Functional Specification

INSTITUTE of TECHNOLOGY
CARLOW
Institiúid Teicneolaíochta Cheatharlach

Interactive Map of IT Carlow

Name: Daniel Polak
Date: 27/11/2020
Supervisor: Hisain Elshaafi

Table of Contents

Table of Figures

Abstract

The purpose of this document is to provide an account of the functionality that will be provided by the Interactive Map of IT Carlow app. This document will outline the functionalities of the system and how they are implemented.

1. Introduction

Interactive map of IT Carlow is an app that will help people that have trouble finding the buildings/room on the campus. Users can choose to use the GPS to get their precise location or use one of the available points on the map as a starting point. Users will then specify the location they want to go to. The aim of this app is to provide the map of IT Carlow campus to students, staff and guests to make it easier for them to find classrooms and buildings. The app will provide registered users an option to get notifications for their classes.

2. Use Case Diagram

The purpose of this diagram is to visualise the main functionality of the app at high level.

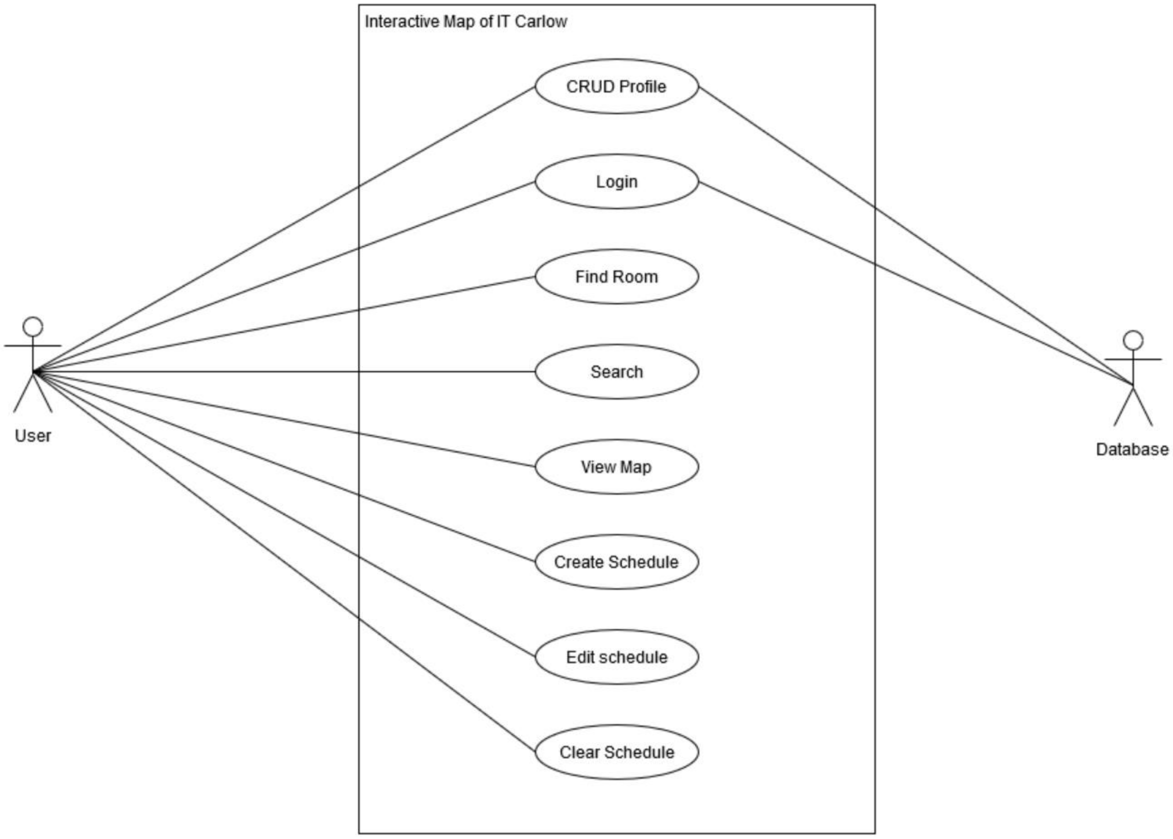


Fig. 1: Use Case Diagram

3. Use Case

Name: CRUD Profile
Actors: User, Database
Description: User tries to sign up.
Main Success Scenario:

