Submission of Thesis and Dissertation

National College of Ireland
Research Students Declaration Form
(Thesis/Author Declaration Form)

Name: Navjot Singh Virk

Student Number: x13112406

Degree for which thesis is submitted: BSC. Hons in Computing (2013-2017)

Project Name: Meeting Room Pro

Material submitted for award

(a) I declare that the work has been composed by myself.

(b) I declare that all verbatim extracts contained in the thesis have been distinguished by quotation marks and the sources of information specifically acknowledged.

(c) My thesis will be included in electronic format in the College

Institutional Repository TRAP (thesis reports and projects)

(d) *Either* *I declare that no material contained in the thesis has been used in any other submission for an academic award.

Or *I declare that the following material contained in the thesis formed part of a submission for the award of

_____

(State the award and the awarding body and list the material below)

Signature of research student: Navjot Singh Virk

Date: 1 August 2017

Submission of Thesis to Norma Smurfit Library, National College of Ireland

Student name: Navjot Singh Virk   **Student number:** x13112406

School: School of Computing   **Course:** BSc. Hons in Computing

Degree to be awarded: Level 8 (BSc. Hons in Computing

Title of Thesis: Final Year Project Report and User Manual – Meeting Room Pro

_____

One hard bound copy of your thesis will be lodged in the Norma Smurfit Library and will be available for consultation. The electronic copy will be accessible in TRAP (http://trap.ncirl.ie/), the National College of Ireland's Institutional Repository. In accordance with normal academic library practice all theses lodged in the National College of Ireland Institutional Repository (TRAP) are made available on open access.

I agree to a hard bound copy of my thesis being available for consultation in the library. I also agree to an electronic copy of my thesis being made publicly available on the National College of Ireland's Institutional Repository TRAP.

Signature of Candidate: Navjot Singh Virk
Date: 1 August 2017

For completion by the School:
The aforementioned thesis was received by_____
Date:_____

This signed form must be appended to all hard bound and electronic copies of your thesis submitted to your school

Final Project Report and User Manual

Project Name: MeetingRoom Pro (MrPro)

Also, Available on Github:

Project Code: https://github.com/Virksaabnavjot/MRPro

Project's Business Website: https://navsingh.org.uk/mrpro/

Projects A-Z Documentation:
https://github.com/Virksaabnavjot/Documentation-MeetingRoom-Pro

Project Aim: The aim of the project is to develop a client IOS application and a Web-API that exposes data from the database to client.

Project Summary: In day to day life weather its work, college or hotel buildings people find it hard to find rooms, even if the people have been working in the same company for years they are not familiar with all parts of the building and the rooms available in it. The problem arises when there is a time shortage and you have to find the room and make on time for the meeting.

So, MRPro will help its users to find and navigate to the different parts of the building.

Students face these problems too, MRPro can help new students to get familiar with their university campus, the app can also be used by city council for different buildings and malls to assist the people (everyone including people living in the city and tourists), moreover hotels can make use of the application to provide exceptional experience for their customers.

Key Features:

IOS APP Features: MRPro allows you to - Find, Book, Review, Look at the Room Photos, Upload Photos, Help Navigate to the room using a map with building drawn on it and room pointed on it.

WEB API Features: Retrieve data from the database and return it to the IOS application in JSON format when requested.

Name: **Navjot Singh Virk**

Student Number: **X13112406**

Email: **virksaabnavjot@gmail.com**

Year: **4th (Software Development Stream)**

Course: **BSc. Honors in Computing at National College of Ireland** *(2013-2017)*

Table of Contents

1 User Manual

Steps to use the application.

Step 1: Install the app

Step 2: Make sure you have an active internet connection to use the application.

Step 3: Open the app

Step 4: Sign in/register as a user (the companies can limit user signup to their company email addresses only or admin can have the choice to pre-register users and only provide only login option).
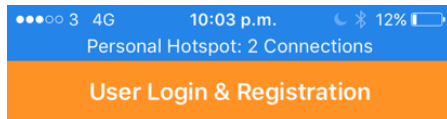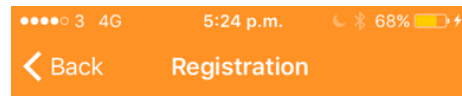
Step 5: Login with your detail
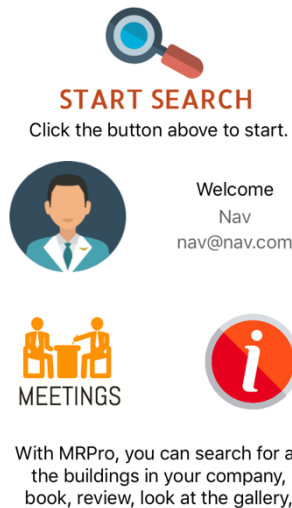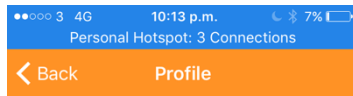
Step 6: You will see your profile page

And help page

Provide profile page and description, include image and point out the navigation options

Here, on this screen you will see your profile details, like your username and email, from here you can procced to – Building List and Search, or you can open the help page that provides short user manual within the app to help users quickly learn to use the app.
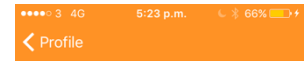
Step 7: Building List and Search – Here you can search and find a list of available buildings (the buildings can be from different locations (buildings in different cities of the countries i.e. city-country), that is globally a building can be in Dublin-Ireland, Galway-Ireland, Bangalore-India)

Step 8: Select a building from the available list.
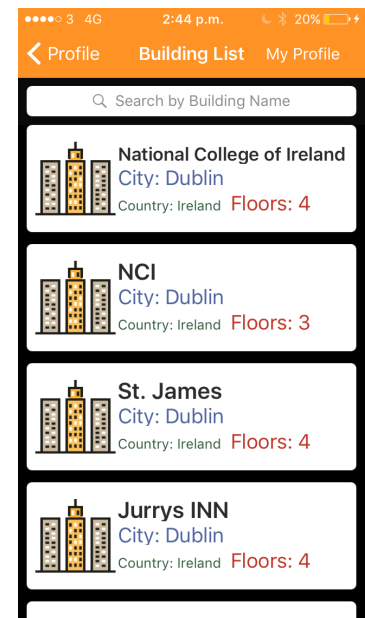
Step 9: Select a meeting room – from the available list of meeting rooms.

Step 10: On the next screen, you can see the building on map, see your current location, information about the meeting room like floor number, name, capacity etc.

- Look at the gallery (Photos of the room)
- Make a Phone call.
- Upload Photos (Upload new photos of the room to the gallery)
- Book the room
- Upload photos
- View map
- View your current location
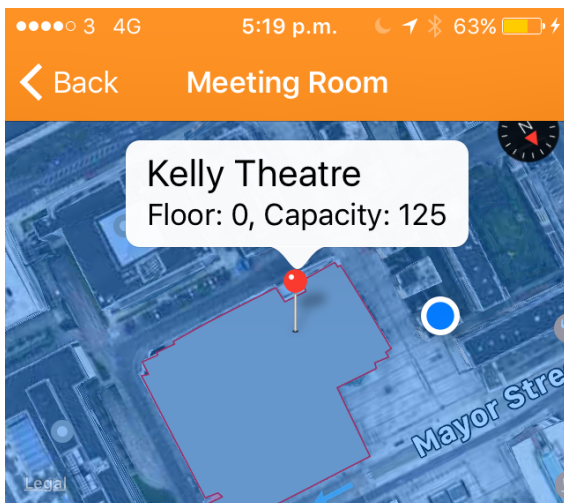
Meeting Room Information (Kelly Theatre)

Name:	Kelly Theatre
Parent Building:	National College of Ireland
Capacity:	125
Email:	kelly@ncirl.ie
Coordinates:	53.348633 ,-6.243084
e:	+353892110549
Type:	Theatre Room
Floor Number:	0

Directions : On entrance walk straight on you
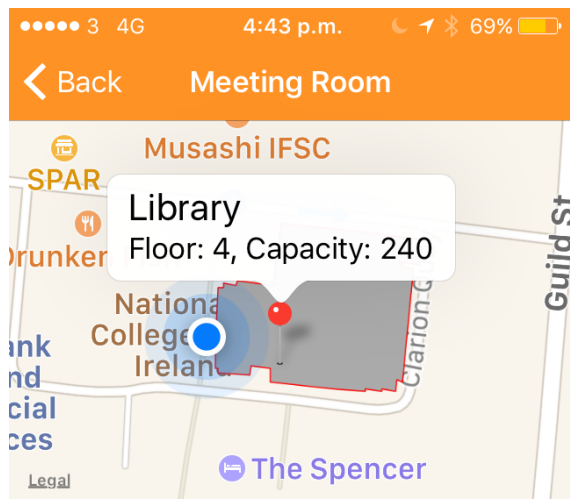
Meeting Room Information (Library)

Name:	Library
Parent Building:	National College of Ireland
Capacity:	240
Email:	library@ncirl.ie
Coordinates:	53.348556 ,-6.243047
e:	+353892110549
Type:	Study Room
Floor Number:	4

Directions : Walk straight from the entrance
your left clode to lifts

➔ **Find all information, regarding room like room's primary contact's email and phone etc.**

Parent Building:	National College of Ireland
Capacity:	240
Email:	library@ncirl.ie
Coordinates:	53.348556 ,-6.243047
Phone:	+353892110549
Type:	Study Room
Floor Number:	0

Directions : Walk straight from the entrance on your left clode to lifts

Book this Room

Photo Gallery for this Room

Upload photos of this Room

Review this Room

Photo Gallery for Room
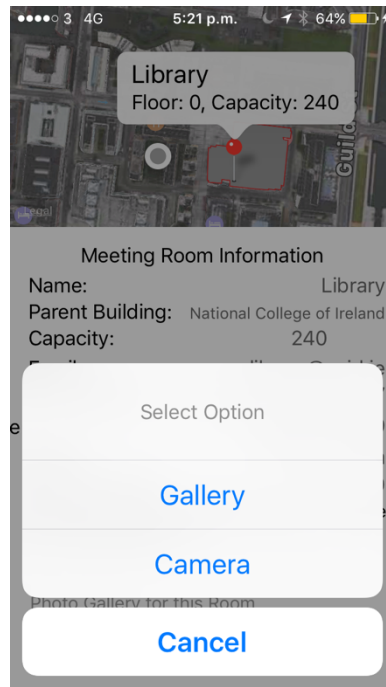
Photo Upload Feature
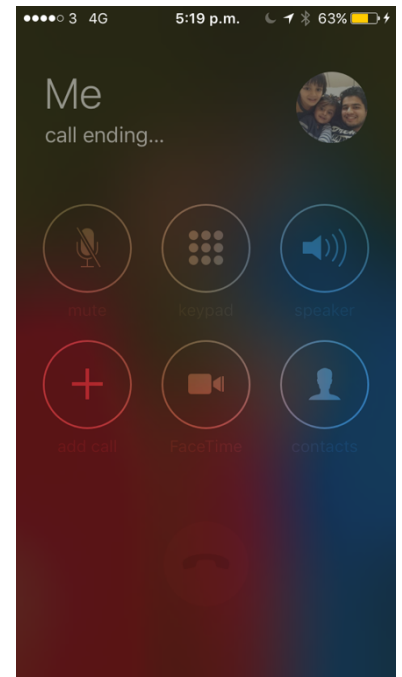
You can make a call
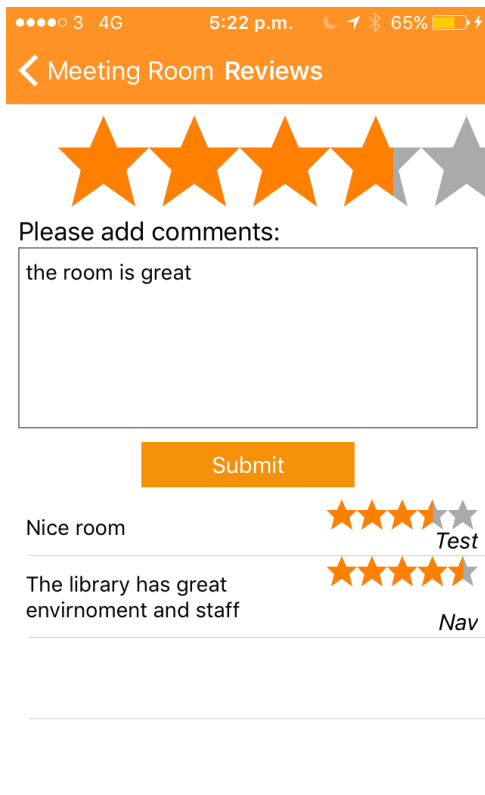
Review Rooms

Or

➔ **Book a Meeting Room**

End of User Manual

Navjot Singh Virk
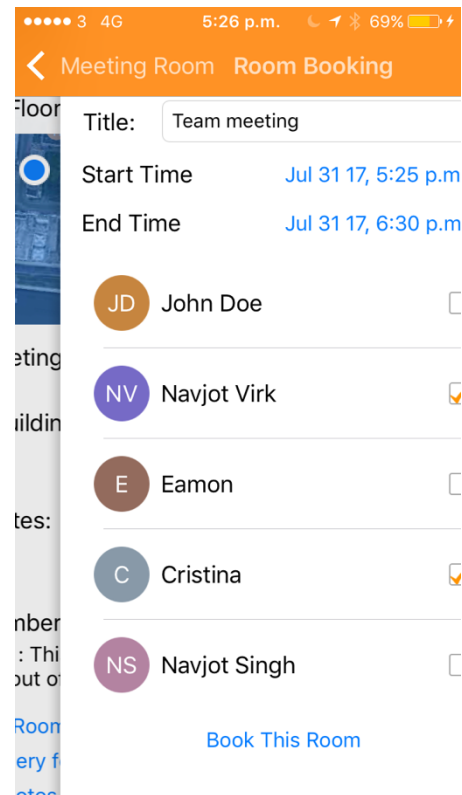
2 Executive Summary

With MeetingRoom Pro I have looked to develop a user friendly IOS application to solve the problem of finding meeting rooms in large buildings. I aimed to address a problem I had encountered personally in the past during my work placement and other people i.e. finding meeting rooms is complex and inconvenient. The application is designed in a way that it can be easily tailored for the use of any organization, with change in data provided through API i.e. building and meeting room coordinates. The app will be available for iPhones and iPads. The application will help people find rooms with ease and navigate through the building with their location shown on the map and also have the option to book the rooms straight from the app.

3 Introduction

3.1 Background

During my work placement at SAP SE, I was engaged in several meetings from my induction to my farewell meeting, there were more than 50 meetings, from team meetings to global team meetings, intern meetings and HR meetings, to employee farewell meetings which I was part of in the course of 7 months of internship.

And number one issue was finding the meeting room due to big size of the buildings, if the meeting room was close enough to my desk, it was easy enough to find the meeting room but the problem used to arise when the meeting rooms were on different floors, even different wing of the same floor and the problem was big when trying to find a meeting room on another building and some fancy names to the meeting rooms didn't make much sense at all and the icing on the top if you are short on time i.e. have consecutive meetings to attend and the only solution available was to ask you collages if they didn't know run towards the reception and they would show you a floor plan which sometimes didn't made sense and overall experience in finding meeting rooms was not up to the standards to which it could be achieved.

I started some research and found other employees were facing the same issues and were just limited to using meeting rooms near their offices due the hassle of finding meeting rooms, after talking to people the user research findings were not just the interns who felt this pain, some of the very senior employees which were there for many years didn't know where some of the meeting rooms were and since SAP is multinational company, there were always employees from different location which travelled to attend meetings and had very hard time finding the meeting room and first have to go to facilities department and the facilities member would manually assist wherever this process can be automated i.e. a simple to use application could solve this problem.

Hence, there is were the idea for the application is developed from but it is not just limited to corporate world the application can be used in universities and college with minor changes.

3.2 Aims

The aim of the project was to develop an easy to use mobile application that will help people find meeting rooms with ease in a big building and provide some other useful functionality like:

Map and Navigation – The user will select a meeting room and the application shall be able to show a detailed map with navigation instructions to the room.

Room Booking – When a user wants to book a room they can do.

I have described the rest of the functionality in the Systems Requirements section.

The application has the potential to be adopted internationally because of its scalability. There is very high possibility that business organizations, universities and colleges will buy the application for use for their staff and students. Essentially the app can become a valuable resource to any individual or organization and provide extraordinary results for people using it.

3.3 Project Commercialization

I found out during User testing that people are liking the idea and the app. As you will see in more detail in the User testing section and results. One example is during data gathering phrase I needed permissions from the college (NCI) authorities if I am permitted to take photographs of the rooms and atrium for the gallery feature of the app. I had to contact the college's marketing department and Accommodation & Commercial department for the permission and when I pitched the idea to them and presented a small demo of the app they loved it, great feedback people asked questions with excitement - Where can I download the app ?, Can I download this app on my android etc.

The marketing department specially looked very excited and suggested it would be really nice if our college have this app. And one person even suggested they would talk to their manager about college getting the app license. So, the response has been great.

Other, example is my past colleges at SAP they loved the idea when I pitched it to them during my internship, my friends who have seen the app demo loved it and asked many questions about it, so I strongly believe this project have a strong selling potential.

I wouldn't be wrong if I say most business projects are developed with the goal of making profits, commercialization of the project is a very important factor in the development of the project and moreover continued development of the software project. I believe this project has a great commercial scope as it can be used in multiple industries naming few of the potential customers –

Multinational Co-operate companies

Like - Google, Yahoo, Microsoft, SAP Apple Inc, etc

As these companies are huge, have buildings and meeting rooms all over the world, mostly multiple buildings at one location. MRPro can be used by these companies and their employees to find and book meeting rooms.

Colleges and Universities

Like NCI, DIT, UCD, TCD etc

For example, UCD has a huge campus MRPro will be a very useful tool for their staff, students and visitors etc.

City Council, Mall and Hotels

Like – Jervis Shopping Centre, Liffy valley Shopping Centre, Jurrys Inn, Hilton Hotel

MRPro will be a useful asset for Malls and hotels as this will provide easy navigation for customers, example a customer in a big hotel can find his/her room with ease or book, view photos of the room or review and many other things.

I have also developed a business website/page for the project available at: https://navsingh.org.uk/mrpro using wordpress.

3.4 Technologies

3.4.1 MVC

Front and Back-end both		**Software Architectural Pattern**
The project follows MVC, pattern on both front-end and back-end. As, we know CodeIgniter follows MVC by default, I have made effort to separate models, views and controllers on the app side (front-end) as well to adhere with MVC. Which, will keep the project clean, more maintainable, understandable and much more.		

3.4.2 Swift

Front-end		**Programming Language**
I decided on Swift 3 as my primary coding language for my application development. I came across Swift as a language during my internship and have a beginner level understanding of the language and would like to advance my understanding of the language through this project. Swift is a fairly new language and is open source and moving towards cross-platform which will		

open new possibilities in the future. I find Swift to be a fast and versatile language.

3.4.3 CodeIgniter

Back-end	CodeIgniter	**PHP Framework**

In the mid-term report I mentioned to use JavaScript for my backend but later in the development I decided to move forward with Code Igniter (because I decided to change my database from SAP Hana to MySQL database – Please refer technology changes section of the report for more details) which is a PHP framework which helps to easily develop web applications and api's.

3.4.4 Swifty Json

Front-end	SwiftyJSON	**Integration: Using Cocoa pods (in pod file)** ➔ **Pod 'SwiftyJSON'**

In Swift, I have experienced it's not very easy to parse JSON, So, I have decided to use an open source Swift library SwiftyJSON to help me with parsing JSON data that the application client will receive from web service.

3.4.5 Cocoapoda

Front-end	‹COCOAPODS›	**Cocoa pods are used as a dependency manager for iOS application projects.**

Cocoapods helps to easily add framework to your ios projects through pod file. Cocoapods installation details are added in Getting Started with Project development section of the report.

Please note: The project uses several other cocoapods other than SwiftyJson and alamofire. Please refer to the podfile in the xcode project folder for a list of all pods used.

