A Project Report on

"Dynamic Navigation System using BLE Beacons"

Submitted in partial fulfillment of the requirement for

Degree in Bachelor of Engineering (Computer Engineering)

By

Akshay Bhosale	101506
Gloria Benny	101512
Robin Jaison	101525
Adil Khot	101537

Guided By

Mrs. Sandhya Pati

Department of Computer Engineering

Fr. Conceicao Rodrigues Institute of Technology Sector 9A, Vashi, Navi Mumbai - 400703

University of Mumbai

2018-19

Approval Sheet

This is to certify that the project entitled

"Dynamic Navigation System using BLE Beacons"

Submitted by

Akshay Bhosale	101506
Gloria Benny	101512
Robin Jaison	101525
Adil Khot	101537

In partial fulfillment of degree of **B.E.** in **Computer Engineering** for term work of the project is approved.

Supervisor : ____________________

Project Coordinator : ____________________

Examiners :

1. ____________________

2.

Head of Department :

Date :
Place :

College Seal:

DECLARATION

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

1. Akshay Bhosale	101506	___________________
2. Gloria Benny	101512	___________________
3. Robin Jaison	101525	___________________
4. Adil Khot	101537	___________________

Date:

ABSTRACT

The navigation system at present is commonly based on the GPS, which provides highest accuracy in locating the latitude and longitude information. But to use this facility there has to be internet connectivity in the device for the satellites to communicate with the mobile equipments sensors. As observed mobile data provided by network providers gets hindered with more number of walls and layers and thus getting proper data services inside a closed infrastructure is sometimes not possible. Thus there is a need to think upon other alternatives that can locate a user. Such alternatives include understanding the user's environment by using the bluetooth signals in the surroundings, understanding the wifi environment and magnetic signals in the environment. All this data can easily be captured using the mobile equipments sensors. Even the bluetooth signals can be used to serve our purpose using devices called beacons. These devices emit continuous radio packets and were conventionally modelled for advertisements, and these same signals can be used to understand the environment indoors. Thus, using sensor fusion technology, we combine all the above mentioned signals data to effectively localize a smartphone inside a building.

A network of beacons is installed in a facility by building a software that will manage the activities of each beacon node of the network. Such a network can be used in Malls, Airports, Hospital, Amusement Parks where it is mostly difficult to navigate to a particular place within the facility. The application is launched as the user enters the premises. It lets the user search for the destination, view the floorplans and start navigation. The user is provided with the estimated time of arrival and directed to the destination effectively. The user can navigate through multiple indoor facilities using the same application.

CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

CHAPTER 1

INTRODUCTION

Navigation can be defined as acknowledging where a person is in an environment and knowing how to get to the desired destination from their current location. Global positioning systems (GPS) and mapping technologies provide accurate and simple means for navigation in outdoor environments. For indoor environments, reading and following signs remains the easiest and most reliable option because GPS and associated advances for outdoor environments do not apply. Indoor environments can be geographically large and intimidating such as grocery stores, airports, sports stadiums, large office buildings, and hotels. In unfamiliar, large indoor spaces, it is common for people to be confused in finding their way around. This could be due to the lack of well marked signs and maps, or not being familiar with the conventions or languages used on these signage.

We developed an admin portal where the facility owners can get themselves registered and upload the floorplans in PDF format. The floorplan thus received is cleaned using AutoCAD software by removing irrelevant data and analysing the measurements mentioned in it. The file is then saved in .png format. the image thus obtained is used for fingerprinting and georeferencing.

Fingerprinting is done initially to record the signal strengths of Wifi, Bluetooth, etc. The beacon placement will be implemented manually by extracting the necessary information from the floor map image of the indoor space for subsequent beacon placement and path computations for navigation.

Using QGIS software, the image is georeferenced and converted into .tiff file format. Georeferencing is a process by which a raster dataset without spatial reference can be matched with a layer that does have spatial reference. Thus, each pixel of the image now has corresponding global coordinate values. This is then used to fix the floorplan on the world map and extrude the rooms in the plan for 3D visualization using Mapbox.

We planned to implement user navigation module using a web app. But we later found that the results wouldn’t be accurate and would be less effective. Thus we developed a dynamic android application that would help users navigate indoors. Unlike traditional implementations, we propose a single app that would provide same functionalities for all indoor locations.

CHAPTER 2

LITERATURE SURVEY

CHAPTER 2

LITERATURE SURVEY

“Technology, like art, is a soaring exercise of the human imagination.” These great words of Daniel Bell reflects in every aspect of day to day life. When outdoor navigation is developing with such great phase, why let indoor navigation remain untouched. The advancement in BLE beacon technology with overwhelming support from Android and iOS developers has led to advancements in indoor navigational features making it a rapidly emerging technology which promises a great future in huge infrastructures like malls, airports, hospitals etc.

2.1 Beacon implemented system for mobile content management

Authors : Richard Graves, *Chris Blanz*, *Beat Zenerino*, *Kevin Huber* and *Greg Thornton*

For large indoor spaces, manual determination of all beacon placement locations and computing the paths between them would be time-consuming and labor-expensive. This approach will require manual identification of walking paths on a floor plan, marking of POIs, determining the distance between any two POIs, deducing the orientation between them for navigation, for calculating shortest paths between them, and successive adjustments to optimize the resulting paths.

2.2 Beacon Placement for Indoor Localization using Bluetooth

Author : Sudarshan S. Chawathe

This paper explains a method to determine the location of a mobile device in an indoor environment using Bluetooth beacons. Bluetooth devices have short range compared to its alternatives such as 802.11. The limited range of Bluetooth can be a boon and bane at the same time. The disadvantage may be that we might require to deploy more number of devices to provide adequate coverage. However, the limited range of Bluetooth is also an advantage when it comes to localization which is performed using a cell-based method that determines the region of intersection of beacon ranges. Localisation using beacons may be achieved using a various techniques, such as triangulation, trilateration, multilateration, and cell-based methods. The cell based method is found to be more accurate and advantageous over the others.

2.3 Survey of Wireless Based Indoor Localisation Technologies

Author- Junjie Liu, junjie.liu@wustl.edu

In this paper, we studied various wireless based indoor localisation techniques.We analysed these techniques as we know infrastructural obstacles act as a hindrance for the propagation of GPS signals.

The three main methods for indoor localisation are:

a. **Vision Based Indoor Localisation** - This technique uses cameras to capture images of the location and then performs localisation by performing training and learning over those images. But the disadvantage in this technique is it will require more computing resource for analysing the images and thus more power. Also, this technique is not scalable considering the cost of cameras.

b. **Wireless Based Indoor localisation** - It is the most commonly used indoor localisation technique because the cost of wireless chips is cheaper than cameras. The power and computing resource utilization is also found to be lower than vision based indoor localisation.

Another technique uses the smartphone sensors like the accelerometer, the compass, and the gyroscope to localise the user with improved precision.

2.4 An Analysis of the Accuracy of Bluetooth Low Energy for Indoor Positioning Applications

Author - R. Faragher, University of Cambridge, UK; R. Harle, University of Cambridge, UK

The paper mainly talks about the advantages of BLE beacons over traditional WiFi systems and how it can be proven to be power efficient and accurate enough if it is configured with correct settings. Experiments were conducted on a conveyor belt and the results were analysed. The conclusion of the study was that increasing the density of beacons per over an area, would increase the positioning accuracy up to a threshold of 68. No further improvement in the accuracy was found beyond this threshold.

Using these results and right amount of power settings a good performance can be achieved. It was also proven through some studies that active Wifi scanning and Wi-Fi network access can lead to faulty signal strength measurements.

2.5 Automatic Floor Plan Analysis for Adaptive Indoor Wifi Positioning System

Author - Xuning Zhang, Albert Kai-sun Wong

The author explains about fetching of key geographic data from an existing floor plan and thus generating a weighted graph for that indoor space automatically. This detects corridors, doors, rooms etc from the floor plan by analysing the interconnections of vertices that is stored in a weighted adjacency matrix.

The detected figures are then associated with useful data. The cartesian coordinates of of the detected figures are stored in a list data structure for easy access in future. This information is used to create a Euclidean graph.

2.6 Geomagnetic Field Based Indoor Localization Using Recurrent Neural Networks

Author- Ho Jun Jang, Jae Min Shin, and Lynn Choi, School of Electrical Engineering Korea University, Seoul, Korea

This Paper highlights the Benefits of using Geomagnetic data as support for indoor localization techniques. RF signals fingerprinting such as those involving WiFi signals and Bluetooth signals were found to have a disadvantage of being time variant and unstable in nature. This gave rise to problems like inconsistent fingerprinted data in case the signals environment are changed (change in router and beacon position) in the future. Thus when compared to RF signals data, the geomagnetic field signals were static and exhibited stable signal strength in the time domain.
This technique of localization though being stable is not self-sufficient in indoor localization as the indoor structures are generally made symmetrical. Such structures exhibit similar magnetic data at different points on the map. Thus Geomagnetic data be used along with other techniques and act a medium to increase the accuracy of those techniques by being used as an add-on.

This paper speaks about the creation of a recurrent neural network model. As mentioned earlier it is possible that there may exist multiple points on the map that exhibit similar geomagnetic behavior. Thus the input to the neural network is provided in terms of mapping of the floor map's image's pixel coordinates say (X, Y) to the corresponding geomagnetic data captured by the smartphones magnetometer sensor say(Mx, My, Mz). As a person walks over the map collecting
data at every point he stores the immediate previous location coordinate along with the current location and magnetic data at that point. Creating the neural network in such a way has an advantage. As the app user now walks in the indoor space and come across a sequence of magnetic fields it will lead to a unique pattern of the sensor readings over time. Thus the recurrent neural network model uses this sequence of reading and the knowledge it has been trained on previously to predict the correct reading of the user. The prediction now is based on both current and past value of readings. So the problem of confusion between similar data points is easily solved in this way.

