



**Diploma in
Information Technology**

**AY2018/2019
Semester 1 (Stage 3A)**

Proposed Project: Directie

Phase 1 Report

Group 1

Members:

Ng Hong Yao	(1625893)
Chia Kwee Cheng	(1626120)
Tay Wei Sern	(1626203)
Garick Chong	(1626555)
Chua Wei	(1639290)

Supervisor: Ms Lin Zhao



1 Project Overview	4
1.1 Objectives	4
1.2 Target Audience	4
1.3 Problem Statements	4
1.4 Requirements	4
1.5 User Story	5
1.6 Critical Success Factor	5
2 Proposed Application	6
2.1 Brief explanation of application name	6
2.2 Functions of Web Application (Admin Portal)	6
2.2.1 Authentication	6
2.2.2 Analytical View	6
2.2.3 Manage Beacons	6
2.2.4 Manage Beacon Directions	7
2.2.5 Manage Beacon Relations	7
2.2.6 Manage Units	7
2.2.7 Manage Facilities	7
2.2.8 Manage Users	7
2.3 Functions of Mobile Application	8
2.3.1 Authentication	8
2.3.2 Text to Speech Navigation	8
2.3.3 Update Profile	8
2.3 Web Application Wireframes	9
2.3.1 Login Screen	9
2.3.2 Dashboard	10
2.3.3 Beacons	11
2.1.4 Add Beacon	12
2.1.5 Update Beacon	13
2.1.6 Beacon Directions	14
2.1.7 Add Beacon Direction	15
2.1.8 Update Beacon Direction	16
2.1.9 Beacon Relations	17
2.1.10 Add Beacon Relation	18
2.1.11 Update Beacon Relation	19
2.1.12 Units	20
2.1.13 Add Unit	21
2.1.14 Update Unit	22



2.1.15 Facilities	23
2.1.16 Add Facility	24
2.1.17 Update Facility	25
2.1.18 Users	26
2.1.19 Update User	27
2.2 Mobile Application Wireframes	28
2.2.1 Splash Screen	28
2.2.2 Login	29
2.2.3 Registration	30
2.2.4 Home	31
2.2.5 Navigation Page	32
2.2.6 Side Menu	33
2.2.7 Edit Profile	34
3 Detailed Specifications	35
3.1 System Architecture	35
3.2 Technologies Used	36
3.2.1 Ionic 3 Framework	36
3.2.1.1 Why Ionic 3 Framework	36
3.2.2 Cloud Computing	36
3.2.2.1 Why Amazon Web Services (AWS)	36
3.3 Database Design	37
3.3.1 User	37
3.3.2 UserInfo	37
3.3.3 Role	38
3.3.4 UserRole	38
3.3.5 AuditLog	38
3.3.6 Beacon	38
3.3.7 BRelation	39
3.3.8 Direction	39
3.3.9 BDirection	39
3.3.10 Room	39
3.3.11 BeaconRoom	40
3.3.12 Facility	40
3.3.13 FacilityRoom	40
3.3.14 Image	40
3.3.15 FacilityImage	40
4 Unified Modeling Language (UML) Diagrams	41
4.1 Front-End Use Case	41



4.2 Back-End Use Case	42
Appendix A - Research	43
A.1 Objectives	43
A.2 Aipoly Vision	44
A.3 Be My Eyes	45
A.4 Conclusion	46
Appendix B - Meeting Minutes	47
Meeting Minute I	47
Meeting Minute II	48
Meeting Minute III	50
Meeting Minute IV	54
Appendix C - Product Backlog	56
Appendix D - Sprint Backlog	57
Appendix E - Project Plan	58
Appendix F - Test Cases	59
Test Case 1	59
Test Case 2	61
Test Case 3	63
Test Case 4	65
Test Case 5	67
Test Case 6	70



1 Project Overview

For this Final Year Project (FYP), our group will be developing an Android mobile application that will be used to assist the visually impaired in navigating within an indoor environment by using beacons. In addition, a back-end administrator portal will also be created to let administrators manage information and resources easily.

1.1 Objectives

To assist the Visually Impaired in navigation within an indoor environment through a mobile application.

1.2 Target Audience

The target audience of this project are the Visually Impaired (VI).

Although our main target audience is the Visually Impaired and designs were made to suit them, the core concept and functions of the application can be used to benefit others as well.

1.3 Problem Statements

The visually impaired may be unfamiliar with the environment and thus, get lost easily. Even when there are helpers around to guide the visually impaired around it is not always the case that they can help out every single one of them too.

1.4 Requirements

A **Front-End** mobile application which should have functions such as

- User Login (Google, Facebook, Directie User)
- Detect user's location
- Routing of shortest path to destination
- Speech navigation
- View user profile
- Edit user profile

A **Back-End** web application which should have functions such as

- Manage Beacons (Creating, Updating & Deleting of beacon)
- Manage Beacon Relations (Creating, Updating & Deleting of beacon relation)
- Manage Beacon Directions (Creating, Updating & Deleting of beacon direction)
- Manage Facilities (Creating, Updating & Deleting of facility)
- Manage Units (Creating, Updating & Deleting of unit)
- Manage Users (Creating, Updating & Deleting of user)



1.5 User Story

A user with visual impairment wants to navigate to a destination within an indoor environment. The user will access the mobile application that is installed on his/her mobile phone. The mobile application will be able to know the user's location within the indoor environment where all rooms will have a beacon planted. The nearest beacon to the user will be used as an indication of the user's current location and it will be indicated to the user through text to speech.

For example, when the user is at T2139 and wants to navigate to the nearest toilet, he/she will launch the mobile application and it will detect the nearest beacon to determine to user's current location and it will be read out to the user to inform the user where he/she is at.

The user would then input where the desired destination would be through speech and the application will guide the user to the destination step by step with the shortest route through speech. Once the user reaches the destination, the application will notify the user through speech as well.

1.6 Critical Success Factor

The design of the application will allow for a simple and easy to use interface. Thus, allowing users to be able to navigate with the application easily by themselves.

In addition, through the simple and intuitive administrator portal, implementation of this system in other locations can be done easily by the administrator.



2 Proposed Application

2.1 Brief explanation of application name

After a few discussion we had in our group, we have proposed a name for our application, Directie. The word Directie comes from direction in Romanian language.

In this project, we are focusing on giving directions to the Visually Impaired, so that they can navigate around freely in homes, nursing homes or schools with special educational needs without having any physical help.

2.2 Functions of Web Application (Admin Portal)

The admin portal will be used by the administrator to manage all resources and assets like beacons, beacon relations, facilities and many more.

2.2.1 Authentication

For the administrator portal, only the administrator will be able to access the page. This means that he/she will require an administrator account to be able access the functions of the admin portal.

2.2.2 Analytical View

Through the dashboard in the administrator portal, the administrator will be able to view analytics on data collected by the mobile application. Analytics such as how many people has been using the mobile app by day and time, the average time taken for people that uses to app to travel from 1 beacon to the other can be viewed by the administrator.

These data will be used to further improve the mobile application. For example, when it takes a long time for the majority of the users to travel from 1 beacon to the other, it could be that they find the route difficult to navigate through. This will then follow up with optimization and improvements to the route/directions given.

2.2.3 Manage Beacons

In the administrator portal, the administrator will be able to View, Add, Update & Delete beacons. When adding a beacon, the admin will have to provide required information regarding the beacon. There will be input validation for all input fields. Once the administrator has filled in the required fields and click on add, the information will be saved into the database and be used by the mobile application. The administrator will also be able to toggle to enable or disable a beacon.



2.2.4 Manage Beacon Directions

The administrator will be able to View, Add, Update & Delete beacon directions as well. When adding a beacon direction, the admin will have to provide required information regarding the beacon direction. There will be input validation for all input fields. Once the administrator has filled in the required fields he can proceed to click on add which will save the information to the database and be used by the mobile application.

2.2.5 Manage Beacon Relations

Similarly, the administrator will be able to View, Add, Update & Delete beacon relation. When adding a beacon relation, the admin will have to provide required information regarding the beacon relation. There will be input validation for all input fields. Once the administrator has filled in the required fields and click on add, the information will be saved into the database and be used by the mobile application.

2.2.6 Manage Units

For units, an administrator can View, Add, Update & Delete too. When the administrator adds a unit, he needs to enter the required fields and is also able to upload an image of the unit. Once the administrator has filled in all the fields and clicks on add, the unit information will be saved into the database and be used by the mobile application.

2.2.7 Manage Facilities

An administrator can View, Add, Update & Delete facilities as well. When the administrator adds a facility, required fields will need to be entered and the administrator is also able to upload an image of the facility. Once the administrator has filled in all the fields and clicks on add, the facility information will be saved into the database and be used by the mobile application.

2.2.8 Manage Users

Unlike the others, the administrator can only View, Update and Delete for users. The updating of a user is only to update the role of a user. The administrator can also delete a user, which means to remove the user's account from the database.



2.3 Functions of Mobile Application

2.3.1 Authentication

To be able to access and use the functions of the mobile application, users will be required to login beforehand. There are 2 methods of authentication for the user to choose from, a Directie account or through their own social media accounts (Google & Facebook). If the user does not have a Directie account, he/she can choose to sign up for one.

2.3.2 Text to Speech Navigation

The user will be able to enter a destination he wants to go to and the application will provide step by step instructions on how to get there through speech. The application will provide feedback and instructions based on beacon, time, error or positive reinforcement.

An example of positive reinforcement is when the user is walking in the right direction as given by the instructions, the application will notify the user that he/she is walking in the right direction and to continue the path.

2.3.3 Update Profile

The user will be able to edit his/her profile information if desired. The profile picture can also be changed.



2.3 Web Application Wireframes

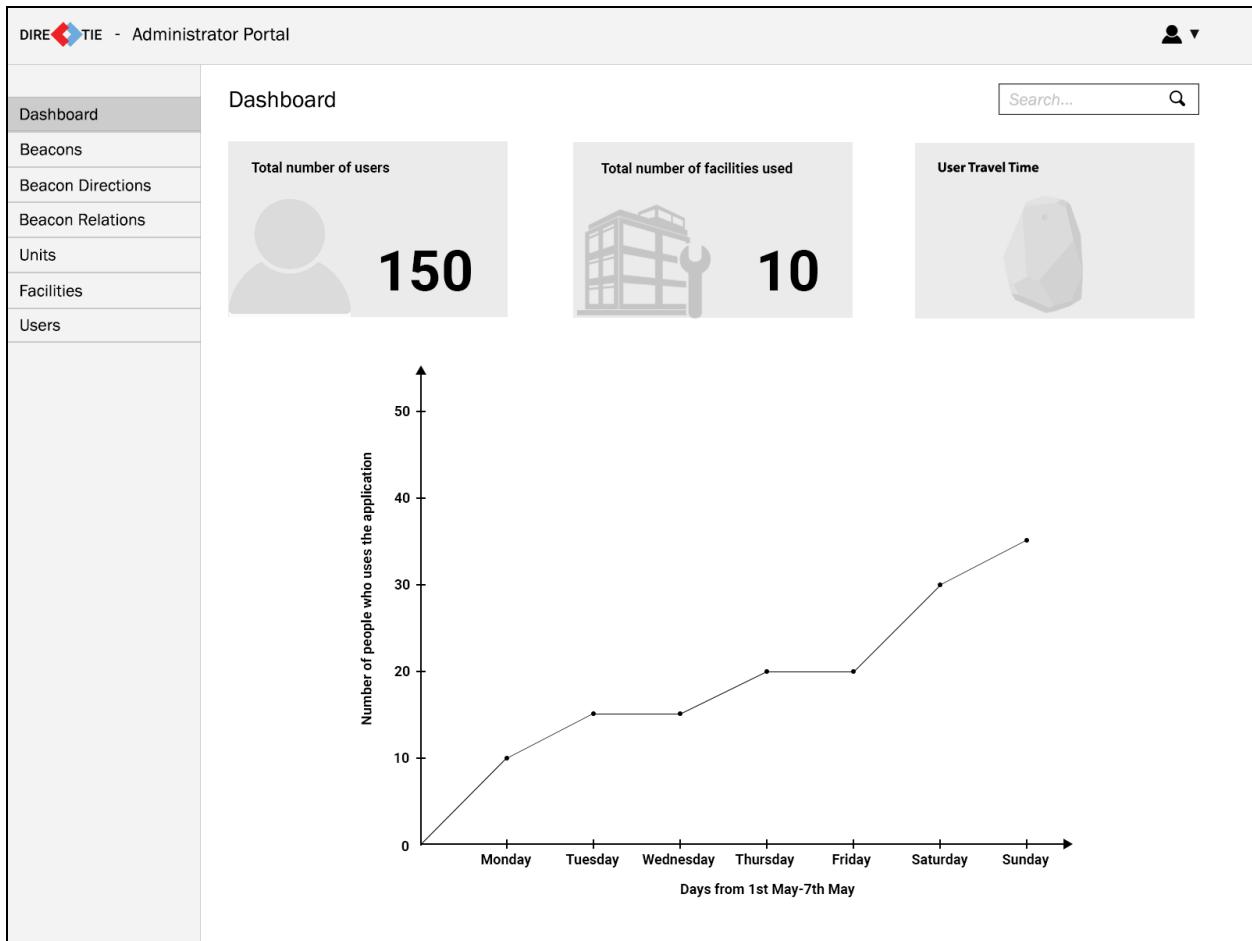
2.3.1 Login Screen



- This is the screen which the administrator will see after loading the web application. He/She will be required to login before being able to access the functions of the admin portal.



2.3.2 Dashboard



- This is the screen that will allow the administrator to view analytics such as number of users that used the application in a week and how long users take to travel from 1 beacon to the other.
- The administrator will be able to view 3 different categories of analytics. Users, Facilities and Beacons by clicking on the 3 rectangles at the top respectively.