Pod file example: ->

```
platform :ios, '9.0'

target 'MRPro' do
    # Comment the next line if you're not using Swift and don't want to use dynamic frameworks
    use_frameworks!

    # Pods for MRPro
    pod 'SwiftyJSON'
    pod 'Alamofire'
    pod 'ChameleonFramework/Swift'
    pod 'SwiftPhotoGallery'

    target 'MRProTests' do
        inherit! :search_paths
        # Pods for testing
    end

    target 'MRProUITests' do
        inherit! :search_paths
        # Pods for testing
    end

end
```

3.4.6 Alamofire

Front-end	ALAMOFIRE	Integration: Using Cocoa pods (in pod file) ➔ Pod 'Alamofire'

In Swift, NSURLConnection class helps to start and stop http URL requests, Alamofire is a library that helps make http request more elegantly with less code (and helps write more cleaner code too).

- Usage
  - **Intro -** Making a Request, Response Handling, Response Validation, Response Caching
  - **HTTP -** HTTP Methods, Parameter Encoding, HTTP Headers, Authentication
  - **Large Data -** Downloading Data to a File, Uploading Data to a Server
  - **Tools -** Statistical Metrics, cURL Command Output
- Advanced Usage
  - **URL Session -** Session Manager, Session Delegate, Request
  - **Routing -** Routing Requests, Adapting and Retrying Requests
  - **Model Objects -** Custom Response Serialization
  - **Connection -** Security, Network Reachability

Image Source: https://github.com/Alamofire/Alamofire

3.4.7 Xcode IDE

Front-end		Code Editor

Xcode IDE was chosen for the development of the application client as it's the default editor for developing IOS Swift apps. Other options were AppCode from JetBrains.

3.4.8 Github

		Version Control
For source control I am using Github. As, it allows me to keep my code available anywhere not worrying about losing it.		

3.4.9 PHP

Back-end	PHP	Server Side Scripting Language
Php is used for the development of Web API in CodeIgniter.		

3.4.10 MySQL

Back-end		Relational Database
MySQL is used to store data including geo-spatial data.		

3.4.11 phpMyAdmin

Back-end	phpMyAdmin	Database Management Tool
Used to manage database and easily insert data into the database tables		

Other technologies used:

WordPress (HTML,CSS, PHP and MySQL) is used to create a business website for the project. The aim of the website is to provide a professional outlook to the application project and work as a marketing tool, help attract customers, shareholder and potential investors etc.

3.4.12 Technology Changes

During the final development of this project some changes to proposed (during Mid-term) technologies were made. They are as follows –

SAP Hana was replaced with MySQL

Database change – Later, during the development of the project I decided to use MySQL database instead of SAP Hana which being used to store geo-spatial data.

Reason for Change: During the mid-term I struggled to find appropriate resources/solution to store geo-spatial data in MySQL, which lead me to use SAP Hana for geo-spatial data storage because some documentation was available for it. But, Hana came with some disadvantages which forced me move back to MySQL.

Problems with Hana: I was using Trail, which had certain limitations like database needed a restart every 12 hours and database was automatically removed/deleted if not used for 6 day. The pro version of Hana is expensive and I decided to research more on storing geo-spatial data in MySQL.

JavaScript replaced with PHP (CodeIgniter)

Language Change: When using Hana I was using JavaScript in .xsjs file whereas

Reason for Change: Using JavaScript was no more a must requirement when I moved from Hana to MySQL**.** I wanted to learn something new things and get job ready while doing this project and I had never worked with PHP before so when it came between choosing Java (Jersey) or PHP for Web API development I choose PHP so I can learn a new programming language.

4   System

4.1   Requirements

4.1.1   Functional requirements

Use Case Diagram
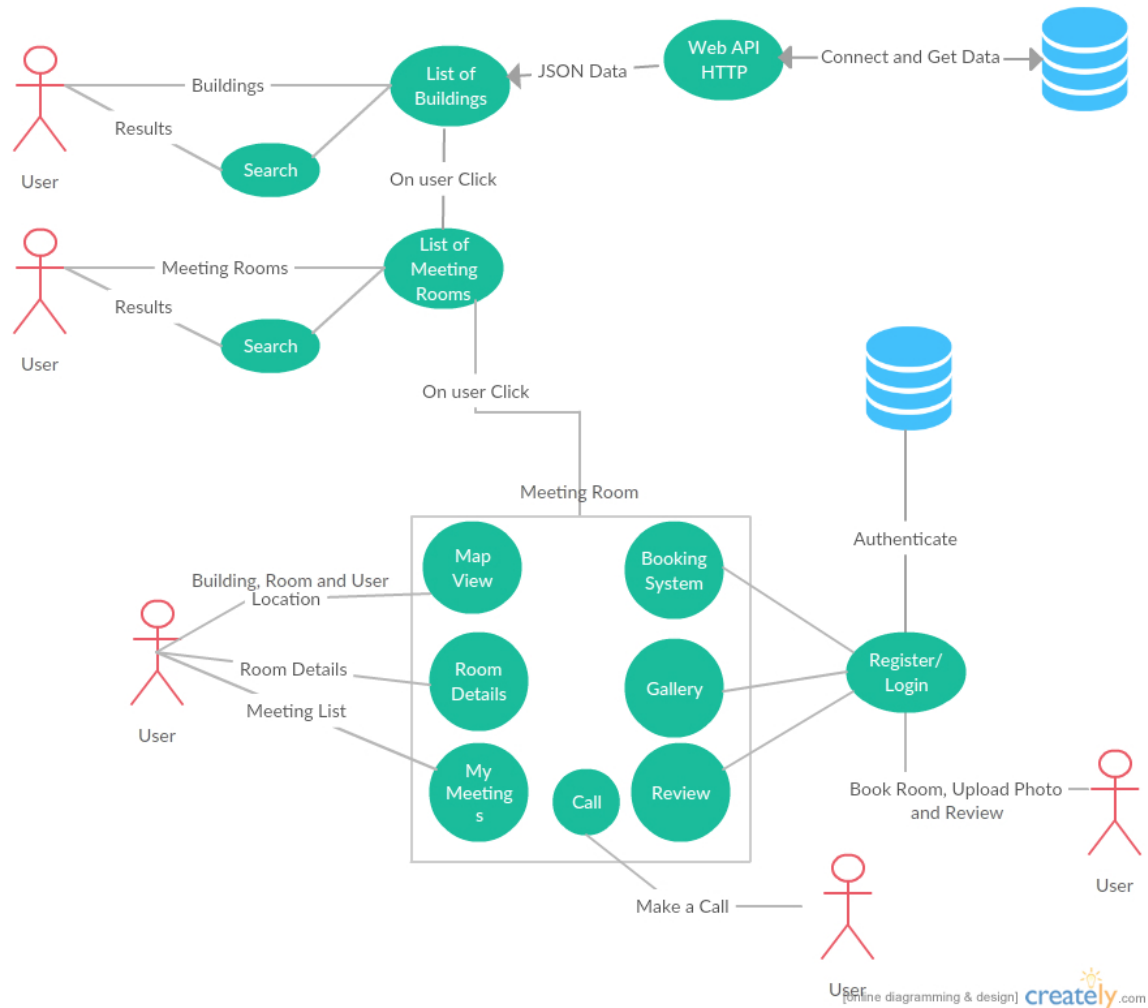
The above Use Case Diagram provides an overview of all functional requirements of this project.

4.1.2   Requirement 1: Easy to Use GUI

4.1.2.1   User Story

As a user I want to have an easy to use interface so as I can navigate through the app easily with little time needed to be spent learning how to use the app

4.1.2.2 Description & Priority

The system shall have an easy to use graphical user interface. Navigation will be possible through the touch screen functionality of the IOS device. The design shall also be responsive to support different screen sizes of iPhones and iPads.

Essential and High Priority Requirement.
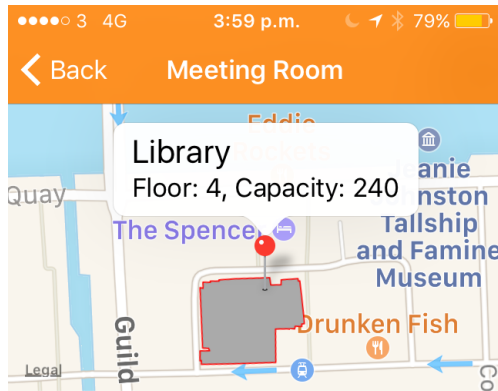
4.1.2.3 Requirement Activation

This requirement will be activated upon starting of the application.

4.1.2.4 Technical Issues

GUI Need internet connection to get data from API which will be shown in the application, slow internet connection can affect performance of the GUI.

4.1.2.5 Risks

Provision must be taking that not all users have a good understanding of technology and the use of mobile platforms. If the GUI is to become too complex, some users may not use the application. This requirement is at the core to the success of the system.

4.1.2.6 Dependencies on other requirements

N/A

4.1.3 Requirement 2: List and Find(Search)

4.1.3.1 User Story

As a user, I want to be able to find buildings and meeting rooms with ease. Like a clickable list of buildings which then browse to list of meeting rooms also search and I can select a meeting room and find it or use other features like room booking, gallery upload etc.

4.1.3.2 Description & Priority

The system shall allow to search and provide list of available buildings and meeting rooms.

High Priority
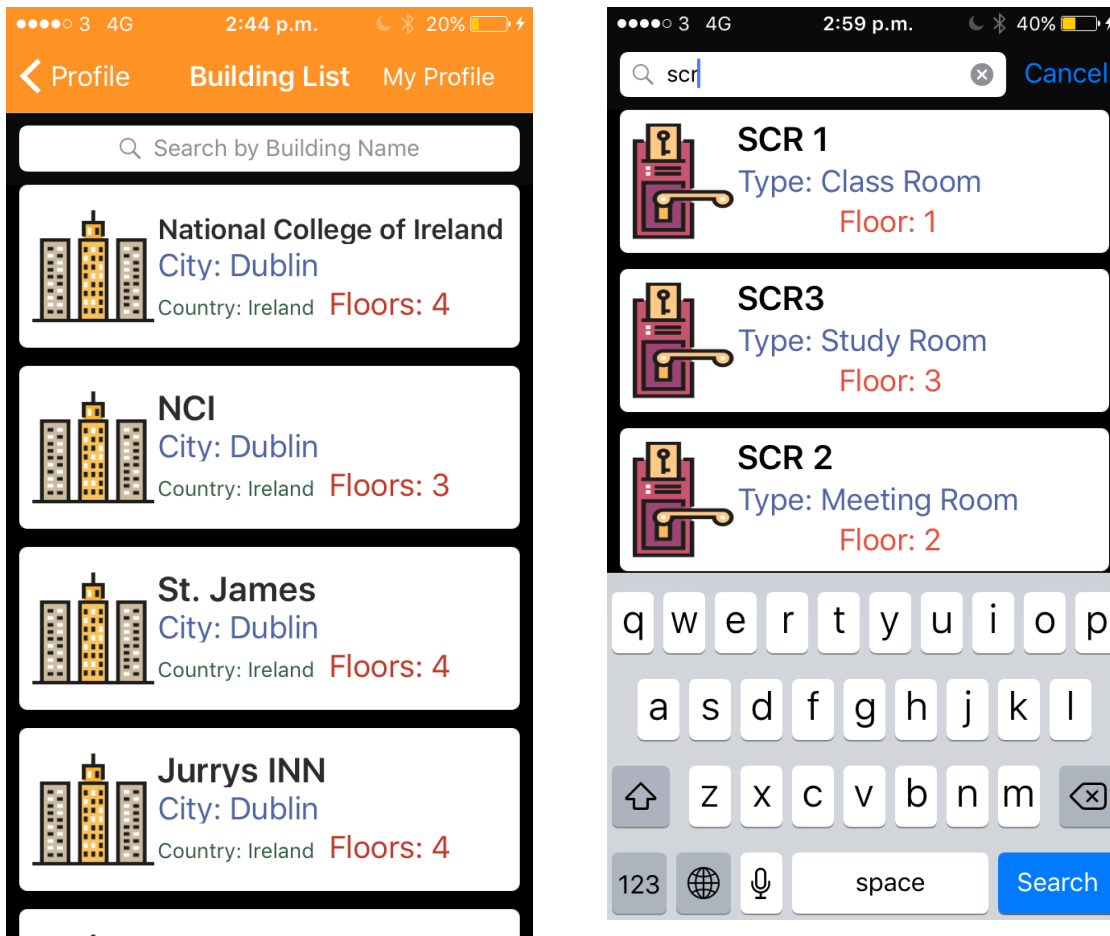
4.1.3.3 Requirement Activation

The requirement to list buildings and meeting rooms will be activated upon starting of the application and search is activated when user enters in search bar.

4.1.3.4 Technical Issues

N/A

4.1.3.5 Risks

This requirement is at the core to the success of the system as most of the requirements hereafter have a dependency on this one.

4.1.3.6 Dependencies on other requirements

This requirement has a dependency on requirement 1 - Easy to use GUI.

4.1.4 Requirement 3: Map / Navigation

4.1.4.1 User Story

As a user, I want to be able to view the building highlighted on the map. Also, the meeting room I am looking for along with my current location and some information about the building and room like floor number.

4.1.4.2 Description & Priority

The system shall have an expandable map with current location of the user and building drawn on the map.
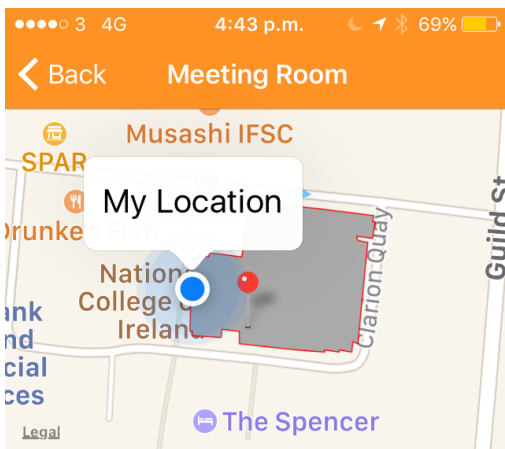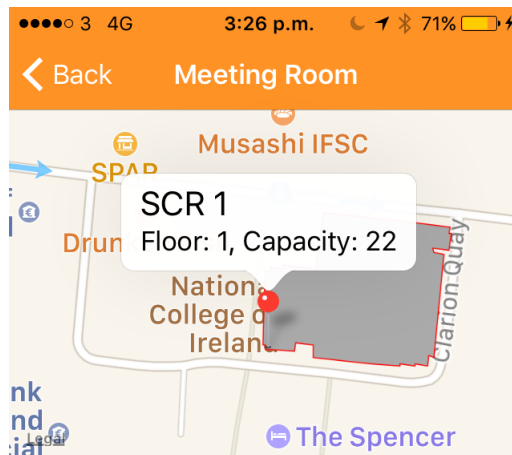
Future Idea: The application should leverage some sensors or beacons implemented in the building for effective indoor navigation as GPS may not provide accurate location indoors.
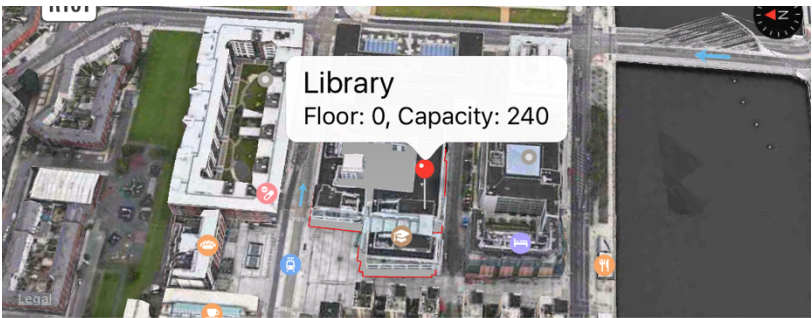
Meeting Room Information

Name:	Library
Parent Building:	National College of Ireland
Capacity:	240

Meeting Room Information

Name:	Library
Parent Building:	National College of Ireland
Capacity:	240
Email:	library@ncirl.ie
Coordinates:	53.348556 ,-6.243047
e:	+353892110549
Type:	Study Room
Floor Number:	0

Directions : Walk straight from the entrance o
your left clode to lifts

Meeting Room Information

Name:	Kelly Theatre
Parent Building:	National College of Ireland
Capacity:	125
Email:	kelly@ncirl.ie
Coordinates:	53.348633 ,-6.243084
e:	+353892110549
Type:	Theatre Room
Floor Number:	0

Directions : On entrance walk straight on you

4.1.4.3 Requirement Activation

The requirement will be activated when the user selects a meeting room from a list of rooms.

4.1.4.4 Technical Issues

N/A

4.1.4.5 Risks

May face some issues with current location of the user while indoors as GPS is not designed for indoors and this may affect the accuracy of the location slightly.

4.1.4.6 Dependencies on other requirements

N/A

4.1.5 Requirement 4: User Profile and User Manual

4.1.5.1 User Story

As a user, I should be able to see my profile and be able to access the help section of the app.

4.1.5.2 Description & Priority

The system shall provide user profile and manual.
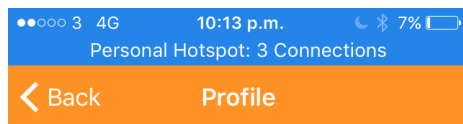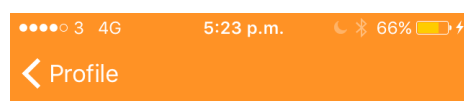
4.1.5.3 Requirement Activation

The requirement is activated on login or if the user navigates to user profile view.

4.1.5.4 Technical Issues

N/A

4.1.5.5 Risks

N/A

4.1.5.6 Dependencies on other requirements

The requirement has a dependency on the requirement: User registration and Login

4.1.6 Requirement 5: User Registration and Login

4.1.6.1 User Story

As a user, I shall be able to easily register within seconds to use the advanced features like posting reviews and uploading photos.

4.1.6.2 Description & Priority

The system shall provide a standard registration and login. The requirements like booking and reviewing require registration to allow users to use these features.
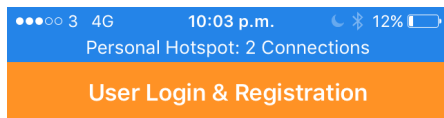
High Priority
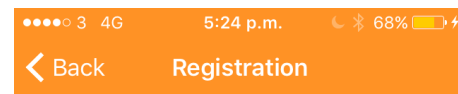
4.1.6.3 Requirement Activation

The requirement is activated when user wishes to book, review meeting room and upload photos of the meeting room.

4.1.6.4 Technical Issues

N/A

4.1.6.5 Risks

N/A

4.1.6.6 Dependencies on other requirements

N/A

4.1.7 Requirement 6: Booking System

4.1.7.1 User Story

As a user, I want to be able to book meeting rooms through my smartphone so as I can book meeting rooms easily from anywhere.

4.1.7.2 Description & Priority

Functionality for users to book meeting room once selected a meeting room. This feature shall have its own page and be easy to use i.e. drop down lists to choose data and time of booking and the user shall be able to see the booked meeting rooms.

High priority

4.1.7.3 Requirement Activation

This requirement will be activated if the client user wishes to book a meeting room.

4.1.7.4 Technical Issues

N/A

4.1.7.5 Risks

This requirement is very important as some of the requirements hereafter have a dependency on this one.

4.1.7.6 Dependencies on other requirements

This requirement has a dependency on requirement 1 - Easy to use GUI and Requirement 2 – List and Search

4.1.8 Requirement 7: Photo Gallery

4.1.8.1 User Story

As a user, I shall be able to see some photos of the meeting room showing the equipment available and size of room straight from the app.

4.1.8.2 Description & Priority

The system shall have a photo gallery with multiple photos of the room.

