

University of Sussex
School of Informatics and Engineering
BSc. Computer Science

An Android Application to Assist with Canine Exercise

Project Supervisor: Charlotte Robinson

William Moog-McNeill
204651
Submitted: 2022

Candidate Number: 204651

Statement of originality

'This report is submitted as part requirement for the degree of BSc Computer Science at the University of Sussex. It is the product of my own labour except where indicated in the text. The report may be freely copied and distributed provided the source is acknowledged. I hereby withhold permission for a copy of this report to be loaned out to students in future years.

Acknowledgements

I would like to thank my supervisor Dr Charlotte Robinson for her support throughout the development of my final thesis as well as the idea for the project. I would also like to thank my family and friends for their support throughout university especially in this last year. I don't know how I'd have done it without your support.

Table Of Contents

Statement of originality	1
Acknowledgements.....	1
Table Of Contents	2
Summary	4
Introduction and problem space	5
Objectives	5
Motivations	5
Problem Area	6
Research and related works.....	7
Market research.....	7
Strava	7
Pooper Snooper	7
Research into android.....	8
What is android?.....	8
Android architecture.....	8
Application Components.....	9
User Experience	9
Code of conduct and ethical considerations.....	11
BCS code of conduct	11
Ethical approval.....	13
Requirements.....	13
Mandatory	13
Desirable	13
Low Fidelity Wireframe.....	14
Novelty requirement.....	14
Project Timeline	14
Design.....	15
Client-side	15
Android studio	15
Java.....	15
XML	15
Server side.....	15
APIs.....	15
Firebase	15

Cloud Firestore.....	16
Firebase Authentication.....	16
Firebase Storage	17
Google Maps API and Google map related APIs	18
Implementation	19
Cloud Firestore Data and its integration with Firebase Authorisation	19
Activity and Fragment Interaction	20
Activities.....	21
Activity Lifecycle.....	21
Project Activities	22
Fragments	23
Maps Fragment	23
Paths Fragment	24
Contacts Fragment.....	25
Multithreading and asynchronous tasks.....	25
Services and Broadcasts.....	25
Backwards API compatibility.....	27
Testing.....	28
Android Emulator.....	28
Functional Requirements Testing	28
Functional Requirement Issues.....	34
Desirable requirements	35
User Testing	35
Desirable requirements unfinished due to time constraints.....	35
Social media desirable requirement	35
Future work and Improvements	35
Conclusions	36
References	38

Summary

The aim of this project is to develop an android application that assists dogwalkers. It will do so by showing the location dog bins on a map retrieved from government data and allowing users to record and share their path with other users. The report will describe the process in the development of the app thoroughly and give readers of all disciplines a good understanding of the app and how it functions.

Introduction and problem space

As of 2021, 59% of households in the UK owned a pet [1] with the most commonly owned pets being dogs and cats, both having an estimated population size of 12 million in the UK [2]. There has been a huge increase in pet owners from the year before, 2020, an increase of 18% [1] likely due to the pandemic. Dog owners have a lot of responsibilities in order to take care of their pets and one of these is walking them. An app for dog walking can help to make this an easier responsibility to deal with.

The average salary of dog walkers being well above minimum wage makes it difficult for people to hire dog walkers when issues arise, which may prevent them from walking their dog [3]. Roughly estimated, there are 2.14 million dogs living in households below the poverty line in the UK [4][5][6]. In the US people who are close to, but still above, the poverty line are left with \$367 at the end of the year after essentials [7]. Assuming this is similar in the UK, this means that it's very difficult for most families to hire a professional, making apps where dog walks can be easily shared important.

The aim of this project is to develop an app to address these issues. Apps for dog owners already exist and are popular, as shown by multiple dog walking, sitting and training apps already available on the android market, with plenty of these apps like Rover, DogHero and Tractive having over a million downloads. It is clear there is a demand for these types of apps.

Although there are similar apps on the market, the app presented here fills an additional need by pulling from government-provided data on council managed resources. This is a useful addition to dog walkers and will make my project an extension on the apps currently available on the market.

Objectives

The objective of this project is to design and implement an app that will be of use to dog walkers. The app will be written in java using android studio to run on android devices.

The app will be used to assist dog owners with walking their dogs by displaying the location of council managed dog bins on a map interface and using GPS tracking to make planning, taking, and sharing dog walks easier while also incorporating some social network features.

When the user opens the app, they will see a map screen. The map will show all of the dog bins located near the user. When the user wants to go for a walk they will click "record" and their location will be tracked until they press again to stop recording. They can look at the map while on the walk to see where dog bins are.

There will be a list of all recorded walks on another tab. If the user wants to go on a walk they have gone on before, they can press on the walk and the app will give them directions. These walks can also be shared between users. This will be useful for users who want to share walks with dog sitters or walkers when they are unable to show them the route in person.

Motivations

This project was suggested by my supervisor Dr Charlotte Robinson and I thought it would be an interesting and challenging project.

First and foremost, I found the idea of working on an android project interesting. I have no prior experience of Android app development, but I do have experience coding in Java, which is the official programming language for android app development [8]. I thought this would be a good way to hone my skills in java while expanding my abilities by taking on an entirely new challenge.

Secondly, the demand for android app developers is already high and increasing [9] so it would be good for my portfolio to have proof of being able to develop android apps.

The project is very closely related to my current course. This app takes aspects from both my Software Engineering and Human Computer interaction modules. It therefore makes sense for me to pursue this as a project.

I also like dogs having been a dog owner myself and developing an app which helps people who are in a difficult spot while helping to improve the quality of life of dogs seems like a worthwhile endeavour.

Problem Area

The main challenge of making this game was learning an entirely new platform and the difficulties that come with that. The consensus seems to be that it takes at least 3 months to become an android developer and can take up to 6 years. Developing an app seems to be agreed to take about 3 months to develop [10] which is in line with time allocated for a final project.

Research and related works

The background research is broken into 3 main parts: 1) Market research, for ideas in development and to better define the app 2) Understanding Android, the platform the project will be on and 3) User experience as it is what turns a functional app into a good app.

Market research

My app is essentially a combination of two things: path tracking apps such as those for biking and running and apps that have locations of dog bins.

Path tracking apps are quite popular. There are a lot of these on the market currently. One of the most popular types of path tracking apps available is run trackers. These apps are heavily used and highly rated by their users. This project will look at one of the most popular run tracking apps on the market to analyse its features: “Strava”.

Strava

Strava is an application that uses GPS location services to track your path while cycling, swimming and running. It allows you to record your runs, recording your location while you move, taking note of your path and time and saving it to the map so that you can see your path after the fact. When you finish a run, it gives you list of details to fill out like name, what type of activity it was and who can see the exercise. It tracks a variety of things like the pace, distance, time, speed at different points in the route and so on. It has heatmaps for popular runs and has a bunch of different challenges for the user to try to partake in. It allows users to share and receive paths.

This app’s features are useful when considering the development of my app in some ways but not in others. The layout is very sleek and easy to use which is good. It has a huge array of features. However, it had so many features that it seemed like it might be a little overwhelming to a first-time user.

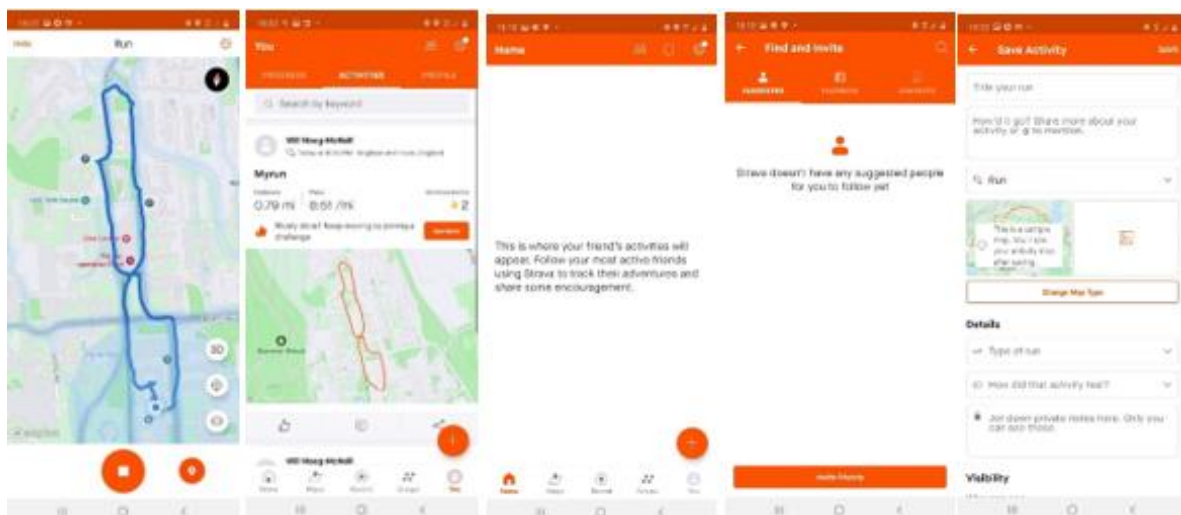


Figure 1: Screens in Strava application. From left to right the screens are: Strava path recording, the Recorded Path Screen, the Friend Path Screen, The Find Friends Screen and finally the Activity Naming Screen

Pooper Snooper

Dog bin apps are far less popular than running apps. The most popular one I could find came in at 1,000+ downloads compared to 10,000,000+ downloads that Strava recorded on the google play store. They are also less highly rated. These apps allow users to find where dog bins are on a map as mine

does. However, these apps locate dog bins through users recording where they are and users report that the experience is usually dropping their own pins on a map.

I looked at one of these apps named Pooper Snooper found a lot of issues. The app only showed dog bins when the user was within 500m therefore showing a blank map when I first entered it. This makes it useless for planning walks to take. It does not have any type of path recording, saving or sharing.

My approach will be different than both of these apps. The general aim of this app is merge these two concepts in a way that will be useful to dog walkers. There are a lot of features of the running app not required in my app as the needs of my user will be different. Things like tracking progress, pace at different points in the walk and challenges are unnecessary from the point of view of my target users. I didn't find any features in my research into dog bin apps that I would find useful to incorporate into my app.

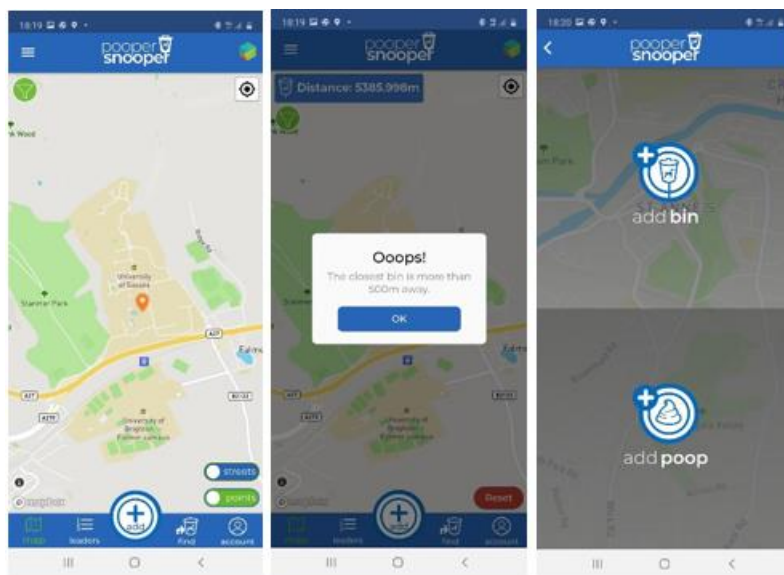


Figure 2: Screens in Pooper Snooper application. From left to right the screens are: the map screen, the find nearest bin screen and the add bins/poop screen.

Research into Android

What is Android?

Android is an open-source operating system used mainly for smartphones and tablets based on the Linux operating system. It was initially developed in 2003 by Android Inc., which was then bought by Google in 2005 and released in 2007. It is now the leading operating system in the world with over 70% of the global market share in June 2021 [11].

Android Architecture

Android has a layered architecture with a stack of software components. This stack consists of 4 layers and 5 components. The components are the Linux kernel, the platform libraries, the Android Runtime, the application framework and the applications themselves. [12][13][14][15]

Everything is built on the main component, the Linux kernel, which is the bottom layer. The Linux kernel handles all of the drivers, essentially allowing the operating system to communicate with individual devices within the phone to allow it to run. It is the direct link through drivers to things like Bluetooth, the camera and Wi-Fi. [12][13][14][15]

Platform libraries include the core libraries for the operating systems supported languages. All of these libraries have their own uses to help with different aspects of the operating system. For example, the SSL library is used for internet security and the SQLite library is used for data storage. This is in the second layer from the bottom. [12][13][14][15]

Sharing the second to bottom layer is the Android Runtime. This contains both the core libraries and Dalvik virtual machine. The Dalvik machine essentially manages memory and allows Android processes to be run simultaneously. The core libraries allow developers to code in the languages native to Android. [12][13][14][15]

The application framework is the second level from the top and provides services which are helpful for applications and simplify their development. Examples of these services would be the location manager which provides the location through GPS or the notifications manager which manages applications, notifications, and the display of these notifications to the user. [12][13][14][15]

Applications are the top layer and are what the user sees. These are standard applications that people know such as settings or camera. [12][13][14][15]

These fundamentals of Android architecture are important to know for my project because it defines the parts of the system that my app will interact with. Knowing that applications are on the top level and that it will be using location services through the application framework on the second layer has been useful when further learning about Android development and creating the application.

Application Components

Within the application are application components. There are four main components of the Android operating system: activities, services, broadcast receivers and content providers.

- Activities are essentially the individual screens which are shown to the user in app. They are both what the user sees (the layout) and how the user interacts with the application (the user interface). An application can have zero activities. [16][17][18]
- Services are essentially the background functions of an application, they update data sources, send notifications, and continue application functions when in other applications. [16][17][18]
- Broadcast receivers basically are used to catch messages from the system and other applications and reacts to them. [16][17][18]
- Content providers are how the application can send internal data to other applications when requested. [16][17][18]

Understanding the application components has been important as I needed to code using these terms and understanding their uses was essential for development of my app. Knowing how android components work was also helpful for the development of the app.

User Experience

There are a lot of components in making a good app. I will be looking at design from a user experience standpoint. When designing my app, the most important thing I need to take into account is usability, as not only is this app designed for all types of people to use but is important for people to be able to use it easily on the first try if it will be effective for people to use in an emergency. This means that all steps need to be easy to use. In the case of my app for this project, the most important features which needed to be designed for easy usability were onboarding, receiving a walk and taking a path.

There is a lot of information about how to make an app with high usability without there being an industry gold standard of the most important principles, so I will name here a few principles that I have incorporated into my design.

An app should have a design that is consistent throughout. This means that if you have a toggle of a certain design all toggles should have the same colour scheme and design. This allows the user to quickly identify the components of the app and increases how easy it is to learn how to use your app. [19]

An app should use symbols that are already recognizable to the user. You should use icons that are consistent with the operating system and that are easy to understand. We all know what a phone icon looks like or what a calculator looks like therefore, using designs similar to these will help the user to quickly recognize how your app works. [19]

An app should prioritize important content. Not all functions in an app are equally important or as commonly used; by making things important to the user more easily seen through size or colour you can improve the users' experience. These are just a few of the core principles of design that are important when trying to create a good user experience; but this gives an idea of the kind of principles used. [19]

Code of conduct and ethical considerations

Following the ethical guidelines is not only important to do for moral reasons but also is required when developing this project. There are two components to having a project that follows the ethics required. The project must adhere to the British computer society (BCS) code of conduct. The BCS is a charity within the UK “dedicated to lead IT industry through its ethical challenges, to support the people who work in the industry, and to make IT good for society.”[20] It must also follow the universities internal ethical process ensuring that it has had the correct level of ethical approval.

