



Specification Document

Project Name: N-Go! (Northumberland Go!)

Team Number 12



Document Information

Project Name:	Northumberland Go! (N-Go!)		
Prepared By:	Team 12	Preparation Date:	11/11/2018
Email / Phone:	E.Wormley@newcastle.ac.uk		
Document Version No:	2.4	Document Version Date:	08/03/2018

Version History

Ver. No.	Ver. Date	Revised By	Description
1.0	01/11/2018	Ethan Wormley	Layout of the specification/formatting
1.1	05/11/2018	Stefan Lip	Roles & Deliverables Project Purpose
1.2	14/11/2018	Ethan Wormley	Project Plan/GANTT charts added
1.3	19/11/2018	Calum & Alex	Requirements and Other Considerations
1.35	22/11/2018	Robert & Ethan	Some definitions of terms Formatting of specification, section explanations.
1.4	27/11/2018	Robert, Sam and Calvin	Background Research Hardware/Software to use More definitions of terms.
1.45	3/12/2018	Robert & Calvin	Expansion on project purpose More definitions of terms References for all external material so far.
1.5	5/12/2018	Calvin & Sam	Background research Additional references Resources

1.6	6/12/2018	Michael Surya	GUI diagrams & Descriptions
1.65	8/12/2018	Ethan Wormley	Specification formatting Written Improvements
1.7	9/12/2018	Ethan, Stefan & Michael	Testing Objectives Test Plan GUI descriptions What the software will/will not do
1.8	11/12/2018	Ethan, Sam & Robert	Formatting and Spell Checking Activity/Flow Diagrams and descriptions Bullet points for missing sections
1.9	11/12/2018	Team 12	Final Editing/Formatting before submission Contribution Matrix
2.0	3/02/2019	Ethan Wormley Stefan Lip	Feedback Improvements: Table of Contents Testing information Semester 2 project plan
2.1	17/02/2019	Ethan Wormley	Formatting of What the software will/will not do Specs of Hardware used (Pc's and Emulators)
2.15	21/02/2019	Ethan, Calvin and Robert	Updated Version History Use Case Diagram Architecture Diagram
2.2	24/02/2019	Michael & Alex	Updated GUI diagrams with new colour scheme New screenshots and descriptions
2.3	27/02/2019	Ethan & Calum	Non-functional requirements
2.35	28/02/2019	Sam Holley	Additional flow diagram and description
2.4	8/03/2019	Team 12	Final Edits and Formatting, Method Names and Descriptions, Diagram Descriptions

Table Of Contents

1. Background & Analysis	7
1.1. Analysis Process	7
2. Roles and Deliverables	11
2.1. Roles – team roles and responsibilities	11
2.2. Deliverables – what we will deliver to the customer	12
3. Project Plan	13
4. Hardware and software platforms to be used:	14
5. Definition of terms	16
6. Solution requirements	18
6.1. Functional Requirements	18
6.2. Non-Functional Requirements	20
7. Other considerations	22
7.1. Assumptions	22
7.2. Constraints and Dependencies	22
8. Software Design	23
8.1 Overview of what the software will do and not do:	23
8.2 Modifications in light of comments made or changes deemed necessary	24
8.3 System Architecture	25
8.4 High level overview of functionality of the system	26
8.5 Package and Class diagrams which show dependencies between components	34
8.6 Dynamic Behaviour of the System - Use Case	35
8.7 Abstract Overview	37
8.8 GUI, Human Interface Views – Describe how the user accesses functionality	40
9. Test Plan	49
10. Resources (List of links to all software/data used in the making of this project)	52
References	53

Purpose/Executive summary

The purpose of this specification document is to outline Team 12's preparation, design and implementation of the app required by the council stated in their project brief. This document will explore our idea and development process as well as the final product we plan to deliver.

Project Brief

The project we have been assigned is the 'Northumberland Tourism Application' which is aimed at providing tourism information to visitors of Northumberland in any method of our choosing. It is for the Council and must not be biased towards any companies or go against the Council's values.

Background & Analysis

This section explores background research into other apps/sites that are similar to the one we have been tasked with creating. We have compiled a list of benefits and drawbacks of each of the apps, allowing us to reach a conclusion as to what type of system we are going to implement.

Roles & Deliverables

Outlines each team members roles and responsibilities which were agreed upon during the creation of our team contract. Our expected deliverables along with deadlines (present on the project plan) have also been listed here.

Project Plan

Overview of Semester 1 and 2 deadlines along with soft deadlines which we have set as a team. These deadlines have been represented in 2 GANTT charts, however small deadlines that may have appeared during the course of development have not been included here.

Hardware & Software

A list of short descriptions of all the hardware and software we have used/plan to use during the development of this application. Examples included software we are using to write the back end code and any API's we are taking advantage of.

Definition of Terms

Simply defines any acronyms or technical jargon which is associated with our app so that anyone can pick up the project and understand each feature we have implemented.

Functional & Non-functional Requirements

This section is a list of functional requirements (i.e. what we want to have implemented by the end of development) and non-functional requirements (i.e. how we are going to measure features working/whether the functional requirements have been met).

Assumptions, Constraints and Dependencies

Any safety concerns, possible constraints with our app and the technologies that we are using, and aspects of the project we are assuming before beginning development. Anything ambiguous in the specification will be explained here, along with how we plan to work around any problems.

Software Design & Implementation

This section contains UML diagrams, design and implementation documentation as well as screenshots of the what the final system should look like and descriptions of how the UI and backend work together.

Test Plan

All aspects of the system we plan to test to prove we have met the specified requirements (and to prove that the system is bug-free) will be shown here. Testing will take place during the spiral development model along with each prototype.

Resources

List of links to all software/data used in the making of this project. Similar to 'Hardware & Software', however anyone reading the specification can now use the same software by following these links.

References

Any external sources, information, software or just inspiration will be correctly referenced here along with the date of access.

1. Background & Analysis

1.1. Analysis Process

Existing Systems

Before we could begin to consider the development process and deliverables of our application, we first needed to perform research of existing systems that already implemented features we planned on including.

TripAdvisor

This system shows user-generated travel and restaurant reviews. The reviews include ratings, images and a Q&A section, as well as other non-user-generated content such as maps and prices. The pages can be filtered and sorted numerous ways such as by location, rating, category, price, name etc. Aside from writing and reading reviews, TripAdvisor can also be used to plan trips with a calendar feature, and communicate with other travellers via forums and private messaging. Its scope is international, averaging 390 million unique visitors a month, as of 2017.^[1.1] It can be used both in a web browser and on mobile devices with a dedicated app.

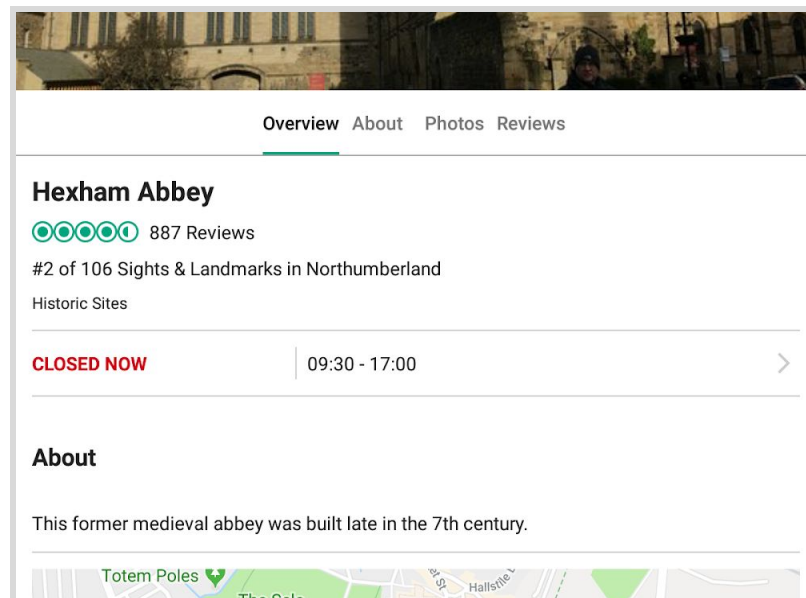


Figure 1, Overview of an attraction on TripAdvisor ^[1.2]

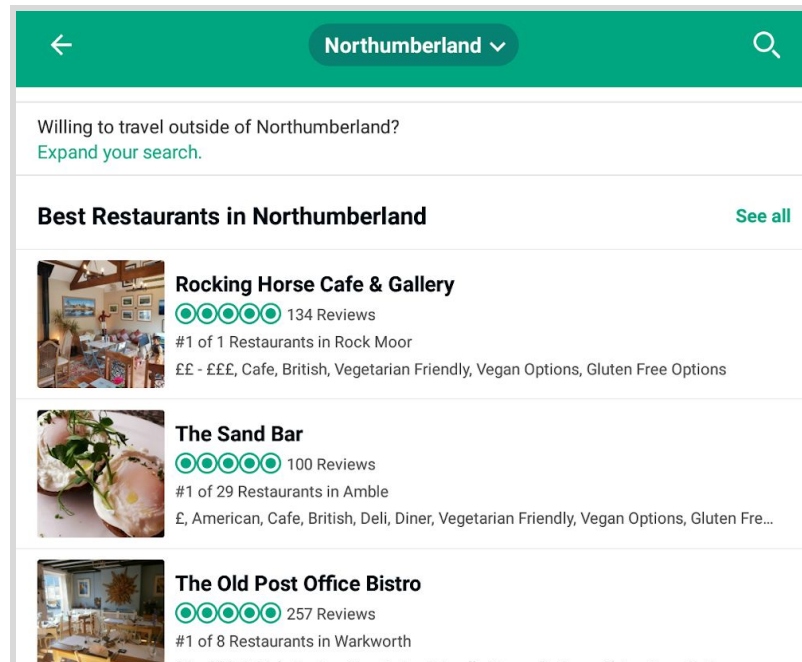


Figure 2, Restaurants in Northumberland ranked on TripAdvisor ^[1.3]

Strengths:

This system meets nearly every requirement. It probably features every attraction that will be included in this project, with a variety of ways to sort and display that information. It also uses Google Maps overlaid with its own interface to show hotels, restaurants, things to do and neighbourhoods. Each is colour-coded and has the ability to be hidden. TripAdvisor has even used Google's Street View to introduce augmented reality on some platforms.^[1.4]

Weaknesses:

TripAdvisor is much larger than this project requires. Anything outside of Northumberland is a distraction, while its collection of attractions within Northumberland is too large. This project aims to create an experience that a user can complete, with an attainable number of achievements.

The fact that TripAdvisor is based on user-generated content also sets the wrong tone; this project aims to present a carefully curated set of attractions at a high quality and in a consistent manner. Figure 1 shows that the 'About' section for Hexham Abbey consists of one short line. This means that a user relies on user reviews and amateur photographs, many of which are very similar, to learn about the attraction.

The requirement of the council remaining neutral to businesses is not met by TripAdvisor. This is shown in Figure 2, where restaurants are ranked from best to worst.

Visit Northumberland

This is a website which describes itself as “the official voice of tourism in Northumberland.” The footer of the site displays the logos of Northumberland Tourism, Northumberland County Council and Visit England. The attractions featured are similar to those featured in TripAdvisor, mostly accommodation and outdoor attractions, which can be browsed by area or activity type.

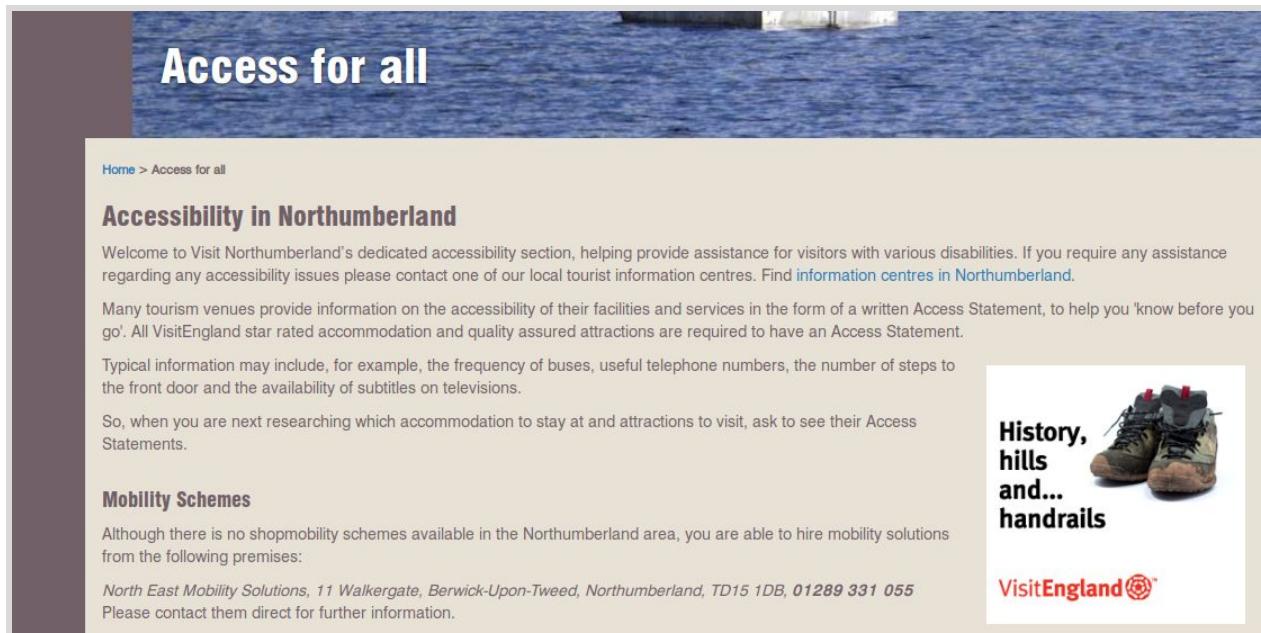


Figure 3, Accessibility information on Visit Northumberland ^[1,5]

Strengths:

Visit Northumberland is intended only for tourism in Northumberland, which leads to a more focused experience and allows for stronger branding. It includes all of the attractions that will be included in this project.

Each attraction’s page includes several paragraphs of information and sometimes a video. This means that a user can learn about the attraction without relying on user-generated content. There are links to TripAdvisor reviews, but these are not prominent or necessary to use the site. The photography used on Visit Northumberland is contributed by organisations and professional photographers. As a result, Visit Northumberland resembles a guidebook more than TripAdvisor does, which is closer to what this project aims for.

There are substantial resources for accessibility, as shown in Figure 3, which is something we will include in our app.

Weaknesses:

The website does not have an application counterpart for mobile devices which could reduce its audience.

The main purpose of the website seems to be to plan and research before embarking on a trip. The website is not interactive and does not have much purpose during or after a trip, except to look at the “also of interest” section at the bottom of an attraction’s page. Our app is intended to accompany users on their trip, and give an achievement for each successfully completed trip.

Pokémon Go

This is a free to play mobile game which uses augmented reality and the device’s GPS to place the titular characters into the user’s view of the real world (Figure 4), where they can be captured, trained or battled. As of 2018 it has 147 million active monthly users.^[1.6]



Figure 4, Augmented Reality in Pokémon Go ^[1.7]

Strengths:

Pokémon Go has been praised for promoting exercise.^[1.8] The use of AR combined with an achievement system is something which we recognised would work well with a tourism app.