2.3.3 Beacons

The screenshot shows the 'Beacons' section of the Administrator Portal. On the left is a sidebar with links: Dashboard, Beacons (which is selected), Beacon Directions, Beacon Relations, Units, Facilities, and Users. The main area has a header 'Beacons' with a search bar. Below is a table with 10 entries. The table columns are: ID, UUID, Major, Minor, Turning Pt, Action, and Toggle. Each row contains a set of these values. At the bottom, it says 'Showing 1 to 10 of 57 entries' and includes navigation buttons for Previous, Next, and page numbers 1, 2, 3.

ID	UUID	Major	Minor	Turning Pt	Action	Toggle
1	12345678-1234-1234-1234-12345678	16	555	✓	<button>Update</button>	<input type="checkbox"/>
2	12345678-1234-1234-1234-12345678	11	666	✓	<button>Update</button>	<input type="checkbox"/>
3	12345678-1234-1234-1234-12345678	12	123	✗	<button>Update</button>	<input type="checkbox"/>
4	12345678-1234-1234-1234-12345678	25	1001	✗	<button>Update</button>	<input type="checkbox"/>
5	12345678-1234-1234-1234-12345678	22	321	✓	<button>Update</button>	<input type="checkbox"/>
6	12345678-1234-1234-1234-12345678	15	1552	✗	<button>Update</button>	<input type="checkbox"/>
7	12345678-1234-1234-1234-12345678	32	1234	✓	<button>Update</button>	<input type="checkbox"/>
8	12345678-1234-1234-1234-12345678	24	222	✓	<button>Update</button>	<input type="checkbox"/>
9	12345678-1234-1234-1234-12345678	28	213	✗	<button>Update</button>	<input type="checkbox"/>
10	12345678-1234-1234-1234-12345678	17	432	✗	<button>Update</button>	<input type="checkbox"/>

- This is the screen that allows an administrator to view all beacon's information such as UUID, Major, Minor and whether it is a turning point or not. All these information will be retrieved from the database.
- When the administrator clicks on the "Add" button at the top right, he will be brought to the add beacon screen.
- When the administrator clicks on the "Update" button under the action column, he will be brought to the update beacon screen.
- The administrator can also toggle to enable or disable a particular beacon using the switch in the "Toggle" column.



2.1.4 Add Beacon

The screenshot shows the 'Administrator Portal' interface. On the left, a sidebar menu includes 'Dashboard', 'Beacons' (which is selected), 'Beacon Directions', 'Beacon Relations', 'Units', 'Facilities', and 'Users'. The main content area is titled 'Beacons' and features a search bar and a table with 10 entries. A modal window titled 'Add Beacon' is overlaid on the table. The modal contains fields for 'BeaconID' (1), 'UUID' (12345678-1234-1234-1234-12345678), 'Major' (88), and 'Minor' (888). It also has a 'Turning Point' checkbox which is checked. At the bottom of the modal are 'Cancel' and 'Add' buttons. The table below the modal shows columns for ID, UUID, Major, Minor, Turning Point, and Action/Toggle. The first row of the table is highlighted.

ID	UUID	Major	Minor	Turning Point	Action	Toggle
1	12345678-1234-1234-1234-12345678	28	213	<input checked="" type="checkbox"/>	Update	<input type="checkbox"/>
2	12345678-1234-1234-1234-12345678	17	432	<input checked="" type="checkbox"/>	Update	<input type="checkbox"/>
3	12345678	1234	1234	<input checked="" type="checkbox"/>	Update	<input type="checkbox"/>
4	12345678	1234	1234	<input checked="" type="checkbox"/>	Update	<input type="checkbox"/>
5	12345678	1234	1234	<input checked="" type="checkbox"/>	Update	<input type="checkbox"/>
6	12345678	1234	1234	<input checked="" type="checkbox"/>	Update	<input type="checkbox"/>
7	12345678	1234	1234	<input checked="" type="checkbox"/>	Update	<input type="checkbox"/>
8	12345678	1234	1234	<input checked="" type="checkbox"/>	Update	<input type="checkbox"/>
9	12345678-1234-1234-1234-12345678	28	213	<input checked="" type="checkbox"/>	Update	<input type="checkbox"/>
10	12345678-1234-1234-1234-12345678	17	432	<input checked="" type="checkbox"/>	Update	<input type="checkbox"/>

- This screen allows an administrator to add in beacons. Information such as UUID, Major, Minor and whether it is a turning point or not will be required.
- All fields will have input validation and must be filled. If a field is not entered, it will be indicated to the admin to fill in the empty fields.
- Once the administrator clicks “Add” the information will be saved into the database.

2.1.5 Update Beacon

The screenshot shows the 'Administrator Portal' interface. On the left is a sidebar with links: Dashboard, Beacons (which is selected), Beacon Directions, Beacon Relations, Units, Facilities, and Users. The main area is titled 'Beacons' and contains a search bar and a table with 10 entries. A modal window titled 'Update Beacon' is open over the table. The modal has fields for BeaconID (1), UUID (12345678-1234-1234-1234-12345678), Major (16), Minor (888), and Turning Point (checkbox checked). At the bottom of the modal are 'Cancel', 'Delete' (disabled), and 'Save' buttons. The table in the background shows columns: ID, BeaconID, UUID, Major, Minor, Turning Point, Z8, Z15, and X. The first row's Z8 value is 28.

- This screen allows an administrator to update a beacon's information.
- When the administrator clicks "Delete", the beacon's information will be removed from the database.
- When the administrator clicks "Save", any changes made to the beacon's information will be saved to the database and overwrite the old information for that particular beacon.



2.1.6 Beacon Directions

DIRE[◆]TIE - Administrator Portal User ▾

Beacon Directions

Search...

Show 10 entries **Add**

BDID	Previous	Current	Next	Direction	Action
1	4	1	2	Straight	Update
2	4	1	3	Left	Update
3	1	4	8	Straight	Update
4	2	1	5	Left	Update
5	2	1	3	Right	Update
6	2	1	4	Straight	Update
7	6	2	1	Straight	Update
8	8	4	1	Straight	Update
9	4	1	5	Right	Update
10	5	1	4	Left	Update

Showing 1 to 10 of 57 entries Previous 1 2 3 Next

- This screen allows an administrator to view the directions a person should take when going from 1 particular beacon to the other. All these information will be retrieved from the database.
- When the administrator clicks on the “Add” button at the top right, he will be brought to the add beacon direction screen.
- When the administrator clicks on the “Update” button under the action column, he will be brought to the update beacon direction screen.



2.1.7 Add Beacon Direction

The screenshot shows the 'Administrator Portal' interface. On the left is a vertical sidebar with links: Dashboard, Beacons, **Beacon Directions**, Beacon Relations, Units, Facilities, and Users. The 'Beacon Directions' link is highlighted. The main area has a title 'Beacon Directions' with a search bar and a dropdown 'Show 10 entries'. A large modal window is open, titled 'Add Beacon Direction'. Inside the modal, there are four dropdown menus labeled 'Previous' (12), 'Current' (11), 'Next' (10), and 'Direction' (Straight). At the bottom of the modal are 'Cancel' and 'Add' buttons. Below the modal is a table with three rows of data:

8	8	4	1	Straight	Action
9	4	1	5	Right	Update
10	5	1	4	Left	Update

At the bottom of the page, it says 'Showing 1 to 10 of 57 entries' with a navigation bar showing pages 1, 2, 3, and Next.

- This screen allows an administrator to add in beacon directions. The administrator will have to select the Previous, Current and Next beacon IDs and give it a direction through the drop-down menu.
- All fields must be filled. If a field is not entered, it will be indicated to the admin to select something for that field.
- Once the administrator clicks “Add” the information will be saved into the database.



2.1.8 Update Beacon Direction

The screenshot shows the 'Administrator Portal' interface. On the left is a vertical navigation menu with options: Dashboard, Beacons, Beacon Directions (which is selected and highlighted in grey), Beacon Relations, Units, Facilities, and Users. The main content area is titled 'Beacon Directions' and contains a search bar with placeholder 'Search...' and a magnifying glass icon. Below the search bar is a dropdown menu 'Show 10 entries'. To the right of the search bar is a blue 'Add' button. A modal window titled 'Update Beacon Direction' is open in the center. It has four dropdown menus: 'Previous' (set to 12), 'Current' (set to 11), 'Next' (set to 10), and 'Direction' (set to Straight). At the bottom of the modal are three buttons: 'Cancel', 'Delete' (highlighted in orange), and 'Save'. Below the modal is a table with three rows of data. The first row has columns: 8, 8, 4, 1, Straight, and a blue 'Update' button. The second row has columns: 9, 4, 1, 5, Right, and a blue 'Update' button. The third row has columns: 10, 5, 1, 4, Left, and a blue 'Update' button. At the bottom of the table are links for 'Showing 1 to 10 of 57 entries', 'Previous', page numbers 1, 2, 3, and 'Next'.

8	8	4	1	Straight	Update
9	4	1	5	Right	Update
10	5	1	4	Left	Update

- This screen allows an administrator to update a specific beacon direction.
- When the administrator clicks “Delete”, the beacon direction will be removed from the database.
- When the administrator clicks “Save”, any changes made to the beacon direction will be saved to the database and overwrite the old information for that particular beacon direction.



2.1.9 Beacon Relations

BRID	BeaconID	Related Beacons	Action
1	1	2,4	Update
2	2	1,5,3	Update
3	3	2,6	Update
4	4	1,5	Update
5	5	2,4,6,8	Update
6	6	5,9	Update
7	7	8	Update
8	8	5,7	Update
9	9	6,10	Update
10	10	9	Update

- This screen allows an administrator to view the relationships between each beacon. All these information will be retrieved from the database.
- When the administrator clicks on the “Add” button at the top right, he will be brought to the add beacon relation screen.
- When the administrator clicks on the “Update” button under the action column, he will be brought to the update beacon relation screen.

2.1.10 Add Beacon Relation

The screenshot shows the 'Administrator Portal' interface. On the left is a vertical navigation menu with options: Dashboard, Beacons, Beacon Directions, **Beacon Relations**, Units, Facilities, and Users. The 'Beacon Relations' option is currently selected. The main content area has a title 'Beacon Relations' with a search bar and a 'Show 10 entries' dropdown. A prominent 'Add' button is located in the top right of this section. A modal dialog box is open in the center, titled 'Add Beacon Relation'. It contains two input fields: 'Beacon ID' with the value '88' and 'Related Beacons' with the value '11'. Below these fields is a '+' button. At the bottom of the dialog are 'Cancel' and 'Add' buttons. In the background, there is a table with four columns and four rows of data. The columns are labeled with symbols: a circle, a square, a triangle, and a rectangle. The rows contain data such as (8, 8, 5,7), (9, 9, 6,10), (10, 10, 9), and (11, 11, 11). Each row has an 'Update' button in its last column. At the bottom of the page, there is a footer with links: Home, Help, Contact, and Log Out.

○	□	△	□
8	8	5,7	Update
9	9	6,10	Update
10	10	9	Update
11	11	11	Update

- This screen allows an administrator to add in a beacon relation. The administrator will have to enter the Beacon ID as well as add in the related Beacon ID(s) to that particular Beacon ID.
- When the “+” button is clicked after entering a related Beacon ID, the field will be cleared and a tag with the Beacon ID will be added below.
- All fields will have input validation and must be filled. If a field is not entered, it will be indicated to the admin to fill in the empty fields.
- Once the administrator clicks “Add” the information will be saved into the database.

2.1.11 Update Beacon Relation

The screenshot shows the Administrator Portal interface. On the left is a vertical sidebar with navigation links: Dashboard, Beacons, Beacon Directions, **Beacon Relations**, Units, Facilities, and Users. The 'Beacon Relations' link is highlighted. The main content area has a header 'Beacon Relations' with a search bar and a dropdown 'Show 10 entries'. A large central modal window is open, titled 'Update Beacon Relation'. It contains fields for 'Beacon ID' (set to 12) and 'Related Beacons' (set to 11). Below these are two small buttons: '22 x' and '24 x'. At the bottom of the modal are three buttons: 'Cancel', 'Delete' (in red), and 'Save' (in green). Below the modal is a table with four rows of data, each with 'Update' buttons. The table columns are labeled with question marks. The data rows are:

