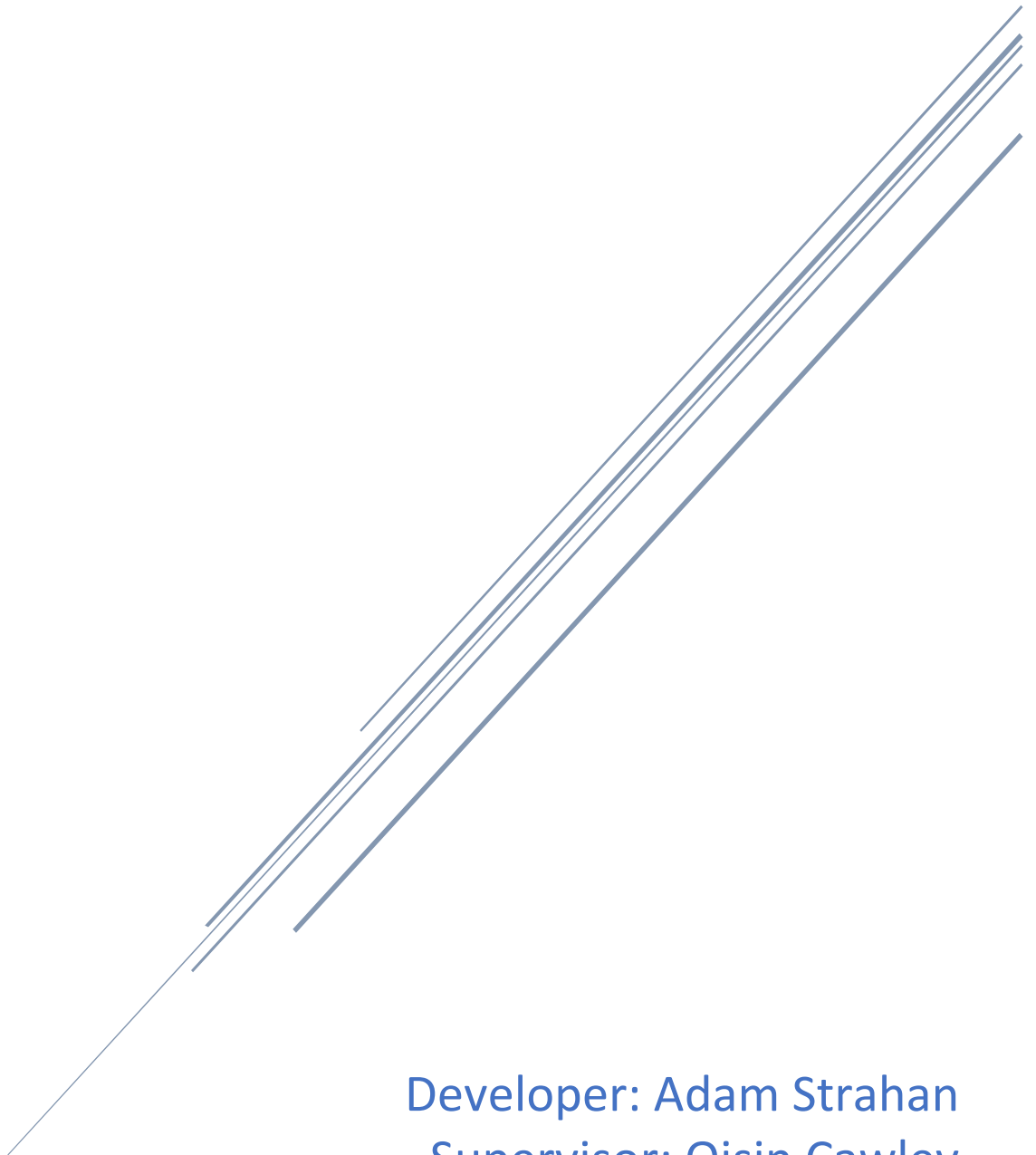


INTERACTIVE ITC MAP

FUNCTIONAL SPECIFICATION



Developer: Adam Strahan

Supervisor: Oisín Cawley

Contents

Table of Figures	2
Table of Tables	2
Introduction	3
Project Description.....	4
Users	4
Use Case Diagram	5
Use Cases	6
FURPS+	9
Functionality	9
Core Functionality.....	9
Secondary Functionality.....	9
Usability	10
Reliability.....	10
Performance	10
Supportability.....	10
'+'	10
Metrics	11
Testing.....	11
Appendix	12
Plagiarism Declaration	12