Medium Priority
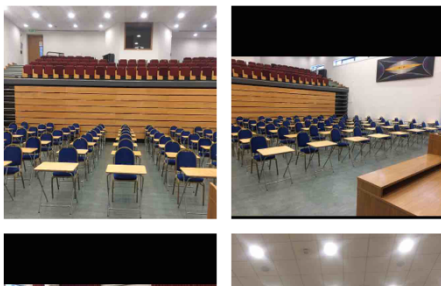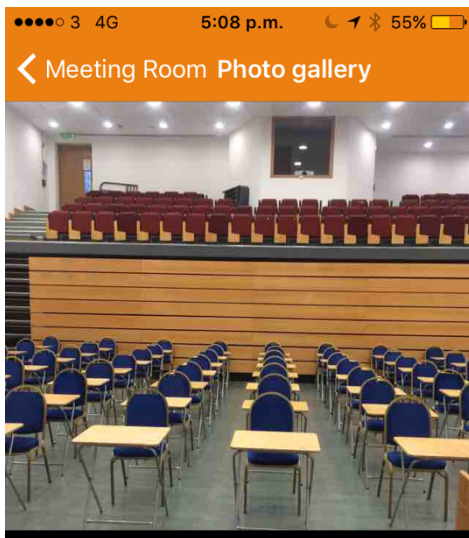
Navjot Singh Virk
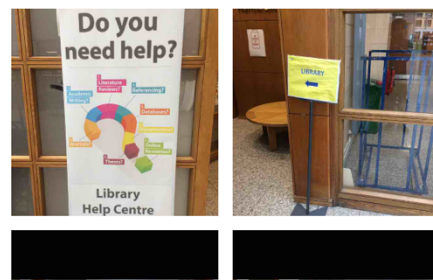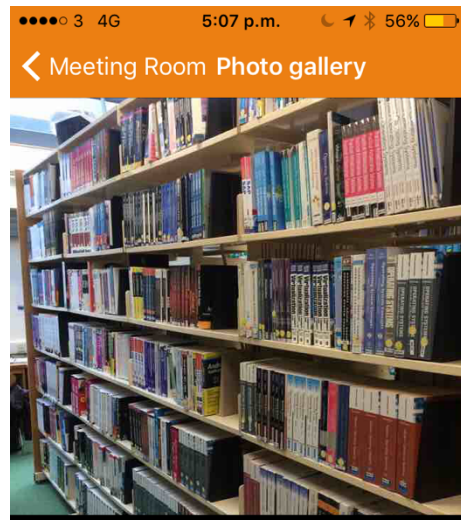
4.1.8.3 Requirement Activation

The requirement will be activated when a user selects a meeting room from the list.

4.1.8.4 Technical Issues

N/A

4.1.8.5 Risks

N/A

4.1.8.6 Dependencies on other requirements

N/A

4.1.9 Requirement 8: Photo Upload

4.1.9.1 User Story

As a user I shall be able to upload some picture from my phone gallery or camera if some pictures are missing in the Gallery.

4.1.9.2 Description & Priority

The system shall allow all registered users to upload images of the rooms to the photo gallery. And also allows users with admins and moderator accounts to delete the low quality images straight from the app.

High Priority
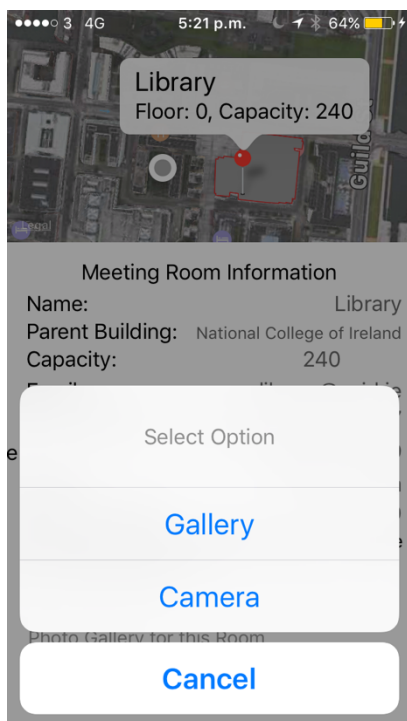
4.1.9.3 Requirement Activation

The requirement will be activated when user wishes to upload an image.

4.1.9.4 Technical Issues

N/A

4.1.9.5 Risks

Some users may try to spam or misuse this feature.

Navjot Singh Virk

4.1.9.6 Dependencies on other requirements

The requirement has a dependency on requirement 7: Photo Gallery

4.1.10 Requirement 9: Review

4.1.10.1 User Story

As a user, I would like to see some reviews available about the room and its equipment and also like the functionality to review it myself.

4.1.10.2 Description & Priority

The system shall allow the registered users to review meeting rooms based on different parameters like – functionality and quality of available equipment, if the room temperature was good and so on as user feedback will help improve the quality of the meeting rooms and the meeting rooms getting low ratings can be improved and get more attention from IT and Facilities department.

Medium Priority

4.1.10.3 Requirement Activation

The requirement will be activated when the user wishes to review meeting room.

4.1.10.4 Technical Issues

N/A

4.1.10.5 Risks

N/A

4.1.10.6 Dependencies on other requirements

N/A

4.1.11 Requirement 10: My Meetings

4.1.11.1 User Story

As a user, I would like to see all my meetings and the room that's booked, meeting start date and time.

4.1.11.2 Description & Priority

The system shall provide a list of meetings booked.
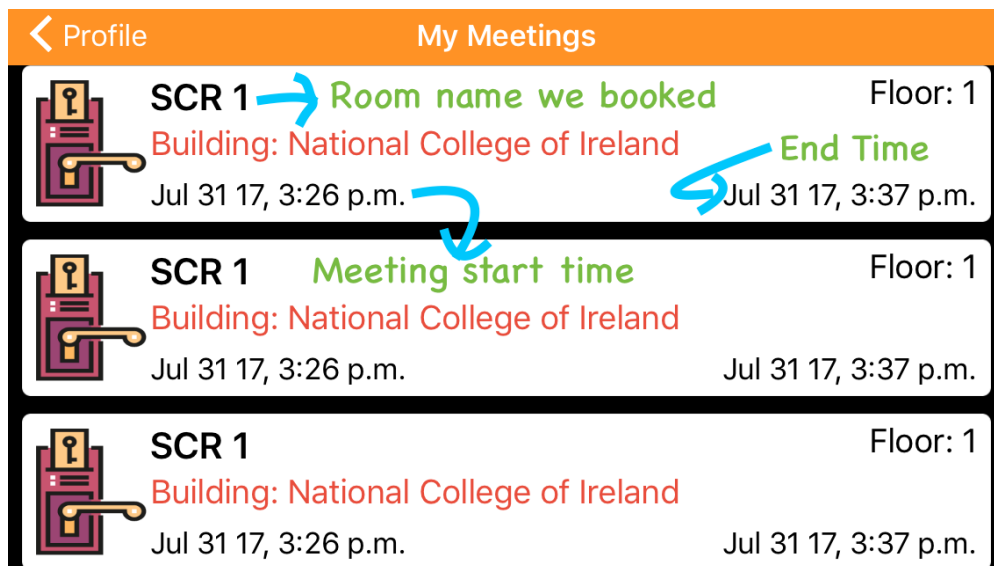
4.1.11.3 Requirement Activation

The requirement will be activated when the user has some meetings in their calendar.

Medium to Low Priority

4.1.11.4 Technical Issues

N/A

4.1.11.5 Risks

N/A

4.1.11.6 Dependencies on other requirements

The requirement has dependency on requirement 6: Booking System

4.1.12 Requirement 11: Call

4.1.12.1 User Story

As a user if I am in a meeting and some equipment is not working I should be able to contact the key contacts for the room that can fix the problem for me i.e. make a quick call from the app.

4.1.12.2 Description & Priority

The system shall provide contact information for key contacts and allow user to make a call.

Parent Building:	National College of Ireland
Capacity:	240
Email:	library@ncirl.ie
Coordinates:	53.348556 ,-6.243047
Phone:	+353892110549
Type:	Study Room
Floor Number:	0

Directions : Walk straight from the entrance on your left clode to lifts

Click the phone number to make a call

Book this Room

Photo Gallery for this Room

Upload photos of this Room

Review this Room
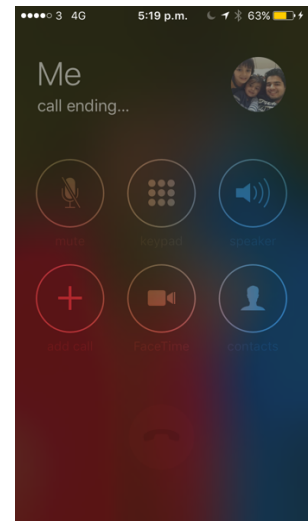
4.1.12.3 Requirement Activation

The requirement is activated if the users wishes to make a call through there phone.

4.1.12.4 Technical Issues

N/A

4.1.12.5 Risks

N/A

4.1.12.6 Dependencies on other requirements

Requires a sim card to make calls via the phone.

4.2 Non-Functional Requirements

4.2.1.1 Scalability requirement

4.2.1.1.1 User Story
N/A

4.2.1.1.2 Description and Priority
With regards to the intended number of users in big and medium cooperate organisation with thousands of employees and teachers and students in educational institutes and the projected load scenarios, the intention is for the system to be able to serve queries in thousands/ day (in large part during the 9-5 peak traffic hours) and the application shall also be scalable to use for any organisation with little to no changes to the app's code.
High Priority

4.2.1.2 Availability requirement

4.2.1.2.1 User Story
As a user, I want to be able to access the app at any time I wish.

4.2.1.2.2 Description and Priority
The app shall be accessible at any time of the day on any day of the year. *High priority*.

4.2.1.3 Physical requirement

4.2.1.3.1 User Story
As a user, I want to be able to access the app from anywhere.

4.2.1.3.2 Description and Priority
The app shall work on mobile devices. *High priority*.

4.2.1.4 Security and Privacy requirement

4.2.1.4.1 User Story
As a user, I want the app to be secure like my password must be safe.

4.2.1.4.2 Description and Priority
The app shall achieve security through encryption and SSL certificate when connecting to the server. *High priority*

4.2.1.5 Reliability requirement

4.2.1.5.1 User Story
As a user, I want to the app to be reliable.

4.2.1.5.2 Description and Priority

The app shall be reliable i.e. stable and consistent of what is expected out of it. *High priority*.

4.2.1.6 Maintainability requirement

4.2.1.6.1 User Story

N/A

4.2.1.6.2 Description and Priority

The app shall be easy to maintain on daily basics, and it should be easy to fix bugs, add new features, increase performance and easier for others to maintain the software.

High Priority

4.3 Data requirements

It has been mentioned before in the document that the application can be tailored for the use of any organization with a simple change of data provided. Which gives you an idea that data plays a major role in this project.

The database will store data/information about buildings and meeting rooms along with reviews, bookings and more.

JSON – Data is transferred through JSON using web api calls.

MySQL Database to store all the data regarding buildings, rooms, bookings, meetings

Web hosting/Cloud – store images for the use of gallery and upload feature.

4.3.1 Database Design

Database consists of 8 tables (can be seen the screenshot)
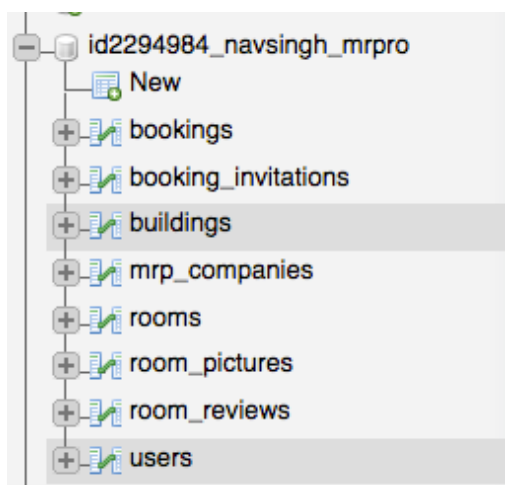
Rooms Table

#	Name	Type	Collation	Attri
1	roomId	int(120)		
2	buildingId	int(120)		
3	name	varchar(120)	utf8_unicode_ci	
4	floorNumber	int(120)		
5	coordinates	point		
6	capacity	int(120)		
7	type	varchar(120)	utf8_unicode_ci	
8	phone	varchar(120)	utf8_unicode_ci	
9	email	varchar(120)	utf8_unicode_ci	
10	directions	varchar(480)	utf8_unicode_ci	
11	location	varchar(1000)	utf8_unicode_ci	

Tree view:
- id2294984_navsingh_mrpro
  - New
  - bookings
  - booking_invitations
  - buildings
  - mrp_companies
  - rooms
  - room_pictures
  - room_reviews
  - users

Buildings Table

#	Name	Type	Collation	Attrib
1	id 🔑	int(11)		
2	name	text	utf8_general_ci	
3	city	text	utf8_general_ci	
4	country	text	utf8_general_ci	
5	floors	varchar(10)	utf8_general_ci	
6	address	text	utf8_general_ci	
7	phone	varchar(40)	utf8_general_ci	
8	email	varchar(30)	utf8_general_ci	
9	coordinates	polygon		
10	location	varchar(1000)	utf8_general_ci	

Bookings Table

#	Name	Type	Collation	Att
1	id 🔑	int(11)		
2	title	varchar(100)	utf8_unicode_ci	
3	roomID	int(11)		
4	userID	int(11)		
5	startDate	int(11)		
6	endDate	int(11)		

Booking Invitations Table

#	Name	Type	Collation	Att
1	id	int(11)		
2	bookingID	int(11)		
3	userID	int(11)		

Room Pictures Table

#	Name	Type	Collation	
1	id 🔑	int(11)		
2	roomID	int(11)		
3	url	text	utf8_unicod	

Room Reviews Table

#	Name	Type	Collation	A
1	id 🔑	int(11)		
2	roomID	int(11)		
3	review	text	utf8_unicode_ci	
4	rating	float		
5	userID	int(11)		
6	userName	varchar(100)	utf8_unicode_ci	
7	time	timestamp		o

Users Table

#	Name	Type	Collation	Att
1	id 🔑	int(11)		
2	name	varchar(30)	utf8_general_ci	
3	email	varchar(40)	utf8_general_ci	
4	phone	varchar(40)	utf8_general_ci	
5	picture	text	utf8_general_ci	
6	deviceToken	varchar(40)	utf8_general_ci	
7	password	varchar(40)	utf8_general_ci	

Company Table

	#	Name	Type	Collation	Attrib
☐	1	**companyId** 🔑	int(120)		
☐	2	**name**	varchar(240)	utf8_unicode_ci	
☐	3	**headquarters**	varchar(480)	utf8_unicode_ci	
☐	4	**website**	varchar(120)	utf8_unicode_ci	
☐	5	**email**	varchar(120)	utf8_unicode_ci	
☐	6	**ceo**	varchar(120)	utf8_unicode_ci	
☐	7	**industry**	varchar(120)	utf8_unicode_ci	
☐	8	**tagline**	varchar(120)	utf8_unicode_ci	

