James Duggan

06357628

HandBook

A Walking Tour App

Table of Contents

[1 Introduction 2](#_Toc95651041)

[1.1 Background to the Problem 2](#_Toc95651042)

[1.2 Description of Project Aims 2](#_Toc95651043)

[2 Technologies 2](#_Toc95651044)

[2.1 Software 2](#_Toc95651045)

[2.1.1 React 2](#_Toc95651046)

[2.1.2 React Native 3](#_Toc95651047)

[2.1.3 Expo 3](#_Toc95651048)

[2.1.4 Leaflet 3](#_Toc95651049)

[2.1.5 React Native Maps 3](#_Toc95651050)

[2.1.6 Firebase 3](#_Toc95651051)

[3 System and Analysis Design 4](#_Toc95651052)

[3.1 User Stories 4](#_Toc95651053)

[3.2 Class Diagrams 5](#_Toc95651054)

[3.3 Functional Requirements 5](#_Toc95651055)

[3.3.1 Core Requirements 5](#_Toc95651056)

[3.3.2 Non-functional Requirements 6](#_Toc95651057)

[3.3.3 Stretch Goals 6](#_Toc95651058)

[3.4 User Interface Design 6](#_Toc95651059)

[3.4.1 Web App 6](#_Toc95651060)

[4 Methodology 7](#_Toc95651061)

[4.1 Methodology Selection 7](#_Toc95651062)

[4.2 Process Outline 7](#_Toc95651063)

[4.2.1 Phase 1: Define and finalize project requirements 7](#_Toc95651064)

[4.2.2 Phase 2: Begin building prototypes 7](#_Toc95651065)

[4.2.3 Phase 3: Gather user feedback 8](#_Toc95651066)

[4.2.4 Phase 4: Testing 8](#_Toc95651067)

[4.2.5 Phase 5: Present your system 8](#_Toc95651068)

[4.3 Project Schedule 8](#_Toc95651069)

[5 Summary 8](#_Toc95651070)

[6 Bibliography 9](#_Toc95651071)

# Introduction

## Background to the Problem

In the current pandemic, face to face interaction and moving in groups is being avoided as people want to feel safe. Because of this, and the simple fact that people aren’t travelling, activities like tours have taken a hit over the past two years.

## Description of Project Aims

The tourism industry is important, not just from a revenue perspective but also in terms of the learning that it can provide. For my project I want to create a walking tour app that can be used by anyone, anywhere in the world to create an audio tour that, in turn, anyone can take on their mobile phone. I think an app like mine would be of great use to people who are wary about travelling in a group, as most walking tours would be, but also to the usual hosts of walking tours themselves. Someone could create a tour and have people taking it while they are still running in-person tours. They could even use the app as a kind of sample tour and mention in the audio that if someone would be interested in taking a more in-depth tour to contact them.

My project will include the following:

* A web app where users can create tours
* Users will need to be able to create a tour with different points of interest or destinations and associate an audio recording with them
* A mobile app where users can take tours
* It should be easy for users taking a tour to find one in their location, possible through use of a search bar

# Technologies

## Software

### React

React is a JavaScript library for building user interfaces. I will be using this to create my web app which a Tour Guide user will interact with to create apps. There are several reasons why I chose React to build my app in the end. Firstly, I am more confident with JavaScript than any other programming language. Also React is one of the most popular JS libraries in use today, being used by large sites such as Facebook, Instagram, and Reddit for their front-end development needs. (Patel, 2022)

React is also component driven, which will assist greatly in the type of app I will be developing. (Pandit, 2021) The main data types I will be using for the app will be Places of Interest (or Destinations) and Tours. These can be displayed in a similar fashion, i.e. with a map and then some data in a text format. As such, reusable components that can work off of either data set would be ideal.

### React Native

React Native is an open source UI software framework developed by Meta Inc that can be used to develop apps for both Android and iOS. It allows the developer to create an app in JavaScript that can run on both Android and iOS.

Being able to write my code for the mobile app in JavaScript will greatly reduce my workload as a lot of my code for the web app should carry over.