BCS code of conduct

The BCS code of conduct is a list of professional standards that all members must follow.

1. Public Interest
You shall:
 - a. have due regard for public health, privacy, security and wellbeing of others and the environment.
 - b. have due regard for the legitimate rights of Third Parties* .
 - c. conduct your professional activities without discrimination on the grounds of sex, sexual orientation, marital status, nationality, colour, race, ethnic origin, religion, age or disability, or of any other condition or requirement
 - d. promote equal access to the benefits of IT and seek to promote the inclusion of all sectors in society wherever opportunities arise.

My project has no ill effect on public health, privacy security or the wellbeing of others and the environment. It could even be said to have a positive effect on public health due to encouraging exercise and assisting to keep waste off the street.

My project will keep in mind the legitimate rights of third parties throughout and any assets used will be checked to ensure that their respective owners allow their use in this way. As this would be a free app there will be no monetary profit from assets used which will make this task easier.

There will be no discrimination on the grounds of any protected or unprotected condition. The app will not ask for any of these details and will be conscious to avoid indirectly affecting these groups.

The foundational idea of the project is to try to support lower class sectors of society. In addition, throughout development there will be a conscious effort to make the app as accessible as possible to everyone. The app will be free on the largest mobile operating system which makes it as accessible as possible while only being on a single platform.

1. Professional Competence and Integrity
You shall:
 - a. only undertake to do work or provide a service that is within your professional competence.
 - b. NOT claim any level of competence that you do not possess.
 - c. develop your professional knowledge, skills and competence on a continuing basis, maintaining awareness of technological developments, procedures, and standards that are relevant to your field.
 - d. ensure that you have the knowledge and understanding of Legislation* and that you comply with such Legislation, in carrying out your professional responsibilities.
 - e. respect and value alternative viewpoints and, seek, accept and offer honest criticisms of work.
 - f. avoid injuring others, their property, reputation, or employment by false or malicious or negligent action or inaction.
 - g. reject and will not make any offer of bribery or unethical inducement

My project was suggested by my supervisor therefore it is within the professional competence of someone at my level in university.

My project will develop my professional knowledge skills and competence as its scope goes beyond what I've learned within an attainable level.

My project will not break any legislations and I will check with my supervisor to ensure that this is the case throughout the project.

I will share the progress I have made as I develop my project and actively seek criticisms of my work by convening with my supervisor.

This project will not cause any injury to persons, property, reputation or employment, the scope of the project doesn't allow for the project to cause any injury.

I will not be bribed or allow any form of unethical inducement.

1. Duty to Relevant Authority

You shall:

- a. carry out your professional responsibilities with due care and diligence in accordance with the Relevant Authority's requirements whilst exercising your professional judgement at all times.
- b. seek to avoid any situation that may give rise to a conflict of interest between you and your Relevant Authority.
- c. accept professional responsibility for your work and for the work of colleagues who are defined in a given context as working under your supervision.
- d. NOT disclose or authorise to be disclosed, or use for personal gain or to benefit a Trustee Board Regulations Schedule 3 v7 – Code of Conduct for BCS Members Page 3 of 5 Reviewed by Trustee Board 9 June 2021 third party, confidential information except with the permission of your Relevant Authority, or as required by Legislation.
- e. NOT misrepresent or withhold information on the performance of products, systems or services (unless lawfully bound by a duty of confidentiality not to disclose such information), or take advantage of the lack of relevant knowledge or inexperience of others.

My app Design will follow all of the guidelines set out by my relevant authority. I will be following the universities guidelines for ethics and expected work while completing this project. I will also use my professional judgement throughout development.

I will seek to avoid any conflicts of interest between my project and the university. I will do this by following the ethical guidelines and by following all requirements set by the university

I accept full responsibility for the success of my project. There are no colleagues who are working under my supervision.

If by any chance I do come across information that is sensitive with relevance to a third party. I will check with my relevant authority the university before disclosing it while also staying in line with current legislation.

My project will not misrepresent or withhold any information about products, services or systems or take advantage of the lack of knowledge of others. I will give a full and accurate description of my project to the best of my ability.

1. Duty to the Profession

You shall:

- a. accept your personal duty to uphold the reputation of the profession and not take any action which could bring the profession into disrepute.
- b. seek to improve professional standards through participation in their development, use and enforcement.

- c. uphold the reputation and good standing of BCS, the Chartered Institute for IT.
- d. act with integrity and respect in your professional relationships with all members of BCS and with members of other professions with whom you work in a professional capacity.
- e. encourage and support fellow members in their professional development.

I accept my personal duty to uphold the reputation of the profession. I will not take any action which could bring the profession into disrepute.

I will follow all current professional standards.

I will not harm the reputation of the BCS.

I don't think I will interact with any other members of the BCS, if I did I would act with integrity as I will with those of all professions I will work with in a professional capacity.

I will do my best to support any other members of the BCS if the opportunity arises.

Ethical approval

The project did not involve any testing. The project does not expose anyone to harm; physical or psychological. The project would be able to take personal data entered by the user but will not as the app will not be put to market, meaning that there will be no actual users apart from myself. The research poses no threat to the environment or society. There aren't any other ethical issues that require further ethical review. For these reasons I did not need any ethics clearance from the university.

As the app will be self-contained and not released on the app market, there will be no users with which my project could raise ethical issues.

Requirements

Mandatory

1. The app shall work on the Android operating system.
2. The app shall have the ability to create users.
3. The app shall have the ability to store created users to allow for logging in.
4. The app shall have the ability to target the users location through location services.
5. The app shall have the ability to pin the locations of dog bins from government data.
6. The app shall have the ability to track the user and save the users path.
7. The app shall have the ability to save the users path to their account.
8. The app shall have the ability to find other users by username.
9. The app shall allow users to create a list of contacts.
10. The app shall allow sharing of the paths recorded by a user to other users.
11. The app shall allow users to follow paths in real time.

Desirable

1. The app should be easy to use.
2. The app should be aesthetically pleasing.
3. The app should allow users to have favourite routes.
4. The app should allow the user to share routes through social media platforms.
5. The app should incorporate multiple sets of government data.
6. The app shall have the ability to record the time it took the user to create the path.
7. The app should allow you to plan a trip before taking it.

Low Fidelity Wireframe

While trying to define my design requirements I created a low fidelity wireframe to get a clearer idea of how my app would function.

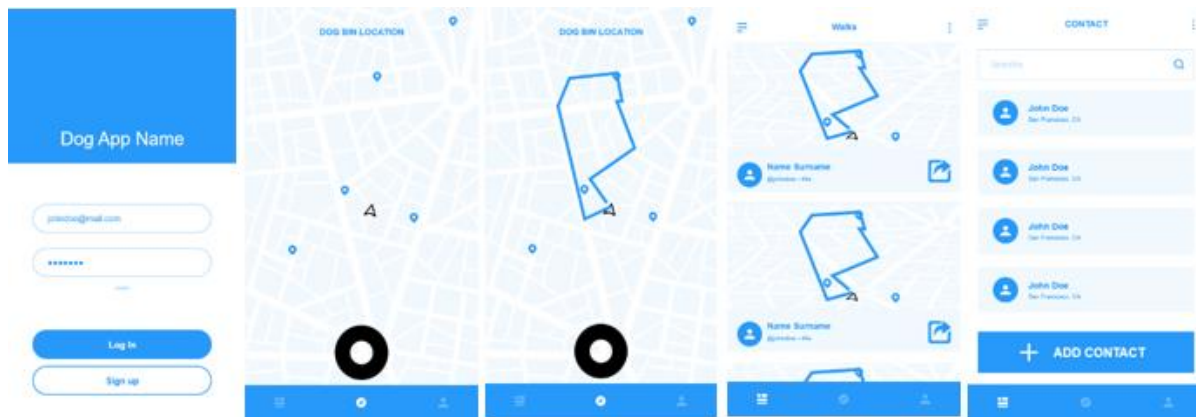


Figure 3: Screens in my low fidelity wireframe. The screens from left to right are: the Login Screen, the Map Screen, the Map Screen when recording, the Paths Screen and the Contacts Screen.

Novelty requirement

The one design requirement that is not centred around designing and making an app is the requirement that separates my app from those currently on the market. My app is the only app that uses government data of the locations of dog bins and pins them onto a map. Other apps exist that use government data but not in the same manner that mine does.

To accomplish this I took files from the data.gov.uk [21] website to use them as local files within my application. To do this, I used and read CSV files. The app takes the latitude and longitude of the dog bin and pins them on Google Maps.

I considered including a function for users to drop pins of their own bins, but in the end I decided not to include it, in order to make sure that the data provided by the app would be as accurate and reliable as possible. A feature to allow user-added bin data and test how to validate this could be included in the future.

Project Timeline

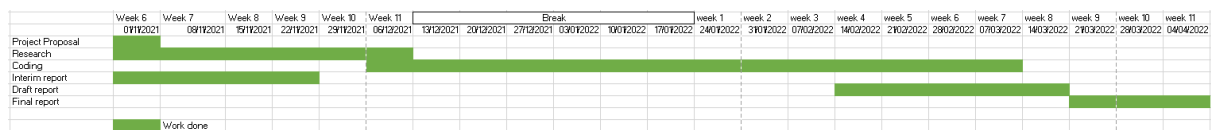


Figure 4: My project timeline.

Design

This mobile application is designed for the Android operating system. There are two main parts to the application: the client side and the server side. The client side consists of actions which take place on the user's device, on the android device in this case. The server side consists of actions which take place on a web server. The client-side section of design will cover Android studio, Java and XML. The server-side section will cover Firebase and other APIs utilised by the application.

Client-side

Android studio

When creating an Android application there are some options regarding which software to use; however, when writing an Android exclusive application, Android Studio is by far the best choice. Android Studio is the official development environment for Android [22] and is built on IntelliJ, a popular IDE for Java [23]. Android Studio and the Android system are both owned and developed by Google[22][24] and therefore are very compatible.

Java

Java used to be the official coding language of Android until 2019, when Kotlin, Google's own language, was announced to be the new official language [25]. Java is still supported by Android Studio and still frequently used by developers. Java is one of the most popular languages in the world reported by PYPL as the second most popular language based on raw data from Google Trends [26]. Java is a high level and object-oriented programming language that has been around since 1995 [27]. Almost all of the documentation and example code on the Android Studio site is available in both Java and Kotlin.

I ultimately chose to code in Java because of the amount of information on it, because it is used for a variety of other types of projects and because I have prior experience in the language.

XML

XML (Extensible Markup Language) is a markup language like HTML (Hypertext Markup Language), and not a programming language. XML is used for storing and representing data in a simple format but does not have much actual functionality. Instead, it is used to represent things to the user and to be referenced by underlying Java code to become useful for Android development. [28]

Server side

APIs

API stands for Application Programming Interface. API's allow communication between products or services in a standardised way. This means that a product or service being developed will be able to communicate with the product or service of the API in a way that both parties understand and one doesn't need to understand all the intricacies of the other to be able to utilise it. In the example of an Android API, there will be a list of commands that a coder can use to connect to a server and do work on the server side instead of the client side. I mainly used two APIs, Firebase and Google Maps, both of which are owned and managed by Google. [29][30]

Firebase

Firebase is a Backend-as-a-Service (BaaS) which has a variety of cloud-based products that programmers can use when creating their own products and services. Firebase is a platform that was founded in 2011 by James Tamplin and Andrew Lee [31] and its first product was the Firebase Realtime Database [32] used to synchronise applications data with an online database. Later it added

new products, Firebase Hosting and Firebase Authentication [33] . In 2014 it was acquired by Google, who developed and added further products to the platform and integrated it into its existing systems [34]. Today it has a variety of products that are helpful for Android, IOS and web developers and available in a variety of languages. What all these services have in common is that they are cloud-hosted, which in the right circumstances can reduce a lot of backend development. As Firebase is also owned by Google it easily integrates into Android Studio with tools and tutorials built into the software. Of the available services, I used three of them: Firebase Authentication, Cloud Firestore and Firebase Storage. [34]

Cloud Firestore

Cloud Firestore is a cloud-hosted, NoSQL and realtime database product available with Firebase. It also allows programmers to edit backend data through the console and provides analytics automatically. [34]

NoSQL Stands for Not Only SQL but it does not mean that there is no SQL, as one would assume. NoSQL came about as storage became less of an issue and cost and responsiveness of an application came more into focus. NoSQL is less rigid and allows storing large amounts of data with less strict data structures, being less focused on the relationships between data than SQL. NoSQL stores data in documents instead of SQLs relational databases and has a very different structure to that of SQL databases. One key difference is how important data duplication is in each system. In SQL normalisation of data is extremely important in order to manage storage sizes, whereas NoSQL is more compatible with duplicated data as it is more focused on retrieving data quickly. Instead of needing a bigger and better single server for a large database, NoSQL scales horizontally meaning that different parts of data can be stored in different servers. Writing tends to be quicker in SQL and reading tends to be quicker in NoSQL. [35] [36]

Cloud hosting in this case means that setting up your own server to communicate with the web as a programmer can be skipped in part or all together depending on the needs of the developer.

A realtime database means that when data in the database is updated either by the developer or users the database can notify all of the users of the update and concurrently utilise this data on connected devices.