4.3.2   Database Table Structure

```
--
-- Table structure for table `bookings`
--

CREATE TABLE `bookings` (
  `id` int(11) NOT NULL,
  `title` varchar(100) COLLATE utf8_unicode_ci NOT NULL,
  `roomID` int(11) NOT NULL,
  `userID` int(11) NOT NULL,
  `startDate` int(11) NOT NULL,
  `endDate` int(11) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci;


--
-- Table structure for table `booking_invitations`
--

CREATE TABLE `booking_invitations` (
  `id` int(11) NOT NULL,
  `bookingID` int(11) NOT NULL,
  `userID` int(11) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci;


--
-- Table structure for table `buildings`
--

CREATE TABLE `buildings` (
  `id` int(11) NOT NULL,
  `name` text NOT NULL,
  `city` text NOT NULL,
  `country` text NOT NULL,
  `floors` varchar(10) NOT NULL,
  `address` text NOT NULL,
  `phone` varchar(40) NOT NULL,
  `email` varchar(30) NOT NULL,
  `coordinates` polygon NOT NULL,
  `location` varchar(1000) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;


--
-- Table structure for table `mrp_companies`
--
```

```sql
CREATE TABLE `mrp_companies` (
  `companyId` int(120) NOT NULL,
  `name` varchar(240) COLLATE utf8_unicode_ci NOT NULL,
  `headquarters` varchar(480) COLLATE utf8_unicode_ci NOT NULL,
  `website` varchar(120) COLLATE utf8_unicode_ci NOT NULL,
  `email` varchar(120) COLLATE utf8_unicode_ci NOT NULL,
  `ceo` varchar(120) COLLATE utf8_unicode_ci NOT NULL,
  `industry` varchar(120) COLLATE utf8_unicode_ci NOT NULL,
  `tagline` varchar(120) COLLATE utf8_unicode_ci NOT NULL
) ENGINE=MyISAM DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci;


--
-- Table structure for table `rooms`
--

CREATE TABLE `rooms` (
  `roomId` int(120) NOT NULL,
  `buildingId` int(120) NOT NULL,
  `name` varchar(120) COLLATE utf8_unicode_ci NOT NULL,
  `floorNumber` int(120) NOT NULL,
  `coordinates` point NOT NULL,
  `capacity` int(120) NOT NULL,
  `type` varchar(120) COLLATE utf8_unicode_ci NOT NULL,
  `phone` varchar(120) COLLATE utf8_unicode_ci NOT NULL,
  `email` varchar(120) COLLATE utf8_unicode_ci NOT NULL,
  `directions` varchar(480) COLLATE utf8_unicode_ci NOT NULL,
  `location` varchar(1000) COLLATE utf8_unicode_ci NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci;


--
-- Table structure for table `room_pictures`
--

CREATE TABLE `room_pictures` (
  `id` int(11) NOT NULL,
  `roomID` int(11) NOT NULL,
  `url` text COLLATE utf8_unicode_ci NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci;


--
-- Table structure for table `room_reviews`
--

CREATE TABLE `room_reviews` (
  `id` int(11) NOT NULL,
  `roomID` int(11) NOT NULL,
  `review` text COLLATE utf8_unicode_ci NOT NULL,
  `rating` float NOT NULL,
  `userID` int(11) NOT NULL,
  `userName` varchar(100) COLLATE utf8_unicode_ci NOT NULL,
  `time` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci;


--
-- Table structure for table `users`
--

CREATE TABLE `users` (
  `id` int(11) NOT NULL,
  `name` varchar(30) NOT NULL,
  `email` varchar(40) NOT NULL,
```

```
    `phone` varchar(40) NOT NULL,
    `picture` text NOT NULL,
    `deviceToken` varchar(40) NOT NULL,
    `password` varchar(40) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

4.3.3 Data Collection

The project included a significant amount of data collection the data that was collected was – Building coordinates in a polygon form, Room coordinates in point form both geo-spatial data types.

It took about 10 hours to collect the data required to populate the application and around 5 hours to prepare data in correct format in order to store in the database and about 2-3 hours to store that data in the database.

Example - Raw coordinates data (below) collected using https://codepen.io/jhawes/pen/ujdgK

new google.maps.LatLng(53.34888,-6.2435), new google.maps.LatLng(53.34875,-6.24351), new google.maps.LatLng(53.34853,-6.24352), new google.maps.LatLng(53.34853,-6.2435), new google.maps.LatLng(53.34851,-6.2435), new google.maps.LatLng(53.3485,-6.24327), new google.maps.LatLng(53.34854,-6.24326), new google.maps.LatLng(53.34853,-6.24312), new google.maps.LatLng(53.34847,-6.24312), new google.maps.LatLng(53.34844,-6.24266), new google.maps.LatLng(53.34843,-6.24243), new google.maps.LatLng(53.34844,-6.24243), new google.maps.LatLng(53.34843,-6.24231), new google.maps.LatLng(53.34844,-6.24231), new google.maps.LatLng(53.34843,-6.24221), new google.maps.LatLng(53.34848,-6.2422), new google.maps.LatLng(53.34847,-6.24216), new google.maps.LatLng(53.34896,-6.24208), new google.maps.LatLng(53.34897,-6.24216), new google.maps.LatLng(53.34899,-6.24217), new google.maps.LatLng(53.34898,-6.24204), new google.maps.LatLng(53.34906,-6.24202), new google.maps.LatLng(53.34906,-6.24204), new google.maps.LatLng(53.34909,-6.24203), new google.maps.LatLng(53.34912,-6.24243), new google.maps.LatLng(53.34916,-6.24304), new google.maps.LatLng(53.34889,-6.2431), new google.maps.LatLng(53.34891,-6.24336), new google.maps.LatLng(53.34889,-6.24337), new google.maps.LatLng(53.3489,-6.24346), new google.maps.LatLng(53.34888,-6.24346)

Example: Coordinates data after processing and ready to be inserted into database

Building 1- NCI Specifically

53.34888 -6.2435, 53.34875 -6.24351, 53.34853 -6.24352, 53.34853 -6.2435, 53.34851 -6.2435, 53.3485 -6.24327, 53.34854 -6.24326, 53.34853 -6.24312, 53.34883 -6.24311, 53.34889 -6.2431, 53.34891 -6.24336, 53.34889 -6.24337, 53.3489 -6.24346, 53.34888 -6.24346, 53.34888 -6.2435

Example: manually drawing polygon slowly and precise accuracy for NCI Building

Building 2 – NCI Area

53.34888 -6.2435, 53.34875 -6.24351, 53.34853 -6.24352, 53.34853 -6.2435, 53.34851 -6.2435, 53.3485 -6.24327, 53.34854 -6.24326, 53.34853 -6.24312, 53.34847 -6.24312, 53.34844 -6.24266, 53.34843 -6.24243, 53.34844 -6.24243, 53.34843 -6.24231, 53.34844 -6.24231, 53.34843 -6.24221, 53.34848 -6.2422, 53.34847 -6.24216, 53.34896 -6.24208, 53.34897 -6.24216, 53.34898 -6.24204, 53.34906 -6.24202, 53.34906 -6.24204, 53.34909 -6.24203, 53.34912 -6.24243, 53.34916 -6.24304, 53.34889 -6.2431, 53.34891 -6.24336, 53.34889 -6.24337, 53.3489 -6.24346, 53.34888 -6.24346, 53.34888 -6.2435

Building 3 – St james hospital

53.33973 -6.29482, 53.33976 -6.29497, 53.33974 -6.29512, 53.33997 -6.29577, 53.33983 -6.29634, 53.33986 -6.29638, 53.33981 -6.29663, 53.33978 -6.29663, 53.33962 -6.29734, 53.33965 -6.29738, 53.33959 -6.29768, 53.33927 -6.29746, 53.33925 -6.29745, 53.3392 -6.29769, 53.33929 -6.2978, 53.33919 -6.29807, 53.33919 -6.29814, 53.33916 -6.29823, 53.33924 -6.29833, 53.33922 -6.29837, 53.33948 -6.29864, 53.33951 -6.29859, 53.33948 -6.29846, 53.33944 -6.29847, 53.33955 -6.29829, 53.3397 -6.29846, 53.33971 -6.29853, 53.33978 -6.29856, 53.33973 -6.29869, 53.33975 -6.29874, 53.33972 -6.29883, 53.33959 -6.29871, 53.33954 -6.29885, 53.33964 -6.29896, 53.3396 -6.29909, 53.33953 -6.29903, 53.33936 -6.29944, 53.339 -6.29901, 53.33898 -6.29892, 53.33908 -6.29866, 53.3391 -6.29869, 53.33914 -6.29861, 53.33927 -6.29873, 53.33869 -6.29809, 53.33867 -6.29816, 53.33852 -6.29799, 53.33854 -6.29794, 53.33843 -6.29787, 53.33829 -6.29734, 53.33823 -6.29758, 53.33834 -6.2973, 53.33823 -6.29722, 53.33838 -6.29641, 53.33858 -6.2965, 53.33869 -6.29603, 53.33837 -6.29579, 53.33842 -6.29555, 53.33876 -6.29575, 53.33874 -6.29582, 53.339 -6.29601, 53.3393 -6.29463, 53.33918 -6.29455, 53.33932 -6.29932, 53.33946 -6.2941, 53.33964 -6.29424, 53.33959 -6.29445, 53.33941 -6.29435, 53.33939 -6.29443, 53.33942 -6.29447, 53.33939 -6.29464, 53.33964 -6.29479, 53.3397 -6.29448, 53.33962 -6.29442, 53.33967 -6.29425, 53.33934 -6.29429, 53.33979 -6.29414, 53.3397 -6.29406, 53.33976 -6.29384, 53.33996 -6.29398, 53.33972 -6.29492, 53.33973 -6.29482
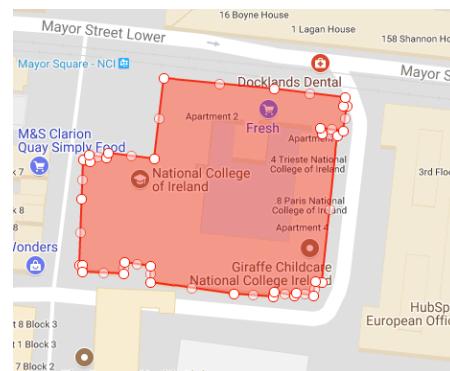
Example: drawing polygon to collect coordinates of ST. James Hospital building (it took around 35 mins to achieve this manually)

Building 4 – SAP
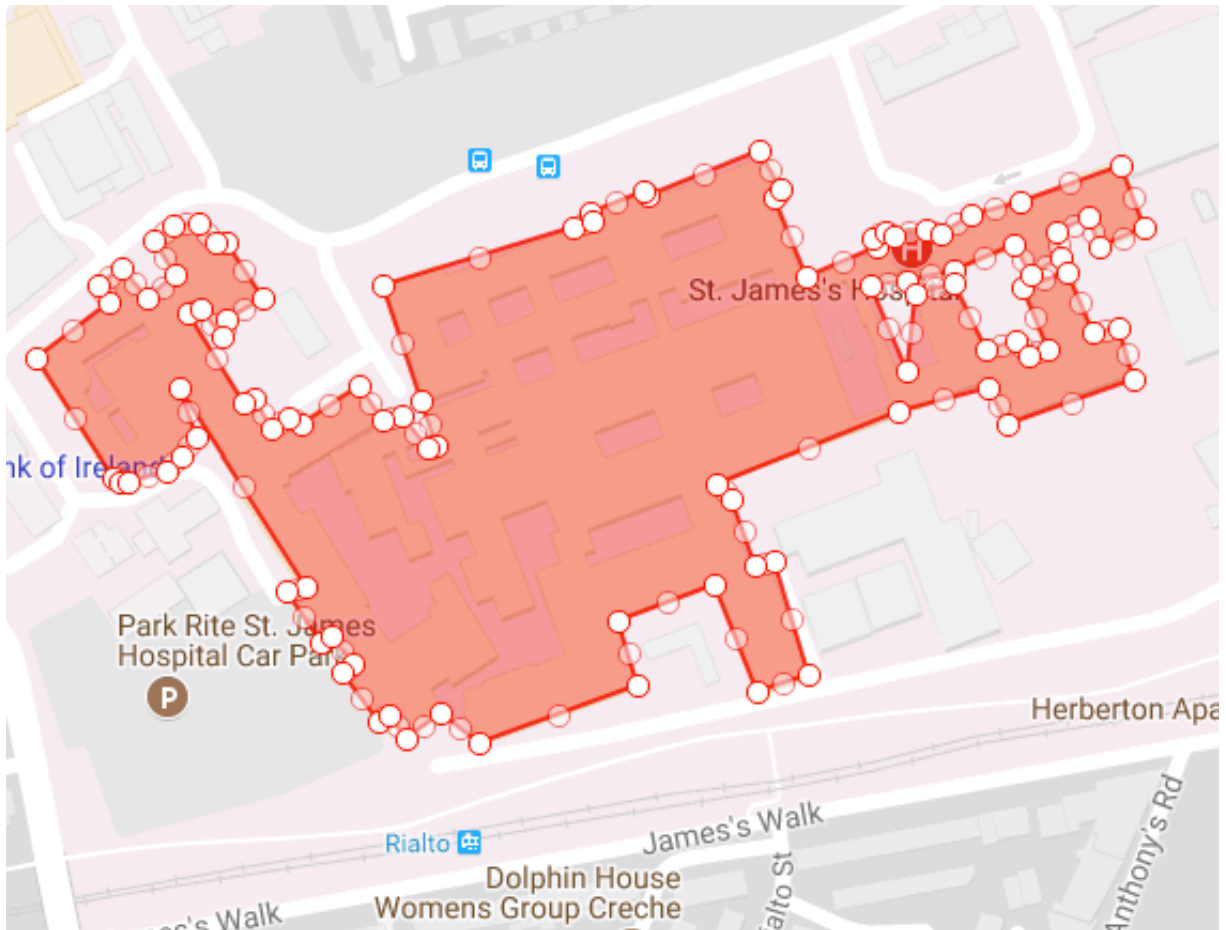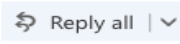
4.3.4 Images Collection

Included going to several rooms in NCI to collect the images for gallery feature of the application the process was easy but time consuming. It took about a day's time to get permissions and figure out images for all rooms the process wasn't difficult I did the image collection after developing the photo upload feature in the app which made it easy to collect the photos of the rooms with my iPhone and iPad and directly storing it to the cloud through the app.

Please Note: Images of the college building/rooms, library and atrium were only taken after getting permissions from the college authorities. Please see the email conversation below:

Joseph Egan
Today, 17:09
Navjot Singh Virk ⌄

> You forwarded this message on 31/07/2017 17:28
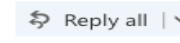
Hello Navjot,
Thank you for the email.
There is no issues with taking photos of the classrooms for use in your college project providing the classroom is empty at the time you take the photo.
Best of luck with the project.
Kind regards,

Joseph Egan,
Conference and Accommodation Administrator,
National College of Ireland,
Lower Mayor Street,
IFSC, Dublin 1.
Tel (Direct) +353 1 4498703
Web: http://www.ncirl.ie

...

Navjot Singh Virk
Hi, Joseph As, we talked few minutes ago. Can you please reply to this email for "permission to take ...   Today, 16:29

Joseph Egan
Hello Navjot, Thank you for the email. There is no issues with taking photos of the classrooms for us...   Today, 17:09

Navjot Singh Virk
Today, 16:29
Joseph Egan ⌄

Hi, Joseph

As, we talked few minutes ago. Can you please reply to this email for "permission to take photos".

This is the email, i sent to the marketing department as i mentioned when we were talking -

Hi Marketing Department

My name is Navjot and i am 4th year computing student, for my main project have developed an application called MRPro for finding and booking rooms, in the applications there is a feature called gallery, i was in the library and asked for permission to take photos of library and few other rooms to prepare a demo for the project report and presentation they asked me to contact the marketing department. I would like to ask permission if i can take pictures of the few rooms, atrium and library.

Thanks
Regards
Navjot

4.4 User requirements

- The user needs to have IOS device.
- The user will require an Apple Appstore account to download the app.

- User needs to have an active internet connection to use the application and will require registering, thus allowing user to login to the app and use all the advanced features of the application.

4.5 Environmental requirements

The server side of the system is implemented in PHP and SQL and requires MySQL Database.

The IOS App requires IOS 10+. And the application's navigation feature requires GPS enabled and a working internet connection.

4.6 Usability requirements

The core usability requirement is to allow for an incredibly user friendly experience on using the application. The application will be concise and easily navigable. The use of lists and search will allow the user to navigate the website with ease. The application will be functional to any user, regardless of experience.

4.6.1 Heuristic Evaluation

For inspection to identify problems, Heuristic Evaluation helps find usability problems by comparing with recognized usability principles. This Heuristic evaluation involves study of UI of app for violation of usability principles.

4.6.1.1 Recognized Usability Heuristics
- Visibility of System Status
- Match b/w System and Real World
- User Control and Freedom
- Consistency and Standards
- Error Prevention
- Recognition rather than recall
- Flexibility and Efficiency of Use
- Aesthetic and Minimalistic design
- Helps users recognise, diagnose, and recover form errors
- Help and documentation

4.6.1.2 Heuristic Evaluation Results Summary
For a listing of questions asked from the users for this heuristic evaluation refer to appendix.

Visibility of System Status –

MRPro app provides links to all important views through my profile page, and provide easy Navigation to My Profile (homepage) through navigation controller.

Match b/w System and Real World

In general, MRPro app uses simple language and app does not contain any set of terms that users may have problems understanding.

User Control and Freedom

User is able to easily navigate back and forth to the starting point. And find the relevant views with ease. Like clicking one of the list item brings up all the meeting rooms in the building.

Consistency and Standards – The app uses the same navigation style, color scheme consistently throughout the app.

Error Prevention – the app have 2 input forms for user data collection – login and signup page. The basic form of error prevention found on the app were easy to understand error messages example - "Email is missing – If the user forget to input email while logging in".

Recognition rather than recall – All the view in the have meaningful titles in the navigation bar, available options are highly visible like search bar.

Flexibility and Efficiency of Use – The app provides an easy to sign-up and login feature.

Aesthetic and Minimalistic design – The apps design is quite simple but very clean, its UI designed to be simple, clean and aesthetic but there is still some room for improvements.

Helps users recognize, diagnose, and recover form errors – all app doesn't get much inputs rather than login and signup and have standard error prevention and nothing to fancy as the app doesn't require more.

Help and documentation – In case the user is not able to find buildings or meeting room links a search bar is provided just below the navigation bar on most of the screens where necessary, the app also contains an info page with a short and easy user manual.

4.7 Design and Architecture

The application is designed and developed mainly through the use Xcode Interface builder and Swift. The design of the application is centered on creating an extremely user friendly GUI.

This is the architecture of the system it shows the client connects the web service through URLRequest/Alamofire and the service exposes the data from the database after authentication.
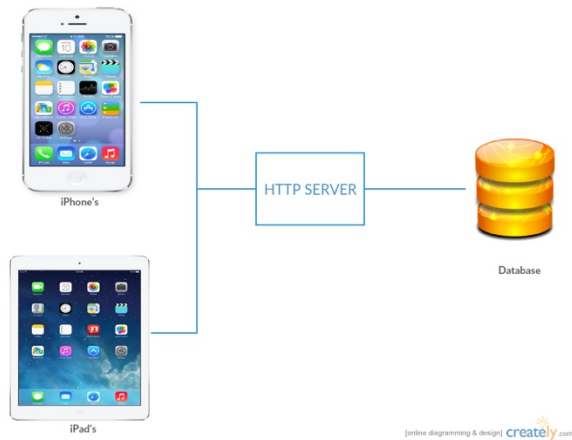
4.8 Implementation

4.8.1 Getting Started with Project development

4.8.1.1 Project Guide

This guide will help understand the requirement for opening and running this project- MeetingRoom Pro is an iOS (Swift) project, to use open the project code we require the following:

Apple Computer or Virtual device running macOS.
(I am using MacBook Pro, 6 GB Ram, 2.3 GHz Intel Core i5)

4.8.1.2 Installing Xcode

Xcode is a code editor. More details on it are included under Technologies section. Install Xcode on your Mac, using Appstore or this link: https://developer.apple.com/xcode/ide/

4.8.1.3 Opening the Project

Use MRPro.xcworkspace file to open the project in xcode.

4.8.1.4 Creating Project

Basic Steps that were involved in Project Creation of this project.

Step 1: File > New > Project
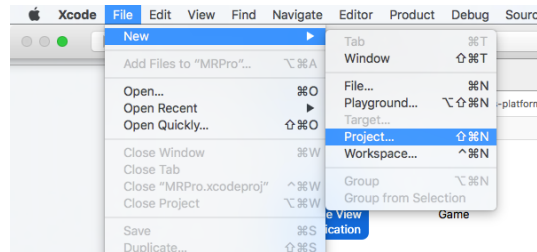
Step 2: Choose template – Single View Application > Next
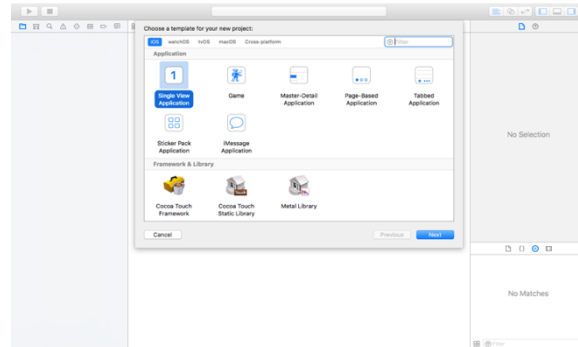
Step 3:

- **Enter Product Name** (aka. Project name) – MRPro (aka. MeetingRoom Pro)
- **Select Team** (Personal ios developer account – just signin with your apple id and xcode will take care of the rest)
- **Organisation Identifier**: ios.nsv (it could be anything (but unique) you like but it should usually make some sence and be short, simple and sweet like in this example- ios signifies its an ios app and nsv – Navjot singh virk)
- **Select Language**: Swift will be used for this project.
- **Devices**: Universal means the app will be supported on both iphones and ipads (you can also choose any one of them if you want to limit certain device type users).
- **Others:** Include UnitTests and UTests
- **Press > Next**
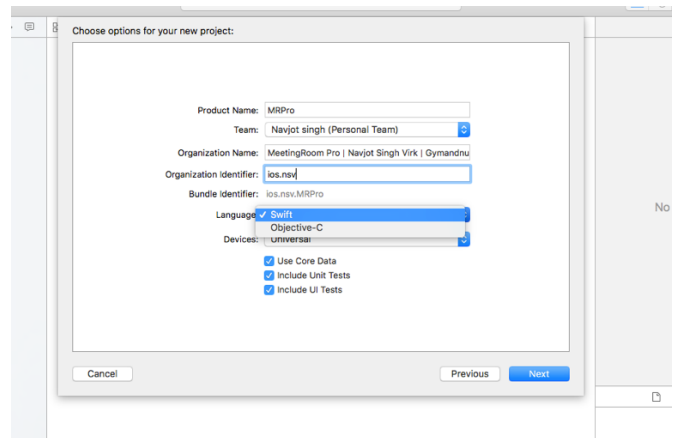
Step 4: Save the project on computer at location of your choice

Step 5: New Xcode project opens in a new window
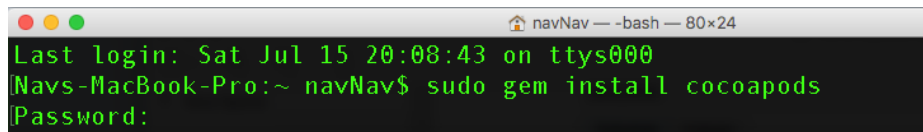
Step 6: At this stage project is ready and can be run on an emulator or a real ios device.

Step 7: Start making changes (write code)

4.8.1.5   Cocoapods

Installing Cocoapods (on my computer which will allow us to use different useful cocoapods available on the internet).

- Open terminal
- Run this command: **$ sudo gem install cocoapods** (cocoapods is implemented in ruby so its included as a gem on our computer, please refer to Technologies section of the report for more information regarding Cocoapods).
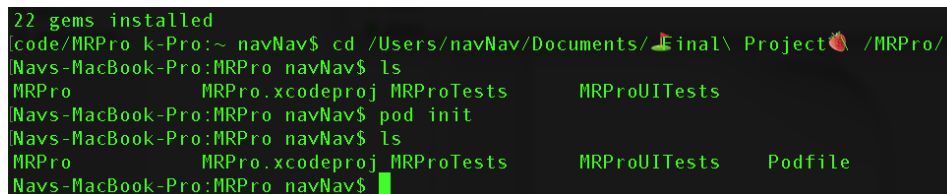
```
Last login: Sat Jul 15 20:08:43 on ttys000
[Navs-MacBook-Pro:~ navNav$ sudo gem install cocoapods
[Password:
```

4.8.1.6   Create Pod File

In order to use Cocoapods we need to create a Podfile for our project.
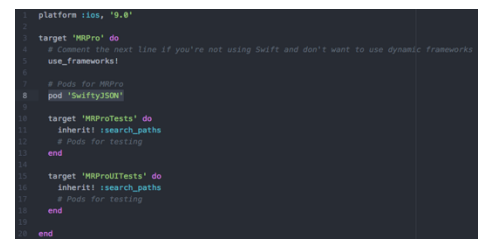
- Open Terminal
- Navigate to the project folder
- To create the **Podfile** run this command in the terminal: **$ pod init**

```
22 gems installed
[code/MRPro k-Pro:~ navNav$ cd /Users/navNav/Documents/🍓inal\ Project🍓 /MRPro/
Navs-MacBook-Pro:MRPro navNav$ ls
MRPro            MRPro.xcodeproj MRProTests        MRProUITests
Navs-MacBook-Pro:MRPro navNav$ pod init
Navs-MacBook-Pro:MRPro navNav$ ls
MRPro            MRPro.xcodeproj MRProTests        MRProUITests    Podfile
Navs-MacBook-Pro:MRPro navNav$
```

As we can see a Podfile was created when we ran the **'ls' command** second time to list the contents of the project folder.

- **Open Podfile** in an editor and write > **pod 'SwiftyJSON'** to add our first pod that will help in handling JSON better (it's a very popular library/pod among ios developer community when it comes to handling JSON in ios Swift).
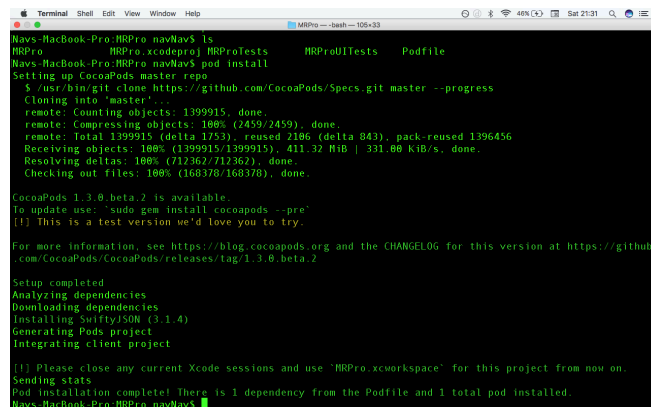
```
platform :ios, '9.0'

target 'MRPro' do
    # Comment the next line if you're not using Swift and don't want to use dynamic frameworks
    use_frameworks!

