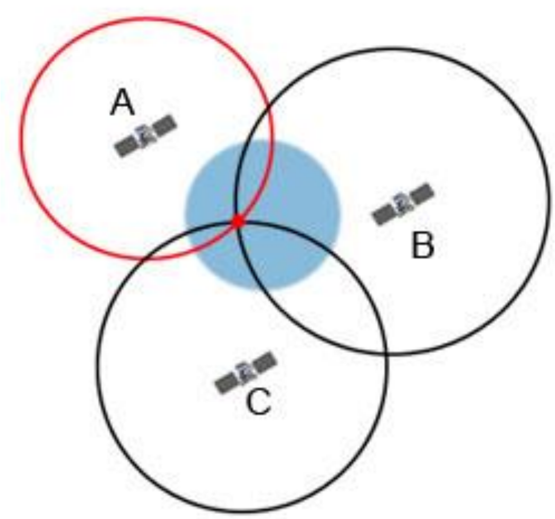


Figure 1: A simplified diagram of trilateration (physics.org, n.d.)

Once the Sat Nav knows its own latitude and longitude, it then has to find the latitude and longitude of the driver's destination, something which it does through the process of geocoding. As the user is clearly not expected to know the geographic coordinates of their desired destination, their textual input is fed into a system called a geocoder which queries a road database to find the coordinates of the location. The road database consists of a vector map with points of interest (i.e. roads and house numbers) stored as geographic coordinates (GIS Center, 2008).

Once the location of the Sat Nav and the location of the destination are known, the next stage is to use a route-finding algorithm to calculate the best route between the two points. As all the roads are stored in the system as a vector map, route finding is a shortest path problem with the roads being used as weighted vectors depending on their length. A version of Dijkstra's algorithm called the A* algorithm can therefore be used to calculate the shortest path, this algorithm factors in the straight-line distance from the target of each node so that only paths that lead to the target node are considered (TechRadar, 2010). Once the route to the destination is generated, the Sat Nav simply has to keep checking the position of the vehicle during the journey, and may have to recalculate if the driver goes off route.

History

Over the years, there have been a great number of commercial fields that have benefitted from technological advancements originally intended for military purposes; it can be argued that without these, fields such as the aeronautical (Braddon, 1999) and computing (Computer History Museum, n.d.) industries may not have grown quite as rapidly as they did. Likewise, satellite navigation started out life as military technology with the first operational system being the Transit satellite, a system that was designed so that the US Navy could accurately fire missiles from their ballistic submarines (Danchik, 1998). However, it was only in 1978, when the Cold War nuclear arms race prompted the US Congress to fund a network of navigation satellites that the GPS satellites we know today were born.

Although the GPS network had clear commercial potential from the start, its existence was initially kept a secret due to the Cold War tensions. This remained true until 1983, when Korean Airlines Flight 007 was shot down after accidentally venturing into Russian airspace. US President Ronald Reagan immediately declassified GPS and allowed its use commercially so that systems could be created to prevent similar future incidents (Brustein, 2014). The commercial signal sent via GPS was initially fuzzed to prevent successful usage against the United States by their enemies but in 2000 President Bill Clinton ordered the end of this signal degradation, allowing civilian usage the same precision as military (United States Office of the Press Secretary, 2000).

Despite commercial satellite navigation only becoming particularly accurate at the turn of the millennium, attempts had been made long before this to create a functional automotive navigation system. One of the earliest and most important came from the Oldsmobile company, whose Guidestar system was developed firstly in rental cars in the early 1990's before being released commercially in 1994 (Jesda.com, 2012). The primitive system cost $2000 and came without now-standard features such as live traffic updates, leading to sales figures of just 2000 units (Orlando Sentinel, 1996).

The next great technological advancement in the industry came with the surge in usage of external plug-and-go Sat Nav systems around the mid-2000's. Companies such as Garmin, TomTom and Magellan competed for market share as the falling prices of the technology led to an increase in the usage of portable devices (Lendino, 2012). From this, the concept of Sat Nav systems being built in to smartphones was born and as the smartphone has become more and more ubiquitous, their usage for in-car navigation has increased similarly (Department for Transport, 2015). TomTom and Garmin have now moved into this area by releasing their own apps and Android phones come with Google Maps Navigation pre-installed.

Since the Oldsmobile Guidestar came out over 20 years ago, Sat Nav technology has continued to advance, with a whole array of new features having been added. On top of the standard route-finding and turn-by-turn instructions, many systems now get real-time data from sources such as the Highways Agency and traffic information services to dynamically alter routes in cases of road congestion (SimpleMotoring.co.uk, 2011). Some systems contain a list of pre-programmed points of interest such as restaurants, petrol stations and ATMs for the user to select from. Certain Sat Nav systems allow the driver to select the type of route they wish to take, with routes optimised based on which is shortest, quickest, most economical and even which gives the best user experience (Pfleging, et al., 2014).

Sat Nav issues

The steady increase in usage of the various types of Sat Nav (see Figure 2) shows that many believe the invention and development of the system fundamentally benefits the modern driver. It is argued that satellite navigation has improved our ability to get to where we want to go in a quick and easy manner, as well as making it very difficult in the modern age for anybody to get truly lost any more. However, there are some who argue that the current batch of Sat Nav systems do not always have a positive impact on the drivers who are using them.

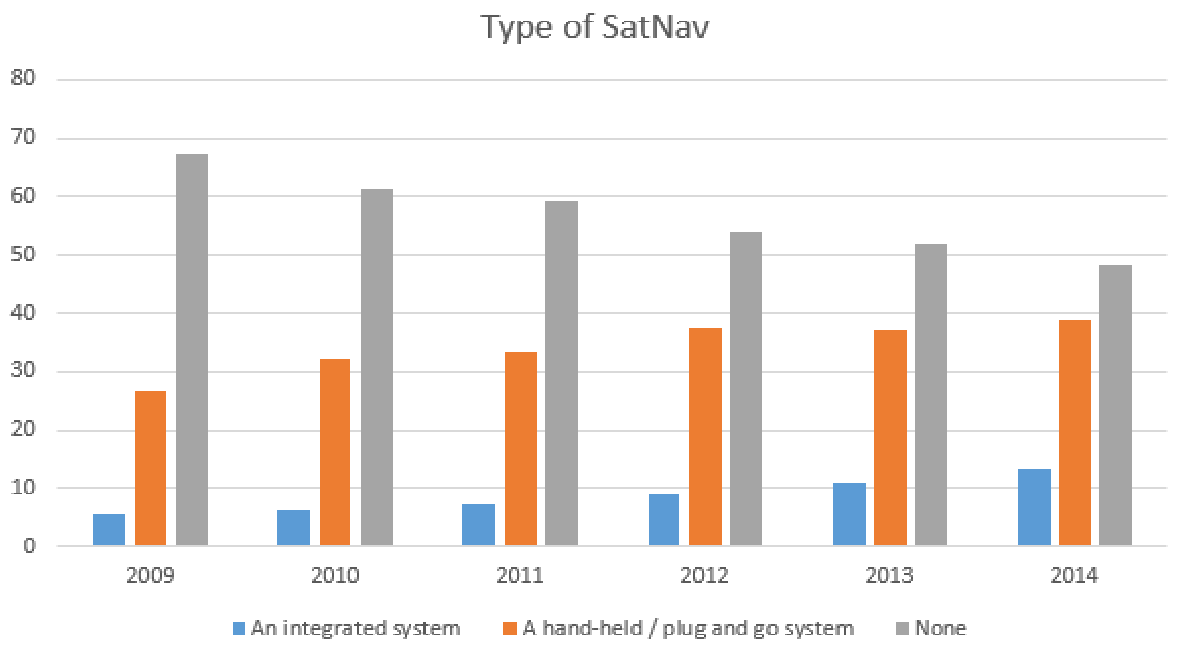


Figure 2: Trends in Sat Nav usage in the UK 2009-2014 (Department for Transport, 2015)

A common complaint posed by Sat Nav sceptics is that the visual element of the display can cause a distraction to a driver who may keep referring to the screen as they are driving along. Back in 1993, before satellite navigation would become commercially available in cars, studies were performed at the University of Michigan to test how much visual attention was commanded by the proposed systems during usage. The study found that having a visual element to the navigation system increased the reaction times of the participants and led to the drivers focussing less on the road ahead of them (Green, et al., 1993). These findings were corroborated in a paper by Jensen et al. who performed further studies into the distraction caused by visual, audio and audio-visual systems. The paper found that visual systems led to drivers taking their eyes off the road more often and causes a decrease in driving performance. It concludes by suggesting there be research into the design of an audio-based navigation system due to it being an *"adequate output modality in terms of road safety"* (Jensen, et al., 2010). Alternatively, work has been done to create prototype visual systems that attempt to lower the salient load of a driver with the aim of decreasing the amount of visual engagement required by the system, therefore increasing the driving performance of the user (Lee, et al., 2008).

Another issue surrounding the use of Sat Nav technology is that of the credibility of the instructions it gives to the driver. Although we live in a technologically advanced age, many people remain sceptical about certain systems, particularly ones that modify interactions they may have been doing for some time without digital guidance. Sat Nav systems are particularly affected by this, and this phenomenon is worsened if incorrect or erroneous instructions are given, potentially leading to a loss of credibility of the system in the eyes of the driver (Schaub, et al., 2013). It is important therefore that systems are designed to remain credible in the eyes of the driver.

A further problem with the current Sat Nav technology is their inability to allow drivers to build up a good knowledge of their surroundings as they go about their journey. Research into pedestrian navigation systems has found that using a turn-by-turn system fails to build up spatial knowledge of the surroundings by the user (Aslan, et al., 2006). Likewise, these findings can also be applied to turn-by-turn systems in vehicles, as an ethnographic study by Cornell University proves (Leshed, et al., 2008). This study finds that drivers with such a system often end up becoming disengaged from the environment they are in, instead becoming focussed on the virtual environment created by the Sat Nav. This can lead to adverse effects in the user of the system, something which is further discussed by Leshed. The paper argues that with turn-by-turn systems, *“no skilled engagement with the environment is needed anymore”* and that *“users can blindly follow the visual and vocal instructions”* to get to their location, a fact backed up by their research findings. It can therefore be argued that navigation and route-finding are skills that are disappearing due to this disengagement with the outside environment.

Alternative Sat Nav systems

To combat the issues that have been discovered about current Sat Nav systems, there have been a number of alternative systems proposed. While some have remained as theoretical ideas, many have also been created in practise in an attempt to improve various aspects of in-car navigation. This section explores a few of these ideas.

In a paper entitled The Normal, Natural Troubles of Driving with GPS (Brown & Laurier, 2012), Barry Brown and Eric Laurier conduct some ethnographic research by filming journeys in which drivers use various forms of in-car navigation technology. After highlighting five key areas in which Sat Nav users may run into trouble, Brown and Laurier highlight some theoretical ideas about where the designers of Sat Nav systems could explore to improve a driver's experience. In particular, they point out the need to give drivers more options when choosing their route and destination. They suggest presenting a secondary route on screen or labelling upcoming streets in terms of their likelihood of taking the driver to the chosen destination, therefore allowing the driver to make their own choice about which route is best for their needs. Implementing either of these ideas would afford the user more freedom in choosing their journey; although as previously mentioned, increasing the visual load of the Sat Nav screen could cause problems in terms of safety and driving performance (Jensen, et al., 2010).

Another option for exploration is proposed by Pfleging et al. who note that while many car manufacturers advertise the driving experience given by their vehicles, Sat Nav manufacturers often do not (Pfleging, et al., 2014). They argue that since the driving experience of a journey is most heavily affected by the route taken, Sat Nav designers should take this into account when writing route selection algorithms. They therefore propose a Sat Nav operating mode in which the route is selected based on factors relating to the driver's emotional response, in the hope that they will arrive at their destination *"more happily and less stressed"*. A web survey conducted for the paper found that the driving factors that are most important during a journey depend on the current situation, something that again points towards the need of giving drivers more choice about route selection.

A third option for improvement of the system involves the usage of audio feedback. As previously mentioned, Jensen et al. discuss the merits of creating an audio-based system due to it being less distracting than a visual-based one. While they set out a number of potential areas to explore, they didn't follow this up with the actual creation of a system. However, a paper by Komninos et al. goes one step further (Komninos, et al., 2012). In it they discuss the creation and testing of a purely audio-based navigation system designed to be used on foot by tourists in a city. As it is intended to be played through headphones, they have come up with the idea of creating an 'audio scent' which emanates from sound beacons spread across the route to the user's destination. As the tourist walks the audio appears to come from a particular direction in the headphone channel, and by following the direction of the sound they can navigate towards their destination. The system is designed so that a tourist is still free to explore the city they are in, affording them the option of planning their own route towards the destination without getting the feeling of being lost.

Leading on from this, work done by Robinson et al. spawned a similar prototype navigation system which relied instead on haptic feedback to guide a user on-foot to a destination (Robinson, et al., 2012). Their system used vibration from a smartphone as feedback when the phone was being pointed in the general direction of their destination. The user would follow the vibration to navigate the route without the need for maps or any kind of route planning. It was found that the system allowed navigation at normal walking pace with negligible modifications required to the user's behaviour.

After completing my state-of-the-art review, I refined my idea for improving a driver's user experience with a Sat Nav. I decided that the problem I would attempt to tackle with my system would be the problem proposed by Aslan et al. of drivers not developing spatial knowledge of their surroundings when using a turn-by-turn system. Furthermore, I agree with the points raised by both Leshed et al. and Aporta & Higgs (Aporta & Higgs, 2005) with regards to the de-skilling of drivers by such systems. I am therefore aiming to create a system that affords drivers the chance to develop basic navigational skills whilst remaining connected enough to technology to not worry about getting lost. Due to the findings put forward by Jensen et al., I decided that an audio-based system would give the driver the best opportunity to learn from and understand their surroundings in order to build up such skills. To implement the system, I decided to borrow ideas from the work of Komninos et al. primarily, to create a similar audio based system for in a vehicle. However due to the limitations of being in a car (i.e. not being able to wear headphones while driving), the system will need to be significantly modified, while keeping the idea of the audio 'emanating' from the destination.

Upon completion of the background research for my project, I decided to update my aim slightly. I decided to base it on a hypothesis I came up with while reading around the topic, which is that:

> *"The reliance on turn-by-turn in-car satellite navigation systems leads to a de-skilling of modern drivers"*.

I therefore modified my aim to be as follows:

> *"To create a working in-car navigation system using implicit instructions to attempt to combat the de-skilling of modern drivers"*

In-Car Testing

As part of the design and implementation process for my app, I knew I would need to complete some in-car testing, both to test the feasibility of it using a prototype and to test the success of the implementation using the actual app. Before I could begin planning my user testing, I researched some driver analysis methods that had been used in similar studies.

The major method of research that is prevalent when studying a driver's interaction with a Sat Nav system is ethnographic research. This is where a research team observes a user interacting with a system in their natural, real-world environment as opposed to in artificial conditions within a laboratory (gov.uk, 2016). By doing so, researchers are able to assess how the subject would use the system in their day-to-day life and from this are able to spot patterns of usage and can observe situations that may otherwise be missed. Ethnographic research is a very popular method of in-car testing as it allows the researcher to be in close proximity to the driver so that data such as driving performance and verbal comments are able to be collected during the journey, examples of which can be found in the work of both Jensen et al. and Brown & Laurier.

Leading on from this, a further form of testing involves the researcher taking a more hands-off approach to running the research phase. Rather than working closely with the subject, they allow them to go off and use the system being tested without immediate guidance from anybody associated with the trial. An example of this can be seen in the testing of the SoNav system by Komninos et al. where participants are encouraged to navigate across a city on their own with only the system for guidance. By doing this, the researchers can get information about how the system works when they are not close to hand to offer advice, in a sense it offers a more realistic view of how the system would work in the real world. A similar method was used by Robinson et al. when testing their haptic system with participants travelling a route across a university campus.

The papers by Jensen et al. and Brown & Laurier also bring up an important point in that the medium in which data is collected also plays an important role in the effectiveness of the outcomes of the study. Both papers contain images taken from video recordings of the trial being conducted, the footage of which is later analysed to form parts of the conclusion of the work. An alternative method of data collection is used by Forlizzi et al. in their paper on navigation strategies, in which they observe how groups of individuals use navigation skills while in a vehicle (Forlizzi, et al., 2010). To record the outcome of their study they record audio data of the conversations between participants while travelling. The advantage of this is that less extraneous data is collected, however audio recording could prove difficult in such a loud environment. Having a version of the research in a digital medium is important as large amounts of data can be collected in a short amount of time and allow for a more objective view of the research findings than by using field notes alone (Knoblauch, 2005).

Whereas the ethnographic research I have looked at focusses on the way the driver interacts with the system being studied, other forms of research forgo the test subject all together and simply test the system itself. An example of this can be seen in the work of Schaub et al. who take various current Sat Nav systems and test them personally to assess credibility concerns during interaction (Schaub, et al., 2013). By doing this more immediate form of research, we can modify the method of testing to focus on any areas in which the need for closer assessment may become more apparent while the study is being conducted. This method is often more useful when researching the state-of-the-art before beginning a development phase or when testing prototypes of a system made with Rapid Application Development.

The methods looked at so far all deal with qualitative data, that is research that explores the problem space and looks to provide unmeasurable insights into the usage of the system (Wyse, 2011). However, it is also important to collect quantitative data too in order to back up any conclusions drawn from the research with statistical data. One way that quantitative data can be collected is through the use of surveys. By studying responses from surveys, researchers are able to collect information from a large range of people in a relatively short space of time in order to generate data that can be used in the development of a system or as part of a systems evaluation process. An example of the former can be seen in the work of Pfleging et al. who issued a web survey in order to assess what Sat Nav users consider the most important factors during route selection. From the results they found that drivers often looked for routes that were less stressful and with less traffic, results which allowed them to consider the viability of their experience-based Sat Nav proposal. Similarly, Komninos et al. used the results from their survey of test participants to prove that the system they had created reached its goal of facilitating exploration in tourists while using their navigation system.

Implementation

Initial User Testing

Design

Before I could begin designing and implementing my solution, I needed to gather some information about how a user may respond to the system. As it is a system that is unlike many others on the market currently, I needed to ensure that the idea would be feasible to use in a real life scenario and that it would satisfy the goals it sets out to meet in practise, not just in theory. I therefore decided to conduct a round of initial user testing using a prototype system. Doing this would not only help validate the feasibility of the idea but would also provide me with some feedback as to how the final solution should look and act.

To create the prototype, I decided to use the Gadgeteer system, which is a prototyping platform that is programmed in C#. It takes a number of external modules connected to a central mainboard and uses them to generate various types of feedback to the user. My crude system involved a button module and the tunes module (a small buzzer) connected to my laptop via a USB cable. When the button is pressed, the tunes module plays a note of frequency 2000 hertz for 300 milliseconds. This imitates the audio feedback from the proposed solution. The code for the prototype can be seen in Appendix item 1.

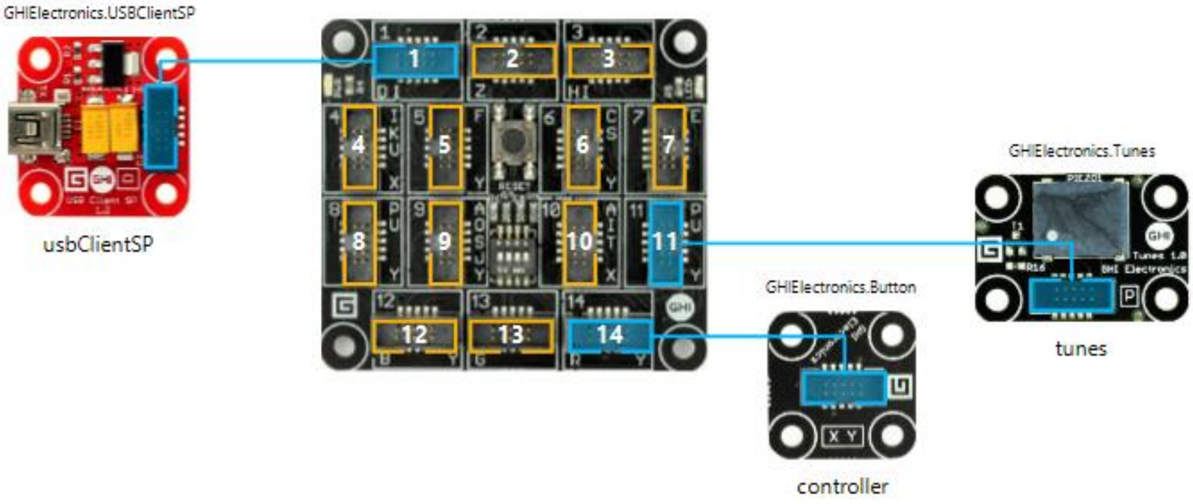


Figure 3: Technical design of the Gadgeteer prototype

To assess the potential benefits and shortfalls of the solution, I decided to perform a test run with a volunteer using the prototype system. The volunteer would be asked to drive around as though they were using the final system to navigate to different places. During the experiment I sat in the passenger seat running the prototype system from my laptop, and would be following our progress on a map from my phone. Using this I could press the button on the prototype according to our progress and distance from the destination. Once the driving portion was finished, the volunteer took part in a short interview where they could share their feelings on the experience. They were allowed to use their own vehicle for the test so that they would be in a comfortable environment, and were allowed to end the experiment at any time.

To begin, I got the volunteer to drive a short distance to a location that they already knew the route to and had been to before, this was to give them some experience in how the system would act and sound. From this location, I got the driver to travel to a second location, this time they wouldn’t know exactly where it was or how to get there. Before we set off, I encouraged the driver to plan their route on a map and suggested that they used waypoints and road signs to assist them in finding the location. Once we had travelled to the second location, I asked the driver to take us to another unknown location using the same technique, before going back home to conduct the interview. In total, I managed to perform tests with three different volunteers so that I could get a range of feedback about the system along with a better idea of ways to improve the final system. Two of the volunteers were male and one was female; the age range of the group was 20-51. The routes taken by each volunteer can be seen in Appendix item 2.

Results

From performing the tests with real end-users, I learned a lot about how the proposed system would work once it had been developed. It was firstly notable that the majority of test journeys ran smoothly, with the system seeming to be useful in guiding the driver towards the destination, indeed volunteer 1 said that the system "*sort of helped me work out how far along the road I would be going".* Each volunteer expressed satisfaction with how the system helped them figure out how far from the destination they were, with volunteer 2 adding that it "*helped me know I was still on route*" and volunteer 3 saying that it "*showed me that I was going in the right direction*". This hopefully means that the idea itself is sound and that creating a fully-functional version is a worthwhile thing to do.

An observation I made from the passenger seat was that each volunteer used signposts to guide them to their destination while using the system, with volunteer 3 in particular commenting on there being a signpost for one of the destinations that they subsequently followed. Furthermore, it seemed to me that when the beeping of the prototype increased in frequency as the driver got closer to the destination, they became more alert to their surroundings, with the prototype system being used to support their navigation rather than being used to explicitly guide them. For example, volunteer 2 said that at that point they knew to "*start looking for signs and stuff to find [the destination]*", which will hopefully mean they will have a better understanding of the environment and landmarks surrounding the destination if they were to navigate there again. From this, I can ascertain that the app I will create should realise its aim of improving a driver's navigation skills.

The tests did not always run smoothly though, and I gave an opportunity in the interview for the volunteers to express any difficulties or frustrations that they encountered during any of the journeys. While most of the drivers navigated to the destination with minimal difficulty (the journeys undertaken were relatively short), volunteer 3 managed to take a wrong turning on their final journey which led to them travelling away from the destination. Recognising this, I slowed down the frequency of feedback from the prototype system, something the volunteer noted: "*the beeping slowed down enough to show me that I wasn't making progress*". The driver was again able to use their own navigation skills to follow "*signs for Willaston*" as I had hoped they would. However, this showed me that the final system must have some way to quickly notify the driver when they have taken a wrong turning, before they travel a great distance away from the destination. This sentiment was echoed by volunteers 1 (who requested a signal "*to indicate when you're going further away*") and 2 (who wanted "*some way of showing that you're going off route*"). This may not be so necessary on a short journey such as the ones undertaken in the testing, as the short distances would likely lead to an obvious decrease in feedback frequency when a wrong turning is taken. However, on a longer journey, it may take a greater amount of time for the driver to notice that the frequency of beeping has decreased to indicate a wrong turning, especially if the mistake is made early on in the journey.