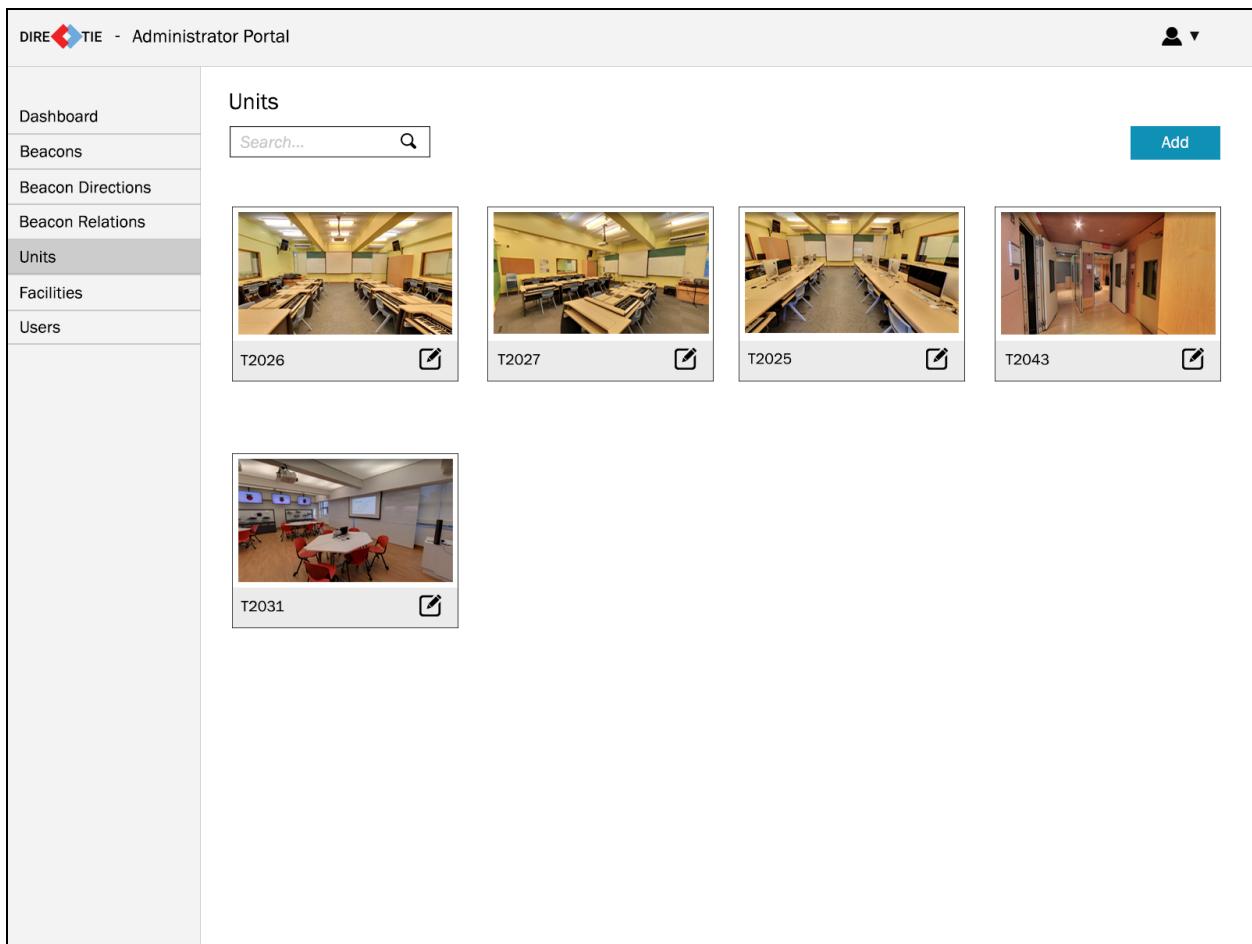
?	?	?	Update
8	8	5,7	Update
9	9	6,10	Update
10	10	9	Update

At the bottom of the page, there is a footer with the text 'Showing 1 to 10 of 57 entries' and a navigation bar with buttons for 'Previous', '1', '2', '3', and 'Next'.

- This screen allows an administrator to update a specific beacon relation.
- When the administrator clicks on the “x” beside the tag, it will remove that particular Beacon ID from the relation.
- When the administrator clicks “Delete”, the beacon relation will be removed from the database.
- When the administrator clicks “Save”, any changes made to the beacon relation will be saved to the database and overwrite the old information for that particular beacon relation.



2.1.12 Units



The screenshot shows the 'Administrator Portal' interface. On the left is a vertical sidebar with navigation links: Dashboard, Beacons, Beacon Directions, Beacon Relations, Units (which is highlighted in grey), Facilities, and Users. The main area is titled 'Units' and contains a search bar with placeholder text 'Search...' and a magnifying glass icon. To the right of the search bar is a blue 'Add' button with a white pencil icon. Below the search bar are five cards, each representing a unit:

- T2026**: Shows a classroom with rows of desks and chairs.
- T2027**: Shows a classroom with desks and chairs.
- T2025**: Shows a room with long tables and computer monitors.
- T2043**: Shows a hallway or room with wooden doors.
- T2031**: Shows a room with a round table and red chairs.

Each card has a small edit icon (pencil) in the bottom right corner.

- This screen allows an administrator to view the units that are available.
- When the administrator clicks on the “Add” button at the top right, he will be brought to the add unit screen.
- There is an edit icon beside the unit number and when the administrator clicks on it, he will be brought to the update unit screen.



2.1.13 Add Unit

DIRE[◆]TIE - Administrator Portal

Units

Add Unit



t2139.png

Block Level

Unit No

T2026 T2031 T2043

- This screen allows an administrator to add in a unit. The administrator will can upload an image by clicking “browse” and select an image file from his computer and is required to provide information regarding the Block number, Level of the unit and Unit number.
- All fields will have input validation and must be filled. If a field is not entered, it will be indicated to the admin to fill in the empty fields.
- Once the administrator clicks “Add” the information will be saved into the database.



2.1.14 Update Unit

DIRECtie - Administrator Portal

Dashboard Beacons Beacon Directions Beacon Relations Units Facilities Users

Units

Search...

Add

T2026

T2031

T2043

Update Unit

t2139updated.png

Block Level

Unit No

- This screen allows an administrator to update a unit.
- When the administrator clicks “Delete”, the information pertaining to the unit will be removed from the database.
- When the administrator clicks “Save”, any changes made to the unit will be saved to the database and overwrite the old information for that particular unit.



2.1.15 Facilities

The screenshot shows the 'Facilities' section of the DIRE TIE Administrator Portal. On the left is a vertical sidebar with links: Dashboard, Beacons, Beacon Directions, Beacon Relations, Units, Facilities (which is selected and highlighted in grey), and Users. The main area has a header 'Facilities' with a search bar and an 'Add' button. Below are five facility cards:

- Hilltop Library: A room with rows of seating and a large screen.
- Cyber Wargame Center: A room with several computer workstations.
- Motion Capture Studio: A room with multiple cameras and computer monitors.
- Visual Effects Studio: A room with a green screen backdrop and equipment.
- M.A.D Studios: An office space with desks and computers.

- This screen allows an administrator to view the facilities that are available.
- When the administrator clicks on the “Add” button at the top right, he will be brought to the add facility screen.
- There is an edit icon beside the facility name and when the administrator clicks on it, he will be brought to the update facility screen.



2.1.16 Add Facility

Screenshot of the "Add Facility" form in the Administrator Portal.

The form includes:

- Image Input:** A file input field labeled "hilltoplibrary.png" with a "Browse" button.
- Facility Name:** An input field containing "Hilltop Library".
- Description:** A text area containing the following text:

The Hilltop Library is a branch library located next to Teaching Block 22. It serves the SP Business School, the School of Communication, Arts and Social Sciences, and the School of Digital Media ,and Infocomm Technology. Surrounded by pleasant greenery of trees and plants
- Buttons:** "Cancel" and "Add" buttons at the bottom right.

In the background, there is a list of facilities including "Visual Effects Studio" with an edit icon.

- This screen allows an administrator to add in a facility. The administrator can upload an image by clicking “browse” and select an image file from his computer and is required to provide information regarding the facility name and description of the facility.
- All fields will have input validation and must be filled. If a field is not entered, it will be indicated to the admin to fill in the empty fields.
- Once the administrator clicks “Add” the information will be saved into the database.



2.1.17 Update Facility

Screenshot of the DIRECtie Administrator Portal showing the "Update Facility" screen.

The main interface shows a sidebar with navigation links: Dashboard, Beacons, Beacon Directions, Beacon Relations, Units, Facilities (selected), and Users. A central content area displays a modal window titled "Update Facility".

Inside the modal:

- A preview image of a building with the words "DREAMS" and "STORIES" on its facade.
- An input field labeled "updatedHilltopLibrary.png" with a "Browse" button.
- A "Facility Name" input field containing "Hilltop Library".
- A "Description" text area containing the following text:

The Hilltop Library is a branch library located next to Teaching Block 22. It serves the SP Business School, the School of Communication, Arts and Social Sciences, and the School of Digital Media ,and Infocomm Technology. Surrounded by pleasant greenery of trees and plants
- Action buttons at the bottom: "Cancel", "Delete" (disabled), and "Save".

In the background, other facility cards are visible, such as "Visual Effects Studio" with an edit icon.

- This screen allows an administrator to update a unit.
- When the administrator clicks “Delete”, the information pertaining to the facility will be removed from the database.
- When the administrator clicks “Save”, any changes made to the facility will be saved to the database and overwrite the old information for that particular facility.



2.1.18 Users

UserID	Role	Email	Date Joined	Action
1	Super Admin	abcd@gmail.com	1-1-2018	<button>Update</button>
2	Super Admin	aaaa@gmail.com	1-1-2018	<button>Update</button>
3	Administrator	abcdef@hotmail.com	2-1-2018	<button>Update</button>
4	Member	abc123@gmail.com	5-1-2018	<button>Update</button>
5	Member	xyz123@gmail.com	5-1-2018	<button>Update</button>
6	Member	qwert@hotmai.com	11-1-2018	<button>Update</button>
7	Member	testing@hotmail.com	15-1-2018	<button>Update</button>
8	Member	testing1234@gmail.com	28-1-2018	<button>Update</button>
9	Member	bbbb@hotmail.com	3-2-2018	<button>Update</button>
10	Member	cccc@gmail.com	9-2-2018	<button>Update</button>

- This screen allows an administrator to view the facilities that are available.
- When the administrator clicks on the “Update” button under the action column, he will be brought to the update user screen.



2.1.19 Update User

The screenshot shows the 'Administrator Portal' interface. On the left is a vertical sidebar with navigation links: Dashboard, Beacons, Beacon Directions, Beacon Relations, Units, Facilities, and Users (which is currently selected). The main area is titled 'Users' and contains a search bar and a table with 10 entries per page. A modal window titled 'Update User' is open over the table. The modal has fields for 'UserID' (set to 3), 'Email' (set to abc123@gmail.com), and 'Role' (set to Member). At the bottom of the modal are three buttons: 'Cancel', 'Delete' (disabled), and 'Save'. The table in the background lists users from 1 to 10, each with a 'Role' column showing 'Member' and an 'Action' column with a 'Update' button.

User ID	Role	Email	Date	Action
1	Member	testing@hotmail.com	15-1-2018	Update
2	Member	testing1234@gmail.com	28-1-2018	Update
3	Member	bbbb@hotmail.com	3-2-2018	Update
4	Member	cccc@gmail.com	9-2-2018	Update
5				Update
6				Update
7				Update
8				Update
9				Update
10				Update

- This screen allows an administrator to update a user's role.
- When the administrator clicks "Delete", the user will be removed from the database, essentially deleting the account.
- When the administrator clicks "Save", any changes made to the user's role will be saved to the database and overwrite the previous role for that particular user.



2.2 Mobile Application Wireframes

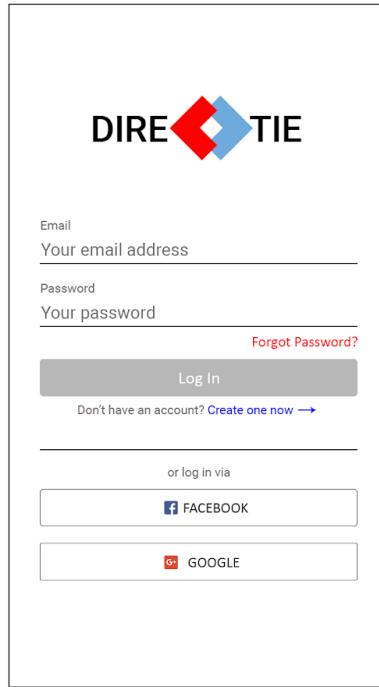
2.2.1 Splash Screen



- This is the splash screen of the mobile application which would be shown when the user launches the app.



2.2.2 Login



- This is the screen that will be shown after the splash screen if the user is not logged in. This will allow the user to login to the application to access the functions of the app.
- The user will be able to choose to either login via social media (Google/Facebook) or through a Directie account.
- If a user does not have a Directie account, he/she can choose to create one on this screen as well.



2.2.3 Registration

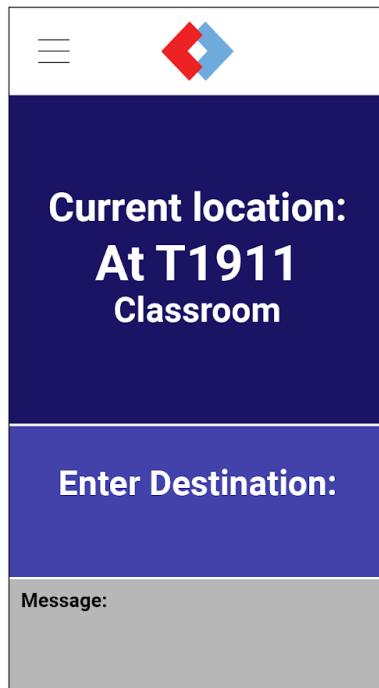
The image shows a registration form for a Directie account. At the top left is a red 'Cancel' button and a top right is a 'Register' button. Below these are five input fields: 'Name' (placeholder 'Your Name'), 'Password' (placeholder 'Your Password'), 'Email' (placeholder 'Your Email'), 'Gender' (dropdown menu showing 'Male'), and 'Date of Birth' (placeholder '01-01-1950'). The Directie logo, featuring the word 'DIRE' in black and 'TIE' in white with a red and blue diamond icon, is centered above the input fields.

Name	Your Name
Password	Your Password
Email	Your Email
Gender	Male ▾
Date of Birth	01-01-1950

- This is the screen that allows a user to create a Directie account. The user will be required to enter their email, password, name, gender and date of birth.
- All fields will have input validation and must be filled. If a field is not entered, it will be indicated to the user to fill in the empty fields.
- Once the user has entered all fields and taps on “Register” on the top right, the account will be created.



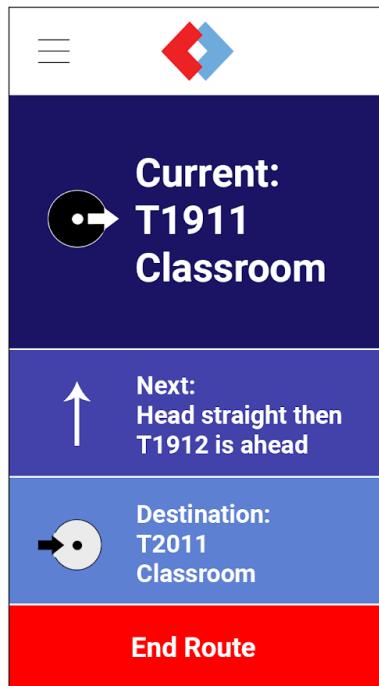
2.2.4 Home



- This is the screen that the user will be brought to after logging in. It will show and read out the user's current location automatically by detecting the nearest beacon to the user.
- The user will be able to enter his/her desired destination.