    # Pods for MRPro
    pod 'SwiftyJSON'

    target 'MRProTests' do
        inherit! :search_paths
        # Pods for testing
    end

    target 'MRProUITests' do
        inherit! :search_paths
        # Pods for testing
    end

end
```

- Save Podfile > Run this command in terminal - **$ pod install** (this can take time (upto 1 to 2 hours depending on your network speed if cocoapods are not used on your computer before in order to setup cocopods master repo).

```
Navs-MacBook-Pro:MRPro navNav$ ls
MRPro            MRPro.xcodeproj MRProTests        MRProUITests    Podfile
Navs-MacBook-Pro:MRPro navNav$ pod install
Setting up CocoaPods master repo
  $ /usr/bin/git clone https://github.com/CocoaPods/Specs.git master --progress
  Cloning into 'master'...
  remote: Counting objects: 1399915, done.
  remote: Compressing objects: 100% (2459/2459), done.
  remote: Total 1399915 (delta 1753), reused 2106 (delta 843), pack-reused 1396456
  Receiving objects: 100% (1399915/1399915), 411.32 MiB | 331.80 KiB/s, done.
  Resolving deltas: 100% (712362/712362), done.
  Checking out files: 100% (168378/168378), done.

CocoaPods 1.3.0.beta.2 is available.
To update use: `sudo gem install cocoapods --pre`
[!] This is a test version we'd love you to try.

For more information, see https://blog.cocoapods.org and the CHANGELOG for this version at https://github
.com/CocoaPods/CocoaPods/releases/tag/1.3.0.beta.2

Setup completed
Analyzing dependencies
Downloading dependencies
Installing SwiftyJSON (3.1.4)
Generating Pods project
Integrating client project

[!] Please close any current Xcode sessions and use `MRPro.xcworkspace` for this project from now on.
Sending stats
Pod installation complete! There is 1 dependency from the Podfile and 1 total pod installed.
Navs-MacBook-Pro:MRPro navNav$
```

- Close project in Xcode.

- Open the newly   generated – **MRPro.xcworkspace**. And now onwards every time to open/work on the project MRPro.xcworkspace should be used instead of MRPro.xcodeproj
- Now at this stage the pod we installed will be available to use (Usage: import SwiftyJSON).
- All the other pods will be installed the same way.

4.8.1.7   Challenges, Build Fails and their solutions

Build fail every time after adding a new pod due to ChameleonFramework

Details: Build failed with 31 errors

Clicking on one of the errors open the file in which errors exist. The problem here is target membership issue. As seen in the right hand pane of Xcode. Please refer to next screenshot.

Problem: **Target membership is checked**
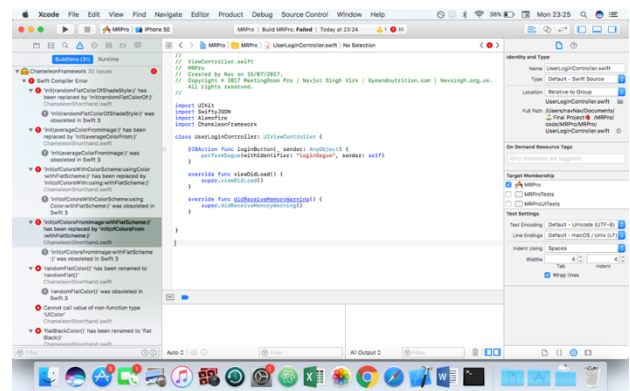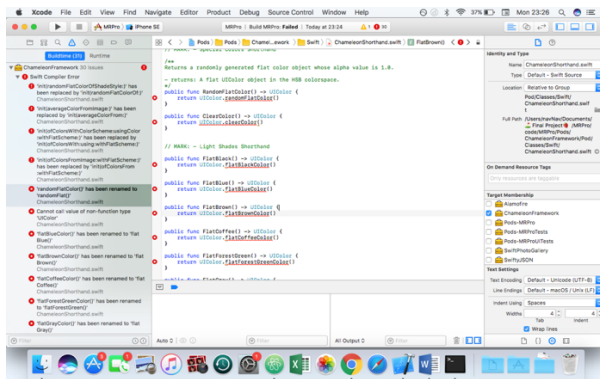
Solution: **Uncheck the target membership. And Clean Build**

Cmd + B
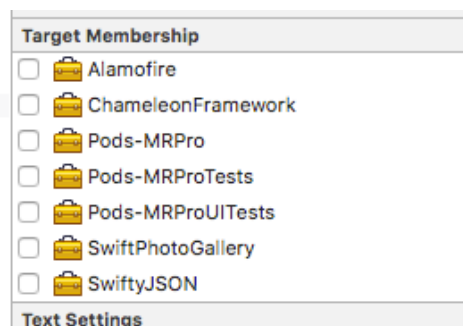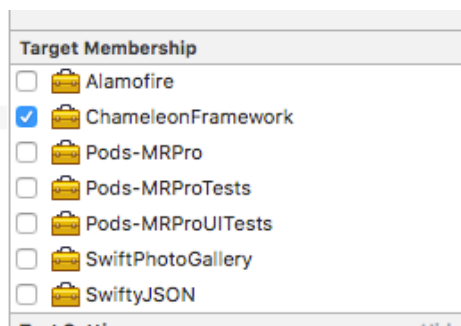
Results : The project is error free and build was successful.
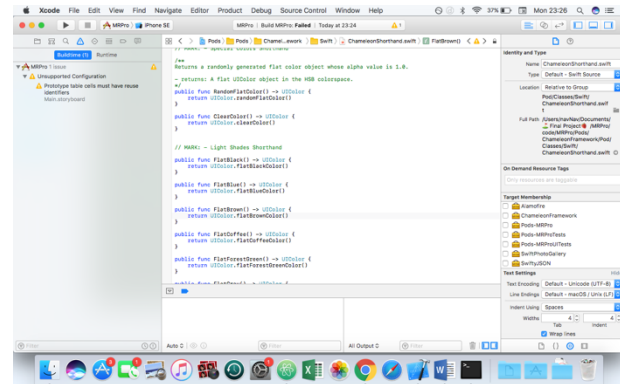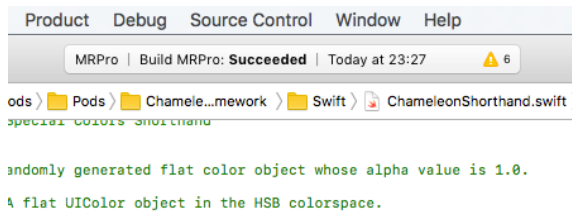
4.8.2   Client Side Implementation

Project Structure
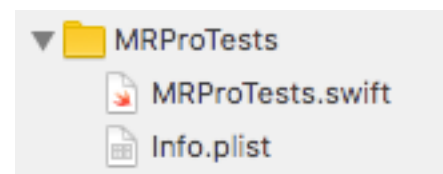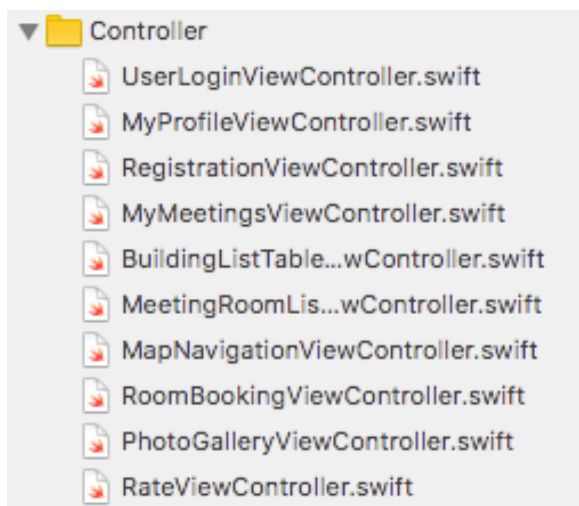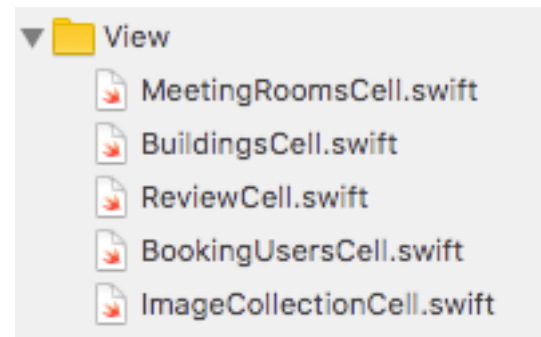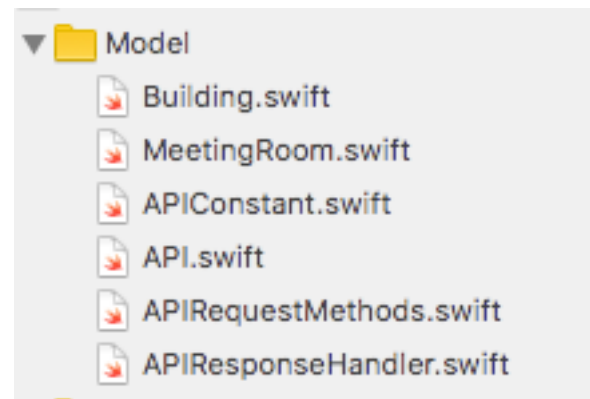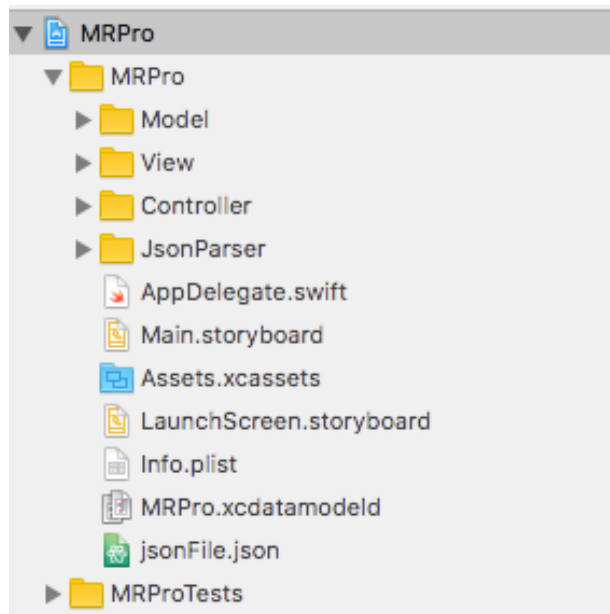
Example Implementation:

```swift
//
//  CustomJsonParser.swift
//  MRPro
//  Purpose: This file helps parse building and meeting room data from json
//  Created by Nav
//  Copyright © 2017 MeetingRoom Pro | Navjot Singh Virk | Gymandnutrition.com |
Navsingh.org.uk. All rights reserved.
//

import Foundation
import SwiftyJSON
import CoreLocation

/*
 *Helps parse building and meeting room data from json
 */
class CustomJsonParser {

    //this function parses the builging data -> usage in unit testing
    func parseBuildingJson(_ jsonData: Data) -> [Building] {

        let json = JSON(data: jsonData, options: .mutableContainers, error: nil)
        let buildingsJson = json["information"]
        var buildings = [Building]()
        var coordinates = [CLLocationCoordinate2D]()

        for index in 0..<buildingsJson.count {
            let buildingJson = buildingsJson[index]

            let meetingRooms = parseMeetingRoomsJson(jsonData, buildingIndex: index)

            for index in 0..<buildingJson["shape"]["coordinates"][0].count {

                let coordinateJson = buildingJson["shape"]["coordinates"][0][index]

                if let coordinate = parseCoordinate(coordinateJson) {
                    coordinates.append(coordinate)
                }
            }

            let building = Building(id: buildingJson["id"].stringValue,
                                    name: buildingJson["name"].stringValue,
                                    numberOfFloors:
buildingJson["numberOfFloors"].intValue,
                                    coordinates: coordinates,
                                    address: buildingJson["address"].stringValue,
                                    city: buildingJson["city"].stringValue,
                                    country: buildingJson["country"].stringValue,
rooms: meetingRooms)

            buildings.append(building)
            coordinates.removeAll()
        }
        return buildings
    }
```

```swift
    //parses the buildings data from the web api
    func parseServerBuildingJson(_ jsonData: Array<Any>) -> [Building] {

        var buildings = [Building]()

        for index in 0..<jsonData.count {
            let buildingJson = jsonData[index] as! [String:String]
            var coordinates = parseLocationString(jsonData: buildingJson)
            let building = Building(id: buildingJson["id"]!,
                                    name: buildingJson["name"]!,
                                    numberOfFloors: Int(buildingJson["floors"]!)!,
                                    coordinates: coordinates,
                                    address: buildingJson["address"]!,
                                    city: buildingJson["city"]!,
                                    country: buildingJson["country"]!, rooms: [])

            buildings.append(building)
            coordinates.removeAll()
        }
        return buildings
    }

    //parses the coordinates for building
    func parseLocationString(jsonData: [String : String]) ->
    [CLLocationCoordinate2D]{
        var response = [ CLLocationCoordinate2D]()
        if  jsonData["location"] != nil {

            let location : String! = jsonData["location"]
            let latLongsArr : [String] = location.components(separatedBy: ",")

            for latLongObj in latLongsArr {

                let coordinatesArr : [String] = latLongObj.components(separatedBy: "
")
                var lat : Double = 0.0
                var long : Double = 0.0
                for value in coordinatesArr {  // some strings have extra spaces
between latitude and longitude so we need to make a filter to trim white spaces

                    if value.characters.count > 3  {
                        if lat == 0.0 {
                            lat = Double(value)!
                        }
                        else if long == 0.0
                        {
                            long =  Double(value)!
                        }
                    }
                }
                let coordinates = CLLocationCoordinate2D.init(latitude: lat,
longitude: long)
                response.append(coordinates)
            }
        }

        return response;

    }

    //parse json meeting room data from web api
    func parseServerMeetingRooms(_ jsonData: Array<Any>) -> [MeetingRoom] {

        var meetingRooms = [MeetingRoom]()
```

```swift
        for index in 0..<jsonData.count {

            var meetingRoomJson = jsonData[index] as! [String:String]
            let latitude : Double = Double(meetingRoomJson["latitude"] as String!)!

            let longitude : Double = Double(meetingRoomJson["longitude"] as
String!)!

            let meetingRoom = MeetingRoom(id: Int(meetingRoomJson["roomId"]!)!,
                                          buildingId:
meetingRoomJson["buildingId"]!,
                                          name: meetingRoomJson["name"]!,
                                          floorNumber:
Int(meetingRoomJson["floorNumber"]!)!,
                                          coordinate:
CLLocationCoordinate2DMake(latitude, longitude),
                                          capacity:
Int(meetingRoomJson["capacity"]!)!,
                                          roomType: meetingRoomJson["type"]!,
                                          fullName: meetingRoomJson["name"]!,
                                          phone: meetingRoomJson["phone"]!,
                                          street: "",
                                          city: "", directions :
meetingRoomJson["directions"]!, email : meetingRoomJson["email"]!)

            meetingRooms.append(meetingRoom)
        }

        return meetingRooms
    }

    //parses json data for meeting rooms -> Usage in unit tests
    func parseMeetingRoomsJson(_ jsonData: Data, buildingIndex: Int) ->
[MeetingRoom] {

        var meetingRooms = [MeetingRoom]()
        let json = JSON(data: jsonData, options: .mutableContainers, error: nil)
        let meetingRoomsJson = json["information"][buildingIndex]["meetingRooms"]

        for index in 0..<meetingRoomsJson.count {

            var meetingRoomJson = meetingRoomsJson[index]

            let coordinateJson = meetingRoomsJson[index]["shape"]["coordinates"]
            let coordinate = parseCoordinate(coordinateJson)

            let meetingRoom = MeetingRoom(id: meetingRoomJson["id"].intValue,
                                          buildingId:
meetingRoomJson["buildingId"].stringValue,
                                          name: meetingRoomJson["name"].stringValue,
                                          floorNumber:
meetingRoomJson["floorNumber"].intValue,
                                          coordinate: coordinate,
                                          capacity:
meetingRoomJson["capacity"].intValue,
                                          roomType:
meetingRoomJson["roomType"].stringValue,
                                          fullName:
meetingRoomJson["fullName"].stringValue,
                                          phone:
meetingRoomJson["phone"].stringValue,
                                          street:
meetingRoomJson["street"].stringValue,
                                          city: meetingRoomJson["city"].stringValue,
```

```
                                                    directions:
meetingRoomJson["directions"].stringValue,
                                                    email:
meetingRoomJson["email"].stringValue)

            meetingRooms.append(meetingRoom)
        }

        return meetingRooms
    }

    //parses the coordinates -> usage in unit tests as well
    func parseCoordinate(_ coordinateJson: JSON) -> CLLocationCoordinate2D? {

        guard let latitudeValue = coordinateJson[0].rawValue as? NSNumber,
            let longitudeValue = coordinateJson[1].rawValue as? NSNumber else {
                print("Unable to parse coordinate values")
                return nil
        }

        let latitude = CLLocationDegrees(latitudeValue)
        let longitude = CLLocationDegrees(longitudeValue)

        let coordinate = CLLocationCoordinate2D(latitude: latitude, longitude:
longitude)

        return coordinate
    }
}
```

4.8.3 Server Side Implementation (Web Api)

Download CodeIgniter: https://codeigniter.com/download

Getting started: application/config/database.php (hostname, database name, username, password)

We will be working, under this directory - application/
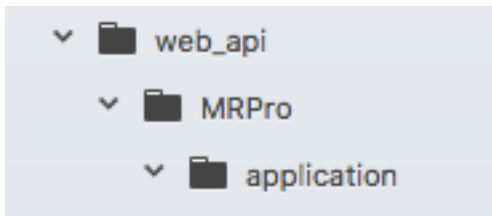
Since, CodeIgniter follows MVC, we will create our service using this pattern.

The folder's we would be looking and files I have implemented are as follows –
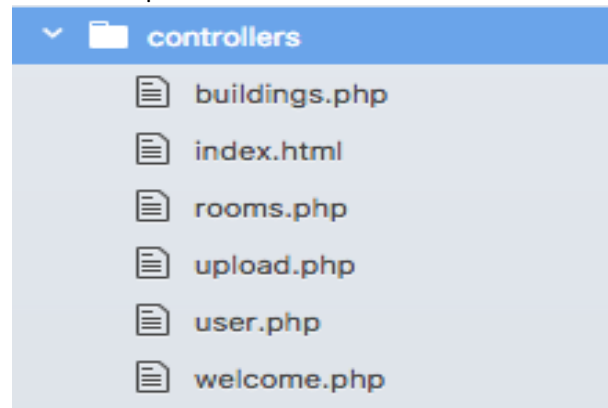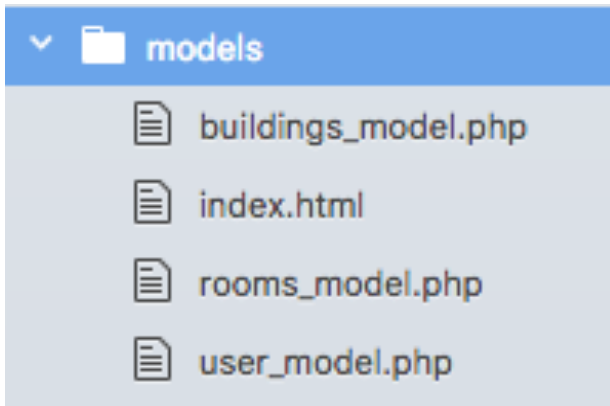
Below is rooms_model implementation and all models and controllers implementations can be found inside- web_api folder (in the submitted code)