### Expo

Expo is an open-source platform for making universal native apps for Android, iOS, and the web with JavaScript and React. It is essentially a bundle with tools and services that allows a developer to create a mobile app in React Native with relative ease and ensures that you do not need to work with native Android or iOS code to get you project up and running. (Ravichandran, 2020)

For someone just starting with React Native, Expo is a natural choice. As well as this, Expo has a mobile app that can allows the developer to view their app on their mobile device during development by scanning a QR code.

### Leaflet

Leaflet is an open source JavaScript library for creating interactive maps in web apps. I have decided to use this for my web app to keep prospective costs low.

### React Native Maps

React Native Maps is a library for React Native that assists in creating interactive maps for mobile apps built in react native. This library is used in conjunction with Google maps so an API key will be required.

### Firebase

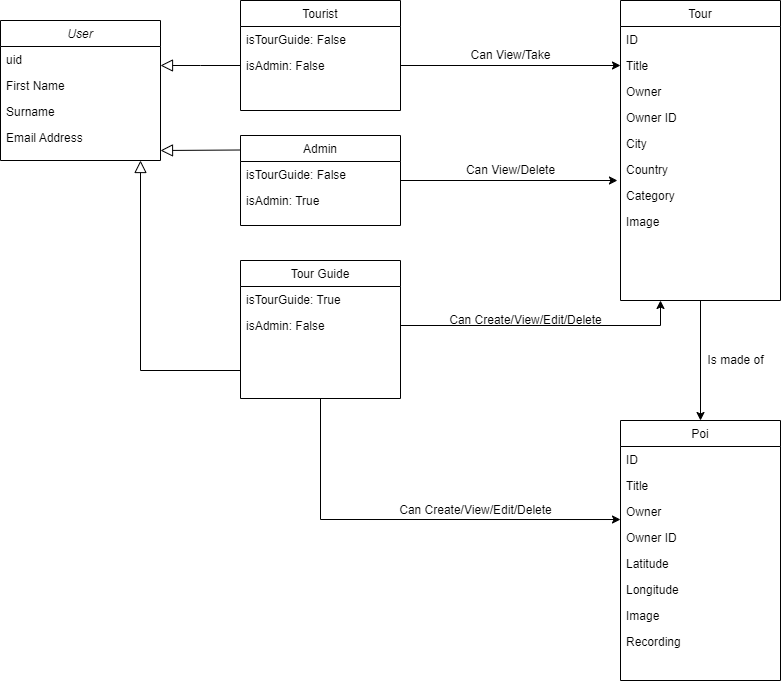
Firebase is a backend service that helps developers build real-time apps for iOS, Android and the web that can store and sync data instantly. (Lardinois, 2014) I will be using it to handle the majority of my backend functionality. This will include storage for images and audio files, a real-time database for storing data for tours, destinations, and user information, and user creation and authentication.

# System and Analysis Design

## User Stories

|  |  |  |  |
| --- | --- | --- | --- |
| # | As a | I want to be able to | So that |
| US1 | Tour Guide | Create an account | I can create tours |
| US1.1 | Tour Guide | Create tours | Customers can view and take them |
| US1.2 | Tour Guide | Create destinations | I can group them together to make a tour |
| US1.3 | Tour Guide | Edit tours | Maintain my tours |
| US1.4 | Tour Guide | Delete tours | Maintain my tours |
| US1.5 | Tour Guide | Edit destinations | Maintain my destinations |
| US1.6 | Tour Guide | Delete destinations | Maintain my destinations |
| US2 | Tourist | Create an account | I can take tours |
| US2.1 | Tourist | View all tours | I can make a decision on what tour I like |
| US2.2 | Tourist | Assign a tour to my account | I can take the tour now or later |
| US2.3 | Tourist | Remove tour from my account | I can maintain my tour list |
| US2.4 | Tourist | Search tours | I can find nearby tours more easily |
| US3 | Administrator | Remove users from the system | To remove any troublesome/abuse users |
| US3.1 | Administrator | Remove tours | To remove any offensive content |
| US3.2 | Administrator | Remove destinations | To remove any offensive content |

## Class Diagrams



## Functional Requirements

### Core Requirements

Users will need to be able to, firstly, create an account and log in. Once logged in they will need to be able to create destinations that will make up their tours. Both tours and destinations will need to be persistent so the data will be stored in a Firebase Realtime Database.