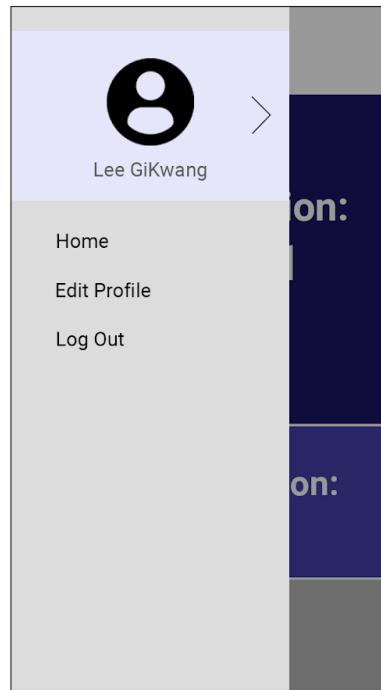
2.2.5 Navigation Page



- This is the screen that will be shown after the user has given a destination to navigate to.
- It will show the current location the user is at and the step to take to go to the next location.
- The user will be able to end the route by pressing the “End Route” button at the bottom.



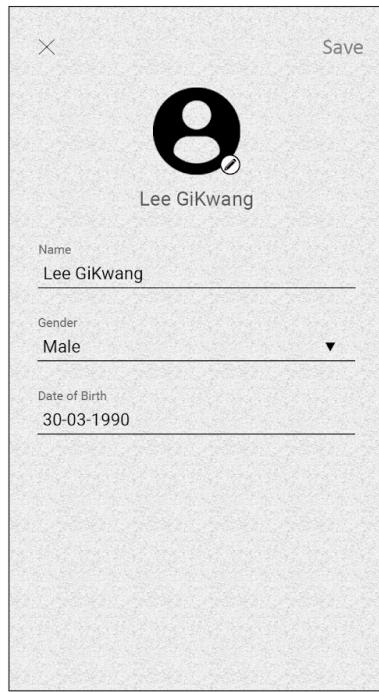
2.2.6 Side Menu



- This is the screen that will be shown when the user taps on the hamburger menu button.
- From here, the user will be able to go to the home screen, view and edit his/her profile and logout.



2.2.7 Edit Profile

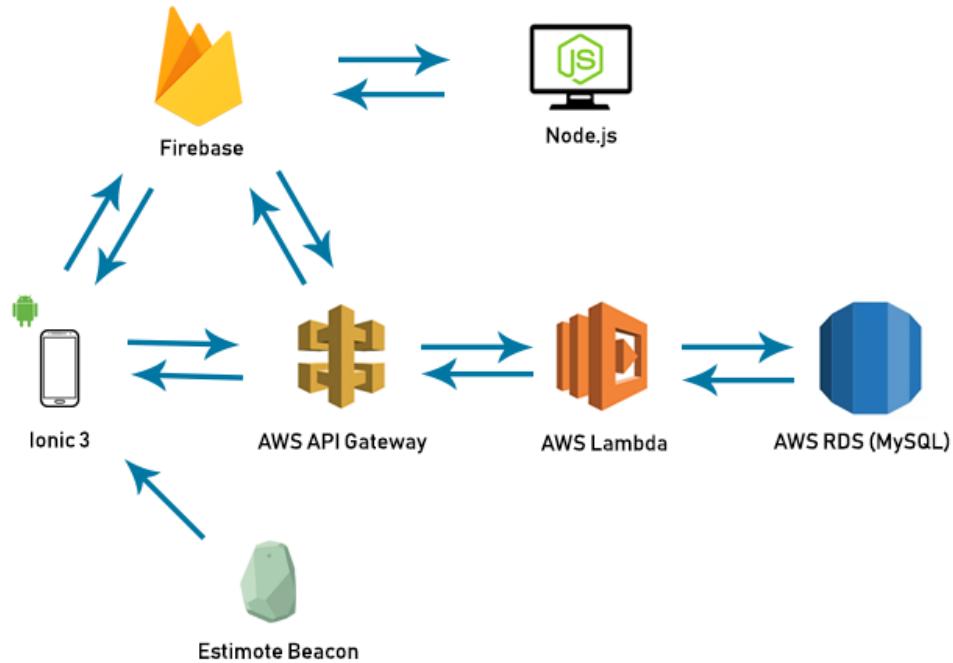


- This is the screen that will allow the user to edit his/her profile information.
- Information like name, gender and date of birth can be changed. The profile picture can also be changed as well.
- Once the user taps on “Save” on the top right, the information will be saved into the database.



3 Detailed Specifications

3.1 System Architecture



On the client-side mobile application, it uses the Ionic 3 Framework while the web application which is the administrator portal will be using Node js.

Both the mobile application and web application will use Firebase Authentication to authenticate users when logging in.

For the APIs of the mobile application and web application (admin portal), it will be through the AWS API Gateway which would allow our application to retrieve and put in data by calling the Api link generated by the AWS API Gateway from our back end service which will connect to the AWS Lambda. From there, the AWS Lambda will query the database depending on what the Api calls and retrieve the data from the database. Lambda will then pass the data back to the Api and then back to the mobile application.



3.2 Technologies Used

3.2.1 Ionic 3 Framework

The Ionic 3 Framework is a multi-platform open source SDK for mobile development that is built on Angular and uses common web technologies like CSS & HTML5. It will allow developers to build a highly interactive native mobile application as well as progressive web apps.

3.2.1.1 Why Ionic 3 Framework

For this project, we have chosen to work with the Ionic Framework initially as it allows us to build a Progressive Web Application (PWA) with it. However, as PWA is not ‘mature’ enough, it is unable to detect the distance of beacons and thus, instead of PWA, we have switched to building a mobile application.

We have decided to continue using the Ionic Framework due to it having multi-platform development support. Even though we are only focusing on developing for the Android Operating System for this project, it will make it much easier to have it expand on to iOS in the future if needed.

3.2.2 Cloud Computing

Cloud service platforms offers computing power, database storage and many other functionalities. With cloud computing, it is very flexible, allowing us to scale up or down depending on our needs within a moment’s notice. In addition to that, since it is cloud computing, it means that servers are off premise. The supplier of the cloud service will maintain it while we can focus on the project.

3.2.2.1 Why Amazon Web Services (AWS)

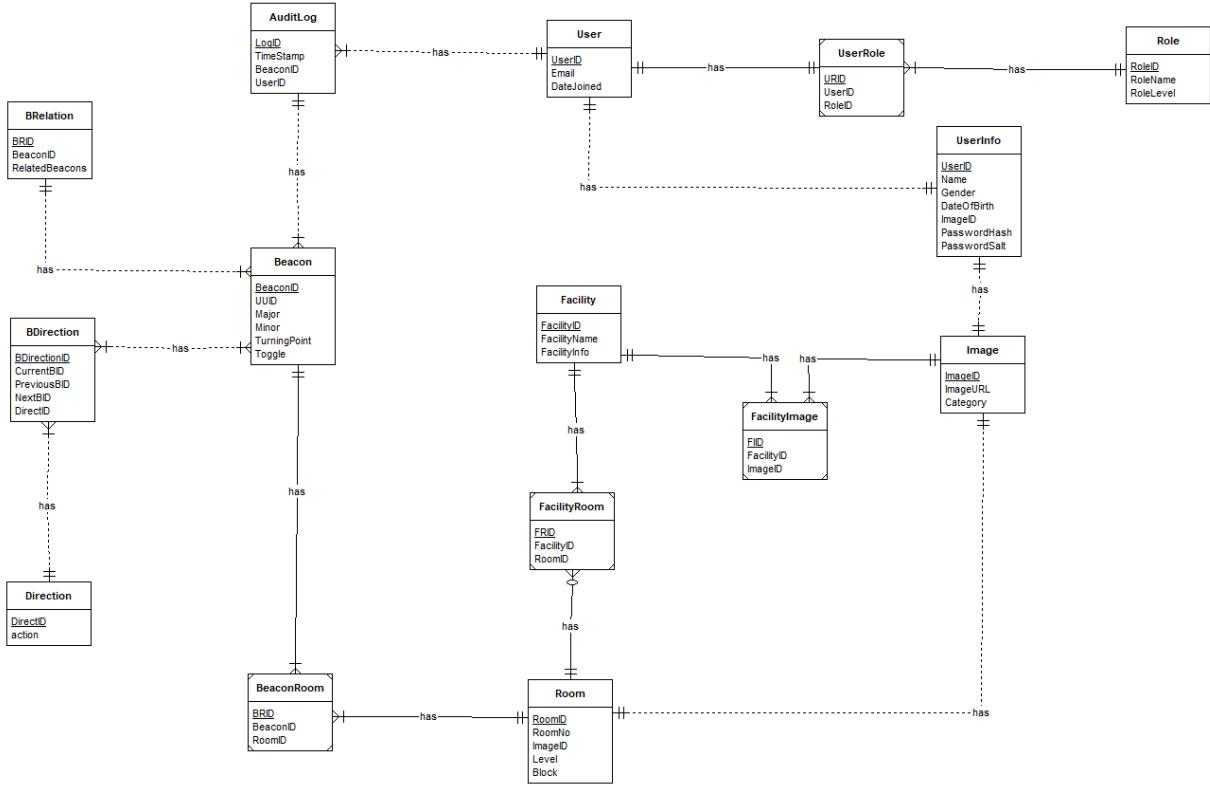
One reason why we are using Amazon Web Services (AWS) as our cloud service platform is due to the cost efficiency. By using AWS API to AWS Lambda and then towards the RDS, it avoids using any server instance which helps in lowering costs. In addition, AWS replaces large upfront expenses with low variable payments that only apply to what we use. This means that we will only need to pay for services that we use from AWS.

In addition to the lower costs, security is also part of the reason why we chose to use AWS. Amazon Web Service’s data center and network architecture is built to meet the requirements of most security-sensitive organizations. Network firewalls are built into Amazon Virtual Private Cloud (VPC) which allows us to create an isolated section of the AWS cloud where we can launch AWS resources in a virtual network defined by us.

AWS also provides scalability and high-performance by using AWS tools which provides Elastic Load Balancing and Auto Scaling, thus giving developers the option to scale up or down depending on their application’s need. It is extremely easy to use for the developers to configure and set up their applications by using AWS Management Console or well-documented web services APIs.



3.3 Database Design



3.3.1 User

The User table is used to store users of the application. UserID is a surrogate key that is automatically generated by the database and it is used as the Primary Key of the table. The Email attribute is used to store the email that the user signed up with. Lastly, the DateJoined attribute keeps the date of when the user has registered the account.

A user record will have a UserInfo that keeps information of the user as well as a UserRole. Each user will also have one or many audit logs.

3.3.2 UserInfo

The UserInfo table is used to store information of users. UserID is a foreign key from the User table and it is used as a primary key in this table. Name is used to store the user's name. Gender is used to store the gender that the user has selected. DateOfBirth is used to store the user's date of birth. PasswordHash is used to store the password hash and PasswordSalt is used to store the password salt. Both the password hash and salt will be used for the user to login to his/her account. ImageID is a foreign key from the Image table which allows us to link to it and get the profile picture.

A UserInfo record must be related to 1 user and it must have 1 image.



3.3.3 Role

The Role table contains information of the different roles that can be set for users. RoleID is a surrogate key that is automatically generated by the database and it is used as the primary key of the table. The RoleName attribute is used to store the name of the role. RoleLevel attribute stores the level of the role in integer. For example, '1' for member and '3' for super admin.

One role can have multiple records in the UserRole table

3.3.4 UserRole

The UserRole is a table connecting the User table and Role table which allows us to track the associations between user and role. The URID is a surrogate key that is automatically generated by the database and it is used as the primary key of the table. The UserID and RoleID are foreign keys that connects the User table and Role table.

3.3.5 AuditLog

The AuditLog table is used to store logs that are created everytime a user enters into the range of a different beacon. LogID is a surrogate key that is automatically generated by the database and it is used as the primary key of the table. The TimeStamp attribute is used to store the time at which the user enters the range of a specific beacon. Both the UserID and BeaconID are foreign keys which comes from the User table and Beacon table respectively. This is to allow us to know which user has been at which beacon at what time.

3.3.6 Beacon

The Beacon table is used to contain information about every beacon used by our application. The BeaconID is used as the primary key of the table. The UUID stores an identifier that is 32 alphanumeric characters long. Both the Major and Minor attributes will store a 3 digit integer. Through the combination of the UUID, Major and Minor for each beacon, we will be able to distinguish one beacon from another. The TurningPoint attribute will store a bool to indicate if the beacon is a turning point or not. The Toggle attribute will also store a bool to indicate if the beacon is enabled or disabled.

A beacon must be related to at least 1 other beacon and it must have a direction. A beacon can be used for more than 1 room as well.



3.3.7 BRelation

The BRelation table stores the relationships between different beacons. Every relation in this table will be pre defined. The BRID is a surrogate key that is automatically generated by the database and it is used as the primary key of the table. BeaconID is a foreign key which links it to the Beacon table. The RelatedBeacons attribute is also a foreign key that contains another related beacon's BeaconID. With these 2 attributes, it allows us to know which particular beacon is related to which to make navigation more accurate.

3.3.8 Direction

The Direction table contains information of the different directions that can be possible to take. DirectID is a surrogate key that is automatically generated by the database and it is used as the primary key of the table. The action attribute is used to store the action the user has to take like "go straight".

A direction can be used in more than one beacon direction.

3.3.9 BDirection

BDirection is a table connecting the Beacon table and Direction table which allows us to define the directions needed to go from 1 beacon to the other. BDirectionID is a surrogate key that is automatically generated by the database and it is used as the primary key of the table. The attributes PreviousBID, CurrentBID and NextBID are used to store the beacon IDs of the previous, current and next beacon respectively. The DirectID is used to get a particular direction from the Direction table. These attributes will be pre defined/set to let the application know what directions it should take when going from 1 beacon to the other beacon.