Weaknesses:

This app does not meet most of the requirements. It can be used in Northumberland but does not put any emphasis on tourist attractions. The focus instead being on the game’s characters. Many accidents and some fatalities have cited Pokemon Go as being a contributing factor.^[1.9] We will select our achievement locations carefully to avoid danger to the users or being a nuisance to other tourists.

Summary of existing systems	
Strengths	<ul style="list-style-type: none"> ● Game elements (achievement system) ● User-friendly interface ● Can be used while outdoors, not just for planning ● Attraction information positive and consistent, i.e. not user-generated
Weaknesses	<ul style="list-style-type: none"> ● No mobile app ● Collection of attractions/achievements too large ● Not focused on Northumberland ● Not neutral to businesses

1.2. Project purpose

The aim of this project is to create a functional and reliable tourism application for the county of Northumberland. The application will provide navigation functionality and provide information on points of interest for users visiting the area. Locations featured will include natural landscapes, historical sites, museums, towns and villages.

The application will represent the Northumberland County Council and so will remain neutral when listing businesses and sites. This means there will be no specific promotions or weights for any listings.

The application will include Augmented Reality as one of its features where when a user has successfully navigated to their destination, an option will appear on the screen that will allow the user to augment a card display of the site's information on their device.

An achievement system will also be in place for users to have an additional incentive to use the application. These achievements will include various tasks to acquire e.g. reaching a certain site, visiting a certain number of sites etc. Social media integration will also be within the application to allow users to share their experiences using the app and also any achievements they acquire.

A website will be developed in conjunction to facilitate the user experience. The website will contain pages of additional information of included locations, references used, a contact us page, and the documentation for the app. Furthermore, we wish to implement a page for users to view and track their achievements online.

2. Roles and Deliverables

2.1. Roles – team roles and responsibilities

Team Leader

(Ethan Wormley)

The Team Leader will ensure all members are contributing and up to date on the project. His responsibilities also involve chairing and organising all informal and formal meetings. In addition, Ethan will attend all Team Leader meetings and submit Team Leader reports, as well as keeping minutes, creating agendas and providing useful information to the team.

Software Developers

(Jamie Child, Sam Holley, Robert Adamson)

Jamie Child has been appointed the Head of Software Development. The team will develop the back-end of the application due to their wider experience and coding knowledge. They are also responsible with developing the main functionality of the application.

Design and UI Developers

(Michael Surya, Calum Harvey, Aleksandar Gavazki)

Michael Surya has been appointed the Head of UI Design while Calum Harvey will head Design Research. The team will design the appearance for the application including colour schemes, UI elements and general appearance for each application page.

Documentation Team

(Calvin Hudson, Stefan Lip, Ethan Wormley)

Calvin Hudson has been appointed the Head of Documentation. If the Team Leader is absent, Calvin will assume the role of Team Leader. Developers will contribute to documents, but it is the Documentation Team's responsibility to ensure they are clear and accurate.

Testers

The documentation team will conduct Black-box testing onto the app through entering various inputs into fields to test for validity checking. UI elements will also be tested for accuracy and responsiveness. Implementation of methods and functions will be tested through test cases within White-box testing where each developer will test their own code. Additionally, the front-end and back-end teams will also test each others code to improve chances of finding unwanted bugs. Any errors or bugs discovered will be documented and reported for debugging.

General Responsibilities

All team members will conduct research for their own sections including coding knowledge and background information for the project. Research will be stored on to shared drives to aid other members. Members are also aware that in the case of absent members or work that is tasking on specific teams, they are expected to take on additional responsibilities. Members can also shadow other team members to gain experience in their fields for future projects if they wish.

2.2. Deliverables – *what we will deliver to the customer*

Final Design Document

Documentation detailing the specification and its solution will be submitted. This document should explicitly communicate our application in terms of its features and implementation of said features. It is imperative for this final design to successfully incorporate requirements set out by the customer. The final design document will be submitted on the 8th March.

Android Application

A mobile application prototype for Android devices will be delivered as the main product of this project. This application will serve as a navigation tool with entertainment features for tourists visiting Northumberland. The system will be delivered as an APK file by 29th April.

Corresponding Website

A series of webpages will also be published in addition to the application as a complementary component. The website will feature help articles, contact-us pages and also additional related links giving more information about attractions that should benefit the end users. The website should also have functionality linked to the application, in the form of achievement tracking.

Testing

Test plans will be submitted detailing the methods and approaches applied throughout the project. Approaches used, for example, will include White-box and Black-box testing for the application. Testing will occur throughout the project as soon as prototypes are developed. Test reports will be produced to document bugs and areas for improvement. The final test plan will be submitted on the 8th March.

Submission and Demonstration of Final System

The final submission date for the system is the 29th April. The demonstration of the system will occur on the 9th of May. The final system will be fully functional and should not include any bugs that any team members are aware of.

Other

For this project, since we have decided upon using the Spiral Model as our development process model, for each new feature and iteration, we will arrange a meeting with the client to present the feature.

3. Project Plan

Semester 1:

Semester 1	18/10/2018	22/10/2018	26/10/2018	30/10/2018	03/11/2018	07/11/2018	11/11/2018	15/11/2018	21/11/2018	28/11/2018	04/11/2018	07/11/2018	12/11/2018
Team Roles/Structure													
Team Contract													
Draft Requirements Specification													
Project Aim													
Expected Deliverables													
Functional & Non-functional Requirements													
Project Constraints/Risks													
Design Ideas/ UI Mock-ups													
Implementation of Main App Components													
Diagrams/Control Flow of each area													
Implementation of corresponding Website													
Test Plan													
Dragons Den Preparation													
Peer Review													

Semester 2:

Semester 2	Christmas Break	28/01/2019	08/02/2019	15/02/2019	22/02/2019	28/02/2019	06/03/2019	08/03/2019	14/03/2019	21/03/2019	29/03/2019	Easter Break	29/04/2019	02/05/2018	03/05/2018	05/05/2018	09/05/2019	
Android Studio Practice																		
Back End Development																		
AR Research																		
Map API Implementation																		
Database Research/Implementation																		
Front End Development																		
Screen Layouts																		
Social Media Integration																		
Help Screen/Tutorials																		
Settings Menu																		
Additional Features (e.g Website)																		
Specification																		
Improvements from feedback																		
UML and Structure Diagrams																		
Trade Fair Poster																		
Make Rough Poster Designs																		
Final Poster																		
Trade Fair Preparation																		
Testing																		
Front End																		
Back End																		
User Testing																		
Technical Demo Preparation																		
Peer Review																		
Meetings With Client (Every Friday)																		

4. Hardware and software platforms to be used:

Software Utilised:

Assets

- GIMP
- Google Documents
- Photoshop

Website

- Google Chrome
- Mozilla Firefox
- Notepad++ 7.3
- Other IDEs preferred(e.g. Brackets, Atom, etc.)

Mobile App

- Android Studio 3.2.1
- Android Studio 3.3.1
- Google Directions
- Google Firebase
- Google Maps
- Gradle 5.0
- MapsForge 0.8.0
- Mockplus
- Open Street Map
- Overpass API 7.55

Hardware Utilised:

Website

- Stable internet connection
- Modern desktop or laptop running modern operating system

Mobile App

- Android devices running android 6.0 or higher
- A combination of internet connected devices and offline devices for testing local storage of maps
- Motorola G5 for showcasing and thorough testing.

Desktops/Laptops Developing the system:

- AMD Ryzen 5 2600 @ 3.4 GHz,Radeon RX VEGA 64, 8GB Installed RAM
- Intel(R) Core(TM) i7-4790K CPU @ 4.00GHz, 4001 Mhz, 4 Core(s), 16GB Installed RAM, Geforce GTX 1060, Microsoft Windows 10 Home
- Intel Core i5-4460 CPU @ 3.2GHz 3201Mha 4 cores, 16GB Installed RAM, NVidia Geforce GTX 970, 3840x2160 + 1080x1920, Displays, Windows 10 Home OS (64 bit).
- Intel Core i5-7200U CPU 2.5ghz, 8GB installed RAM, Intel HD , Graphics 620, 1920 * 1080 display, Windows 10 Home OS (64 bit)
- Intel Core i7-7700HQ CPU 2.8ghz, 16GB RAM, Geforce GTX 1060, 1920*1080 display, Windows 10 Home OS (64 bit).
- Intel i7-7700 CPU @ 2.80GHz, 8GB RAM , Nvidia GTX 1050.

Android Emulators:

- Nexus 5X API 28 1080x1920 pixels , 2.26GHz quad-core processor , 2GB RAM.
- Pixel 2 API 24 (with google play services)
- Pixel 2 XL 1440x2880 pixels, 4GB RAM, Snapdragon 835, 3520mAh, API 28
- Pixel API 28 5.0" 1080x1920 pixels, 4GB RAM, Snapdragon 821, 2770mAh.

Android Devices:

- Motorola G 5, Android 7.0, Qualcomm Snapdragon™ 410 (MSM8916) processor with 1.4 GHz quad-core CPU, 3GB RAM
- Oneplus 3 (a3003), Android 8.0.0, 6GB RAM

- Samsung Galaxy S8, Android 8.0.0, 4GB RAM

5. Definition of terms

Application Programming Interface / API - A pre-defined set of functions and tools which can be called upon for functionality and behaviour when building software.^[5.1]

Android Application Package File / APK - A file, with an .apk extension, which runs an Android application on an android device. It is a compressed file which contains the code files, the manifest file and the application resource files.^[5.2]

Augmented reality / AR - To superimpose virtual information (words, images, sounds) over a view of the real-world.^[5.3]

GIT - A multi-user development system that allows teams to upload and manage projects, whilst keeping track of updates and versions. Commonly used in open-source development as it allows for version branching and complete back-tracking.^[5.4]

Library - A set of pre-compiled routines - such as new methods or data types - used by developers in programming to carry out frequently performed tasks in a program more easily.^[5.5]

Nav-bar - A part of the UI that allows users to access various important areas of an application or system quickly and easily. Typically a bar on the top or on the left-hand side of the user display listing subheadings for navigation, that is either always displayed or can be made to be displayed at any time.^[5.6]

OpenStreetMap / OSM - A project which creates a free map of the world, used on thousands of websites and mobile apps. It is built and maintained by a community of mappers.^[5.7]

Query - A request for information from a database, either by a Structured Query Language or means specific to that database.^[5.8]

User Interface / UI - The way in which a user interacts with software or hardware. This can be visual, called a graphical user interface, or text-based, called the command line. Graphical user interfaces are very common and include windows, toolbars and icons.^[5.9]

XML - Extensible Markup Language, a language that defines rules for encoding documents (such as web-pages or graphical UIs) that are both human-readable and machine-readable.^[5.10]

6. Solution requirements

6.1. Functional Requirements

Requirement	Priority (H, M, L)	Supplier Compliance (Full, partial or will not be delivered)	Supplier Comment
Mobile Application			
1.11 GPS will be used to track the users location while using the app	H	Full	Necessary to calculate a route.
1.12 The application will provide a list of the closest attractions to the user	H	Full	Users are more likely to visit close locations to obtain achievements.
1.13 The application will show an updated map of the user's current location	H	Full	Necessary so user knows how close they are to the attraction.
1.14 The application will give the users directions to the attraction they have selected	H	Full	Without this the user will not be able to find the attractions.
1.15 The application saves attractions the user has been to, in order to track achievements	M	Full	Necessary in order for there to be a trackable element to the app
1.16 The user will be able to create an account on the app	H	Full	Important as will allow them to track their progress.
1.17 The application will allow the user to sign in using google, facebook or by using an account created on the app.	H	Full	Need accounts in order to save data/progress

1.18 The application will allow the user to view the achievements they have unlocked	M	Full	Gives incentive to keep users coming back to the app.
1.19 The application will be linked to a web-based system that we will implement	H	Partial	Will definitely be a corresponding website, however may or may not have functionality.
1.20 The application will show the user the distance to the attraction that they want to go to	M	Full	Important as gives the user help on which attractions to choose.
1.21 The user will be able to specify the mode of transport for their journey	H	Full	Walking routes will be different to road routes
1.22 The application will display the estimated time to the destination	M	Full	Important as this gives the user help on which attractions to choose.
1.23 When the user reaches the destination, the application will allow the user to use AR	H	Full	One of the main features of the app. Makes it more interactive
1.24 When in AR, if the user is facing the attraction, more information will be displayed	M	Partial	Will be displayed after button click, if phone is pointing in direction of location
1.25 The application will allow the user to search for locations	M	Partial	Time constraint will depend on how extensive the search function is.
1.26 The application will allow the user to view previously visited locations	M	Partial	Still deciding whether to store this information locally, so recent places may be

			cleared after closure of app.
1.27 The application will show attractions with disabled access	H	Full	Ensure that the application is accessible to all
1.28 The application will notify the user not to use the app while driving	H	Full	Safety concern which must not be forgotten
Web Application			
1.29 The user will be able to create an account on the website	H	Full	Important as will allow them to track their progress.
1.30 The user will be able to login with google, facebook or by using an account on the app	H	Full	Same sign in methods as the application itself
1.31 The user will be able to view their achievements	M	Partial	Will depend on time constraints
1.32 Possibly show other stats, like distance travelled?	L	Partial	Again depends on time constraints

6.2. Non-Functional Requirements

Requirement	Priority (H, M, L)	Supplier Compliance (Full, partial or will not be delivered)	Supplier Comment
1.1 Software Documentation	H	Full	Necessary to allow people unfamiliar to understand the application.

1.2 Mobile Device	H	Full	Needed to run the application as minor functionality exists just on the web page.
1.3 Wi-fi or Mobile Data	H	Full	Necessary to calculate routes/sign in and unlock achievements
1.4 Transitions between screens will be fast and fluid	H	Full	Needed to keep up to date with current applications on the marketplace
1.5 User will be able to look around the map while a route is being calculated	M	Full	If possible, this means the application won't be frozen while waiting for route calculation
1.6 User will become familiar with the app after following the tutorial	M	Full	Allows for less technologically savvy people to use the app easily.
1.7 Passwords will be encrypted when stored	H	Full	Will attract more users as their data is secure.
1.8 The map can be changed without rebuilding the app	H	Full	Saves time and allows for fast route calculating and changing.
1.9 The app should load within 20 seconds	H	Full	Maximum time to load as otherwise users may think it has frozen and close the app/uninstall
1.10 The app should calculate a route within seconds	M	Full	Any longer and users may abandon the route/close the app.

7. Other considerations

7.1. Assumptions

There are some assumptions we needed to make when developing this application...

- The users are assumed to be people exploring the Northumberland area.
- Most users are assumed to have a Facebook or a Google account in order to register to the app.
- They must also have an android device with an OS equal to or greater than “Marshmallow”.
- GPS and mobile data must be enabled and stable in order for the app to work as intended/not malfunction.
- For the majority of the app we are assuming that users can reach all locations, however disabled access locations can also be selected if this is not the case.