```php
<?php

/**
*@author : Navjot Singh Virk
*File: rooms_model.php
*Project: MRPro
*Purpose: Handle room information
*Github: https://github.com/mrpro/
*Copyrights 2017-2018
**/

class Rooms_model extends CI_Model
{

  function __construct() {
    parent::__construct();
  }

  function allRooms($buildingID)
  {
    //creating an array
    $response = array();
    //query
    $query = $this->db->query("SELECT  rooms.roomId,rooms.buildingId,
rooms.floorNumber,rooms.name,rooms.capacity,rooms.type,rooms.phone,rooms.email,
rooms.directions , X(rooms.coordinates) as latitude ,Y(rooms.coordinates) as
longitude FROM rooms  where buildingId = '$buildingID'");


    if($query->num_rows()>0)
    {
      $response['status'] = 'success';
      $response['data']=$this->findRating($query->result_array());

    }
    else
    {
      $response['status'] = 'failure';
      $response['code'] = 'database failure';

    }
    return $response;
```

```php
    }

    // return average rating of user reviews
    function findRating($data)
    {
      $result = array();
      for ($i = 0; $i< count($data);$i++){

        $obj = $data[$i];
        $roomID = $data[$i]["roomId"];

        $obj["rating"] = "0";

        //query and get the average rating for the user reviews for meeting room given
by usres
        $query = $this->db->query("SELECT AVG(rating) as rating FROM
room_reviews  where roomID = $roomID");
        if($query->num_rows()>0)
        {
          $obj["rating"] = !$query->result_array()[0]["rating"]? "0.0" :$query-
>result_array()[0]["rating"];
        }
        else
        {
          $obj["rating"] = "0";

        }
        array_push($result, $obj);
      }
      //return the result
      return $result;

    }

    //Handles Meeting room booking and ensures if the room is already booked on a
certain time it can be booked
    //again by the same or other users
    function bookRooms($roomID,$title,$userID,$startDate, $endDate,$otherUsers){

      $result = $this->canBookRoom($startDate,$endDate,$roomID);
      if ($result == "FALSE")
      {
        $response['status'] = 'failure';
        $response['code'] = 'This room is already Booked on these time slots';
        return $response;
      }
      $add_data =
array("roomID"=>$roomID,"title"=>$title,"startDate"=>$startDate,"endDate"=>$endDate,
"userID"=>$userID);
      if ($this->db->insert('bookings', $add_data)) {
        $insert_id = $this->db->insert_id();
        for ($i = 0 ; $i < count($otherUsers); $i++)
        {
          $obj = $otherUsers[$i];

          $data = array("bookingID"=>$insert_id,"userID"=>$obj);
          $this->db->insert('booking_invitations', $data);

        }
        $response['status'] = 'success';
        $response['code']="Meeting Created Successfuly";
      }
      else
      {
```

```php
      $response['status'] = 'failure';
      $response['code'] = 'database failure';
    }
    return $response;

  }

  //also handles room booking
  function canBookRoom($startDate,$endDate,$roomID)
  {
    $result = "TRUE";
    $query = $this->db->query("SELECT * FROM bookings  where roomID = $roomID");
    if($query->num_rows()>0)
    {
      $bookings=$query->result_array();


      for ($i =0 ; $i < count($bookings);  $i++)
      {
        $bookStartTime = $bookings[$i]["startDate"];
        $bookEndTime = $bookings[$i]["endDate"];

        if ($startDate >= $bookStartTime  && $startDate <=$bookEndTime)
        {
          $result = "FALSE";
          return $result;
        }
        if ($endDate >= $bookStartTime  && $endDate <= $bookEndTime)
        {
          $result = "FALSE";
          return $result;

        }
      }
    }
    return $result;

  }

  //func to get all the bookings/My Meeting
  function getAllBookings($userID){
    $query = $this->db->query("SELECT bookings.*,buildings.name as buildingName,
buildings.id as buildingID ,rooms.name as roomName,rooms.floorNumber ,
rooms.type  from bookings join rooms on rooms.roomId = bookings.roomID join
buildings on buildings.id = rooms.buildingId where userID = $userID");

    if($query->num_rows()>0)
    {
      $response['status'] = 'success';
      $response['data']=$this->getParticipantsForMeetings($query->result_array());
    }
    else
    {
      $response['status'] = 'failure';
      $response['code'] = 'database failure';

    }
    return $response;
  }

  //func to get booked meetings invites/participants
  function getParticipantsForMeetings($array)
  {
    $result = array();
    for ($i=0; $i < count($array); $i++) {
```

```php
        $bookingID = $array[$i]["id"];

        $obj = $array[$i];

        $query = $this->db->query("SELECT bookingID, users.* from
booking_invitations  join users on users.id = booking_invitations.userID where
bookingID = $bookingID");
        if($query->num_rows()>0)
        {
          $obj["users"] = $query->result_array();

        }
        else
        {
          $obj["users"] = array();

        }
        array_push($result, $obj);
      }

      return $result;
    }

    //func for adding new pics to photo gallery
    function updateRoomPics($roomID,$pictureurl){
      $add_data = array("roomID"=>$roomID,"url"=>$pictureurl);

      if ($this->db->insert('room_pictures', $add_data))
      {
        $response['status'] = 'success';
        $response['code'] = 'image uploaded Successfuly';

      }
      else
      {
        $response['status'] = 'failure';
        $response['code'] = 'database failure';

      }

      return $response;
    }

    // func to get all the photos of a room to show in the photo gallery of the app
    function geteRoomPics($roomID){
      $response = array();
      $query = $this->db->query("SELECT * FROM room_pictures  where roomID =
$roomID");
      if($query->num_rows()>0)
      {
        $response['status'] = 'success';
        $response['data']=$query->result_array();

      }
      else
      {
        $response['status'] = 'failure';
        $response['code'] = 'database failure';

      }
      return $response;
    }

    //allows the usre to post a review/rating for a meeting room
    function postReview($roomID,$review,$rating,$userID,$userName){
```

```php
    $response = array();
    $add_data =
array("roomID"=>$roomID,"review"=>$review,"rating"=>$rating,"userID"=>$userID,"userN
ame"=>$userName);

    if ($this->db->insert('room_reviews', $add_data))
    {
      $response['status'] = 'success';
      $response['code'] = 'added Successfuly';
      $roomQuery = $this->db->query("SELECT * FROM room_reviews  where roomID =
$roomID");
      if($roomQuery->num_rows()>0)
      {
        $response['data']=$roomQuery->result_array();

      }
      else
      {
        $response['data']= array();
      }

    }
    else
    {
      $response['status'] = 'failure';
      $response['code'] = 'database failure';

    }

    return $response;

  }

  //function to return all the user reviews
  function getReviews($roomID){
    $response = array();

    $query = $this->db->query("SELECT * FROM room_reviews  where roomID = $roomID");
    if($query->num_rows()>0)
    {
      $response['status'] = 'success';
      $response['data']=$query->result_array();

    }
    else
    {
      $response['status'] = 'failure';
      $response['code'] = 'database failure';

    }
    return $response;

  }
}
```

4.8.4   Web API Request/Calls (API giving JSON data to the client)

```swift
//Webservice URL
let API_URL = "https://mrpro.000webhostapp.com/MRProApp/MRPro/MRPro/index.php/"

//api request end-points
let API_DeviceRegistration = "registerDevice.php"
let API_Register = "user/postUser"
let API_Login = "user/signIn"
```

```
let API_BuildingList = "buildings/allBuildings"
let API_RoomsList = "rooms/allRooms"
let API_BuildingSearch = "buildings/buildings_search"
let API_GetAllUsers = "user/getUsers"
let API_BookRoom = "rooms/bookRoom"
let API_GetPhotos = "rooms/getRoomsImage"
let API_UploadImage = "rooms/room_image"
let API_PostReview = "rooms/postReview"
let API_MyMeetingRooms = "rooms/getAllBookings"
let API_RoomRatings = "rooms/getRoomReviews"
```

To use the api you need to be logged in to retrieve data from api.

Login with: Username – nav@nav.com & Password - password

URL: **https://mrpro.000webhostapp.com/MRProApp/MRPro/MRPro/index.php/user/signIn**

```
"API Name :::::::::::::::::: %@" https://mrpro.000webhostapp.com/MRProApp/MRPro/MRPro/index.php/user/
signIn

$ curl -v \
    -X POST \
    -H "Content-Type: application/x-www-form-urlencoded; charset=utf-8" \
    -H "Accept-Language: en-IE;q=1.0, hi-IE;q=0.9" \
    -H "User-Agent: MRPro/1.0 (ios.nsv.MRPro; build:1; iOS 10.2.1) Alamofire/4.5.0" \
    -H "Accept-Encoding: gzip;q=1.0, compress;q=0.5" \
    -d "email=nav%40nav.com&password=password" \
    "https://mrpro.000webhostapp.com/MRProApp/MRPro/MRPro/index.php/user/signIn"


{
    data =      {
        deviceToken = SIMULATOR;
        email = "nav@nav.com";
        id = 6;
        name = Nav;
        password = password;
        phone = "+353892110548";
        picture = "";
    };
    status = success;
}
{
  "status" : "success",
  "data" : {
    "phone" : "+353892110548",
    "password" : "password",
    "id" : "6",
    "picture" : "",
    "email" : "nav@nav.com",
    "name" : "Nav",
    "deviceToken" : "SIMULATOR"
  }
}
```

URL:
https://mrpro.000webhostapp.com/MRProApp/MRPro/MRPro/index.php/buildings/allBuildings

```
$ curl -v \
    -X POST \
    -H "Content-Type: application/x-www-form-urlencoded; charset=utf-8" \
    -H "Accept-Language: en-IE;q=1.0, hi-IE;q=0.9" \
    -H "User-Agent: MRPro/1.0 (ios.nsv.MRPro; build:1; iOS 10.2.1) Alamofire/4.5.0" \
    -H "Accept-Encoding: gzip;q=1.0, compress;q=0.5" \
    -d "" \
    "https://mrpro.000webhostapp.com/MRProApp/MRPro/MRPro/index.php/buildings/allBuildings"


{
    data =      (
                {
            address = "Mayor Street, IFSC, Dublin 1, D01 Y300";
            city = Dublin;
            country = Ireland;
            email = "virksaabnavjot@gmail.com";
            floors = 4;
            id = 2;
            location = "53.34888 -6.2435, 53.34875 -6.24351, 53.34853 -6.24352, 53.34853 -6.2435,
53.34851 -6.2435, 53.3485 -6.24327, 53.34854 -6.24326, 53.34853 -6.24312, 53.34847 -6.24312, 53.34844
-6.24266, 53.34843 -6.24243, 53.34844 -6.24243, 53.34843 -6.24231, 53.34844 -6.24231, 53.34843
-6.24221, 53.34848 -6.2422, 53.34847 -6.24216, 53.34896 -6.24208, 53.34897 -6.24216, 53.34899 -6.24217,
53.34898 -6.24204, 53.34906 -6.24202, 53.34906 -6.24204, 53.34909 -6.24203, 53.34912 -6.24243, 53.34916
-6.24304, 53.34889 -6.2431, 53.34891 -6.24336, 53.34889 -6.24337, 53.3489 -6.24346, 53.34888 -6.24346,
53.34888 -6.2435";
            name = "National College of Ireland";
            phone = "+353892110549";
        },
                {
            address = "Mayor Street, IFSC, Dublin 1, D01 Y300";
            city = Dublin;
            country = Ireland;
            email = "virksaabnavjot@gmail.com";
            floors = 3;
            id = 3;
            location = "53.34888 -6.2435, 53.34875 -6.24351, 53.34853 -6.24352, 53.34853 -6.2435,
53.34851 -6.2435, 53.3485 -6.24327, 53.34854 -6.24326, 53.34853 -6.24312, 53.34883 -6.24311, 53.34889
-6.2431, 53.34891 -6.24336, 53.34889 -6.24337, 53.3489 -6.24346, 53.34888 -6.24346, 53.34888 -6.2435";
            name = NCI;
```

Quick Links:

URL: https://mrpro.000webhostapp.com/MRProApp/MRPro/MRPro/index.php/rooms/allRooms

URL: https://mrpro.000webhostapp.com/MRProApp/MRPro/MRPro/index.php/user/getUsers

URL: https://mrpro.000webhostapp.com/MRProApp/MRPro/MRPro/index.php/rooms/bookRoom

URL:
https://mrpro.000webhostapp.com/MRProApp/MRPro/MRPro/index.php/rooms/getRoomsImage

4.9   Graphical User Interface (GUI) Layout

The app will run on IOS devices that use touch screen functionality. Displaying an easy to use interface.

4.9.1 GUI Implementation
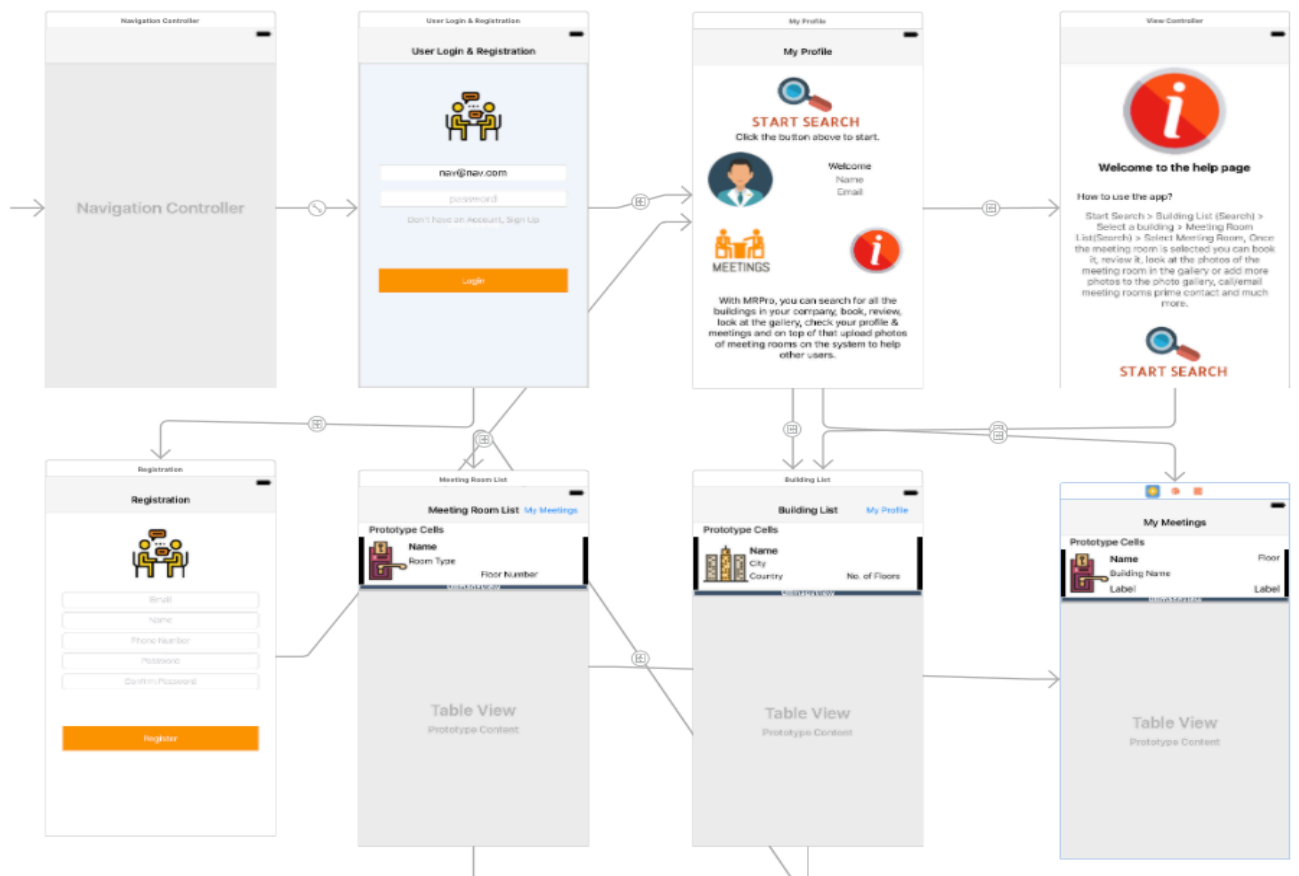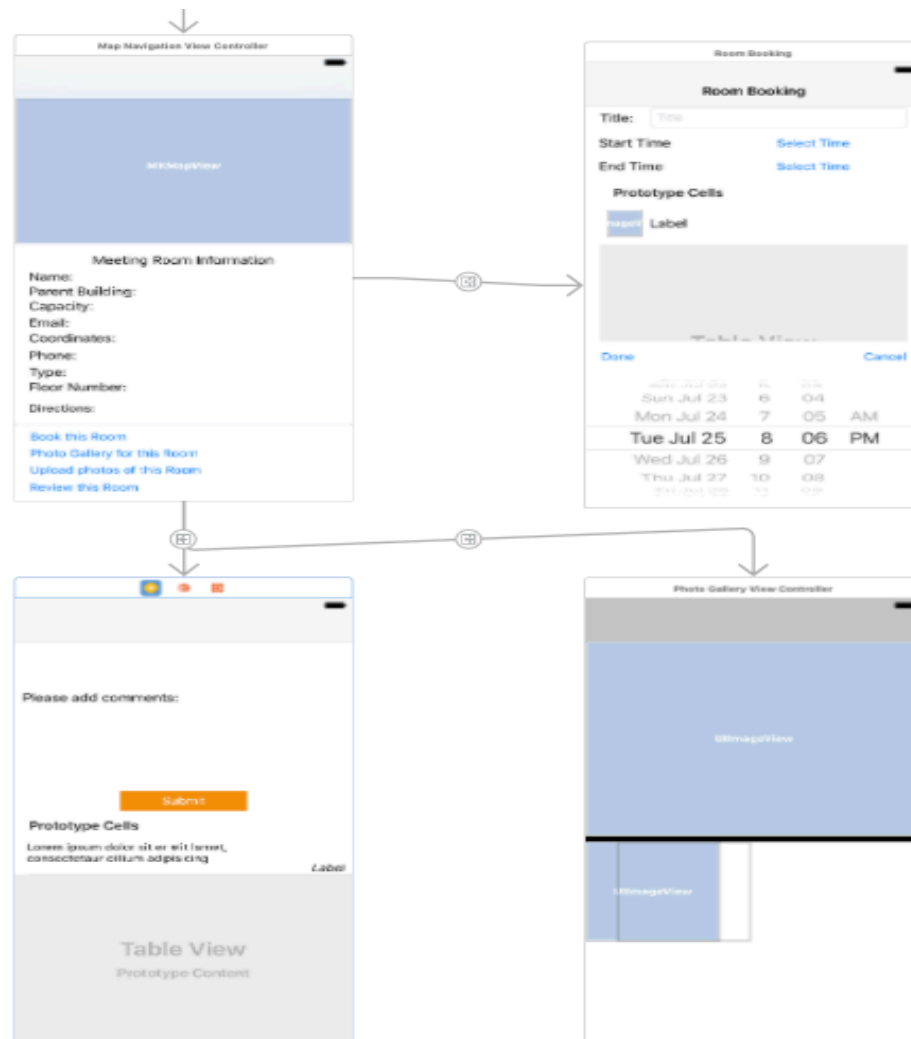
4.10 Testing

Testing is a very important part of software development. Continuous testing provides several benefits including less bugs and high quality production grade product/code.

For this project, I have focused a lot and spend a good amount to time to ensure sufficient testing is carried out throughout the development of the project.

I have incorporated End User Testing, Unit testing for this project.

4.11 Unit Testing

Unit testing helps to ensure functionality of the all core components of is correct. For this project, I used XCTest to create unit tests and verify the results.