3.3.10 Room

The Room table stores all the rooms that will be in the location. RoomID is a surrogate key that is automatically generated by the database and it is used as the primary key of the table. RoomNo stores the number of the room like T2139. ImageID is a foreign key from Image table to get the associated image for the room. Level is to store the floor at which the room is and the Block attribute is for which block the room is at like T21. This is to make giving instructions to navigate to the room easier as the floor and block of the room is kept.

A room will be associated with at least 1 beacon. A room can have 1 image and a room may or may not be a facility.



3.3.11 BeaconRoom

BeaconRoom is a table connecting the Beacon table and Room table. BRID is a surrogate key that is automatically generated by the database and it is used as the primary key of the table. BeaconID and RoomID are foreign keys from the Beacon table and Room table respectively. This will allow us to keep track of which rooms has which beacons as a room can have more than 1 beacon and 1 beacon can be used in multiple rooms.

3.3.12 Facility

The Facility table stores all facilities in the location. The FacilityID is a surrogate key that is automatically generated by the database and it is used as the primary key of the table. FacilityName is used to store the name of the facility. FacilityInfo is used to store additional information like operating hours about the facility as well as the description of the facility.

A facility must have an image and a facility must be associated with at least 1 room.

3.3.13 FacilityRoom

FacilityRoom is a table connecting the Facility table and Room table so as to make the Room table flexible and only store information related to room. FRID is a surrogate key that is automatically generated by the database and it is used as the primary key of the table. FacilityID and RoomID are foreign keys from the Facility table and Room table respectively. This table will make it easier to manage the relationship between facility and rooms.

3.3.14 Image

The Image table is used to store images used in the application like profile pictures, facility images and room images. ImageID is a surrogate key that is automatically generated by the database and it is used as the primary key of the table. ImageURL is used to store the url for the image stored in the system to allow us to get the image. Category is used to store the category of which the image is used for like facility or room organise the data kept.

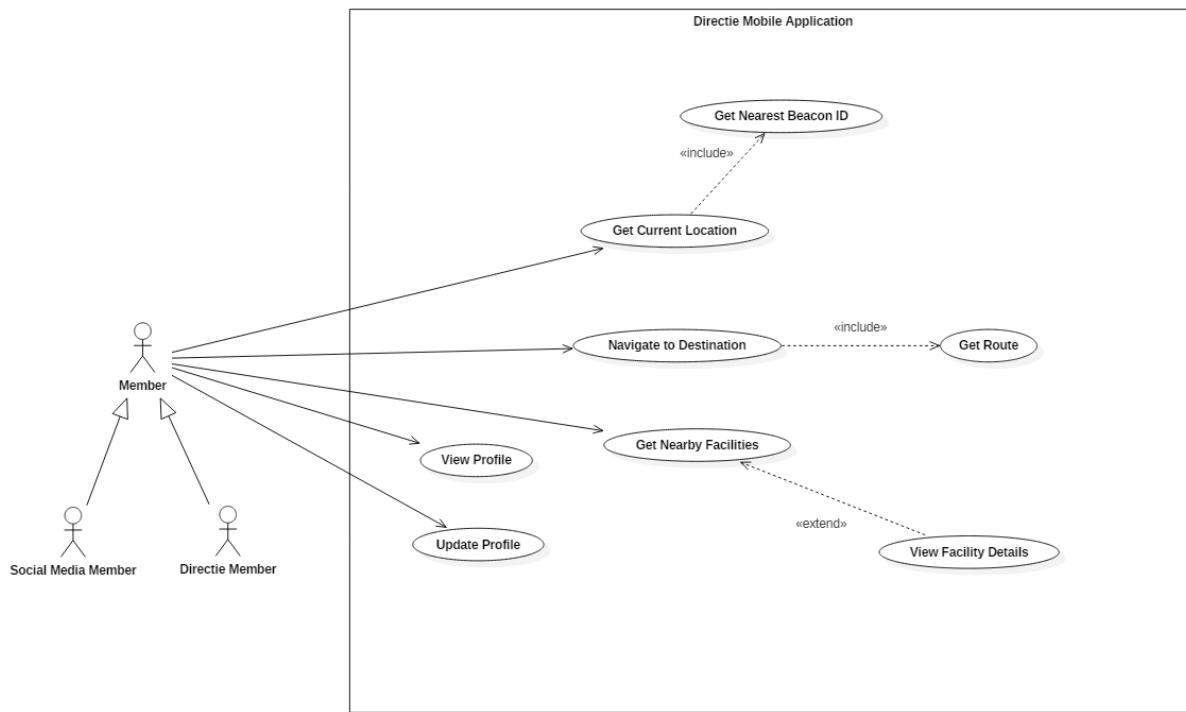
3.3.15 FacilityImage

FacilityImage is a table connecting the Facility table and Image table so as to make the Facility table flexible and only store information related to facility. FIID is a surrogate key that is automatically generated by the database and it is used as the primary key of the table. FacilityID and ImageID are foreign keys from the Facility table and Image table respectively get a specific image for the facility.



4 Unified Modeling Language (UML) Diagrams

4.1 Front-End Use Case



Front-End Use Case Diagram

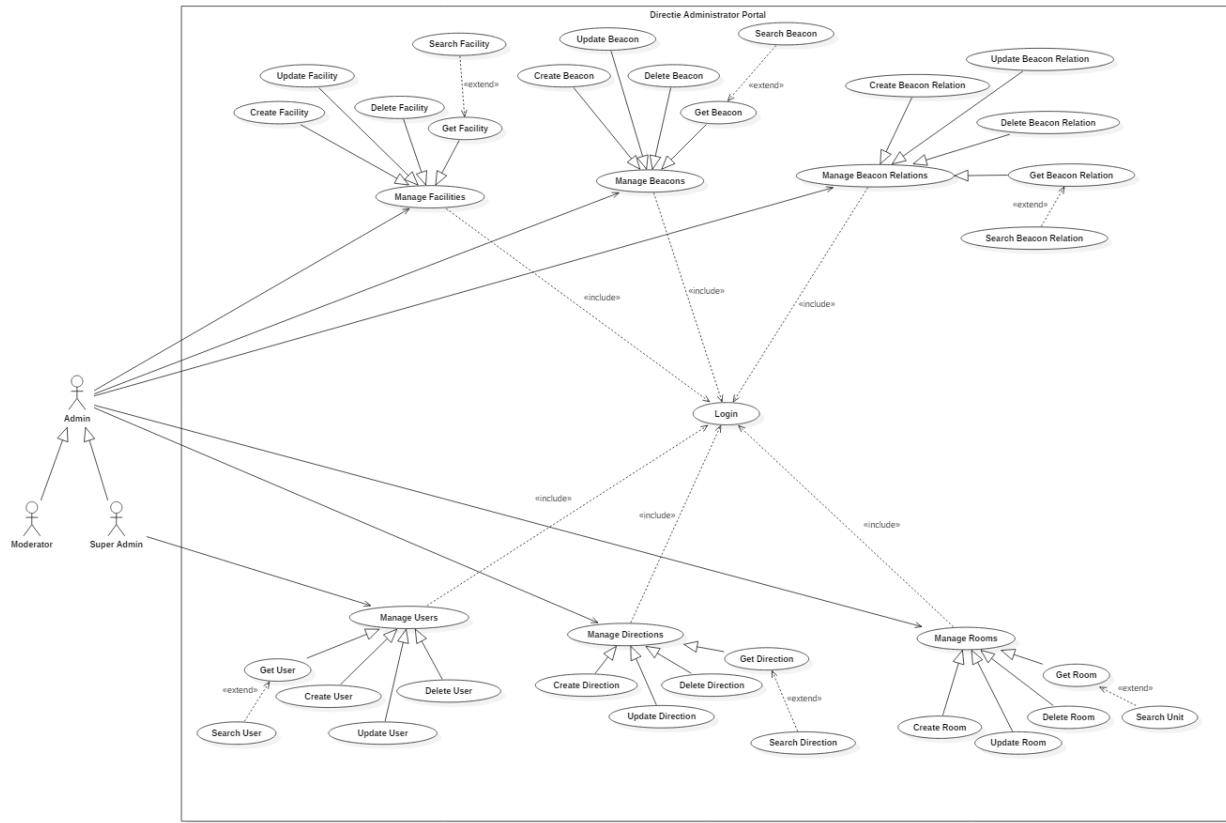
The actor of this use case is User which can either be a member that logged in through their social media account to use our service or a member that has registered an account with us.

A User will be able to know their current location as the app will detect the nearest beacon and indicate it to the user where he/she is at. The User can also get the shortest route to his desired destination.

The user will also be able to view his/her profile and update profile information.



4.2 Back-End Use Case



Back-End Use Case Diagram

The 2 actors of this use case “Moderator” and “Super Admin” are generalized under Admin.

A Moderator will be able to View, Create, Update & Delete facilities, beacons, beacon relations, beacon directions and rooms.

A Super Admin has the ability to perform every function a Moderator can as well as to View, Create, Update & Delete users.

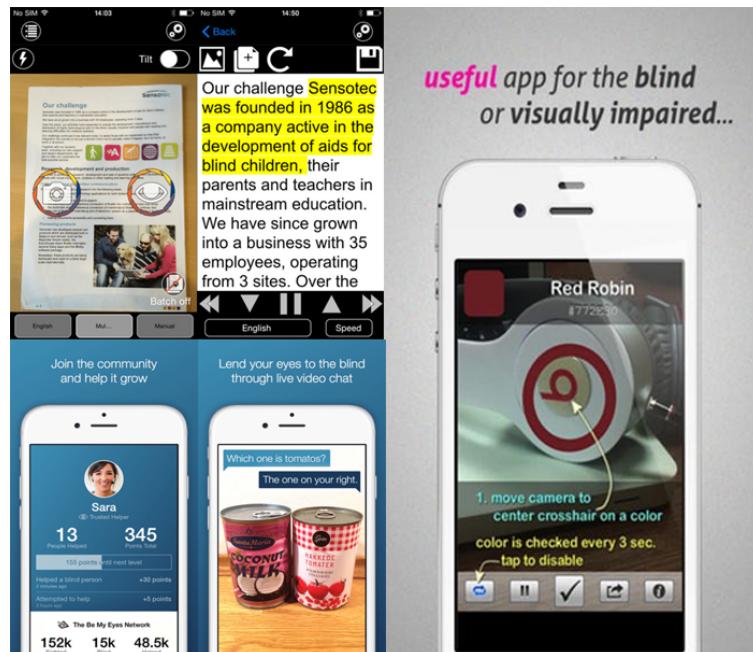
To be able to perform any of the functions in the administrator web portal, the admin has to login with an admin account.

Appendix A - Research

A.1 Objectives

Through this research, we aim to understand the flow of the VI application and how the User Interface (UI) is being properly structured and designed.

In the Google Play Store/Apple App Store, there are only limited number of applications designed for the VI. Some of the applications can help to read out physical text from books and newspapers, determine the color of an physical object, or help to identify an object through recognition/online volunteer.



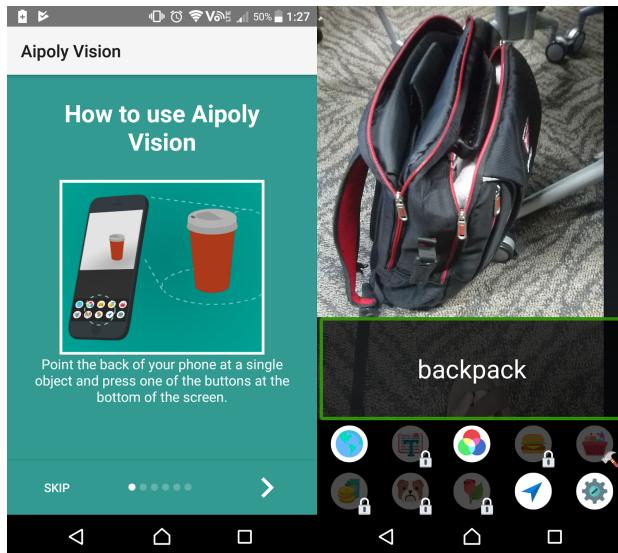
Screenshots of some applications designed for VIs

These applications are usually accompanied with Google's TalkBack or Apple's VoiceOver, as every text is read out based on what they have tapped on. In this research, we are going to look on the two popular apps for VI: Airpoly Vision and Be My Eyes.

A.2 Aipoly Vision



Aipoly Vision is an application where they can recognize and voice out an object or something more precise like a plant, animal, food or a colour by using a camera.



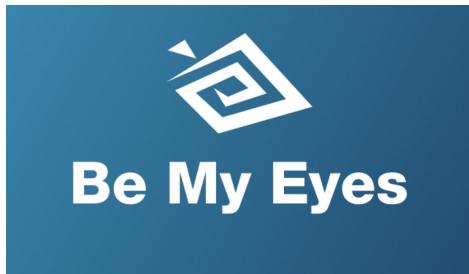
Screenshot of Aipoly Vision App

When you first start the application, instructions are shown on how to use it. If TalkBack/VoiceOver is enabled, the instructions can be read by tapping on the words, including the 'next' symbol. After the instructions have been completed, it will not display for every time you startup the application.