2.7 Geomagnetic Fingerprint Maps for Indoor Positioning

Author- Xiaolong Xu and Licheng Lin , Nanjing University of posts and Telecommunications. Nanjing,China

This paper analyzes the information of the geomagnetic field, compares the schemes of constructing the geomagnetic fingerprint map via experiments, and promotes the further development of the geomagnetic field in indoor positioning technology. In order to better achieve indoor positioning, how to effectively build a fingerprint map has become the first indoor positioning technology difficulty. As the indoor environment changes, signal block, fingerprint map update is not timely and other issues will make the accuracy of indoor positioning significantly reduced. So how to build a high-precision and effective fingerprint map has become the focus of the indoor positioning technology research.

The paper highlights the advantages of using the geomagnetic field to achieve indoor positioning. The geomagnetic field does not rely on external energy and it always exists. Using the geomagnetic field as the basis for indoor positioning is not dependent on the source of the device, especially if there is no power support. Compared to the Wi-Fi positioning scheme, the use of geomagnetic field positioning program does not need to set up any auxiliary equipment in the room.

The geomagnetic field information collected by the smartphone is represented by the data of the three axes. This is synthesized to represent rich information of the magnetic field. This information can be used in localization. For the construction of the geomagnetic fingerprint map, an off-line geomagnetic fingerprint map has to be constructed at first. It is done by dividing the indoor environment into a virtual grid, and then collecting the geomagnetic information of each point in the grid and saving it to the database. Secondly, the indoor path can also be used to store the magnetic field information, with the information stored in these paths. A fingerprint database based on the indoor geomagnetic information will also be required to build a geomagnetic fingerprint map. In the positioning phase, the information collected by the smartphone sensor is matched with the fingerprint map database. The methods of constructing geomagnetic fingerprint map were verified via experiments. Experimental results show that the accuracy of the scheme which uses the magnetic field strength values of special landmarks is not enough to constructing efficient geomagnetic fingerprint maps. The method of combining the magnetic field strength value with the path can not only solve the numerical difference caused by the microelectronic mechanical systems sensor effectively but also has high positioning accuracy.

It was experimentally found that it is feasible to use the path combined with the change of the total intensity of the geomagnetic field. It was also found that the results of the geomagnetic field varied when recorded from two different smartphones. Due to the errors caused by the sensor equipment, magnetic field information was not fully utilized, but it is believed that with the

2.8 Nearby Messages and Nearby Notifications

You can add attachments to your beacons, and access those attachments as messages, with your own app using Nearby Messages and Nearby Notifications, which does not require your app to be installed. Since the messages are stored in the cloud, you can update them as often as you like without the need to update the beacons themselves.

Nearby Notifications helps users to discover what's around them, by surfacing location-specific notifications for apps and websites, with no prior app installs required.

There are two basic types of experiences that you can provide using Nearby Notifications:

- Link to an HTTPS URL to provide a notification to the user, and open the URL in the browser when the user taps the notification.
- Trigger an app intent to launch an app that is already installed and perform a specific action. If the app isn't installed, the user is taken to the Play Store where the app can be installed; the user can then continue to the feature specified by the developer.

Neither experience requires that an app is installed on the user's device. For Interacting with the Physical Web which enables quick and seamless interaction with beacons, the beacons have to be configured using a https type website i.e. the website has to be certified secure in order to use it as a part of the physical web. The configured beacon will continuously emit eddystone url frames at set interval of time. The power of the signal has to be adjusted to get better result in signal detection. A high power results in faster battery drain and lower power may lead in addition of noise in signal. Thus the power level and time interval have to be set to get efficient battery utilization and high probability of signal detection. This compressed URL can be read by Nearby Notifications and by Chrome using the Physical Web.

2.9 The Eddystone Format

Eddystone is an open beacon protocol developed by Google which features numerous advantages over iBeacon, most important of all being its open source license. Eddystone is an open beacon format developed by Google and designed with transparency and robustness in mind. Eddystone can be detected by both Android and iOS devices. The Eddystone data field contains the same information as the iBeacon one, only formatted in a different way. Additionally, different kinds of Eddystone frames are available, depending on the beacon’s purpose. There are three kinds of Eddystone frames:

- Eddystone-UID: A unique, static ID with a 10-byte Namespace component and a 6-byte Instance component.
- Eddystone-URL: A compressed URL that, once parsed and decompressed, is directly usable by the client.
- Eddystone-TLM: Beacon status data that is useful for beacon fleet maintenance, and powers Google Proximity Beacon APIs diagnostics endpoint. -TLM should be interleaved with

an identifying frame such as Eddystone-UID or Eddystone-EID (for which the encrypted eTLM version preserves security).

Here we are using Eddystone frame type called URL. The data in this kind of frame, as its name suggests, is a URL of up to 17 bytes in length. Any device passing by, with the necessary functionality built in to it, can pick and interact with this URL.

2.10 Mapbox GL JS

This tool is used to build custom map styles for various applications, providing services to Android, iOS and web applications. They can be used to design suitable indoor maps that are user friendly and look more realistic. These vector maps can be created using the Mapbox Studio style editor. It is a full-featured map editor that gives you total control over the style of your map directly in your browser.

- Create styles that change dynamically based on zoom level.
- Use custom fonts.
- Set custom alignment, pitch, offset, and more for your labels.
- Set colors, weights, and opacity for your map layers.
- Filter tilesets based on attributes.

Mapbox GL JS is a JavaScript library for creating interactive, customizable maps from Mapbox styles and vector tiles. This library is built on the WebGL platform which was built initially to provide faster and smoother video gaming experience in the browser environment.This library enables you to build advanced interactions into your maps, including smooth zooming, map bearing and pitch, querying underlying map data, and dynamically filtering the data you choose to display. The maps can be made using the mapbox default style templates or custom style sheets can be used that are to be prepared according to the mapbox documentations specified.The map can store additional data other than what we want to display which is useful in providing features like navigation, route finding etc.

2.11 User Positioning on map using Indoor Atlas Tool

It is found from numerous articles that accurate positioning of user on map is not possible just using one technology. Hence using Beacons alone would not be able to identify the users accurate step by step movements and even if we use grid layout of beacons to determine with a better accuracy it would increase the implementation cost and won't be a better solution for positioning as beacons were not intended for tracking users, rather their use was meant for advertisement purposes. Hence we need to use a fusion of various data that are available to us.

Indoor Atlas provides a solution by combining wifi, BLE signals and geomagnetic data that is captured by the smartphones sensors to better understand the environment and give accurate results.

The maximum accuracy is obtained by triangulation of magnetic field readings that are captured by the smartphone's inbuilt magnetic sensor. The readings are a result of interaction of the magnetic field of the infrastructure's steel components and earth's magnetic field. The BLE radiations and WiFi also helps in determining the location over certain accuracy. The result of fusing these information is used to create a heatmap of the floor which stores these readings and a bigger dataset of such data when fed into the system results in more accurate positioning results that can detect smooth and swift movements of the user too.

2.12 Kontakt SDK for Android

This API provides direct access to all resources available on Kontakt.io Management platform. It allows to implement beacon administration functionality into 3rd party solutions without the need to build all underlying logic from the ground up. This include configuring the beacon data like setting the url, analysing the beacon usage, monitoring battery level etc. The beacon configuration can also be done using the app provided by kontak.io which is “Kontakt.io Administration App” which can be installed from play store. Also, this SDK can be used in our applications to read the attached data with beacons.

2.13 QGIS(Quantum Geographic Information System)

QGIS (previously known as Quantum GIS) is a free and open-source cross-platform desktop geographic information system (GIS) application that supports viewing, editing, and analysis of geospatial data.

The image file is a raster file and to place it with the correct scale onto the world coordinate system, each pixel of the image is assigned with geospatial data like latitude and longitude value. Such data mapping operations can be performed easily using QGIS software and the process is called georeferencing of the image. The georeferencing step involves setting Ground Control Points(GCP) on the raster image in the QGIS software. These GCPs are collected by doing field surveys with a GPS device and noting down the correct coordinates at few points on the map. These are then uploaded to the appropriate pixel by clicking on the image and entering the data found. The QGIS software then uses these reference points and fills all the remaining pixels in the image with correct location coordinates.

The new image that is generated by the software containing geospatial data is returned in “.tiff” format. Tiff stands for Tagged Image File Format. Now this “.tiff” file when uploaded on the world map using mapbox service gets correctly aligned with the world coordinate system. Using this alignment as reference polygons can be extruded, walls can be constructed in 3D. Moreover now, these polygons and lines have vector properties, thus as we change the zoom level they also get stretched or compressed and unlike raster image they don't get shattered into pixels at higher zoom levels. Thus using QGIS the correct location of a map on world coordinates system is obtained and in the process, the raster map image is vectorized.

CHAPTER 3
EXISTING SYSTEM

CHAPTER 3

EXISTING SYSTEM

3.1 Guide Maps

To guide people inside a shopping mall, guide maps are mounted at certain location that help in locating stores, food courts & restrooms.

Fig 3.1 Sample Guide Map

The problem with this system is that a lost person has no idea about the direction he is facing hence looking at static map won't be of any help. Moreover guide maps cannot be made available at every location of the infrastructure thus it may be inaccessible to someone who needs it at the right time. Some users may not be able to interpret the map so it may not be of use to them.

3.2 Mumbai T2 App

Mumbai International Airport has launched India's first ever indoor navigation app with Augmented Reality feature for the convenience of passengers traveling through Terminal 2.

The app plays a complete navigation guide with user-friendly technology, helping passengers locate anything at the airport with a single click process. Passengers travelling from any airline just need to select the flight to get constant information through push notification system and the Mumbai T2 App will assist them with their travel.

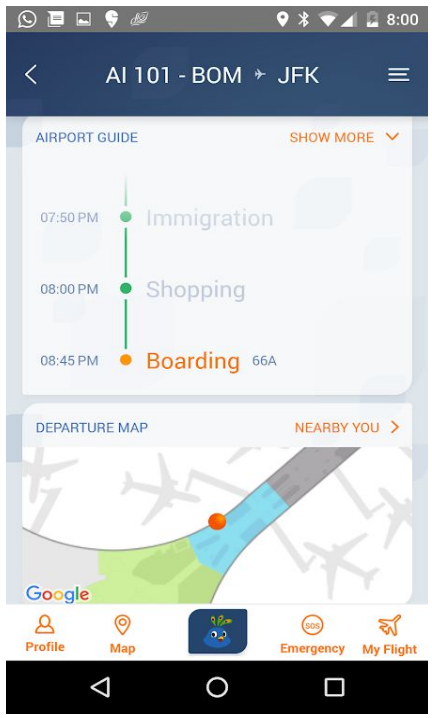

Fig 4.2 App snippet of Mumbai T2 App

The disadvantage of such applications is that the navigational features provided in the app are of no use to user if the application is not installed in his device. Moreover there is no use of implementing such an application if the user is unaware of the existence of such services.