7.2. Constraints and Dependencies

- The application uses GPS to track the user’s movement and generate relevant information about the location he/she is currently at. Like every technology, GPS is not perfect and sometimes small mismatches with the actual location may occur.
- The app heavily relies on a stable internet connection in order to accurately track the user and not misguide him/her.
- Another constraint is that the offline map requires a large amount of storage/data to download. This should therefore be made optional.
- People with disabilities may have problems with some of the routes/locations and problematic places must be indicated before the user chooses a route.

8. Software Design

8.1 Overview of what the software will do and not do:

What the software will do	What the software will not do
<ul style="list-style-type: none"> The app will allow users to sign up with email or log in via social media accounts. 	<ul style="list-style-type: none"> The app will not post on the user's social media without the user's consent.
<ul style="list-style-type: none"> The app will allow users to log out of their accounts. 	<ul style="list-style-type: none"> The app will not ask users for sensitive information such as postal addresses or mobile numbers.
<ul style="list-style-type: none"> The app will use the user's location data to show where the user is on the map. 	<ul style="list-style-type: none"> The app will not share the user's location.
<ul style="list-style-type: none"> The app will allow the users to find nearby attraction spots within the location range of Northumberland. 	<ul style="list-style-type: none"> The app will not favor any business or location over another.
<ul style="list-style-type: none"> The app will use the augmented reality feature to show information regarding the attraction spots whenever the users are in the vicinity. 	<ul style="list-style-type: none"> The app will not provide any information/directions to any spots outside of the Northumberland area.
<ul style="list-style-type: none"> The app will allow users to store map data offline through downloads. 	<ul style="list-style-type: none"> The app will not share or use any information it does collect from the user.
<ul style="list-style-type: none"> The app will show whether a location has accessibility considerations in their descriptions. 	<ul style="list-style-type: none"> The app will not put the user in harm's way on route to/at an attraction
<ul style="list-style-type: none"> The app will unlock and track achievements based on the users app usage. 	<ul style="list-style-type: none"> The app will not ask the user for any money/there will be no paid boundaries, so all features of the app will be accessible.
<ul style="list-style-type: none"> The app will provide users routes via arrows/directions to their chosen location for walking or car. 	<ul style="list-style-type: none"> The app will not take up any additional space once it has been installed.

<ul style="list-style-type: none"> The website will feature a user guide that will include step-by-step instructions on how to use the app. 	<ul style="list-style-type: none"> The website will not store or use any login information, similarly to the app
<ul style="list-style-type: none"> The website will contain contact information which will allow users to get help/support. 	<ul style="list-style-type: none"> The website will not be required for users to use all functionality of the app.
<ul style="list-style-type: none"> The website will allow users to view and track their achievements online. 	<ul style="list-style-type: none"> The website will not provide navigational functions.
<ul style="list-style-type: none"> The website will showcase a list of links to provide users with additional information on locations. 	<ul style="list-style-type: none"> The website will not be drastically different depending on the browser.

8.2 Modifications in light of comments made or changes deemed necessary

AR CHANGES

In the initial prototype, when a location was viewed in AR mode, a detailed description of the attraction took up a large area of the screen. It was decided that users would want to be able to navigate towards the attraction with an unobstructed view. To achieve this, the AR mode was modified to include only a notification if the achievement for that attraction has been unlocked, and a button labelled 'find out more' which opens another page containing a full description. This button was also added to the page that tells the user an attraction's distance from their current location. The full description that was previously on this page has been replaced with a short one so that a user does not have to read much before deciding to travel to a particular attraction.

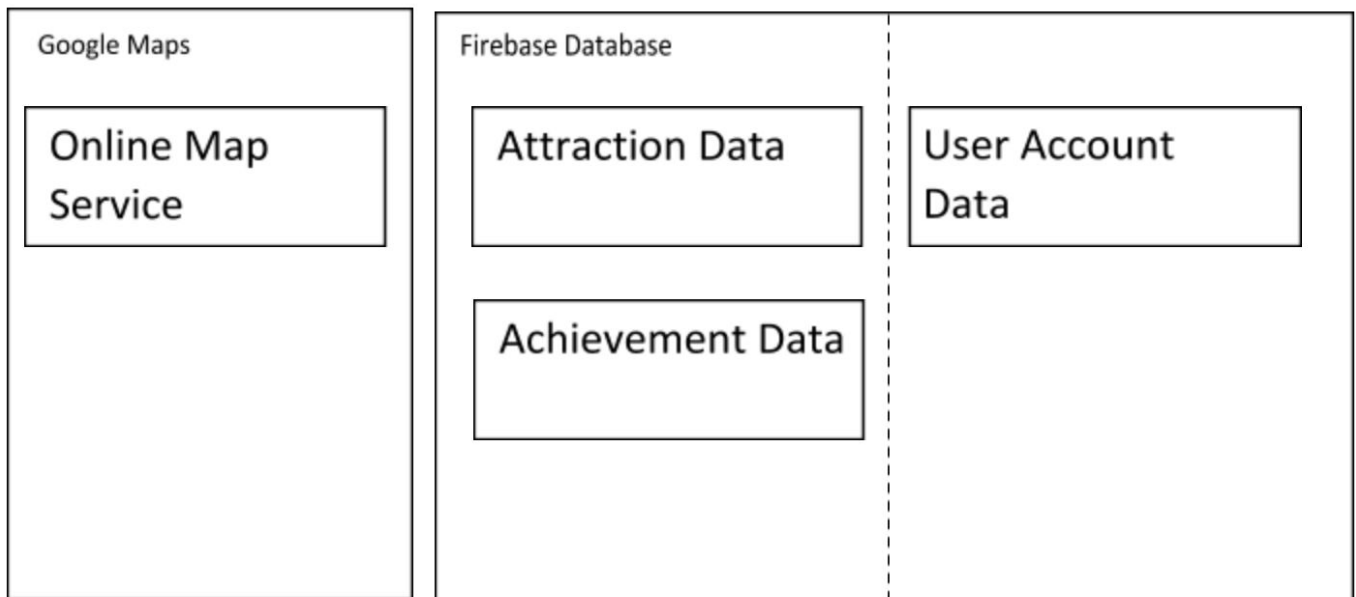
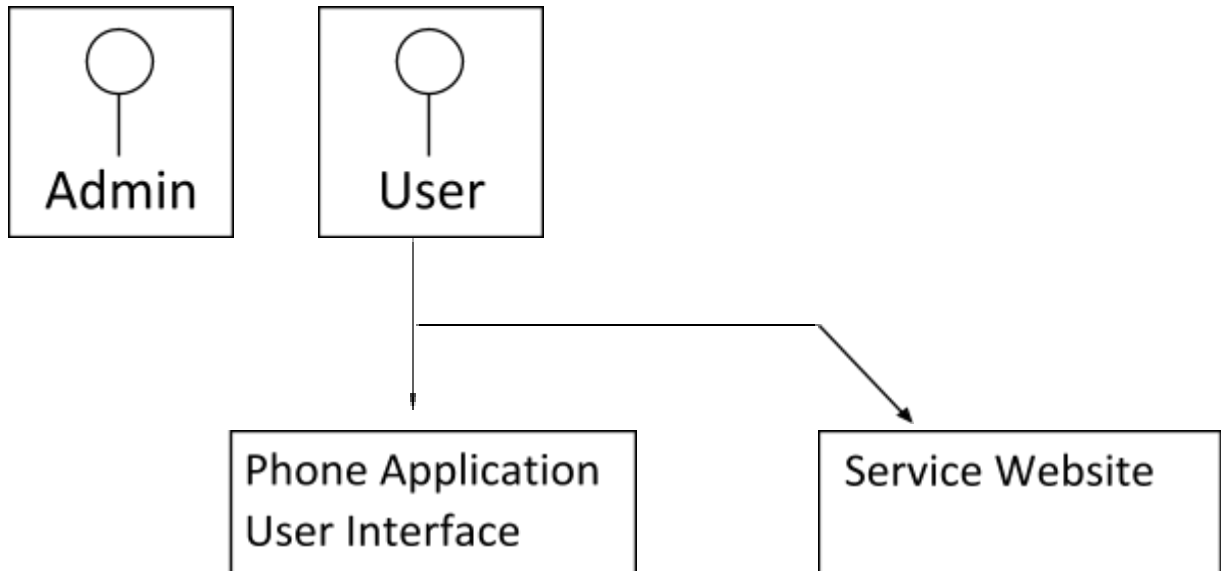
LOCAL MAP STORAGE

As the app is intended to be used in many rural locations without WiFi, the initial plan was to have the app download the map of Northumberland to the user's phone. This would enable its use without any network connectivity. However, we received feedback that this may be unwanted by some users as the map would take up a lot of storage on a device perhaps more space than a user has. In response to this it was decided to make the downloadable map optional.

COLOUR SCHEME

The original red and yellow colour scheme we wanted to include (relating to the colours of the Northumberland flag) was deemed too bright and in some cases meant that the text on screen was difficult to read. We have now changed the colour scheme to be that of grey and light blue, with hints of red and yellow dotted throughout the design to still represent Northumberland.

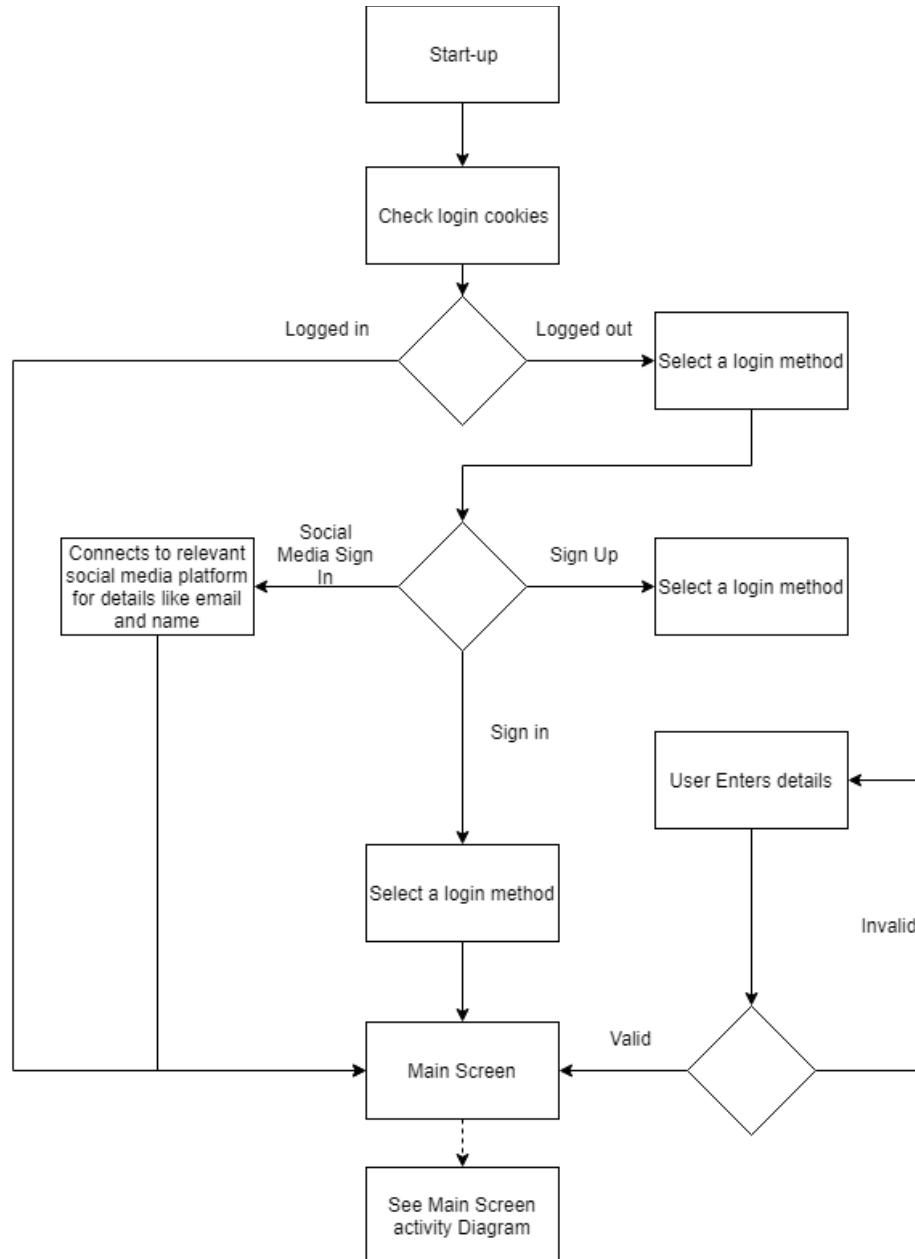
8.3 System Architecture



Users have the option of using the mobile application or the partner website (which will hopefully contain functionality). Google Maps has been included for the online map service, however if 'Open Street Map' can be used then this would be preferable.

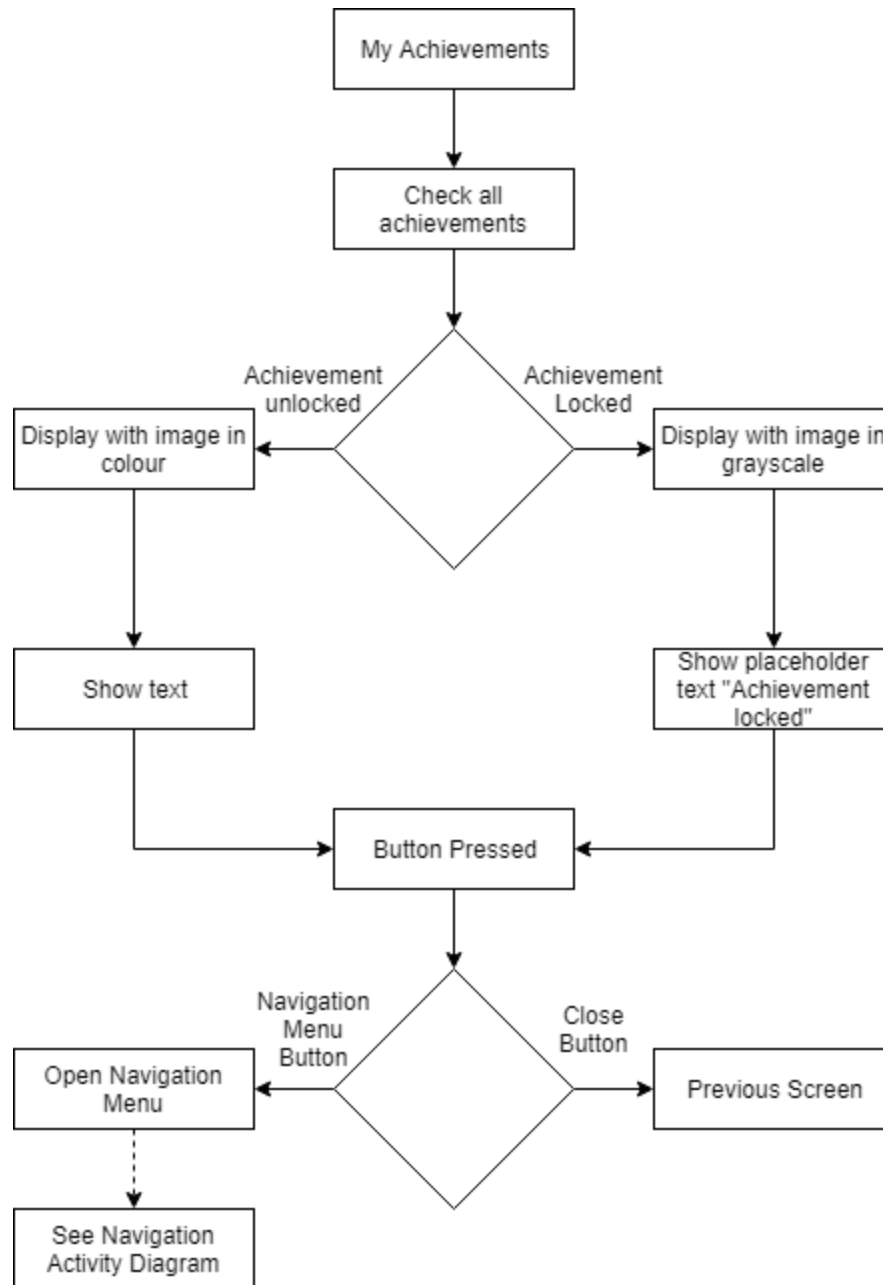
8.4 High level overview of functionality of the system