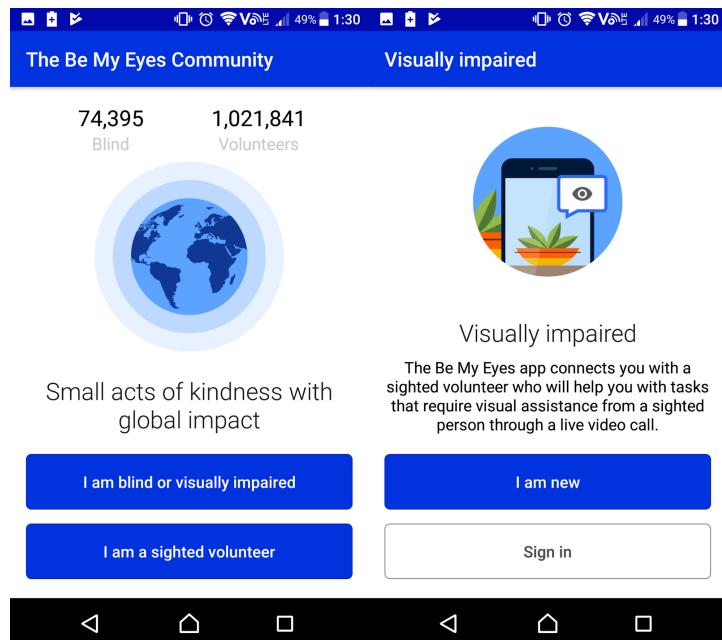
The application requires you to allow your permission to use the rear camera, after which you are free to use your camera and identify an object. When they tap the object on the camera screen, it will read out the name of the object e.g. Backpack, laptop. The icons below show the type of object the user wants to know.



A.3 Be My Eyes



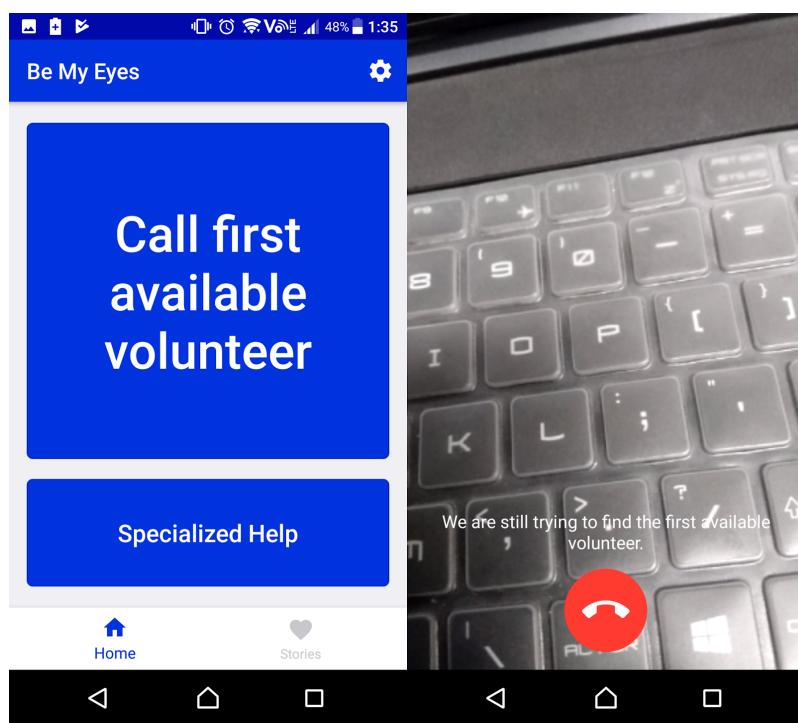
Be My Eyes is an application that connects VI to volunteers online through live video calls. From there, VI users can ask a volunteer to identify what object is it, how many objects are there, etc.



Screenshot of Be My Eyes application

When you first start the application, it will prompt you to agree the Privacy Policy and Terms of Service. Subsequently, you are required to choose whether you are a VI or volunteer, and also sign up an account. In this case, I will choose as a VI user.

The process is much longer than Airpoly Vision's first time setup.



Screenshot of Be My Eyes application

When you are done verifying your email, you are able to call a volunteer. Calling a volunteer will turn on the rear camera and the application will look for an active volunteer.

A.4 Conclusion

Through the research on the two applications, we can deduce that for the first time setup, it is usually accompanied with someone who is not VI, and they will help them to register an account and provide instructions on how to use it. Although it is possible for a VI to do it alone, it will take huge amounts of effort and time. For instance, the Be My Eyes application requires you to comply the terms, sign up an account and verify your email, which is a long process.

Secondly, the design of the application is clean and simple as there are not many components on every main page. The details on every page are less and short, which is easy for the VI to understand when they tap on any text.



Appendix B - Meeting Minutes

Meeting Minute I

Date : 11 April 2018
Time : 2:00 PM - 4:30 PM
Venue : DMIT Meeting Room 1

Present : Chua Wei, Joshua, Wei Sern, Garick

Minuted by : Wei Sern

Vetted by : Chua Wei

	Item Discussed	Action By
1.	Things done <ul style="list-style-type: none">● Introduction of the project● Aims and objectives of the progressive web application● Brief discussion on how to detect the user's location using beacons<ul style="list-style-type: none">○ Will have to keep track of current and past beacons as well to give accurate directions to the destination● Floor plans to be used in SVG format	
2.	Technologies to be used <ul style="list-style-type: none">● AWS● Firebase Authentication● Given a choice of using Angular or Ionic	



Meeting Minute II

Date : 18 April 2018
Time : 2:00 PM - 4:00 PM
Venue : DMIT Conference Room (T1942)

Present : Chua Wei, Joshua, Wei Sern, Garick, Hong Yao

Minuted by : Wei Sern

Vetted by : Chua Wei

	Item Discussed	Action By
1.	Things done <ul style="list-style-type: none">● Came out with group name● Brief discussion on the database design	
2.	Requirements Model <ul style="list-style-type: none">● User Story<ul style="list-style-type: none">○ Craft a scenario where a person uses the app● Target Audience● Problem Statement● Client Requirements● Critical Success Factor<ul style="list-style-type: none">○ With the app, the simplicity and ease of use can let everyone easily pick up and use it	Wei Sern



3.

Product Backlog

DATE	TIME	Product Backlogs
① Chen Wei		① ERD] 3,4
② Joshua		③ Use Case] 3,4
③ Hong Xao		④ Login with FB/Google 2
④ Wei-Sern		⑤ Design logo] 5
⑤ Granick		⑥ Design UI.] 5
		⑦ Provider in Ionic. 1,2
		⑧ AWS - Lambda - API] 1,3
		⑨ Setup DB & MySQL] 1,3
		⑩ Registration Page 2





Meeting Minute III

Date : 26 April 2018
Time : 2:00 PM - 4:30 PM
Venue : DMIT Meeting Room 1

Present : Chua Wei, Joshua, Hong Yao, Wei Sern, Garick

Minuted by : Wei Sern

Vetted by : Chua Wei

	Item Discussed	Action By
1.	Use Case Diagrams <u>Front-End</u> There should be the use case boundary Login not to be in the use case diagram. To show the main functions and how the function works Get current location (hilltop, business school, etc..) <u>Back-End</u> Only super admin to be able to access manage user Manage comments and manage users should have Create and Update as well Rename units to room to be more clear All plural (Beacons, Floor Maps)	Hong Yao Wei Sern
2.	Location Page If GPS located, skip location page and go to the map directly if the user is in the area of the location Users at home can tap on the different location and see the map. (Others are greyed out except hilltop for this project)	



3.	Main things to focus on 1. Navigate to destination and get route 2. Get current location 3. Get nearby facilities	Joshua
4.	Beacon Relation 8 4 5 1 3 2 6 [{4 : [{‘PB’: ‘8’, NB: ‘1’, ‘Dir’: ‘Straight’}, {‘PB’: ‘1’, NB: ‘8’, ‘Dir’: ‘Straight’}]}, {1 : [{ }]}, Current Beacon - Previous Beacon - Next Beacon - Direction 4 - 8 - 1 - go straight 4 - 1 - 8 - go straight 1 - 4 - 3 - turn left 1 - 2 - 5 - turn left Beacon 1 is a turning point Beacon 4 related to Beacon 3 (Direction -> go straight, turn left) Beacon 4 related to Beacon 5 (Direction -> go straight, turn right)	



5	<p>ERD Diagram</p> <p>2 facilities can be in 1 unit 2 units can be 1 facility</p> <p><u>FacilityUnit</u> Make unit table flexible, only store information related to unit. To link facility and unit together</p> <p><u>AuditLog</u> Should have beacon as well, get BeaconID to check this guy been through where at what time</p> <p><u>Facility</u> Include FacilityImage between Facility and Image</p> <p><u>Role</u> (RoleID, RoleName, RoleLevel) • role level (eg. disable 0, member 1 , admin 2, SA 3)</p> <p><u>UserRole</u> (UserID, RoleID)</p> <p><u>BRelation</u> (link to Beacon) CurrentBID, PreviousBID, NextBID, DirectID</p> <p><u>Location Table</u> (include this table)</p>	Hong Yao Wei Sern
6	<p>Design</p> <p><u>Logo</u> Not too long Remove shadow Should do in illustrator as vector</p> <p><u>From and Destination</u> Use drop down menu to choose to be more secure</p> <p><u>Floor Selection</u> Too dark Ability to hide/show it</p> <p>Note</p> <ul style="list-style-type: none">Indicate user's current location	Garick



7	Research on <ul style="list-style-type: none">● How to let the visually impaired know when to turn. (how to deliver message in what kind of manner)● Google speech to text	
8	Product Backlog <ol style="list-style-type: none">1. ERD, Use Case Diagram - 50%2. Login by Google/Facebook - 80% (bugs to be fixed)3. UI Design (more screens, entire flow)4. Provider in Ionic - 0%5. Setup DB (to be done first before AWS)6. AWS7. Registration (to include password hash)8. Login with manual flow9. Get nearest Beacon/Ownership10. Minutes11. Research on how to deliver the message to visually impaired12. Research on text ⇔ speech13. Display SVG Map	



Meeting Minute IV

Date : 3 May 2018
Time : 1:00 PM - 2:00 PM
Venue : DMIT Meeting Room 1

Present : Chua Wei, Joshua, Wei Sern, Garick, Hong Yao

Minuted by : Wei Sern

Vetted by : Chua Wei

	Item Discussed	Action By
1.	<p>IOS voice over Android voice assistant</p> <ul style="list-style-type: none">● Read out what's on the screen that's tapped● Can give instruction for text to speech to read out <p>More important to focus on important function Navigate within a small room first</p> <p>Current implementation there might be some difficulties to travel from 1 beacon to another beacon</p>	Joshua Chua Wei
2.	<p>UI to fit the VI more</p> <ul style="list-style-type: none">● Only current and next● Have more space● Buttons should not be too close together	Garick



3	<p>Is there a way to help people with VI to go to the bathroom and back</p> <p>Setup test case to navigate within a room.</p> <ul style="list-style-type: none">● Inside a room put more beacons● Different room layouts● Consider about obstacles (tables, chairs, etc..)● Different obstacle placements● Different routes in the room● Indicate a beacon to be door● Some may have multiple doors	Wei Sern
4	<p>Apps that are targeted to VI</p> <ul style="list-style-type: none">● Be my eyes	Garick
5	<p>W3C standards design for VI</p> <p>When to give feedback/instructions on where they are</p> <ul style="list-style-type: none">● Beacon based● Time based● Error based● Positive reinforcement	Joshua



Appendix C - Product Backlog

Priority	Tasks Description	Estimated Effort (manday)
5	As a user, I want to register an account in the application	3
6	As a user, I want to login in the application	5
8	As a user, I want to login with an external provider authentication (Facebook, Google)	7
7	As a user, I want to edit my profile	3
1	As a user, I want to know my current location	7
3	As a user, I want to communicate with the application using Speech Recognition	14
2	As a user, I want to be able to navigate from my current location to a destination	30
9	As an administrator, I want to be able to login to the admin portal	3
4	As an administrator, I want to perform CRUD operations on Location, Facility, Unit, Beacon and Floor Plan	14
6	As an administrator, I want to view statistics of a user or unit	14



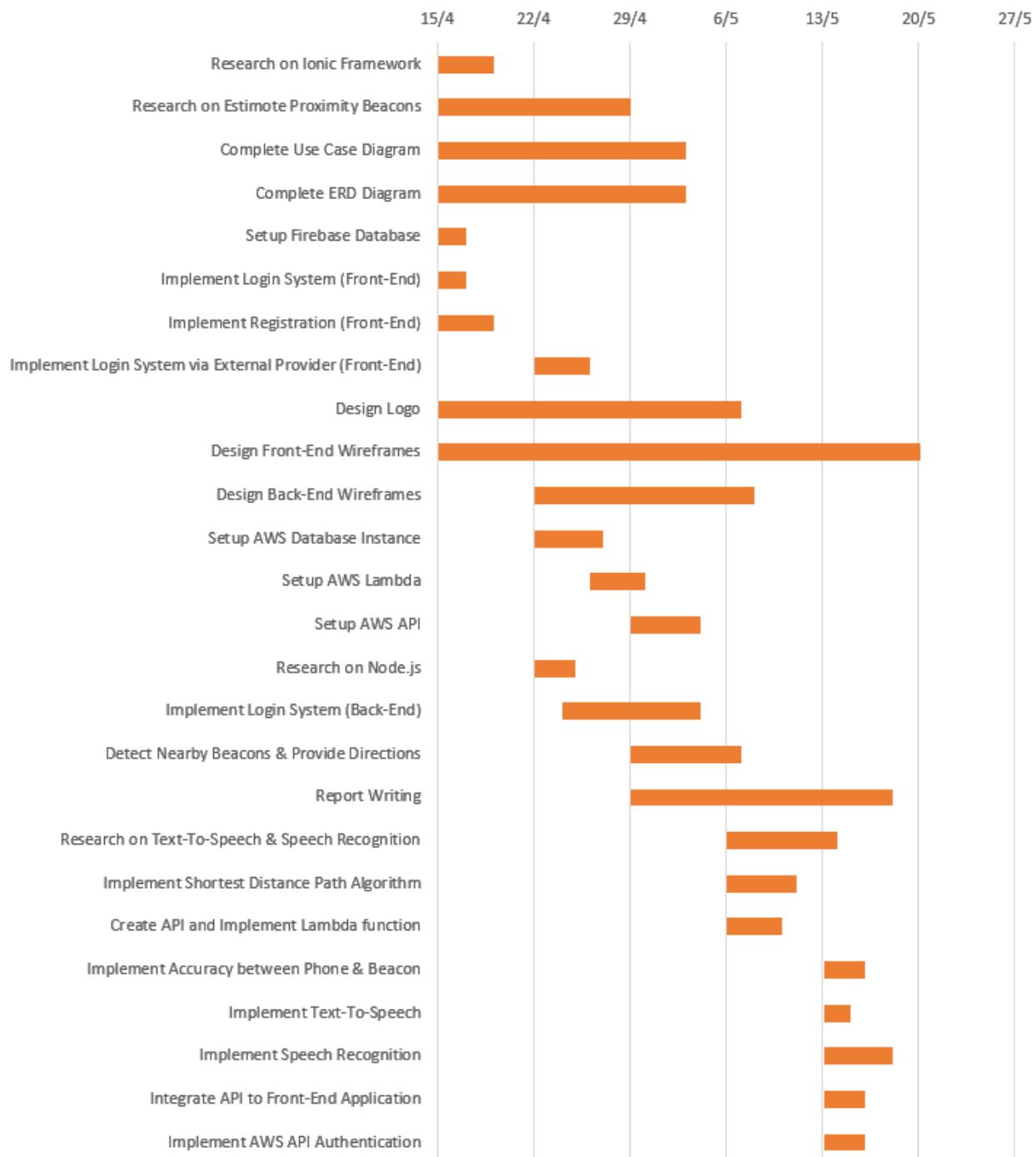
Appendix D - Sprint Backlog

Sprint Backlog 1 - Week 1 to 3

Tasks Description	Estimated Effort (manday)
Research on Ionic Framework	7
Research on Estimote Proximity Beacons	7
Complete the Use Case Diagram	3
Complete the ERD Diagram	3
Setup Firebase Database	5
Implement Login System for Front-End	5
Implement Registration Page for Front-End	3
Implement Login System via external provider (Facebook, Google)	5
Design Logo	5
Design Front-End UI	7
Design Back-End UI	5
Research on AWS Database & Lambda	7
Setup AWS Database instance (MySQL Database)	2
Setup AWS Lambda	7
Setup AWS API	7
Research on Node.js	3
Implement Login System for Back-End	5
Detect Nearby Beacons & Provide Directions to next Beacon	7