During the interviews, I allowed the volunteers to propose any ideas of concepts that they would like to see in the final design. A good idea that was proposed by volunteer 3 was the concept of a mute button on the app interface for when the driver gets to a point in the journey where they know exactly how to get to the destination. At this point the audio feedback from the system could become annoying, as it seemed to for each volunteer when they were navigating to the destination they already knew. At this point the driver said they "*knew I wasn't going to get lost at that point*", so didn't need the extra distraction of the system. A large button on the interface of the app that can be pressed to disengage and reengage the audio would be helpful so that the driver does not have to shut off the whole app and could still get the audio track back on if they were to need it again.

Aside from this, there were other bits of feedback from the volunteers that I decided not to utilise in the final design. For example, volunteer 2 found driving around an unfamiliar area "*a bit scary*" and suggested the option of engaging turn-by-turn commands in a situation such as that. However, I decided not to take this idea on board for fear that my final solution may end up seeming like a poor quality version of a regular satellite navigation system. Instead I decided to distance myself from the idea of a turn-by-turn based system entirely, albeit acknowledging that a successful version of my system could be incorporated as an optional mode on a regular SatNav. Another complaint was from volunteer 1, who claimed that they had difficulty "*figuring out when to take turnings*". I decided that since the aim of my system is to boost a driver's navigational skills, this complaint relates to a skill that should end up being developed with use of the system, functional changes should therefore not need to be made to compensate for this.

The full transcripts for the initial user interviews can be found in Appendix item 3.

Requirements Elicitation

From the feedback gained from the initial user testing coupled with the ideas generated from my background reading, I was able to develop a list of requirements that the final solution to the problem would need to fulfil.

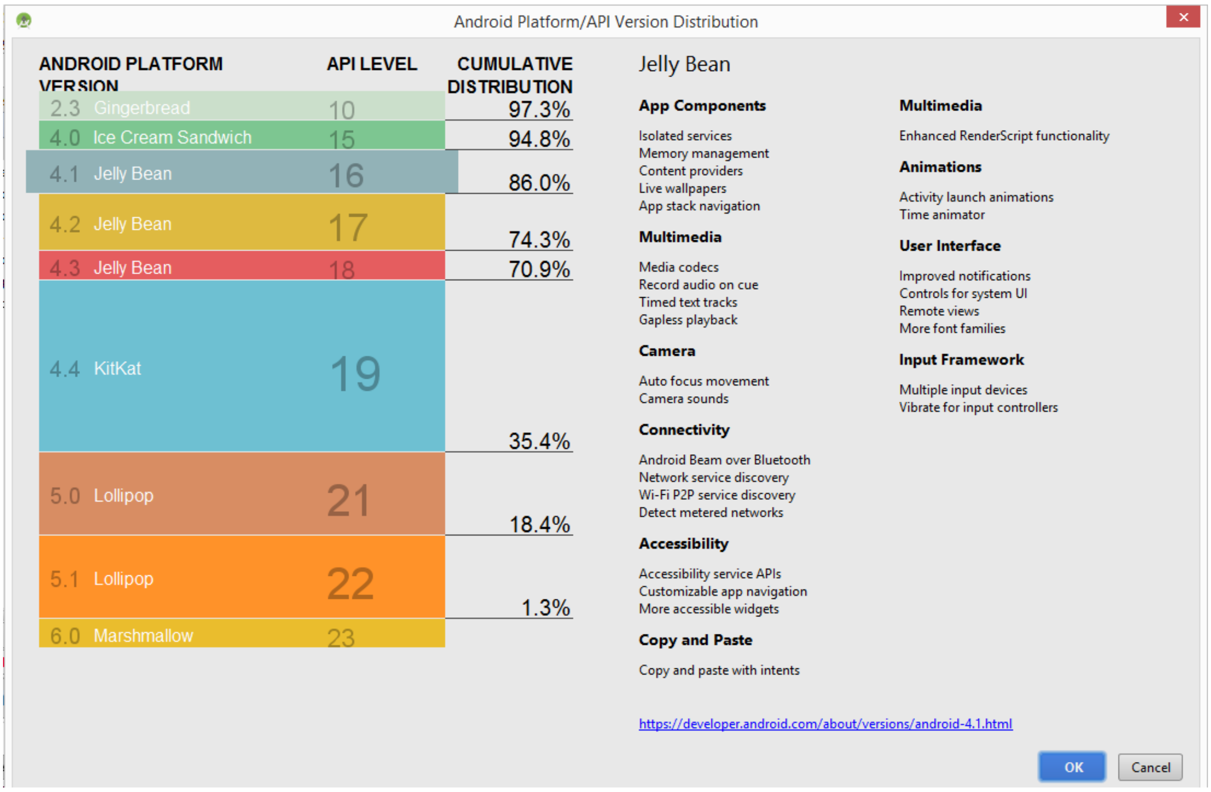
The full list of requirements can be found in Appendix item 4.

Tools

In order to create the functionality of the app, there were a number of decisions that needed to be made regarding the development tools used.

To begin with, I had to decide on a platform to develop my app on, the three main choices being Android, iOS or Windows. In the end I chose to develop on the Android platform, this decision was based on a number of factors. First and foremost, I currently own an Android phone (Samsung Galaxy A3), this therefore meant that when it comes to testing any version of my app I would be able to do it on my own phone rather than having to go through an emulator, which is often slower and more difficult to use. This also meant that I could use my phone as if I were a real user of the system when testing it in-car.

On top of this, I have programmed basic Android applications in previous modules taken at University, I therefore know some of the basics of developing for this platform which I can base my current work upon. Programming Android applications is also done using the Java programming language, which is one I am particularly familiar with, having 3 years of experience already in using it. Comparatively, iOS development involves the use of the Swift language while Windows development requires knowledge of either C# or Visual Basic, none of which I have any particular experience in. The focus of this dissertation has been decided to be in testing the concept of the application as opposed to it being a programming exercise. Therefore, although I feel that the general programming knowledge I currently have would help me in learning one of these languages, sticking with a language I already know seemed like a better idea, particularly given the relatively short time frame of the project.



ANDROID PLATFORM VERSION		API LEVEL	CUMULATIVE DISTRIBUTION
2.3	Gingerbread	10	97.3%
4.0	Ice Cream Sandwich	15	94.8%
4.1	Jelly Bean	16	86.0%
4.2	Jelly Bean	17	74.3%
4.3	Jelly Bean	18	70.9%
4.4	KitKat	19	35.4%
5.0	Lollipop	21	18.4%
5.1	Lollipop	22	1.3%
6.0	Marshmallow	23	

Jelly Bean

App Components

Isolated services
Memory management
Content providers
Live wallpapers
App stack navigation

Multimedia

Media codecs
Record audio on cue
Timed text tracks
Gapless playback

Camera

Auto focus movement
Camera sounds

Connectivity

Android Beam over Bluetooth
Network service discovery
Wi-Fi P2P service discovery
Detect metered networks

Accessibility

Accessibility service APIs
Customizable app navigation
More accessible widgets

Copy and Paste

Copy and paste with intents

Multimedia

Enhanced RenderScript functionality

Animations

Activity launch animations
Time animator

User Interface

Improved notifications
Controls for system UI
Remote views
More font families

Input Framework

Multiple input devices
Vibrate for input controllers

https://developer.android.com/about/versions/android-4.1.html

Figure 4: Android SDK information

When I began creating the app, I had to select the SDK level that it would be built to be compatible with, by setting values in the Gradle build file. I first had to decide the minimum system API level that would be built to, below which the app would be unable to be installed. As the requirement NFR2.1 states that I should be targeting as wide a range of devices as possible, I chose API 16 (Jelly Bean 4.1) to target the approximately 94.8% of Android devices that run to this level. I also had to select the target SDK version which states the API level that the app will be tested to in order to maintain compatibility. As the phone that I will be testing the app on runs at API level 21 (Lollipop 5.0), this is the level I chose to target.

The final considerations that needed to be made were with regards the actual functionality of the final app. I knew that in order to convert the users input into a coordinate format that could be used for route-finding I would need to use a geocoder. The two approaches to using a geocoder are to do it client-side or server-side. Client-side uses the inbuilt geocoder provided by the Android Location Services while server-side requires a call to the Google Maps Geocoding API and parsing of the JSON response. As I am geocoding from a user input as opposed to a batch dataset I have decided to use the built-in client-side geocoder.

To convert the coordinates generated by the geocoder into data that can be used for the audio feedback, there were a number of possible options I could have used. Google Maps has its own API for Android that comes built in with Android Studio which could be used for calculating routes between the current location and the destination (Google Developers, 2016). Likewise, there are a number of different routing services that have been developed on top of the OpenStreetMap project (wiki.openstreetmap.org, 2016), in particular the Skobbler app which has an open-source SDK for use on mobile devices. However, my app doesn’t need a visual representation of the quickest route between points, instead it just needs the distance of the quickest route. Therefore, a lot of these implementations have too many extra unnecessary features for what I need. I decided instead to use the Google Maps Distance Matrix API which simply gives a distance value based on the optimal route between two points (Google Developers, 2016).

Design

The first part of the design phase involved planning the layout of the app. As I am primarily focussed on testing the concept I decided not to make the app too complex in terms of extra features, instead I decided to only implement the essential parts to make the concept work. I therefore decided against having any kind of back-end database as no data needs to be stored from the app at this time. Parts of the computation required for the functionality involves calls to external resources, namely the geocoder, Distance Matrix API and Location Services, this is all handled within the app itself. In terms of the layout, I decided on a simple design of just two pages. The first page, 'Destination' is where the user enters their desired destination and selects their choice from a list provided by the geocoder. The second page is the 'Journey' page where this choice is set as the destination and audio feedback guides them towards this location. The frequency of this audio feedback is determined by the distance from the current location, a figure that is calculated by calls to external resources.

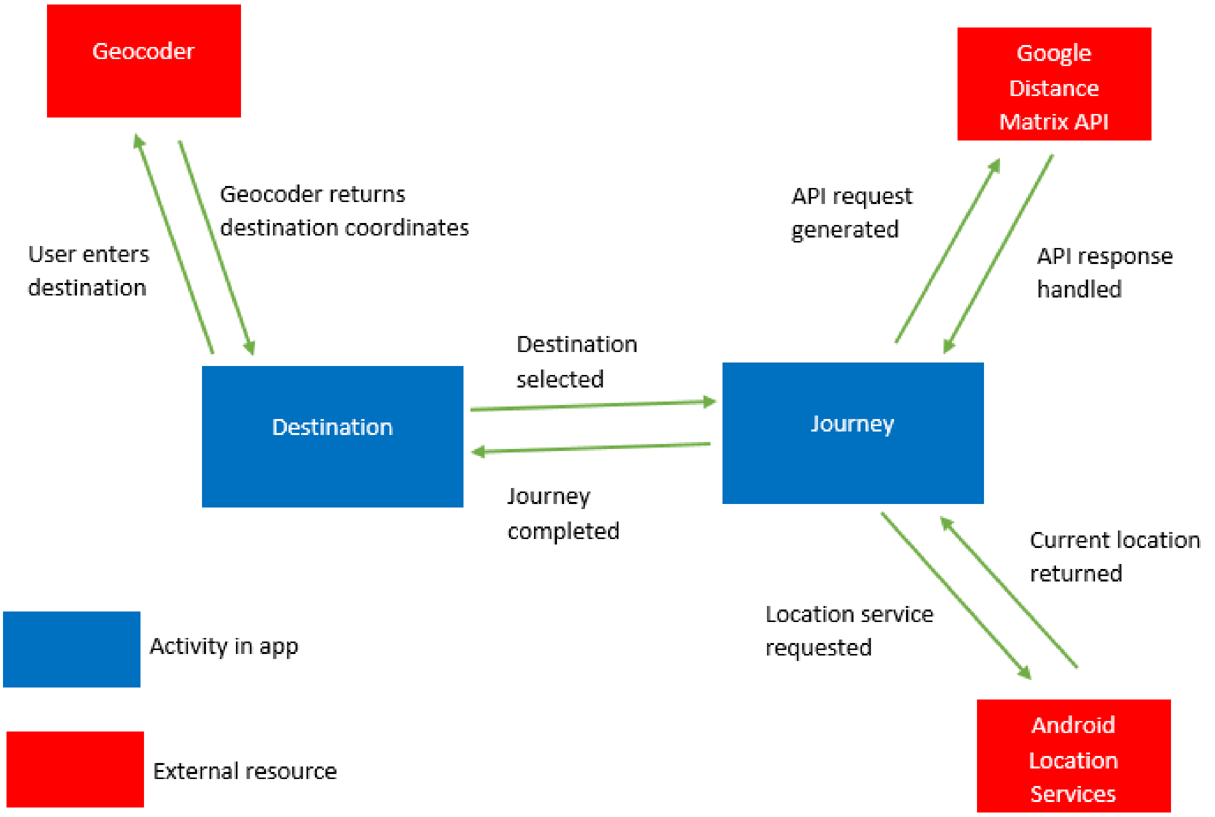


Figure 5: Design map for application

On the Destination page, I planned on having a text field for the user to enter their desired destination in, which would then return a set of possible results from the geocoder. When the desired result was selected from the list below, its coordinated would be sent to the Journey page. The Destination page would also have a notification system if the user's mobile data, location or media volume were turned off, so that the system would work when they are driving.

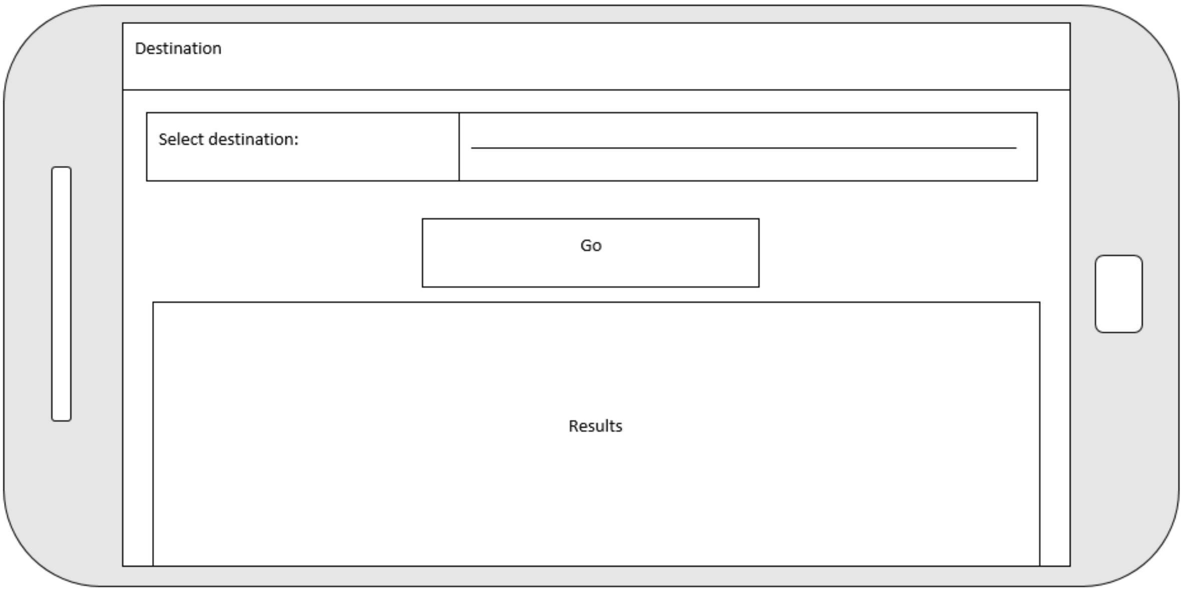


Figure 6: Destination page design

The Journey page would display the current destination that has been set by the user but no information about how to get to it or the distance left on the journey. There would be an audio track playing a ‘beep’ sound that would change speed based on the distance left to the destination, along with a button to mute or unmute the track. When the current calculated location is within a close proximity to the destination location, the audio track will cease and a notification will alert the user that they have arrived at their destination.

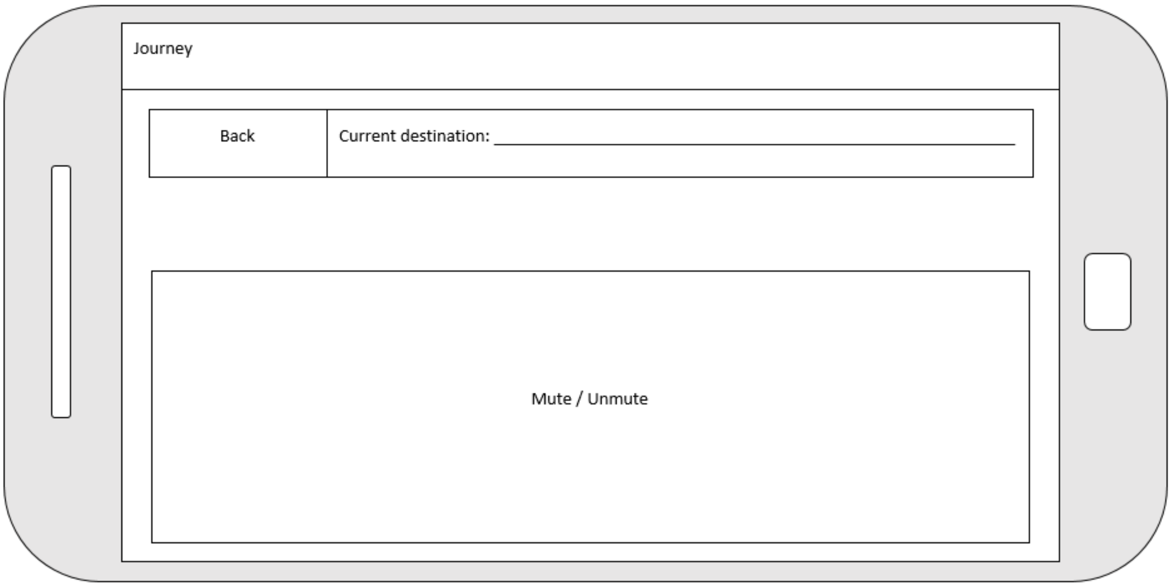


Figure 7: Journey page design

The audio track was split into two parts. When the destination is more than 1km away, the beeps will be 10 seconds apart and the audio track will pause if no progress is being made. This is for two reasons. Firstly, if the user is sat at a junction or in a traffic jam, they will not want the system to continue to beep at them as it could get frustrating. Secondly, if the user takes a wrong turn and starts heading away from their destination, the audio feedback will stop and they will realise that they are travelling in the wrong direction, satisfying requirement FR1.4.

When the destination is approaching (i.e. less than 1km away), the driver would want more responsive feedback. I therefore proposed a system where there will be 5 separate levels of feedback speed, meaning that the audio speeds up for every 1/5 of a mile closer to the destination that the driver is. The delay between beeps in this section will be 5 seconds, 2.5 seconds, 1 second, 0.5 seconds and 0.25 seconds. All timing and distance variables within the app will need to be tweaked during testing to ensure they are appropriate in a real-world scenario.

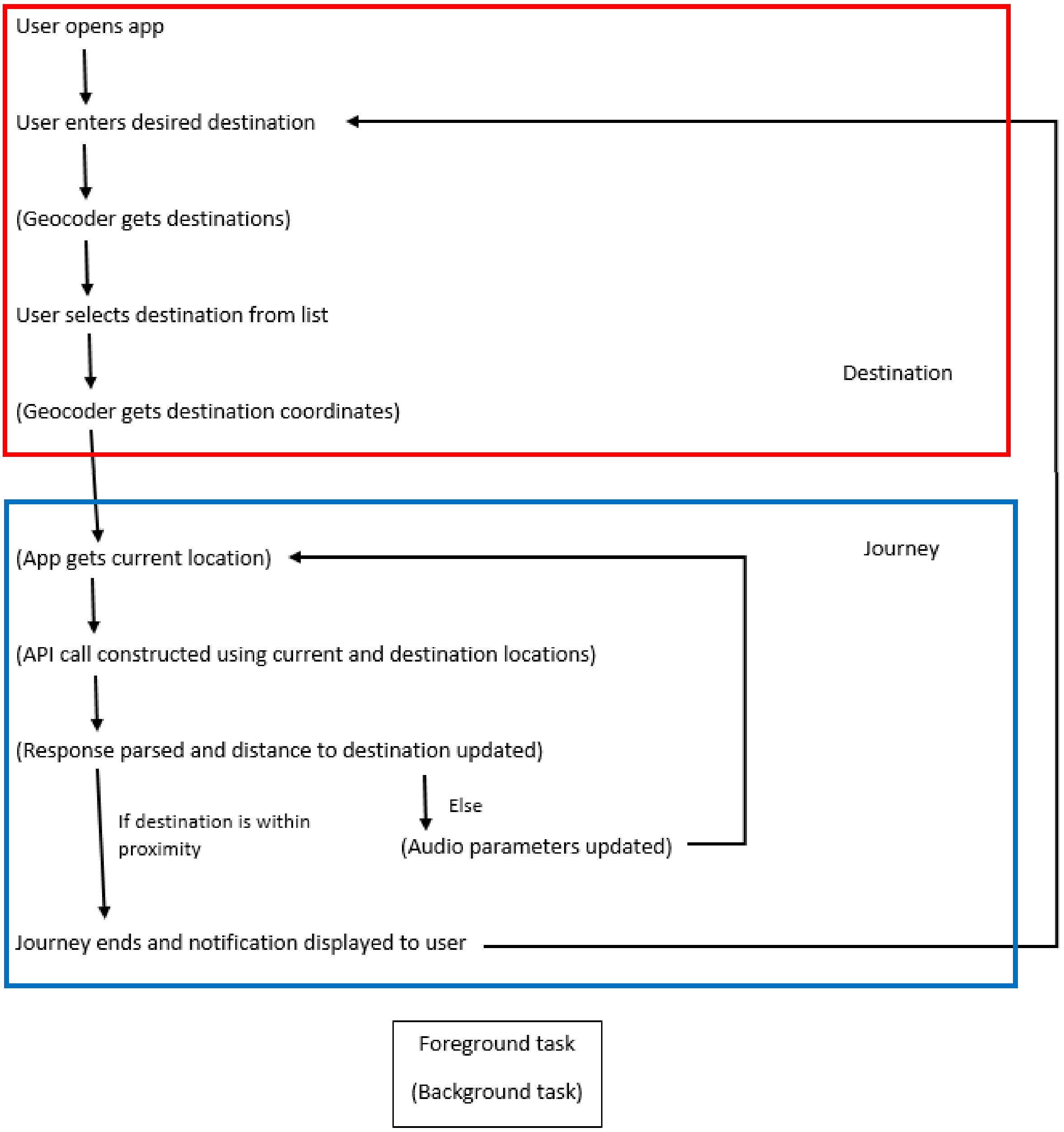

User opens app
User enters desired destination
(Geocoder gets destinations)
User selects destination from list
(Geocoder gets destination coordinates)
Destination
Journey
(App gets current location)
(API call constructed using current and destination locations)
(Response parsed and distance to destination updated)
If destination is within proximity
Else
(Audio parameters updated)
Journey ends and notification displayed to user
Foreground task
(Background task)

Figure 8: App process diagram

Figure 8 contains the process flow diagram for the app which shows all the sub-tasks required in order for a user to navigate a particular journey. From this, I was able to break the implementation down into the following seven sections to be implemented:

1. Layout, Activities & Intents
   This stage will ensure the satisfying of requirements FR2.2 and NFR1.1

2. Geolocationing of destination
   This stage will ensure the satisfying of requirements FR1.1, FR1.2 and FR2.1

3. Calculating current location
   This stage will ensure the satisfying of requirement NFR2.6

4. Refreshment of distance on location update
   This stage will ensure the satisfying of requirement FR1.6

5. Creating and handling distance call to API
   This stage will ensure the satisfying of requirements NFR2.2 and NFR2.3

6. Generation of audio feedback
   This stage will ensure the satisfying of requirements FR1.3, FR1.4, FR1.7 and FR2.4

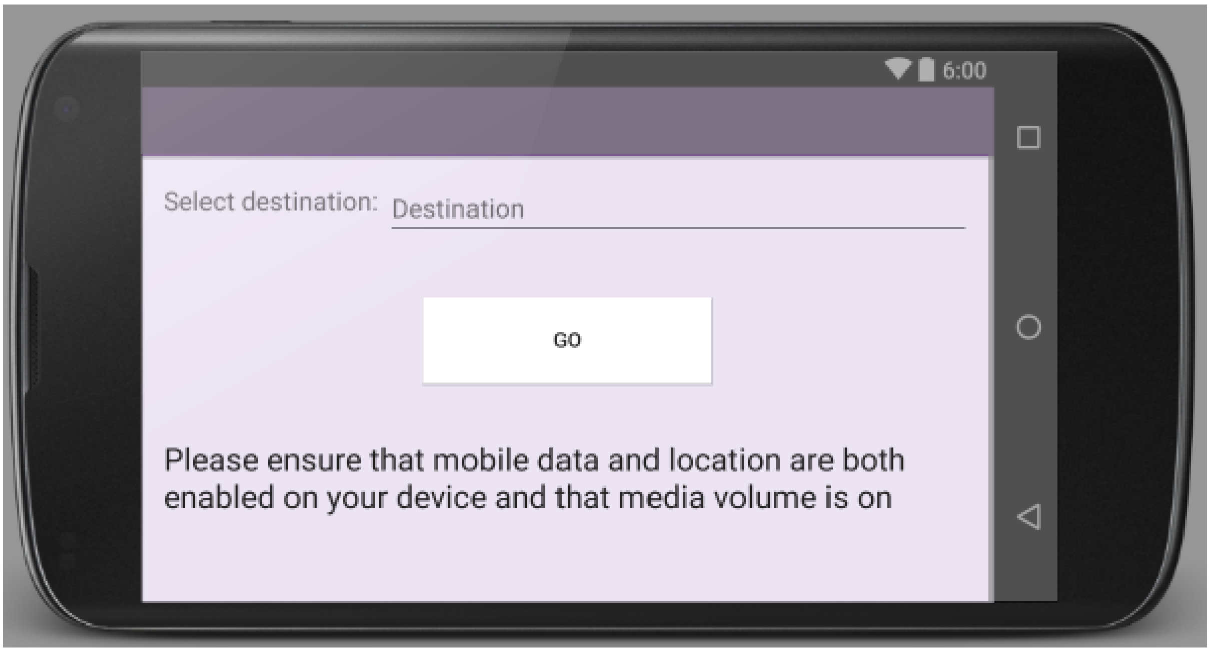
7. Programming of extra features (i.e. journey end, mute button, notifications etc.)
   This stage will ensure the satisfying of requirements FR1.5 and FR2.3

All other requirements should be satisfied by the overall completion of the application.