Also such application are useful only for that particular places. Similar apps may be built for different places too, but it is infeasible to keep several apps that provide same functionality but just differ in the location where they are placed.

CHAPTER 4

SCOPE

CHAPTER 4

SCOPE

The scope of the project is to provide navigation inside huge infrastructures where GPS services are hard to access.The plan was to use hybrid combination of GPS, wifi signals, Bluetooth beacon, Geomagnetic fingerprint map to achieve accuracy of up to 1 meter. Also the plan suggested to add voice assist in future development phase so that it benefited blind people as well. Various access and permissions have to be granted to the app in order to update, manage, and view information such as GPS, Bluetooth and the geomagnetic position which are accessed on the SmartPhone.

In the initial stage, main focus was to understand the floor map if available, else one had to be created for the client and uploaded to a central repository to match with world coordinates. The plan was to achieve this using qgis software for georeferencing. Designing of building was planned to be done using mapbox sdks for android and web app.

The android app would detect these beacons and notify the users about indoor navigation in that province. The app is capable of navigating the user to his/her desired destination. Clients can use the statistics provided from the beacons and use it for maximizing their business. The main goal was to build a dynamic ecosystem so that user need not download multiple apps for enjoying the benefits of beacon services. It could have also served as a source of broadcast to pass on vital information in an infrastructure (e.g. fire alert or theft alert etc.)

CHAPTER 5

PROPOSED SYSTEM

CHAPTER 5

PROPOSED SYSTEM

To create an indoor navigation system for the general public. Development of an application which has two components.

5.1. Android Application

a. As soon as the user enters the vicinity of the region, our app understands the location and automatically opens the floor map of that building.
b. User is directed to the local indoor map of that area.
c. User gets the route to navigate with the help of which he reaches the destination.

5.2. Admin Panel

a. The owner of the area uploads the floor plan on to the admin panel.
b. The location coordinates of the facility gets updated
c. This changes are updated on the central server

CHAPTER 6

REQUIREMENT GATHERING

CHAPTER 6

REQUIREMENT GATHERING

For developing the proposed system, the hardware and software platforms to be used are mentioned below:

6.1 Hardware Requirements:

Hardware requirements are listed as below:

1. Smartphone with
• Bluetooth
• GPS
• Accelerometer
• Gyrometer
• Magnetometer
• 512 MB RAM
2. Kontakt Beacons (Eddystone)

6.2 Software Requirements:

6.2.1 Software Requirements for development are listed below:

1. Android Beacon Library
2. Indoor Atlas SDK
3. Mapbox SDK
4. Kontakt SDK
5. Qgis

6.2.2 Software Requirements for usage are listed below:

1. Android version 4.1 and up
2. Internet Connection

CHAPTER 7
SYSTEM DESIGN

CHAPTER 7

SYSTEM DESIGN

This section describes the diagrams which define the complete flow of modules as described in the proposed system.

7.1 System Overview

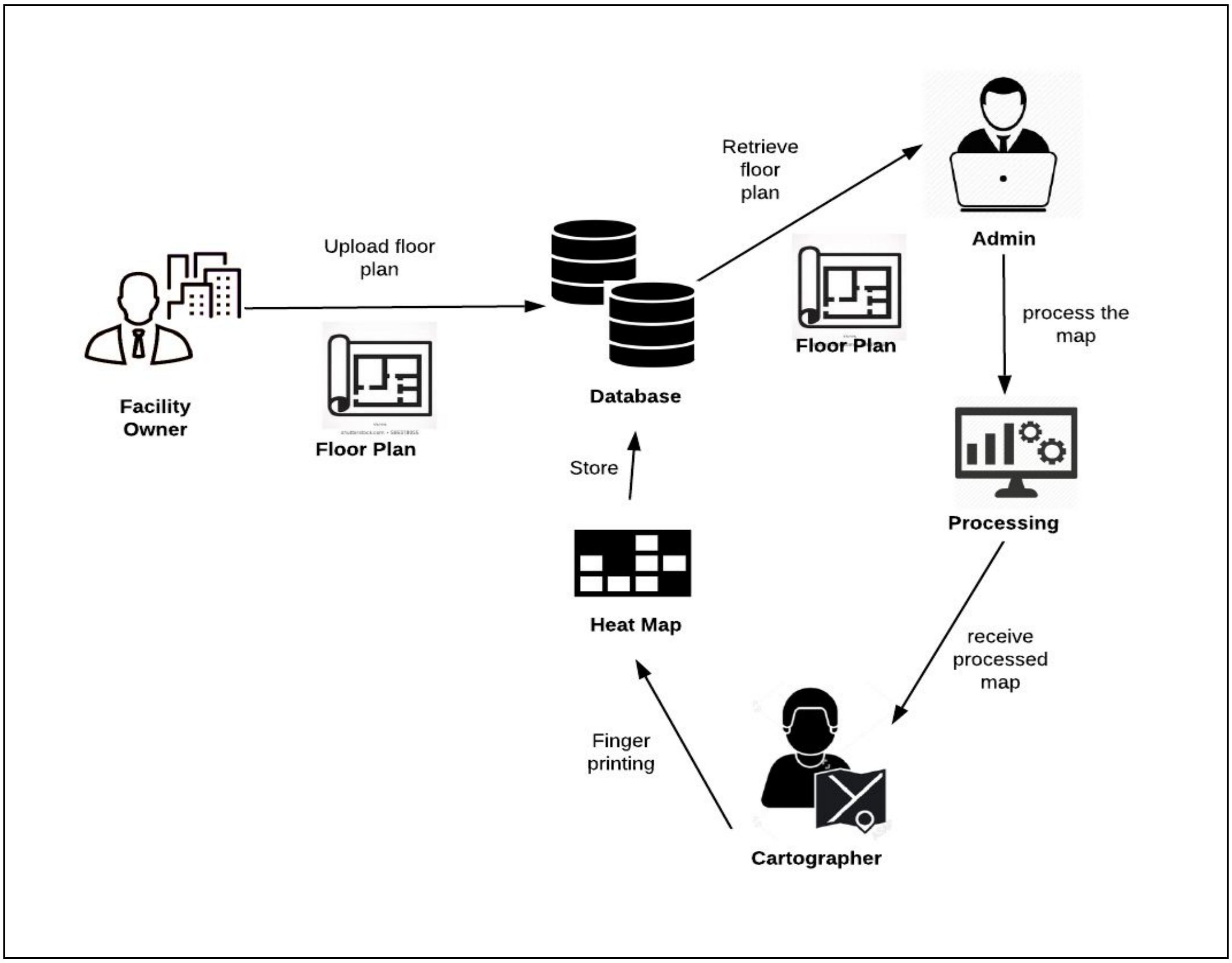

Fig 7.1 System Overview

7.2 System Workflow

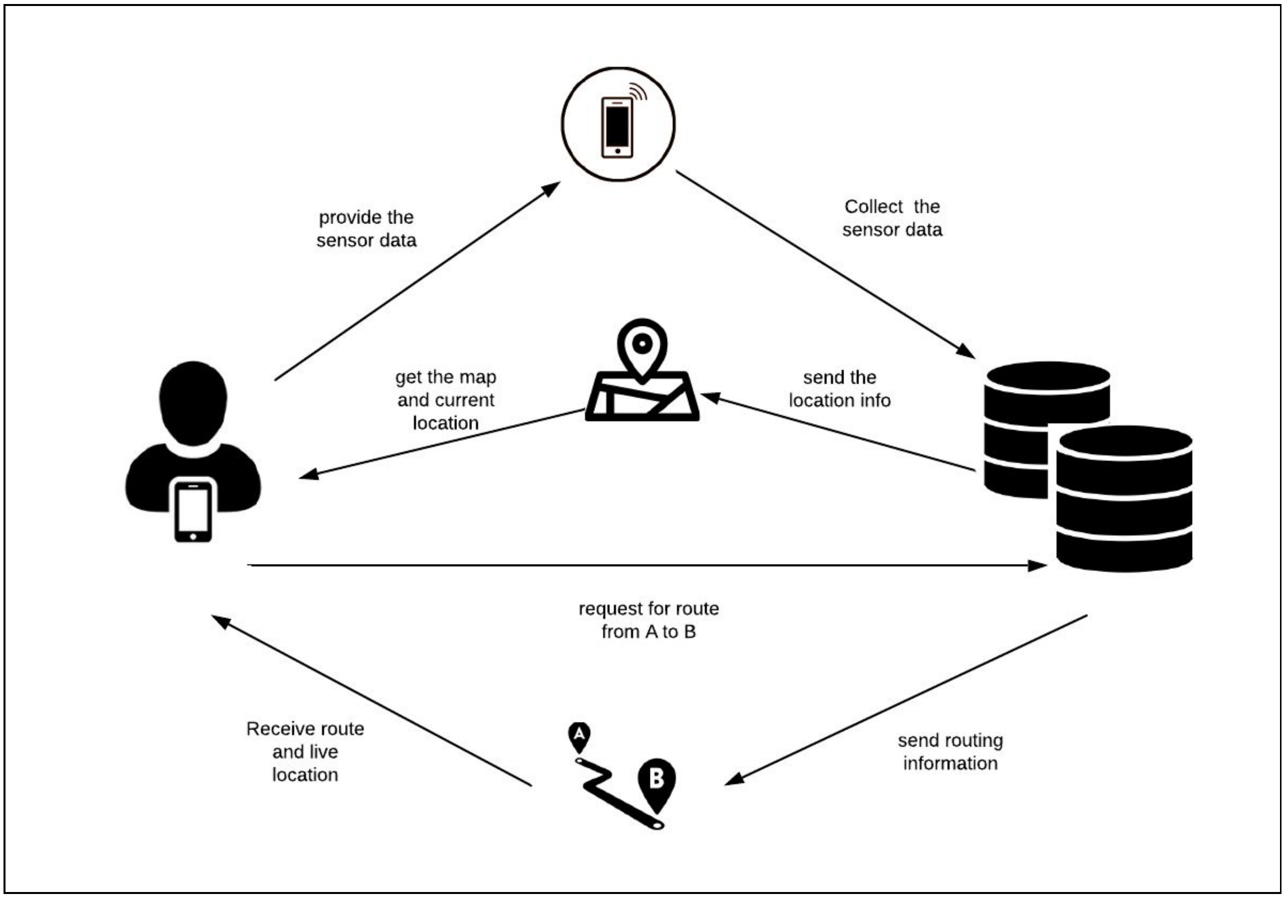

Fig 7.2 System Workflow

Fig 7.1 represents the system in general. The system can be modeled consisting of the following entities.