Figure 8.4.1 Login Flow Diagram



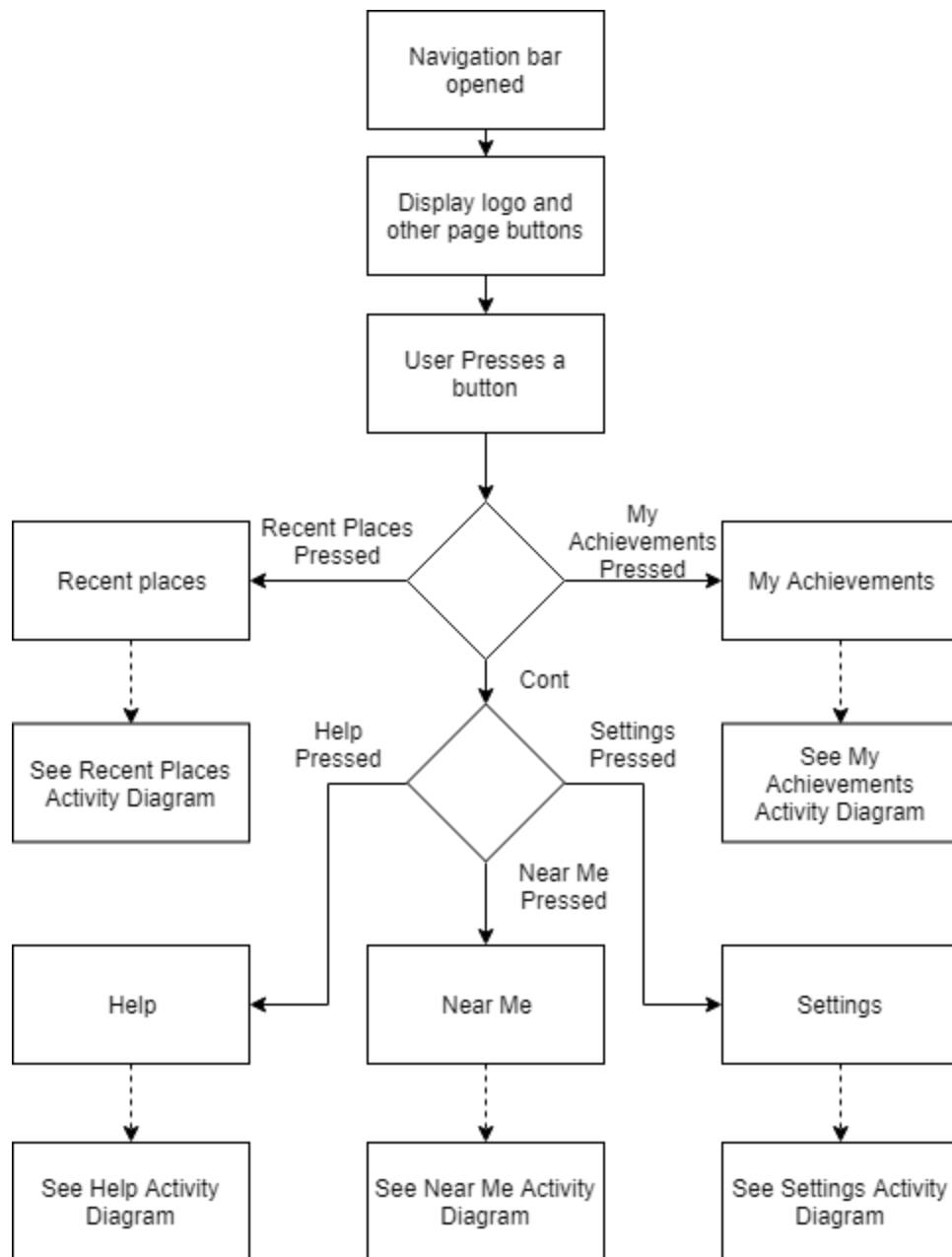
This diagram shows the login process that occurs on startup. The program will check cookies to see whether or not a user is already logged in, and if not asks the user to choose a login method. This can either be a direct login, or a login through a social media platform. If a direct login is chosen, the user will need to enter login details which will need to be validated before the user is logged in.

Figure 8.4.2 My Achievements Activity Flow Diagram



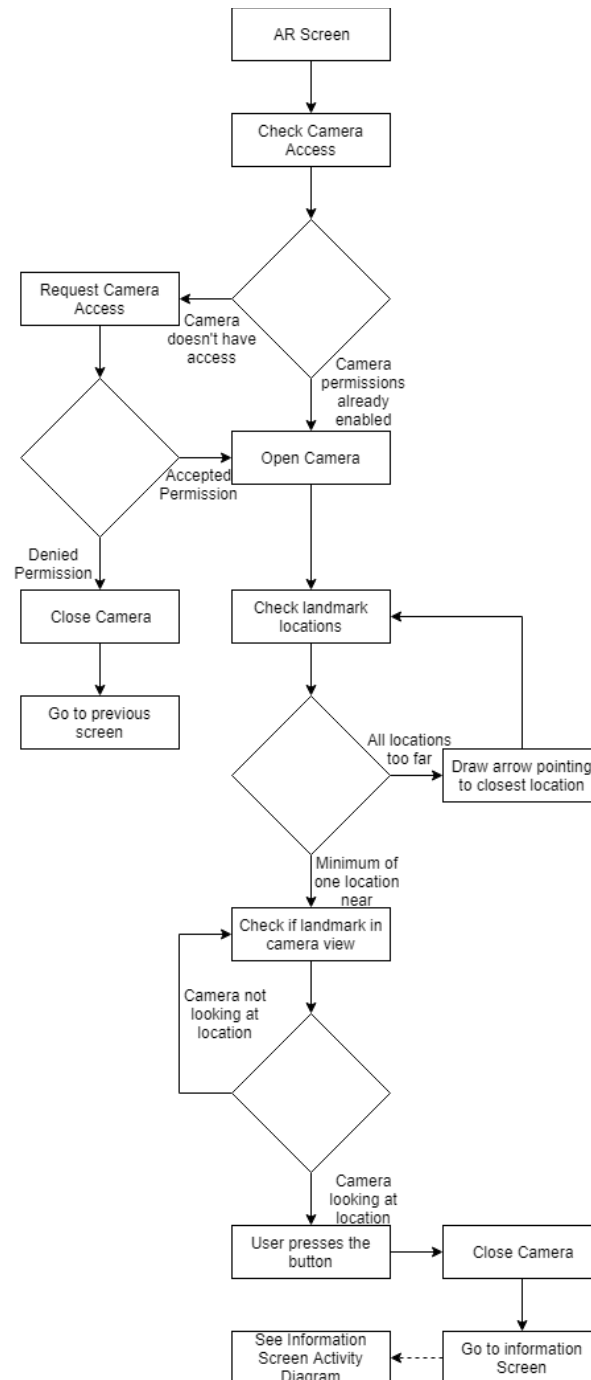
This diagram shows the process of accessing the My Achievements screen. When this happens the program checks for each achievement whether or not it has been unlocked by the user, and shows the appropriate display (in colour or grayscale). From My Achievements, the user can also access the Navigation Bar.

Figure 8.4.3 Navigation Bar Flow Diagram



This diagram shows the options available when a user accesses the navigation bar. From the navigation bar, other parts of the program can be reached by the user quickly. This diagram links to the corresponding diagrams depending on the activity selected in the menu.

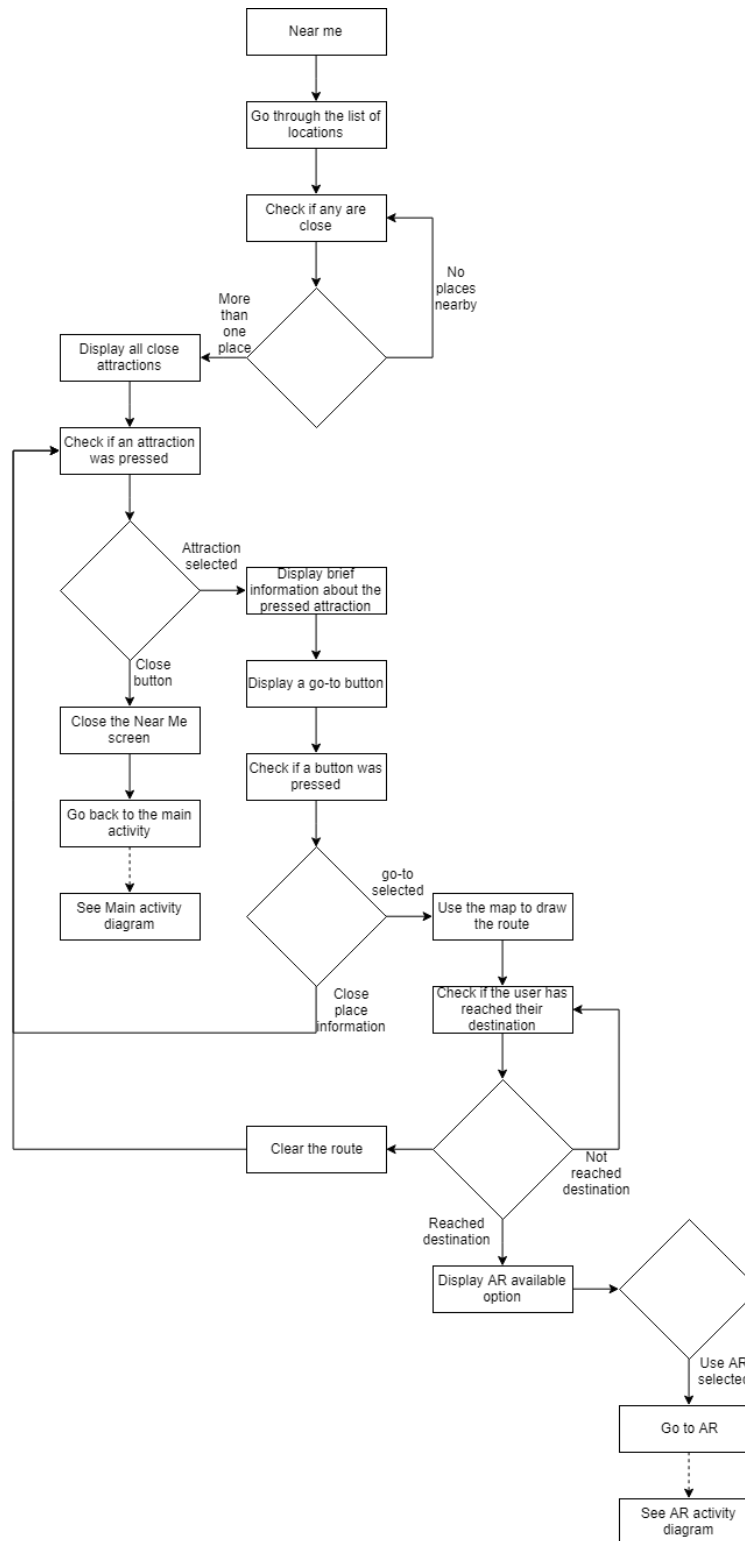
Figure 8.4.4 AR Activity Flow Diagram



This diagram shows the process of initialising and running the in-app AR. The program will check whether it has been granted permission to use the camera, and if not request permission.

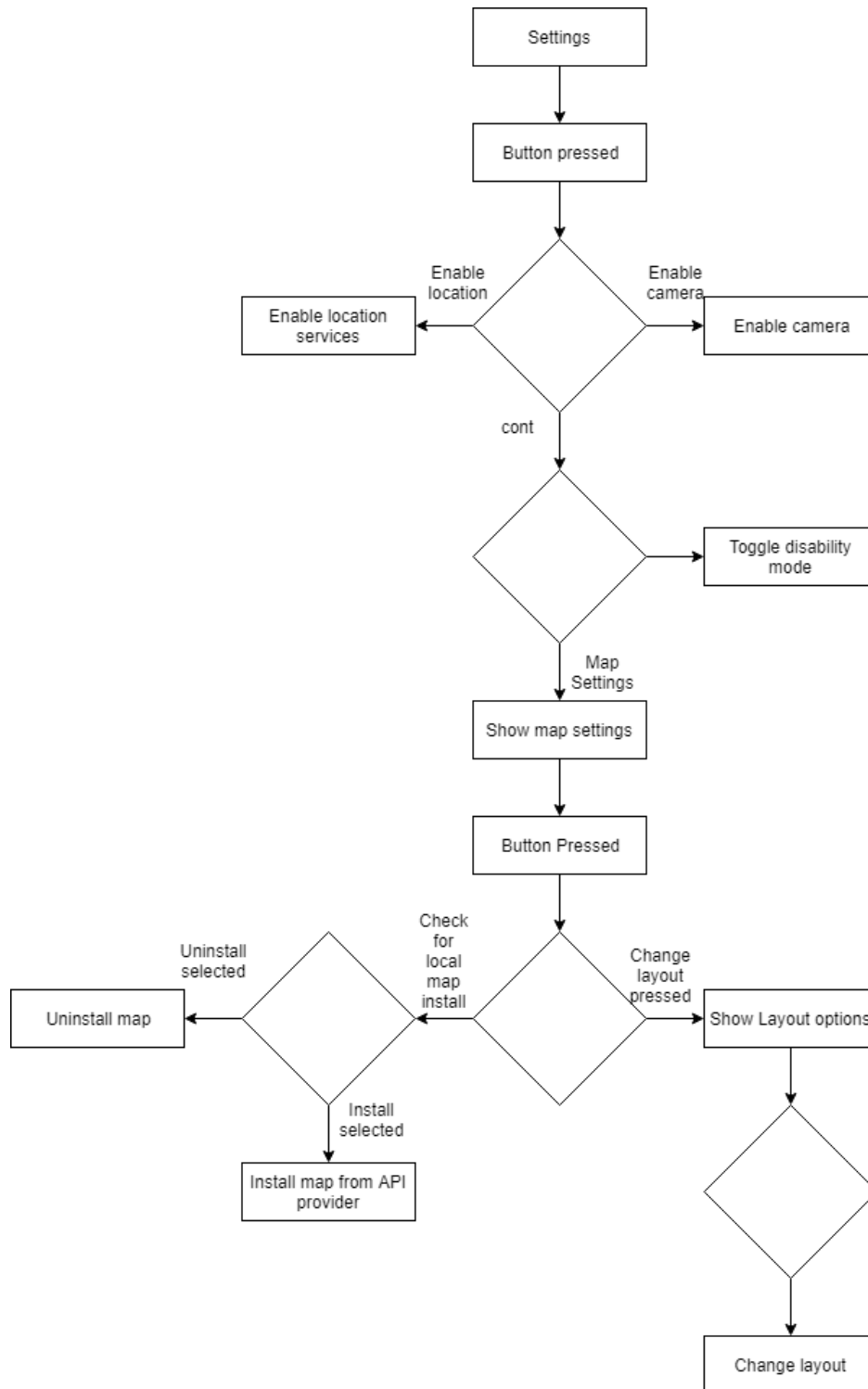
If it is denied the user will be sent back to the previous menu. When the camera has been opened the program will check if there are any locations within a given range, and if not will display an arrow pointing to the nearest location on the camera screen. If the user is in range of a location, and the user is pointing the camera at it, the program will give the user the option to press a button for more information. This will close the camera.