Implementation

1. Layout, Activities & Intents

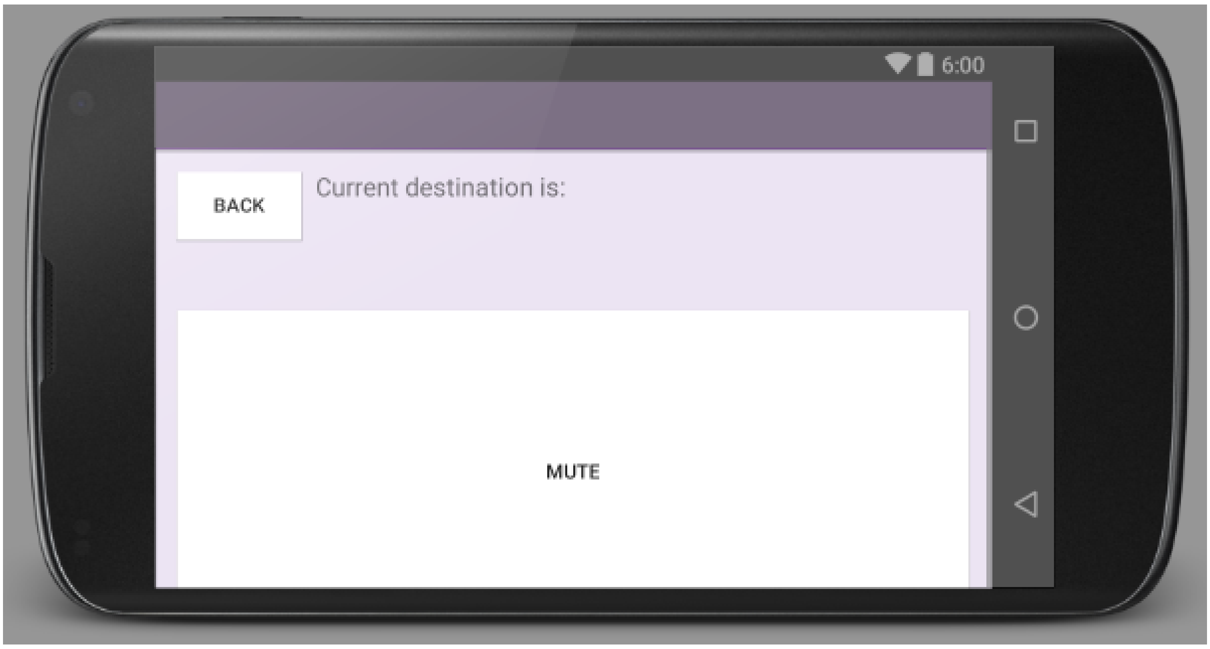
For this first section, I created the Destination and Journey activities to be the base for which all the functionality and layout of the app would be built on top of. They were laid out to follow the design storyboard and given a purple colour scheme. Intents were then created for navigation between the two pages. Full screenshots of the final implementation can be found in Appendix item 5.



Select destination: Destination

Please ensure that mobile data and location are both enabled on your device and that media volume is on

Figure 9: The final Destination page

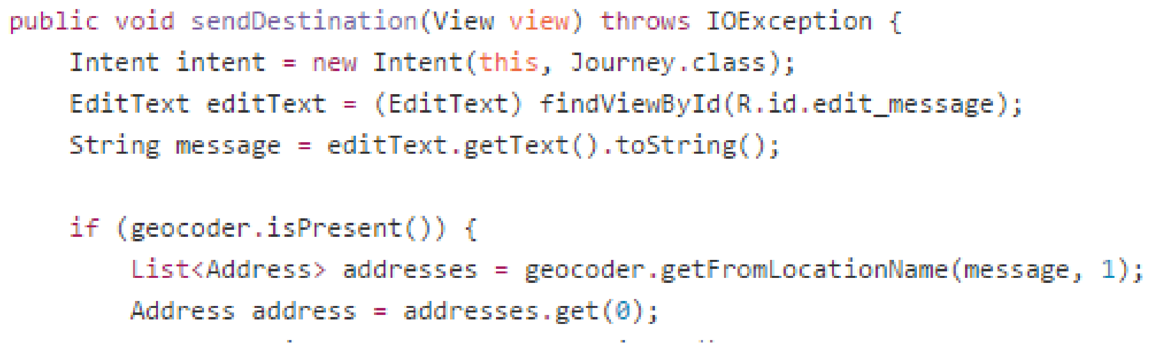


Current destination is:

Figure 10: The final Journey page

2. Geolocationing of destination

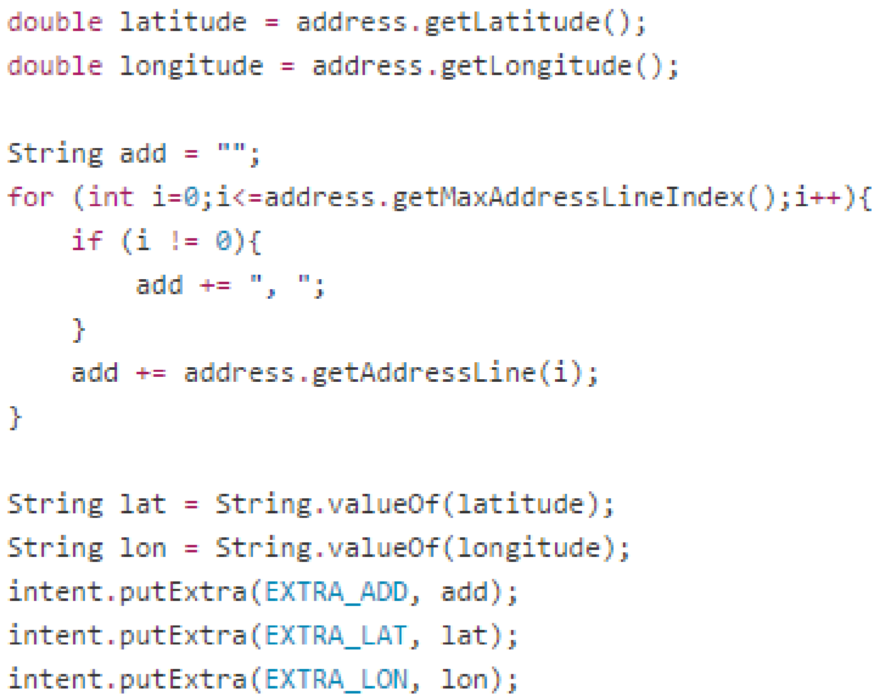
This task required the use of the Android in-build geocoder as previously discussed. It was created and used in the Destination page to convert the users desired destination input into a set of coordinates. After calling the geocoder using this string, the best matching result is returned from the location database. Initially I intended to return a list of possible results and allow the user to select which they meant but due to time constraints this feature was not implemented. I instead deemed it acceptable for just the best match to be selected, noting that since the full address of the returned result will be displayed on the Journey page, the user will be able to go back and enter a more specific search term should an erroneous result be returned.



```java
public void sendDestination(View view) throws IOException {
    Intent intent = new Intent(this, Journey.class);
    EditText editText = (EditText) findViewById(R.id.edit_message);
    String message = editText.getText().toString();

    if (geocoder.isPresent()) {
        List<Address> addresses = geocoder.getFromLocationName(message, 1);
        Address address = addresses.get(0);
```

Figure 11: Code for geolocating the destination

Once the geocoding is complete, the system will contain the full address of the desired destination along with its coordinates, which can then be used to get the remaining distance on the journey.



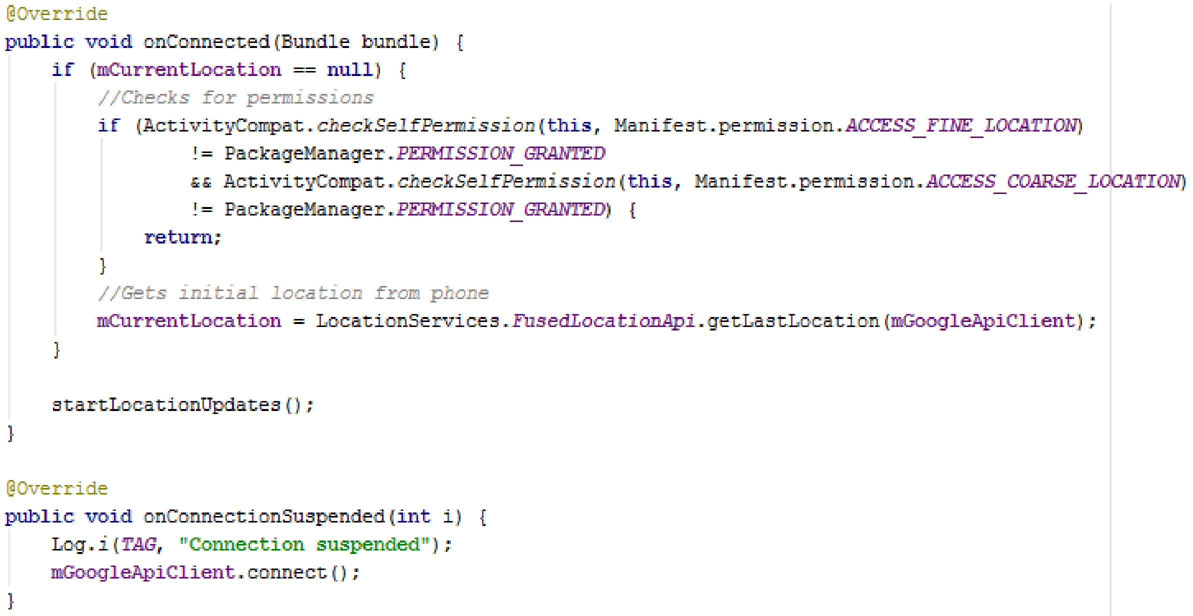
```java
double latitude = address.getLatitude();
double longitude = address.getLongitude();

String add = "";
for (int i=0;i<=address.getMaxAddressLineIndex();i++){
    if (i != 0){
        add += ", ";
    }
    add += address.getAddressLine(i);
}

String lat = String.valueOf(latitude);
String lon = String.valueOf(longitude);
intent.putExtra(EXTRA_ADD, add);
intent.putExtra(EXTRA_LAT, lat);
intent.putExtra(EXTRA_LON, lon);
```

Figure 12: Code for generating the full destination address

3. Calculating current location

After getting one of the coordinate sets needed to calculate the journey distance, the app needed to get the other set from the current location of the system, using the Google Location Services API. The API is more powerful than Android's built in location services and *"automates tasks such as location provider choice and power management"* (Android Developers, n.d.).

After setting the application dependencies and the API client, I created methods to handle connection and disconnection with the client. Once the Location Services were connected to, getting the current location of the device was simply a matter of calling the inbuilt method getLastLocation to inform the app of its last known location.

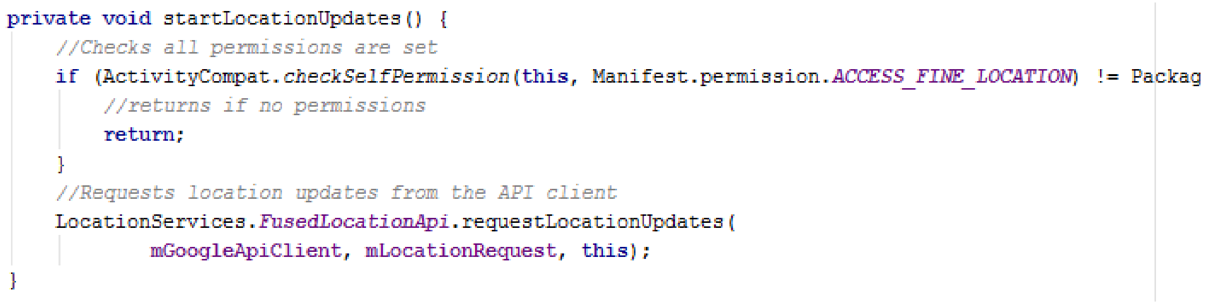


```
@Override
public void onConnected(Bundle bundle) {
    if (mCurrentLocation == null) {
        //Checks for permissions
        if (ActivityCompat.checkSelfPermission(this, Manifest.permission.ACCESS_FINE_LOCATION)
                != PackageManager.PERMISSION_GRANTED
                && ActivityCompat.checkSelfPermission(this, Manifest.permission.ACCESS_COARSE_LOCATION)
                != PackageManager.PERMISSION_GRANTED) {
            return;
        }
        //Gets initial location from phone
        mCurrentLocation = LocationServices.FusedLocationApi.getLastLocation(mGoogleApiClient);
    }

    startLocationUpdates();
}

@Override
public void onConnectionSuspended(int i) {
    Log.i(TAG, "Connection suspended");
    mGoogleApiClient.connect();
}
```

Figure 13: Code for finding the current location of the device

4. Refreshment of distance on location update

In order for the calculated distance to affect audio feedback frequency, it needed to be refreshed every time the user’s location changes. To do this, I first created a Location Request with a specified time set between updates. This variable was one that needed to be tweaked during testing as having too frequent updates drains the battery of the device while too infrequent means that the audio feedback will be less responsive to changes in the distance remaining. I could then request location updates from the API with the current location and distance remaining being updated each time.



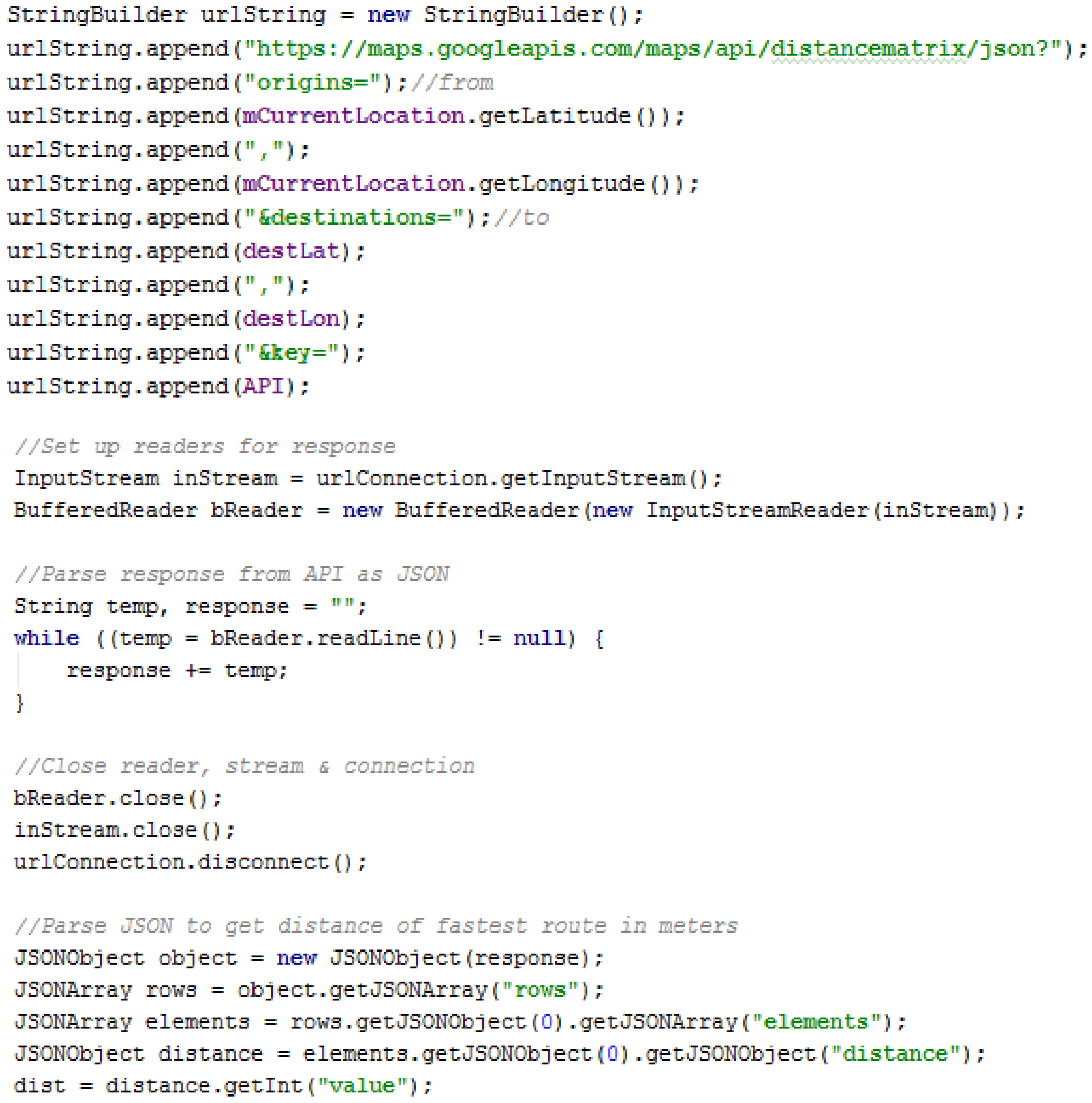
```
private void startLocationUpdates() {
    //Checks all permissions are set
    if (ActivityCompat.checkSelfPermission(this, Manifest.permission.ACCESS_FINE_LOCATION) != Packag
        //returns if no permissions
        return;
    }
    //Requests location updates from the API client
    LocationServices.FusedLocationApi.requestLocationUpdates(
            mGoogleApiClient, mLocationRequest, this);
}
```

Figure 14: Code for refreshing the location of the device

At this point, Version 0.1 of the application was complete and could be tested.

5. Creating and handling distance call to API

Now that I had the current location of the user and the location of the required destination, I built the request to be sent to the Google Distance Matrix API in order to get the current distance from the destination. After inputting the previously generated coordinates and API key into the URL, the request was sent using a GET method. The response came in the JSON format and was then parsed in order to get the distance result generated by the API, which could then be stored for use by the audio track once it had been implemented. Due to the intensiveness of the distance call and response handling, this entire section was done separately to the main UI thread so that all calculations could be performed in the background.



```java
StringBuilder urlString = new StringBuilder();
urlString.append("https://maps.googleapis.com/maps/api/distancematrix/json?");
urlString.append("origins=");//from
urlString.append(mCurrentLocation.getLatitude());
urlString.append(",");
urlString.append(mCurrentLocation.getLongitude());
urlString.append("&destinations=");//to
urlString.append(destLat);
urlString.append(",");
urlString.append(destLon);
urlString.append("&key=");
urlString.append(API);
```

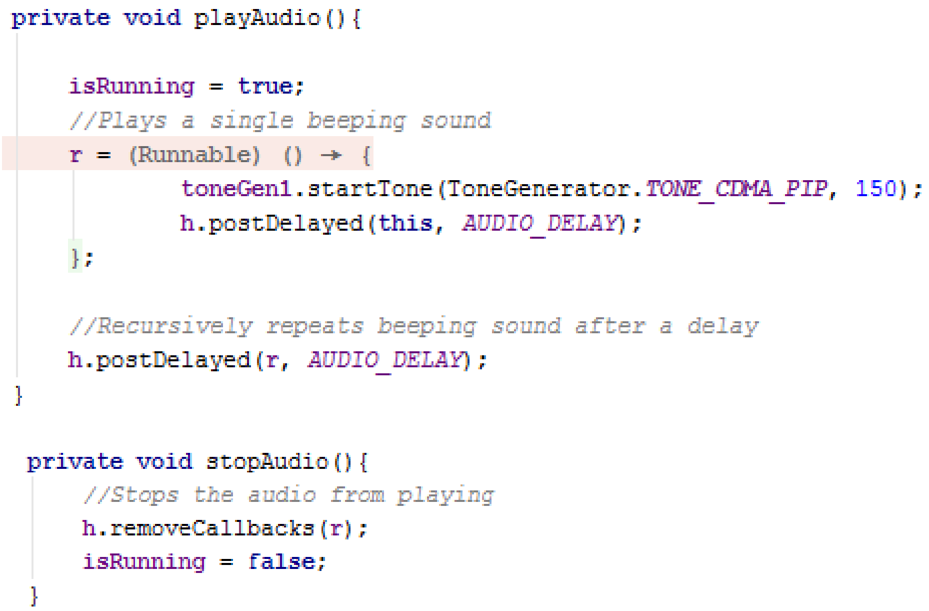
```java
//Set up readers for response
InputStream inStream = urlConnection.getInputStream();
BufferedReader bReader = new BufferedReader(new InputStreamReader(inStream));

//Parse response from API as JSON
String temp, response = "";
while ((temp = bReader.readLine()) != null) {
    response += temp;
}

//Close reader, stream & connection
bReader.close();
inStream.close();
urlConnection.disconnect();

//Parse JSON to get distance of fastest route in meters
JSONObject object = new JSONObject(response);
JSONArray rows = object.getJSONArray("rows");
JSONArray elements = rows.getJSONObject(0).getJSONArray("elements");
JSONObject distance = elements.getJSONObject(0).getJSONObject("distance");
dist = distance.getInt("value");
```

Figure 15: Code for creating and handling the API request/response

At this point, Version 0.2 of the app was completed and tested.