Table of Figures

Figure 1. Application Use Case Diagram	5
--	---

Table of Tables

Table 1. View Map Use Case	6
Table 2. View Room Details Use Case	6
Table 3. Plot Route Use Case	6
Table 4. Create Account Use Case	6
Table 5. Edit Account	7
Table 6. Delete Account Use Case	7
Table 7. Create Location Use Case	7
Table 8. Edit Location Use Case	8
Table 9. Delete Location Use Case	8

Introduction

For this project I am creating an interactive map of the Institute of Technology Carlow. This functional specification will outline the more technical aspects of the project, including:

- A description of the project.
- Details about the users.
- A use case diagram.
- Use cases.
- FURPS+.
- Metrics by which we can judge the success of the project.
- Testing for the project.

Project Description

The Institute of Technology Carlow campus can be daunting for new students, especially after spending 2 years learning from home during the Covid-19 pandemic. That combined with the Institute's continued growth, becoming the '*Technological University for South-East Ireland*' (TUSEI) makes it clear that newcomers to the campus will need an easy and convenient method to help them find their way around. That is where my project comes in.

The idea for this application is to make a map that will allow staff and students to find classrooms and other points of interest on the college campus easily. It is a lightweight application for smartphones that will allow users to enter their current location (or utilize the phone's GPS) along with the classroom or point of interest they wish to get to, then let the application plot the shortest route for them to take to get there. They will also be able to view a map of the campus, zoom in on buildings and see the building layouts. Users will also be able to click on rooms to get more information about them, such as what classes usually take place and when.

College staff will also be able to create accounts. This will allow them to:

- Create, edit, and delete buildings.
- Create, edit, and delete rooms.
- Create, edit, and delete other campus points of interest.

By allowing authenticated staff members to have these privileges, we can future proof the application by allowing it to grow and change with the campus.

Users

For this document, every standard user who will use the application will be referred to as '*students*'. They are people who utilize the application to find points of interest and plot routes. They have no special privileges within the application.

There are also users who will be registered and will be able to edit rooms and building details, along with other points of interest. These accounts will be reserved for college staff so they can update the campus layout as it continues to grow. These users will be referred to as '*admin*' in this document.