Instead of the table format of SQL, Firestore data is organised into collections, which contain documents, and documents, which contain fields and subcollections. These subcollections can contain documents, which can contain fields and subcollections, and so on. Fields are key value pairs where the key signifies what the data is, and the value is the data itself. To access documents in subcollections the user must query through all its predecessors. When retrieving data from these documents queries are shallow, meaning that it when getting a document only the document is retrieved and not its subcollections if not specifically queried. [37]

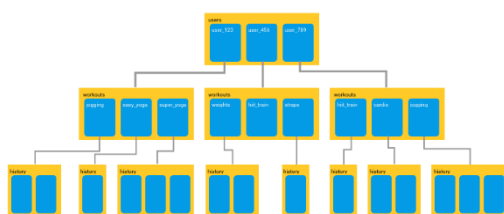


Figure 5: Structure of Cloud Firestore

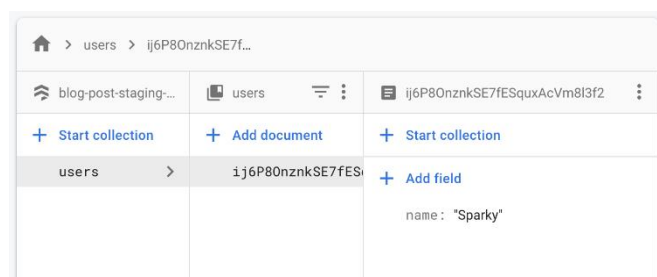


Figure 6: Screenshot of Cloud Firestore in the Firebase Console

Firebase Authentication

Authentication is used to determine whether a user is who they say they are, protect private data and allow users access to data that they have been allowed to access. Firebase authentication takes the

email and password of a user when they sign up and keeps the password secure; even those with access to the Firebase Authorization console do not have access to the user's password. When logging in, the credentials entered by the user are sent via the Firebase SDK, checked against the users and an authentication token is returned. Firebase Authentication can also allow the user to stay logged into an app after closing it. It can also allow signing in with accounts on select other services like with a Google account or a Facebook account. It is also easily integrated with other Firebase services. [34]

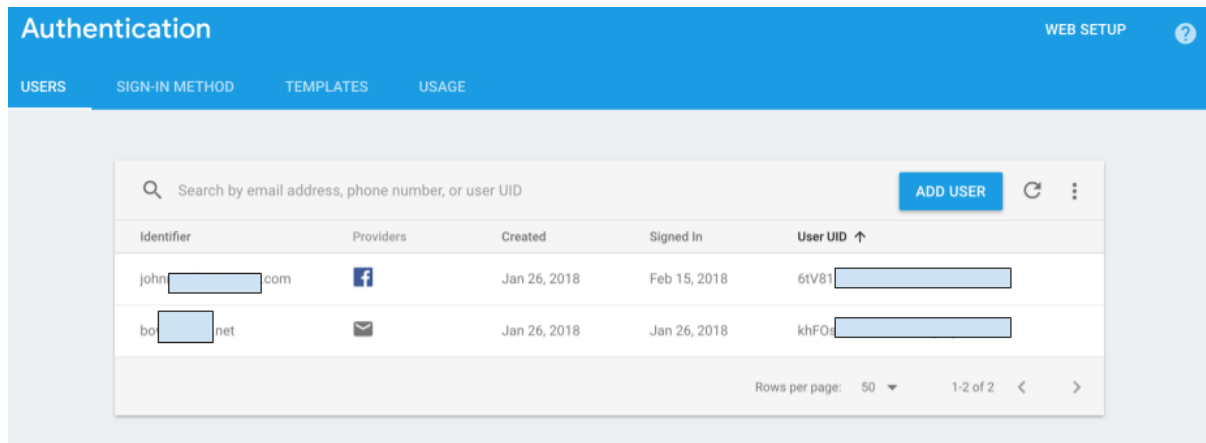


Figure 7: Example of Firebase Authentication in Firebase Console

Firebase Storage

Firebase storage is essentially a bucket that stores files on the cloud. The files are uploaded in their native format and are therefore better for storing files than Cloud Firestore which requires decoding files into a string format for storage and can have issues storing special characters. It is technically a Google Cloud product but is accessible through the firebase console along with Cloud Firestore and Firebase Authentication. [34]

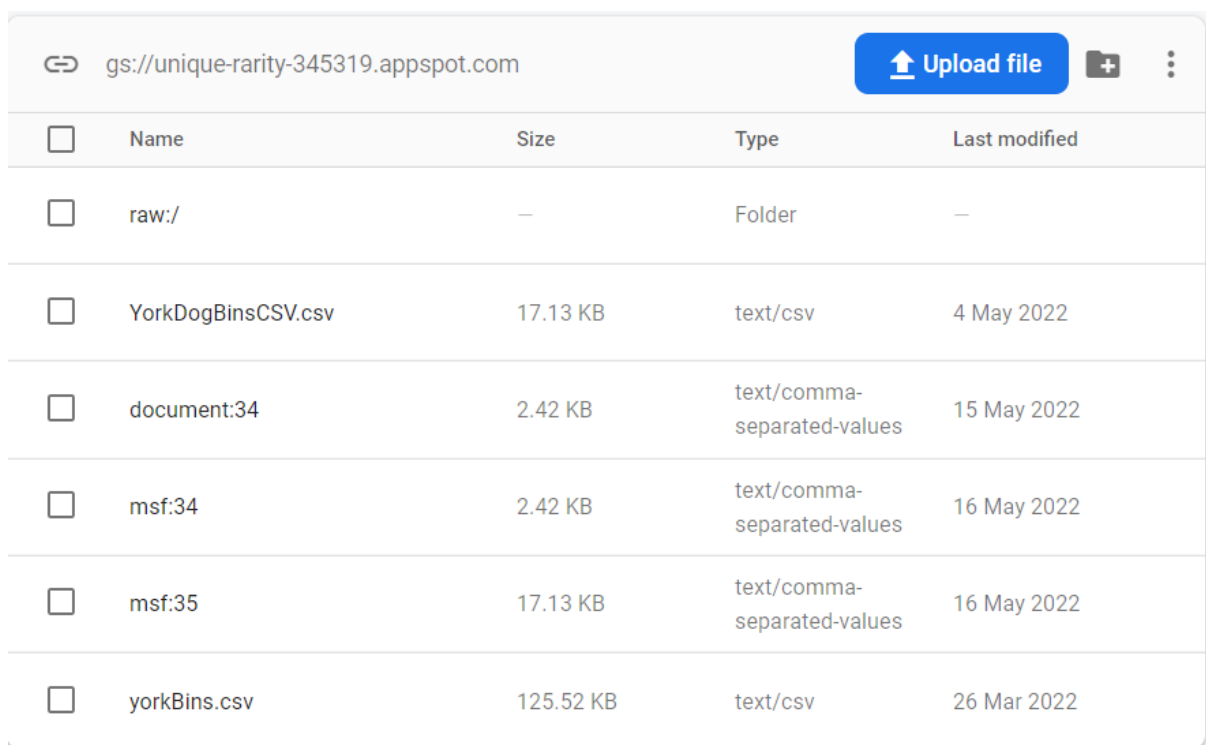


Figure 8: Screenshot of my Cloud Storage in the Firebase Console

Google Maps API and Google map related APIs

Google Maps is the most popular navigation app in the U.S., with 67% of all mapping users relying on Google Maps [38] and 5 times the amount of users of Apple Maps. Google Maps has 1.5 billion users [39] and in a recent survey of over 2000 people, 69% said that it is their favourite map app [39]. With Google once again owning the service it makes sense to use it in an Android application.

Google Maps API allows a user to embed a map into their app, place markers on the map, draw paths on the map and allows the GPS locations as well as a multitude of other things. The API allows client-side code to interact with the map. [40]

I also utilised another API based around Google Maps: Maps Static API. Maps Static API allows sending a HTTP request which returns a static image of a section of Google Maps specified by the HTTP request. There are a lot of settings for the static map returned that can be changed, like adding markers, changing zoom and adding lines. [41]

Implementation

This part of the report will cover a how the application was implemented. It will cover parts of the Android architecture, how they were used and the interactions between them, services used by the application and how they were utilised and how data was structured.

Cloud Firestore Data and its integration with Firebase Authorisation

As explained earlier, Cloud Firestore data is stored in collections, documents and subcollections. Each time data is needed, that data in the database must be queried and wait for a response from the server to ensure that the data is up to date. In the absence of up-to-date data the data is retrieved from the device cache. Each user has fields containing their username, email and full name and has subcollections. These collections do not have to exist when the user is added but are added dynamically later, when there is information to put in these sub-collections.

Firebase Authorisation only records the user's email and password and is the way the application knows which user is using the app.

Therefore, when retrieving user-specific data the app takes the users email from Firebase Authorisation and cross-references it with the user document that has a matching email field.

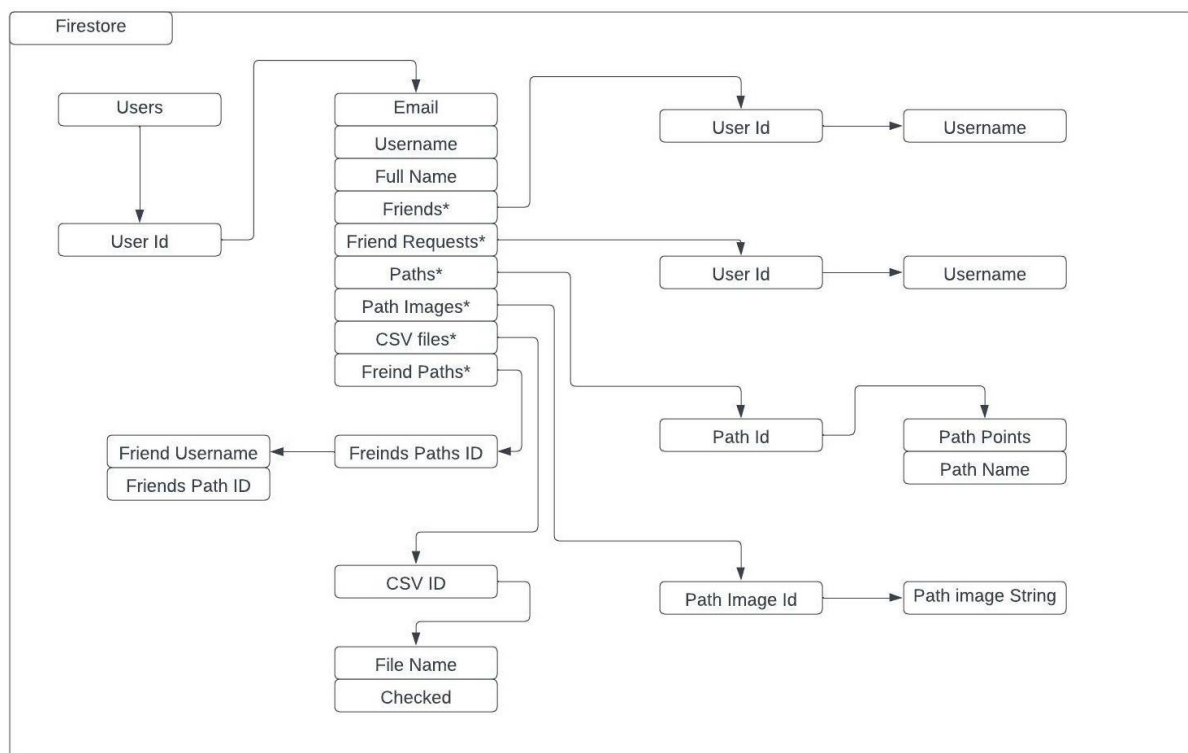


Figure 9: Diagram of my Cloud Firestore Database structure

Identifier	Providers	Created ↓	Signed in	User UID
user3@email....	✉	4 Ap...	4 Ap...	kf6777yIZieSmNA...
user2@email....	✉	4 Ap...	4 Ap...	2fHSjCZHYvaxfrQ...
user1@email....	✉	4 Ap...	9 Ap...	qC07S0BFOAWfFI...
williammalac...	✉	4 Ap...	28 A...	40Wt8oCwSkhvU...

Rows per page 50 1 - 4 of 4

Figure 10: Screenshot of my Firebase Authentication

Activity and Fragment Interaction

Any interaction the user has with the application is done through an activity. Activities have a layout file written in a markup language, XML, which is similar to HTML. The layout file is what the user will see and interact with. However, a layout file by itself has no functionality, other than showing the user information. The figure below shows how my activities and fragments interact.

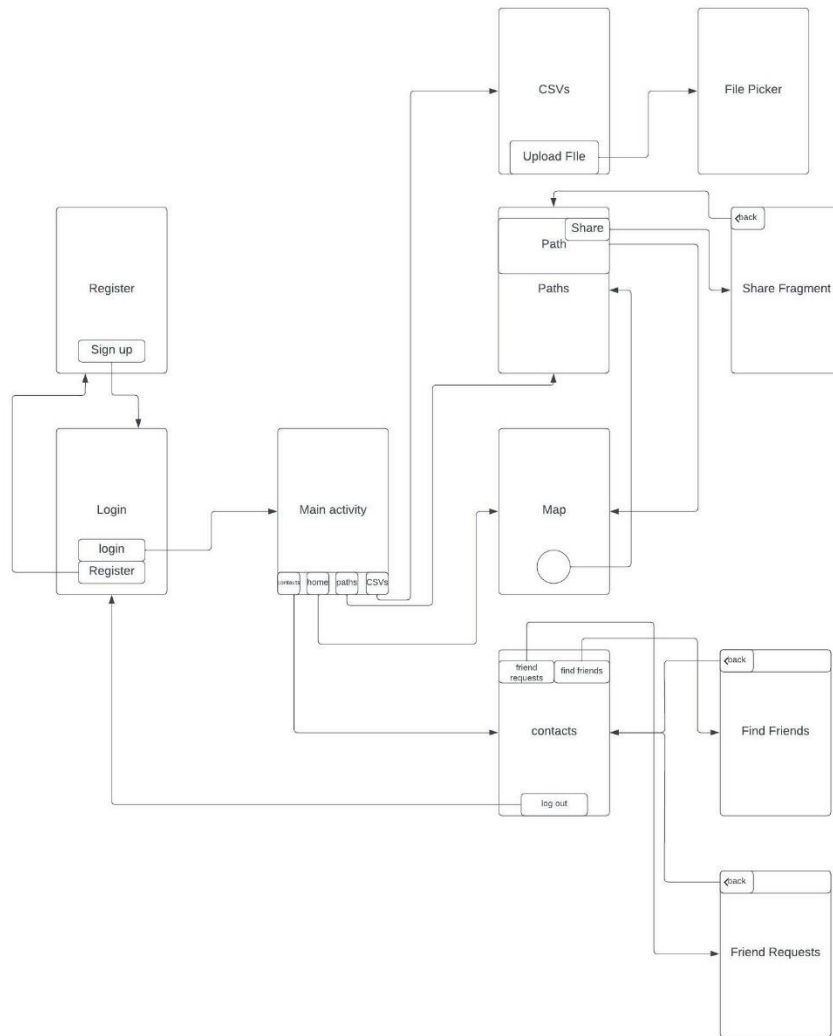


Figure 11: Diagram of the interactions between my Activities and Fragments

Activities

Activities are the base of any screen on the app. When the app starts, the application must start an activity which is declared through its manifest.xml file; without an activity the application cannot start by being opened by the user. The activity is linked to an XML layout file that will be displayed to the user, and all interactions with the layout and app as a whole are controlled by code within the activity.

Activity Lifecycle

Activities have stages and states defined by the activity lifecycle. There are six callbacks that are part of the activity class: onCreate() onStart() and onResume() are called when the app is started and onPause(), onStop() and onDestroy() are complements of these. onResume() requires onStart() and onStart() requires onCreate(). These callbacks can be overridden to add additional functionality and to stop errors. The fragment lifecycle is similar to the activity lifecycle but has some key differences.