- **Central Server**:- It is the platform where the website/app will be hosted. This will also hold static client data i.e. the client of the facility can upload his map onto the server, monitor beacons etc.
- **Admin Panel**:- The admin panel is the entity that is used by the building owner to upload his/her floor plan onto the server, and configure the beacons and be updated about the beacon battery status.
- **Smartphone**:- This entity plays role of detecting the surroundings using sensors like Bluetooth, GPS, Accelerometer, Gyrometer, Magnetometer. The bluetooth and mobile data should also be enabled. Also this entity will be used to get the maps hosted on the server when it comes within the range of a configured beacon.

- **Beacon Network**:- This is just an arrangement of beacons which are preconfigured to provide an eddystone url packet frame of the site hosted on central server.

7.3 Block Diagram

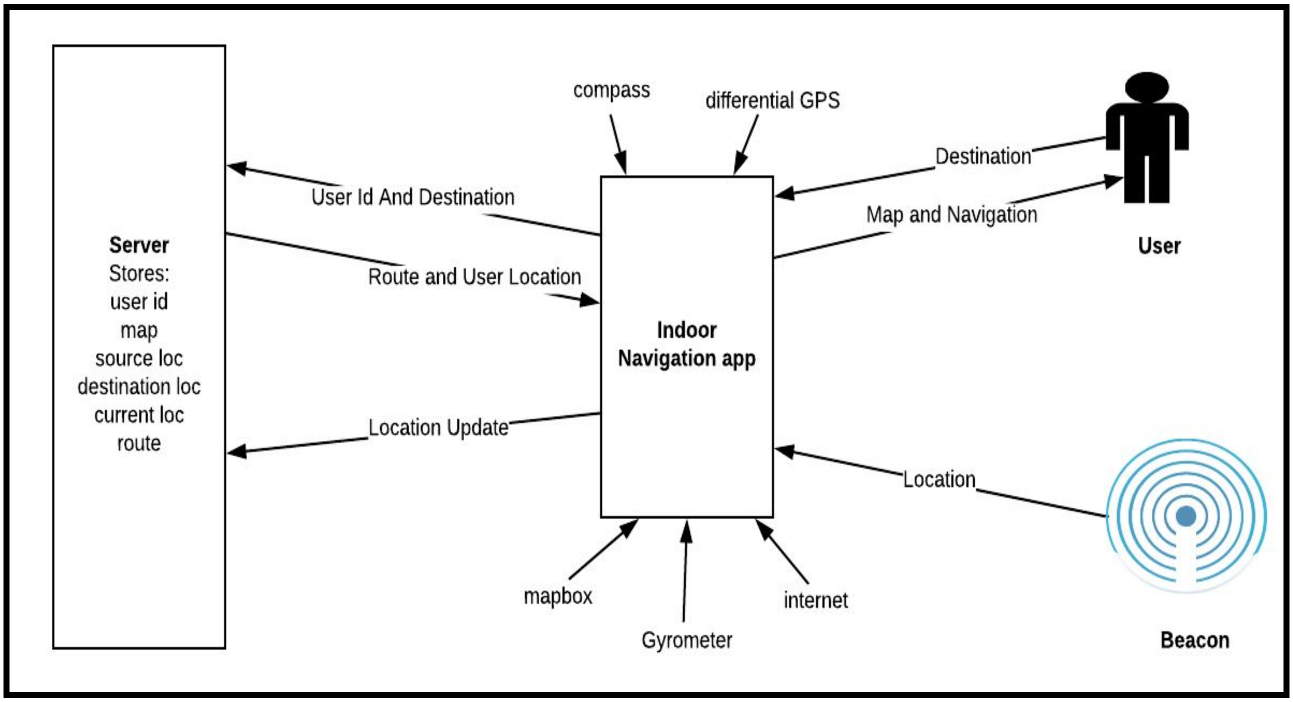

Fig 7.3 Block Diagram of Indoor Navigation System using BLE Beacons

Fig 7.3 explains the information exchange that occurs in a session of using the app

- The user gets the map data from the application and the user types in the destination he desires to navigate to.
- The application uses internet and different sensors to capture the available data readings in real time .These are sent to the main server via internet. Along with this the user's desired destination is also forwarded to the server.
- The server uses this reading that it has obtained just now and compares with the values it already has learnt about over the map and tries to locate the user on the map.
- Next it uses the destination information and calculates the route to that destination from the location it just analysed.
- This route information is then sent back to the app.