1. User chooses to create a profile.
2. User clicks on ‘Signup’ in ‘Main Menu’.
3. User fills out the form that is presented with their details such as name, email and password.
4. User clicks on the ‘Signup’ button at the bottom of the form.
5. The app sends the information to the database.
6. Database stores the information.
7. The app displays a message “Sign up successful” to the user.
8. The user is redirected to the ‘Login’ page.

Alternatives:

3a. User entered an existing email.

1. The app informs the user that the email already exists and goes back to step 3.

6a. Database cannot store the information

1. The app informs a user that it is unable to upload information to the database.
2. The app stores informs a user to try again later.
3. User goes back to step 3.

Name: CRUD Profile
Actors: User, Database
Description: User tries to update their profile information.
Main Success Scenario:

1. User clicks on ‘My Profile’ in ‘Main Menu’.
2. User clicks ‘Change Info’ in profile.
3. User is presented with a form and fills it out with information such as Name, email and password.
4. User clicks the ‘Update’ button at the bottom of the form.
5. The app sends information to the database.
6. The database changes the information in the database.
7. The app displays a message “Updated successfully” to the user.
8. User is redirected to the ‘Profile’ page.

Alternatives:

3a. User entered an existing email.

1. The app informs the user that the email already exists and goes back to step 3.

6a. Database cannot store the information

    1. The app informs a user that it is unable to upload information to the database.
    2. The app stores informs a user to try again later.
    3. User goes back to step 3.

Name: CRUD Profile
Actors: User, Database
Description: User tries to remove their profile.
Main Success Scenario:

1. User clicks ‘My Profile’ in ‘Main Menu’.
2. User clicks ‘Remove Profile’ on the profile page.
3. User is prompted to type in their password.
4. The app presents the user with a message “Are you sure you want to remove your profile?”.
5. The user clicks the ‘Confirm’ button.
6. Database flags the information as deleted.
7. The app displays a message “Profile Deleted”.
8. The user is redirected to the ‘Home’ page.

Alternatives:

5a. User does not want to remove their profile..

1. User clicks on the ‘Cancel’ button and goes back to step 2.

Name: Login
Actors: User
Description: The user tries to login.
Main Success Scenario:

1. User clicks ‘Login’ in ‘Main Menu’.
2. User fills out the form presented with email and password and clicks on the ‘Login’ button at the bottom of the form.
3. The app verifies the email and password.
4. User is redirected to ‘Main Menu’.

Alternatives:

2a. User entered the wrong email and/or password.

1. The app informs the user that the email and/or password is incorrect.
2. User goes back to step 2.

Name: Search
Actors: User
Description: The user tries to search a room.
Main Success Scenario:

1. User clicks ‘Find’ in ‘Main Menu’.
2. User chooses the building from a dropdown list.
3. User chooses the room from a dropdown list.
4. The app calculates the path to the location chosen.
5. User is redirected to the map.

Name: View Map
Actors: User
Description: The user tries to View the Map.
Main Success Scenario:

1. User clicks on ‘View Map’ in ‘Main Menu’.
2. User is redirected to the map.
3. User chooses their location on a map.
4. User chooses a building and a room on a map.
5. The app calculates the path to the desired location.

Name: Find Room
Actors: User
Description: The user tries to find a room.
Main Success Scenario:

1. User clicks ‘Find’ in ‘Main Menu’.
2. User chooses the building from a dropdown list.
3. User chooses the room from a dropdown list.
4. The app calculates the path to the location chosen.
5. User is redirected to the map.

Name: Create Schedule
Actors: User
Description: The user tries to create a new schedule.
Main Success Scenario:

1. User clicks ‘View Schedule’ in ‘Main Menu’.
2. User clicks on the ‘Create’ button in the ‘View Schedule’ screen.
3. User clicks on a day of the week and fills out information for each hour such as Class name, room, and lecturer.
4. User repeats step 3 until done.
5. User clicks on the ‘Save’ button on the ‘Create Schedule’ screen.
6. Schedule is saved.
7. User is redirected to the ‘View Schedule’ screen.