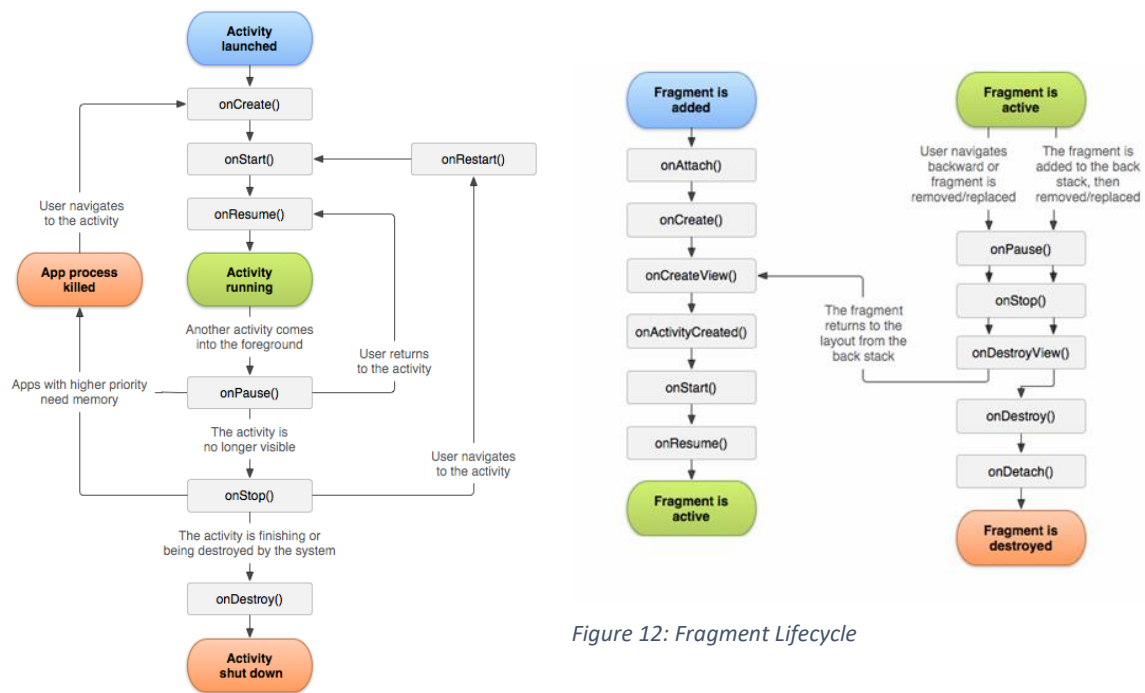


Figure 12: Fragment Lifecycle

Figure 13: Activity Lifecycle

Project Activities

I have three activities: main activity, log in activity and sign-up activity.

The app starts with the login activity, as declared by the manifest file. By overriding the start() function built into the activity, the app first checks if the user is already logged in, by checking with Firebase Authentication. This check works whether the app is connected to the internet or not, as, if it is offline, it checks the cache provided through Firebase's offline support. If the user is already logged on, the main activity is automatically started without any interaction from the user. This will happen quickly enough that the user will not see the log in screen.

```

public void onStart() {
    super.onStart();
    FirebaseAuth currentUser = mAuth.getCurrentUser();
    updateUI(currentUser);
}

public void updateUI(FirebaseUser account) {
    if (account != null) {
        Intent intent = new Intent( packageContext: this, MapsActivity.class);
        startActivity(intent);
    }
}
  
```

Figure 14: Login Activity code snippet showing automatic user login

If the user is not logged in, the user will be given the opportunity to log in with their email and password, or to register. Logging in will send their login credentials to the Firebase Authorisation and

will check if the credentials match any of the users. If the credentials are correct, Firebase sends back a token to the application that allows the user to log on. This token has the user's email in it, which is used to query their personal data in Cloud Firestore.

If the user chooses to register, the app will open the sign-up activity. The sign-up activity will ask the user to enter their name, email, username and password. When they click "sign up", the data is checked to see if all the data is there and in acceptable form, e.g., that the password meets some loose requirements. The username, email and full name of the user is stored in Cloud Firestore and the email and password are stored in Firebase Authorisation. If the log in is successful, the user is registered with Firebase Authentication and they are returned to the login activity.

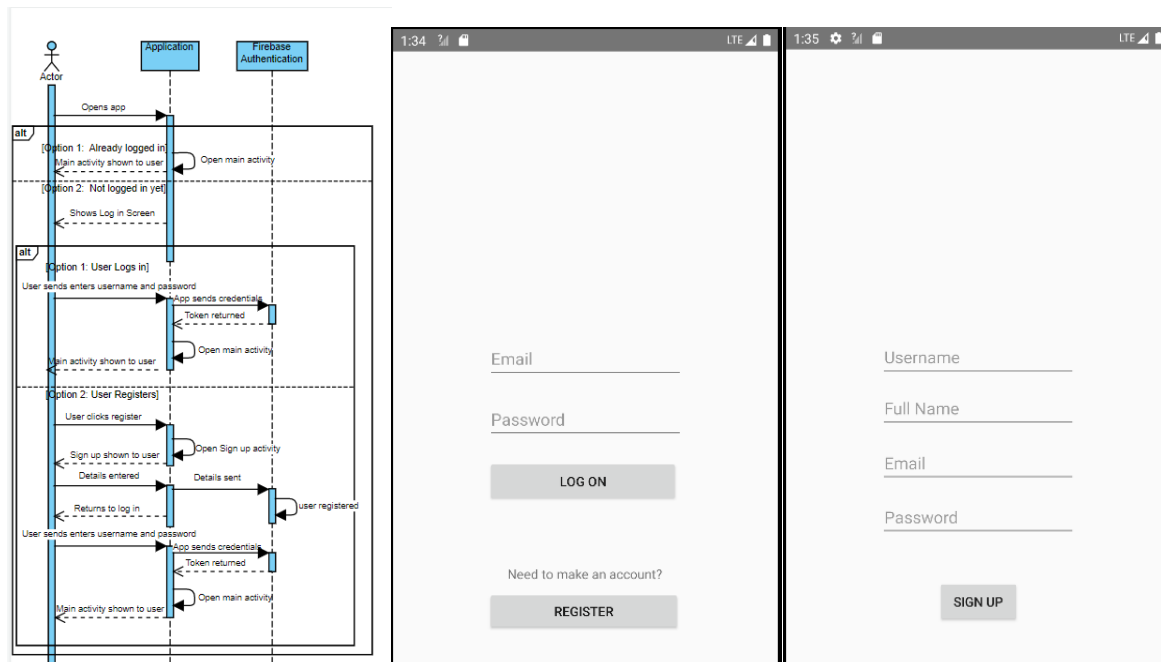


Figure 15: Automatic log in sequence diagram Figure 16: Log in Activity Figure 17: Registration Activity

The function of the main activity in this application is to serve as a host for the fragments and the navigation bar which is used to navigate between the fragments. Above the navigation bar is a view which houses all four fragments, which are run simultaneously on the start of the activity. Clicking on the navigation bar only serves to hide or show the selected fragment. This was important to do, as otherwise, when switching between fragments they would be destroyed and recreated each time they were used. The main activity also has functions which allow the fragments to communicate to each other.

Fragments

Fragments are in essence activities which are reliant on a parent activity in order to exist. Fragments are managed by their parent activity and cover a section of their parent activity's UI. Multiple fragments can be on the screen at the same time and they have the ability to house their own fragments. My main activity houses four activities: the map fragment, the contacts fragment, the paths fragment and the CSV fragment.

Maps Fragment

The maps fragment is the biggest and most complex fragment and also takes the longest to render. The maps fragment manages an embedded map, manages GPS location, retrieves and uses

information from CSV files, requests app permissions, records paths and adds them to the map in real time, and saves paths and images of those paths to the user's account.

The embedded map is requested from Google Maps through the Google Maps API with an API Key linked to the project. An asynchronous thread is started to load the map and its data so as not to slow the rest of the app. Once the map is ready, a callback is triggered and data on the map is edited.

As soon as the map has loaded, the application needs the user's location to show on the map. The user's location is sensitive and protected information, and permission must be requested from the user. Android has two location services that run in the background as long as the user allows location on their Android device, `COARSE_LOCATION` and `FINE_LOCATION`. The user is asked for permission for access to these services and when accepted the user's location is added and the map zooms into their current location. When a user declines the permission twice, Android no longer allows any app to ask for permission again. [42][43]

While the permission is being requested the map takes the CSV and adds the locations and the names of the locations to the map. CSV (Comma Separated Values) files are plain text files which contain data separated by commas and new lines. Each item on a row that would be in a table is separated by a comma instead. The new line signifies each row. The application takes the user's email from Firebase Authorisation and queries the Cloud Firestore user collection for users with a matching email address. When it finds a matching user document it queries its sub-collection, which houses the IDs of the CSV files the user has access to and then fetches the corresponding CSV files from Firebase Storage. The application takes the CSV files, which have three 'columns': latitude, longitude and name of location and adds these locations as markers with names to the map.

Hovering above the bottom of the map is a floating button that is used to start recording a path. When the button is clicked it starts a foreground service which returns the location of the user every second. The points are then plotted onto the map every and connected with a line. Once the user is done recording their path and clicks the button again the service stops, and an image of the path is shown to the user in a popup with a textbox to edit the name of the path. The user can either submit the path and the points of this line are added to the corresponding user's paths sub-collection under a path ID, or they can click back. The path then disappears from the map.

The image of the path is retrieved when the user stops recording. This image is retrieved through an HTTP request to Google's Static Maps API. The path data is added to the HTTP request which allows the API to use implicit positioning that adjusts the zoom and centre of the image dynamically to fit the path and the surrounding area. The image shows the path and the surrounding map. The image is turned into a bitmap and then encoded into a string and saved into the user's path image sub-collection under the same ID as the corresponding path in the paths subcollection. Data in the paths fragment is then notified through the main activity. [41]

Paths Fragment

The paths fragment contains a scrollable list of paths with their corresponding images. It gets the user's email from Firebase Authorisation, finds the matching user from the Cloud Firestore users document, and gets all the paths and their corresponding images and adds them to the list of paths. When the user finishes recording a path on the maps fragment, the path fragment will be notified and updated.

When a path image is clicked, the user is brought to the maps fragment and the path is added to the map removing any other paths currently on the map. The user can then follow this path.

There is also a share button that the user can click to share their paths with their friends. When the share button is clicked for a specific path, the user is taken to a child fragment containing a list of the users friends with send buttons next to their names. The user can send the path to as many people as they like. When the path is sent to a friend, the friends' friends paths subcollection is updated and will show up in their paths fragment when they use the app with the name of the user who shared it.

Users can also delete any paths from their path list.

Contacts Fragment

The contacts fragment holds a list of all of the users that the current user has added as a friend and allows them to remove these friends. The contacts fragment also has two buttons, “Friend Requests” and “Find Friends”, which are child fragments of the contacts fragment. When the user clicks “Find Friends”, they will be taken to a screen that shows only a search bar. When they search for a friend by username, the app will ask Cloud Firestore if any users have that username. If so, that username will show up along a “Send Friend Request” button, which will add the current user to their friend request sub-collection. The “Friend Requests” button will take the user to a screen with a list of all their friend requests and allow them to accept the requests. This will add the user to their friends’ subcollections and will show in their contacts fragment and be available to share paths with.

There is a delete account button that allows the user to delete their account. When clicked, a notification pops up warning them that all of their data will be deleted and asks if they are sure. When the user clicks “Yes” the user is prompted to log in again as firebase authentication only allows account deletion programmatically when the user has recently logged in with credentials. There is also a “Log Out” button that logs the user out and returns them to the login activity.

CSV Fragment

The CSV fragment holds a list of all of the CSV files uploaded by the user with a check box next to each CSV file. The Fragment also holds two buttons: Upload CSV and Confirm CSVs. Upload CSV opens an activity that is provided by Android that allows users to select files that are on the users’ devices’ storage. Before opening the file picker it notifies the user of the filetype and the required format of the CSV data. When a file is selected by the user it is uploaded to Firebase Storage and a reference to the file is saved in the users’ CSVs collection in Cloud Firestore. It then updates the list of CSVs. The Confirm CSVs button takes all of the files checked by the user in the list of CSVs and updates the map to put markers in all the selected documents’ points.

Multithreading and asynchronous tasks

Many Android apps use the internet. When using the internet asynchronous tasks are important because interacting with the internet is time consuming and needing to wait for these interactions to complete before progressing with other code slows the functionality of the entire application. To combat this, Android allows multiple threads to run at the same time, this way the application can allocate resources to other threads if one thread needs to wait for a response from the internet. However, this asynchrony can lead to issues affecting the application once the application does get the response. This means that programmers need to be cognisant of when certain data is available and be prepared to effectively deal with network issues.

Services and Broadcasts

When the user exits an app or locks their phone, the app’s currently running activity stops. When this happens all of the processes running on the app are paused. One of these processes that would be stopped in the case of this app would be the location of the user being tracked and the points being added to the map when the user is recording a path. Keeping the app open throughout recording a path would be very inconvenient to the user, they would need to have their phone unlocked throughout recording a path and be unable to use any other apps while recording a path. Android provides a solution to this by allowing services.

There are two types of services that programmers can use: foreground services and background services. Background services run without the user seeing them and can be used to get user location; however, as of API level 26 they can only retrieve the user’s location a few times each hour[44] which is not sufficient for tracking a user location for plotting a line. Foreground services are services

that add a notification to the notification bar when they are started. Foreground services can be used to get the user's location more frequently and are therefore a better fit to use for this application.[45]

When the app is not currently running in the foreground, it is unable to receive any information from a foreground service. This means that the foreground service cannot send the map activity any information during this time. Functions to send data will therefore not work. The information recorded by the service is needed when the app is opened again; so, we need to use broadcast services to pass this information.

Broadcast services send an intent out to apps on the device that are registered to receive them. These apps set up broadcast receivers to receive the broadcasts sent. All of the receivers that are currently running receive every broadcast sent, but filter which broadcasts are meant for them by the action attached to the broadcast. [46]

This application has a variety of broadcast receivers but the most important one is the one that records the location of the users when they click "record". A broadcast receiver is set up in both the maps fragment and in the service. The service sends a broadcast with the location of the user out every second and this is picked up by the receiver in the map fragment. When the app is closed the service stops sending the points but continues to record them. Once the app is opened again it sends out a broadcast notifying the service and the service sends a broadcast back containing a list of all of the points which the map fragment can then add to the map. If the app is fully shut down, as in not running in the background, or if the record button is pressed again, a broadcast is sent out notifying the service so it can destroy itself.

Broadcasts and broadcast receivers were used throughout the rest of the app to allow fragments and activities to communicate with each other.