7.4 Use Case Diagram

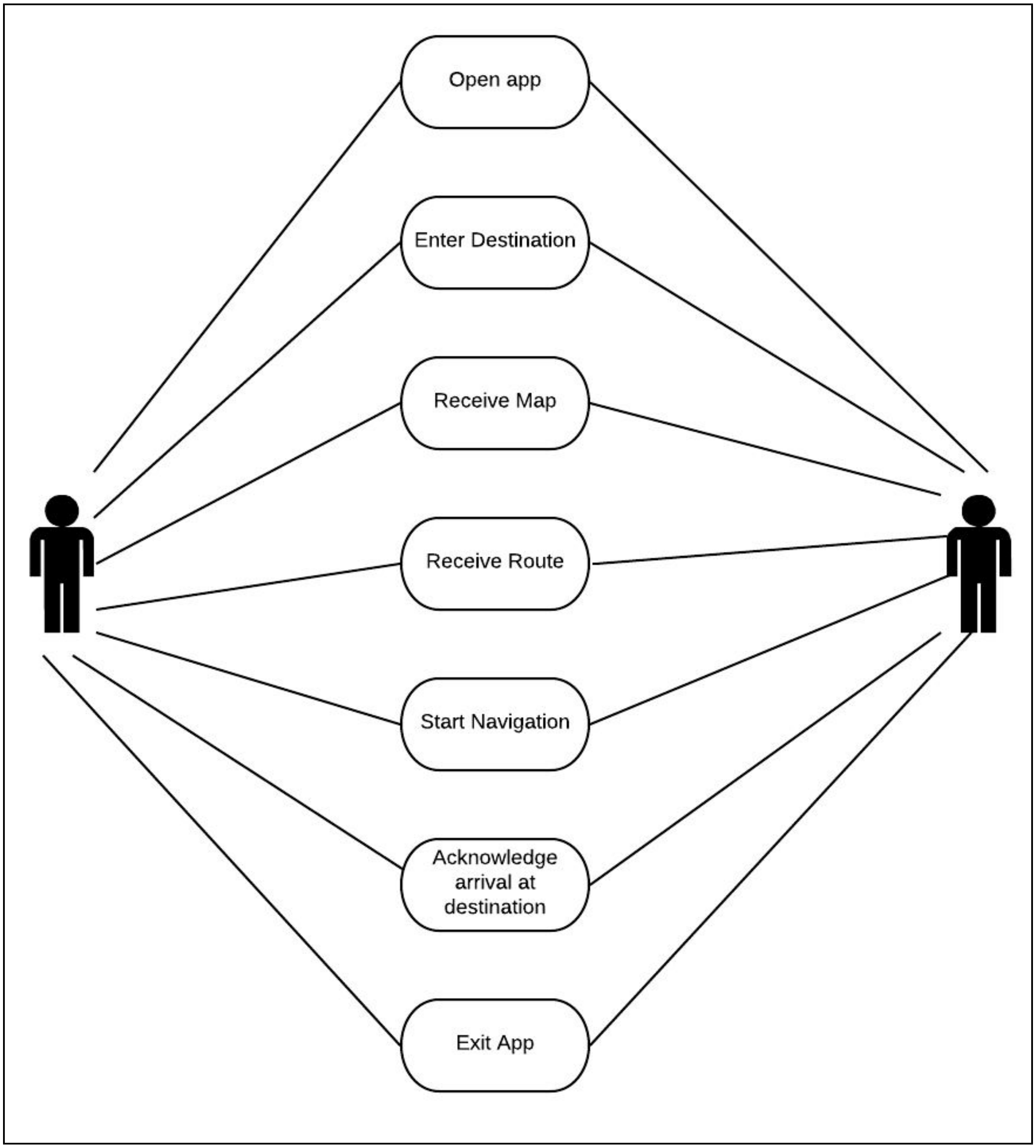

Fig 7.4 Use Case Diagram

7.5 Flowchart

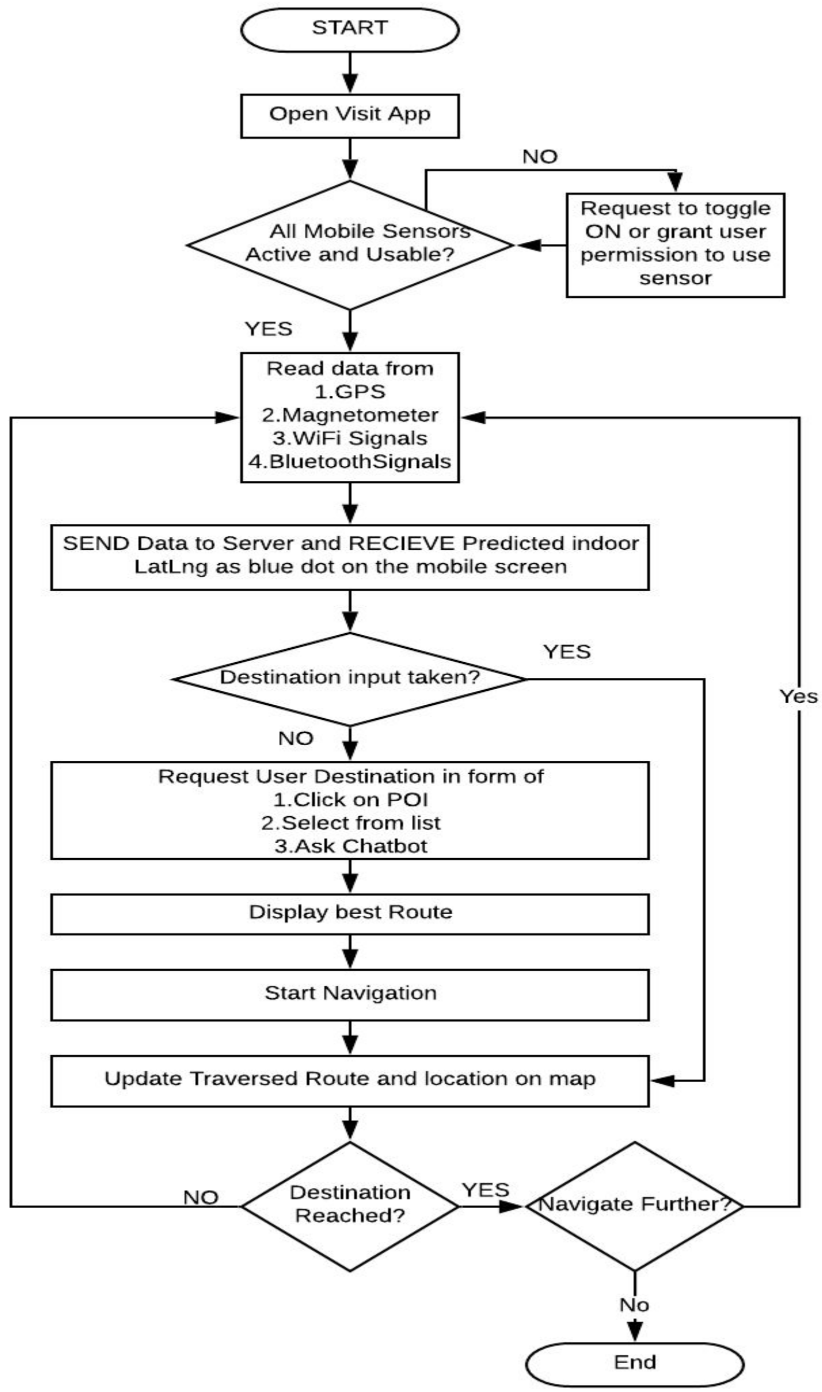

Fig 7.5 Flowchart for user navigation App

CHAPTER 8

IMPLEMENTATION

CHAPTER 8

IMPLEMENTATION

8.1 Configuring Beacons

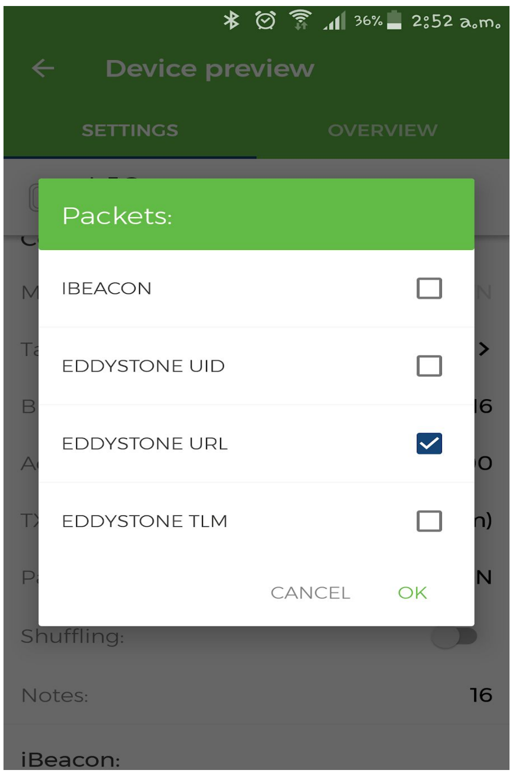

Fig 8.1.1 Select Eddystone frametype

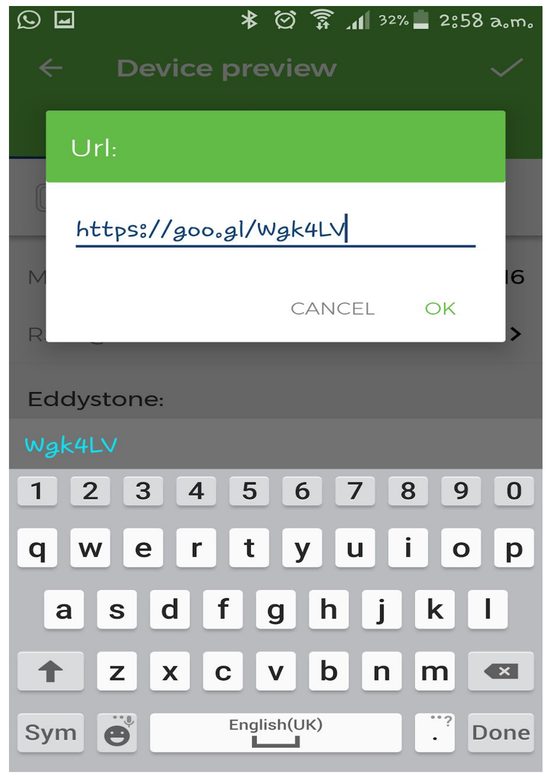

Fig 8.1.2 Configure using a url of type https://

- Kontakt beacons were used for this demonstration. The beacons were configured using the official kontakt.io app on playstore. The SDK available for the same will be used in future implementations while building the admin panel.
- The frame format selected was eddystone url format and a https type link is been configured with this beacon.
- Thus any device enabled with bluetooth and nearby location services enabled was able to catch this notification.