Storage will also be required as each destination will have an associated image and recording, and each tour will have an associated image. Users will need to be able to upload the files to online storage, again handled by Firebase, then the URLs for these files will need to be saved to the Tour or Destination model.

### Non-functional Requirements

In terms of security, this will generally be handled by Firebase Auth, which uses its own password hashing. If I were to ever look at giving Tour Guides the ability to charge for their tours, then further security measures will need to be looked into. I will also need to implement validation for my forms.

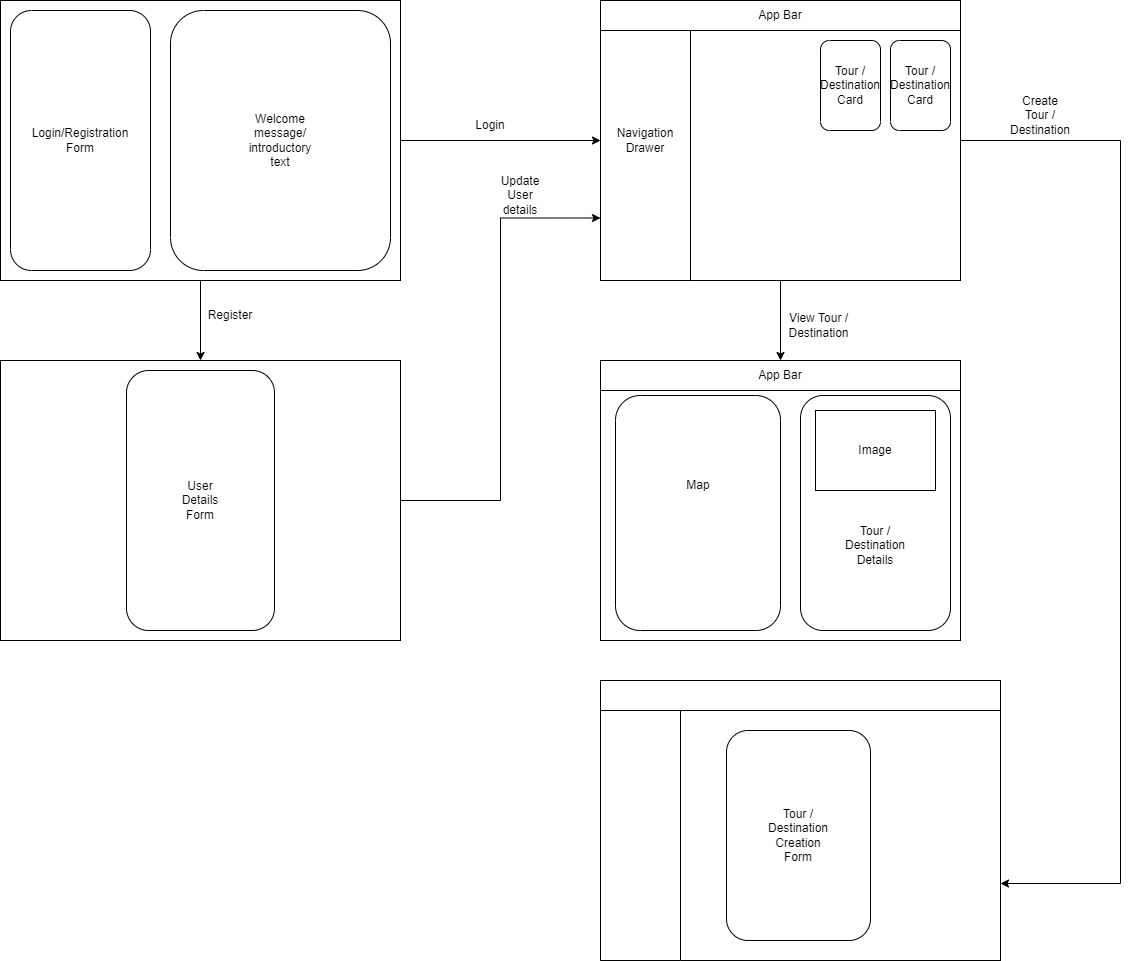
### Stretch Goals

The first stretch goal I would like to aim for once the base apps are complete, would be to give users the ability to create an orienteering style tour. With this type of tour, markers will not appear on the map and the tourist users will use recorded clues to find the next destination.