Figure 8.4.5 Near Me Activity Flow Diagram



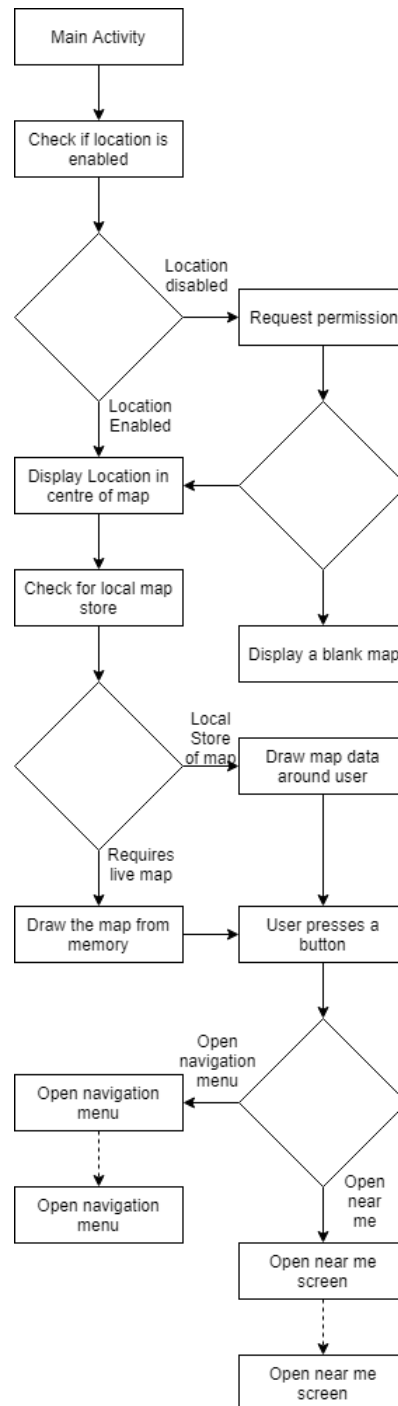
This diagram shows the Near Me process, which checks whether any locations are near the user's current location. If there are locations nearby, the user will have the option to view information and directions to the location.

Figure 8.4.6 Settings Activity Flow Diagram



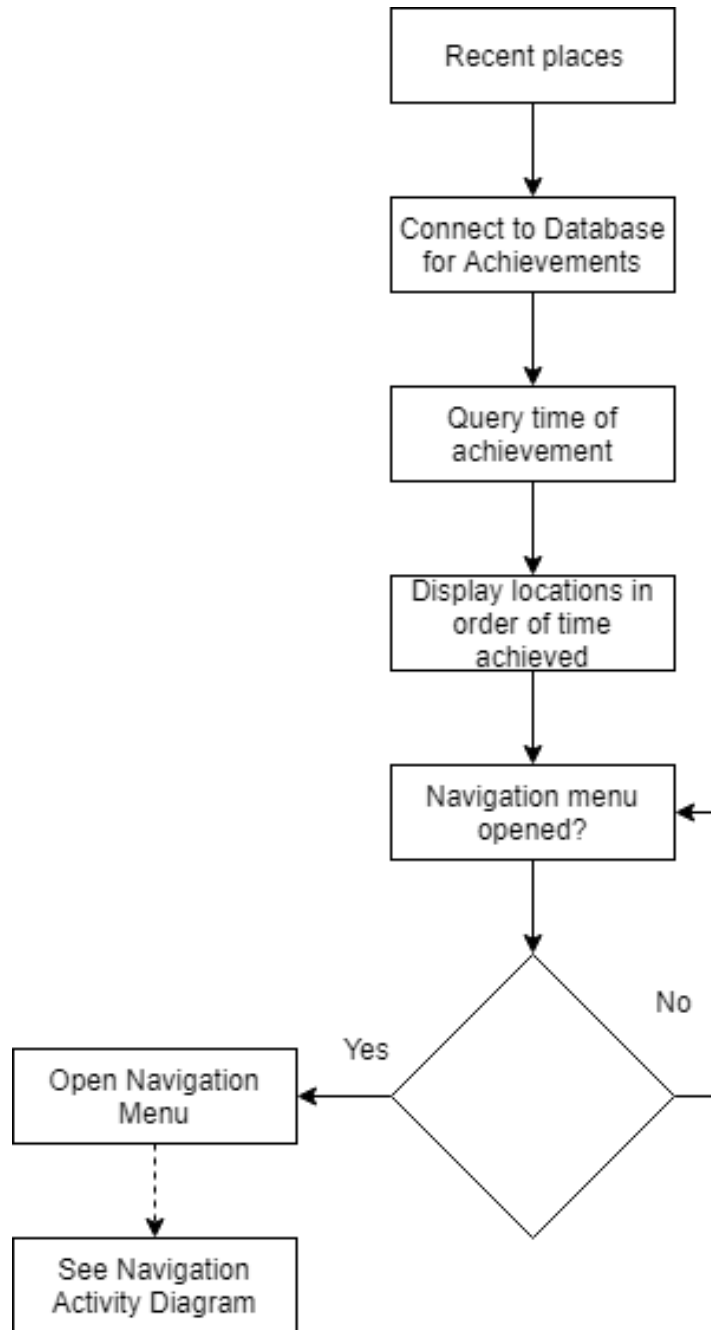
This diagram shows how the various options are accessed in the Settings menu. Map settings - where the option to install the downloadable map for offline is located - has its own separate sub-menu.

Figure 8.4.7 Main Activity Flow Diagram



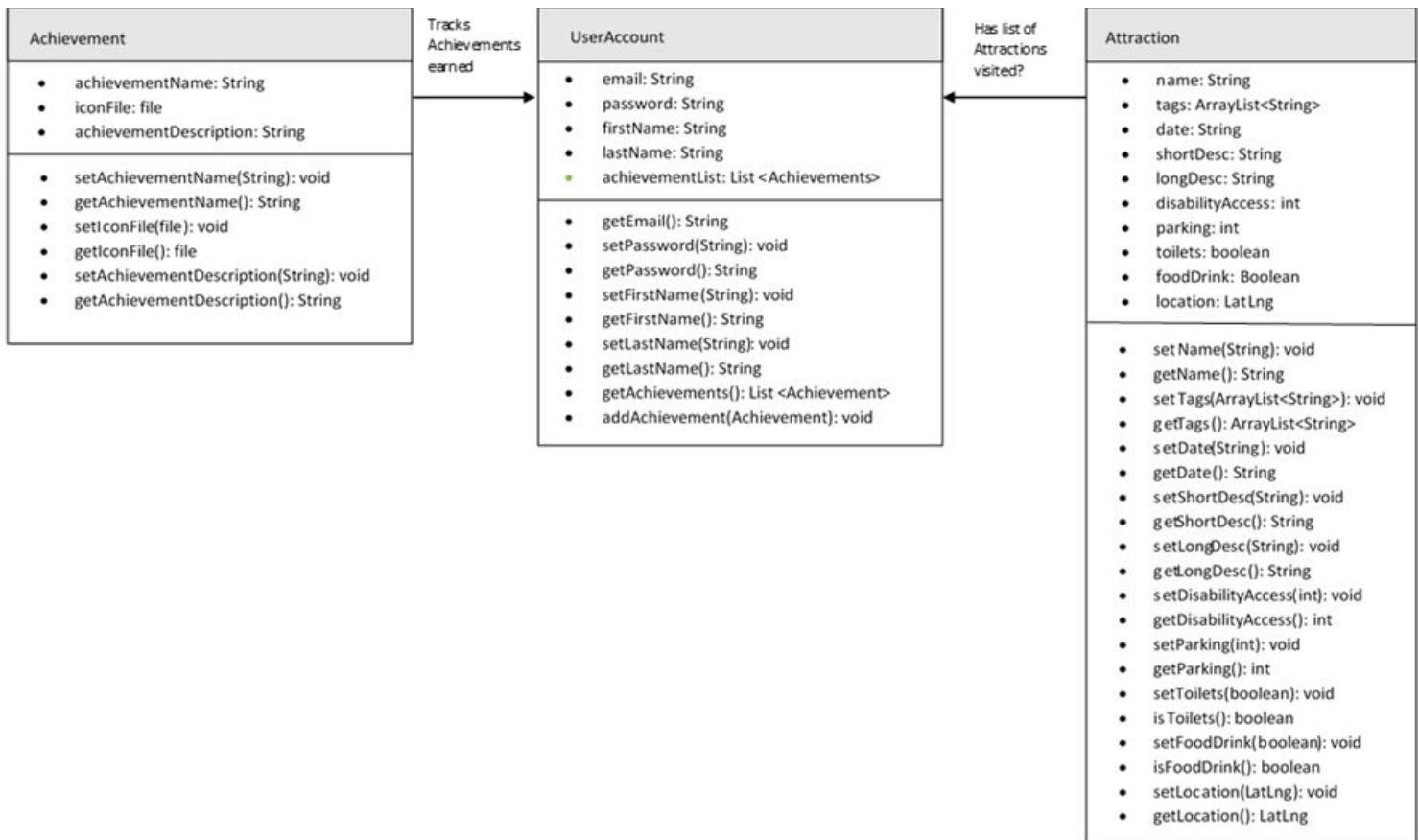
This diagram shows the functionality of the main screen of the app. If location is enabled the user will be able to see themselves on the map. The map information will be loaded depending on whether they have the downloadable map installed. The near me button will display nearby locations to the user (near me screen), and the nav bar can be opened to navigate to alternate screens shown in other diagrams.

Figure 8.4.8 Recent Places Flow Diagram



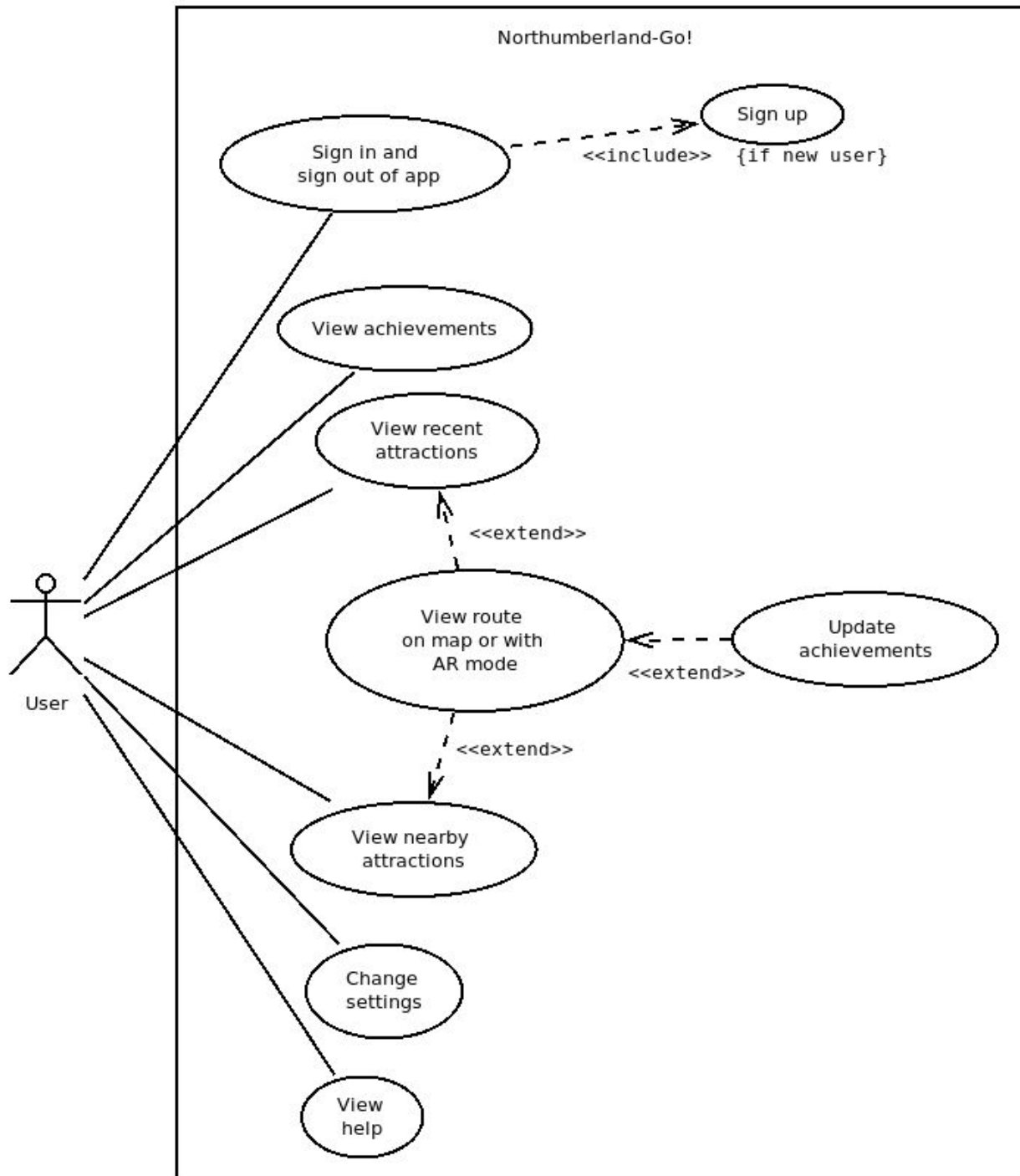
The diagram shows how we will be utilising the achievements system, along with the time that the achievements were unlocked, to find out the order in which the attractions have been visited by the user, so that we may order said attractions on screen.