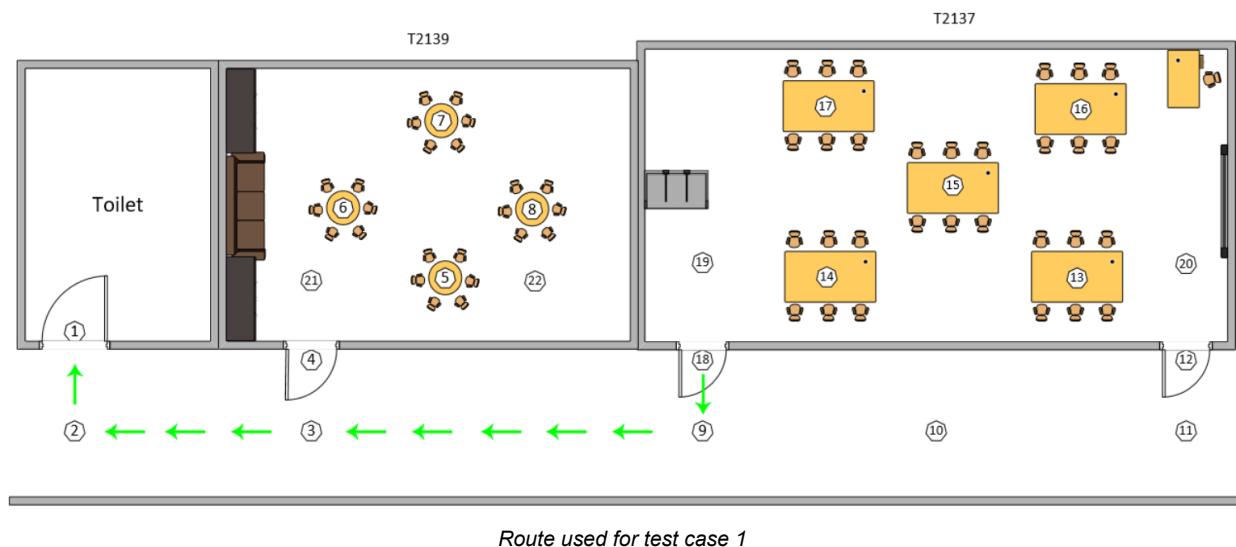
Appendix E - Project Plan





Appendix F - Test Cases

Test Case 1



Test Execution Date: 17/05/2018

Test Duration: 1 minute 4 seconds

Pre-Conditions:

- User is not blindfolded.
- User must be logged in to an account.
- Bluetooth and Microphone must be granted permission on the phone.

Description: Navigation from beacon 18 which is the backdoor for T2137 to beacon 1 which is the toilet.

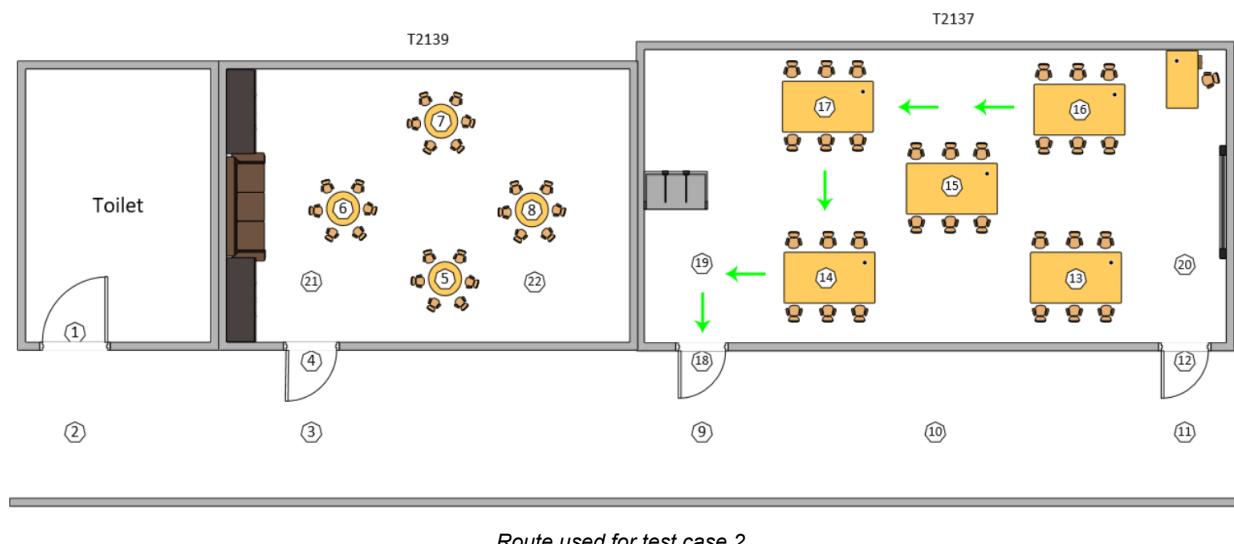
Step	Test Steps	Expected Results	Actual Results	Status (Pass/Fail)
1	User starts from beacon 18 and wants to navigate to beacon 9.	A voice message will tell the user to walk straight from beacon 18 to 9. It will also tell the user that beacon 9 is a turning point. User will then reach beacon 9 successfully.	A voice message containing the step was read out to the user, helping him to navigate from beacon 18 to 9. User successfully navigated to beacon 9.	Pass



2	User navigates from beacon 9 to beacon 3.	A voice message will tell the user to turn right and walk straight from beacon 9 to beacon 3. User will reach beacon 3 successfully.	A voice message containing the steps was read out to the user, helping him to navigate from beacon 9 to 3. User successfully navigated to beacon 3.	Pass
3	User navigates from beacon 3 to beacon 2.	A voice message will tell the user to walk straight from beacon 3 to 2. It will also tell the user that beacon 2 is a turning point. User will reach beacon 2 successfully.	A voice message containing the step was read out to the user, helping him to navigate from beacon 3 to 2. User successfully navigated to beacon 2.	Pass
4	User navigates from beacon 2 to beacon 1.	A voice message will tell the user to turn right and walk straight from beacon 2 to 1. User will reach beacon 1 successfully.	A voice message containing the steps was read out to the user, helping him to navigate from beacon 2 to 1. User successfully navigated to beacon 1.	Pass



Test Case 2



Route used for test case 2

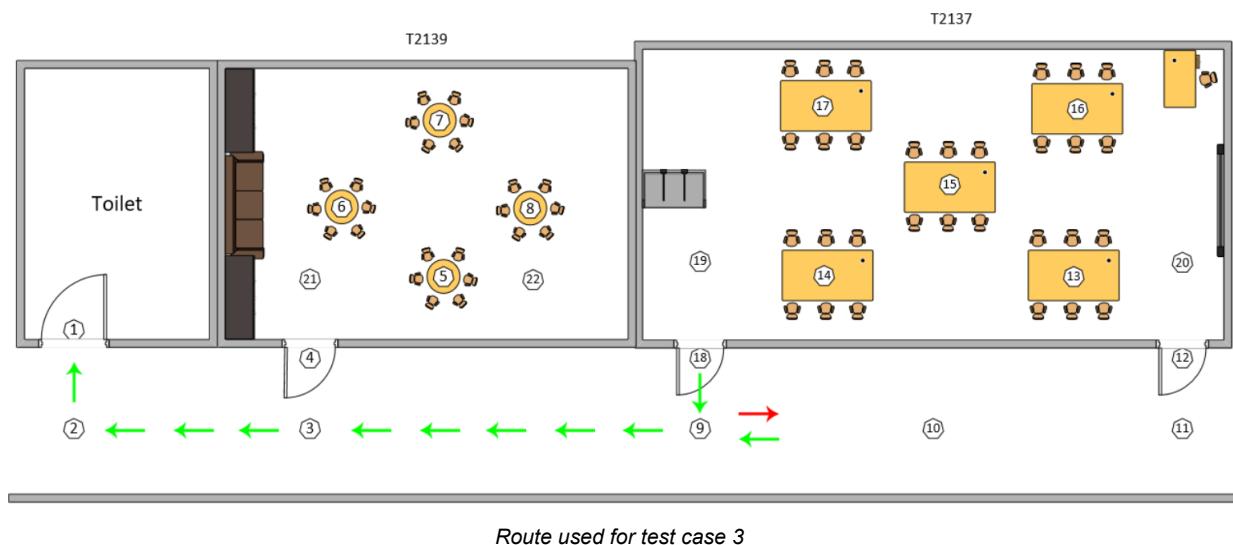
Test Execution Date: 10/05/2018				
Test Duration: 1 minute 05 seconds				
Pre-Conditions:				
<ul style="list-style-type: none"> User is not blindfolded. User must be logged in to an account. Bluetooth and Microphone must be granted permission on the phone. 				
Description: Navigation with multiple turns from beacon 16 which is a desk in T2137 to beacon 18 which is the backdoor of T2137.				
Step	Test Steps	Expected Results	Actual Results	Status (Pass/Fail)
1	User starts from beacon 16 and wants to navigate to beacon 17.	A voice message will tell the user to walk straight from beacon 16 to 17. It will also tell the user that beacon 17 is a turning point. User will then reach beacon 17 successfully.	A voice message containing the step was read out to the user, helping him to navigate from beacon 16 to 17. User successfully navigated to beacon 17.	Pass



2	User navigates from beacon 17 to beacon 14.	A voice message will tell the user to turn left and walk straight from beacon 17 to beacon 14. It will also tell the user that beacon 14 is a turning point. User will reach beacon 14 successfully.	A voice message containing the steps was read out to the user, helping him to navigate from beacon 17 to 14. User successfully navigated to beacon 14.	Pass
3	User navigates from beacon 14 to beacon 19.	A voice message will tell the user to turn right and walk straight from beacon 14 to 19. It will also tell the user that beacon 19 is a turning point. User will reach beacon 19 successfully.	A voice message containing the steps was read out to the user, helping him to navigate from beacon 14 to 19. User successfully navigated to beacon 19.	Pass
4	User navigates from beacon 19 to beacon 18.	A voice message will tell the user to turn left and walk straight from beacon 19 to 18. User will reach beacon 18 successfully.	A voice message containing the steps was read out to the user, helping him to navigate from beacon 19 to 18. User successfully navigated to beacon 18.	Pass



Test Case 3



Test Execution Date: 17/05/2018

Test Duration: 1 minute 33 seconds

Pre-Conditions:

- User is not blindfolded.
- User must be logged in to an account.
- Bluetooth and Microphone must be granted permission on the phone.

Description: Purposely navigating the user from beacon 18 which is the backdoor for T2137 to beacon 1 which is the toilet.