```
//
//  MRProTests.swift
//  MRProTests
```

```swift
//  Purpose: This class is used for unit testting of important functionality of the
app
//  Created by Nav
//  Copyright © 2017 MeetingRoom Pro | Navjot Singh Virk | Gymandnutrition.com |
Navsingh.org.uk. All rights reserved.
//

import XCTest
import CoreLocation
@testable import MRPro

class MRProTests: XCTestCase {

    override func setUp() {
        super.setUp()
    }

    override func tearDown() {
        super.tearDown()
    }


    //function to return data from the file
    func jsonFromFile() -> Data {
        //setting the path to our local json file
        let path = Bundle.main.path(forResource: "jsonFile", ofType: "json")
        //get the json data from the file
        let jsonData = try? Data(contentsOf: URL(fileURLWithPath: path!))

        return jsonData!
    }


    /*
     function to ensure the json data for building object can be parsed
     sucessfully or not
     */

    func testCanParseBuidingInfoFromJson() {

        //Custom class for parsing building and meeting room json
        let jsonParser = CustomJsonParser() //creating instance of parser class
        let jsonData = jsonFromFile() //getting the json form file
        let buildings = jsonParser.parseBuildingJson(jsonData)
        let extractedBuilding = buildings[0]// getting the building at index zero

        //saving all the building information in variables
        let buildingName = extractedBuilding.name
        let buildingId = extractedBuilding.id
        let city = extractedBuilding.city
        let country = extractedBuilding.country
        let numberOfFloors = extractedBuilding.numberOfFloors

        //testing with the data from file and check if tests fail or pass
        XCTAssertEqual(buildingName, "NCI")
        XCTAssertEqual(buildingId, "DUB01")
        XCTAssertEqual(city, "Dublin")
        XCTAssertEqual(country, "Ireland")
        XCTAssertEqual(numberOfFloors, 3)
    }

    /*
     Function to test if building coordinates from json can be parsed
     */
```

```swift
    func testCanParseBuidingCoordinatesFromJson() {
        //creating instance of parser class
        let jsonParser = CustomJsonParser()
        //calling the json from file function that returns json data
        let jsonData = jsonFromFile()

        //creating an array of coordinates
        let buildingCoordinates = [
            CLLocationCoordinate2D(latitude: 53.294974000000003, longitude: -
6.4266310000000004),
            CLLocationCoordinate2D(latitude: 53.294846999999997, longitude: -
6.4264190000000001),
            CLLocationCoordinate2D(latitude: 53.294288999999999, longitude: -
6.4268879999999999),
            CLLocationCoordinate2D(latitude: 53.294325999999998, longitude: -
6.4271940000000001),
            CLLocationCoordinate2D(latitude: 53.294974000000003, longitude: -
6.4266310000000004)
        ]

        //creating a variable and seeting it equal to all the building data
        let buildings = jsonParser.parseBuildingJson(jsonData)
        //get the building at index zero
        let extractedBuildingShape = buildings[0]

        //for loop - that iterates through the building coordinates and
        //check if the longitude and latitude at each index matches or not
        for i in 0..<buildingCoordinates.count {

            //long and lat from the created array of coordinates -
buildingCoordinates
            let longitude = buildingCoordinates[i].longitude
            let latitude = buildingCoordinates[i].latitude

            //long and lat from the building from json file
            let extractedLatitude = extractedBuildingShape.coordinates[i].latitude
            let extractedLongitude = extractedBuildingShape.coordinates[i].longitude

            //testing
            XCTAssertEqual(longitude, extractedLongitude)
            XCTAssertEqual(latitude, extractedLatitude)
        }
    }


    /*
     Function to test if room information from json can be parsed
     */

    func testCanParseMeetingRoomInfoFromJson() {
        let jsonParser = CustomJsonParser()
        let jsonData = jsonFromFile()
        //getting all the meeting rooms available in the building
        let meetingRooms = jsonParser.parseMeetingRoomsJson(jsonData, buildingIndex:
1)
        //getting the first meeting room
        let meetingRoom = meetingRooms.first!

        //saving the room information in variables
        let roomName = meetingRoom.name
        let roomId = meetingRoom.id
        let floorNumber = meetingRoom.floorNumber
        let capacity = meetingRoom.capacity
        let buildingId = meetingRoom.buildingId
```

```
        //test
        XCTAssertEqual(roomName, "SCR 3")
        XCTAssertEqual(roomId, 6)
        XCTAssertEqual(floorNumber, 3)
        XCTAssertEqual(capacity, 3)
        XCTAssertEqual(buildingId, "National College of Ireland")
    }


    /*
     Function to test if meeting room coordinates from json can be parsed
     */

    func testCanParseMeetingRoomCoodinatesFromJson() {
        let jsonParser = CustomJsonParser()
        let jsonData = jsonFromFile()

        //creating a set of coordinate using CLLocationCoordinate2D
        let coordinate = CLLocationCoordinate2D(latitude: 43.294846, longitude: -
7.426421)

        //getting the meeitng rooms
        let meetingRooms = jsonParser.parseMeetingRoomsJson(jsonData, buildingIndex:
1)

        //getting the lat and long of meeting room at index zero
        let latitude = meetingRooms[0].coordinate?.latitude
        let longitude = meetingRooms[0].coordinate?.longitude

        //test
        XCTAssertEqual(latitude, coordinate.latitude)
        XCTAssertEqual(longitude, coordinate.longitude)
    }
}
```

4.12 User/Customer testing

For user testing, the users were introduced & explained the idea of the app and were given chance to use the app and give their feedback through the survey. (Users were given a chance to use and navigate around the app without any help to see how easily new users can pick up on the app).

4.12.1 Survey

Survey Questions

Section 1

- How easy is to navigate around the app without any help?
- How attractive are the features of the app?
- What do you like about the app?
- Would you like to use the app in future?
- Do you like the user interface of the app?
- How easy is to learn to use the app ?
- Did you liked the color schemes of the app UI?

Section 2

- Which feature did you like the most and Why?
- Do you like the app logo?
- What other features would you like to see in the app?
- Part 1: How excited are you to test the app ?
- Part 2: Do you think this is a useful application?

Section 3

- How many times have you felt lost in a building?
- How often do you need to find a meeting room in large co-operate buildings?
- How often do you need to find room in hotels?
- What do you do (Employee/student etc)?

Section 4

- How easy it was to find rooms with this application in the building?
- Did you like the map view?
- Was the app fast to use?
- Was the app slow and leggy to use?

Section 5

- Did the app crash while you were using it?
- How many times did the app crash?
- Is the GUI/UI of the app responsive and clean?

Section 6

- How many stars would you give to user experience?
- How many stars would you give to app reliability?
- How many stars would you give to apps performance?
- How many stars would you give to app features?
- How many stars would you give to app design?
- Other thoughts, improvement or comments

The results and full survey can be found here:
https://docs.google.com/forms/d/e/1FAIpQLSf6OT1ni3DJoIWozhp1RxZ6kJi0L0zdAWo9H75-8ZvdJquKvA/viewform

Survey results: 10 people took the survey and inputted their response people included colleges at work, friends, family members and some gym buddies that work in multinational companies. The response was great, below are some of the responses (full results – please reference appendix).

How easy is to navigate around the app without any help?

8 responses

- ● Very Easy
- ● Easy
- ● Fairly Difficult
- ● Easy, but can be a lot better
- ● Difficult
- ● Extremely Difficult
- ● nice

12.5% 12.5% 12.5% 62.5%
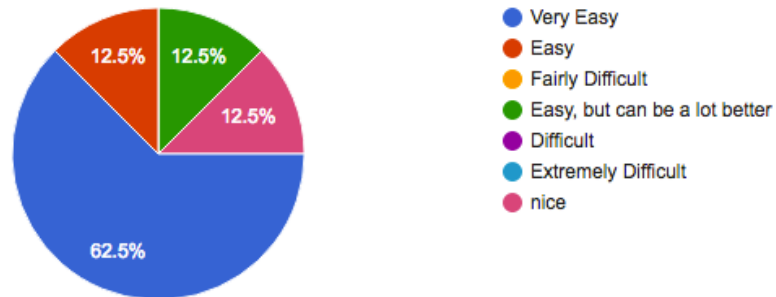
How attractive are the features of the app?

9 responses

- ● Very Attractive
- ● Good
- ● Fair
- ● Bad
- ● I don't know

22.2% 77.8%
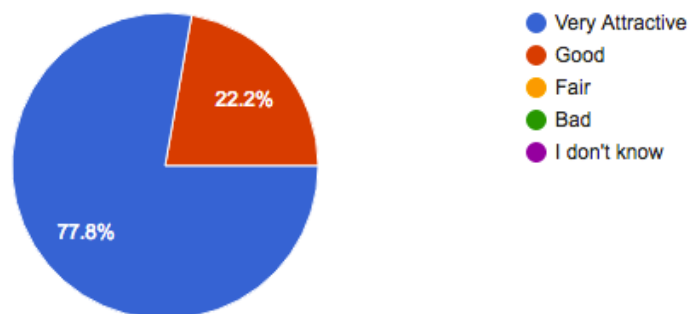
Would you like to use the app in future?

9 responses

- ● Yes
- ● No
- ● Maybe

22.2% 77.8%
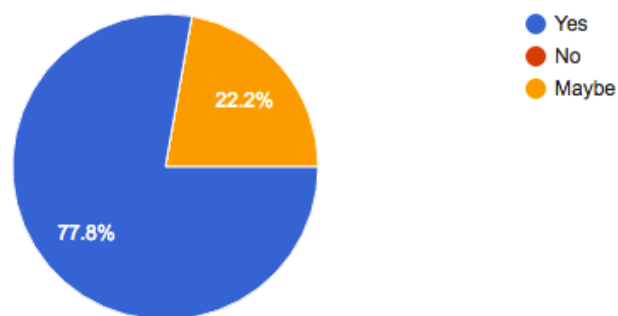
Part 1: How excited are you to test the app ?

9 responses

- Yes, I am very excited
- Looking forward to it
- Not excited
- Normal

22.2%

11.1%

66.7%

How many times have you felt lost in a building?

Column 1

| Every time i am in a hurry for a meeting | Every time i want find room on different floors or part of the building | I sometimes feel lost finding my hotel room |
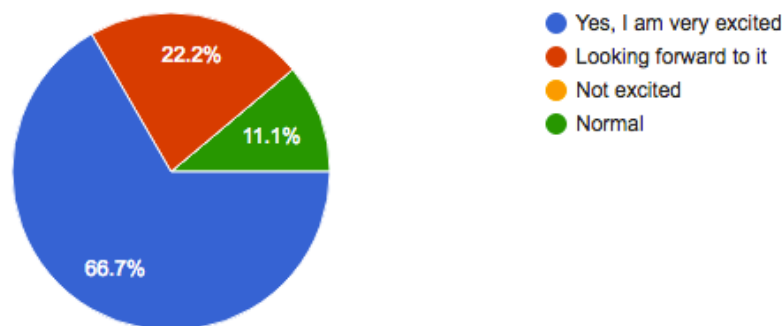
What do you do ?

9 responses

- Corporate Emplooye
- College Staff
- Student
- IT Department Staff
- Facilities Department Staff
- Hotel Staff
- Hotel Customer

22.2%

11.1%

11.1%

11.1%

11.1%

33.3%
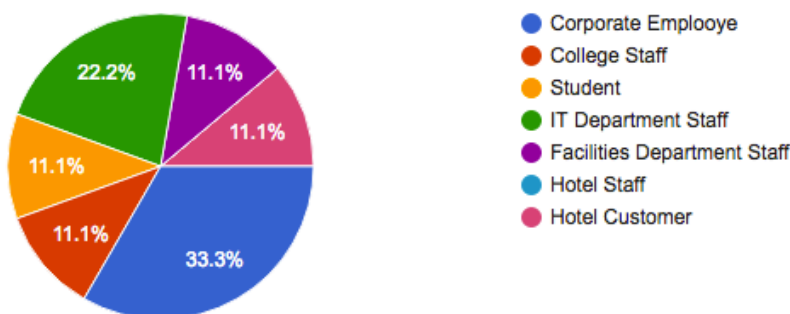
How many stars would you give to app reliability ?

9 responses

5 Star	6 (66.7%)
4 Star	3 (33.3%)
3 Star	0 (0%)
2 Star	0 (0%)
1 Star	0 (0%)
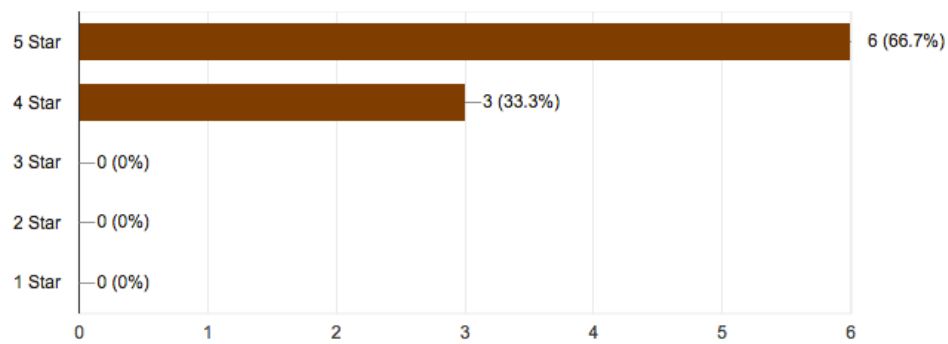
How many times did the app crash?

9 responses

- 0 times
- 1time
- 2 times
- 3 times
- more than 3
- more than 5

33.3% 11.1% 55.6%
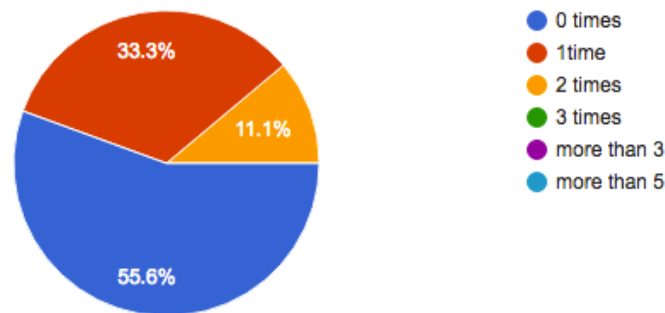
Other thoughts, improvement or comments

4 responses

I liked the app would like to see future releases and features of the app, i would like this app at my workplace to find meeting rooms

I enjoyed testing the app

nice app

very nice app i think all hotels should have it

So, as you can see the user's rated the app well in all areas, the users were from different backgrounds and all of them enjoyed using the app and user testing process for the project.

5   Conclusions

The project was quite difficult as there was not very clear documentation available on the new Geometry types in MySQL, i.e. spatial data types to store and retrieve data from the database, the core challenge of the project was to find solution to this problem through research which took a long time as the document was not available I had to try so many things to before solving the problem (the same way Thomas Edison had to try 1000's of times before he invented the light bulb and these days the bulbs are everywhere, cheap and easy to manufacture but doing it for the first time is always hard and challenging) and then implement the project. Other, factors were I was new to all the technologies I was using for this project so there was a big learning curve.

I believe the project has accomplished its goals and requirements that were proposed in the Technical report, the project has turn out to be innovative (theirs's no other project in the iOS or other markets that is using geometry data types in SQL and providing features like building and room listings, maps, gallery, bookings and more all combined in one product), the project is exploiting new technologies (SWIFT) and features (Geometry data in SQL) and despite me having some serious health issues this year from which I am still recovering, but I believe all the challenges I faced actually have made me more stronger mentally. Moreover, college was very supportive in understanding the health circumstances.

5.1   Application Improvements

There is always room for improvements. I believe for further success of the project, significant improvements can be achieved by working on app UI, adding a admin panel to the app which would allow the admin to add the buildings and rooms data straight from the app rather than using backend tools like phpMyAdmin.

Doing the project was such a great experience as I learned so many new things while doing this project – Example the technologies I used for this project I learned them from scratch i.e by finishing this project I now know few extra technologies – Swift 3, Php, CodeIgniter, Geo-Spatial Data storing and manipulation (in two different languages actually – JavaScript and php).

5.2   Implementation Improvements

Currently, the application uses GPS location to show the current location of the user which is not very accurate when indoors. It would be useful to develop an algorithm for indoor navigation along with the Usage of beacons and Wi-Fi triangulation to provide precise indoor location of the user. But again, this could be expensive to implement and maintain.

6   References

The project was not possible without the help of tutorial and videos and other resources available on the internet intensive care has been taken in order to properly reference if any work including images, text or code is used from third party resource. And I thank, deep from my heart to the authors of this work.

The report makes use of all resource available on https://moodle.ncirl.ie for example- the template, previous report as a starting base.

The project also makes use of several online resource including websites, blogs, papers, github repos, cocopods and other countless google searches and research which helped make this project a reality.

6.1 Research and Code

Codeigniter.com. (2017). *Getting Started With CodeIgniter — CodeIgniter 3.1.5 documentation*. [online] Available at: https://www.codeigniter.com/user_guide/overview/getting_started.html [Accessed 25 Jul. 2017].

Developer.apple.com. (2017). *NSURLConnection - Foundation | Apple Developer Documentation*. [online] Available at: https://developer.apple.com/documentation/foundation/nsurlconnection [Accessed 28 May 2017].

Developer.apple.com. (2017). *Xcode - IDE - Apple Developer*. [online] Available at: https://developer.apple.com/xcode/ide/ [Accessed 6 Jul. 2017].

eNotes. (2017). *Advantages Of Commercialization*. [online] Available at: https://www.enotes.com/homework-help/what-advantages-disadvantages-commercialization-338930 [Accessed 11 Jul. 2017].

GitHub. (2017). *Alamofire/Alamofire*. [online] Available at: https://github.com/Alamofire/Alamofire [Accessed 16 Mar. 2017].

GitHub. (2017). *SwiftyJSON/SwiftyJSON*. [online] Available at: https://github.com/SwiftyJSON/SwiftyJSON [Accessed 2 Jan. 2017].

Moodle.ncirl.ie. (2017). *Moodle resources*. [online] Available at: https://moodle.ncirl.ie [Accessed 28 Jul. 2017].

Swift Developer Blog. (2017). *User login and Registration with Swift Archives - Swift Developer Blog*. [online] Available at: http://swiftdeveloperblog.com/category/user-login-and-registration-with-swift/ [Accessed 15 Nov. 2016].

Team, C. (2017). *CocoaPods.org*. [online] Cocoapods.org. Available at: https://cocoapods.org/ [Accessed 28 Jul. 2017].

The Official Cloudways Blog. (2017). *How to Start PHP Programming: Basic PHP Scripts*. [online] Available at: https://www.cloudways.com/blog/how-to-start-php-programming/ [Accessed 28 Jul. 2017].

Useful documentation on MySQL Spatial Data Types

http://www.w3resource.com/mysql/mysql-spatial-data-types.php

Inserting Coordinates into MySQL (Stack Flow)

http://stackoverflow.com/questions/15453084/inserting-coordinates-into-mysql-polyfromtext-sql-syntax-error-returning-nul

6.2   Images

Intensive care has been taken to make sure all the images used in this report are properly referenced and the reference source url is included just below every image used form the internet. Any logo images used are the property of their individual owners and are only included in the report for educational purpose of showcasing/explaining the technology to the reader of the report and make the report more appealing to the reader as humans respond well to a combination of both text and images (then just text). The other mages used in this project report are screenshots from my own work.

Thirdly, I would like to thank my Supervisor Cristina Muntean for her availability during the project development, her useful suggestions during all the project meetings from the start to the last meeting. And her, giving me time to respond to all emails including some with silly questions.

My special thanks to our teacher Eamon Nolan as well as he was very helpful at every stage of the project development and helped resolve my confusions and gave useful answers when I needed them the most.

At, last I would like to thank my mother and family for being there for me whenever I was stressed and putting me back on track.

7 Appendix

7.1 Independent Research (Problem Solving – Storing Geo-Spatial data)

Aim: The aim of the document is to successfully research ways to be able to store geo spatial data in MySQL database

Why: Currently (24 Feb, 2017) I have been using SAP Hana HCP but the issue is its not free and is very expensive to move from trail to pro version, so the better alternative is MySQL if I am successfully able to implement what I was able to do with HCP. And, the advantages will be free of cost, HCP trail needs restart every 12 hours and auto deletes after 6 days which is a big disadvantage, MySQL will allow 99% uptime for web service.

- Creating a Sample table with geometry elements
- create table myspatialdata (id integer(7), bpolygon Geometry, mpoint Geometry);

Field	Type	Null	Key	Default	Extra
id	int(7)	YES		NULL	
bpolygon	geometry	YES		NULL	
mpoint	geometry	YES		NULL	

Inserting spatial data into MySQL

INSERT INTO myspatialdata (id, bpolygon, mpoint)

VALUES ('1','POLYGON(18 23, 0 0, 3 6, 8 9, 18 23)','POINT(18, 23)');
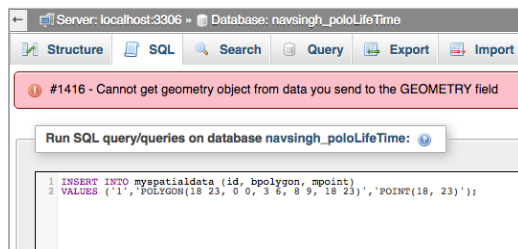
- Working SQL Statement to insert spatial data in MySQL (Polygon and Point)
- INSERT INTO myspatialdata (id, bpolygon, mpoint) VALUES (1,GeomFromText('POLYGON(18 23, 0 0, 3 6, 8 9, 18 23)'),GeomFromText('POINT(18, 23)'));
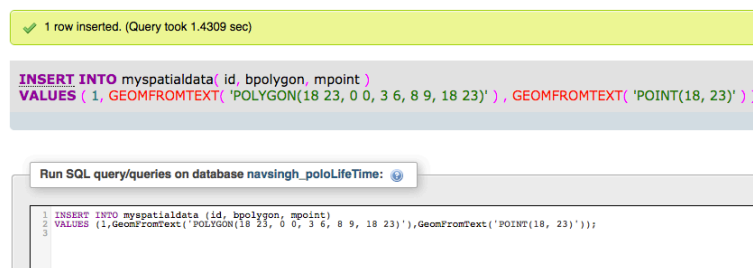
INSERT INTO myspatialdata (id, bpolygon, mpoint)

VALUES (2,GeomFromText('POLYGON(-74.13591384887695 40.93750722242824,-74.13522720336914 40.929726129575016,-74.15102005004883 40.9329683629703,-74.14329528808594 40.94256444133327)'),GeomFromText('POINT(-74.13591384887695,40.93750722242824)'));