8.5 Package and Class diagrams which show dependencies between components



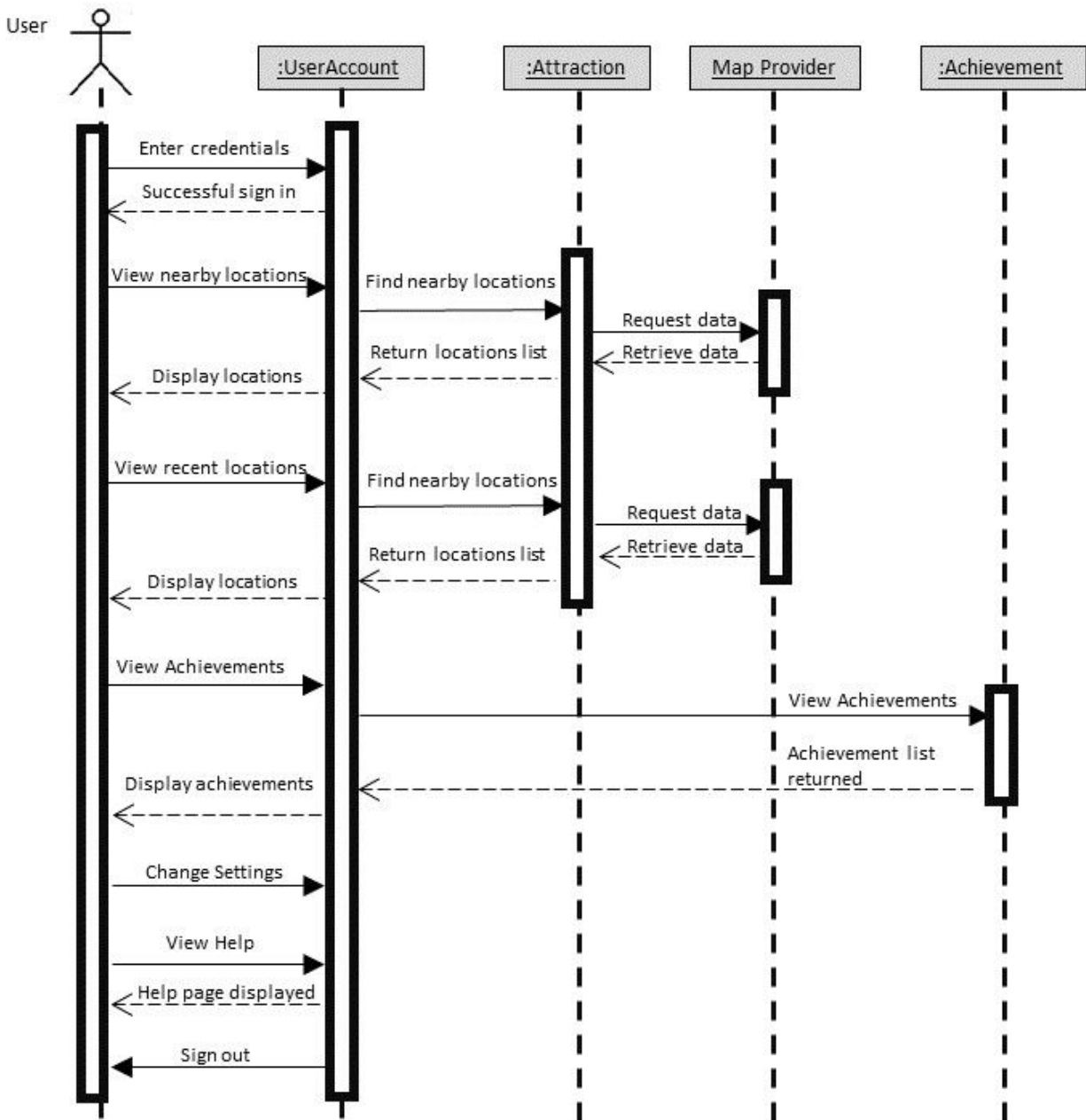
This basic class diagram for our system shows the 3 core components. A User Account, storing basic information such as the users email and password (which will be hashed in the database), as well as the achievements they have locked/unlocked. User Accounts may also store a list of attractions visited, however this may be saved locally for each instance of the app/could just be determined from the achievements they have unlocked. Attractions will have 'fact file' style information associated with them, as well as the lat/lng information for determining their location. Finally, each Achievement will store its icon, title and description (of how to achieve it).

8.6 Dynamic Behaviour of the System - Use Case



Use Case diagram which demonstrates the different options a user has when using the application. The main activity/map screen is interlinked with recent places and nearby attractions, and the users progress is continually tracked, hence the `<<extend>>` to Update achievements.

Dynamic Behaviour of the System - Sequence



The UML sequence diagram combines the main interactions (Use Case diagram) between the different objects (Class diagram) over a given period of time. The Map Provider, which was not seen in the previous diagrams, acts as an external object that will be called by the application when external data is required. This can be seen when the application requests location data and information for an attraction.

8.7 Abstract Overview

MainActivity.java

Method name	Type	Parameters	Description
onCreate	Protected void	Bundle savedInstanceState	Callback method. Construct the UI, facebook and login button when activity begins.
onActivityResult	Protected void	int requestCode, int resultCode, Intent data	To get back the result from either facebook / google login
onStart	Protected void	-	Checks on startup if the user has previously signed in with google
onClick	Public void	View v	When the google login button is clicked call signIn
signIn	Private void	-	Begin google login process
handleSignInResult	Private void	Task<GoogleSignInAccount> completedTask	Takes result of google sign in, deals with whether it was successful or not
updateUI	Public void	GoogleSignInAccount account	Moves to home activity if google login was successful

SignInActivity.java

Method name	Type	Parameters	Description
onCreate	Protected void	Bundle savedInstanceState	Callback method. Construct the UI, facebook and login button when activity begins.
signIn	Public void	View v	Read the user inputs and matched the inputs to the database

SignUpActivity.java

Method name	Type	Parameters	Description
onCreate	Protected void	Bundle savedInstanceState	Callback method. Construct the UI, facebook and login button when activity begins.
signUp	Protected void	View v	Read the user inputs and add them to the database

HomeActivity.java

Method name	Type	Parameters	Description
onCreate	Protected void	Bundle savedInstanceState	Construct the UI, start the map and set onclick functions for certain parts of the UI.
requestLocationPermission	Private void	-	Uses the request permission feature for location services.
requestCameraPermission	Private void	-	Uses the request permissions feature for camera services, specifically the rear camera view
startCamera	Private void	-	Uses requestCameraPermission along with checking permission to start up the camera view
onMapReady	Private void	GoogleMap mMap	Sets the map view to show the map provided by Google. Declares map related variables like location pins
getDirections	Private void	Attraction destination	Checks for the directions from the users location or chosen location to the attraction they have selected

updateLocationUi	Private void	-	Sets a Google maps dot for the users location if permissions have been allowed
onNavigationItemSelected	Private void	Menu item menuItem	Depending on the item selected, it will open a new display overlay for the respective page, along with data in those pages.

HelpFragment.java

Method name	Type	Parameters	Description
onCreate	Protected void	Bundle savedInstanceState	Callback method. Construct the UI, facebook and login button when activity begins.
init()	Private void	-	Initialise the custom list view adapter and transfer the data from a list to the list view

SettingFragment.java

Method name	Type	Parameters	Description
onCreate	Protected void	Bundle savedInstanceState	Callback method. Construct the UI, facebook and login button when activity begins.
getLocationPermission	Public void	-	To set the location permission to true in the android manifest
getCameraPermission	Public void	-	To set the camera permission to true in the android manifest

8.8 GUI, Human Interface Views – Describe how the user accesses functionality

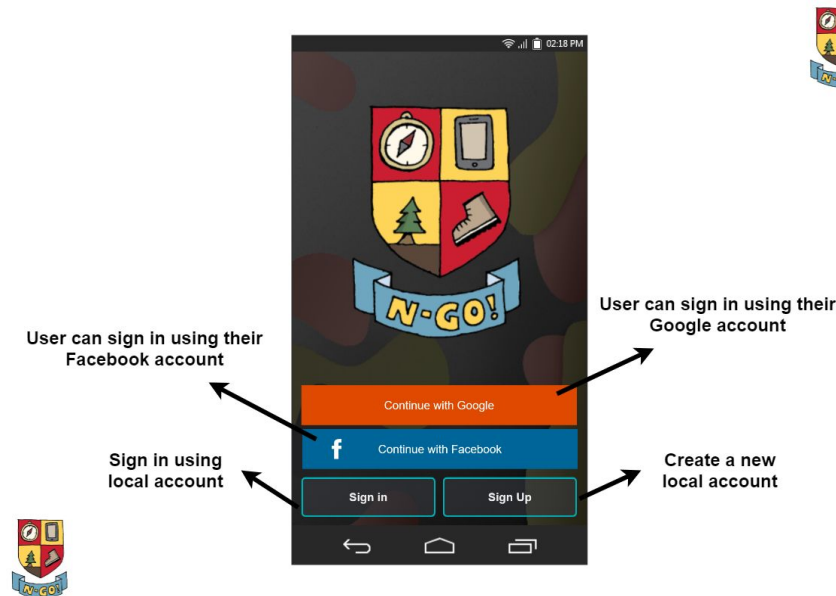


Figure 8.8.1 Login Page: This is the initial screen that appears when the user opens the application. The user will have the option of logging in using 3 different methods, that the buttons show. If the user decides to log in with either Facebook or Google, only basic information will be taken from these accounts that allow the app to run. An account can also be created within the app itself, that the user can use to log in. (Requirement 1.17)

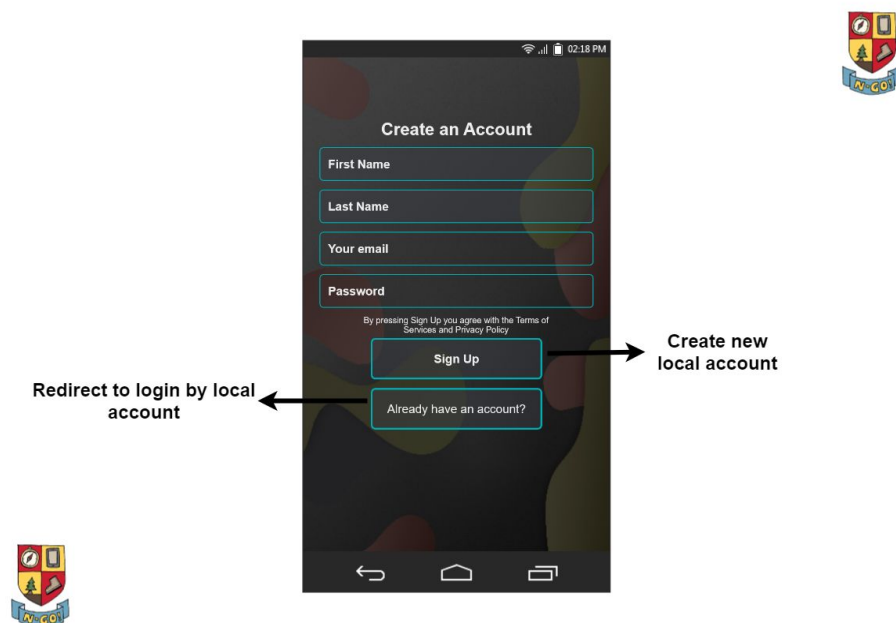


Figure 8.8.2 Sign up Page: This page allows the user to create a local account on the app. It uses only basic data that will be used within the app. It will ask the user to input their name, email and a password that will allow them to log back into the app anytime. Requirement (1.16)

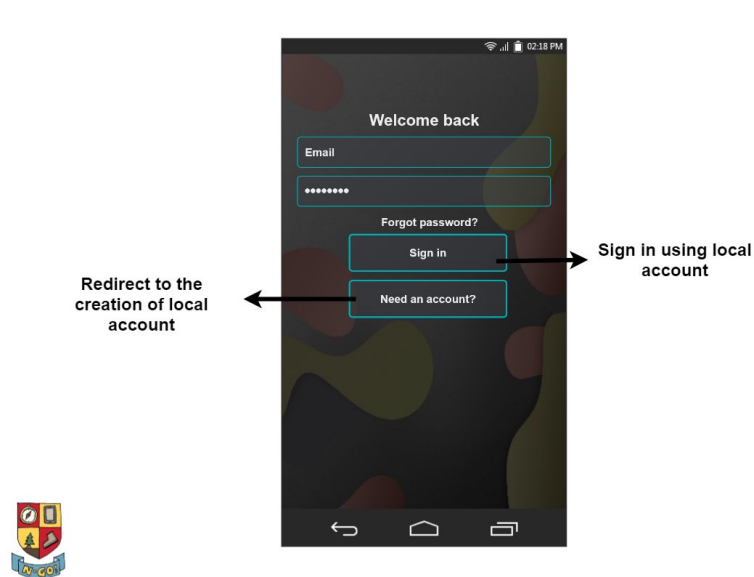


Figure 8.8.3 Sign in Page: This is the page for logging into the app when the user already has an account. The user can log in using their email and the password that was used when they initially signed up. It also has a button to the signup page for the user to create an account.

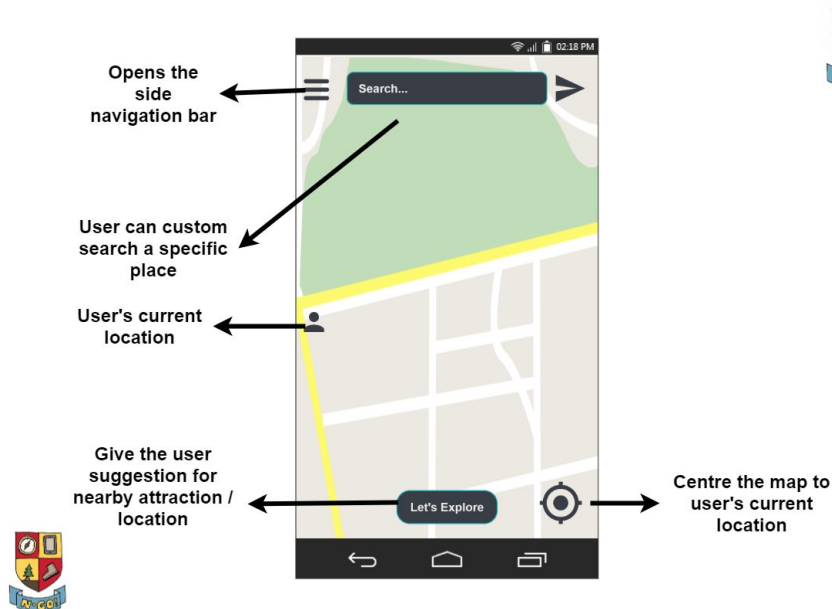


Figure 8.8.4 Main Page: This is the main page of the app and the first page the user sees when they login. A map of the surrounding area shows users the attractions that are nearby. There is a button in the top left that will open the side navigation bar which will allow the user to use more of the functionality of the app. At the top of the screen there is a search function that will allow the user to search for specific attractions within Northumberland, making it more user friendly/cuts out the middleman (Requirement 1.25). At the bottom right there is a button to centre the map on the user's location. The "Let's Explore" button will show the user a list of nearby attractions.