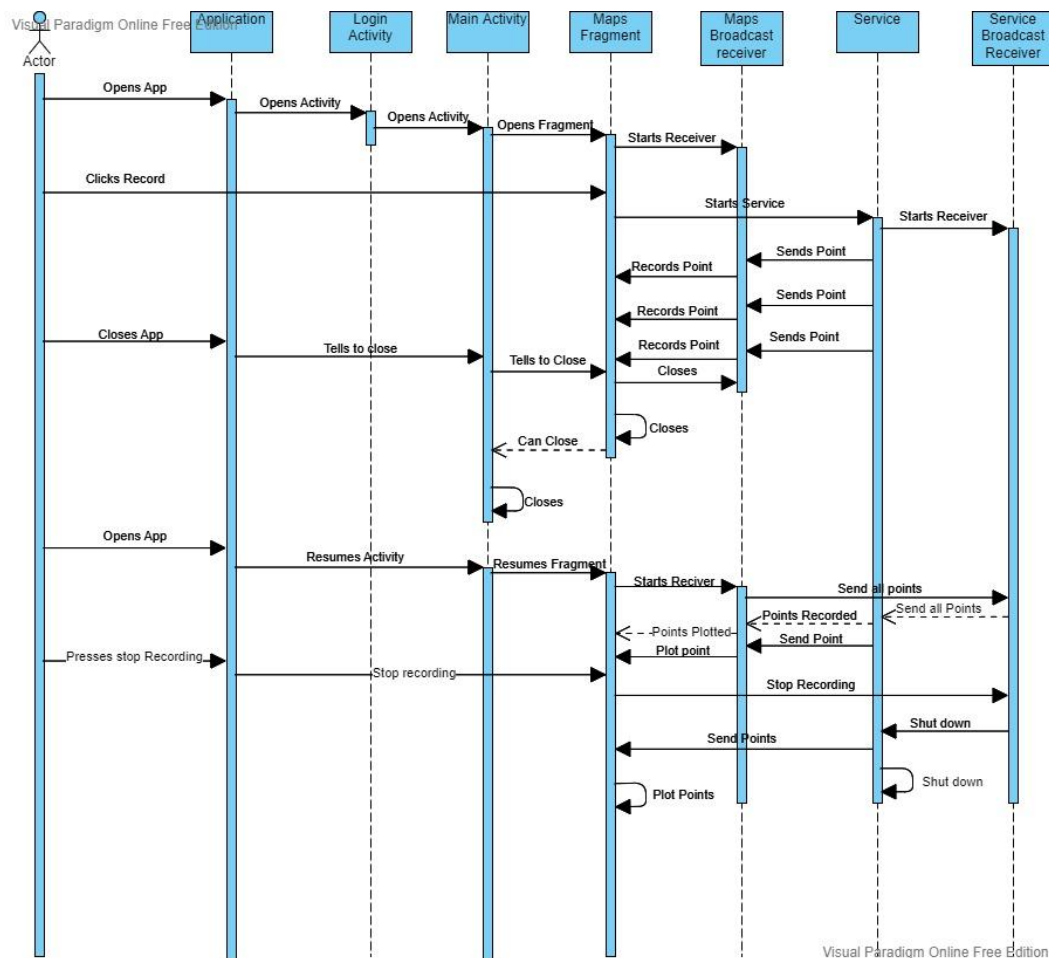


Figure 18: Service sequence diagram in specific use case

Backwards API compatibility

The rules and abilities of services and permissions have changed between API levels. Because of this the foreground service doesn't work on any API levels lower than API level 30. To allow my app to be used by users with an older device there is a secondary function which records the user's path in the maps fragment. This function works in a similar way to the foreground service by getting the user's location repeatedly and adding it to the map. There could hypothetically be a version of the tracking function for each API level but as the location permissions change so frequently from API level to API level this is not realistic to accomplish within the scope of the project.

Testing

Android Emulator

When testing the functionality of the application I made use of the Android Emulator built into Android Studio. It allows you to run the application on a variety of machines and API levels. It allows you to set the location of the device as well as set a route which the machine follows changing the location at defined speeds. It allows you to download and use other applications. Files can also be uploaded to the device.

Functional Requirements Testing

While the app was being made the app's functionality was regularly tested. Shown below is the result from the final round of testing. These unit tests cover all of my mandatory requirements as well as my desirable requirement that the data should be able to use multiple sets of government data.

Requirement	Test	Results	Issues
The app works on the Android operating system	The app was run on the Android Virtual Device Emulator on a range of API levels	The app runs on devices with a higher API level 25.	The Bottom Navigation bar isn't compatible with API levels lower than 25

Requirement	Test	Results	Issues
The app shall have the ability to create users.	Users were registered in app.	Users successfully created.	N/A
Users have unique usernames.	Tried to register a username that was already registered.	"Username already exists"	N/A
Users have unique emails.	Tried to register a email that was already registered.	"Email already exists"	N/A
User passwords must be over 8 characters.	Tried to register a password with less than 8 characters.	"Password Must Contain an Uppercase Letter, a Lowercase Letter and a Number and be at Least 8 Characters Long"	N/A
User passwords must contain an upper-case character.	Tried to register a password without an upper-case character.	"Password Must Contain an Uppercase Letter, a Lowercase Letter and a Number and be at Least 8 Characters Long"	N/A
User passwords must contain a lower-case character.	Tried to register a password without a lower-case character.	"Password Must Contain an Uppercase Letter, a Lowercase Letter and a Number and be at Least 8 Characters Long"	N/A
User passwords must contain a number.	Tried to register a password not containing a number.	"Password Must Contain an Uppercase Letter, a Lowercase Letter and a Number	N/A

		and be at Least 8 Characters Long"	
--	--	------------------------------------	--

Requirement	Test	Results	Issues
The app shall store users to allow for logging in.	Users were registered in app.	Users successfully created.	N/A
Users are registered with Firebase Authorization.	A user was registered in the app. Firebase Authorization was checked for the user.	The user was successfully created.	N/A
User documents are created and added to the users Cloud Firestore collection.	A user was registered in the app. Cloud Firestore was checked for the user.	The user document was successfully created.	N/A
Users can log in once registered.	A user was registered in the app. The login was given the users credentials.	The user was successfully logged in.	N/A

Requirement	Test	Results	Issues
The app shall have the ability to target the user's location through location services.	The app was tested for its ability to retrieve user's location.	The app successfully gets the user's location.	The app doesn't work if the user refuses location permissions.
The app shows the user's location on the map.	The app was started.	The app successfully shows the user's location on the map.	N/A
The app zooms in on the user's location	The app was started.	The app usually correctly zooms onto the user's location.	Sometimes the user location isn't zoomed in on. This seems to be an issue with the emulator. Sometimes it will zoom into the google headquarters as that is where the location is initially set for the emulator.
The user's location on the map is updated as the user moves.	The app was started. A route was set in the Google emulator which changes the emulator's location over time.	The app successfully updates the location of the user.	N/A

Requirement	Test	Results	Issues
The app shall have the ability to pin the locations of dog bins from government data.	The app was tested for if it could add locations from CSV files.	The app successfully adds points to the map from a CSV File.	N/A
The app allows the upload of CSV files	The upload button in the CSV fragment was	The file showed up in Firebase storage.	N/A

with location data to Firebase Storage.	pressed and a file was selected. Firebase storage was checked for the file.		
The app shows a document with the name of the file to the users CSV subcollection.	The upload button in the CSV fragment was pressed and a file was selected. Cloud Firestore was checked for the document.	The document showed up in the user's subcollection.	N/A
The CSV fragment shows uploaded files.	The upload button in the CSV fragment was pressed and a file was selected. The CSV fragment was checked.	The file showed up in the user's file list.	N/A
CSV Files can have the locations contained added to the map.	A CSV file was selected from the list and the "Confirm CSV Files" button was pressed.	The user was taken to the maps fragment and the locations from the files were added.	The app loads these points somewhat slowly.
Multiple CSV Files can have the locations contained added to the map.	CSV files were selected from the list and the "Confirm CSV Files" button was pressed.	The user was taken to the maps fragment and the locations from the files were added.	The app loads these points somewhat slowly.
The points can be cleared from the map.	CSV files were unselected from the list and the "Confirm CSV Files" button was pressed.	The user was taken to the maps fragment and the map was cleared.	N/A

Requirement	Test	Results	Issues
The app shall have the ability to track the user and save the user's path.	The app's ability to track and save user paths was tested.	The path was successfully tracked and saved.	N/A
The app starts the location recording service.	The "Record" button was pressed.	The notification bar was created.	N/A
A line is drawn and shown to the user when they are recording.	The "Record" button was pressed. A route was set in the Google emulator which changes the emulator's location over time.	The line was drawn successfully.	N/A
The user can stop recording location.	The "Record" button was pressed. The "Record" button was pressed again.	The service shut down.	N/A
A user is shown an image of the path and asked if they would	The "Record" button was pressed. A route was set in the Google emulator which	A pop-up was shown to the user showing the map with the path superimposed onto it.	N/A

like to name and save it.	changes the emulator's location over time. The "Record" button was pressed again.	They were given a textbox to type a name for the path and a "Submit" and a "Back" button.	
The user can save the path. The path shows up in the user's Cloud Firestore under their paths and path images subcollection. The paths have the same ID.	The "Record" button was pressed. A route was set in the Google emulator which changes the emulator's location over time. The "Record" button was pressed again. A name was given to the path and the "Submit" button was pressed.	The path shows up in the user's Cloud Firestore under their paths and path images subcollection. The paths have the same ID.	N/A
User taken to paths fragment and the list of paths is updated when the path is saved. The paths are sorted alphabetically.	The "Record" button was pressed. A route was set in the Google emulator which changes the emulator's location over time. The "Record" button was pressed again. A name was given to the path and the "Submit" button was pressed.	The user is taken to the paths fragment. The path recorded is there. The path is in the correct placement alphabetically.	The paths fragment loads quickly; however, the user is shown images of another path during this short time.
The map is cleared of the line once saved.	A path was recorded and submitted.	The line is cleared from the path.	N/A
The map is cleared of the line when the "Back" button is pressed.	A path was recorded and the "Back" button was pressed.	The line is cleared from the path.	N/A
A second path can be recorded after the first without any issues.	A path was recorded and submitted. This was done again.	Both paths show up in the user's paths fragment.	N/A
The path can be recorded while the user is not running the app in the foreground.	The "Record" button was pressed. A route was set in the Google emulator which changes the emulator's location over time. The device "Home" button was pressed. After 30 seconds the app was opened again. The "Record" button was pressed again. A name was given to the path and the "Submit" button was pressed.	The path was recorded and saved like normal.	N/A

Requirement	Test	Results	Issues
-------------	------	---------	--------

The app shall have the ability to find other users by username.	The app was tested for its ability to find other users.	The app finds other registered users.	N/A
The app will allow the user to find other users by searching for their username.	The “Find Friends” button was pressed in the contacts fragment. Another registered username was typed into the searchbar and searched for.	The app found the user.	N/A
The app allows users to send friend requests to users they search for.	The “Find Friends” button was pressed in the contacts fragment. Another registered username was typed into the searchbar and searched for. The “Add Friend” button was clicked.	The friend request was sent.	N/A
The app adds a friend request document to the user the friend request is sent to.	The “Find Friends” button was pressed in the contacts fragment. Another registered username was typed into the searchbar and searched for. The “Add Friend” button was clicked. The Cloud Firestore user database was checked.	The friend request document was present.	N/A
Users cannot send friend requests to users they have already sent a friend request to.	The “Find Friends” button was pressed in the contacts fragment. Another registered username was typed into the searchbar and searched for. The “Add Friend” button was clicked. The user was then searched for again.	The button says “Requested” instead of “Add Friend” with a tick next to the username and is unclickable.	N/A

Requirement	Test	Results	Issues
The app shall allow users to make a list of contacts.	The app contact list was tested.	The app allows users to make a list of contacts.	N/A
The app allows users to accept friend requests sent to them.	A friend request was sent from another account. The “Friend Request” button in the contacts fragment was clicked. The accept button was then clicked.	The button says “Accepted” with a tick and is unclickable.	N/A

The app adds a friend document to the current user's friend subcollection of the user they accept the friend request from	A friend request was accepted. The user's friends subcollection was then checked for the friend users' username.	The friend document was found.	N/A
The app adds a friend document to the user that the friend request was accepted from	A friend request was accepted. The friend's users friends subcollection was then checked for the user's username.	The friend document was found.	N/A
The app allows users to delete users from their friends list.	The "Delete" button was pressed. The warning popup was accepted.	The friend disappeared from the contacts list.	N/A
The app removes the deleted user friend document from the user's friends sub-collection	The "Delete" button was pressed. The warning popup was accepted. The users friends subcollection was checked.	The friend's document was deleted from the user's friends subcollection.	N/A
The app removes the user friend document from the friends friends sub-collection	The "Delete" button was pressed. The warning popup was accepted. The friends friends subcollection was checked.	The friends document was deleted from the users friends subcollection.	N/A
The app removes all shared path references from both users	The delete button was pressed. The warning popup was accepted. The friend's path subcollection was checked.	All documents of that from the user to the friend and from the friend to the user were deleted.	N/A
The app removes all of the friend's paths from the paths fragment.	The delete button was pressed. The warning popup was accepted. The user's paths list was checked.	No paths from the deleted friend users showed in the path list.	N/A
The friends list updates when a friend request is accepted	A friend request was accepted. The "Back" button was clicked.	The contact list was updated.	N/A

Requirement	Test	Results	Issues
The app shall allow users to share paths with other users.	The ability of the app to share paths was tested.	The app was able to share paths.	N/A
The app shows a list of friends to send paths to when the "Share" button of a path is pressed	The "Share" button was clicked on a path in the paths fragment.	A page with a list of the user's friends popped up with a "Send" button next to each username.	N/A

The app doesn't let users send the same path to a user more than once	The "Share" button was clicked on a path in the paths fragment that had already been shared to a user.	The button next to the user the path has been shared to already says "Sent" with a tick and is unclickable.	N/A
The app saves a reference to the path in the friends path subcollection of the user the path was sent to	The "Share" button was clicked on a path in the paths fragment. The "Send" button was clicked next to a user. The user's friend paths subcollection was checked.	A document with the name of the friend and a path reference was present.	N/A
The app shows users' paths sent to them by other users and tells them which user sent the path	A path was sent to the user. The path list was checked for the path.	The path was in the list and said who it was from.	N/A
The app allows users to delete paths sent from other users	The "Delete" button was pressed on a path shared with the user.	The path was deleted from the path list. The path document was also deleted from the friends path subcollection	N/A

Requirement	Test	Results	Issues
The app shall allow users to follow paths that have been saved	A path in the path list was clicked.	The path in the path list was plotted on the map.	N/A
The app shall allow users to follow paths that have been shared with them	A path shared by a friend in the path list was clicked.	The path in the path list was plotted on the map.	N/A
The app clears any line currently on the map when a path is clicked	A path in the path list was clicked. A different path in the path list was clicked afterwards.	The first path appeared on the map. When the second path was pressed the first path disappeared and the second path was plotted on the map.	N/A

Functional Requirement Issues

Most of the issues with my app are related to slow performance. These issues would be nice to remedy by figuring out alternative ways to manage resources; however, these delays are not of huge concern within the scope of the project.

The issue of the navigation bar requiring a minimum of API level 25 is not much of an issue as 85.65% of Android devices have this API level or above, as revealed in a report from Google.[47] Implementing an alternative for the navigation bar would be nice to implement however it is not a pressing issue.

The issue of zooming in on the user is only present on the app due to the emulator and therefore not an issue with the app.

As described earlier the issue of the app not working if location permission is denied cannot be remedied due to the Android system restrictions. However, there could be steps taken to better inform users of issues with denying permissions and having more validation checks in place to not allow users to attempt to use certain features when permission has been denied.

Generally however the functional requirements were met almost perfectly due to continuous testing and bug fixes.

Desirable requirements

The desirable requirements apart from multiple sets of government data being incorporated fell into three different categories: those that needed to be tested through user testing, those that weren't implemented due to time; and the final social media desirable requirement.

User Testing

Two of my desirable requirements could not be tested with unit testing. To be able to test these I had planned to do some limited user testing. I planned on doing this by giving participants a short task to use the app for its main intended purpose: adding points to the map local to you, recording a path, adding a friend and sending it to them, followed by them receiving a path and trying to use it for a walk.

The participant would have been given the phone and asked to do the following in order:

1. Make an account
2. Log in with their account
3. Add a friend (experimenters account)
4. Record a walk for under 5 minutes
5. Send the path to the friend's account

They would have then been asked to use a path sent by the experimenter.

They would have then been asked to rate how easy the app was to use and how aesthetically pleasing the app was on a scale of one to five.