Use Case Diagram

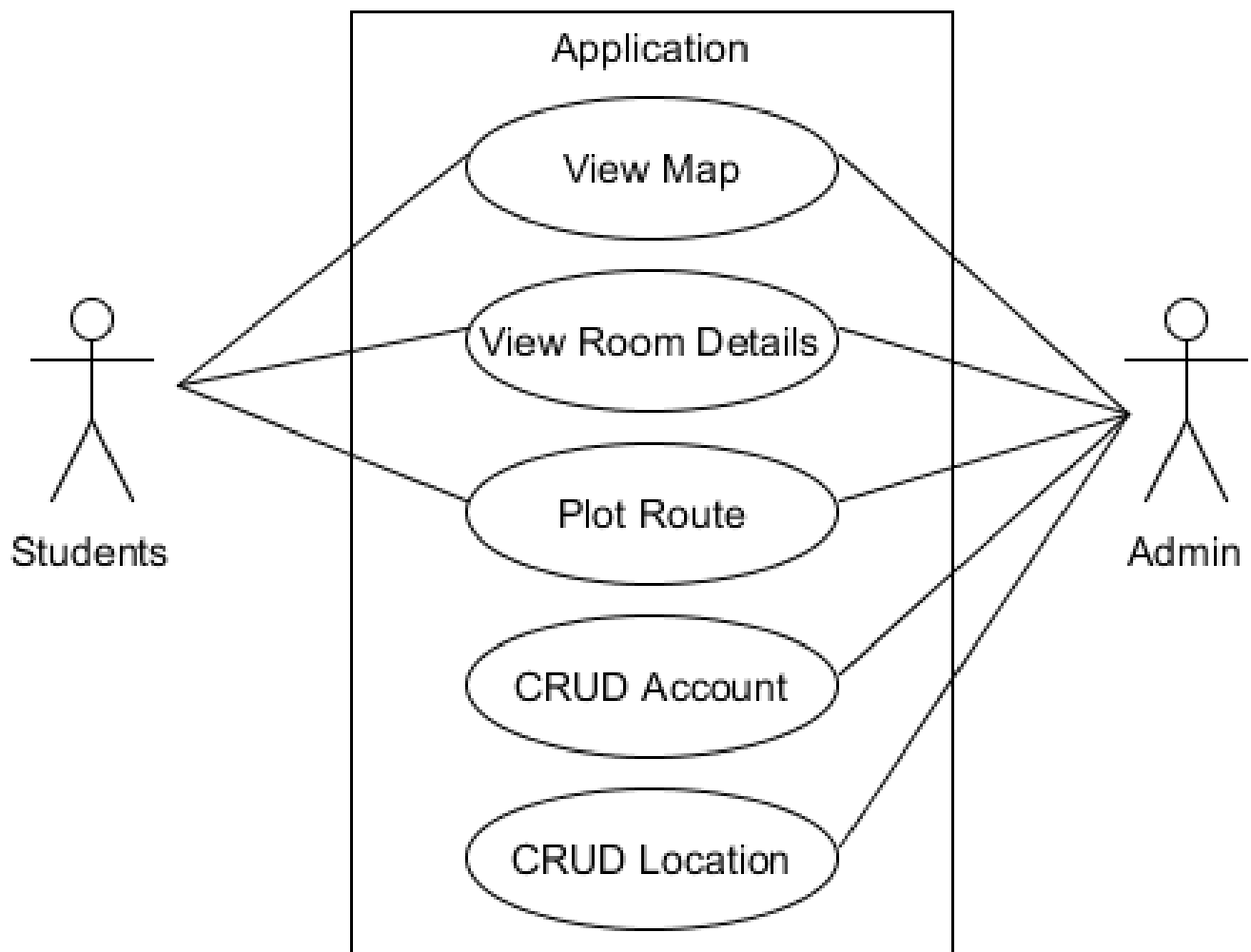


Figure 1. Application Use Case Diagram

Use Cases

Name	View Map
Actors	Students/Admin
Preconditions	Student/Admin has downloaded the application
Activity	<ol style="list-style-type: none"> 1) Student/Admin opens the application 2) Student/Admin clicks on the map
Postconditions	Student/Admin has opened the map
Alternatives	<ol style="list-style-type: none"> 1) Student/Admin opens the application <ol style="list-style-type: none"> 1a) Student/Admin chooses to close the application

Table 1. View Map Use Case

Name	View Room Details
Actors	Students/Admin
Preconditions	Student/Admin has opened the map
Activity	<ol style="list-style-type: none"> 1) Student/Admin types in room number 2) The map shows the entered room
Postconditions	Student/Admin can read the room details
Alternatives	<ol style="list-style-type: none"> 1) Student/Admin types in unknown room number <ol style="list-style-type: none"> 1a) Error message is displayed

Table 2. View Room Details Use Case

Name	Plot Route
Actors	Students/Admin
Preconditions	Student/Admin has opened the map
Activity	<ol style="list-style-type: none"> 1) Student/Admin enters room number 2) Student/Admin chooses to get directions
Postconditions	The route to the room is shown on the screen
Alternatives	<ol style="list-style-type: none"> 1) Student/Admin clicks on a room on the map <ol style="list-style-type: none"> 1a) Student/Admin chooses to get directions

Table 3. Plot Route Use Case

Name	Create Account
Actors	Admin
Preconditions	Admin has opened the application
Activity	<ol style="list-style-type: none"> 1) Admin chooses create account 2) Admin enters their details 3) Admin submits their details
Postconditions	Admin account has been created
Alternatives	<ol style="list-style-type: none"> 3) Admin submits incomplete/incorrect details <ol style="list-style-type: none"> 3a) Error message is displayed

Table 4. Create Account Use Case

Name	Edit Account
Actors	Admin
Preconditions	Admin has an Admin account and is logged in
Activity	<ol style="list-style-type: none"> 1) Admin chooses to edit their details 2) Admin enters new details 3) Admin submits new details 4) Admin re-enters password
Postconditions	Admin has edited their account details
Alternatives	<ol style="list-style-type: none"> 4) Admin enters incorrect password <ol style="list-style-type: none"> 4a) Error message displays 4b) Admin is returned to the details form

