# Project Document Information

|  |  |
| --- | --- |
| **Project name:** | Trekking App |
| **Date:** | 03/02/21 |
| **Authors:** | Software Project Team 9  118410722 Adam Evans  116451016 Eoin O’Connell  118332823 Oliwia Kobos  118359221 Pádraig Ó Cróinín  118476662 Adrian Lamug |
| **Product Owner:** | Jason Quinlan |
| **Version:** | v0.3 |
| **Colour Code:** | Newly added updates  Updates added from the peer review |

# Definition

|  |  |
| --- | --- |
| **Main Goal:** | To provide an easy to use mobile application for finding walking paths. |
| **Desired Outcomes:** | Easy to find trails near your location with the ability to customise the paths based on user needs. |
| **Constraints and Assumptions:** | App will be available to mobile devices that have access to a working internet connection  User will confirm access to GPS location  Google Maps limitations to available routes  Current Google Maps API plan provides plenty of credits to use however when scaling the application, the costs scale as well. |
| **Interfaces:** | Google Maps, trails API |
| **~~Project Approach:~~** | ~~Our team has decided to push comfort zones and decided to create a mobile application using a newly learned language, Kotlin.~~  trello, kanban, user stories etc measuring by moscow requirements done |

# Key Stakeholders

|  |  |
| --- | --- |
| **Major Stakeholder** | **Notes** |
| **Jason Quinlan** | PRODUCT OWNER |
| **Adam Evans** | APPLICATION DEVELOPER |
| **Eoin O’Connell** | APPLICATION DEVELOPER |
| **Oliwia Kobos** | UI/UX DESIGN |
| **Pádraig Ó Cróinín** | SOFTWARE TESTER, (Creating a second activity, a maps activity and trying to enabling switching between MainActivity and MapsActivity) |
| **Adrian Lamug** | SYSTEMS ARCHITECT |

# Project Objectives

|  |  |  |
| --- | --- | --- |
|  | **Target** | **Tolerance** |
| **Scope** | Ideally hitting our MOSCOW targets | Hitting just the must targets |
| **Time** | 7 weeks | 8 weeks |
| **Cost** | Hopefully no costs will arise |  |
| **Quality** | A polished, working product |  |
| **Risks** | Using up Google Maps credits |  |
| **Benefits** |  |  |

# 

# Outline Business Case

Since the introduction of lockdown measures due to Covid-19 many people are finding themselves spending more time than normal at home. Therefore, there is a bigger interest in making up for lost movement through exercising at home. Our application identifies the possible market of mobile users who are stuck in the same monotonous routine that makes walking seem more burdensome than necessary.

The Trekking App intends to use the readily available Google Maps API to create and track customisable routes, allowing users to have a new experience every time they venture outside. Users can avail of features such as filters that allow them to set the maximum or minimum distance of a route, adhering to the exercise guidelines established. The application can include scenery or places of interest during the walk, tap into local weather news to predict path conditions, track user stats, and more.

Unlike other applications on the market, the Trekking App’s main benefit is in combining both rural and urban landscapes, minimising the need for major travel and creating something available to all without limiting features based on someone’s locations.

# Product Description

Our project is a reaction to the health crisis being caused by the covid-19 pandemic. In order to try and promote exercise and help people get out of their homes to destress we plan on creating a mobile app that will pick out walking paths for people that abide by the Covid-19 guidelines at the time.

This application will serve as a recommendation guide for the user to make their walk more interesting, not as a navigation system to get from place A to place B. Therefore it is up to the user’s discretion on how closely they choose to follow the path.

**Must-Have Features**

Certain features are vital to consider the project a success. These are so-called “must-haves”. These features underpin the main purpose and functionality of the product.

The app must provide users with trails that comply with the current Covid-19 guidelines effective in their area. For example only suggesting trails that will not take a user over 5km away from their home. As Covid-19 regulations change these changes will be reflected in the trails suggested to the user.  
Implementation: The user will receive trails from the Trails API. Their location will then be used to find and present to the user the suitable trails for them.

The user must be notified of changes to Covid-19 guidelines and weather warnings relevant to them. Weather warnings and changes to Covid-19 guidelines are liable to change depending on your locality.

Implementation: A Queue service could be used from Google Cloud to send user notifications.

GPS tracking will be used to keep track of a user’s completion of routes. A user will also be able to see how far along they are on a particular route.

A favourites tab will enable a user to save and rate trails. They will also be able to sort their rated trails based on their given rating.

Implementation: Store locally the trails that a user has completed and the ratings they have given them.