Alternatives:

6a. The app cannot store the information.

1. The app informs a user that it is unable to store the information and informs a user to try again later.
2. User goes back to step 5.

Name: Edit Schedule

Actors: User

Description: The user tries to update an existing schedule.

Main Success Scenario:

1. User clicks ‘View Schedule’ in ‘Main Menu’.
2. User clicks on the ‘Edit’ button in the ‘View Schedule’ screen.
3. User chooses the day of the week and fills out information such as Class name, room, and time.
4. User repeats step 3 until done.
5. User clicks on the ‘Save’ button on the ‘Update Schedule’ screen.
6. The app saves the schedule.
7. User is redirected to the ‘View Schedule’ screen.

Alternatives:

6a. The app cannot store the information.

1. The app informs a user that it is unable to store the information and informs a user to try again later.
2. User goes back to step 5.

Name: Clear Schedule

Actors: User

Description: The user tries to clear the schedule.

Main Success Scenario:

1. User clicks ‘View Schedule’ in ‘Main Menu’.
2. User clicks on the ‘Clear’ button in the ‘View Schedule’ screen.
3. The app asks for confirmation.
4. User confirms the clear.
5. The app clears the schedule.

Alternatives:

4a. User decides to not clear a schedule.

1. User clicks on the ‘Cancel’ button.
2. User goes back to step 1.

4. Metrics

The success of this application will be determined by the following criteria.

- The app should allow for the following:
    - creating/editing/removing a profile
    - logging in
    - creating/editing/removing a schedule
    - viewing a map
    - searching
    - finding a room.
- The app should find the shortest path to the location requested by the user.
- The app should notify the user before a class.
- The app should allow for secure login and storing of the information.

5. FURPS+

This section will outline all the requirements of the system under the following headings.

5.1. Functionality

- The app must always be connected to the internet to access the database.
- The user must be able to register/login.
- The user must be able to view the map.
- The user must be able to view/create/edit/remove a schedule.
- The user must be able to remove their profile.
- The user must be able to search for the room.
- The user must be able to find a room.

5.2. Usability

- The user should be able to login to the system with one click after filling out a form.
- A clear error message should be displayed if the user enters incorrect details.
- All forms should be labeled clearly and the user should be able to fill them out very quickly.
- All forms should take the average user 1 minute to finish.
- The user should be able to view the map in less than 10 seconds.
- The user should only be allowed to create a schedule once they have a profile.
- The user should be able to create a schedule in less than 2 minutes.
- The user should be able to edit a schedule in less than 30 seconds.
- The user should be able to clear a schedule in less than 5 seconds.
- The user should be able to search for the room in less than 10 seconds.
- The user should be able to remove their profile in less than 10 seconds.

5.3. Reliability

- The app must have 99% uptime.
- The app must have little to no bugs and handle all errors in a way that prevents it from crashing.
- The app must create/edit a schedule 99% of the time.