Following this, I would also like to include some more social functionality such as being able to like or dislike a tour, or a star based review system. The ability to comment on tours would also be a useful feature.

## User Interface Design

### Web App



# Methodology

## Methodology Selection

After looking at a few different application development methodologies, I have decided to go with a version of the rapid application development model (RAD). The RAD approach is a form of Agile software development emphasizing rapid prototype releases and iterations. (Cox, 2019)

There are many benefits to the RAD method that makes it the ideal working ethos to apply to my own project. First is flexibility as developers can make adjustments to their code quickly during the development process. As I am working with a clear deadline, the speed of delivery when working using this method is also an advantage.

The greatest advantage of the method in relation to my own project, however, is the encouragement of code reuse. (Cox, 2019) This is a natural fit working with component based library such as React as I will be creating components and then re-using them for different pages/purposes within the web app. Considering I am then moving on to create the mobile app in React Native, where a lot of my code from the web app will carry over, something along the lines of the RAD method is the obvious choice.

The RAD method consists of 5 distinct phases, which I will be tweaking to fit my current situation, i.e. working by myself on a college project as opposed to as part of a Team.

## Process Outline

### Phase 1: Define and finalize project requirements

During this phase, stakeholders in the project will meet to define and finalize project requirements such as project goals, expectations, timelines, and budget. (Cox, 2019) At a macro-level, the stakeholders are myself and my supervisor and we will be having regular meetings to discuss the project but in terms of what the project is and what it needs to do, this has been decided early on. I will also be applying this phase, and all phases, on a micro level. This would mean on a component by component basis. I will be planning what the component needs to do, what it needs to look like, etc. At this level, I am the only stakeholder.

### Phase 2: Begin building prototypes

In this phase, designers and developers will work closely with clients to create and improve upon working prototypes until the final product is ready. (Cox, 2019)

For this phase, and subsequent phases. I will be approaching it on a component by component basis. As I am, in a way, the client, the developer, and the designer, this phase will largely involve drawing up design for a component, keeping in mind what its purpose is, and then reproducing this idea in code.

### Phase 3: Gather user feedback

Once the components are put together as a page, one would generally gather user feedback. In this case the feedback would be coming from myself so if I am not happy with how the project looks and functions, it will be back to the drawing board.

### Phase 4: Testing

This process will involve testing the product, both as individual components and together as a whole project.

### Phase 5: Present your system

Under the RAD method this phase will involve data conversion, user training, and (possibly) more testing. (Cox, 2019) For me, this will mainly involve setting up usable user accounts and tours that can be taken with these accounts. It will also, hopefully, include deploying both apps so that it can be used by external users.

## Project Schedule

|  |  |  |
| --- | --- | --- |
| Week | Date | Development Work |
| Week 1 | 17 January | Work on web app |
| Week 2 | 24 January | Work on web app |
| Week 3 | 31 January | Have basic web app completed |
| Week 4 | 7 February | Work on interim report/Start mobile app |
| Midterm | 14 February | Continue mobile app |
| Week 5 | 21 February | Continue mobile app |
| Week 6 | 28 February | Have basic mobile app completed |
| Week 7 | 7 March | Work on UI/Stretch goals |
| Week 8 | 14 March | Stretch goals |
| Week 9 | 21 March | Work on Final Report |
| Week 10 | 28 March | Project Submission |
| Week 12 | 04 April | Work on Video and Presentation |

# Implementation

## Web App

### Phase 1: App Creation and Login

#### Planning

During the planning phase, I decided that the best way to get started with the app was to create the app and then go straight into creating the Login Page and implementing Firebase Authentication. My reasoning for this was to avoid backtracking. The Login Page is the first page users will come into contact with so it should be the first thing I work on. I also wanted to get Firebase Auth up and running as this would mean, once I started working on the User Model, a lot of the base information will already be there and be functional, e.g. the uid, email, etc.

#### Review

I began by running Create React App which does a lot of the heavy lifting with regard to creating a React web app, creating a modern build setup with no configuration. Once this was done and the app was created, I created a test page that I could use to test the different components I would be creating as I went along and then began work on setting up Firebase Auth and putting together a login form.