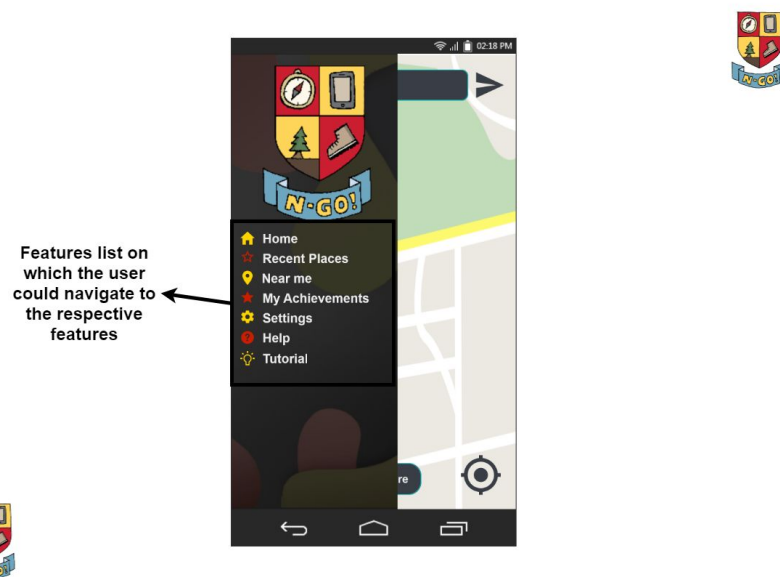


Figure 8.8.5 Side Navigation Bar: When the side navigation bar is opened on the main page, these are the options that appear for the user to choose. Recent places recently visited locations by the user in the app. Near me, will show the user attractions that are near the position of the user. My achievements will show a list of locked/unlocked achievements. Settings let the user change options about the app and Help will offer assistance with using the application.



Figure 8.8.6 Nearby Suggestion: If the let's explore button is clicked on figure 8.7.4 then the app will show the user a list of nearby attractions along with a picture of the location, satisfying the key requirement 1.12. These locations will change depending on where the user is located and what attractions are nearby to them at that time.



Figure 8.8.7 Location Brief Description: Once a nearby location is tapped by the user, the app will display a brief description about the attraction, and will allow the user to view more information about the attraction that they want to visit. The app will also display the approximate distance away from the attraction from the user's current location, and will give the user directions to the attraction with the click of a button. Requirement 1.14.

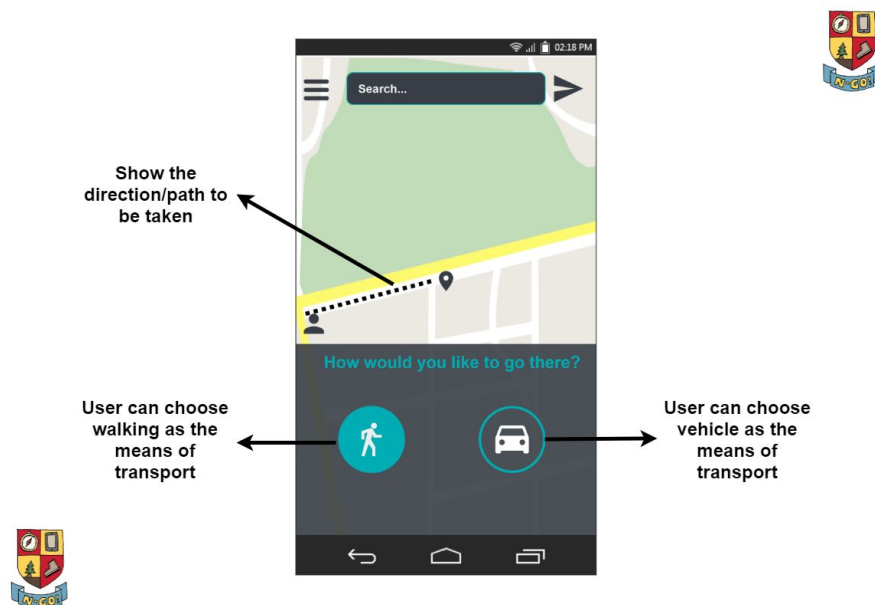


Figure 8.8.8 Means of Transport

After choosing the destination the app will show the shortest way to get there and will ask the user whether he/she is on foot or using a vehicle.

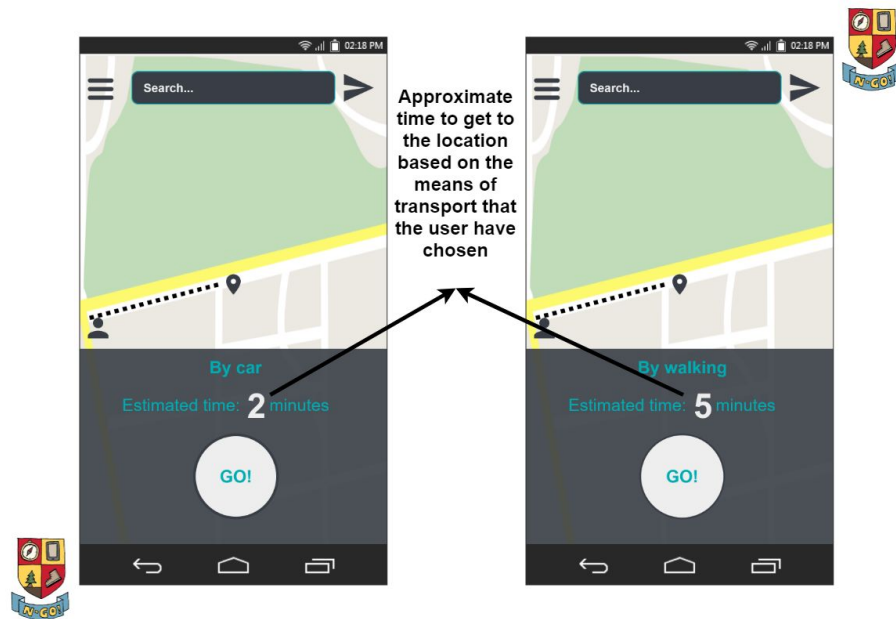


Figure 8.8.9 Approximate time

After choosing a means of transport, the app will display an approximate time in which the user will get to the chosen destination, estimated by the average speed of his/her option and have a button “GO!” to start active navigation.

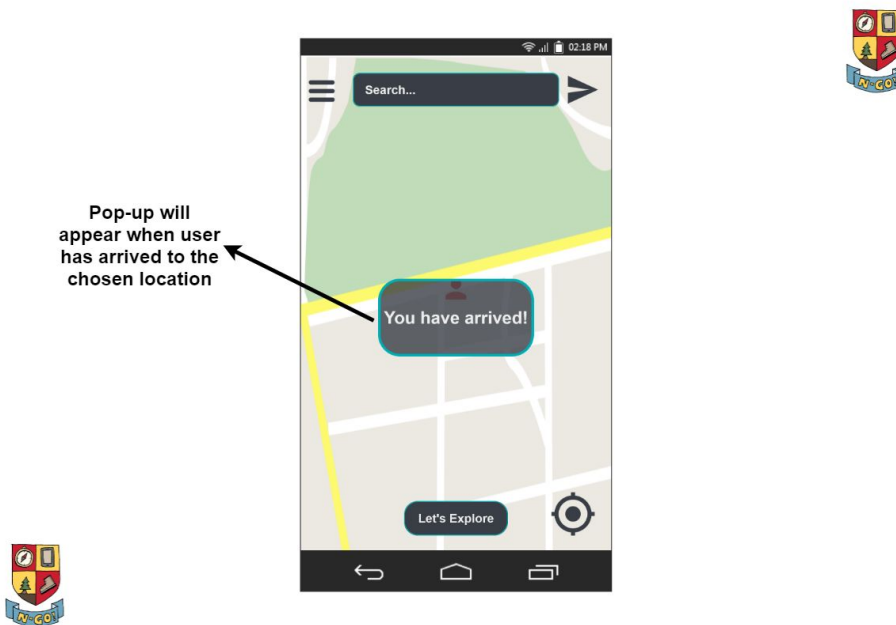


Figure 8.8.10 Arrived Pop-up

Once the user gets to his/her chosen location, a pop-up will appear telling them they have arrived.

User can click the button to activate the AR feature. The button will appear when the user are inside certain distance radius of the selected locations.

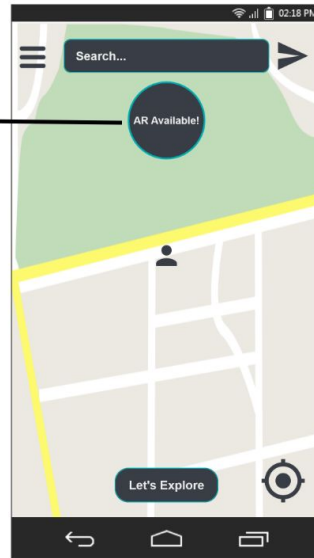
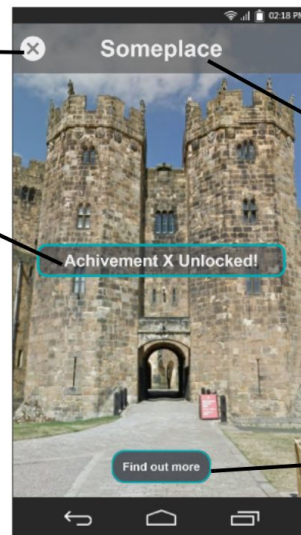


Figure 8.8.11 AR Prompt

Whenever the user gets near a destination an “AR Available!” button will appear. Pressing the button will turn on the device’s camera to add a layer of augmented reality. Requirements 1.23/4.

Close the AR feature

Pop-up will appear when user unlocked new achievement



Location's name

Redirect the user to the detailed information page



Figure 8.8.12 AR Visual

After pressing the “AR available” button, the device will start using its camera. At the top of the screen a small shadowed overlay has the name of the location with an “X” next to it to close the AR feature. A button at the bottom saying “Find out more” leads to a screen with more information about the location. Whenever the user unlocks an achievement a pop-up will appear notifying him of what he/she did.

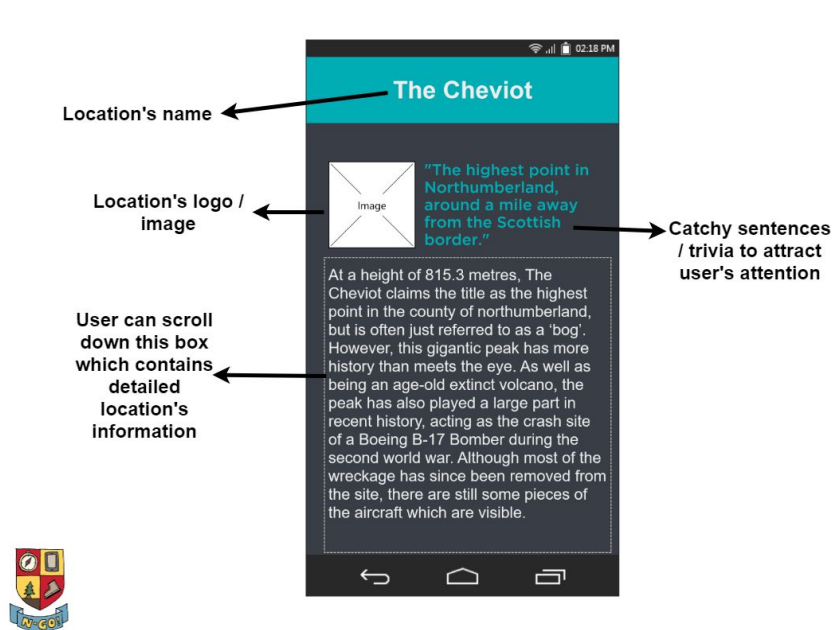


Figure 8.8.13 Detailed Information

After pressing the "Find out more" button a screen with a small image of the location and a catchy sentence/trivia about the location will appear. Under these there will be interesting information about the location, allowing the user to learn more about Northumberland. The website will allow for even further research to be conducted if this is what they desire.

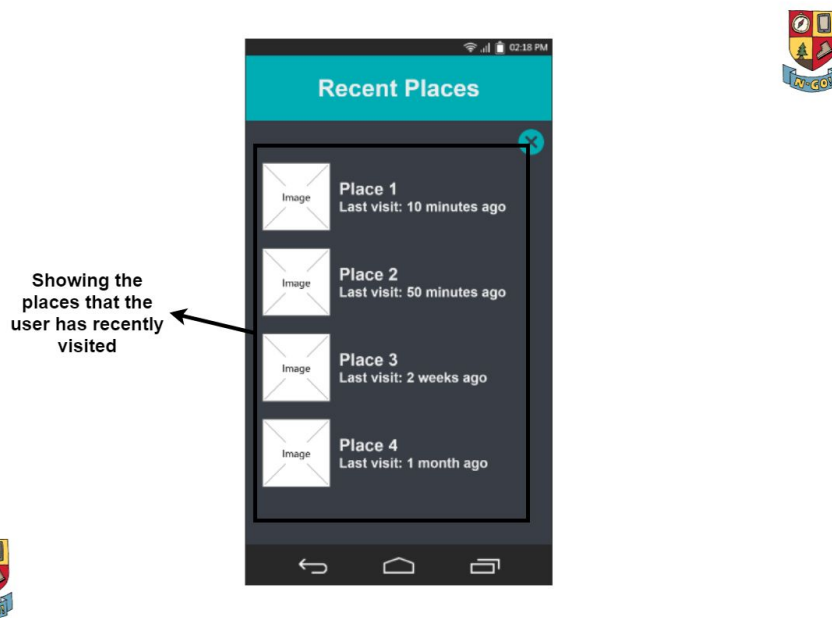


Figure 8.8.14 Recent Places

The Recent Places screen displays locations you have previously visited in chronological order, pressing one of the locations will redirect you to the information page about the selected place.

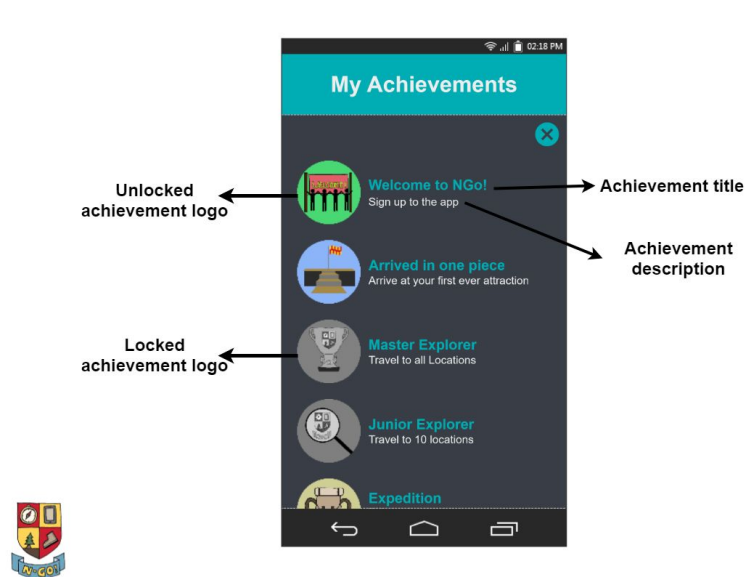


Figure 8.8.15 Achievement Page

The Achievement Page is where the user can see their achievements, the user will have all achievements locked when he first logs in and after completing one of them a pop up on the screen will appear notifying the user he has earned an achievement. There are hidden achievements that will not have instructions.