6. Generation of audio feedback

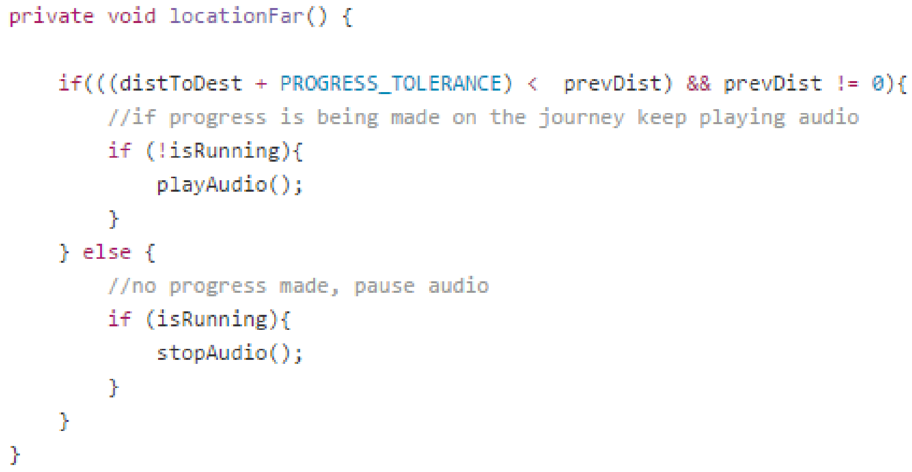
In order to create the audio that would be played to guide the user, I decided to use Android’s built in ToneGenerator, a decision partly because of the inbuilt ‘TONE_CDMA_PIP’ sound that it has. I chose the tone due to its unobtrusiveness and similarity to sounds generated by other systems such as active sonar and the Geiger counter which my system is based on. The audio track was created using a recursively called Runnable which repeatedly plays the pip followed by a pause. The callbacks for this particular runnable could then be removed at any point to stop the audio when required.



```
private void playAudio(){

    isRunning = true;
    //Plays a single beeping sound
    r = (Runnable) () -> {
            toneGen1.startTone(ToneGenerator.TONE_CDMA_PIP, 150);
            h.postDelayed(this, AUDIO_DELAY);
    };

    //Recursively repeats beeping sound after a delay
    h.postDelayed(r, AUDIO_DELAY);
}

private void stopAudio(){
    //Stops the audio from playing
    h.removeCallbacks(r);
    isRunning = false;
}
```

Figure 16: Code for handling audio

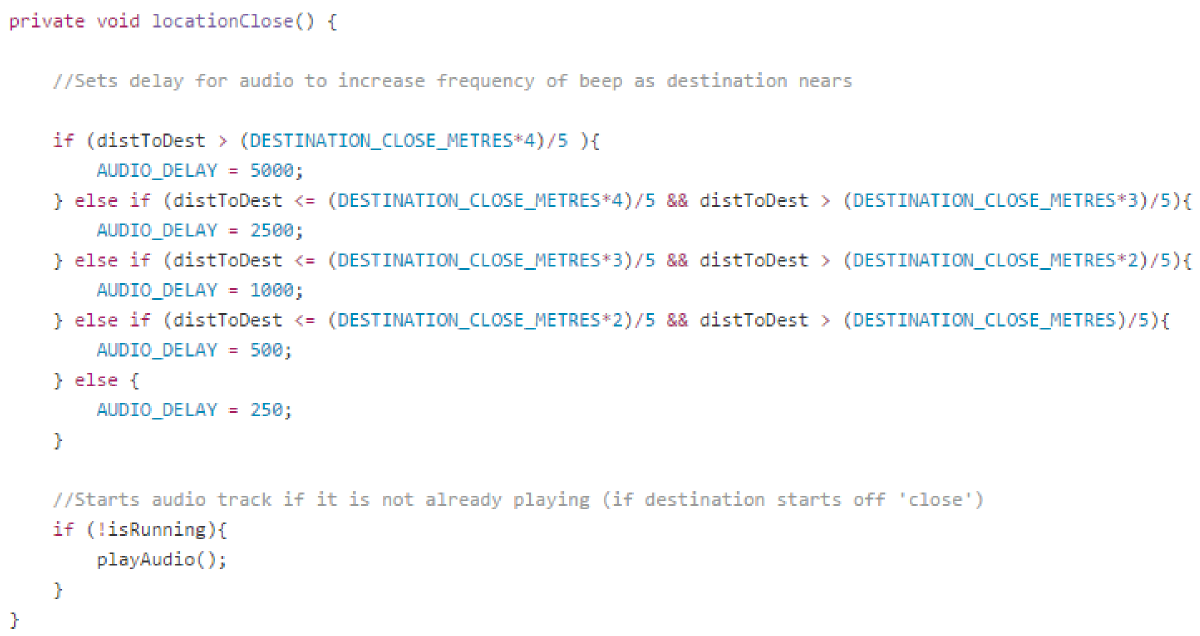
The original locationFar method was designed for when the driver was further than 1km from the destination, it was intended to keep the audio track playing steadily as long as the driver was making progress towards the destination. To do this I implemented a check to see if the distance remaining had decreased from the last update (allowing a tolerance for satellite errors). If so then the audio would continue to play, else it would stop until progress was made again.



```
private void locationFar() {

    if(((distToDest + PROGRESS_TOLERANCE) <  prevDist) && prevDist != 0){
        //if progress is being made on the journey keep playing audio
        if (!isRunning){
            playAudio();
        }
    } else {
        //no progress made, pause audio
        if (isRunning){
            stopAudio();
        }
    }
}
```

Figure 17: Original locationFar code

The original locationClose method decreased the audio delay variable as the distance to the destination decreased, so that the audio would speed up in ‘bands’ of 200 metre increments from the destination.



```
private void locationClose() {

    //Sets delay for audio to increase frequency of beep as destination nears

    if (distToDest > (DESTINATION_CLOSE_METRES*4)/5 ){
        AUDIO_DELAY = 5000;
    } else if (distToDest <= (DESTINATION_CLOSE_METRES*4)/5 && distToDest > (DESTINATION_CLOSE_METRES*3)/5){
        AUDIO_DELAY = 2500;
    } else if (distToDest <= (DESTINATION_CLOSE_METRES*3)/5 && distToDest > (DESTINATION_CLOSE_METRES*2)/5){
        AUDIO_DELAY = 1000;
    } else if (distToDest <= (DESTINATION_CLOSE_METRES*2)/5 && distToDest > (DESTINATION_CLOSE_METRES)/5){
        AUDIO_DELAY = 500;
    } else {
        AUDIO_DELAY = 250;
    }

    //Starts audio track if it is not already playing (if destination starts off 'close')
    if (!isRunning){
        playAudio();
    }
}
```

Figure 18: Original locationClose code

It was at this point that I hit a problem with the implementation. Upon testing of this version of the app, it was discovered that there was a delay in between getting the current location of the vehicle and updating the audio feedback based on this locations distance from the destination. This meant that measuring whether progress had been made while calling the locationFar method was difficult, as by the time this had been calculated the driver may have moved on. From this, the audio feedback provided by the method became useless as it would often pause at the wrong time in the journey. Methods were tried to overcome this including decreasing the time between location updates but the system was still struggling to determine whether the vehicle was making progress or not. In the end, the decision was made to remove locationFar and implement audio feedback in a different way.

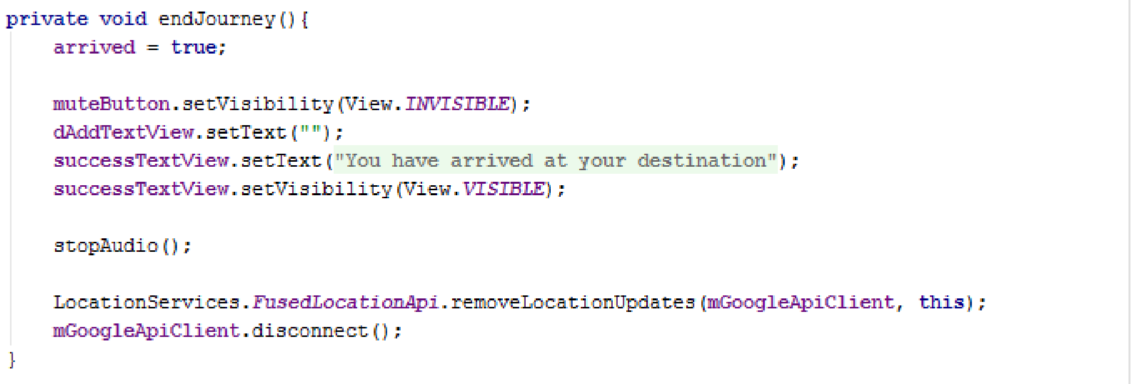
The solution I decided on involved adapting the locationClose method to work across the entire journey rather than just in the final kilometre. The reason for not selecting this method at first was down to the worry that the feedback from this method could potentially annoy a driver across an entire journey, particularly a longer journey.

In order to implement it, I kept a record of the initial distance to the destination when the journey began. I then sectioned off the journey into ‘bands’ calculated as a percentage of the original distance, so that after completing 20% of the journey the audio will speed up to let the user know they are getting closer. Once the user gets within the final three levels of audio frequency, the frequency of location updates is increased to give more responsive feedback as the user nears the destination. If the journey being undertaken is less than 1km, the old locationClose method is still used, as the user will not want too many levels of audio frequency for such a short distance. The full locationDelay method for setting the speed of the audio track can be seen in Appendix item 6.

Upon completing this section, version 0.7 of the app was ready to be tested and built upon.

7. Programming of extra features

Once the audio had been implemented, the basic functionality of the app was complete, as a journey could be made with responsive audio based on proximity to the destination. However, a few more features were added to improve the user experience. Firstly, I implemented another method which recognised when a journey was completed and ceased the audio feedback, along with hiding all unnecessary layout items and displaying a success message to the user. It is called when the newly updated location is within a certain distance from the destination to allow for a driver driving past their destination (for example to find a parking space).



```
private void endJourney(){
    arrived = true;

    muteButton.setVisibility(View.INVISIBLE);
    dAddTextView.setText("");
    successTextView.setText("You have arrived at your destination");
    successTextView.setVisibility(View.VISIBLE);

    stopAudio();

    LocationServices.FusedLocationApi.removeLocationUpdates(mGoogleApiClient, this);
    mGoogleApiClient.disconnect();
}
```

Figure 19: Checking when journey has ended

Another feature added was one that was requested by a test user, which is the mute button for the audio. To do this I used a toggle button which started or stopped the audio when pressed, along with a layout to clearly signify the buttons state to the user at a glance.

The addition of a mute button also raised another problem that was discovered during testing which was that when the screen goes to sleep during a journey, the mute button is not easily accessible and becomes dangerous to access while operating a vehicle (as the phone needs to be unlocked again). On top of this, the screen going to sleep pauses the activity being run, meaning that the audio track from the app goes off. The Journey activity therefore needs the screen to be kept on during use, which I accomplished using a WindowManager flag.

The final extra feature implemented involved displaying error-checking notifications to the user, should they use any part of the app in a way that was not intended. This was used when the user enters an unrecognised destination or one where a route cannot be found from the current location.

After these features were implemented, version 1.0 of the app was finished.

Version Testing

Throughout the development of the app, and in accordance with the development method I selected before beginning my implementation, I created a number of incremental builds for testing. The findings from each test were used to improve the code that had been written, with the ultimate goal of testing a complete working version of the app. While many features that were implemented (particularly during the earlier stages of development) could be tested while being programmed, certain parts required real-world unit testing due to the nature of the app. These tests often involved making journeys to various locations to test how certain aspects of the system would perform in real-world scenarios.

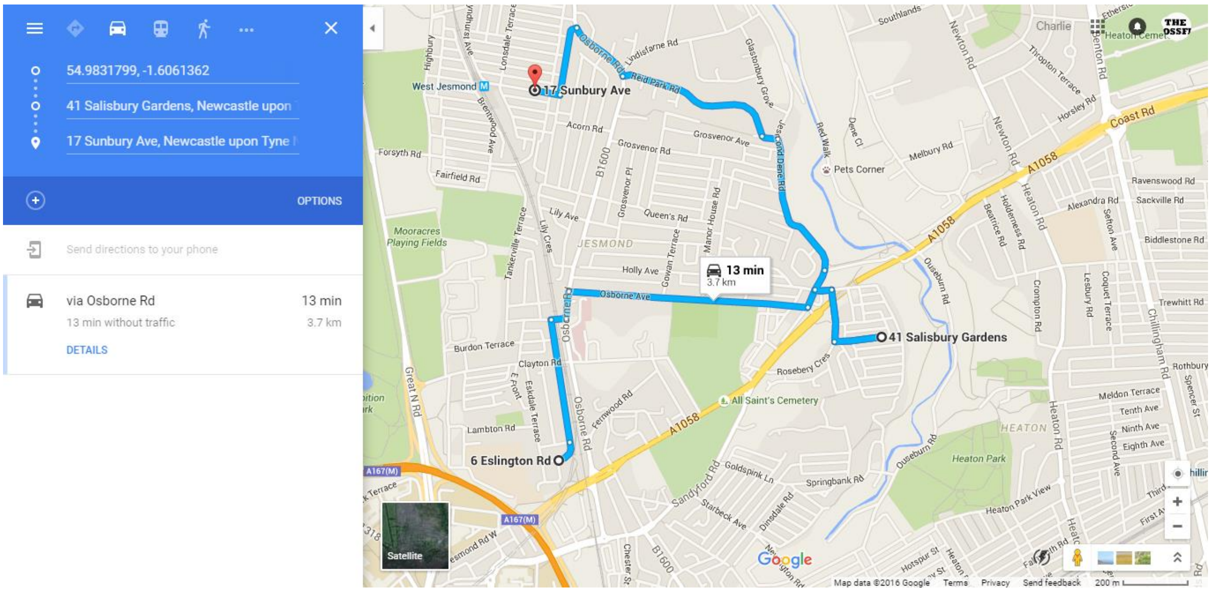
From these tests, a number of important developmental challenges were discovered, from difficulties in connecting with the API client during a journey to the difficulty in creating the locationFar method that ultimately led to its removal from the final system. A full list of these version tests can be found in the Appendix, item 7.

Results & Evaluation

Planning of Testing

In order to test how well the concept of my app worked in the real world, I decided that I needed to compare it to the type of system it was designed to improve upon. I therefore planned some final user testing in which volunteers were made to use both my app and a turn-by-turn based system. Through the collection of qualitative and quantitative data, I would hopefully then be able to assess how well the concept worked in practice.

The structure of the testing was as follows: after meeting up with the volunteer in their own vehicle, we travelled to the start location while I explained the concept for my app and what the testing would entail. Once at the start point, we travelled to the first destination using one of the navigation systems before pulling over. When the first destination was reached, we swapped over to using the other navigation system before travelling to the second destination. After pulling over at the second destination, I encouraged the volunteer to fill out a questionnaire (seen in Appendix item 8), before engaging in a short interview about the experience. When using the turn-by-turn system, the driver simply entered the destination address and followed the instructions given. When using my system, before entering the address I encouraged the driver to plan a route using a map provided or a digital map on their smartphone. I also advised them to use navigational skills such as looking out for road or street signs as well as using local landmarks as route markers. During the journeys I had no interaction with the driver, allowing them to follow their own route without interfering.



54.9831799, -1.6061362

41 Salisbury Gardens, Newcastle upon

17 Sunbury Ave, Newcastle upon Tyne

via Osborne Rd 13 min

13 min without traffic 3.7 km

Figure 20: Route taken during the final user testing, the journeys being from Eslington Road (Jesmond Metro station) to Salisbury Gardens, then to Sunbury Avenue.

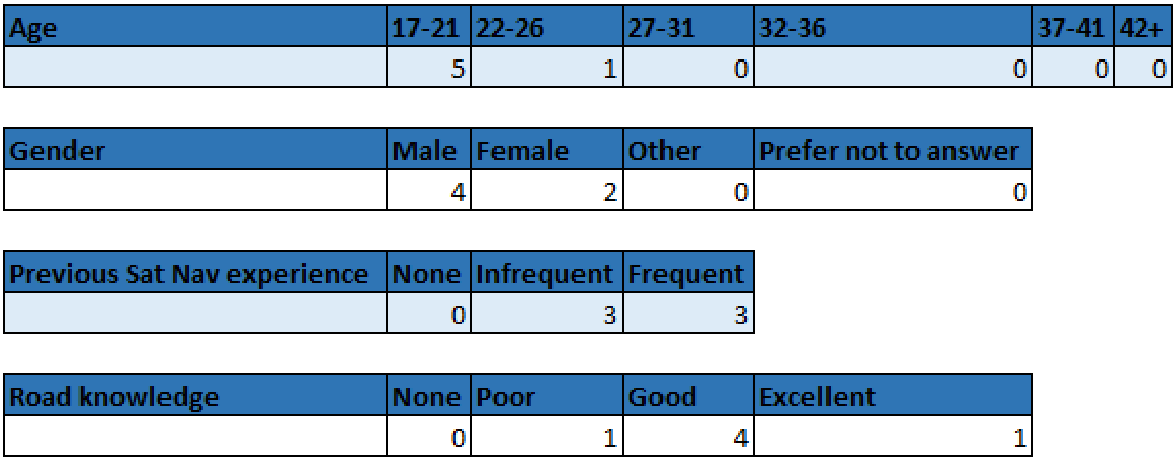
To run the testing, I first had to decide on a route that would be driven by the volunteers. As there were two systems to test, I specified two destinations to be driven to, so that one journey could be done with each system. The test began at the car park outside Jesmond metro station with the two destinations being addresses on Salisbury Gardens and Sunbury Avenue, both also in Jesmond. In the interest of scientific fairness, volunteers 1, 3 and 5 were made to use the turn-by-turn system first and volunteers 2, 4 and 6 used my system first, this was to ensure that if either of the routes selected were more difficult than the other, some would use the professionally made system for it and the others would use my system. The fastest routes between the destinations can be seen in Figure 20.

The routes were planned to take roughly the same amount of time (around seven minutes each) and to have roughly the same complexity (seven junctions meaning seven decisions to be made as to the direction of travel, assuming the driver follows the fastest route for the journey). I chose the two destinations to be house addresses in residential areas, the reasoning for this was twofold. Firstly, I predicted that the majority of volunteers for the testing would be familiar with the area already due to it being a student area and my volunteers being predominantly students. I therefore decided it would be pointless to use local landmarks or well-known places as the destinations as the likelihood of the volunteers knowing how to get there already would be high, meaning the system would not get tested properly. By using residential addresses, the volunteers could properly explore the area and search for a destination they wouldn’t know the exact location of. The second reason for choosing these destinations was that the residential areas they are on often have multiple streets that look very similar, particularly in the area surrounding the second destination. By setting the end points to be on one of these roads, it encourages the volunteer to use the feedback from the app to gauge their proximity to the correct street.

Results

During the course of testing, I was able to obtain feedback from six different volunteers. All the volunteers partook in both parts of the phase (the road tests and the interview/questionnaire), giving me a range of responses to analyse. The questionnaire results can be found in Appendix item 9 while the final user transcripts can be found in Appendix item 10.

From the first few questions on the feedback, I was able to gain an insight into the basic demographic of the volunteers. The group contained four males and two females, with all being in the 17-21 or 22-26 age ranges. Although this leads to an underrepresentation of older people in my results, I found it difficult to find anyone in this range who would take part. Half of the volunteers described their Sat Nav usage as frequent, while half described it as infrequent. All claimed to have used turn-by-turn systems previously. Finally, one out of the six volunteers said they had poor knowledge of the local roads on which the tests took place, four said they had good knowledge and one answered excellent. All six volunteers had some form of driving experience in the area prior to the test.



Age	17-21	22-26	27-31	32-36	37-41	42+
	5	1	0	0	0	0

Gender	Male	Female	Other	Prefer not to answer
	4	2	0	0

Previous Sat Nav experience	None	Infrequent	Frequent
	0	3	3

Road knowledge	None	Poor	Good	Excellent
	0	1	4	1

Figure 21: Tabulated demographic results

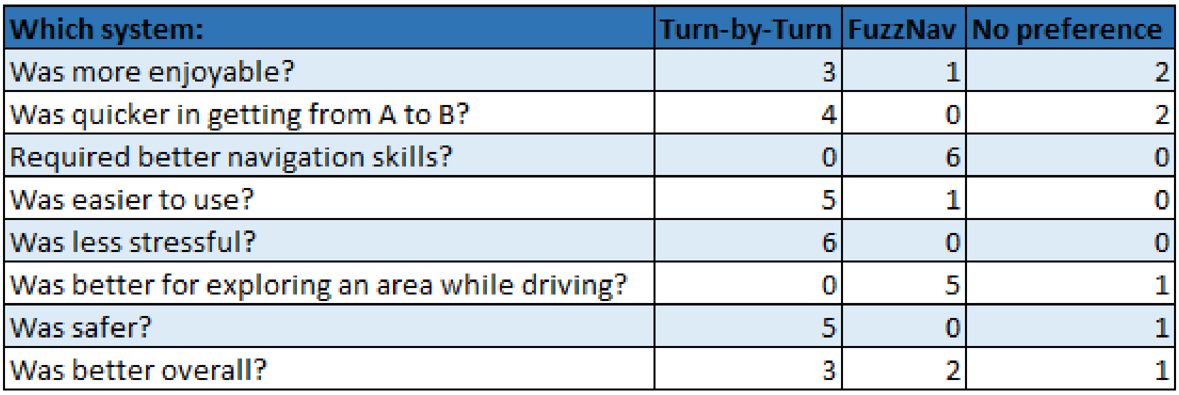
Although care was taken to ensure that the tests were run as similarly as possible for each driver, there were a number of factors that were out of my control that may have affected the driving performance during some of the volunteers. While the destinations remained the same for each test, I allowed the drivers to select their own route there and deliberately chose journeys with multiple viable routes due to the nature of the navigational concept being tested. This meant that some drivers selected roads with speed bumps/difficult junctions/more traffic etc. that could have affected their answers when giving feedback. Due to time constraints, two of the drivers performed the testing in the evening, where the dark conditions may have negatively impacted their ability to use road signs and route markers for navigation. Furthermore, one of these two tests also took place in foggy conditions, further worsening the visibility and potentially lessening the effectiveness of my system.

The majority of the volunteers performed well while using the stock system with explicit instructions. As all of the drivers had prior experience with using a turn-by-turn system they were all able to set and follow the instructions given by the device without problem; despite the fact that as volunteer 5 said, the system I provided was *“really difficult to program”*. This observation corroborates Speake’s conclusion that the vast majority of student-aged people were *“confident and happy”* using turn-by-turn Sat Navs as their main mode of navigation (Speake, 2015).

From observation, I found that user’s performance with the implicit system was more mixed in its success. On the whole, everyone managed to navigate through and program the app, owing to its simple interface. Most of the volunteers managed to follow the audio feedback to navigate towards the destination, with some giving positive feedback about how it helped them gauge the distance left to the destination (volunteer 3) or helped them know when to *“start looking for road signs”* (volunteer 6). On the other hand, volunteers 1 & 6 admitted that while using the system they felt a sense of being lost; while volunteers 2 & 4 admitted that using the system was *“a bit stressful”*. The results set gained could have been skewed slightly by the fact that some of the volunteers had prior knowledge of the testing area, with volunteer 5 stating in the interview that they *“kind of knew where I was going”* and volunteer 1 echoing this sentiment during the journey.

Aside from the main functionality of the app, some of the extra features were also used during the testing. For example, volunteer 6 used the mute button when nearing the destination and began looking for the correct house number by eye. Volunteer 2 initially entered the wrong address and was returned an address in a different city, however due to the visual feedback on the Journey page, they quickly noticed their mistake and re-entered the address correctly.

The second half of the questionnaire enabled me to collect some quantitative feedback about how my system compared directly to the turn-by-turn system. It asked the volunteers to choose which system was best for a number of different factors. By evaluating this feedback, I was able to assess which aspects of navigation my system worked well in and which it performed poorly in.



Which system:	Turn-by-Turn	FuzzNav	No preference
Was more enjoyable?	3	1	2
Was quicker in getting from A to B?	4	0	2
Required better navigation skills?	0	6	0
Was easier to use?	5	1	0
Was less stressful?	6	0	0
Was better for exploring an area while driving?	0	5	1
Was safer?	5	0	1
Was better overall?	3	2	1

Figure 22: Tabulated test results

Arguably the most positive result from the questionnaire was gained from the volunteers’ answers as to which system required better navigational skills. Every one of the drivers picked my system for this answer, showing that the use of such a system by drivers would hopefully lead to the development of navigational skills through regular usage. Throughout the tests I observed drivers using basic techniques of navigation when using the implicit system, with drivers seeming more engaged with the environment outside the vehicle. Despite giving generally negative feedback for the system, Volunteer 6 admitted that she felt like she knew the test area better after getting lost in it during the journey.

A similar result was observed in the exploration category, with five out of the six volunteers agreeing that the implicit system was better for exploring an area during the journey. Volunteer 1 stated that FuzzNav is *“much better if you want to just explore the area”* due to the fact that the user does not have to *“stare at my phone all the time”*. This clearly leads on from the engagement with the environment previously mentioned as the driver is forced to pay attention to turnings, landmarks and the general geography of the route taken. By actually looking at and memorising the route between the start and end locations as the volunteers were encouraged to do, the drivers spatial awareness of their surroundings is improved. Interestingly, the only volunteer to have no preference between systems in this category was also the only volunteer to have a poor knowledge of the local roads. This could potentially indicate that an implicit system demands more focus from drivers in an unfamiliar environment and therefore allows less scope for observing the surrounding environment due to the driver needing to memorise a completely new route. If this is true, it could partly explain the volunteer in question repeatedly alluding to feeling lost during the test.