8.2 Fingerprinting

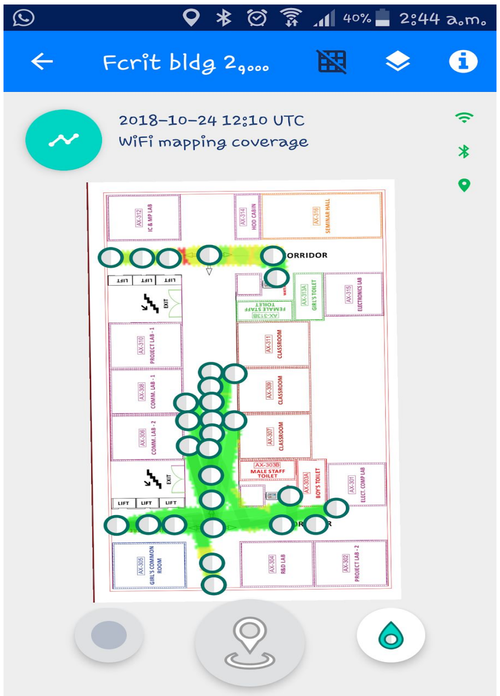

Fig 8.2.1 Wifi Data Heatmap

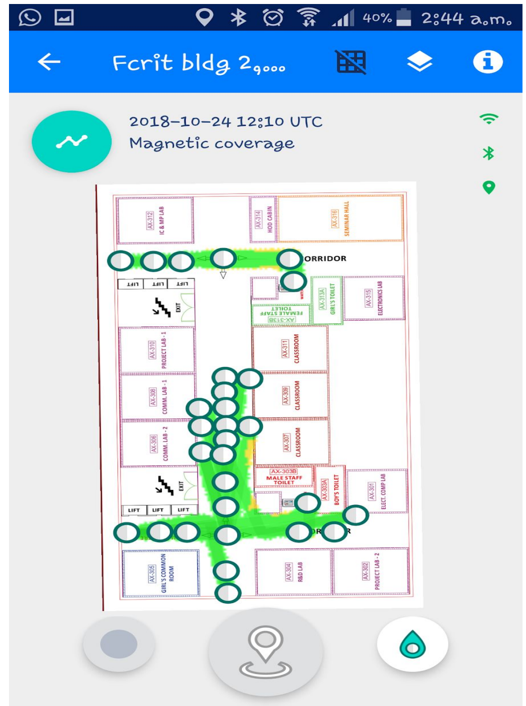

Fig 8.2.2 Magnetic Data Heatmap

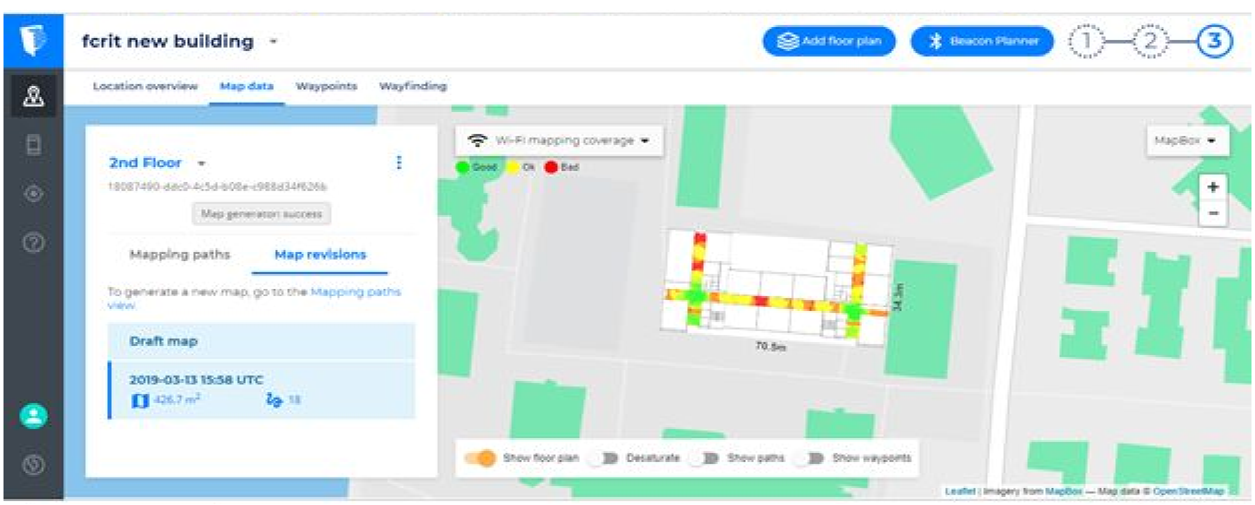

Fig 8.2.3 Wifi Data Heatmap for fourth floor

8.3 Qgis (Georeferencing)

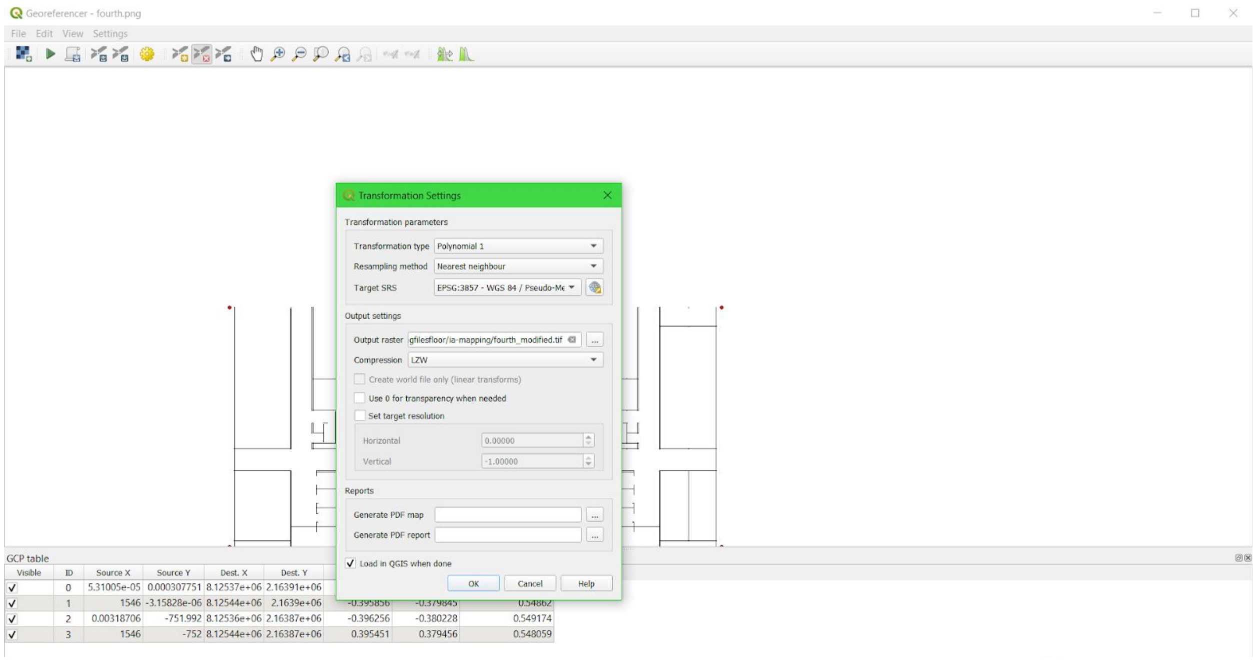

Fig 8.3.1 Loading floor plan image on QGIS

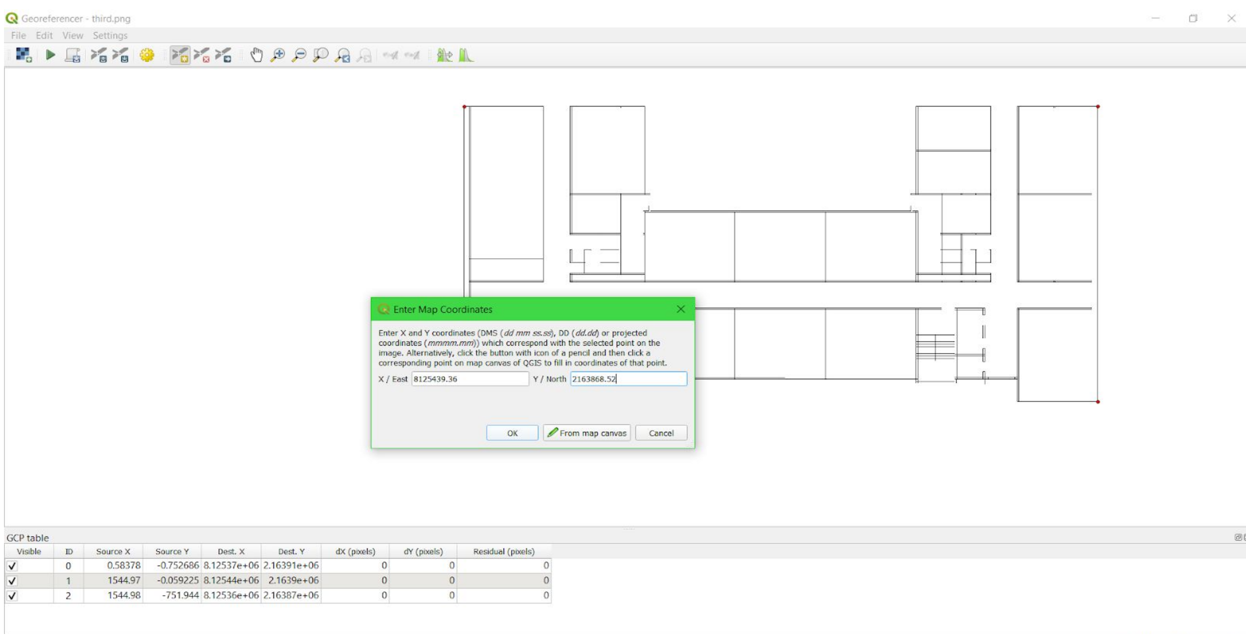

Fig 8.3.2 Floor plan on QGIS Software

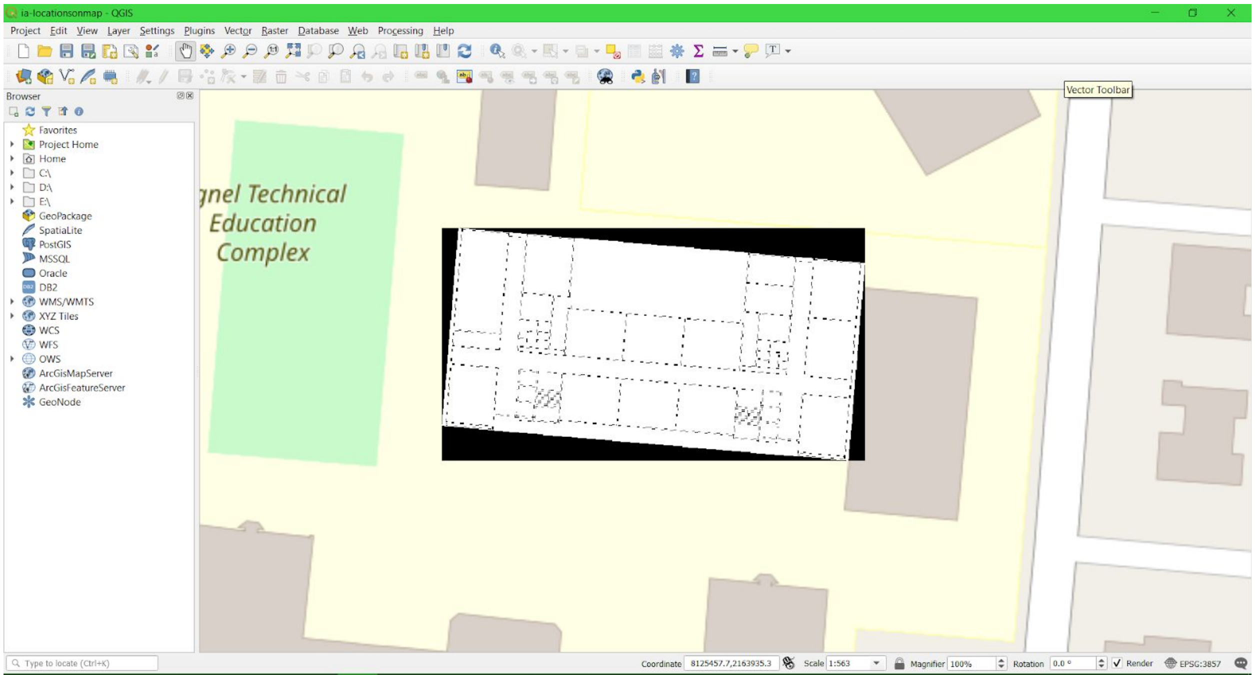

Fig 8.3.2 Floor plan is assigned global coordinates on QGIS Software