Out of a total of ten potential participants, an average score for how easy the app was to use was and how aesthetically pleasing the app was would be retrieved and I would have used these scores to determine if my app met these requirements.

Desirable requirements unfinished due to time constraints

Due to time constraints, three of the desirable requirements couldn't be finished. These are: allowing the user to plan a trip before taking it; allowing the user to have favourite routes; and the app having the ability to record the time a path took for a user. Although these requirements would have been nice to implement, they were not very important in accomplishing the main mission of the project. If there had been more time, I would have liked to implement these desirable requirements.

Social media desirable requirement

Having later considered the actual ethical implications of allowing social media to be tied in to the app, it was decided that it did in fact open up the app to some ethical grey areas. Allowing users to share paths they walk could be a threat to their safety: the path may show a place they walk regularly or end at their home. Also, in the case that there were a security flaw in the app, there could be bad actors that might take advantage of these and access information not shared by users. For these reasons it was deemed better not to implement this feature.

Future work and Improvements

The main thing I would have liked to implement in this app, but was not yet able to, would be allowing the user to follow a recorded path in a similar way to how Google Maps allows you to follow directions. I was able to get a JSON file of directions from Google's Directions API but could not figure out how to actually have the user follow it in a similar way to how the Google Maps app does. It seems that there is no way to allow the users to follow the line in this way without implementing it myself from scratch.

I also would have liked to have a feature in the app that allowed you to plan a trip before taking it, although I am unsure how this would have been possible. In the best case, you would be able to select multiple dog bins and the app would connect them in order. Actually implementing this would have been a lot of work for a small feature and may have been difficult to implement with the tools that Google Maps API gives you.

There is an issue with framerate when interacting with the map. This issue is unsolved online, but I would like to find a way to try to reduce the framerate to make the map more responsive and user friendly.

To get images of the map with the path, I used Google Maps Static API. To correctly frame the path I used the APIs' implicit positioning feature which sets the zoom and centre intuitively based on features passed to it. However, this meant that I could not include the markers in my image as adding all the markers would change the zoom obscuring path. To add select markers, I would like to use another Google API: Google Places API. This would allow the app to register when a user is close to a marker on their path, record these points, and allow me to add only those markers to the Static Image. Google Places API would also allow for a second feature, where the user could be given a notification whenever they get are near to a marked location, allowing the user to be notified if they were walking past a dog bin.

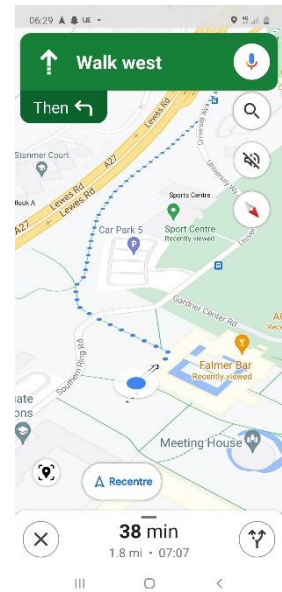


Figure 19: Screenshot of Google Directions

Conclusions

The application met all the main aims it set out to achieve. The overall goal of developing an Android application that assists dogwalkers was met. It successfully showed the location of dog bins on a map retrieved from government data and allowed users to record and share their path with other users. In the accomplishment of these goals, all of the mandatory requirements set later in planning were met, even going beyond the mandatory requirements without any meaningful errors.

Through writing over 3000 lines of java code and over 1000 lines of xml, my personal aim of expanding my knowledge and understanding of Android development was more than met. Although the project was demanding for a developer who had no prior experience in Android development, it became easier to understand as time went on, even with the application becoming more complex. I acquired a deeper understanding of all of the parts of the application framework and a range of resources commonly used in Android applications.

Although the project is based around dog walking and helping dog owners and meets the aims that were founded on this being the theme of the app, there is no limitation in terms of the app being used for this exact purpose in its current state. It could be used to add any CSV list of locations to a map and having the same functionality for this. It could be used for rental bikes, bars or restaurants, fire hydrants or potholes; the base of the app could be expanded upon in any direction where users would want a list of locations and to be able to share a path they've walked.

I found the project very interesting and would potentially like to continue working on it and publish it on the Google Play Store.

References

- [1] “Pet ownership in the UK 2022,” *Statista*. <https://www.statista.com/statistics/308235/estimated-pet-ownership-in-the-united-kingdom-uk/> (accessed May 17, 2022).
- [2] “Pet population in the UK 2022,” *Statista*. <https://www.statista.com/statistics/308201/leading-ten-pets-ranked-by-population-size-in-the-united-kingdom/> (accessed May 17, 2022).
- [3] “Dog Walking Cost - How Much Do Dog Walkers Charge in 2022?” <https://job-prices.co.uk/dog-walking/> (accessed May 17, 2022).
- [4] “Families and households in the UK - Office for National Statistics.” <https://www.ons.gov.uk/peoplepopulationandcommunity/birthsdeathsandmarriages/families/bulletins/familiesandhouseholds/2020> (accessed May 17, 2022).
- [5] “Measuring poverty,” *CPAG*, Aug. 16, 2018. <https://cpag.org.uk/child-poverty/measuring-poverty> (accessed May 17, 2022).
- [6] “Dog Population 2019 | PFMA.” <https://www.pfma.org.uk/dog-population-2019> (accessed May 17, 2022).
- [7] “Very Sad Graph: How Much Americans Have Left to Spend After Essentials, Today - The Atlantic.” <https://www.theatlantic.com/business/archive/2012/08/very-sad-graph-how-much-americans-have-left-to-spend-after-essentials-today/260606/> (accessed May 17, 2022).
- [8] 718 Views 19 Mar 2021 and 01:06 PM Priyamvada PS, “The most popular programming languages for Android app development in 2021,” *TechGig*. <https://content.techgig.com/the-most-popular-programming-languages-for-android-app-development-in-2021/articleshow/81584865.cms> (accessed May 17, 2022).
- [9] “Mobile App Development Jobs Trending in 2019 | HackerNoon.” <https://hackernoon.com/trending-now-the-state-of-mobile-app-development-jobs-p51x040gz> (accessed May 17, 2022).
- [10] J. Humphrey, “How Long Does It Take to Develop a Mobile App?,” *The Startup*, Oct. 23, 2017. <https://medium.com/swlh/how-long-does-it-take-to-develop-a-mobile-app-77574df9d18d> (accessed May 17, 2022).
- [11] “• Mobile OS market share 2021 | Statista.” <https://www.statista.com/statistics/272698/global-market-share-held-by-mobile-operating-systems-since-2009/> (accessed May 17, 2022).
- [12] “Android Architecture,” *GeeksforGeeks*, Sep. 12, 2019. <https://www.geeksforgeeks.org/android-architecture/> (accessed May 17, 2022).
- [13] “Android - Architecture.” https://www.tutorialspoint.com/android/android_architecture.htm (accessed May 17, 2022).
- [14] “Android Architecture | Android Software Stack - javatpoint,” *www.javatpoint.com*. <https://www.javatpoint.com/android-software-stack> (accessed May 17, 2022).
- [15] “An Overview of the Android Architecture - Techotopia.” https://www.techotopia.com/index.php/An_Overview_of_the_Android_Architecture (accessed May 17, 2022).
- [16] “Android - Application Components.” https://www.tutorialspoint.com/android/android_application_components.htm (accessed May 17, 2022).
- [17] “Android Application Components - Techplayon.” <https://www.techplayon.com/applications-component/> (accessed May 17, 2022).
- [18] “Android Application Components (Activities, Intents, Views, Layouts, Services) - Tutlane.” <https://www.tutlane.com/tutorial/android/android-application-components-activities-intents-views-layouts-services> (accessed May 17, 2022).
- [19] W. L. in R.-B. U. Experience, “10 Usability Heuristics for User Interface Design,” *Nielsen Norman Group*. <https://www.nngroup.com/articles/ten-usability-heuristics/> (accessed May 17, 2022).
- [20] “BCS Code of Conduct | BCS.” <https://www.bcs.org/membership-and-registrations/become-a-member/bcs-code-of-conduct/> (accessed May 17, 2022).
- [21] “Find open data - data.gov.uk.” <https://data.gov.uk/> (accessed May 17, 2022).

- [22]“Meet Android Studio,” *Android Developers*. <https://developer.android.com/studio/intro> (accessed May 17, 2022).
- [23]“IntelliJ IDEA dominates the IDE market with 62% adoption among JVM developers | Snyk,” Feb. 05, 2020. <https://snyk.io/blog/intellij-idea-dominates-the-ide-market-with-62-adoption-among-jvm-developers/> (accessed May 17, 2022).
- [24]“Google made its best acquisition nearly 16 years ago: Can you guess what it was?,” *Android Authority*, May 21, 2021. <https://www.androidauthority.com/google-android-acquisition-884194/> (accessed May 17, 2022).
- [25]“Kotlin is now Google’s preferred language for Android app development | TechCrunch.” https://techcrunch.com/2019/05/07/kotlin-is-now-googles-preferred-language-for-android-app-development/?guccounter=1&guce_referrer=aHR0cHM6Ly93d3cuZ29vZ2xlLnNvbS8&guce_referrer_sig=AQAAAMOBg23OYlZxRNp6_6fQncnbeNegwibZXqsw5lgsJ4FI5YQEVaiVI30bVINjhZED2ArqNw1RfMasUWeLhPC1I8aL85F7yGJkqyRUA6G8rU0u6lA_O_1LrB8IuiiLcElur0qOqIRFG0RS3s3PwTbLO9OnDryfWO_1N03_6_xmoSjR (accessed May 17, 2022).
- [26]“PYPL PopularitY of Programming Language index.” <https://pypl.github.io/PYPL.html> (accessed May 17, 2022).
- [27]“History of Java - Javatpoint,” *www.javatpoint.com*. <https://www.javatpoint.com/history-of-java> (accessed May 17, 2022).
- [28]“What is XML (Extensible Markup Language)?,” *WhatIs.com*. <https://www.techtarget.com/whatis/definition/XML-Extensible-Markup-Language> (accessed May 17, 2022).
- [29]“What is an API? (Application Programming Interface),” *MuleSoft*. <https://www.mulesoft.com/resources/api/what-is-an-api> (accessed May 17, 2022).
- [30]“Google Acquires Firebase To Help Developers Build Better Real-Time Apps | TechCrunch.” <https://techcrunch.com/2014/10/21/google-acquires-firebase-to-help-developers-build-better-realtime-apps/> (accessed May 17, 2022).
- [31]“Google acquires Firebase – theEword Blog.” <https://www.theeword.co.uk/blog/google-acquires-firebase/> (accessed May 17, 2022).
- [32]“The Good and the Bad of Firebase Backend Services,” *AltexSoft*. <https://www.altexsoft.com/blog/firebase-review-pros-cons-alternatives/> (accessed May 17, 2022).
- [33]“Firebase Adds Web Hosting To Its Database Platform,” *TechCrunch*. <https://social.techcrunch.com/2014/05/13/firebase-adds-web-hosting-to-its-database-platform/> (accessed May 17, 2022).
- [34]D. Stevenson, “What is Firebase? The complete story, abridged,” *Firebase Developers*, Oct. 25, 2018. <https://medium.com/firebase-developers/what-is-firebase-the-complete-story-abridged-bcc730c5f2c0> (accessed Apr. 26, 2022).
- [35]“What Is NoSQL? NoSQL Databases Explained | MongoDB.” <https://www.mongodb.com/nosql-explained> (accessed May 17, 2022).
- [36]“NoSQL Databases – What They Are and Why You Need One.” <https://www.couchbase.com/resources/why-nosql> (accessed May 17, 2022).
- [37]“Cloud Firestore Data model | Firebase Documentation.” <https://firebase.google.com/docs/firestore/data-model> (accessed May 17, 2022).
- [38]“17+ Google Maps Statistics to Survey in 2022,” *WebTribunal*. <https://webtribunal.net/blog/google-map-statistics/> (accessed Apr. 27, 2022).
- [39]“How Many People Use Google Maps Compared to Apple Maps?,” *justinobeirne.com*. <https://www.justinobeirne.com/how-many-people-use-google-maps-compared-to-apple-maps> (accessed Apr. 27, 2022).
- [40]“Add a map | Maps SDK for Android | Google Developers.” <https://developers.google.com/maps/documentation/android-sdk/map> (accessed May 17, 2022).
- [41]“Get Started | Maps Static API | Google Developers.” <https://developers.google.com/maps/documentation/maps-static/start> (accessed May 17, 2022).
- [42]“Request location permissions,” *Android Developers*. <https://developer.android.com/training/location/permissions> (accessed May 17, 2022).
- [43]“Request app permissions | Android Developers.” <https://developer.android.com/training/permissions/requesting> (accessed May 17, 2022).

- [44] “Background Location Limits | Android Developers.”
<https://developer.android.com/about/versions/oreo/background-location-limits> (accessed May 17, 2022).
- [45] “Services overview | Android Developers.”
<https://developer.android.com/guide/components/services> (accessed May 17, 2022).
- [46] “Broadcasts overview,” *Android Developers*.
<https://developer.android.com/guide/components/broadcasts> (accessed May 17, 2022).
- [47] “Google’s latest Android version distribution numbers show 11 in dead heat with 10,” *Android Police*, Nov. 22, 2021. <https://www.androidpolice.com/googles-latest-android-version-distribution-numbers-show-11-in-dead-heat-with-10/> (accessed May 17, 2022).

Appendices

Appendix A: Figure List

Figure 1: Screens in Strava application. From left to right the screens are: Strava path recording, the Recorded Path Screen, the Friend Path Screen, The Find Friends Screen and finally the Activity Naming Screen	7
Figure 2: Screens in Pooper Snooper application. From left to right the screens are: the map screen, the find nearest bin screen and the add bins/poop screen.....	8
Figure 3: Screens in my low fidelity wireframe. The screens from left to right are: the Login Screen, the Map Screen, the Map Screen when recording, the Paths Screen and the Contacts Screen.....	14
Figure 4: My project timeline.....	14
Figure 5: Structure of Cloud Firestore & Figure 6: Screenshot of Cloud Firestore in the Firebase Console	16
Figure 7: Example of Firebase Authentication in Firebase Console.....	17
Figure 8: Screenshot of my Cloud Storage in the Firebase Console	17
Figure 9: Diagram of my Cloud Firestore Database structure	19
Figure 10: Screenshot of my Firebase Authentication.....	20
Figure 11: Diagram of the interactions between my Activities and Fragments	21
Figure 12: Fragment Lifecycle	22
Figure 13: Activity Lifecycle.....	22
Figure 14: Login Activity code snippet showing automatic user login.....	22
Figure 15: Automatic log in sequence diagram & Figure 16: Log in Activity Figure 17: Registration Activity	23
Figure 18: Service sequence diagram in specific use case	26
Figure 19: Screenshot of Google Directions	36

An Android App for Dog walking Interim Report

Introduction and problem space

As of 2021, 59% of households in the UK owned a pet [1] with the most commonly owned pets being dogs and cats both having an estimated population size of 12 million in the UK [2]. There has been a huge increase in pet owners from the year before, 2020, going up with an increase of 18% [1] likely due to the pandemic. Dog owners have a lot of responsibilities to take care of their pets and one of these is walking them. An app for dog walking can help to make this an easier responsibility to deal with.