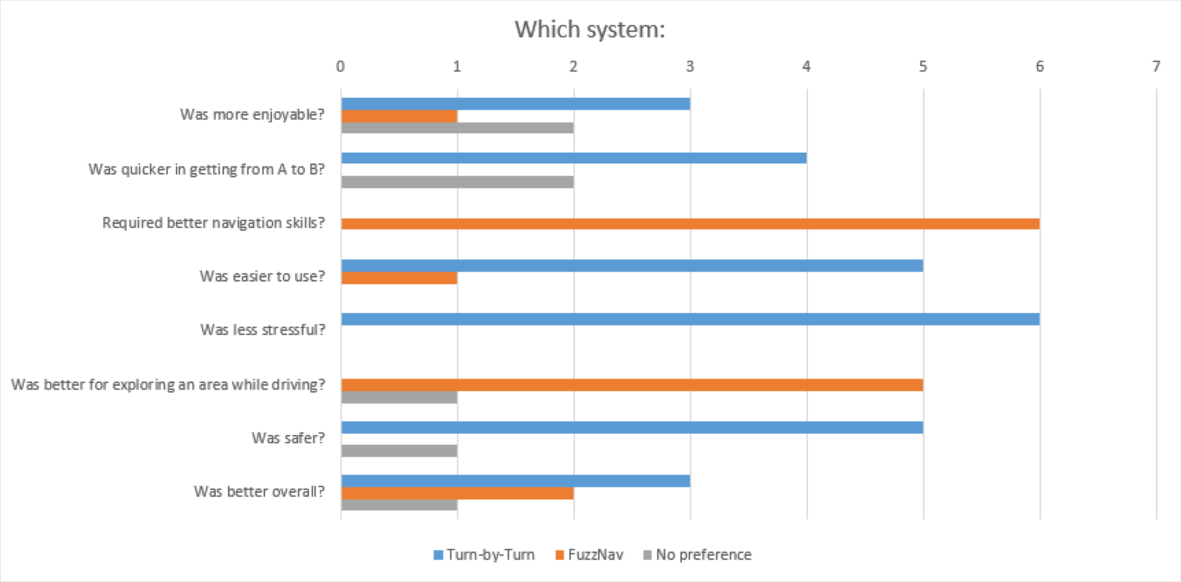


Figure 23: Graphical test results

On the other hand, there were clearly some areas in which my system was deemed inferior to a turn-by-turn system, as can be seen from the graphical results. My system was perceived by all volunteers to be more stressful to use than the standard system, with volunteers 2 and 4 in particular highlighting this point in the subsequent interview. This could be due to a number of factors, in particular the fact that it is most likely the first time the volunteers would have used an implicit navigation system of any kind, let alone an audio-based one. More research would therefore be needed to observe whether this feeling of being stressed during usage would persist once the driver had used the system over a longer period of time and gotten used to the way it works. On the other hand, every single volunteer had at least some experience with using a turn-by-turn system before so knew what to expect during the journey in which they used one.

Another factor that could have contributed to this response would be the choice of audio feedback used. The audio aspect involved a lot of beeping, particularly as the volunteer got closer to the destination, and the feedback had to remain at a loud volume in order to be distinguishable over the ambient noise of the vehicle and surroundings. The choice of audio feedback led to volunteer 2 jokingly describe it as *“like a bomb about to go off”* during the journey. This was exacerbated when a driver took a wrong turn near the destination as was the case with volunteers 1, 2 and 6. A wrong turning at this point led to the driver having to listen closely to the audio to work out that they had made an error, further adding to the stress of veering off route. This stress is likely what led to volunteer 6 muting the app when nearing the destination.

Another category in which my system was deemed inferior to the turn-by-turn system was with regard the ease of use. Five of the six volunteers stated that the professional system was easier to use during the test, with volunteers 2, 3 and 6 in particular stating that my system was more difficult to use. Volunteer 1 went in depth as to the particular area they found difficult, stating that they found it tricky due to *“missing the aspect of looking at a map”*. Despite the negative feedback for this area, it is not something that concerns me when evaluating the overall effectiveness of the concept. The fundamental idea behind the concept means that the implicit system is more difficult to use by definition, as it is designed to help develop skills, whereas the turn-by-turn system is specifically designed to make journeys as easy as possible for the user. Therefore, this was not an area of consideration when I was planning the system. The flip side to this is that one of the volunteers said that mine was easier to use, although this could be attributed to the fact that they *“don’t really like using Sat Navs”* and finding the one used *“really difficult to program”*. On top of this, some of the volunteers enjoyed the added difficulty of the system, with volunteer 4 stating that FuzzNav is good for *“people who like a challenge”* and volunteer 3 saying that using the app to navigate was *“rewarding”*.

An altogether more concerning set of responses were to the question of which system is safer to use. All but one of the volunteers agreed that my system was more dangerous, with volunteer 1 explaining that they were concerned by the need for *“last minute turns”* and *“looking […] around while I was driving”*. This is clearly a more serious issue to be brought up than any of the other feedback as any system that introduces a substantial amount of danger into driving would never become commercially viable in the real world. The safety of vehicle occupants and others around the vehicle has to be paramount when designing a navigation system due to the nature of the environment in which it will be used.

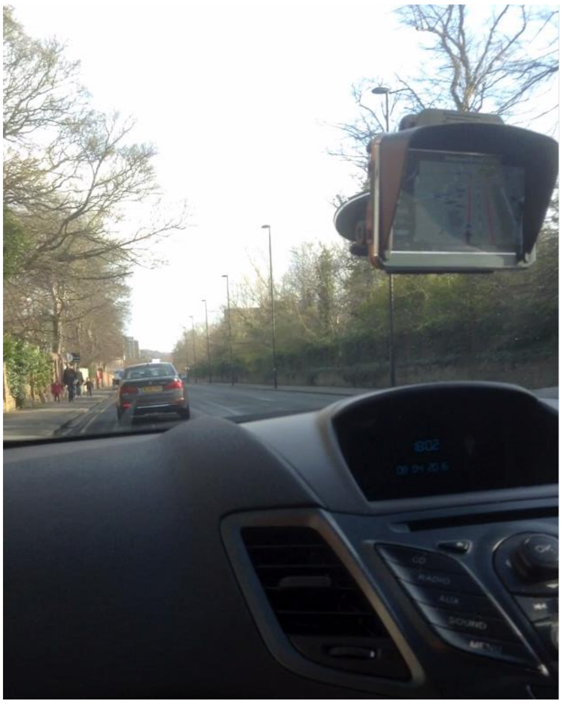


Figure 24: A still image from video taken during the testing phase – this journey was using the turn-by-turn system.

The other side of this argument is that driving will never be a completely safe task due to the likelihood of human error. This is further exacerbated by the use of a distracting navigation device such as a turn-by-turn Sat Nav system which has itself been accused of causing safety issues during usage. A study by Brake shows that 15% of drivers have made *“illegal or risky manoeuvres”* when following Sat Nav instructions (Brake, 2015), with programming the device while driving and focussing on the screen instead of the road being highlighted as particular problems. In contrast, the very concept of an audio-based system eliminates the former problem, something that was highlighted by volunteer 1 (*“I didn’t have to stare at my phone”*) and volunteer 4 (*“not much on the screen to distract you”*). A study into the feasibility of my system therefore needs to further compare its safety against the currently available Sat Nav market before any further implementation choices can be considered worthwhile.

In general, the volunteers found the turn-by-turn system quicker for navigation between locations, with 4 of the responses stating this. However, two people stated that they saw no difference, these were volunteers 4 & 5. Both of these drivers took the optimum route when using my system and navigated to the destination without missing any turnings meaning there was probably not much actual difference in the times for each journey. This is not a category in which I am aiming to improve upon the turn-by-turn system so the results are expected. The explicit instructions will always guide the driver along the optimum route and the ease of following such instructions makes it unlikely for a driver to make any incorrect decisions that prolong the journey. It is therefore almost impossible to improve on this system – the aim of the driver while using the implicit system should be to make the journey in a similar time as if explicit instructions were used by navigating the route perfectly.

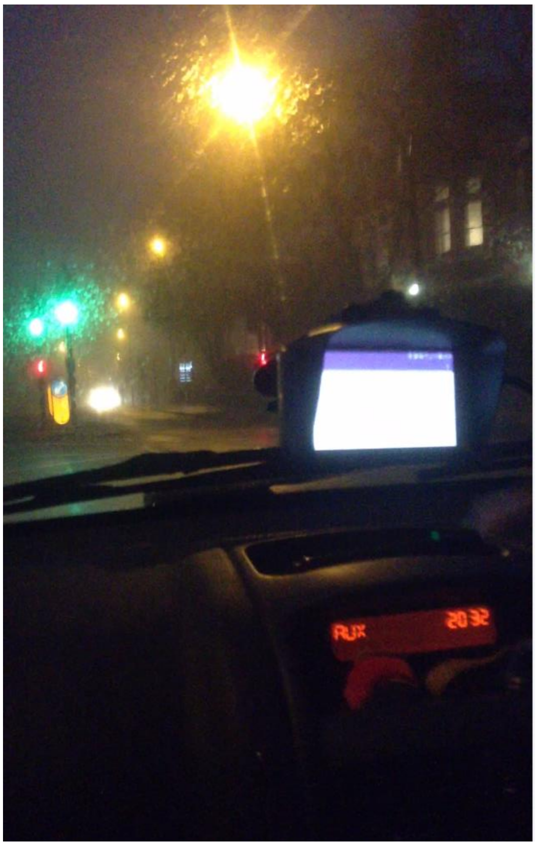


Figure 25: A still image from video taken during the testing phase – this journey was using the implicit system.

A more subjective set of responses was gained from the question of which system the driver enjoyed using more. Three people said the explicit system, two said they had no preference and one said my system was more enjoyable. The response given seemed to depend more on the personal traits of the driver as people tend to look for different things in a navigation system. For example, volunteer 2 stated that they preferred using my system as they *"liked finding areas by myself rather than using technology"*, adding that they found the experience *"fun"*. The experience can be a double-edged sword though, with volunteer 3 describing the test as "*rewarding*" but *"difficult"*.

The final question asked the driver to select which system was better overall. The result was surprising, with two of the volunteers claiming my system was better compared to three choosing the turn-by-turn system. As the turn-by-turn system was built by a professional company and is an altogether more complete package than my system, I was surprised by these results. However, they could be skewed by the fairly open-ended question and the fact that the volunteers knew that I had created the implicit system and were answering the questionnaire in front of me. The response therefore does not hold much weight and therefore cannot be analysed too closely. However, it is interesting to note that drivers with poorer knowledge of the surroundings generally tended to prefer the turn-by-turn system as it is easier to use when in completely unfamiliar areas. Drivers with better knowledge tended to prefer my system as it is clearly easier to navigate a route that a driver is already somewhat acquainted with. This will hopefully mean that as a driver uses the implicit system further, the resulting development of knowledge of the surroundings will lead to an increased enjoyment of the system.

Evaluation of Project

In this section, I will attempt to evaluate the overall approach taken while undertaking this project. I will look at each section of work individually and look into how it contributed to the completion of the overall objectives of the project, as well as where any improvements could have been made.

Background

During the course of the background research phase, I read around a number of areas in order to get an overview of the state-of-the-art for Satellite Navigation technology. Successful completion of this chapter would satisfy objective 1:

> *“To complete a thorough literature review to provide a knowledge base for the design and implementation of my system.”*

Design & History of Satellite Navigation Systems

This section gave me a platform of knowledge on the general workings of Sat Nav systems, as it is not an area I have previously studied in any great detail. In particular, it helped me understand the stages required for a Sat Nav system to get its own location and plot a route to its destination, information which was later used to create the seven stages of my implementation. This section helped me understand the technical details of how the systems location can be determined from a call to a satellite and of how the quickest route to a destination would be calculated by the Google Distance Matrix API that I ended up using.

Sat Nav Issues

The knowledge gained from this section allowed me to focus in on the exact problem that I wanted my implicit system to tackle. It allowed me to fully justify the reasons behind why I was creating the system and led to me altering the aim of the project based on the information I learned. I also read about how navigation systems can distract drivers during this section, information from which could be useful if I were to do further work on assessing the safety issues of my system as mentioned in the testing results section.

Alternative Systems

This section helped me pinpoint how exactly my implicit system would work, with the firm decision on the use of audio-based feedback being made upon completion of this topic of reading. It also opened my eyes as to the myriad of different methods navigation systems can use to guide a driver to their destination and helped me think outside the box about navigation in general.

In-Car Testing

From this section I was able to design a more complete testing scheme for my final solution that contained different forms of user feedback to analyse. It helped me design parts of my ethnographic research such as the filming of journeys to be displayed as screenshots in my report, as well as helping me design the questionnaire section based on the testing performed by Komninos et al.

Overall I think the completion of this section met the objective required. The knowledge gained from this research was used in the preceding sections and by building upon it I created and tested a working system. I do believe I should have done more research however, particularly in areas regarding actual implementation details. When I began the implementation section, I had to continually find out how to implement certain features to do with locationing and playing audio in particular. This slowed my implementation progress down and inevitably led to my time-issues and relatively simple implementation. If I had more thoroughly researched how Sat Nav applications are created, I feel I could have improved upon my final system.

Design

During this phase I planned and performed a set of initial user tests using a self-designed prototype made using a Gadgeteer kit, thus fulfilling objective 2:

> *"To create and use a Wizard-of-Oz prototype in order to test basic user interaction concepts."*

Furthermore, after deciding on the tools and methodology most suitable for the implementation, I began designing how the app would be laid out and how the functionality would work in order to guide the user to their destination. With this, I was attempting to satisfy objective 3:

> *"To use findings from the initial user tests to design a suitable application."*

Initial User Testing

On the whole, this section was a big help during the early stages of my project. It helped me get a much clearer idea in my mind of how the system would actually work in practise and I feel that without it, my final design may not have been as focussed or effective. The stage gave me some good ideas for features which ended up being added to the final system (such as the mute button) as well as features that I attempted to implement (such as the more responsive audio scheme).

The problems I encountered however related to the programming and using of the Gadgeteer system. As I had never programmed in C# before, I found it particularly difficult to get the system to work as I wanted it, this was exacerbated by a faulty buzzer module that occasionally gave unnecessary extra feedback. Programming the prototype slowed me down considerably during the early months of my project and led to less time being spent on background reading than I would have liked.

During the planning for this section, I devised a testing method that I deemed too complex for what I needed at that time. However, I later revisited it and used it as the basis for my final user testing where the feedback required needed to be in more depth.

Design

I made the design choice to keep the application as simple as possible aside from all required functionality due to the fact I was testing the concept rather than my app-making skills. This is a decision I stick by although with hindsight I could have had a little more time to spend on the implementation so could have added some more features.

Looking back, the choice of design for the back-end (i.e. API calls etc.) was not entirely suitable for the application, the time delay in getting the current location, creating the Distance Matrix API call and parsing the response meant the system was not as real-time as I would have liked. This had a knock-on effect on the audio section of the app as this relied on responsive distance data from the back-end. This led to audio not being able to be muted when the vehicle was stopped and not being quick enough to respond when a driver takes a wrong turning. From this, I made the choice to simplify the audio feedback at a later stage, a decision I would have preferred not to have made.

Overall, I feel I have met objective 2 completely. After some difficulty I was able to develop the prototype and gained valuable information from the testing phase it was used in.

Although I ran into the aforementioned design problems, I still feel I have somewhat completed objective 3. While the system that was created was not quite as designed, it still uses implicit audio instructions to guide the driver to their destination.

Implementation

For this section I created the application by implementing features one at a time on top of each other, attempting to improve on the previous iteration and satisfy objective 4:

> *"To create a series of working versions to evaluate the suitability for each to meet the user's needs."*

The conclusion to this section involved the completion of the final solution, thereby meeting objective 5:

> *"To create a final working system for testing"*

The version numbers for the following are taken from the Version Testing log.

Version 0.1 – Finding and updating locations

This was the first point in which I had a testable system, it could obtain, update and display location coordinates. By now the following requirements had been achieved:

FR1.1, FR1.2, FR2.2, NFR2.1, NFR2.4 & NFR2.5

Version 0.2 – Distance call to API

By this point, the system could call to the Google API to return the optimal route distance between the set locations, thus satisfying:

FR1.6, NFR2.2, NFR2.3

Version 0.4-0.55 – Initial audio

This was where I was attempting to implement the complex audio system I had designed using the distances returned from the API. By now I had satisfied:

FR1.3, FR1.7, FR2.4

Version 0.6-0.7 – Completion of audio

At this point I had redesigned my audio feedback system to a simpler one that was able to work better alongside the back-end already implemented. I had now completed:

FR1.4

Version 0.8 – Final features

Here I had added in extra features such as notifications, the mute button etc. thereby satisfying requirements:

FR1.5, FR2.3

Version 1.0 – Full working system

At this stage, I completed the layout of the application as previously I had kept it blank to allow for test data to be displayed. At this point, the app was complete, thereby satisfying most of the remaining requirements:

FR3.1, NFR1.1, NFR2.6, NFR2.7

The only requirement that remained unfulfilled was:

FR2.1 – When the user's input destination is ambiguous, a list of potential destinations must be displayed for them to select from

This was due to time constraints and to me not knowing how I would go about implementing this at the time. As it was not a vital feature that did not contribute to the main functionality of the app, I chose to skip it.

I believe that the overall implementation of the app was good, I managed to create it to match very closely to the initial design, with nearly all the required features having been included. The app does as it is supposed to by aiding navigation for the driver. However, there are areas in which I am disappointed with the final product, the main one being the audio feedback for reasons previously mentioned. Despite this, I feel that I have still met objectives 4 and 5 for this project.

Final User Testing

This section involved me testing my solution in the real world and analysing the findings in order to complete the objective:

> *“To test my final system with real users against a turn-by-turn system to compare the strengths of both implementations.”*

Overall, I feel satisfied that I achieved this objective. I was able to collect both qualitative and quantitative data from a group of people that I could then analyse to come to a meaningful conclusion. The method in which the testing was conducted remained scientific throughout; by using exactly the same routes and having half the drivers use each system for each route the testing remained fair and controlled.

There were some areas in which I feel the final user testing could have been improved however. For one, I felt the turnout for the phase was poor as only six people were able to be tested upon. When I began planning the testing I wanted to aim for a minimum of 12 people in order to get a range of different opinions, likewise I was hoping for a more varied group of individuals. I found it difficult to find any non-students to test the system upon and struggled to find people I knew in Newcastle due to the majority of testing being done over the Easter break. If I were to do the project again, I would attempt to advertise the testing to a larger range of people to get a more rounded set of opinions.

Conclusions

Aim & Objectives

Upon completion of the background section of this project, I updated the aim to the following:

> *“To create a working in-car navigation system using implicit instructions to attempt to combat the de-skilling of modern drivers”*

In order to meet the aim, I set a series of objectives to be met during the course of the project. The extent to which I met these objectives has previously been discussed in the evaluation section of this report, with the overall conclusion being that each was met to at least some extent.

To work towards the given aim, I created an Android navigation application to be used on car journeys with implicit audio-based instructions to guide the driver to their destination. I was then able to take the app that I created and test it on volunteer drivers in real-world scenarios in order to assess its strengths and weaknesses against an off-the-shelf system with explicit verbal instructions.

The results from this testing suggested that navigating using my implicit system required greater use of navigational skills on the part of the driver and a greater engagement with the external environment during the journey. This would indicate that travelling using the system over a longer period of time would lead to the sharpening of such skills. However, such development is hypothetical at the moment due to the lack of research performed on long-term usage of the system. Research in this area would be required before I could be confident that my system actually did combat the de-skilling of drivers.

It is also important to note that throughout the development and testing of my system, I have been able to observe certain situations in which its usage would be appropriate and other situations where it would not be useful. The very nature of the audio feedback renders the system useless on longer journeys without further operating modes being added. If a user were to take a wrong turning at the beginning of a long journey, the audio feedback would not be responsive enough to inform them of this fact for a long time. Furthermore, if the driver were to travel a large distance using the system, the repetitive audio track would no doubt become annoying and possibly even distracting. It should therefore be noted that the system was only ever intended to be used on short journeys, particularly in an urban environment where its feedback can be better used to navigate around similar-looking city streets. This was made clear during the testing of the system, where all journeys made during both version testing and user testing remained over short distances only.

Future Work

The system that I created was a simple application that was designed primarily to test the underlying concept of implicit navigation, it was not designed for the purpose of being used in day-to-day life by drivers looking to navigate to their destination. Now that the feasibility of the concept has been proven, further work can be done to embed the idea within people's day-to-day lives. The application can for example be extended to add in further features such as more responsive audio feedback. Further user testing would be required for this to happen in order to get feedback on the sort of features users would like to see in the system.

From the final user testing feedback I obtained, it was clear that most users still prefer some of the features of the classic turn-by-turn system and I believe that developing a standalone implicit system is the wrong way to approach the problem of de-skilling. There are still many situations in which the ease and simplicity of following explicit verbal instructions is preferential within day-to-day life. I therefore believe that the implicit audio feedback would be better suited as an operating mode within a regular Sat Nav system. Implementing the system in this way would thereby offer drivers who wish to navigate using this more challenging and rewarding method the opportunity to, while still allowing drivers to use the explicit method for all other journeys.

Before my system could begin to be integrated into current systems, the issues brought up in the final user testing section would have to be addressed. Primarily, the feedback pertaining to the safety of the concept would need to be evaluated further, with much more testing required within the area to ensure that the concept would be safe for drivers to use while navigating in the real world.

Skills Learned

Before starting the project, I had a limited knowledge of how Sat Nav systems functioned in order to calculate the optimal route from the systems own location to a user-specified location. After gaining first-hand experience with using techniques such as Geolocationing, trilateration and algorithmic routing, I feel I now have a much better understanding about how such systems work. While programming my final solution I utilised a number of programming techniques I was unfamiliar with, examples being API calls, AsyncTasks and Androids inbuilt geocoder. I now feel comfortable working with these features and would be able to use them again should a future project require it.

The need for in-car testing to be performed also required new skills to be developed as it was not something I had ever done before. Through research of both a primary and secondary nature, I was able to further my knowledge in scientific testing and logging of user data in order to use the results in my report. The running of the initial user testing also introduced me to working with the Gadgeteer physical computing kit and the C# programming language. Although the prototype I ended up programming remained fairly simple, I was still required to learn the basics of programming the system, skills which will again be transferrable to any projects in the area.

Finally, the research required prior to the projects commencement allowed me to hone my critical evaluation skills when studying scientific papers and articles. This is an area that I have done little previous work in, but I now feel more confident in assessing and utilising information from such research papers.

Final Thoughts

Overall, I believe that the FuzzNav project has been a success. From the initial testing results, it seems as though the concept meets its goal of combatting the de-skilling of drivers by requiring the development of navigation skills during usage. It has also provided a base of research onto which any future work in the area of implicit navigation can expand on with the long-term goal of solving the problem set out in my hypothesis. On a personal level, I feel that through the development and testing of the system I have met the aim set out in the project. In doing so, I have learned and developed valuable skills of my own that I hope to be able to transfer into future work.

References

Android Developers, n.d. *Location and Maps.* [Online]
Available at: http://developer.android.com/guide/topics/location/index.html
[Accessed 1 April 2016].

Aporta, C. & Higgs, E., 2005. Satellite culture: Global positioning systems, Inuit wayfinding, and the need for a new account of technology. *Current Anthropology,* 46(5), p. 745.

Aslan, I., Schwalm, M., Baus, J., Krüger, A. & Schwartz, T., 2006. *Acquisition of spatial knowledge in location aware mobile pedestrian navigation systems.* Helsinki, Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '06).

Braddon, D., 1999. *Commercial Applications of Military R&D: U.S. and EU Programs Compared.* Pittsburgh, European Union Studies Association.

Brake, 2015. *Turn around when possible: one in seven risking lives to correct sat-nav mistakes.* [Online]
Available at: http://www.brake.org.uk/news/1329-incartech-jan15
[Accessed 20 April 2016].

Brown, B. & Laurier, E., 2012. *The normal natural troubles of driving with GPS.* Austin, Texas, Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12).

Brustein, J., 2014. *GPS as We Know It Happened Because of Ronald Reagan.* [Online]
Available at: http://www.bloomberg.com/bw/articles/2014-12-04/gps-as-we-know-it-happened-because-of-ronald-reagan
[Accessed 3 March 2016].

Computer History Museum, n.d. *ENIAC.* [Online]
Available at: http://www.computerhistory.org/revolution/birth-of-the-computer/4/78
[Accessed 2 March 2016].

Danchik, R. J., 1998. An Overview of Transit Development. *Johns Hopkins APL Technical Digest,* 19(1), pp. 18-26.

Department for Transport, 2015. *Table NTS0907: Cars with or without satellite navigation technology by type: England, since 2009.* [Online]
Available at: https://www.gov.uk/government/statistical-data-sets/nts09-vehicle-mileage-and-occupancy
[Accessed 6 March 2016].

Forlizzi, J., Barley, W. C. & Seder, T., 2010. *Where should i turn: moving from individual to collaborative navigation strategies to inform the interaction design of future navigation systems.* New York, Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '10).

GIS Center, 2008. *Geocoding in ArcGIS.* [Online]
Available at: http://ocw.tufts.edu/data/54/626652.pdf
[Accessed 1 March 2016].

Goessl, L., 2012. *GPS fail: Japanese tourists follow course into Australian waters.* [Online]
Available at: http://www.digitaljournal.com/article/321344
[Accessed 1 March 2016].