8.4 Mapbox 3D extrusion

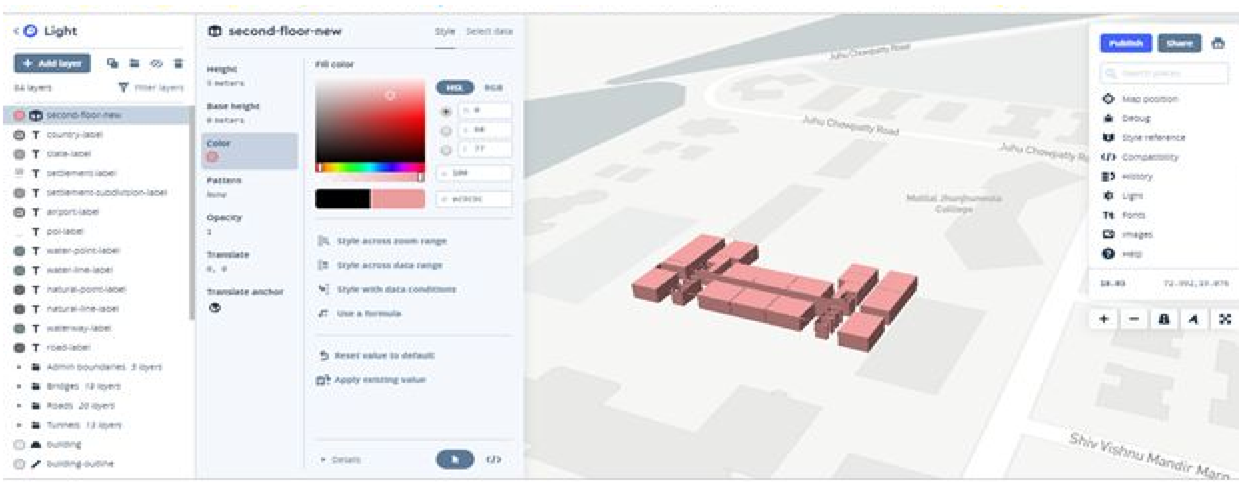

Fig 8.4.1 3D extrusion of floor plan

8.5 Mobile Application

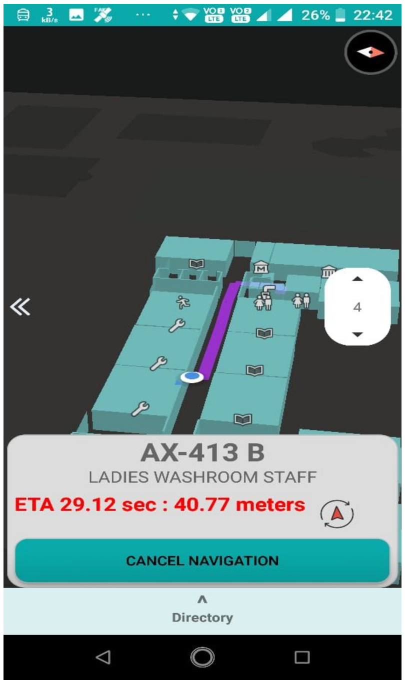

Fig 8.5.1 Navigation in VISIT App

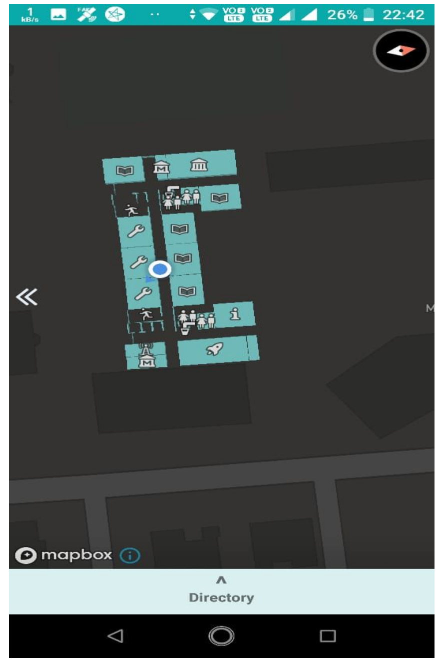

Fig 8.5.2 Locating user

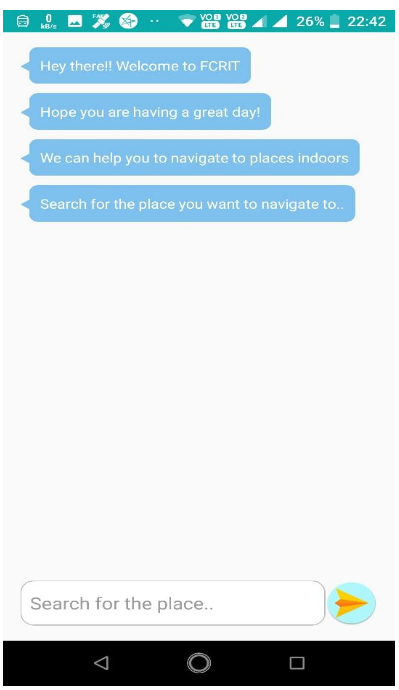

Fig 8.5.3 Chatbot

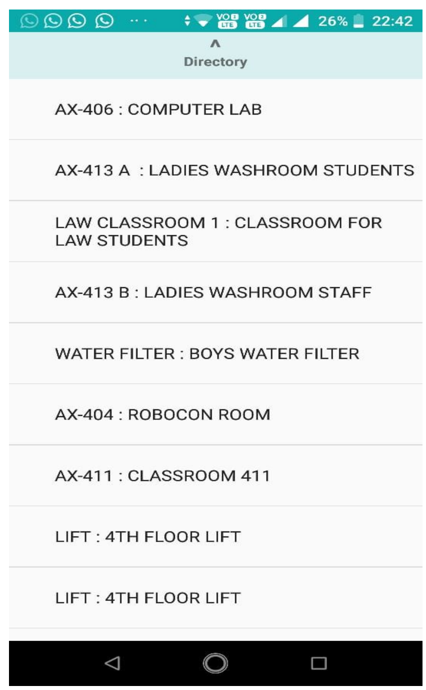

Fig 8.5.4 Drawer

8.6 Admin Panel

Admin panel can be used by the owner to upload the floor maps and feed in the location of his facility. He can monitor the status of his submission from this portal.