The average salary of dog walkers being well above minimum wage makes it difficult for people to hire dog walkers when issues arise which may prevent them from walking their dog [3]. Roughly estimated, there are 2.14 million dogs living in households below the poverty line in the UK [4][5][6]. In the US poor people who are close to but still above the poverty line are left with \$367 at the end of the year after essentials [7]. Assuming this is similar in the UK, this means that it's very difficult for most families to hire a professional, making apps where dog walks can be easily shared important.

The aim of this project is to develop an app to address these issues. Apps for dogs already exist and are popular as shown by multiple dog walking, sitting and training apps already on the android market with plenty of these apps having over a million downloads. It is clear there is a demand for these types of apps.

Although there are similar apps on the market, mine is unique as it uses government data to pin dog bins onto the map. This is a useful addition to dog walkers and will make my project an extension on the apps currently available on the market.

Objectives

The objective of this project is to design and implement an app that will be of use to dog walkers. The app will be written in java using android studio to run on android devices.

The app will be used to assist dog owners with walking their dogs by pinning dog bins on a map and using GPS tracking to make planning, taking, and sharing dog walks easier incorporating some social network features.

When the user opens the app it they will see a map screen. The map will show all of the dog bins around the user. When the user wants to go for a walk they will click record and their location will be tracked until they press to stop recording. They can look at the map while on the walk to see where dog bins are.

There will be a list of all recorded walks on another tab. If the user wants to go on a walk they have gone on before they can press on the walk and the app will give them directions. These walks can also be shared between users. This will be useful for users who want to share walks with dog sitters or walkers when they are unable to show them the route in person.

Motivations

This project was suggested by my supervisor Dr. Charlotte Robinson and I thought it would be an interesting and challenging project.

First and foremost, I found the idea of working on an android project interesting. I have no prior experience of Android app development, but I do have experience coding in Java, which is the

official programming language for android app development [8]. I thought this would be a good way to hone my skills in java while expanding my abilities by taking on an entirely new challenge.

Secondly, the demand for android app developers is already high and increasing so it would be good for my portfolio to have proof of being able to develop android apps.

The project is very closely related to my current course. This app takes aspects from both my Software Engineering and Human Computer interaction modules. It therefore makes sense for me to pursue this as a project.

I also like dogs having been a dog owner myself and developing an app which helps people who are in a difficult spot while helping to improve the quality of life of dogs seems like a worthwhile endeavor.

Problem Area

The main challenge of making this game will likely be learning an entirely new platform and the difficulties that come with that. The consensus seems to be that it takes at least 3 months to become an android developer and can take up to 6 years. Developing an app seems to be agreed to take about 3 months to develop. This means that I will be very pressed for time in developing it in the next 5 months.

Research and related works

The background research is broken into 3 main parts: Understanding Android, the platform the project will be on, Market research, for ideas in development and to better define the app and User experience as it is what turns a functional app into a good app.

Research into android

What is android?

Android is an open-source operating system used mainly for smartphones and tablets based on the Linux operating system. It was initially developed in 2003 by android inc. being bought by google in 2005 and released in 2007. It is now the leading operating system in the world with over 70% of the global market share in June 2021[9].

Android architecture

Android has a layered architecture with a stack of software components. This stack consists of 4 layers and 5 components. The components are the Linux Kernel, the platform libraries, the Android Runtime, The Application Framework and the Applications themselves.

Everything is built on the main component, the Linux kernel, which is the bottom layer. The Linux kernel handles all of the drivers, essentially allowing the operating system to communicate with individual devices within the phone to allow it to run. It is the direct link through drivers to things like Bluetooth, the Camera and Wi-Fi.

Platform libraries include the core libraries for the operating systems supported languages. All of these libraries have their own uses to help with different aspects of the operating system. For example, the SSL library is used for internet security and the SQLite library is used for data storage. This is in the second layer from the bottom.

Sharing the second to bottom layer is the android runtime. This contains both the core libraries and Davlik virtual machine. The Davlik machine essentially manages memory and allows android processes to be run simultaneously. The core libraries allow developers to code in the languages native to Android.

The application framework is the second level from the top and provides services which are helpful for applications and simplify their development. Examples of these services would be the location manager which provides the location through GPS or the notifications manager which manages applications, notifications, and the display of these notifications to the user.

Applications are the top layer and are what the user sees. These are standard applications that people know such as settings or camera.

These fundamentals of android architecture are important to know for my project because it defines the parts of the system that my app will interact with. Knowing that applications are on the top level and that it will be using location services through the application framework on the second layer will no doubt turn out to be useful when creating the application and when further learning about android development.

Application Components

Within the application are application components. There are four main components of the android operating system: Activities, services, Broadcast receivers and Content providers.

Activities are essentially the individual screens which are shown to the user in app. They are both what the user sees (the layout) and how the user interacts with the application (the user interface). An application can have zero activities.

Services are essentially the background functions of an application, they update data sources, send notifications, and continue application functions when in other applications.

Broadcast receivers basically are used to catch messages from the system and other applications and reacts to them.

Content providers are how the application can send internal data to other applications when requested.

Understanding the application components is important as later down the line I will need to code using these terms and understanding their uses will be essential. Knowing how android components work will also be helpful for planning my development of the app.

Market research

My app is essentially a combination of two things: path tracking apps such as those for biking and running and apps that have locations of dog bins.

Path tracking apps are quite popular. There are a lot of these on the market currently. One of the most popular types of path tracking apps available is run trackers. These apps are heavily used and highly rated by their users. This project will look at one of the most popular run tracking apps on the market to analyse its features: "Strava".

Strava is an application that uses GPS location services to track your path while cycling, swimming and running. It allows you to record your runs, recording your location while you move, taking note of your path and time and saving it to the map so that you can see your path after the fact. When you finish a run, it gives you list of details to fill out like name, what type of activity it was and who can see the exercise. It tracks a variety of things like the pace, distance, time, speed at different points in the route and so on. It has heatmaps for popular runs and has a bunch of different challenges for the user to try to partake in. It allows users to share and receive paths.

This app's features are useful when considering the development of my app in some ways but not in others. The layout is very sleek and easy to use which is good. It has a huge array of features. However, it had so many features that it seemed like it might be a little overwhelming to a first-time user.

Dog bin apps are far less popular than running apps. The most popular one I could find came in at 1,000+ downloads compared to 10,000,000+ downloads that Strava recorded on the google play store. They are also less highly rated. These apps allow users to find where dog bins are on a map as mine does. However, these apps locate dog bins through users recording where they are and users report that the experience is usually dropping their own pins on a map.

I looked at one of these apps named Pooper Snooper found a lot of issues. The app only showed dog bins when the user was within 500m therefore showing a blank map when I first entered it. This makes it useless for planning walks to take. It does not have any type of path recording, saving or sharing.

My approach will be different than both of these apps. The general aim of this app is merge these two concepts in a way that will be useful to dog walkers. There are a lot of features of the running app not required in my app as the needs of my user will be different. Things like tracking progress, pace at different points in the walk and challenges are unnecessary from the point of view of my target users. I didn't find any features in my research into dog bin apps that I would find useful to incorporate into my app.

User Experience

There are a lot of components in making a good app. I will be looking at design from a user experience standpoint. When designing my app, the most important thing I need to take into account is usability, as not only is this app designed for all types of people to use but is important for people to be able to use easily on the first try if it will be effective for people to use in an emergency. This means that all steps need to be easy to use. Onboarding, receiving a walk and taking a path are the most important aspects for this.

There is a lot of information about how to make an app with high usability without there being an industry gold standard of the most important principles so I will name a few principles that I plan to incorporate into my design.

An app should have a design that is consistent throughout. This means that if you have a toggle of a certain design all toggles should have the same color scheme and design. This allows the user to quickly identify the components of the app and increases how easy it is to learn how to use your app.

An app should use symbols that are already recognizable to the user. You should use icons that are consistent with the operating system and that are easy to understand. We all know what a phone icon looks like or what a calculator looks like therefore, using designs similar to these will help the user to quickly recognize how your app works.

An app should prioritize important content. Not all functions in an app are equally important or as commonly used, by making things important to the user more easily seen through size or color you can improve the users' experience.

This is obviously a small part of the principles of design that are important when trying to create a good user experience, but this gives an idea of the kind of principles used.

Code of conduct and ethical considerations

Following the ethical guidelines is not only important to do for moral reasons but also is required when developing this project. There are two components to having a project that follows the ethics required. The project must adhere to the British computer society (BCS) code of conduct. The BCS is a charity within the UK “dedicated to lead IT industry through its ethical challenges, to support the people who work in the industry, and to make IT good for society.” It must also follow the universities internal ethical process ensuring that it has had the correct level of ethical approval.

BCS code of conduct

The BCS code of conduct is a list of professional standards that all members must follow.

1. Public Interest

You shall:

- a. have due regard for public health, privacy, security and wellbeing of others and the environment.
- b. have due regard for the legitimate rights of Third Parties* .
- c. conduct your professional activities without discrimination on the grounds of sex, sexual orientation, marital status, nationality, colour, race, ethnic origin, religion, age or disability, or of any other condition or requirement
- d. promote equal access to the benefits of IT and seek to promote the inclusion of all sectors in society wherever opportunities arise.

My project has no ill effect on public health, privacy security or the wellbeing of others and the environment. It could even be said to have a positive effect on public health due to encouraging exercise and assisting to keep waste off the street.

My project will keep in mind the legitimate rights of third parties throughout and any assets used will be checked to ensure that their respective owners allow their use in this way. As this would be a free app there will be no monetary profit from assets used which will make this task easier.

There will be no discrimination on the grounds of any protected or unprotected condition. The app will not ask for any of these details and will be conscious to avoid indirectly affecting these groups.

The foundational idea of the project is to try to support lower class sectors of society. In addition, throughout development there will be a conscious effort to make the app as accessible as possible to everyone. The app will be free on the largest mobile operating system which makes it as accessible as possible while only being on a single platform.

2. Professional Competence and Integrity

You shall:

- a. only undertake to do work or provide a service that is within your professional competence.
- b. NOT claim any level of competence that you do not possess.
- c. develop your professional knowledge, skills and competence on a continuing basis, maintaining awareness of technological developments, procedures, and standards that are relevant to your field.
- d. ensure that you have the knowledge and understanding of Legislation* and that you comply with such Legislation, in carrying out your professional responsibilities.
- e. respect and value alternative viewpoints and, seek, accept and offer honest criticisms of work.

- f. avoid injuring others, their property, reputation, or employment by false or malicious or negligent action or inaction.
- g. reject and will not make any offer of bribery or unethical inducement

My project was suggested by my supervisor therefore it is within the professional competence of someone at my level in university.

My project will develop my professional knowledge skills and competence as its scope goes beyond what I've learned within an attainable level.

My project will not break any legislations and I will check with my supervisor to ensure that this is the case throughout the project.

I will share the progress I have made as I develop my project and actively seek criticisms of my work by convening with my supervisor.

This project will not cause any injury to persons, property, reputation or employment, the scope of the project doesn't allow for the project to cause any injury.

I will not be bribed or allow any form of unethical inducement.

3. Duty to Relevant Authority

You shall:

- a. carry out your professional responsibilities with due care and diligence in accordance with the Relevant Authority's requirements whilst exercising your professional judgement at all times.
- b. seek to avoid any situation that may give rise to a conflict of interest between you and your Relevant Authority.
- c. accept professional responsibility for your work and for the work of colleagues who are defined in a given context as working under your supervision.
- d. NOT disclose or authorise to be disclosed, or use for personal gain or to benefit a Trustee Board Regulations Schedule 3 v7 – Code of Conduct for BCS Members Page 3 of 5 Reviewed by Trustee Board 9 June 2021 third party, confidential information except with the permission of your Relevant Authority, or as required by Legislation.
- e. NOT misrepresent or withhold information on the performance of products, systems or services (unless lawfully bound by a duty of confidentiality not to disclose such information), or take advantage of the lack of relevant knowledge or inexperience of others.

My app Design will follow all of the guidelines set out by my relevant authority. I will be following the universities guidelines for ethics and expected work while completing this project. I will also use my professional judgement throughout development.

I will seek to avoid any conflicts of interest between my project and the university. I will do this by following the ethical guidelines and by following all requirements set by the university

I accept full responsibility for the success of my project. There are no colleagues who are working under my supervision.

If by any chance I do come across information that is sensitive with relevance to a third party. I will check with my relevant authority the university before disclosing it while also staying in line with current legislation.

My project will not misrepresent or withhold any information about products, services or systems or take advantage of the lack of knowledge of others. I will give a full and accurate description of my project to the best of my ability.

4. Duty to the Profession

You shall:

- a. accept your personal duty to uphold the reputation of the profession and not take any action which could bring the profession into disrepute.
- b. seek to improve professional standards through participation in their development, use and enforcement.
- c. uphold the reputation and good standing of BCS, the Chartered Institute for IT.
- d. act with integrity and respect in your professional relationships with all members of BCS and with members of other professions with whom you work in a professional capacity.
- e. encourage and support fellow members in their professional development.

I accept my personal duty to uphold the reputation of the profession. I will not take any action which could bring the profession into disrepute.

I will follow all current professional standards.

I will not harm the reputation of the BCS.

I don't think I will interact with any other members of the BCS, if I did I would act with integrity as I will with those of all professions I will work with in a professional capacity.

I will do my best to support any other members of the BCS if the opportunity arises.

Ethical approval

The project will not involve human participants nor will the development of it involve animals. The project does not expose anyone to harm; physical or psychological. The project would be able to take personal data entered by the user but will not as the app will not be put to market, meaning that there will be no actual users apart from myself. The research poses no threat to the environment or society. There aren't any other ethical issues that require further ethical review. For these reasons I do not need any ethics clearance from the university.

As the app will be self-contained and not released on the app market there will be no users with which my project could raise ethical issues.

Requirements

Mandatory

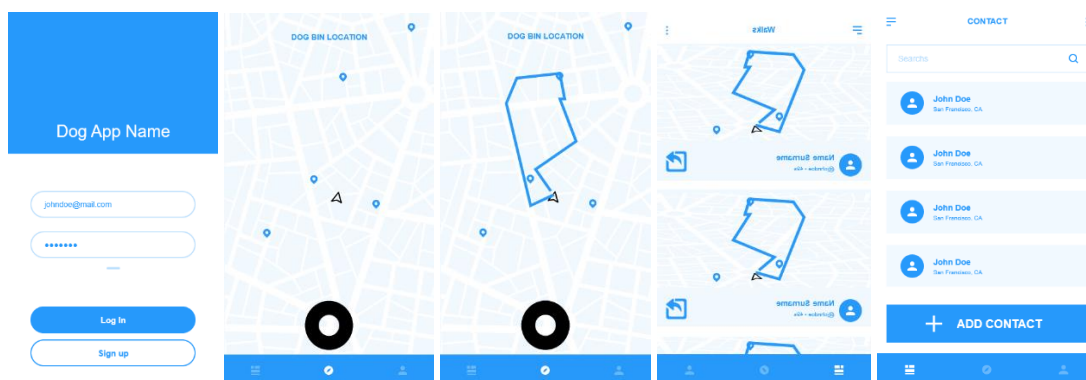
1. The app shall work on the android operating system.
2. The app shall have the ability to create users.
3. The app shall have the ability to store created users to allow for logging in.
4. The app shall have the ability to target the users location through location services.
5. The app shall have the ability to pin the locations of dog bins from government data.
6. The app shall have the ability to track the user and save the users path.
7. The app shall have the ability to save the users path to their account.
8. The app shall have the ability to find other users by username.
9. The app shall allow users to create a list of contacts.
10. The app shall allow sharing of the paths recorded by a user to other users.
11. The app shall have the ability to record the time it took the user to create the path.