Table 5. Edit Account

Name	Delete Account
Actors	Admin
Preconditions	Admin has an Admin account and is logged in
Activity	<ol style="list-style-type: none"> 1) Admin chooses to delete their account 2) Admin confirms they wish to do this 3) Admin re-enters password
Postconditions	Admin account has been deleted
Alternatives	<ol style="list-style-type: none"> 2) Admin chooses not to delete their account <ol style="list-style-type: none"> 2a) Admin is returned to their account screen 3) Admin enters incorrect password <ol style="list-style-type: none"> 3a) Error message displays 3b) Admin is returned to the delete account screen

Table 6. Delete Account Use Case

Name	Create Location
Actors	Admin
Preconditions	Admin has an Admin account and is logged in
Activity	<ol style="list-style-type: none"> 1) Admin chooses to create a new location 2) Admin enters details of new location 3) Admin submits details of new location 4) Admin re-enters password
Postconditions	New location has been added
Alternatives	<ol style="list-style-type: none"> 3) Admin chooses not to submit details of new location <ol style="list-style-type: none"> 3a) Admin is returned to add new location screen 4) Admin enters incorrect password <ol style="list-style-type: none"> 4a) Error message displays 4b) Admin is returned to add new location screen

Table 7. Create Location Use Case

Name	Edit Location
Actors	Admin
Preconditions	Admin has an Admin account and is logged in
Activity	<ol style="list-style-type: none"> 1) Admin chooses to edit location details 2) Admin enters new details 3) Admin submits new details 4) Admin re-enters password
Postconditions	Location details have been edited
Alternatives	<ol style="list-style-type: none"> 4) Admin enters incorrect password <ol style="list-style-type: none"> 4a) Error message displays 4b) Admin is returned to the location details form

Table 8. Edit Location Use Case

Name	Delete Location
Actors	Admin
Preconditions	Admin has an Admin account and is logged in
Activity	<ol style="list-style-type: none"> 1) Admin chooses to delete a location 2) Admin confirms they wish to do this 3) Admin re-enters password
Postconditions	Location is deleted
Alternatives	<ol style="list-style-type: none"> 2) Admin chooses not to delete the location <ol style="list-style-type: none"> 2a) Admin is returned to the map 3) Admin enters incorrect password <ol style="list-style-type: none"> 3a) Error message displays 3b) Admin is returned to the map

Table 9. Delete Location Use Case

FURPS+

'FURPS' is an acronym for Functionality, Usability, Reliability, Performance, Supportability and is used to outline the functional requirements of a project. The '+' was added to outline any additional considerations for non-functional requirements such as design, implementation, interface, or physical constraints. FURPS+ is used widely in the Software Development Industry as it may be used easily to outline many of the requirements quickly.

Functionality

This refers to the main functionality and features of the application. This can be divided into 2 sub-categories: Core functionality and secondary functionality.

Core Functionality

The core functionality of the application is comprised of the essential functions the application needs to be considered complete. The application should have the following core functionality:

View the map

Allowing all users to view the map is the most important part of the applications functionality. Without the map there can be no plotted routes, thus making the application useless.

Select a location

Selecting a location is vital for plotting the routes. These locations will be the nodes in my graph and the ability to select one is essential.

Plot a route

Creating a pathfinding application is the primary selling point for this project. It is essential that any user of the application can plot a route from a chosen location to a chosen destination.

Admin accounts

Creating, editing, and deleting admin accounts will be necessary for the long-term maintenance of the application. As the campus expands buildings and room details may need to be updated. It must also be a feature that is accessible to verified users who work at the college. If unverified (student) users had access to this feature it would be a nightmare to maintain and ensure accuracy.

Secondary Functionality

The secondary functionality of the application is comprised of the rest of the functions that are non-essential but make the application better. The application should have the following secondary functionality:

Displaying room details

While displaying room details is not essential for the application to function, it would make for a better and more informative user experience.

Usability

Usability refers to the user experience of using the application. This can often be measured by the time it takes for tasks to be completed. Usability will be a key part of making this application a success. If we want people to use the application, it needs to be easy and convenient for users. The application should:

- Display the map within 2 seconds 98% of the time.
- All forms and inputs should be clearly marked and labelled allowing for ease of use. This should lead to forms being completed within 2 minutes 95% of the time.
- Searching for a room or location should take less than 10 seconds 99% of the time.