Google Developers, 2016. *Google Maps Android API.* [Online]
Available at: https://developers.google.com/maps/documentation/android-api/
[Accessed 29 March 2016].

Google Developers, 2016. *Google Maps Distance Matrix API.* [Online]
Available at: https://developers.google.com/maps/documentation/distance-matrix/
[Accessed 29 March 2016].

gov.uk, 2016. *Ethnographic research.* [Online]
Available at: https://www.gov.uk/service-manual/user-centred-design/user-research/ethnographic-research.html
[Accessed 15 March 2016].

Green, P., Hoekstra, E., Williams, M., Wen, C. & George, K., 1993. *Examination of a Videotape-Based Method to Evaluate the Usability of Route Guidance and Traffic Information Systems,* Ann Arbor: The University of Michigan Transport Research Institute.

Jensen, B. S., Skov, M. B. & Thiruravichandran, N., 2010. *Studying driver attention and behaviour for three configurations of GPS navigation in real traffic driving.* Atlanta, Georgia, Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '10).

Jesda.com, 2012. *How In-Dash Navigation Worked In 1992 – Olds Was First.* [Online]
Available at: http://jesda.com/2012/01/11/how-in-dash-navigation-worked-in-1992-olds-was-first/
[Accessed 3 March 2016].

Knoblauch, H., 2005. *Focused Ethnography.* [Online]
Available at: http://www.qualitative-research.net/index.php/fqs/article/view/20/43
[Accessed 15 March 2016].

Komninos, A., Barrie, P., Stefanis, V. & Plessas, A., 2012. *Urban exploration using audio scents.* San Francisco, Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '12).

Lee, J., Forlizzi, J. & Hudson, S. E., 2008. Iterative design of MOVE: A situationally appropriate vehicle navigation system. *International Journal of Human-Computer Studies,* 66(3), pp. 198-215.

Lendino, J., 2012. *Early GPS Systems, Portables, and Cell Phones.* [Online]
Available at: http://uk.pcmag.com/consumer-electronics-reviews-ratings/64623/news/the-history-of-car-gps-navigation?p=2
[Accessed 3 March 2016].

Leshed, G., Velden, T., Rieger, O., Kot, B. & Sengers, P., 2008. *In-car gps navigation: engagement with and disengagement from the environment.* Florence, Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '08).

Mio Technology, 2012. *GPS explained.* [Online]
Available at: http://eu.mio.com/en_gb/global-positioning-system_how-a-sat-nav-works.htm
[Accessed 1 March 2016].

Orlando Sentinel, 1996. *Oldsmobile's Guidestar Needs To Navigate Around Obstacles.* [Online]
Available at: http://articles.orlandosentinel.com/1996-07-25/topic/9607230882_1_guidestar-oldsmobile-larry-cummings
[Accessed 3 March 2016].

Pfleging, B., Schneegass, S., Meschtscherjakov, A. & Tscheligi, M., 2014. *Experience Maps: Experience-Enhanced Routes for Car Navigation.* Seattle, Washington, Proceedings of the 6th International Conference on Automotive User Interfaces and Interactive Vehicular Applications (AutomotiveUI '14).

physics.org, n.d. *How does GPS work?.* [Online]
Available at: http://www.physics.org/article-questions.asp?id=55
[Accessed 28 April 2016].

Robinson, S., Jones, M., Williamson, J., Murray-Smith, R., Eslambolchilar, P. & Lindborg, M., 2012. Navigation your way: from spontaneous independent exploration to dynamic social journeys. *Personal and Ubiquitous Computing,* 8 December, 16(8), pp. 973-985.

Schaub, F., Hipp, M., Kargl, F. & Weber, M., 2013. On credibility improvements for automotive navigation systems. *Personal and Ubiquitous Computing,* June, 17(5), pp. 803-813.

SimpleMotoring.co.uk, 2011. *Sat Nav Live Traffic Data – How Does It Work?.* [Online]
Available at: http://www.simplemotoring.co.uk/car-technology/sat-nav-live-traffic-how-does-it-work/#.VtxaVPmLTIU
[Accessed 6 March 2016].

Speake, J., 2015. ‘I’ve got my Sat Nav, it’s alright’: Users’ Attitudes towards, and Engagements with, Technologies of Navigation. *The Cartographic Journal,* 52(4), pp. 345-355.

TechRadar, 2010. *How your sat-nav works out the best route.* [Online]
Available at: http://www.techradar.com/news/car-tech/satnav/how-your-sat-nav-works-out-the-best-route-677682/2
[Accessed 2 March 2016].

United States Office of the Press Secretary, 2000. *Statement by the president regarding the United States' decision to stop degrading Global Positioning System accuracy.* [Online]
Available at: http://clinton4.nara.gov/WH/EOP/OSTP/html/0053_2.html
[Accessed 2 March 2016].

wiki.openstreetmap.org, 2016. *Routing/online routers.* [Online]
Available at: http://wiki.openstreetmap.org/wiki/Routing/online_routers
[Accessed 29 March 2016].

Wyse, S. E., 2011. *What is the Difference between Qualitative Research and Quantitative Research?.* [Online]
Available at: http://www.snapsurveys.com/blog/what-is-the-difference-between-qualitative-research-and-quantitative-research/
[Accessed 17 March 2016].

Bibliography

Android Developers, n.d. *<uses-sdk>.* [Online]
Available at: http://developer.android.com/intl/es/guide/topics/manifest/uses-sdk-element.html
[Accessed 28 April 2016].

Google Developer, 2016. *Distance Matrix Requests.* [Online]
Available at: https://developers.google.com/maps/documentation/distance-matrix/intro#DistanceMatrixRequests
[Accessed 1 April 2016].

Looije, R., te Brake, G. M. & Neerincx, M. A., 2007. *Usability engineering for mobile maps.* New York, Proceedings of the 4th international conference on mobile technology, applications, and systems and the 1st international symposium on Computer human interaction in mobile technology (Mobility '07).

Merriman, C., 2014. *A turn by turn history of GPS.* [Online]
Available at: http://www.theinquirer.net/inquirer/feature/2329336/a-turn-by-turn-history-of-gps
[Accessed 28 April 2016].

Parker, M., 2013. *Sat nav vs Maps.* [Online]
Available at: http://www.theaa.com/newsroom/news-2013/satnav-vs-maps.html
[Accessed 28 April 2016].

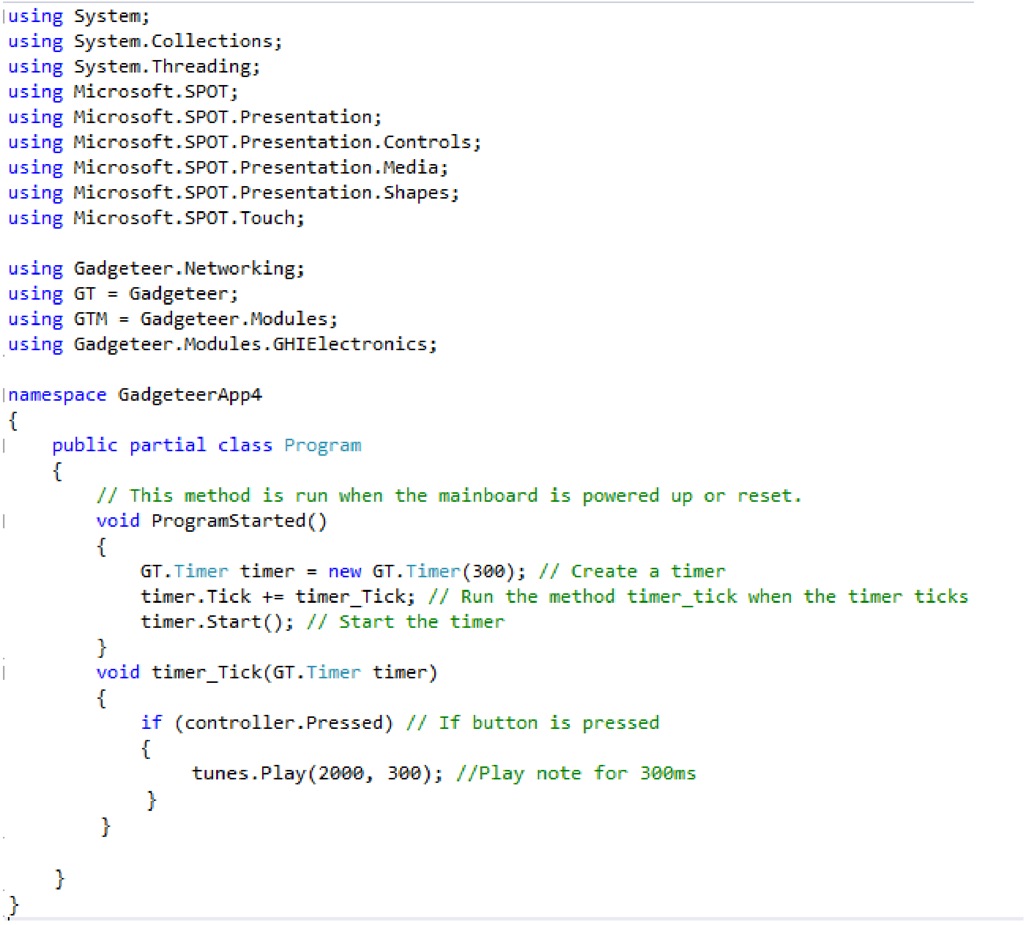
RavKrish, 2010. *History of Automotive navigation system.* [Online]
Available at: http://navigationnews.blogspot.co.uk/2010/07/history-of-automotive-navigation-system.html
[Accessed 28 April 2016].

The AA, 2011. *Sat nav safety.* [Online]
Available at: http://www.theaa.com/motoring_advice/safety/satnav-tentips.html
[Accessed 28 April 2016].

The Upcoming, 2014. *The history and theory of satellite navigation.* [Online]
Available at: http://www.theupcoming.co.uk/2014/08/14/the-history-and-theory-of-satellite-navigation/
[Accessed 28 April 2016].

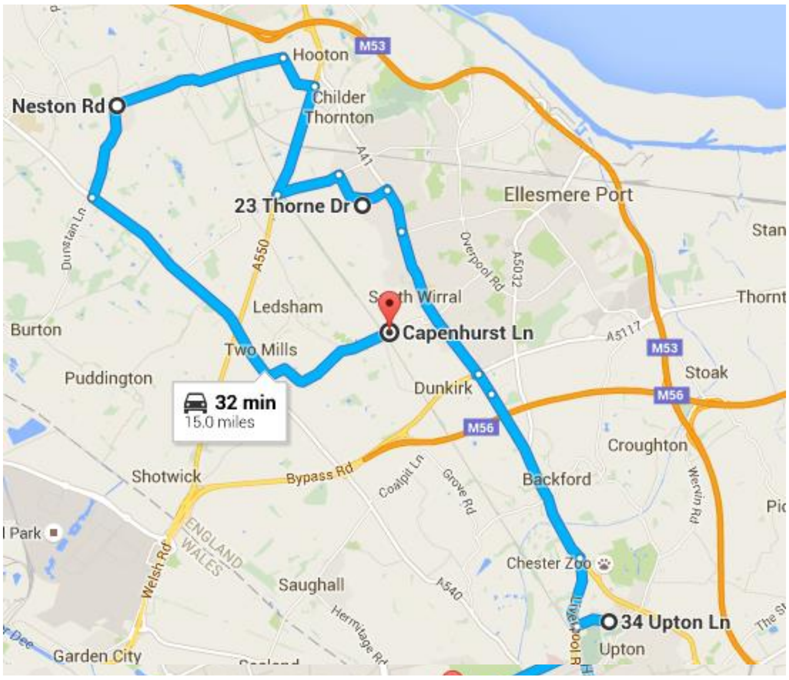
Appendices

Item 1: Gadgeteer prototype code

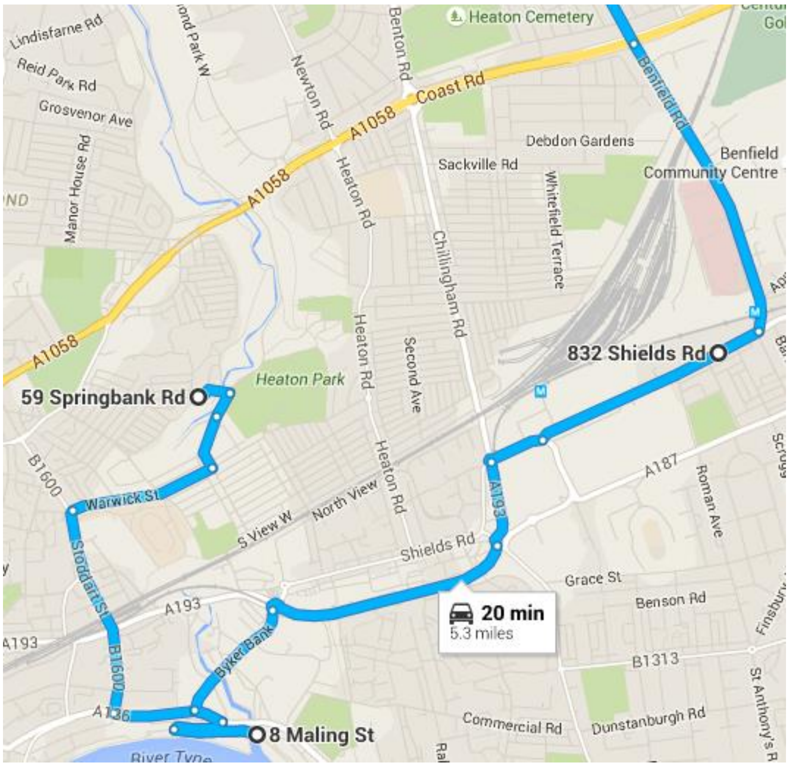


```
using System;
using System.Collections;
using System.Threading;
using Microsoft.SPOT;
using Microsoft.SPOT.Presentation;
using Microsoft.SPOT.Presentation.Controls;
using Microsoft.SPOT.Presentation.Media;
using Microsoft.SPOT.Presentation.Shapes;
using Microsoft.SPOT.Touch;

using Gadgeteer.Networking;
using GT = Gadgeteer;
using GTM = Gadgeteer.Modules;
using Gadgeteer.Modules.GHIElectronics;

namespace GadgeteerApp4
{
    public partial class Program
    {
        // This method is run when the mainboard is powered up or reset.
        void ProgramStarted()
        {
            GT.Timer timer = new GT.Timer(300); // Create a timer
            timer.Tick += timer_Tick; // Run the method timer_tick when the timer ticks
            timer.Start(); // Start the timer
        }
        void timer_Tick(GT.Timer timer)
        {
            if (controller.Pressed) // If button is pressed
            {
                tunes.Play(2000, 300); //Play note for 300ms
             }
         }

    }
}
```

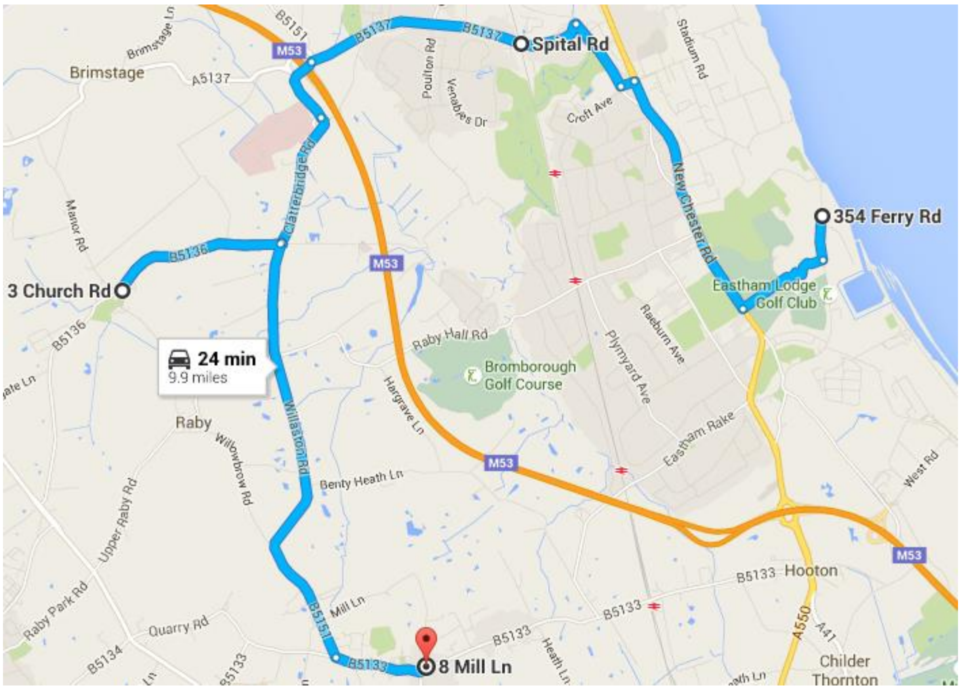
Item 2: Initial user testing routes



Volunteer 1



Volunteer 2


Spital Rd
354 Ferry Rd
3 Church Rd
24 min
9.9 miles
8 Mill Ln
Brimstage
Raby
Bromborough Golf Course
Eastham Lodge Golf Club
Hooton
Childer Thornton
M53
B5137
B5151
B5136
B5133
A5137
Clatterbridge Rd
Willaston Rd
New Chester Rd
Raby Hall Rd
Benty Heath Ln
Eastham Rake
Plymyard Ave
Raeburn Ave
Croft Ave
Venables Dr
Poulton Rd
Stadium Rd
Hargrave Ln
Willowbrow Rd
Upper Raby Rd
Raby Park Rd
Quarry Rd
Manor Rd
Mill Ln
Heath Ln
West Rd
A550
A41
B5134
Brimstage Ln

Volunteer 3

Item 3: Initial user testing transcripts

Volunteer 1

Interviewer: So could you briefly describe the experience you had using the navigation system?

Volunteer: Firstly, we took the navigation system and we used it to get to a location that I already knew how to get there, that was all fine. Then we went to a place where I knew generally where it was but not the exact place we were going to.

I: And how did you find that?

V: I found it was actually pretty good to use, you could sort of use your own common sense to work out the place we were going to would be on a main road, so the beep sort of helped me work out how far along the road I would be going to get to that place.

I: And were there any difficulties or frustrations that you found while we used it?

V: Not personally, the only thing I could think of that might be difficult is figuring out when to take turnings, finding the location that way.

I: Ok. And if you were to use the system in a day-to-day environment, could you think of any features you would like adding or any ways it could be improved?

V: There could be a signal that was slightly different to indicate when you're going further away, that way you don't have to rely on the beeps getting further apart, you could just hear a different signal that says 'you're going the wrong way'.

I: Ok. Is there anything else you would like to add?

V: That's all.

I: Thanks for your time.

Volunteer 2

Interviewer: Hi. Could you please describe how you found the experience of using the navigation system?

Volunteer: Yeah sure, so I thought that overall it worked pretty well. For the places I didn't know, I think the beeping of the system helped me know when I was getting closer to the place, so I could start looking for signs and stuff to find it. The places we went to were both signposted when we were close so I could find them that way but it helped me know I was still on route before that.

I: OK, and were there any difficulties you found?

V: I've never really driven around the Quayside before so not having proper instructions around there was a bit scary, I think if I'd have gotten lost there it could have been a bit stressful.

I: Yeah, I know what you mean.

V: I think if there was a way to turn the commands on if you're really stuck, that could bring a bit more peace of mind in a situation like that. You don't want to pull over and check your phone again in the town centre or whatever.

I: Yeah. And if you were going to use the system in day-to-day life, is there any other features you would like to see added?

V: Maybe some way of indicating that you're going off route so that you can start trying to work out where you went wrong and how to fix it maybe.

I: Is there anything else you would like to add?

V: Nope, I think that's it.

I: Thanks for your time.

Volunteer 3

Interviewer: Could you please describe your experience with the prototype system?

Volunteer: Well first we went out and drove to the train station to get me used to the SatNav, which was fine because I knew where that was already. Then after that we went out to Thornton Hough to find a pub.

I: How did you find working with the system when you didn't know the destination beforehand?

V: Again there were no problems really, I knew it was off the main road when you go into the village because I looked it up, so the noise showed me that I was going in the right direction once I got there, and helped me know when I was close to it.

I Ok, and what about the third journey, how did that go?

V: Yeah, we found it in the end didn't we? I took the wrong turning and ended up in Raby, which I realised when I had done it, but the beeping slowed down enough to show me that I wasn't making progress. There were signs for Willaston though so once we were back on track it was easy to get to, and the SatNav was helpful again when we got into the village.

I: So apart from ending up in Raby, were there any other difficulties or frustrations you had when we were out?

V: Not really, I guess on the first bit once I had got used to the system and I was getting near the station, the beeping got a bit annoying because I knew I wasn't going to get lost at that point, I knew where I was going.

I: So would you like to have had the option to turn the sound off at that point?

V: Yeah, maybe like a button on the app to mute it at times. I think if I was on a longer journey too, like on the motorway, having a button to mute it would come in handy.

I: Is there anything else you could think of that could be added?

V: Nothing I can think of no.

I: Ok, that’s everything, thanks for your time.