```
INSERT INTO myspatialdata( id, bpolygon, mpoint )
VALUES ( 2, GEOMFROMTEXT(
'POLYGON(-74.13591384887695 40.93750722242824,-74.13522720336914 40.929726129575016,-74.15102005004883 40.932968362
) , GEOMFROMTEXT( 'POINT(-74.13591384887695,40.93750722242824)' ) )
```

[Edit] [Create PHP Code]

Run SQL query/queries on database navsingh_poloLifeTime: ⓘ

```
1  INSERT INTO myspatialdata (id, bpolygon, mpoint)
2  VALUES (2,GeomFromText('POLYGON(-74.13591384887695 40.93750722242824,-74.13522720336914 40.929726129575016,-74.15102005004883
   40.9329683629703,-74.14329528808594 40.94256444133327)'),GeomFromText('POINT(-74.13591384887695,40.93750722242824)'));
3
```

****

RESEARCH RESULTS: FAILED – The data is inserted in the

table but in the wrong format and I am not able to retrieve it.

****

Attempt: X

Inserting Geo-spatial Polygon into the MySQL Database

INSERT INTO `Buildings` (`Name`, `Shape`) VALUES ('Apt 15', PolyFromText('POLYGON((50.866753 5.686455, 50.859819 5.708942, 50.851475 5.722675, 50.841611 5.720615, 50.834023 5.708427, 50.840744 5.689373, 50.858735 5.673923, 50.866753 5.686455))'));

Reference: https://gis.stackexchange.com/questions/23900/how-to-add-polygon-in-mysql-database

Column	Type	Function	Null	Value	
Name	varchar(50)			Apt 16	
Shape	polygon	PolyFromText		POLYGON((53.37585 -6.31014,53.38348 -6.1844,53.32062 -6.204!	✏ Edit/Insert
					Go

****

RESEARCH RESULTS: PASSED (Working)

– The data is inserted in the table and able to retrieve it.

****

7.2   Monthly Journals

This section contains all the monthly journals during the project development, I attended all the supervisor meetings with my supervisor Cristina Hava Muntean, she helped me a lot during the meetings and through emails whenever I had questions or doubts or needed help.

7.2.1   Reflective Journal 1 (September 2016)

7.2.1.1   Introduction

My name is Navjot Singh, 21 years old, passionate about technology, research and sports, finished my 7 months' work placement with SAP SE in August 2016 to get back to college to finish 4th year of BSc honors in Computing in my chosen stream Software Development at National College of Ireland. At, SAP I worked with different projects with focus on iOS development using Swift which I love now. I feel satisfied while coding in general but love Java and Swift the most and enjoy making software, websites and mobile application both iOS and

Android in my free time. Alongside, college I do freelancing since first year of college which provides me with real world experience and help me support my financials and college tuition and expenses. Currently, I am working with a client in Dublin on a marketplace website project.

Computing is my first love but I enjoy other things too like I love to read emotions in people and learn about human psychology, communication skills and some business and marketing as well which connects me to my roots which is my family business where I used to help my Dad while back home in school years.

I feel very confident to start my 4th year with the support of teachers and fellow classmates and friends and want to perform the best of me as in this last year I want to get better each and every day in every aspect of my life studies, gym, table tennis as these college years won't come back and I want to create memories here at National College of Ireland.

And I am absolutely delighted about starting the Software Project and would love to create a lovely piece of software that could be useful to people and make my well-wishers proud.

7.2.1.2   First month 19th Sept to 7th October

Attended first class on 19th of September with joy of finishing 3rd year and realising 3 years has gone fast and this final year is the one that need the most attention and I have decided to dedicate few hours each day to my studies and some extra hours on weekends and focus on the project continuously and try to work on project with an agile approach which I followed during my work placement.

Week 1 (19th to 23rd)

Attended first class from Eamon and received good tips and notes to get us started and Eamon explained us how things with work for this semester and a brief overview of 4th year project after class reconnected with classmates and went to library to research on an idea I was thinking could be a good software project for my 4th year project and also looking for alternate ideas as well just having my options open.

Continued the research for on the idea and to decide which programming language, tools and technology to use to get best out of my skills and also ensuring the project, I select/work on should stand out and push me out of my comfort zone to get results that I wish to get in my final year project.

Read the past year project reports available on Moodle and information on plagiarism and time management for good results.

Received the Project Pitch schedule were in the 3rd week all students had to present their idea in front of a panel of teacher without any props like presentation files, wireframes etc. Which was a little breath taking for some of us?

Overall, I believe first week was a good eye opener on what challenges to aspect in later weeks.

Week 2 (26th to 30th)

Monday, continued the research and discussed my ideas with Eamon and asked questions to clarify my doubts on What Eamon and other teachers will be expecting from us during the project pitch and on Thursday decided to go ahead with an idea that I researched about in my work placement and uploaded the Presentation file for the project pitch next which other lecturers would see to look if my idea had a potential to be developed further.

This week has been though since making a choice out of 100's of ideas floating in your mind has never been an easy task for me and I believe almost everyone in the class felt the same emotion.

Week 3 (3rd to 7th October)

Monday started with a lot of questions asked in class and Eamon answered them all which I feel was helpful plus Eamon also delivered a little harsh speech to motivate the students to going to find an idea if they haven't already.

And Wednesday 5th Oct, at 4:10 pm this was the time for my project pitch went 15 mins early outside the room to ensure I would be calm and collect during the project pitch. Exactly at 4:10 pm, Paul Stynes the chairperson for my project pitch came to call me for the pitch was a tough moment because know I had to present the idea I had been researching all this time and I have to deliver.

Started my Project Pitch,

My Project idea is to Create an iOS application for finding meeting rooms and people in a organisation or an educational institute, I have chosen this idea as I believe the idea has a good commercial implication as I when I was doing my work placement at SAP SE, I myself struggled finding meeting rooms at times and also found later other people were facing the same problem as well even employees that have been there for years would not know the meeting rooms and people in other SAP building and sometimes on different floors of the same building.

7.2.1.3 My Achievements

This month had been a good productive month can't wait to know if the panel will approve my idea or ask me for some improvements as they suggested during the project pitch.

Overall, a good month. Worked well and would like to continue to work the same way and but would like to work even harder next month plus also send more time in library to research and work on the project.

7.2.1.4 Supervisor Meetings

We haven't been assigned a supervisor yet but during this month I had a lot of chats with Eamon regarding project and asked questions and cleared my doubts.

I have presented my project idea to the panel of teachers.

Mr. Paul Stynes (Chairperson)

Mr. Ralf Bierig (Judge 2)

Mr. Eugene McLaughlin (Judge 3)

Date of Meeting: Wednesday, 5th Sept 2016 at 4:10 pm

Items discussed: Project Idea

Action Items: Waiting for approval on project idea.

7.2.2   Reflective Journal 2 (October 2016)

7.2.2.1   Introduction
Last month every student presented their ideas to a panel of teachers and this document will provide a summary of the tasks and progress I made on weekly basis in second month after the completion of first month for my software project.

Week 4 (10th Oct - 14th Oct)

First few days of the week went nail biting waiting for the project pitch results. Then, our teacher Eamon announced the results in class which students got their project idea approved and which were not luck enough and had to select from teachers proposed ideas.

And luckily, the panel – Mr. Paul Stynes (Chairperson), Mr. Ralf Bierig (Judge 2)

Mr. Eugene McLaughlin (Judge 3)

Approved my project idea with 3 Yes ticks from all 3 teachers and I felt very confident and happy about it, I felt teachers appreciated my idea and know its my responsibility towards them that I deliver well and prove myself that the idea I proposed can be developed and will help people solve daily problems of finding rooms through my iOS application – Room Assistant.

And towards, the end of the week I started the project proposal document.

Week 5 (17th Oct - 21st Oct)

On Monday, worked on Project proposal and created a Github repo for the project, for source control.

Github repository: https://github.com/Virksaabnavjot/MRPro

But sadly by Tuesday, I felt sick and went to saw a doctor and was recommended a weeks rest and could not work on the proposal or attend college for the rest of the week.

Project proposal submission on Friday (I submitted what I finished before getting sick and informed my teacher Eamon Nolan).

Week 6 (24th Oct - 28th Oct)

Felt a little healthy and again started working and completed the things that were left to finish in the project proposal.

By Wednesday, I started Requirements Specification Document and worked on it rest of the week.

And during this week, I was assigned a supervisor.

My supervisor: Cristina Hava Muntean

Week 7 (31st Oct - 4th Nov)

Reading Week

Reading week is the week I love since first year, it allows to catch up if left behind in any module and prepare for the upcoming tests and assignments.

Since, we have major CA test coming up for Web Services and API's when back to college after reading week, like most of the students have been studying for the test and also continued

dedicating some time each day to requirements specification document and some research on the project.

7.2.2.2 My Achievements

This month was good, recovered from bad health and completed a lot of work.

Finished the Project proposal, working on requirements specifications document and also worked hard in other modules and submitted a chess project feeling very motivated good progress happy with myself the way I am going and hope to go the same positive path for the rest of the year.

7.2.2.3 Supervisor Meeting

I have contacted my supervisor and the first meeting is due on 10 November.

Supervisor: Cristina Hava Muntean

7.2.3 Reflective Journal 3 (November 2016)

7.2.3.1 Introduction

Last month was good and project proposal was completed and I was working on requirements specification document.

7.2.3.2 My Achievements

This month, I have finished the requirements specification document and stepped a SAP Hana Trail account, finished the database design and implemented it in the database, implemented a JavaScript service that returns JSON which our client uses to display information and provide functionality in the app.

With suggestions from my supervisor made some changes to project proposal document and requirements spec document to add few more feature which I will be working on next semester.

Implement the application in Swift 3 using Xcode and implemented features like list view to show buildings and meeting rooms and a map using Mapkit to display building as a polygon and meeting room as a point on the map and display current location of the user using CoreLocation.

Also, working on the Technical report about 70% finished and getting the project ready for mid term presentation.

7.2.3.3 My Reflection

Absolutely delighted with this month and a lot of functionality was implemented this month and some big documents like tech report are almost done. So, I feel proud about this month the only thing that didn't go smooth in the beginning was parsing JSON in Swift but luckily found an open source library to take care of that effectively.

7.2.3.4 Intended Changes

Finish Technical Report and Keep updating project documents to achieve highest possible quality.

7.2.3.5 Supervisor Meeting

Supervisor: Cristina Hava Muntean

Supervisor meetings this month were very productive Cristina asked a couple of questions about the project and suggested improvements and provided with good advice on how to manage time and get better results and develop a high quality application.

7.2.4 Reflective Journal (December 2016)

7.2.4.1 Introduction
This month, we have a mid-term presentation and exams soon after that in Jan 2017.

7.2.4.2 My Achievements
Continuing the work from last month, I finished the rest of the technical report and I worked in Xcode ide and using swift and started and implemented GUI and some features like JSON Parsing, list views for buildings and meeting rooms and map view to show/draw the National college of Ireland's building on the map and show meeting rooms available in it.

Testing – Started testing using XCTest in Xcode.

Presented the project on Mid-term presentation day (19th Dec 2016, at 10:30 am) in Executive 3 (Room, at NCI).

And, also trying to manage time between modules and give some time to other module projects as we have an extra module and project this year.

7.2.4.3 My Reflection
Overall, a useful month and very busy month not just for this project but other modules too. So, a lot finished this month. And received, valuable suggestions and recommendations from teachers.

7.2.4.4 Intended Changes and Supervisor Meeting
Supervisor: Cristina Hava Muntean

(This moth I had mid-term presentation with our Supervisor and other teacher Vikas Sahani)

And the teacher's recommendation was to add a business case – ios vs (android and hybrid applications) and write some project limitations for the final report and presentation.

Recommended 1 day research on, Xamarin, 3d Visualisation.

7.2.5 Reflective Journal (January 2017)

7.2.5.1 Introduction
December was a great productive month, we did mid-term presentation and got really busy with end of semester exams.

7.2.5.2 My Achievements
January was a busy month due to exams and about 70% of the months time was spent towards studying for end of semester exams and achieve good grades.

When, the exams were finished I took a small 3 day break and started working again on the project, worked on database design for room booking functionality and work with Ios app a bit

and currently trying to implement room booking functionality in the app. As, I said this month was mostly taken by exams I plan to do a lot of work in the project next month.

7.2.5.3 My Reflection
Overall, a useful month and very busy month and later received great results in final exams.

7.2.5.4 Supervisor Meeting
Supervisor: Cristina Hava Muntean

Once, again a productive meeting. Cristina talked us through how we got marked on the mid-term presentation. And, we discussed what to do next for the semester.

7.2.6 Reflective Journal (February 2017)

7.2.6.1 Introduction
January flew away fast Christmas, new year and then exams a lot done in this month and fairly good progress on the project and implemented the room booking functionality of the application.

7.2.6.2 My Achievements
February the college has started again and these days I am spending a lot of time on the project I have been able to implement photo gallery, review system, photo upload functionality, user registration and call functionality to the application. Alongside, I have been doing unit tests for all these functionalities to make sure the quality of code is at higher standards and also the application is less prone to bugs and errors. Overall, very happy with progress still lot more to do but I am excited because I love programming in Swift.

And outside college, as a hobby project I started a new website: http://gymandnutrition.com which is a forum based site with a blog, to help people with their fitness goals. Loving this year. Its being productive and I am achieving more than I expected.

7.2.6.3 My Reflection
Brilliant month from the getting things done point of view I was able to spend a lot more time I was able to before on the project and the results are lovely I have fully functional bits of application done along with unit tests. Next, month I will start some user testing alongside as well.

7.2.6.4 Supervisor Meeting
Supervisor: Cristina Hava Muntean

Cristina is great help as always, all the students that have Cristina as supervisor sat down together, she asked us questions and provided suggestions and improvements also this group activity helped all the students to learn from each other as well.

7.2.7   Reflective Journal (March 2017)

7.2.7.1   Introduction
One of the best months while doing this project was February, now looking forward to march as an another opportunity to do a lot of work on the project.

7.2.7.2   My Achievements
This month I was able to start doing user testing on the college students, friends and colleges at part-time work to get their views on the application and I also carried out heuristic evaluations on the application using the knowledge learned from the final year module Usability design in order to make sure the application's usability is up-to the mark and it does not violate the recognized usability design principles because the users crave a good gui which is easy to use and understand and easy to navigate. On the coding part, I am working with Geo location notifications, navigation bit of the application.

7.2.7.3   My Reflection
A good month, a lot of work done but a lower pace than previous moth because I have also being busy doing assignments, projects, CA's for other modules as well which are important and require a lot of learning time.

7.2.7.4   Supervisor Meeting
Supervisor: Cristina Hava Muntean

I felt sick in March, was not able to attend the meeting later this meeting was rescheduled in July 2017.

During the meeting, I discussed my progress with Cristina after recovering from bad health, informed her I wanted to make few changes to the selected technologies for the project. She gave her feedback on progess and I went back and started making the changes based on the feedback received during the supervisor meeting.

7.3   Heuristic Evaluation

Visibility of System Status

The question to be answered here is "Where I am in the and where to go next?"

The following tasks/questions will be included in heuristic testing script –

- Does each page have same design and colour schemes?
- Does each page have a consistent navigational structure allows you to freely move from view to view?
- Does each page have easy navigation and links through navigation controller?

Match b/w System and Real World

The app should use easy to understand language that everyone can understand –

The following tasks/questions will be included in heuristic testing script –

- Does the app use any complex words?

User Control and Freedom

When the user uses the app it is important that they feel comfortable and in control of their experience. If a user become lost and is unable to find path, it is very likely the user straight forward exits your site.

The following tasks/questions will be included in heuristic testing script –

- Is there a navigation controller on every page?
- Can a user easily move from a view to different view using the apps navigational scheme?
- Is the navigational scheme consistent?
- Does the app allow you to change background colours ?

Consistency and Standards

The app needs to be consistent and the user does not have to wonder whether different buttons, actions or words mean the same.

The following tasks/questions will be included in heuristic testing script –

- Do the navigation elements use the same words throughout the app?
- Does the view have different name in other parts of the app? Example: Info and User manual but not both.
- Is the site layout consistent?

Error Prevention

Errors and mistakes happen everyday. How app responds to errors and prevent them is important.

- Does the app display clear error messages when wrong inputs are given?

Recognition rather than recall

If the user has to recall things its not a good thing, minimizing the user load by making options, actions and objects highly visible helps.

The following tasks/questions will be included in heuristic testing script –

- Is there inconsistent graphics representing the same function?
- Is there a use of more than 1 logo?
- Does every view have a title that describes the topic of the view?

Flexibility and Efficiency of Use

Your app's interface should be flexible and very efficient to use.

You should offer users number of options to finding content and be able to achieve their goals.

The following tasks/questions will be included in heuristic testing script –

- Is the app Flexible example if it includes, lists, basic search forms, alphabetical index etc.?

- Is the provided list view designed in a logical manner?
- Do the app allows you to register? If yes, if then check if you can login and recover any information provided previously.

Aesthetic and Minimalistic design

This heuristic state the rarely needed information is accessible on the app through a link without interfering with more relevant information.

The following tasks/questions will be included in heuristic testing script –

- Does the site have a "Info" or "User manual" or similar links?
- Is there a need for scrolling when viewing content?

Helps users recognize, diagnose, and recover form errors

This heuristic state that your error messages should be provided in simple language with no codes and should give a detailed info of the error and possibly suggest solution.

The following tasks/questions will be included in heuristic testing script –

- Are the error messages easy to understand?
- Do the errors can be seen easily and be cancelled and go to the previous step?
- Try to register for on the app and input an invalid email format and see if the error correctly explains the error and the type of language used?

Help and documentation

Generally, all apps require some form of help section to cater all types of users to be on the safe side. For example, an elderly person may face problems doing some basic tasks that everyone else can perform with ease.

The following tasks/questions will be included in heuristic testing script –

- Is there help section available on the site?
- Is there search feature available?
- Can you easily register and login back and go to My profile (homepage)?

7.4 IOS vs Android vs Hybrid

During the mid-term presentation, I was asked to write a short description in the final report why I choose to develop an IOS app rather than android or hybrid.

To keep it short and simple I will bullet point the main reasons for selection of IOS over others –

- I am a vivid user of apple devices in general and wanted to learn to develop apps for iPhone and iPads.
- IOS app have more lucrative market than android apps in general.
- I wanted to learn something new as I had a lot of experience doing android programming I didn't wanted to repeat and instead learn new technologies.
- Why not hybrid? Hybrid is great but I wanted to learn a native IOS application.
- Wanted to learn Swift 3.

7.5 End-User Survey Results

Can be found here: https://docs.google.com/forms/d/1YfKj3BdmXif87Yk1yuFjm1akQO-4RnWyYkCB3YrKRjs/edit#responses (you will need to do the survey to see the results, the survey explains everything and have an app demonstration video (that will stimulate the same experience as if you were using the app by yourself and you wont have any problem doing the survey

or

End User Testing (Survey)_MRPro.pdf (file submitted with the code deliverable).

The End

MRPro Business Page: https://navsingh.org.uk/mrpro