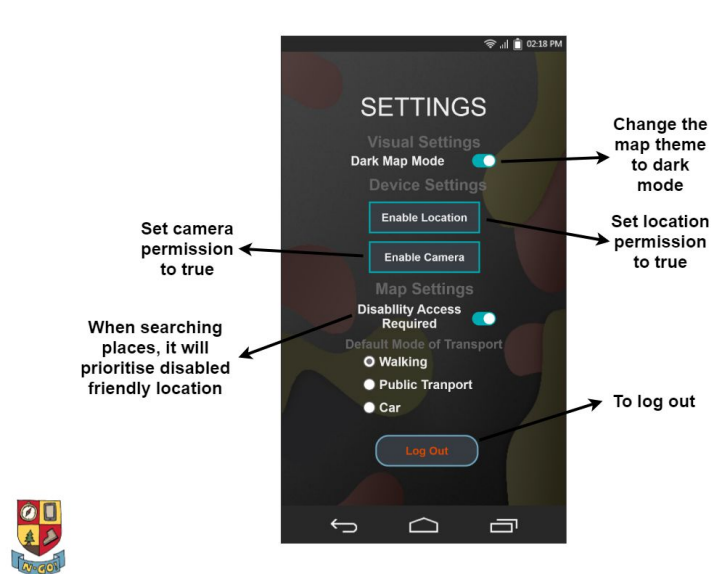


Figure 8.8.16 Settings

The settings page has options which let the user customize the colour scheme of the map and enable location and camera services. The user can also select a default method of transport and even select whether they need disability access (which will limit the attractions shown). The User can also log out of their account here.

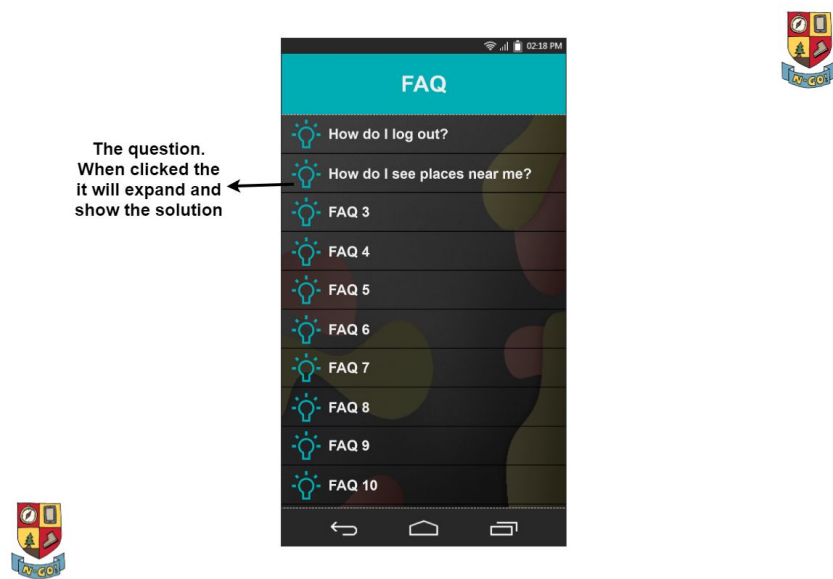


Figure 8.8.17 FAQ

The FAQ page contains a collection of all questions the user may have on how to use the application, pressing one of the questions opens an answer to the specific question. Each section of the application should be queryable/have questions associated with it, so that users are aware of how to use 'Northumberland GO!' to its full potential.

9. Test Plan

Testing Objectives

We will attempt to test our implementation within our team and with possible future users, in order to ensure we meet the requirements defined in the brief/by our client. The results from our tests should tell us whether our system works as expected, and if not, will allow us to improve and perfect the functionality of each feature.

Prototype testing will occur multiple times due to the nature of our Spiral development life cycle. However, the test plan below defines what features we plan to test for our final system to pass our set requirements. These features will most likely be tested as we develop the code through White-box testing, to avoid testing large amounts of the code at once. Black-box testing will be conducted by the documentation team to see how our implementation reacts to unplanned behaviour/a normal user.

For the Black-box testing, the documentation team will report any errors/bugs found to the development team, categorizing them as either **Low**, **Medium**, or **High** priority.

Test Number	Area of the Application	Description of Test	Expected Outcome
1.1	Login/Sign Up Page	Continue with Facebook	User can use their Facebook account to sign in to the application
1.2	Login/Sign Up Page	Continue with Google	User can use their Google Account to sign in to the application
1.3	Login/Sign Up Page	Valid Sign Up Process	User can create their own account to sign in with, using just a Name, Email and Password
1.4	Login/Sign Up Page	Valid Sign In Process	User can sign in to the application using their email and previously created password.
1.5	Login/Sign Up Page	Redirection from Sign In/Sign Up screens	Users can travel to the sign up page from the sign in page if they already have an account, and vice versa if they do not.
1.6	Login/Sign Up Page	Invalid Login Process with empty/invalid parameters	Will display an error to the user asking them to re-enter their information
2.1	Main/Map Screen	Let's Explore Button Click	Shows user some nearby locations which they can travel to
2.2	Main/Map Screen	Searching for a location	User can use the top search bar to search for a specific location. Matching location will appear if exists.

2.2.1	Main/Map Screen	Searching with Tags	The user should be able to search for locations containing a substring using '@' and search for a specific category using '#'.
2.3	Main/Map Screen	Centering the Map	The user can find their precise location by pressing the centre button.
2.4	Main/Map Screen	Showing the Navigation Sidebar	The user can reveal the Sidebar by pressing the top left button, revealing options to look at achievements, nearby locations, settings, help, and possibly recent trips.
2.5	Main/Map Screen	Available AR	If the user has reached a specified location, an AR pop-up will appear that will use the users camera if clicked to view nearby locations (to then show information)
3.1	Navigation/Sidebar	Recent Places Page Link	Selecting Recent Places will show the user the past few locations visited, and perhaps some information about each visit, e.g. time/date.
3.2	Navigation/Sidebar	Near Me Page Link	Similar to the Let's Explore button, will show nearby places the user can visit in a bit more detail
3.3	Navigation/Sidebar	Achievements Page Link	Should show the user all the achievements they have collected so far, and greyed out achievements they have yet to collect
3.4	Navigation/Sidebar	Settings Page Link	Any adjustable features of the app will be displayed here.
3.5	Navigation/Sidebar	Help Page Link	Should show the help page(s)/ a guide of how to use the app.
3.6	Navigation/Sidebar	Logging out of the application with button	Should sign the user out of the application and return them to the login/sign up screen
4.1	Location Selection	Directions Button	Should ask the user how they plan on reaching their desired destination, before giving them directions.
4.2	Location Selection	Find out more button	Will give the user a detailed description of the location, similar to the information they would receive if they were to reach it.
5.1	Transport Selection	Getting Directions on foot	Will show user expected time taken and produce a Go! button (must be for walking)
5.2	Transport Selection	Getting Directions by car	Will show expected time taken and produce a Go! button (must be for driving)
5.2.1	Transport Selection - Start Route	Getting Directions by pressing Go! button	Once transport is selected, pressing the Go! Button will revert to the map screen, showing directions to reach the destination (suitable for the mode of transport)
5.3	Directions	Directions to a location	The implemented Map API should give directions to the specified location similar to that of Google Maps/Uber, and tell the user when they have arrived.

6.1	Settings Page	Dark Map Mode	Button which switches the google maps theme to dark mode for night time usage.
6.2	Settings Page	Enable Location/Camera services	Buttons will enable these settings if not already allowed by the user, and will also tell the user if they are already enabled.
6.3	Settings Page	Disability Access	Button will allow only locations with disabled access to be found/searched for by the user.
6.4	Settings Page	Default Mode of Transport	Depending on which button the user selects, should automatically calculate a route for that mode of transport, without asking the user how they want to travel.
7.1	AR/Camera	Site within camera view	Should give the user a button option to find out more information about the site and display the Name of the attraction on screen
7.2	AR/Camera	Selecting to view more information	Will overlay detailed information about the attraction over the top of the camera screen which will be scrollable.
7.3	AR/Camera	Exiting the AR/Camera view	Pressing the 'X' at the top left will return the user to the main map screen.
7.4	AR/Camera	Reaching a new location to trigger achievement	Will unlock an achievement specific to the location visited, only if the user has reached and can see the location through their camera view.
8.1	Website	Login Tests	Same options as on the android application, same error checking will also take place
8.2	Website	Attraction Page link	Users can navigate to a page on the website which will show all attractions on the app, with brief descriptions.
8.3	Website	Selecting an attraction	Selecting an attraction will show the detailed information that can be found on the App, as well as links to other resources/sites where the user could conduct more research on the location.
8.4	Website	Achievements Page link	Scrollable achievements with descriptions on how to unlock them. Locked achievements will be greyed out. All scrollable
8.5	Website	User Guide and Contact Information page link	All documentation and contact info if user's require help can be viewed when navigating to this page
8.5.1	Website	Contact email address	Will open an email client to allow you to email directly.

10. Resources (List of links to all software/data used in the making of this project)

- Android Studio 3.2.1 & 3.3.1 (<https://developer.android.com/studio/>)
- Google Directions (<https://developers.google.com/maps/documentation/directions/start>)
- Google Firebase (<https://console.firebase.google.com/?pli=1>)
- Google Maps API(<https://developers.google.com/maps/documentation>)
- Gradle 5.0 (<https://gradle.org/releases/>)
- MapsForge 0.8.0 (<https://github.com/mapsforge/mapsforge>)
- Mockplus (<https://www.mockplus.com>)
- Notepad++ 7.3 (<https://notepad-plus-plus.org/>)
- Open Street Map (<https://www.openstreetmap.org/#map=6/54.910/-3.432>)
- Overpass API 7.55 (https://wiki.openstreetmap.org/wiki/Overpass_API/versions)

Contribution Matrix

Section	Ethan Wormley	Jamie Child	Sam Holley	Calvin Hudson	Robert Adamson	Aleksandar Gavazki	Calum Harvey	Stefan Lip	Michael Putra
Specification Overview	C,R	R	R	R	R	R	R	M,R	R
Background & Analysis	M,R	R	R	C,M,R	R	R	R	M,R	R
Project Purpose	M,R	R	R	R	R	R	R	C,M,R	R
Roles & Deliverables	M,R	R	R	R	R	R	R	C,M,R	R
Project Plan	C,M,R	R	R	R	R	R	R	R	R
Hardware & Software	M,R	R	C,M,R	M,R	R	R	R	M,R	R
Definition of Terms	M,R	R	R	C,M,R	C,M,R	R	R	R	R
Requirements	M,R	R	R	M,R	R	R	C,M,R	R	M,R
Other Considerations/Modifications	M,R	R	R	C	R	C,M,R	R	R	R
High Level/UML Diagrams	R	R	C,M,R	C,MR	C,M,R	R	R	C,M,R	R
GUI Diagrams & Analysis	M,R	R	R	R	R	C,M,R	C,M,R	R	C,M,R
Test Plan	C,M,R	R	R	R	R	R	R	R	R
Resources	M,R	R	C,M,R	M,R	R	R	R	M,R	R

Key : C = Create , M = Modify , R = Review

References

- 1.1 Tripadvisor.com. (2017). *TripAdvisor Network Effect and the Benefits of Total Engagement*. [online] Available at: <https://www.tripadvisor.com/TripAdvisorInsights/w828> [Accessed 5 Dec. 2018].
- 1.2 TripAdvisor. (2018). *Screenshot by author from Android app* [Accessed 27 Nov. 2018].
- 1.3 TripAdvisor. (2018). *Screenshot by author from Android app* [Accessed 27 Nov. 2018].
- 1.4 May, K. (2011). *TripAdvisor loves Google again, puts Street View into iPad app | PhocusWire*. [online] Phocuswire.com. Available at: <https://www.phocuswire.com/TripAdvisor-loves-Google-again-puts-Street-View-into-iPad-app> [Accessed 5 Dec. 2018].
- 1.5 Northumberland Tourism. (2018). *Access for all | Visit Northumberland*. [Screenshot by author] Visitnorthumberland.com. Available at: <https://www.visitnorthumberland.com/access-for-all> [Accessed 4 Dec. 2018].
- 1.6 Phillips, T. (2018). *Pokémon Go active player count highest since 2016 summer launch*. [online] Eurogamer.net. Available at: <https://www.eurogamer.net/articles/2018-06-27-pokemon-go-player-count-at-highest-since-2016-summer-launch> [Accessed 5 Dec. 2018].
- 1.7 The Verge (2016). *Augmented Reality in Pokemon Go*. [image] Available at: [https://cdn.vox-cdn.com/thumbor/yOoC7hewGGx2iQf8C6SwXwKNvHU=/0x0:1500x1000/920x613/filters:format\(webp\)/cdn.vox-cdn.com/uploads/chorus_image/image/50518821/akrales_160714_1141_A_0008.0.0.jpg](https://cdn.vox-cdn.com/thumbor/yOoC7hewGGx2iQf8C6SwXwKNvHU=/0x0:1500x1000/920x613/filters:format(webp)/cdn.vox-cdn.com/uploads/chorus_image/image/50518821/akrales_160714_1141_A_0008.0.0.jpg) [Accessed 5 Dec. 2018].
- 1.8 Belluz, J. (2016). *Pokémon Go wants to be the exercise app that actually works*. [online] Vox. Available at: <https://www.vox.com/2016/7/12/12159198/pokemon-go-exercise-increase> [Accessed 5 Dec. 2018].
- 1.9 McCormick, R. (2016). *Driver distracted by Pokémon Go kills woman in Japan*. [online] The Verge. Available at: <https://www.theverge.com/2016/8/25/12637878/pokemon-go-driver-kills-woman-japan> [Accessed 5 Dec. 2018].
- 5.1 Wikipedia The Free Encyclopedia. (2005). *API*. [online] Available at: https://en.wikipedia.org/wiki/Application_programming_interface. [Accessed 1 March 2019].
- 5.2 Griffiths, D. and Griffiths, D. (2015). *Head First Android Development*. 1st ed. O'Reilly Media, Inc, USA.
- 5.3 Emspak, J. (2018). *What is Augmented Reality?*. [online] Live Science. Available at: <https://www.livescience.com/34843-augmented-reality.html> [Accessed 6 Mar. 2019].

5.4 Wikipedia The Free Encyclopedia. (2005). *Git*. [online] Available at: <https://en.wikipedia.org/wiki/Git>. [Accessed 2 December 2018].

5.5 Sierra, K. and Bates, B. (2005). *Head First Java*. 2nd ed. O'Reilly Media, Inc.

5.6 Christensson, P. (2011). *Navigation Bar Definition*. [online] TechTerms. Available at: https://techterms.com/definition/navigation_bar [Accessed 6 Mar. 2019].

5.7 OpenStreetMap. (n.d.). *OpenStreetMap*. [online] Available at: <https://www.openstreetmap.org/about> [Accessed 6 Mar. 2019].

5.8 Connolly, T. and Begg, C. (2015). *Database Systems*. 6th ed. Pearson Education UK.

5.9 Christensson, P. (2009). *User Interface Definition*. [online] TechTerms. Available at: https://techterms.com/definition/user_interface [Accessed 6 Mar. 2019].

5.10 Wikipedia The Free Encyclopedia. (2003). *XML*. [online] Available at: <https://en.wikipedia.org/wiki/XML>. [Accessed 2 December 2018].