I had previously set up Firebase Auth on a Kotlin app but this was my first time doing so with React so I consulted a YouTube video for a step by step guide on how to implement (Simplified, 2020). Once I had set up a project on the Firebase site, I then went on to create an .env file which would hold the API key and various other vital values for firebase to be operational on my app. This was followed by creating a firebase.js file which would handle communication with and imports from Firebase. This can be seen below.

Text

Description automatically generated

I then created a separate contexts folder in my application and a JavaScript file to house my authentication methods, which I then implemented into my login form.

After testing the form on my test page to ensure both registration and login functions worked, I created a login page which I then put the finished form into. The process was fairly straight forward and I was then able to create a new users for my web app.

Graphical user interface

Description automatically generated

#### Evaluation

All in all, implementing authentication through Firebase is straightforward with a good deal of supporting material online to assist. One issue arising from this is how to approach setting up the user model. The model will require the uid but this is not created until the user is created on Firebase Auth. As such, I will need to implement a two-step registration process down the line.

#### Reflection

These initial steps have made me more confident in my choices of technology. React and Firebase work together fairly well and the setup was not very intensive.

### Phase 2: Navigation

#### Planning

With regard to planning my navigation, I decided to use React Router to handle this as this is the most common practice. My plan was to have a Login Page act as the landing page, which would then lead to several protected pages depending on if the user logged in or registered. A diagram outlining the navigation can be seen above.

#### Implementation

We had previously covered navigation with React in the *ICT Skills 2* module so I was confident that putting together navigation for the app would be relatively straightforward, and that I could base my Private Routes off of previous assignments. However, in v6 of React Router the implementation of Private Routes has changed completely. Luckily, I was able to find a comprehensive guide on a new method to implement private routing (Luca, 2021).

I began by creating a Private Route component. The code for this can be seen below, with the code of the previous method of creating a Private Route commented out:

Text

Description automatically generated

As you can see the code has been simplified greatly, going from multiple lines of code to just two.

#### Evaluation

Having read up on the new process for creating Private Routes, it was easier to implement than the initial method that I had learned.

#### Reflection

This process has really highlighted just how quickly things can change in the world of Web Development. It was interesting to see that within less than a year a method that we had learned was completely obsolete and would not work at all.

### Phase 3: Maps

### Phase 4: Firebase Realtime Database

### Phase 5: Firebase Storage

### Phase 6: Media Player

### Phase 7: Further CRUD Implementation

## Mobile App

### Phase 1: App Creation on Expo and Firebase Auth

### Phase 2: Accessing Realtime Database

### Phase 3: Maps

### Phase 4: Media Player

### Phase 5: Geolocation

### Phase 6: User Model

# Bibliography

Cox, T., 2019. *Capterra.* [Online]   
Available at: https://blog.capterra.com/what-is-rapid-application-development/#definition  
[Accessed 12 February 2022].

Lardinois, F., 2014. *TechCrunch.* [Online]   
Available at: https://techcrunch.com/2014/10/21/google-acquires-firebase-to-help-developers-build-better-realtime-apps/?guccounter=1&guce\_referrer=aHR0cHM6Ly9lbi53aWtpcGVkaWEub3JnLw&guce\_referrer\_sig=AQAAAGeodZJSMgS3exGPN2pxnED2gAprUT3i32Y\_b0uBPCZg4Grl01QN-Prc06WhZNanH  
[Accessed 13 February 2022].

Pandit, N., 2021. *C#Corner.* [Online]   
Available at: https://www.c-sharpcorner.com/article/what-and-why-reactjs/  
[Accessed 06 February 2022].

Patel, J., 2022. *Monocubed.* [Online]   
Available at: https://www.monocubed.com/why-use-react/  
[Accessed 06 February 2022].

Ravichandran, A., 2020. *Adhithi Ravichandran.* [Online]   
Available at: https://adhithiravi.medium.com/building-react-native-apps-expo-or-not-d49770d1f5b8#:~:text=With%20Expo%2C%20you%20can%20build,iOS%20and%20Android%20with%20ease.&text=With%20Expo%2C%20you%20will%20never,while%20building%20apps%20using%20Expo.  
[Accessed 6 February 2022].