Item 4: Full Requirements

Functional Requirements

FR1 – Route Finding

FR1.1 – User must be able to type in the destination they require to search for it

FR1.2 – Application must be able to find the destination entered by the user

FR1.3 – Application must audibly indicate when the user gets closer to the destination

FR1.4 – Application must audibly indicate when the user gets further away from the destination

FR1.5 – When the user arrives at their destination they should be alerted by the application

FR1.6 – The application must be flexible when the user strays away from the optimal route

FR1.7 – The application must not give any explicit or turn-by-turn instructions of any kind

FR2 – Interface

FR2.1 – When the user's input destination is ambiguous, a list of potential destinations must be displayed for them to select from

FR2.2 – The application should display the user's desired destination during the journey

FR2.3 – The audio feedback on the application must be able to be muted and unmuted during the journey and the button to do so must be large enough to be pressed safely while driving

FR2.4 – The audio feedback should be adjustable in volume and must be an appropriate volume for an automotive environment

FR3 – Documentation

FR3.1 – Any code written should be clearly commented so that future modifications can be made easily

Non-Functional Requirements

NFR1 – Front End

NFR1.1 – The interface should be simple to use and free from clutter

NFR2 – Back-End

NFR2.1 – The application must run on Android systems and should target as wide a range of devices as possible

NFR2.2 – Distance and routing calculations should be handled by external APIs

NFR2.3 – The application should use free, open source APIs

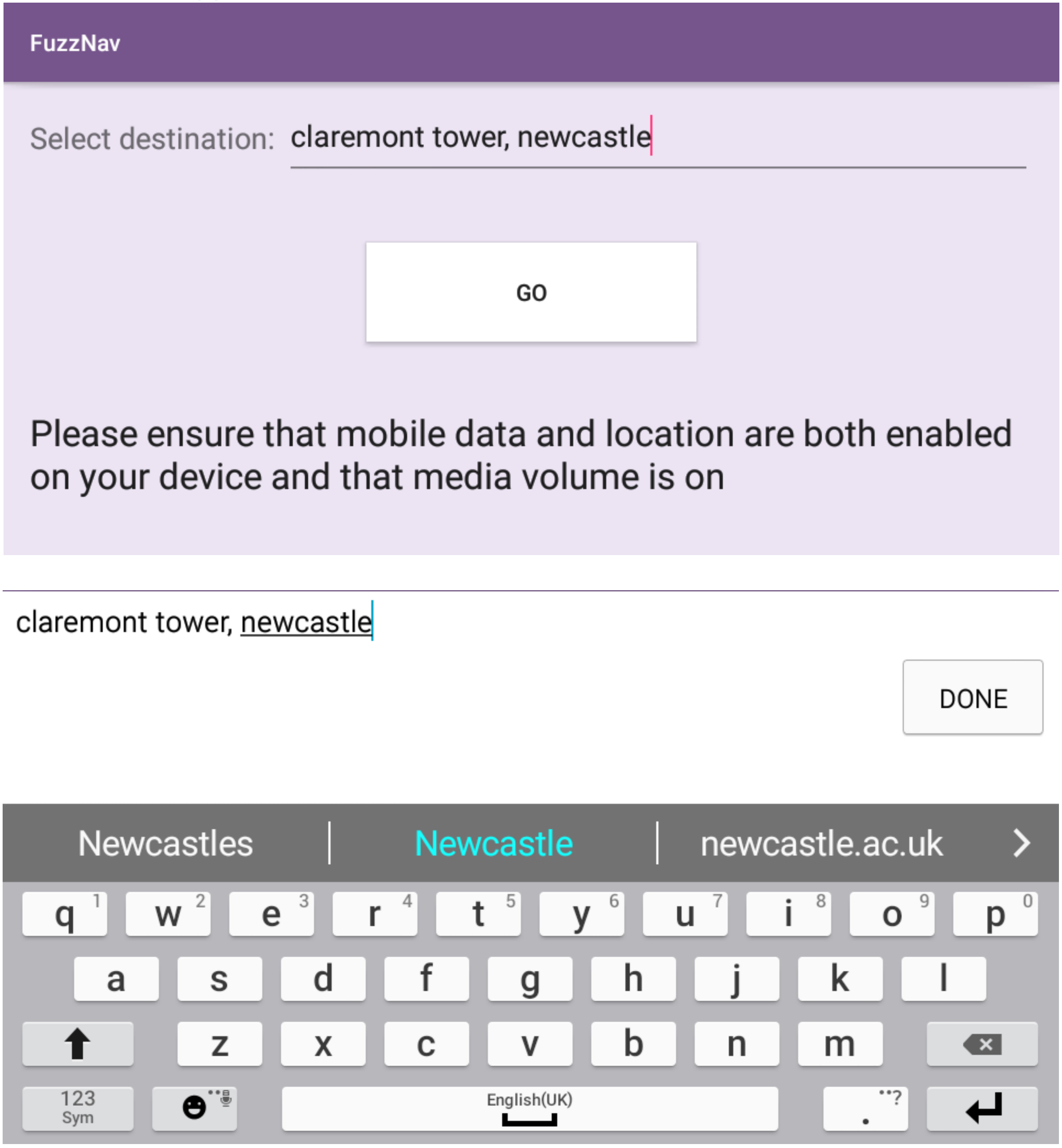
NFR2.4 – The application should require permission to use the users fine and coarse locations, as well as their internet connection

NFR2.5 – The application must require the use of the systems internet and location services

NFR2.6 – The application shouldn't need to store any user information apart from their current location

NFR2.7 – All written code must be set out in a way that is easy to read

Item 5: Final app screenshots

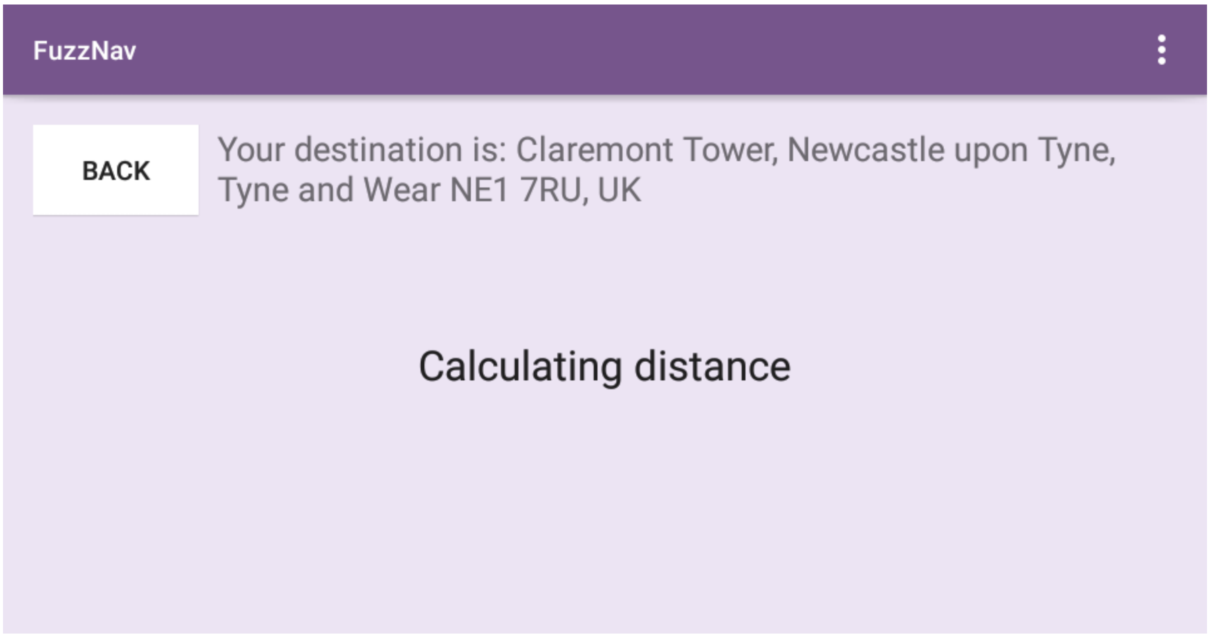


FuzzNav

Select destination: claremont tower, newcastle

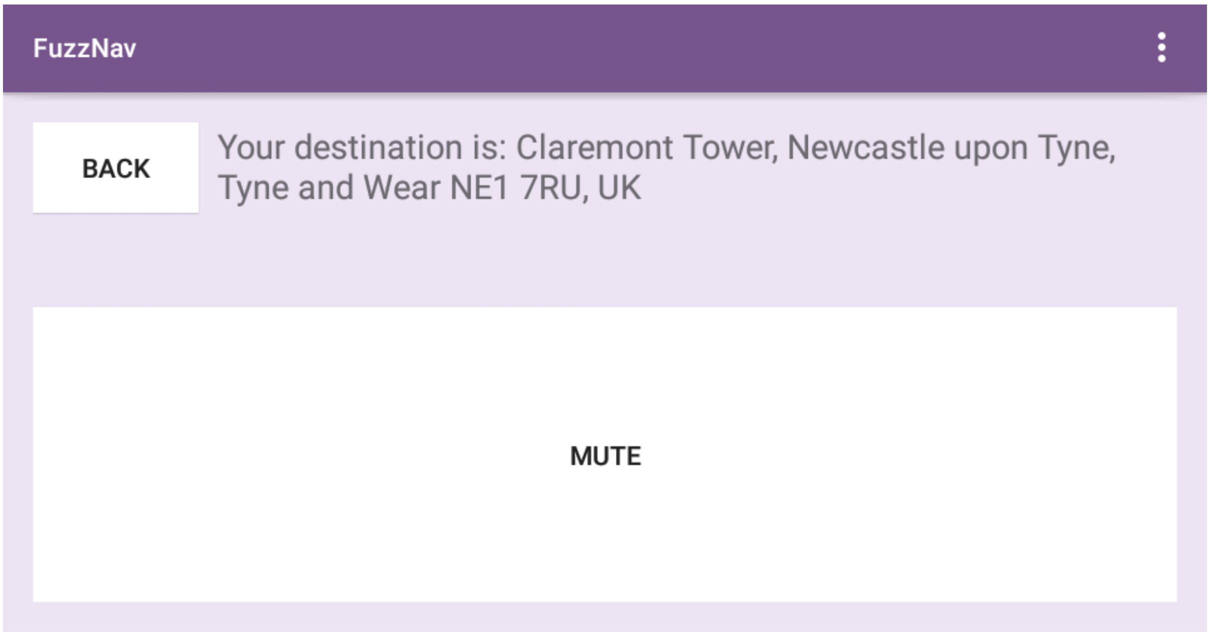
Please ensure that mobile data and location are both enabled on your device and that media volume is on

claremont tower, newcastle

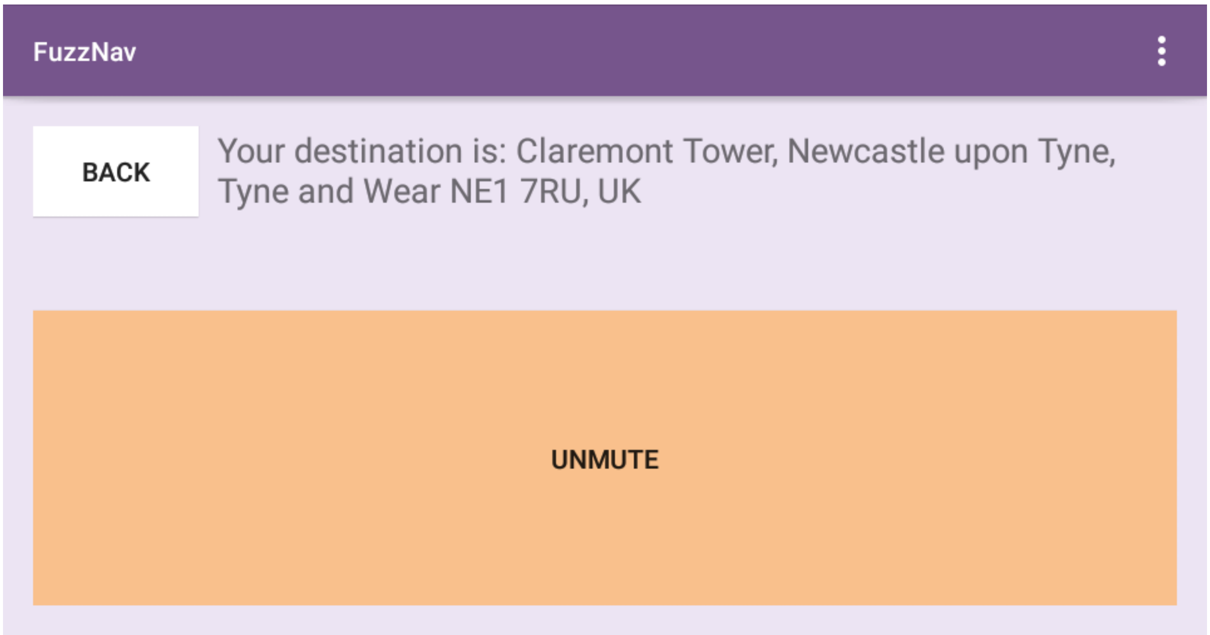


Your destination is: Claremont Tower, Newcastle upon Tyne, Tyne and Wear NE1 7RU, UK

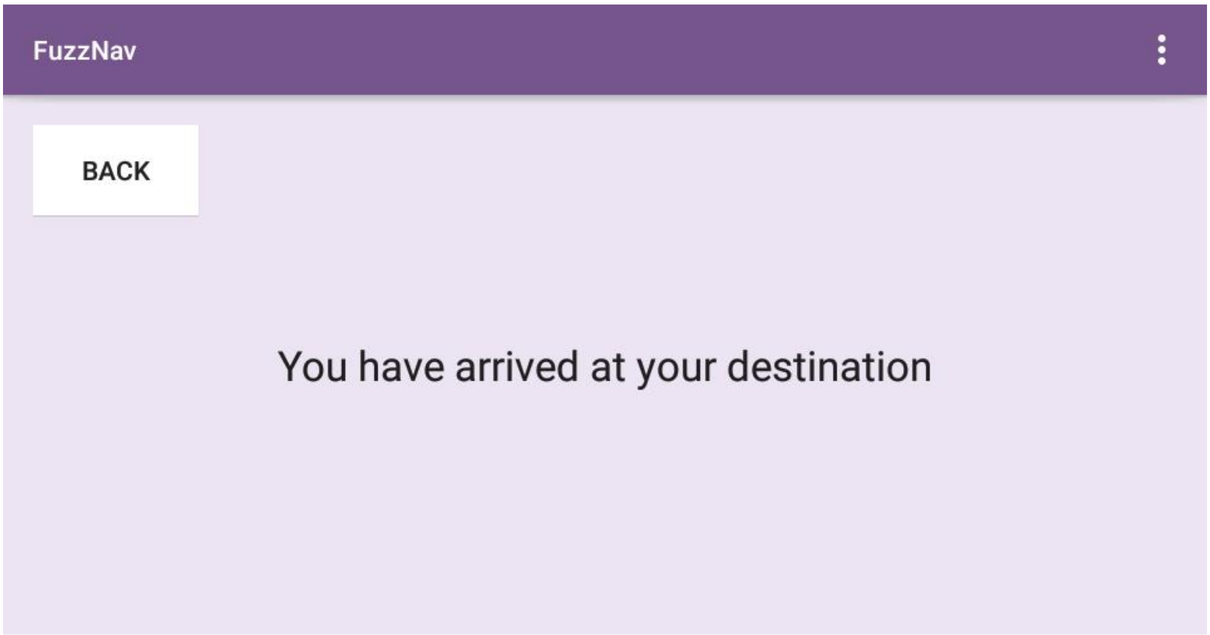
Calculating distance



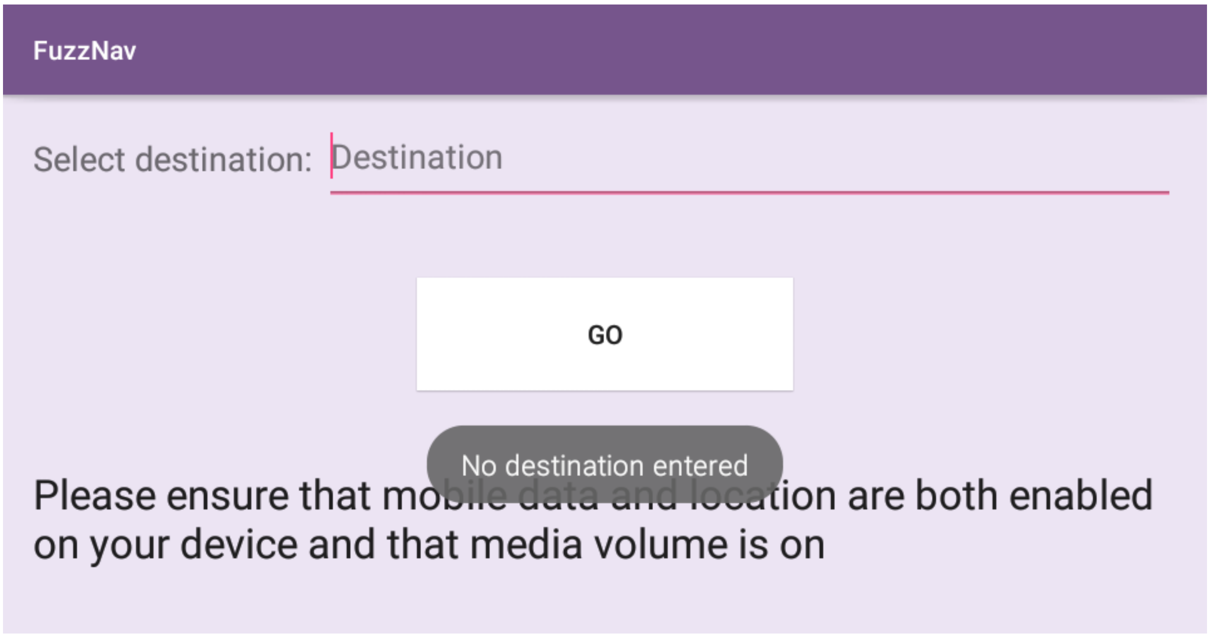
Your destination is: Claremont Tower, Newcastle upon Tyne, Tyne and Wear NE1 7RU, UK



Your destination is: Claremont Tower, Newcastle upon Tyne, Tyne and Wear NE1 7RU, UK

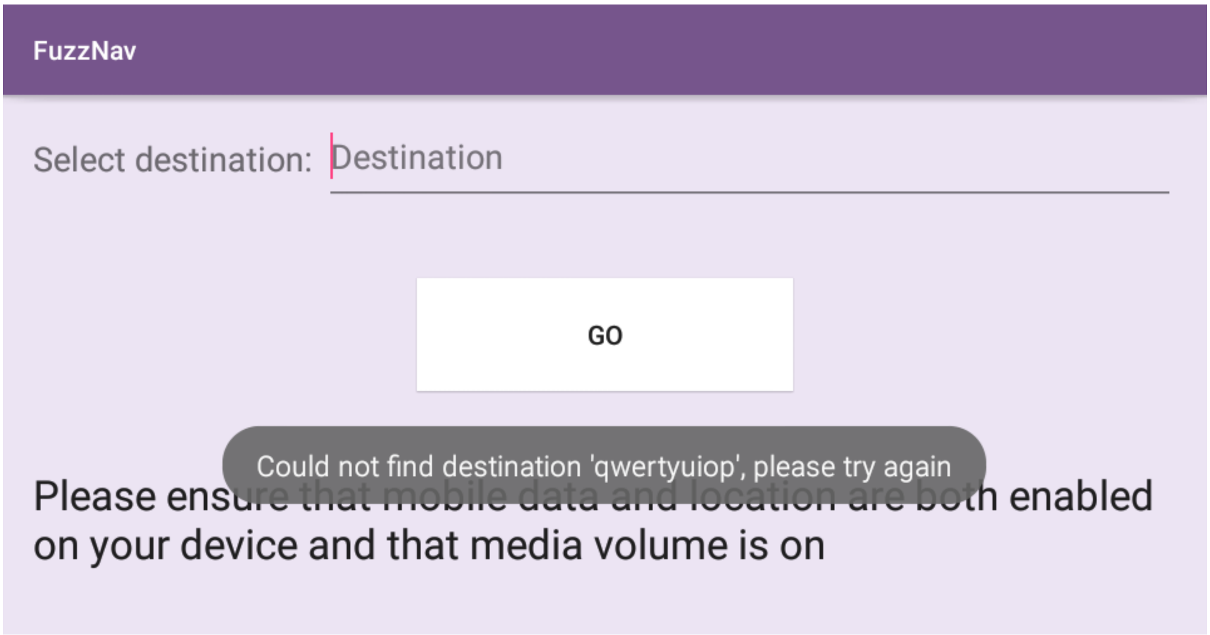


You have arrived at your destination



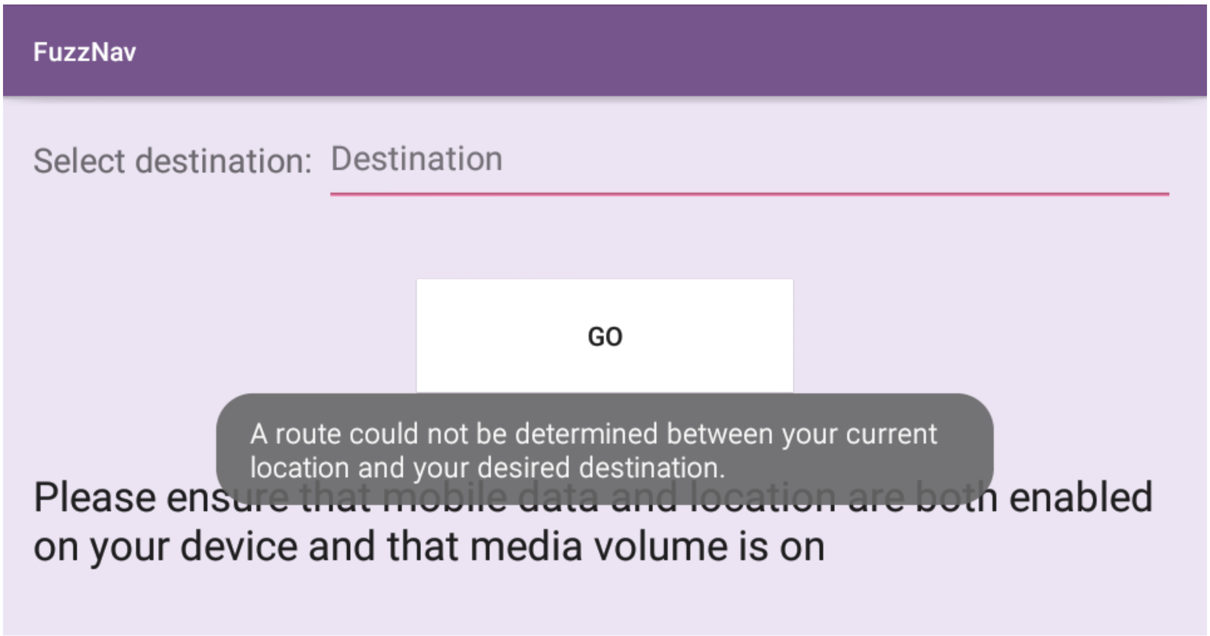
No destination entered

Please ensure that mobile data and location are both enabled on your device and that media volume is on



Could not find destination 'qwertyuiop', please try again

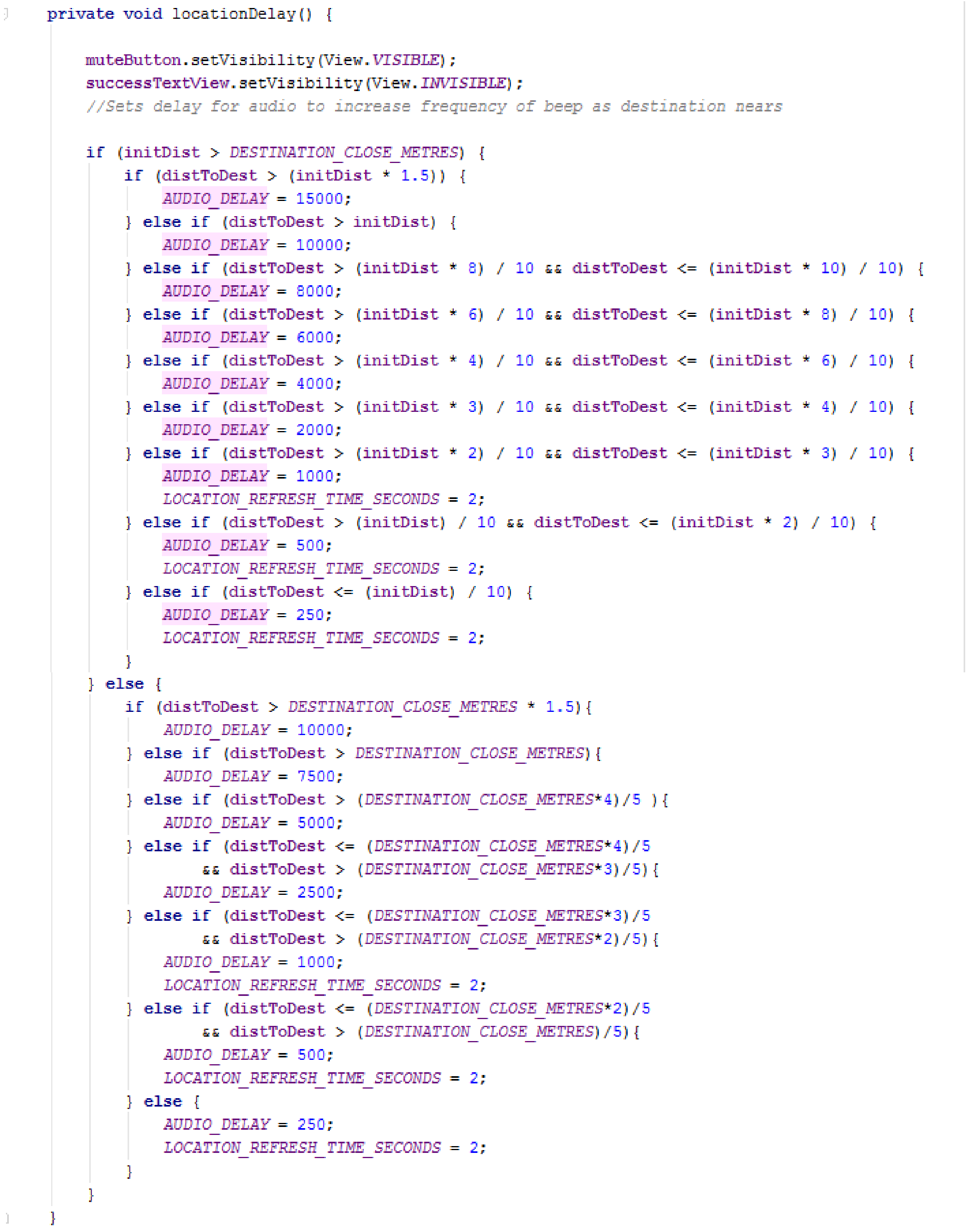
Please ensure that mobile data and location are both enabled on your device and that media volume is on



A route could not be determined between your current location and your desired destination.