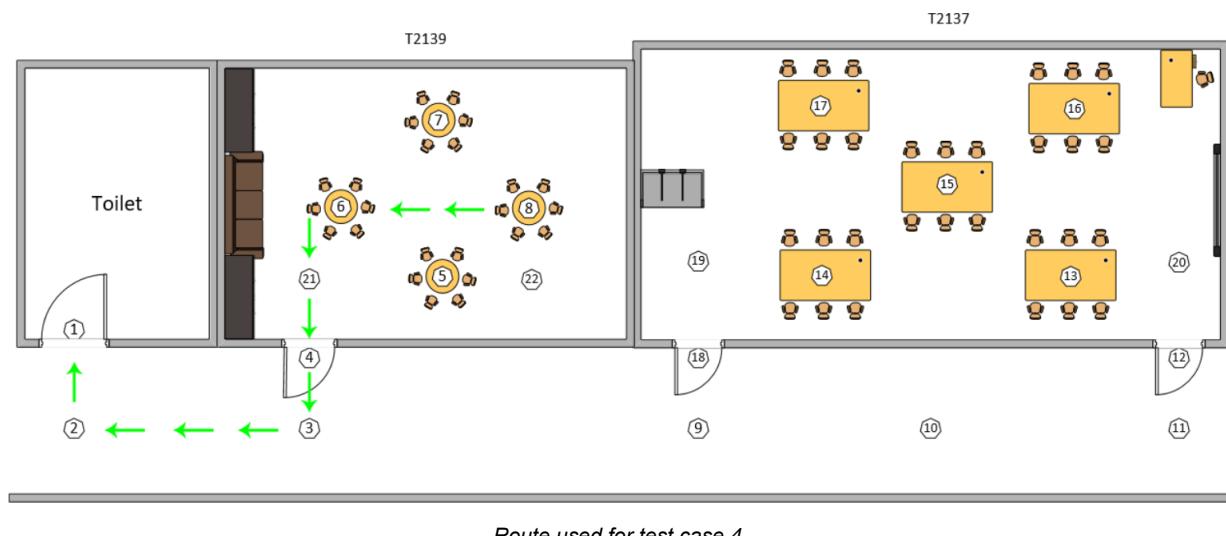
Step	Test Steps	Expected Results	Actual Results	Status (Pass/Fail)
1	User starts from beacon 18 and wants to navigate to beacon 9.	A voice message will tell the user to walk straight from beacon 18 to 9. It will also tell the user that beacon 9 is a turning point. User will then reach beacon 9 successfully.	A voice message containing the step was read out to the user, helping him to navigate from beacon 18 to 9. User successfully navigated to beacon 9.	Pass



2	User purposely navigates to the wrong direction from beacon 9.	The application will notify the user that he/she is walking in the wrong direction and guide them back to the correct direction.	A voice message has told the user that he was walking in the wrong direction and prompted him to make adjustments.	Pass
3	User navigates back to beacon 9 (the origin beacon for the current step)	User will arrive back at beacon 9.	User has successfully arrived back to beacon 9.	Pass
4	User navigates from beacon 9 to beacon 3.	A voice message will tell the user to walk straight from beacon 9 to 3. User will then reach beacon 3 successfully.	A voice message containing the step was read out to the user, helping him to navigate from beacon 9 to 3. User successfully navigated to beacon 3.	Pass
5	User navigates from beacon 3 to beacon 2.	A voice message will tell the user to walk straight from beacon 3 to 2. It will also tell the user that beacon 2 is a turning point. User will then reach beacon 2 successfully.	A voice message containing the step was read out to the user, helping him to navigate from beacon 3 to 2. User successfully navigated to beacon 2.	Pass
6	User navigates from beacon 2 to beacon 1.	A voice message will tell the user to turn right from beacon 2 to 1. User will then reach beacon 1 successfully.	A voice message containing the step was read out to the user, helping him to navigate from beacon 2 to 1. User successfully navigated to beacon 1.	Pass



Test Case 4



Route used for test case 4

Test Execution Date: 18/05/2018

Test Duration: 1 minute 04 seconds

Pre-Conditions:

- User is not blindfolded.
- User must be logged in to an account.
- Bluetooth and Microphone must be granted permission on the phone.

Description: Navigation from beacon 8 which is a desk in T2139 to beacon 1 which is the toilet.

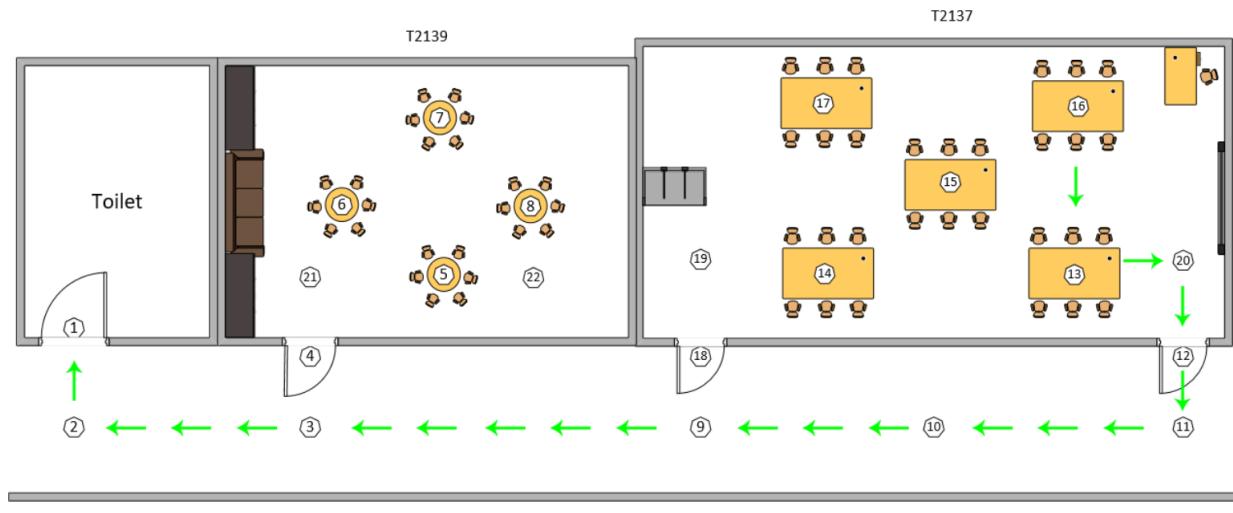
Step	Test Steps	Expected Results	Actual Results	Status (Pass/Fail)
1	User starts from beacon 8 and wants to navigate to beacon 6.	A voice message will tell the user to walk straight from beacon 8 to 6. It will also tell the user that beacon 6 is a turning point. User will then reach beacon 6 successfully.	A voice message containing the step was read out to the user, helping him to navigate from beacon 8 to 6. User has successfully navigated to beacon 6.	Pass



2	User navigates from beacon 6 to beacon 21.	A voice message will tell the user to turn left from beacon 6 to 21. User will then reach beacon 21 successfully.	A voice message containing the step was read out to the user, helping him to navigate from beacon 6 to 21. User has successfully navigated to beacon 21.	Pass
3	User navigates from beacon 21 to beacon 4.	A voice message will tell the user to go straight from beacon 21 to 4. User will then reach beacon 21 successfully.	A voice message containing the step was read out to the user, helping him to navigate from beacon 21 to 4. User has successfully navigated to beacon 4.	Pass
4	User navigates from beacon 4 to beacon 3.	A voice message will tell the user to go straight from beacon 4 to 3. It will also tell the user that beacon 3 is a turning point. User will then reach beacon 3 successfully.	A voice message containing the step was read out to the user, helping him to navigate from beacon 4 to 3. User has successfully navigated to beacon 3.	Pass
5	User navigates from beacon 3 to beacon 2.	A voice message will tell the user to turn right from beacon 3 to 2. It will also tell the user that beacon 2 is a turning point. User will then reach beacon 2 successfully.	A voice message containing the step was read out to the user, helping him to navigate from beacon 3 to 2. User has successfully navigated to beacon 2.	Pass
6	User navigates from beacon 2 to beacon 1.	A voice message will tell the user to go straight from beacon 2 to 1. User will then reach beacon 1 successfully.	A voice message containing the step was read out to the user, helping him to navigate from beacon 2 to 1. User has successfully navigated to beacon 1.	Pass



Test Case 5



Test Execution Date: 18/05/2018

Test Duration: 2 minute 03 seconds

Pre-Conditions:

- User is not blindfolded.
- User must be logged in to an account.
- Bluetooth and Microphone must be granted permission on the phone.

Description: Navigation from a desk in T2137 to the toilet (beacon 16 to beacon 1)

Step	Test Steps	Expected Results	Actual Results	Status (Pass/Fail)
1	User starts from beacon 16 and wants to navigate to beacon 13.	A voice message will tell the user to walk straight from beacon 16 to 13. It will also tell the user that beacon 13 is a turning point. User will then reach beacon 13 successfully.	A voice message containing the step was read out to the user, helping him to navigate from beacon 16 to 13. User has successfully navigated to beacon 13.	Pass



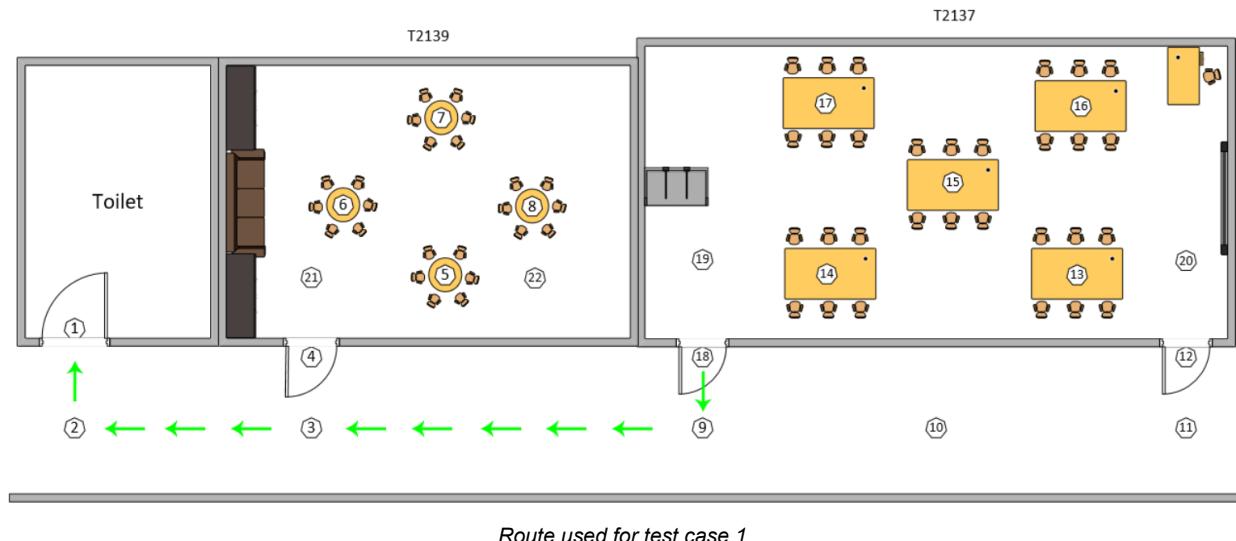
2	User navigates from beacon 13 to beacon 20.	A voice message will tell the user to turn left from beacon 13 to 20. It will also tell the user that beacon 20 is a turning point. User will then reach beacon 20 successfully.	A voice message containing the step was read out to the user, helping him to navigate from beacon 13 to 20. User has successfully navigated to beacon 20.	Pass
3	User navigates from beacon 20 to beacon 12.	A voice message will tell the user to turn right from beacon 20 to 12. User will then reach beacon 12 successfully.	A voice message containing the step was read out to the user, helping him to navigate from beacon 20 to 12. User has successfully navigated to beacon 12.	Pass
4	User navigates from beacon 12 to beacon 11.	A voice message will tell the user to go straight from beacon 12 to 11. It will also tell the user that beacon 11 is a turning point. User will then reach beacon 11 successfully.	A voice message containing the step was read out to the user, helping him to navigate from beacon 12 to 11. User has successfully navigated to beacon 11.	Pass
5	User navigates from beacon 11 to beacon 10.	A voice message will tell the user to turn right from beacon 11 to 10. User will then reach beacon 10 successfully.	A voice message containing the step was read out to the user, helping him to navigate from beacon 11 to 10. User has successfully navigated to beacon 10.	Pass
6	User navigates from beacon 10 to beacon 9.	A voice message will tell the user to go straight from beacon 10 to 9. User will then reach beacon 9 successfully.	A voice message containing the step was read out to the user, helping him to navigate from beacon 10 to 9. User has successfully navigated to beacon 9.	Pass



7	User navigates from beacon 9 to beacon 3.	A voice message will tell the user to go straight from beacon 9 to 3. User will then reach beacon 3 successfully.	A voice message containing the step was read out to the user, helping him to navigate from beacon 9 to 3. User has successfully navigated to beacon 3.	Pass
8	User navigates from beacon 3 to beacon 2.	A voice message will tell the user to go straight from beacon 3 to 2. It will also tell the user that beacon 2 is a turning point. User will then reach beacon 2 successfully.	A voice message containing the step was read out to the user, helping him to navigate from beacon 3 to 2. User has successfully navigated to beacon 2.	Pass
9	User navigates from beacon 2 to beacon 1.	A voice message will tell the user to turn right from beacon 2 to 1. User will then reach beacon 1 successfully.	A voice message containing the step was read out to the user, helping him to navigate from beacon 2 to 1. User has successfully navigated to beacon 1.	Pass



Test Case 6



Test Execution Date: 17/05/2018

Test Duration: 1 minute 46 seconds

Pre-Conditions:

- User is blindfolded.
- User must be logged in to an account.
- Bluetooth and Microphone must be granted permission on the phone.

Description: Navigation from beacon 18 which is the backdoor for T2137 to beacon 1 which is the toilet.

Step	Test Steps	Expected Results	Actual Results	Status (Pass/Fail)
1	User starts from beacon 18 and wants to navigate to beacon 9.	A voice message will tell the user to walk straight from beacon 18 to 9. It will also tell the user that beacon 9 is a turning point. User will then reach beacon 9 successfully.	A voice message containing the step was read out to the user, helping him to navigate from beacon 18 to 9. User successfully navigated to beacon 9.	Pass
2	User navigates from beacon 9 to beacon 3.	A voice message will tell the user to turn right and walk straight from beacon 9 to beacon 3. User will reach beacon 3 successfully.	A voice message containing the steps was read out to the user, helping him to navigate from beacon 9 to 3. User successfully navigated to beacon 3.	Pass



3	User navigates from beacon 3 to beacon 2.	A voice message will tell the user to walk straight from beacon 3 to 2. It will also tell the user that beacon 2 is a turning point. User will reach beacon 2 successfully.	A voice message containing the step was read out to the user, helping him to navigate from beacon 3 to 2. User successfully navigated to beacon 2.	Pass
4	User navigates from beacon 2 to beacon 1.	A voice message will tell the user to turn right and walk straight from beacon 2 to 1. User will reach beacon 1 successfully.	A voice message containing the steps was read out to the user, helping him to navigate from beacon 2 to 1. User successfully navigated to beacon 1.	Pass