Desirable

12. The app should be easy to use.
13. The app should be aesthetically pleasing.
14. The app should allow users to have favorite routes.
15. The app should allow the user to share routes though social media platforms.
16. The app should incorporate multiple sets of government data.
17. The app should allow you to plan a trip before taking it.
18. The app shall allow users to follow paths in real time.

Low Fidelity Wireframe

While trying to define my design requirements I drafted up a basic low fidelity wireframe to get a clearer idea of how my app will function.



Novelty requirement

The one design requirement that is not centered around designing and making an app is the requirement that separates my app from those currently on the market. My app is the only app that uses government data of the locations of dog bins and pins them onto a map. Other apps exist that use government data but none quite in the way that mine does.

To accomplish I will take files from the data.gov website use them as local files within my application. I will be using and reading csv files as I have handled them before, admittedly only in python. The app will take the latitude and longitude of the dog bin and pin them on google maps.

Project plan

At the time of writing this report I am slightly behind schedule. I will need to do a lot more research specifically in how to code in android before I dive into coding the actual project. Once I have done more research and practice in coding android apps I will be able to flesh out my project plan.

References

- [1] Statista. 2021. *Pet ownership in the UK 2021* / Statista. [online] Available at: <<https://www.statista.com/statistics/308235/estimated-pet-ownership-in-the-united-kingdom-uk/>> [Accessed 18 November 2021].
- [2] Statista. 2021. *Pet population in the UK 2021* / Statista. [online] Available at: <<https://www.statista.com/statistics/308201/leading-ten-pets-ranked-by-population-size-in-the-united-kingdom/>> [Accessed 18 November 2021].
- [3] Job-prices.co.uk. 2021. *Dog Walking Cost - How Much Do Dog Walkers Charge? [2021]*. [online] Available at: <<https://job-prices.co.uk/dog-walking/>> [Accessed 18 November 2021].
- [4] Ons.gov.uk. 2021. *Families and households in the UK - Office for National Statistics*. [online] Available at: <<https://www.ons.gov.uk/peoplepopulationandcommunity/birthsdeathsandmarriages/families/bulletins/familiesandhouseholds/2020#:~:text=There%20were%20an%20estimated%2027.8,over%20the%20last%2010%20years.>> [Accessed 18 November 2021].
- [5] CPAG. 2021. *Measuring poverty*. [online] Available at: <<https://cpag.org.uk/child-poverty/measuring-poverty>> [Accessed 18 November 2021].
- [6] Pfma.org.uk. 2021. *Dog Population 2019*. [online] Available at: <<https://www.pfma.org.uk/dog-population-2019>> [Accessed 18 November 2021].
- (The above references (4,5,6) combine to make the rough estimate I give [5] says there were an estimated 27.8 million households in the UK in 2020, an increase of 5.9% over the last 10 years. [6] says 22% of households are below the poverty line and [7] says that 25% of households have dogs at 1.4 per household. Combining these statistics we get a rough estimate: $27.8 * 0.22 * 0.25 * 1.4 =$ Roughly 2.14 million dogs living in households below the poverty line)
- (the following resource calculated number of pets in poverty in a similar way)
- Core.ac.uk. 2021. [online] Available at: <<https://core.ac.uk/download/pdf/235411465.pdf>> [Accessed 18 November 2021].
- [7] Thompson, D., 2021. *Very Sad Graph: How Much Americans Have Left to Spend After Essentials, Today*. [online] The Atlantic. Available at: <<https://www.theatlantic.com/business/archive/2012/08/very-sad-graph-how-much-americans-have-left-to-spend-after-essentials-today/260606/>> [Accessed 18 November 2021].
- [8] Pick, N., Test, S., Wizards, E., Communities, J. and Guide, T., 2021. *The most popular programming languages for Android app development in 2021*. [online] TechGig. Available at: <<https://content.techgig.com/the-most-popular-programming-languages-for-android-app-development-in-2021/articleshow/81584865.cms>> [Accessed 18 November 2021].
- [9] Statista. 2021. *Mobile OS market share 2021* / Statista. [online] Available at: <<https://www.statista.com/statistics/272698/global-market-share-held-by-mobile-operating-systems-since-2009/>> [Accessed 18 November 2021].

Bibliography

- About us | BCS [WWW Document], n.d. URL <https://www.bcs.org/about-us/> (accessed 11.18.21).
- Almost 400 apps that use uk gov open data - documentation portal [WWW Document], n.d. URL <http://http://sites.linkeddata.center/help/business/cases/296-apps-that-use-uk-datasets> (accessed 11.18.21).
- Android - Architecture [WWW Document], n.d. URL https://www.tutorialspoint.com/android/android_architecture.htm (accessed 11.17.21).
- Android Architecture - Tutlane [WWW Document], n.d. URL <https://www.tutlane.com/tutorial/android/android-architecture> (accessed 11.17.21).
- Android Operating System : Introduction, Features & Its Applications [WWW Document], 2013. . ElProCus - Electronic Projects for Engineering Students. URL <https://www.elprocus.com/what-is-android-introduction-features-applications/> (accessed 11.17.21).
- Android OS Architecture - Techplayon [WWW Document], n.d. URL <https://www.techplayon.com/android-os-architecture/> (accessed 11.17.21).
- Best answer: How long will it take to learn Android development? [WWW Document], 2021. . OS Today. URL <https://ostoday.org/без-пубрики/best-answer-how-long-will-it-take-to-learn-android-development.html> (accessed 11.17.21).
- Francis-Devine, B., 2021. Poverty in the UK: statistics.
- Mobile OS market share 2021 [WWW Document], n.d. . Statista. URL <https://www.statista.com/statistics/272698/global-market-share-held-by-mobile-operating-systems-since-2009/> (accessed 11.17.21).
- Ogono, U., 2021. How to Become an Android Developer: What Is the Best Android Developer Career Path? [WWW Document]. Career Karma. URL <https://careerkarma.com/blog/how-to-become-an-android-developer/> (accessed 11.17.21).
- The Main Components of Android, n.d. . Skillsoft. URL <https://www2.skillsoft.com/courses/5525673-the-main-components-of-android/> (accessed 11.16.21).
- What is Android? Here's everything you need to know [WWW Document], 2021. . Android Authority. URL <https://www.androidauthority.com/what-is-android-328076/> (accessed 11.16.21).

Appendices

Project Proposal

An android app for dog walkers

Candidate Number: 204651

Supervisor Name: Dr. Charlotte Robinson

Aims and objectives

Primary objectives

The main objective of the project is to make an android app that would be a benefit to dog walkers.

The app needs to use publicly available data from multiple councils in the UK made available on the UK government website. The data used for the primary objective will be the location of dog and litter bins. The locations of these will be put over a map and will be able to be searched for by residents of the locations whose councils have uploaded this data.

The app needs to take into consideration apps already available on the market. It will do this in a systematic way by finding what similar apps are currently available and finding what issues the users have with said apps or the suggestions that they have for the current popular apps. Fixing these issues and implementing these suggested changes will allow for this project to extend on what is currently available to be uniquely helpful to users. Using google play reviews may count as social media use if this route is taken. If this counts as social media ethical approval may be needed.

The app will need to be fully functional and easy to use. This will require through bug testing and some feedback on current designs. It will require implementation of research into intuitive app design and current app design practices.

Extensions

Being able to save and reuse walks through the app. This will involve having location services track, save and store walks as the user goes on them.

Being able to send saved walks to friends or share on social media. It will also require individual users to be registered and differentiated. Involves social media integration.

Relevance

The project is very closely related to my degree course of computer science as the main part of it is developing an app. Specifically it is related to software engineering as it is about planning, designing and developing software and human computer interaction as it involves designing an effective and enjoyable product through user centered design. It will be an individual project of a similar scale to the group project completed in software engineering so although it will lack some of the issues of

working as a team it will be more difficult in other aspects such as an increased workload and a competency in all aspects of the software production process. Similarly human computer interaction is also a group module, and this project will require all of the design ideas to come solely from one individual.


The project will require me to use java which I have learned in other modules but requires app development which will be a new territory not covered by my course and will require me to develop and utilize skills beyond my current university education.

Resources Required

This project does not require the university to provide any resources beyond those afforded by my course.

Personal timetable

Monday 15 Nov	Tuesday 16 Nov	Wednesday 17 Nov	Thursday 18 Nov	Friday 19 Nov
08:00				
08:30				
09:00	Comparative Programming (Lecture 1) Shawcross AS01		Introduction to Computer Security (Laboratory 1) Chichester 1 CHI 016 (09:00-11:00)	
09:30			Introduction to Computer Security (Laboratory 1) Chichester 1 CHI 017 (09:00-11:00)	
10:00			Introduction to Computer Security (Laboratory 1) Chichester 1 CHI 018 (09:00-11:00)	
10:30				
11:00				
11:30				
12:00				
12:30				
13:00	Introduction to Computer Security (Lecture 1) Arts A A02		Human-Computer Interaction (Seminar 2) Chichester 1 CHI 027 (13:00-15:00)	Introduction to Computer Security (Lecture 1) Arts A A02
13:30			Human-Computer Interaction (Seminar 2) Chichester 1 CHI 028 (13:00-15:00)	
14:00				
14:30				
15:00	Human-Computer Interaction (Lecture 1) Online See Canvas			
15:30				
16:00			Individual Project (Lecture 1) Fulton Building FULTON B	
16:30			Comparative Programming (Laboratory 2) Chichester 1 CHI 014 (17:00-18:00)	
17:00			Comparative Programming (Laboratory 2) Chichester 1 CHI 015 (17:00-18:00)	
17:30				
18:00				
18:30				
19:00				
19:30				
20:00				
20:30				
21:00				
21:30				

 Information about subscribing to your Sussex Direct timetable from an external calendar.

Bibliography

Dawson, C.W. (2009). *Projects in computing and information systems*. Harlow: Prentice Hall.

Interim log

User Testing Compliance Form for UG and PGT Projects*

School of Engineering and Informatics

University of Sussex

1. Participants were not exposed to any risks greater than those encountered in their normal working life.

Investigators have a responsibility to protect participants from physical, mental and emotional harm during the investigation. The risk of harm must be no greater than in ordinary life. Areas of potential risk that require ethical approval include, but are not limited to, investigations that require participant mobility (e.g. walking, running, use of public transport), unusual or repetitive activity or movement, physical hazards or discomfort, emotional distress, use of sensory deprivation (e.g. ear plugs or blindfolds), sensitive topics (e.g. sexual activity, drug use, political behaviour, ethnicity) or those which might induce discomfort, stress or anxiety (e.g. violent video games), bright or flashing lights, loud or disorienting noises, smell, taste, vibration, or force feedback.

2. The study materials were paper-based, or comprised software running on standard hardware.

Participants should not be exposed to any risks associated with the use of non-standard equipment: anything other than pen-and-paper, standard PCs, mobile phones, and tablet computers is considered non-standard.

3. All participants explicitly stated that they agreed to take part, and that their data could be used in the project.

Participants cannot take part in the study without their knowledge or consent (i.e. no covert observation). Covert observation, deception or withholding information are deemed to be high risk and require ethical approval through the relevant C-REC.

If the results of the evaluation are likely to be used beyond the term of the project (for example, the software is to be deployed, the data is to be published or there are future secondary uses of the data), then it will be necessary to obtain signed consent from each participant. Otherwise, verbal consent is sufficient, and should be explicitly requested in the introductory script (see Appendix 1).

4. No incentives were offered to the participants.

The payment of participants must not be used to induce them to risk harm beyond that which they risk without payment in their normal lifestyle. People volunteering to participate

*This checklist was originally developed by Professor Steven Brewster at the University of Glasgow, and modified by Dr Judith Good for use at the University of Sussex with his permission.

in research may be compensated financially e.g. for reasonable travel expenses. Payments made to individuals must not be so large as to induce individuals to risk harm beyond that which they would usually undertake.

5. No information about the evaluation or materials was intentionally withheld from the participants.
Withholding information from participants or misleading them is unacceptable without justifiable reasons for doing so. Any projects requiring deception (for example, only telling participants of the true purpose of the study afterwards so as not to influence their behaviour) are deemed high risk and require approval from the relevant C-REC.
6. No participant was under the age of 18.
Any studies involving children or young people are deemed to be high risk and require ethical approval through the relevant C-REC.
7. No participant had a disability or impairment that may have limited their understanding or communication or capacity to consent.
Projects involving participants with disabilities are deemed to be high risk and require ethical approval from the relevant C-REC.
8. Neither I nor my supervisor are in a position of authority or influence over any of the participants.
A position of authority or influence over any participant must not be allowed to pressurise participants to take part in, or remain in, any study.
9. All participants were informed that they could withdraw at any time.
All participants have the right to withdraw at any time during the investigation. They should be told this in the introductory script (see Appendix 1).
10. All participants have been informed of my contact details, and the contact details of my supervisor.
All participants must be able to contact the investigator and/or the supervisor after the investigation. They should be given contact details for both student and supervisor as part of the debriefing.
11. The evaluation was described in detail with all of the participants at the beginning of the session, and participants were fully debriefed at the end of the session. All participants were given the opportunity to ask questions at both the beginning and end of the session.
Participants must be provided with sufficient information prior to starting the session, and in the debriefing, to enable them to understand the nature of the investigation.

12. All the data collected from the participants is stored securely, and in an anonymous form.
All participant data (hard-copy and soft-copy) should be stored securely (i.e. locked filing cabinets for hard copy, password protected computer for electronic data), and in an anonymised form.

Project title: An Android Framework Based Application to Assist with Canine Exercise

Student's Name: William Moog-McNeill

Student's Registration Number: 21806816

Student's Signature:



Date: 17/05/2022

Supervisor's Name: _____

Supervisor's Signature: _____

Date: _____

Final Year Project, 2021-22

Will Moog-McNeill

The aim of this study is to investigate the suitability of a new application. We cannot tell how good this application is unless we ask those people who are likely to be using it, which is why we need to run studies like these.

I will be observing you while you perform the tasks following tasks:

1. Make an account
2. Log in with your account
3. Add my account as a friend
4. Record a path for under 5 minutes
5. Send the path to my account

I will then send you a path to use from my account

If you have any questions or issues, please ask me, and please let me know when you are finished.

I will ask you to rate the useability and the aesthetic of the app at the end of the study. Please remember that it is the system, not you, that is being evaluated.

You are welcome to withdraw from the study at any time Please be assured that any data collected will be stored securely and in an anonymous table.

Do you agree to take part in this evaluation? Do you have any questions before we start?

Sign here: _____

Android Application User Testing

Final Year Project, 2021-22

Will Moog-McNeill

The aim of this study was to investigate the suitability of a new application. Do you have any comments or questions about the experiment?

Here are my contact details, and those of my supervisor, and please let us know if you have any further questions about this study:

My email: wm210@sussex.ac.uk

My supervisor's email: Charlotte.Robinson@sussex.ac.uk

Thank you for your help.