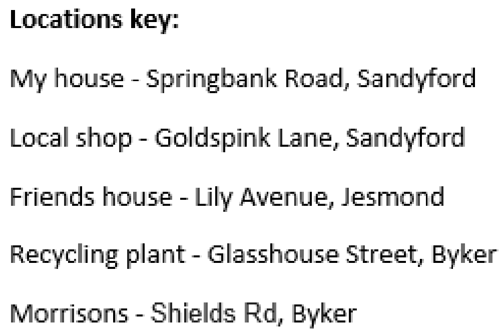
Please ensure that mobile data and location are both enabled on your device and that media volume is on

Item 6: Full locationDelay method



```java
    private void locationDelay() {

        muteButton.setVisibility(View.VISIBLE);
        successTextView.setVisibility(View.INVISIBLE);
        //Sets delay for audio to increase frequency of beep as destination nears

        if (initDist > DESTINATION_CLOSE_METRES) {
            if (distToDest > (initDist * 1.5)) {
                AUDIO_DELAY = 15000;
            } else if (distToDest > initDist) {
                AUDIO_DELAY = 10000;
            } else if (distToDest > (initDist * 8) / 10 && distToDest <= (initDist * 10) / 10) {
                AUDIO_DELAY = 8000;
            } else if (distToDest > (initDist * 6) / 10 && distToDest <= (initDist * 8) / 10) {
                AUDIO_DELAY = 6000;
            } else if (distToDest > (initDist * 4) / 10 && distToDest <= (initDist * 6) / 10) {
                AUDIO_DELAY = 4000;
            } else if (distToDest > (initDist * 3) / 10 && distToDest <= (initDist * 4) / 10) {
                AUDIO_DELAY = 2000;
            } else if (distToDest > (initDist * 2) / 10 && distToDest <= (initDist * 3) / 10) {
                AUDIO_DELAY = 1000;
                LOCATION_REFRESH_TIME_SECONDS = 2;
            } else if (distToDest > (initDist) / 10 && distToDest <= (initDist * 2) / 10) {
                AUDIO_DELAY = 500;
                LOCATION_REFRESH_TIME_SECONDS = 2;
            } else if (distToDest <= (initDist) / 10) {
                AUDIO_DELAY = 250;
                LOCATION_REFRESH_TIME_SECONDS = 2;
            }
        } else {
            if (distToDest > DESTINATION_CLOSE_METRES * 1.5){
                AUDIO_DELAY = 10000;
            } else if (distToDest > DESTINATION_CLOSE_METRES){
                AUDIO_DELAY = 7500;
            } else if (distToDest > (DESTINATION_CLOSE_METRES*4)/5 ){
                AUDIO_DELAY = 5000;
            } else if (distToDest <= (DESTINATION_CLOSE_METRES*4)/5
                    && distToDest > (DESTINATION_CLOSE_METRES*3)/5){
                AUDIO_DELAY = 2500;
            } else if (distToDest <= (DESTINATION_CLOSE_METRES*3)/5
                    && distToDest > (DESTINATION_CLOSE_METRES*2)/5){
                AUDIO_DELAY = 1000;
                LOCATION_REFRESH_TIME_SECONDS = 2;
            } else if (distToDest <= (DESTINATION_CLOSE_METRES*2)/5
                    && distToDest > (DESTINATION_CLOSE_METRES)/5){
                AUDIO_DELAY = 500;
                LOCATION_REFRESH_TIME_SECONDS = 2;
            } else {
                AUDIO_DELAY = 250;
                LOCATION_REFRESH_TIME_SECONDS = 2;
            }
        }
    }
```

Item 7: Version testing



Locations key:

My house - Springbank Road, Sandyford

Local shop - Goldspink Lane, Sandyford

Friends house - Lily Avenue, Jesmond

Recycling plant - Glasshouse Street, Byker

Morrisons - Shields Rd, Byker

Date	13/02/2016
Version	0.05
Description	Walked from my house to the local shop and back again.
Findings	As I moved, the current location coordinates changed, showing that the app was tracking my change in location.
Changes made	Implemented Destination coordinates to begin calculating distances.

Date	13/02/2016
Version	0.1
Description	Walked from my house to the local shop and back again.
Findings	Location was being updated regularly throughout the journey. Output on UI of app showed that the ‘current position’ coordinates were updated every 10 seconds as expected while destination coordinates remained stationary throughout.
Changes made	Location portion of the implementation is now complete. Began creating call to Distance Matrix API.

Date	15/02/2016
Version	0.2
Description	Walked from my house to the local shop. Before getting to the shop, I turned around and came back home.
Findings	Distance to destination started at roughly 230m and periodically updated to decrease as I got closer to the shop. After I turned around, it started periodically increasing again as I neared my house.
Changes made	Distance section now complete. Began linking distance to audio feedback.

Date	18/02/2016
Version	0.4
Description	Walked from my house to the local shop, approximately 230m. Before getting to the shop, I turned around and came back home.
Findings	The audio track was set so that anywhere less than 300m away was considered close, leading to the locationClose() method being invoked. The delay was set appropriately at each section of the walk, and the audio sped up as I got closer and slowed down as I got further away as intended. On two occasions (once on the way there, once on the way back), the audio cut out before returning at the same rate as before. I believe this to be because the distance to the location could not be determined at

	that point, possibly due to loss of communication with the satellite. When I checked the display after the second occurrence, it stated that the distance was being calculated. Finally, the audio track was very quiet even at full volume, although this may be because it was being played through headphones rather than through the sound system of a vehicle.
Changes made	Set checks in handleNewLocation that check if distToDest is set to 0. If this is the case it is assumed that satellite connection is lost and this is ignored. prevDest is also only set if distToDest is not equal to 0, meaning the previously calculated distance is always stored. This stops the audio track from cutting out at any point, even if the connection is lost.

Date	26/02/2016
Version	0.5
Description	Drove from my house to a friend’s house and then to the recycling plant.
Findings	Overall the app worked fairly well across the whole journey. The segue between locationClose and locationFar worked smoothly as I got within a kilometre of the destination. At the top of my road, I stopped at the junction while pulling out. However, the app carried on beeping despite me not making any progress. This showed me that the locationFar method is not properly working for when the driver isn’t making any progress. I believe it could be to do with the storing of the previous distance value and comparing it to establish whether progress is being made. I plan on looking into whether there is another way to ascertain if the phone is moving or not. Another problem was that the beeping seemed to be delayed slightly, it only hit the fastest level of beeping as I was very near the destination (closer than the 200m it should have been) and as I drove past it the fastest beeping carried on until I was further away from the destination than I expected. I suspect this is due to the delay in getting the distance information from the API, the system is not as real-time as I would like, and is something I will have to compensate for.
Changes made	Added in a tolerance value for locationing (initially 5 metres) to allow for differences in location when no progress is being made. Changed values in locationClose to give more time for audio to be beeping fastest to allow for the delay when nearing a destination.

Date	03/03/2016
Version	0.55
Description	Drove from my house to the recycling plant and back.
Findings	Changes that were made did not make any difference. On the outbound journey, I stopped at some traffic lights and again the audio continued to play despite the fact that I was stationary and the previous distance was the same as the current distance (tolerance value taken into account). As I approached the tip, I missed the turning without realising it and started to go away from my destination. Using the audio feedback from the app, I realised this and was able to turn at a roundabout to come back to the destination. As I got home, I deliberately drove past my house to check whether the changes to locationClose had made a difference, however I found this

	not to be the case. I therefore decided to instead shorten the length of time between location updates to see if this would make any difference instead. Also as I approached home, the phone screen went off and another audio track started playing at the same time as the first. I couldn't get it to stop without terminating the whole app, which is something that needs fixing.
Changes made	Changed LOCATION_REFRESH_TIME_SECONDS to 2 when the location is close to avoid delay in feedback. Added startAudio and stopAudio to onPause and onResume methods to pause beeping when screen is closed.

Date	03/03/2016
Version	0.55
Description	Drove from my house to Morrisons and back.
Findings	The updated refresh time seemed to give less lag as I approached the destination, with the beeping hitting its fastest speed as I was around the corner from the supermarket and fading back down as I entered the car park (presumably the coordinates for the store were set to the front entrance, hence why the car park was seen as further away). The beeping seemed to be better at pausing when I got to lights/junctions, although more testing will be required to check that this is the case. I realised I needed to implement some kind of end point when the car is within a certain distance from the destination to let the user know that the journey has ended. On the way back, I typed in my home address, however as I left the car park, another audio thread began so the feedback became out of time and incorrect. I will need to fix this, which is where the idea of implementing the end point came from as hopefully this will fix the issue. I will also look into ways of making it so that only one audio thread can run at a time. Also, on the drive there, I forgot to turn on my mobile data, meaning the app didn't work until I pulled over and fixed it. I will therefore implement a notification message on the initial page to remind the user to turn on their location and data, and could look into reminding them only if it is not already turned on.
Changes made	Created a method to be called when the system gets within 10 metres of the destination that stops the audio and displays a success message to the user. It also disconnects from location services and from the API client. Added in prompt for user to enable their location and mobile data before continuing on Destination page.

Date	09/03/16
Version	0.55
Description	Drove from my house towards Morrisons but intentionally made a wrong turning half way through and came back.
Findings	After I made the wrong turning I was hoping that the beeping would stop as I was going away from my destination. However, multiple audio threads were started and carried on as I drove away.
Changes made	Removed entire locationFar method and created locationDelay to handle entire journey instead. Set a variable on first call to API holding

	initial distance to destination. Set locationDelay to make audio frequency increase as percentage of this initial distance (similar to locationClose).

Date	09/03/16
Version	0.6
Description	Drove from my house to Morrisons and back.
Findings	System worked much better without locationFar. Audio sped up as I got closer to Morrisons, then went off as I arrived at my destination. On the way back I programmed the destination to be Byker Metro station and deliberately drove away from it. The audio kept to the longest delay once I got further away than the initial distance. Will need to implement audio levels for further away than initial distance (if driver starts off by going the wrong way). Possibly too many (12) audio levels at the moment, hard to tell difference in speed between them at some points towards the start.
Changes made	Removed some audio levels from locationDelay so now there is only 7 (plus 2 for when the driver is further away than the initial distance). Implemented the old locationClose method for if the initial destination is less than 1km away.

Date	09/03/16
Version	0.7
Description	Drove from my house to a friend’s house and back again.
Findings	System is working correctly now. Audio sounds more spaced out now that I have removed some levels, makes it a bit easier to know that you’re going in the right direction. System stopped when I got to the destination as expected. No problems.
Changes made	Began adding in final extra features such as the mute button.

Date	12/03/16
Version	0.8
Description	Drove from my house Morrisons and back. When nearing Morrisons, I pressed the mute button to pause the audio and kept it paused until the journey was over. On the way back I muted and then unmuted it again during the journey.
Findings	Completed journey as expected. The mute button paused and played the audio as intended. Found it difficult to tell while driving which state the button was in, so will need to create some obvious way of showing the state of the button.
Changes made	Audio section of implementation complete and all features created. Began work on the layout of the app, including removing all test data fields and implementing colour scheme. Also implemented colour scheme for mute button to show when it is checked.

Date	17/03/16
Version	1.0
Description	Drove from Jesmond Metro station to Salisbury Gardens, Jesmond. Then drove to Sunbury Avenue, Jesmond. This route is the route I have planned for my final user testing.

Findings	Journey ran smoothly with no problems encountered. Application is complete, layout is suitable for real-world use and functionality works as expected.
Changes made	-

Item 8: Questionnaire

Questionnaire

(Please circle the best matching answer for each question)

1. **Age**

   17-21 22-26 27-31 32-36 37-41 42+

2. **Gender**

   Male Female Other Prefer not to answer

3. **Previous experience with turn-by-turn Sat Nav systems**

   None Use infrequently Use frequently

4. **Knowledge of roads in local area (Newcastle)**

   None Poor Good Excellent

5. **Out of the two systems you have used today, which of them:**

   a. **Was more enjoyable?**
   Turn-By-Turn FuzzNav No preference

   b. **Was quicker in getting from A to B?**
   Turn-By-Turn FuzzNav No preference

   c. **Required better navigation skills?**
   Turn-By-Turn FuzzNav No preference

   d. **Was easier to use?**
   Turn-By-Turn FuzzNav No preference

   e. **Was less stressful?**
   Turn-By-Turn FuzzNav No preference

   f. **Was better for exploring an area while driving?**
   Turn-By-Turn FuzzNav No preference

   g. **Was safer?**
   Turn-By-Turn FuzzNav No preference

   h. **Was better overall?**
   Turn-By-Turn FuzzNav No preference

Item 9: Questionnaire Responses

Volunteer 1

Questionnaire

(Please circle the best matching answer for each question)

1. **Age**

   [✓ 17-21] 22-26 27-31 32-36 37-41 42+

2. **Gender**

   [✓ Male] Female Other Prefer not to answer

3. **Previous experience with turn-by-turn Sat Nav systems**

   None Use infrequently [✓ Use frequently]

4. **Knowledge of roads in local area (Newcastle)**

   None Poor [✓ Good] Excellent

5. **Out of the two systems you have used today, which of them:**

   a. **Was more enjoyable?**
   [✓ Turn-By-Turn] FuzzNav No preference

   b. **Was quicker in getting from A to B?**
   [✓ Turn-By-Turn] FuzzNav No preference

   c. **Required better navigation skills?**
   Turn-By-Turn [✓ FuzzNav] No preference

   d. **Was easier to use?**
   [✓ Turn-By-Turn] FuzzNav No preference

   e. **Was less stressful?**
   [✓ Turn-By-Turn] FuzzNav No preference

   f. **Was better for exploring an area while driving?**
   Turn-By-Turn [✓ FuzzNav] No preference

   g. **Was safer?**
   [✓ Turn-By-Turn] FuzzNav No preference

   h. **Was better overall?**
   [✓ Turn-By-Turn] FuzzNav No preference

Volunteer 2

Questionnaire

(Please circle the best matching answer for each question)

1. **Age**

   [✓ 17-21] 22-26 27-31 32-36 37-41 42+

2. **Gender**

   [✓ Male] Female Other Prefer not to answer

3. **Previous experience with turn-by-turn Sat Nav systems**

   None Use infrequently [✓ Use frequently]

4. **Knowledge of roads in local area (Newcastle)**

   None Poor [✓ Good] Excellent

5. **Out of the two systems you have used today, which of them:**

   a. **Was more enjoyable?**
   Turn-By-Turn [✓ FuzzNav] No preference

   b. **Was quicker in getting from A to B?**
   [✓ Turn-By-Turn] FuzzNav No preference

   c. **Required better navigation skills?**
   Turn-By-Turn [✓ FuzzNav] No preference

   d. **Was easier to use?**
   [✓ Turn-By-Turn] FuzzNav No preference

   e. **Was less stressful?**
   [✓ Turn-By-Turn] FuzzNav No preference

   f. **Was better for exploring an area while driving?**
   Turn-By-Turn [✓ FuzzNav] No preference

   g. **Was safer?**
   [✓ Turn-By-Turn] FuzzNav No preference

   h. **Was better overall?**
   Turn-By-Turn [✓ FuzzNav] No preference

Volunteer 3

Questionnaire

(Please circle the best matching answer for each question)

1. **Age**

   [✓ 17-21] 22-26 27-31 32-36 37-41 42+

2. **Gender**

   Male [✓ Female] Other Prefer not to answer

3. **Previous experience with turn-by-turn Sat Nav systems**

   None [✓ Use infrequently] Use frequently

4. **Knowledge of roads in local area (Newcastle)**

   None Poor [✓ Good] Excellent

5. **Out of the two systems you have used today, which of them:**

   a. **Was more enjoyable?**
   Turn-By-Turn FuzzNav [✓ No preference]

   b. **Was quicker in getting from A to B?**
   [✓ Turn-By-Turn] FuzzNav No preference

   c. **Required better navigation skills?**
   Turn-By-Turn [✓ FuzzNav] No preference

   d. **Was easier to use?**
   [✓ Turn-By-Turn] FuzzNav No preference

   e. **Was less stressful?**
   [✓ Turn-By-Turn] FuzzNav No preference

   f. **Was better for exploring an area while driving?**
   Turn-By-Turn [✓ FuzzNav] No preference

   g. **Was safer?**
   [✓ Turn-By-Turn] FuzzNav No preference

   h. **Was better overall?**
   [✓ Turn-By-Turn] FuzzNav No preference

Volunteer 4

Questionnaire

(Please circle the best matching answer for each question)

1. **Age**

   [✓ 17-21] 22-26 27-31 32-36 37-41 42+

2. **Gender**

   [✓ Male] Female Other Prefer not to answer

3. **Previous experience with turn-by-turn Sat Nav systems**

   None [✓ Use infrequently] Use frequently

4. **Knowledge of roads in local area (Newcastle)**

   None Poor [✓ Good] Excellent

5. **Out of the two systems you have used today, which of them:**

   a. **Was more enjoyable?**
      [✓ Turn-By-Turn] FuzzNav No preference

   b. **Was quicker in getting from A to B?**
      Turn-By-Turn FuzzNav [✓ No preference]

   c. **Required better navigation skills?**
      Turn-By-Turn [✓ FuzzNav] No preference

   d. **Was easier to use?**
      [✓ Turn-By-Turn] FuzzNav No preference

   e. **Was less stressful?**
      [✓ Turn-By-Turn] FuzzNav No preference

   f. **Was better for exploring an area while driving?**
      Turn-By-Turn [✓ FuzzNav] No preference

   g. **Was safer?**
      Turn-By-Turn FuzzNav [✓ No preference]

   h. **Was better overall?**
      Turn-By-Turn FuzzNav [✓ No preference]

Volunteer 5

Questionnaire

(Please circle the best matching answer for each question)

1. **Age**

   17-21 [✓ 22-26] 27-31 32-36 37-41 42+

2. **Gender**

   [✓ Male] Female Other Prefer not to answer

3. **Previous experience with turn-by-turn Sat Nav systems**

   None [✓ Use infrequently] Use frequently

4. **Knowledge of roads in local area (Newcastle)**

   None Poor Good [✓ Excellent]

5. **Out of the two systems you have used today, which of them:**

   a. **Was more enjoyable?**
   Turn-By-Turn FuzzNav [✓ No preference]

   b. **Was quicker in getting from A to B?**
   Turn-By-Turn FuzzNav [✓ No preference]

   c. **Required better navigation skills?**
   Turn-By-Turn [✓ FuzzNav] No preference

   d. **Was easier to use?**
   Turn-By-Turn [✓ FuzzNav] No preference

   e. **Was less stressful?**
   [✓ Turn-By-Turn] FuzzNav No preference

   f. **Was better for exploring an area while driving?**
   Turn-By-Turn [✓ FuzzNav] No preference

   g. **Was safer?**
   [✓ Turn-By-Turn] FuzzNav No preference

   h. **Was better overall?**
   Turn-By-Turn [✓ FuzzNav] No preference

Volunteer 6

Questionnaire

(Please circle the best matching answer for each question)

1. **Age**

   [✓ 17-21] 22-26 27-31 32-36 37-41 42+

2. **Gender**

   Male [✓ Female] Other Prefer not to answer

3. **Previous experience with turn-by-turn Sat Nav systems**

   None Use infrequently [✓ Use frequently]

4. **Knowledge of roads in local area (Newcastle)**

   None [✓ Poor] Good Excellent

5. **Out of the two systems you have used today, which of them:**

   a. **Was more enjoyable?**
   [✓ Turn-By-Turn] FuzzNav No preference

   b. **Was quicker in getting from A to B?**
   [✓ Turn-By-Turn] FuzzNav No preference

   c. **Required better navigation skills?**
   Turn-By-Turn [✓ FuzzNav] No preference

   d. **Was easier to use?**
   [✓ Turn-By-Turn] FuzzNav No preference

   e. **Was less stressful?**
   [✓ Turn-By-Turn] FuzzNav No preference

   f. **Was better for exploring an area while driving?**
   Turn-By-Turn FuzzNav [✓ No preference]

   g. **Was safer?**
   [✓ Turn-By-Turn] FuzzNav No preference

   h. **Was better overall?**
   [✓ Turn-By-Turn] FuzzNav No preference

Item 10: Final user testing transcripts

Volunteer 1

Interviewer: So, could you briefly describe how you found the experience of using my navigation system?

Volunteer: I think it was a little bit more dangerous than the standard system. I felt a little bit lost and looking for street names and looking more around while I was driving rather than being able to focus on the roads. There was definitely an aspect of last minute turns and being slightly confused of where to go next, whether to go left or right and there was no indication on that.

I: Ok. Was there anything you liked about the system?

V: It's definitely much better if you want to just explore the area. I quite liked the fact I didn't have to stare at my phone all the time and focus on the screen

I: Ok, yeah. And was there any difficulties or frustrations apart from the ones that you have mentioned that you would like to add?

V: No, no, I think I was missing the aspect of looking at a map all the time and knowing where I am.

I: And is there anything else you would like to add, any other comments or anything?

V: No.

I: Alright, thanks for your time.

Volunteer 2

Interviewer: So could you briefly describe how you found the experience of using my Sat Nav, FuzzNav?

Volunteer: Er, I thought it was really good, it was more enjoyable at exploring the surrounding area, but I missed the turning which ended up in my route taking longer which was a bit stressful.

I: Alright. And what did you like about using the system?

V: Er, as I just said, it was really enjoyable, I quite liked finding areas by myself rather than using technology so it was more enjoyable that way, and it was quite fun.

I: Alright. Were there any difficulties or frustrations you found, apart from missing your turning?

V: Erm, not really, apart from missing the turning and kind of not knowing where you're going, that's probably the only part I'd say is bad.

I: Alright, and is there any other comments you would like to make or anything?

V: Erm, it's simple to use I think, I think it's more enjoyable overall, but it's probably easier to use a turn-by-turn system.

I: Alright. Thanks for your time.

Volunteer 3

Interviewer: Okay, so could you briefly describe how you found the experience of using the implicit system?

Volunteer: Yeah, erm, I thought it seemed like a good idea when you were telling me about it beforehand, but after using the normal Sat Nav it was just a bit more difficult to use. I did like the idea of it though and I guess it did help me when we got near to the end to know how far away we were.

I: Alright, and was there anything else you enjoyed about using the system?

V: Er, I suppose if I was stuck trying to find somewhere and I was nearby, it'd be more rewarding to use your app to find somewhere rather than going on Google Maps or whatever.

I: And were there any other frustrations or difficulties you found with it?

V: No, it was just a bit nerve-wracking at times not knowing exactly where I was going, you know?

I: That's how people used to live before Sat Navs were invented I suppose.

V: Yeah true.

I: And is there any other comments you'd like to make?

V: Nothing I can think of.

I: Ok, thanks for your time.

Volunteer 4

I: So could you please describe how you found the experience of using my Sat Nav system?

V: Yeah sure. So I thought it was pretty straightforward to use, the app itself is really simple and there's not much on the screen to distract you while you're driving, there's no maps to be looking at or anything. I think I'd still prefer to use the turn-by-turn one for day-to-day driving but I did like using yours too.

I: Is there anything else you liked about the app?

V: It was almost like a bit of a game, like I knew pretty much where I was going so we got there pretty fast but I could imagine if you didn't know where you were going that it'd be fun to use to find where you were going.

I: Ok, and were there any difficulties you found, anything you didn't like about it?

V: Er, I think it was more difficult to use, it required a bit more skill like you said. I don't think I'd want to use it if I was stressed or whatever, I think I'd get a bit annoyed at it.

I: And is there anything else you'd like to add?

V: I think both the things were good, both were looking at it in different ways. I think the Sat Nav one would be better for people who want to be comfy and relaxed when they drive but yours is better for people who like a challenge I guess.

I: That’s great, thanks for your time.

Volunteer 5

Interviewer: Could you please describe how you found the experience of using my system?

Volunteer: Yeah, er, well I liked the way it guided you to the address, I liked how the experience was a bit different to using the Sat Nav, it was a nice change. I don’t really like using Sat Navs anyway, especially that one you gave me so it’s definitely more appealing to use yours.

I: Out of interest, what is it you don’t like about the turn-by-turn system?

V: The one you gave me was really difficult to program, I thought. I think I get a bit distracted by them when I use one, like looking at the screen and that.

I: Ok, and was there anything else in particular you liked about FuzzNav then?

V: I liked that when we got by the Tesco’s, it started beeping faster so I knew to start looking for road signs. I think when that happened I knew I was close to the address so I could look out and that.

I: And were there any difficulties or frustrations you found when using it?

V: Er, not really because I kind of knew where I was going, I don’t think I needed it so much for most of the journey. I’m not sure how difficult it would be if I got really lost on the way, maybe then I’d rather have the first system.

I: Ok, and is there anything else you’d like to add?

V: No.

I: Alright, cheers for your time.

Volunteer 6

Interviewer: So can you please describe how you found the experience of using FuzzNav?

Volunteer: Of using yours?

I: Yeah.

V: Well I don’t think it was as easy as using the proper Sat Nav, for actually navigating places I don’t think I’d use it again. I think I felt quite lost at some parts, especially when we got near the end, I’ve never been near there before so I wasn’t sure where I was going.

I: Do you feel like you know it a bit better now you’ve navigated round there?

V: Yeah I suppose so.

I: So were there any parts you did like?

V: As you say, I think it maybe would help to learn places by driving through them, like landmarks and stuff.

I: And were there any more things you didn’t like, anything you found difficult?

V: Er, not really, just the being lost part, I don’t really like feeling lost when I’m driving.

I: Ok, and are there any other comments you’d like to make?

V: Not really.

I: Thanks for your time then.