**Should-Have Features**

The “Should Haves” aren’t completely necessary for the successful delivery of the project but they are definitely high-priority features that provide high value to the application. The app should have a feature for the user to filter route types such as “Scenic Routes”. This function will also work with the ratings feature, giving the user the option to filter by rating and popularity. Another feature that relates to this function is filtering routes by the length of the trail. For example, filtering under 5km, will only show trails nearby that have an approximate length of under 5km.

Stat tracking is an important feature that is common to fitness/health apps. For this application we will include user stats such as steps taken, distance travelled, average pace, etc. At the end of each week, there will be a user report, based on their performance. They will have the option to share this on social media.

**Could-Have Features**

Due to the limited amount of time available in the development of this app, some features have to be prioritised over others. The lesser prioritised items can be considered as “could have” elements, meaning elements that developers feel that they could implement in favourable circumstances or that are for vanity purposes mostly. Our application has some of these elements that we would like to implement in an ideal setting.

A path creator feature is our most ambitious feature which would allow the user to create the trails manually through the means of click and drag technology. While the application will use customisable filtering to search ready made paths, ideally a user could create a route if they already have a certain trail they like to follow.

Implementation: Allow a user to declare that they are starting a trail. Periodic updates of their location will be taken using Google Maps. This will lead to a sequence of co ordinates being generated, which will represent their created trail.

“Marathon Mode” could allow a user to build up longer walking or running distances over time, while at the same time creating healthier habits. Similar to the “Couch to 5k” challenge, a user can push their limits and watch their achievements and goals steadily rise through the use of this app. Unlike the challenge, our application provides unique routes that can be undertaken instead of just distance goals.

To follow the Covid-19 safety regulations, our application could keep track of trail “traffic”; the amount of users that are currently using that particular route. The traffic system would be directly correlating to a simple to read heat map that shows the amount of users just by colour.

**Won’t-Have Features**

Including a “won’t have features” list seems counterintuitive at first, however, the advantages of such a list helps our team set and keep track of realistic goals and what features we can expect to be out of scope for the application due to time or other constraints. We can also set much more specific requirements and specifications that our application requires of the user.

One such feature, or lack thereof, is offline support. Reading the peer reviews allowed us to see how others view our project and a question that came up more than once was relating to how our application will handle a dropped internet connection. Ideally we would like to have a working offline application but due to the short nature of this school project it seems unlikely. Therefore, we shall have to make assumptions that any user with access to our mobile application will also have access to Wi-Fi and mobile data.

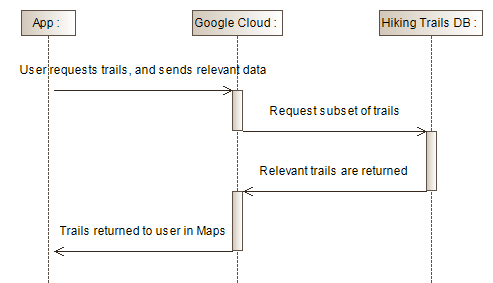
This list will also include the feature of caching the trails before the user goes on walks.

**Project Architecture:**

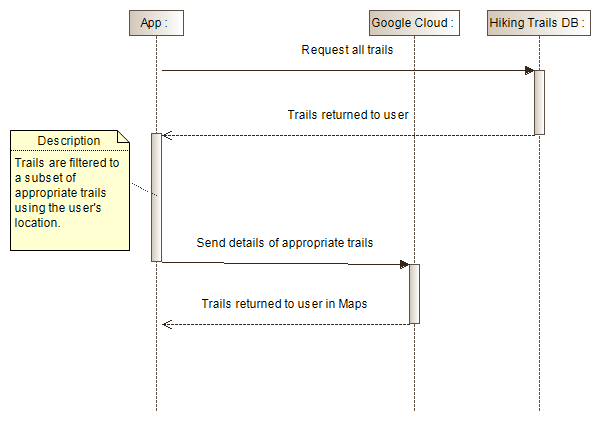
The interfaces that the application will be interacting with are the Trails API as well as the Google Maps API. Using a user’s location, Google Maps should filter trails from the Trails API, and return to the user the appropriate trails.

The initial plan for this has changed. In order to comply with GDPR and to ensure confidentiality, the user’s location will not leave their device. Therefore all Trails will be sent to the user from the Trails API. The work will then be done locally to calculate, using the user’s location, which routes are appropriate. These routes will then be visualised using Google Maps. The user’s location will not leave their device. If this changes in future than other concerns will arise regarding secure storage of users data

**Sequence Diagram**



**New Sequence Diagram**

****