5.4. Performance

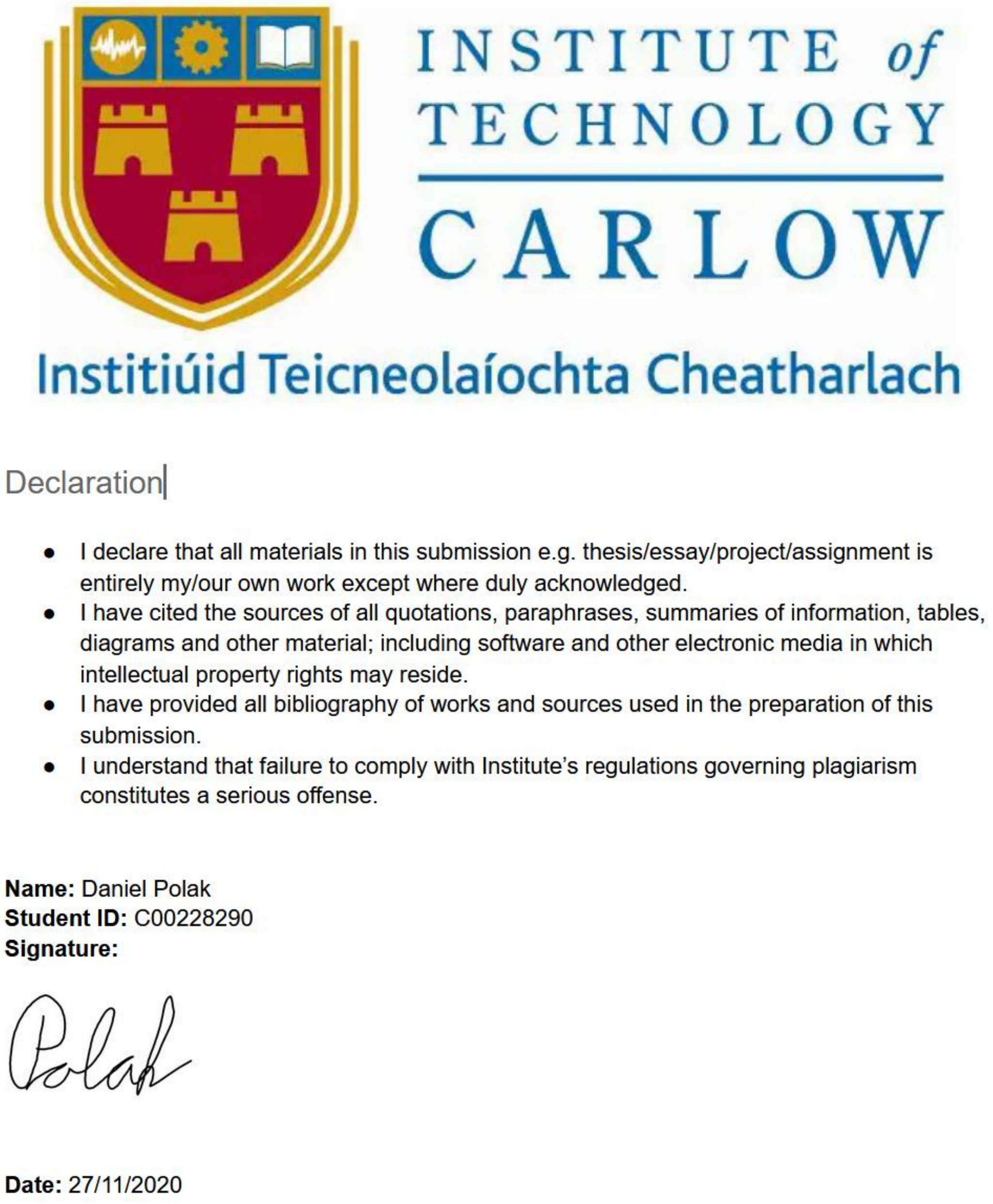
- The app must open in less than 5 seconds.
- Adding or editing records in a database should take less than 2 seconds.
- The app must calculate a path in less than 15 seconds.

- Each page should open in less than 2 seconds.
- All the features should be easy to find.
- The app should be scalable for different sized screens and resolutions.

5.5. Security

- The app must salt and hash a password of the user.
- User’s schedule must be kept secure.
- The password should contain at least one lowercase letter, one uppercase letter, one number and one special character. Password should be at least 8 characters long.
- An unauthenticated user should be locked out from the app for 2 minutes after 3 unsuccessful attempts.

6. Declaration



INSTITUTE *of* TECHNOLOGY

CARLOW

Institiúid Teicneolaíochta Cheatharlach

Declaration

- I declare that all materials 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 and other material; including software and other electronic media in which intellectual property rights may reside.
- I have provided all bibliography of works and sources used in the preparation of this submission.
- I understand that failure to comply with Institute's regulations governing plagiarism constitutes a serious offense.

Name: Daniel Polak
Student ID: C00228290
Signature:

Polak

Date: 27/11/2020