Reliability

This refers to how reliable the software is and can often be measured by considering the applications downtime. The application should:

- The application should have downtime of less than 5 hours per year.

Performance

The application should:

- Have an average response time of 1 second. This will be sufficient for the application to run efficiently.
- Launch in less than 5 seconds 99% of the time.
- Plot a route in less than 10 seconds 95% of the time.

Supportability

Supportability refers to multiple things, such as having readable code, supporting multiple languages and releasing on multiple platforms. This application should:

- Be usable on Android phones at launch, with iOS to follow.
- Have well written, structured and commented code. This will make it easier for future developers to maintain it or expand it.
- Launch with English support as English is the dominant language in the Carlow region. Additional languages can be added in the future.

'+

Security

Security is an integral part of any application and this one is no different. Having proper security features and protocols will protect users and the application from malicious users. To be sufficiently secure, the application should:

- Lock a user out of an account for 5 minutes after 5 incorrect password attempts.
- Ask for a password before completing any admin tasks, such as creating, editing, or deleting locations on the map.
- Enforce passwords with a minimum of 8 characters, 1 uppercase letter, 1 lowercase letter, and a special character.

Metrics

Metrics in the context of this project describes how the success of the application may be measured. The success of the application can be measured by the following metrics:

- The application should be launchable on Android phones, followed by iOS at a later date.
- Students and Admins should be able to view the map.
- Students and Admins should be able to view the details of the rooms in the college.
- College Staff should be able to create, edit and delete admin accounts.
- Admins should be able to create, edit and delete locations on the map.
- Admins should be able to log into their account only.
- Admins should be able to log out.
- Students should not be able to access an admins account.
- Students should not be able to create, edit or delete an admin account.
- Students should not be able to create, edit or delete locations on the map.
- The map should find the shortest available route from the entered starting point to the chosen location.

Testing

The testing for the application will be done as it is being built. I will be testing the functionality as I build it. As I am using Java and Android Studio to make the application, I will be able to use the developer settings on my phone to install and test the application on it. I would also like to get neutral testers in to test the application as it gets closer to completion to get a few unbiased opinions on the software.

Appendix

Plagiarism Declaration

Projects & Plagiarism

All projects are individual projects. The project is expected to be all your own work. Under certain circumstances, you may be able to incorporate material from other sources. This "incorporation" needs to be discussed with, and agreed upon by, lab supervisor and all such material must be referenced appropriately.

- Plagiarism is defined as " ... *presenting someone else's work as if it were your own, whether you mean to or not. 'Someone else's work' means anything that is not your own idea, even if it is presented in your own style. It includes material from books, journals or any other printed source, the work of other students or staff, information from the Internet, software programs and other electronic material, designs and ideas. It also includes the organization or structuring of any such material ...* " [taken from: Victoria University of Wellington, New Zealand, <http://www.victoria.ac.nz/home/glossary.aspx#p>].
- What this means is that it is wrong to copy another's work or download material for incorporation into your project from the Internet (without the express permission of your supervisor). Even if you take someone else's algorithm/code and rewrite it, this must be documented in your program to state where you got the original algorithm from. Failure to do so is considered plagiarism. If you are in any doubt, always ask your lab supervisor.
- Plagiarism is a **serious academic offence** (equivalent to cheating on an exam) and the consequences of being found guilty of plagiarism are severe.
- Students can use the internet to research information on their projects, they can ask lab supervisors for help with their project and they can also discuss problems with their fellow class mates.
- Students should be able to explain fully their code and may be asked to do so by one of your lab supervisors.

Work submitted for assessment which does not include this declaration will not be assessed.

DECLARATION

*I declare that all material in this submission e.g. thesis/essay/project/assignment is entirely my/our own work except where duly acknowledged.

*I have cited the sources of all quotations, paraphrases, summaries of information, tables, diagrams or other material; including software and other electronic media in which intellectual property rights may reside.

*I have provided a complete bibliography of all works and sources used in the preparation of this submission.

*I understand that failure to comply with the Institute's regulations governing plagiarism constitutes a serious offence.

Student Name: (Printed) ADAM STRAHAN

Student Number(s): COO216499

Signature(s): Adam Strahan

Date: 26/10/21

Please note:

The Institute regulations on plagiarism are set out in Section 10 of Examination and Assessment Regulations published each year in the Student Handbook.