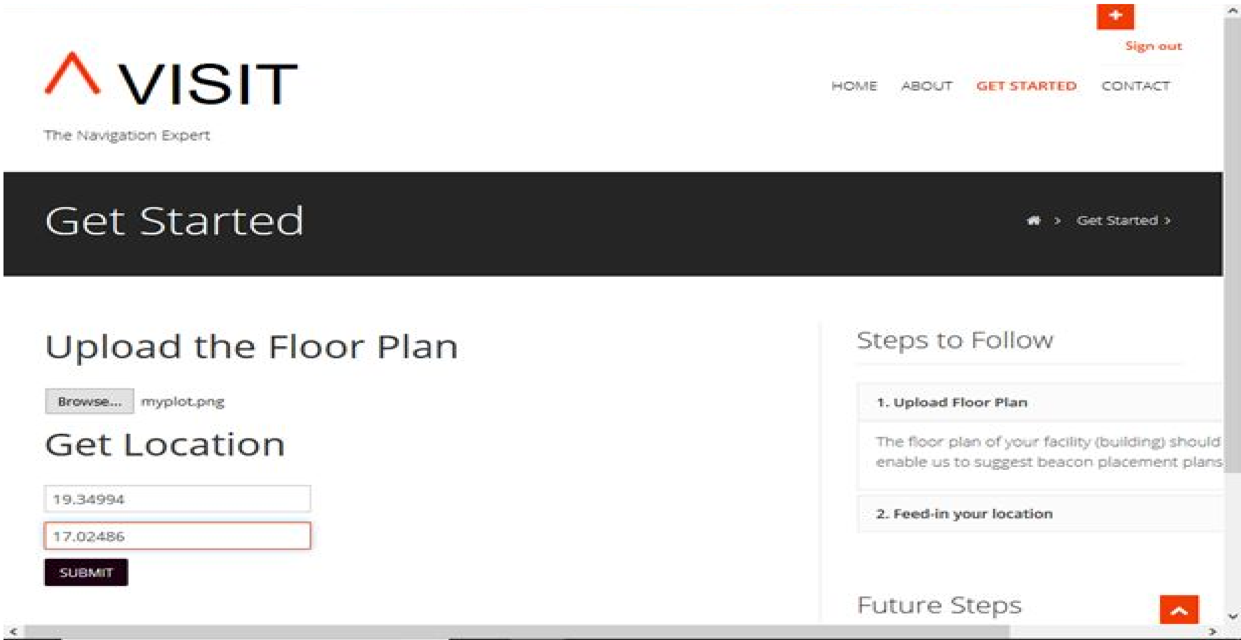

Fig 8.6.1 Upload page in Admin Panel

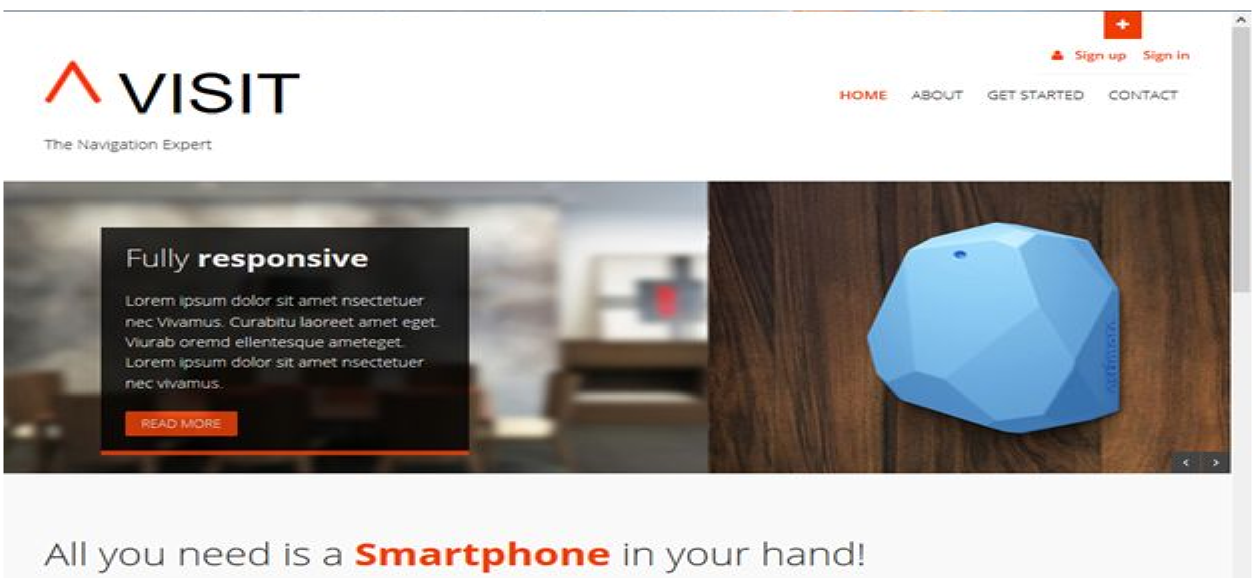

Fig 8.6.2 Homepage of Admin Panel

CHAPTER 9
TIMELINE CHART

CHAPTER 9

TIMELINE CHART

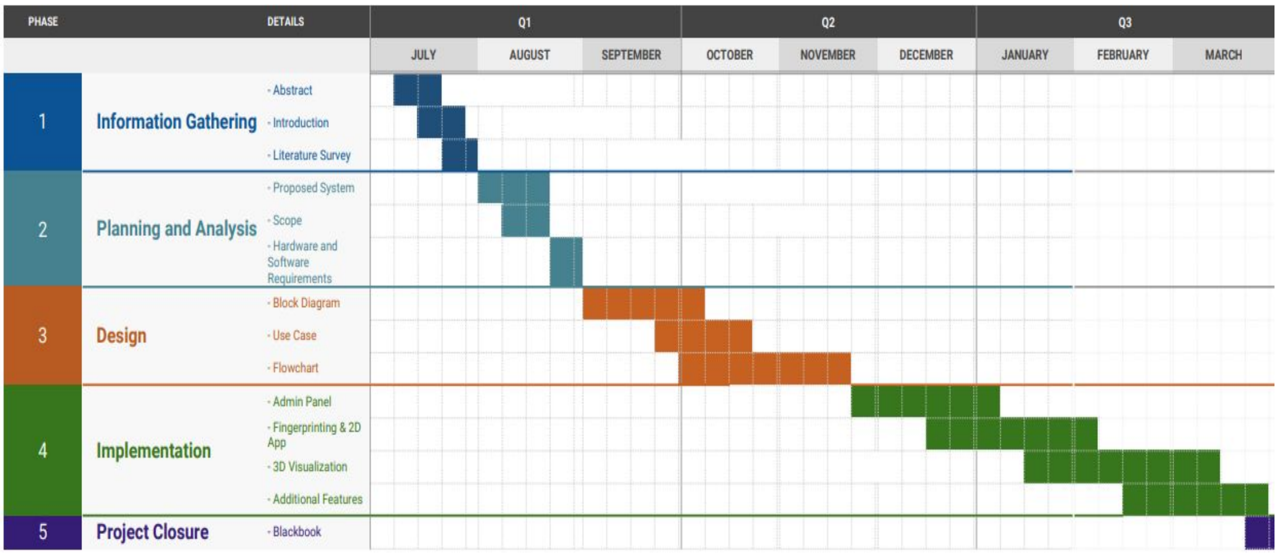

Fig 9.1 Project Timeline Chart

CHAPTER 10

APPENDIX

CHAPTER 10

APPENDIX

Important Code Snippets

1. Getting Location Updates with Android

```
private IALocationListener mIALocationListener = new IALocationListener() {

    // Called when the location has changed.
    @Override
    public void onLocationChanged(IALocation location) {

        Log.d(TAG, "Latitude: " + location.getLatitude());
        Log.d(TAG, "Longitude: " + location.getLongitude());
        Log.d(TAG, "Floor number: " + location.getFloorLevel());
    }
};
```

2. Automatic Outdoor-Indoor Detection

```
private IARegion.Listener mRegionListener = new IARegion.Listener() {
    // when null, we are not on any mapped area
    // this information can be used for indoor-outdoor detection
    IARegion mCurrentFloorPlan = null;

    @Override
    public void onEnterRegion(IARegion region) {
        if (region.getType() == IARegion.TYPE_FLOOR_PLAN) {
            // triggered when entering the mapped area of the given floor plan
            Log.d(TAG, "Entered " + region.getName());
            Log.d(TAG, "floor plan ID: " + region.getId());
            mCurrentFloorPlan = region;
        }
        else if (region.getType() == IARegion.TYPE_VENUE) {
            // triggered when near a new location
            Log.d(TAG, "Location changed to " + region.getId());
        }
```

```
    }

    @Override
    public void onExitRegion(IARegion region) {
        // leaving a previously entered region
        if (region.getType() == IARegion.TYPE_FLOOR_PLAN) {
            mCurrentFloorPlan = null;
            // notice that a change of floor plan (e.g., floor change)
            // is signaled by an exit-enter pair so ending up here
            // does not yet mean that the device is outside any mapped area
        }
    }
};
```

3. Automatic Floor Detection

```
private IARegion.Listener mRegionListener = new IARegion.Listener() {
    IARegion mCurrentFloorPlan = null;
    @Override
    public void onEnterRegion(IARegion region) {
        if (region.getType() == IARegion.TYPE_FLOOR_PLAN) {
            Log.d(TAG, "Entered " + region.getName());
            Log.d(TAG, "floor plan ID: " + region.getId());
            mCurrentFloorPlan = region;
        }
    }
    @Override
    public void onExitRegion(IARegion region) {}
};
```

4. Using Wayfinding

```
private IAWayfindingListener mWayfindingListener = new IAWayfindingListener() {
```

```
    @Override
    public void onWayfindingUpdate(IARoute route) {
        List<IARoute.Leg> legs = route.getLegs();
        // visualize route
    }
};

IAWayfindingRequest wayfindingRequest = new IAWayfindingRequest.Builder()
    .withFloor(1) // destination floor number
    .withLatitude(60.1696597) // destination latitude
    .withLongitude(24.932497) // destination longitude
    .build();

mIALocationManager.requestWayfindingUpdates(wayfindingRequest,
mWayfindingListener);
```

5. On map ready

```
public void onMapReady(@NonNull MapboxMap mmapboxMap) {
    MainActivity.this.mapboxMap = mmapboxMap;
    mapboxMap.setStyle(new
Style.Builder().fromUrl("mapbox://styles/adil-khot/cjrs2yradf2g42tocp5czguq5"),
            style -> {
                mStyle=style;
                final List<Point> boundingBox = new ArrayList<>();

                boundingBox.add(Point.fromLngLat(72.9914219677448, 19.0762151432401));
                boundingBox.add(Point.fromLngLat(72.9920905083418, 19.0761612762958));
                boundingBox.add(Point.fromLngLat(72.9913944751024, 19.075908418288));
                boundingBox.add(Point.fromLngLat(72.9920636862516, 19.075853917514));
                boundingBoxList = new ArrayList<>();
                boundingBoxList.add(boundingBox);

                mapboxMap.addOnCameraMoveListener(() -> {
                    if(mapboxMap.getCameraPosition().zoom < 21){
                        if(cardVisible && !check){
                            poiCard = (LinearLayout)findViewById(R.id.poicard);
                            poiCard.setVisibility(View.GONE);
                            cardVisible=false;
                        }
                    }
                    if (mapboxMap.getCameraPosition().zoom > 18) {
```

```
                if
(TurfJoins.inside(Point.fromLngLat(mapboxMap.getCameraPosition().target.getLongitude(),
                        mapboxMap.getCameraPosition().target.getLatitude()),
Polygon.fromLngLats(boundingBoxList))) {
                    if (scrollableNumberPickerA.getVisibility() != View.VISIBLE) {
                        showLevelButton();
                    }
                }
            } else if (scrollableNumberPickerA.getVisibility() == View.VISIBLE) {
                hideLevelButton();
            }
        });


        indoorBuildingSource = new GeoJsonSource(
                "indoor-building", loadJsonFromAsset("fourth.geojson"));
        style.addSource(indoorBuildingSource);
        featureCollection =
FeatureCollection.fromJson(loadJsonFromAsset("fourth.geojson"));
        List<Feature> featureList = featureCollection.features();
        if(!mPoiList.isEmpty()){
            mPoiList.clear();
        }
        for (int i = 0; i < featureList.size(); i++) {
            if(featureList.get(i).hasProperty("name")){
                mPoiList.add(featureList.get(i).getStringProperty("name")+" :
"+featureList.get(i).getStringProperty("description"));
            }
        }
        adapter.notifyDataSetChanged();
        setupLayer();
        enableLocationComponent(mStyle);
        mapboxMap.addOnMapClickListener(this);

        Intent mainIntent = getIntent();
        int pos = mainIntent.getIntExtra("pos",0);
        setSelected(pos);
    });

ImageView side = findViewById(R.id.side);
side.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        List<Feature> featureListChat=featureCollection.features();
```

```
            if(!chatList.isEmpty()){
                chatList.clear();
            }
            if(!featureListChat.isEmpty()){
                for(int i = 0; i< featureListChat.size(); i++){
                    if(featureListChat.get(i).hasProperty("name")){
                        chatList.add(featureListChat.get(i).getStringProperty("name")+" : "+
featureListChat.get(i).getStringProperty("name"));
                    }
                }
            }
            Intent intent = new Intent(MainActivity.this,chat.class);
            intent.putExtra("list",chatList);
            startActivity(intent);
        }
    });

}
```

6. Load JSON file from Assets

```
private String loadJsonFromAsset(String filename) {

    try {
        InputStream is = getAssets().open(filename);
        int size = is.available();
        byte[] buffer = new byte[size];
        is.read(buffer);
        is.close();
        return new String(buffer, "UTF-8");

    } catch (IOException ex) {
        ex.printStackTrace();
        return null;
    }
}
```

7. Route Visualization

```
private void updateRouteVisualization() {

    clearRouteVisualization();
```

```
    if (mCurrentRoute == null) {
        return;
    }
    int count=0;
    for (IARoute.Leg leg : mCurrentRoute.getLegs()) {

        if (leg.getEdgeIndex() == null) {
            continue;
        }
        count++;
        PolylineOptions opt = new PolylineOptions();
        opt.add(new LatLng(leg.getBegin().getLatitude(), leg.getBegin().getLongitude()));
        opt.add(new LatLng(leg.getEnd().getLatitude(), leg.getEnd().getLongitude()));
        Log.d("Linepoint lat",Double.toString(leg.getBegin().getLatitude()));
        Log.d("Linepoint long",Double.toString(leg.getBegin().getLatitude()));

        if(count==1)
        {
            dirLat=leg.getEnd().getLatitude();
            dirLong=leg.getEnd().getLongitude();
        }
        if (leg.getBegin().getFloor() == mFloor && leg.getEnd().getFloor() == mFloor) {

            opt.color(0xFF9932cc);
        } else {
            opt.color(0x300000FF);
        }

        mPolylines.add(mapboxMap.addPolyline(opt));
    }
}
```

CHAPTER 11
CONCLUSION

CHAPTER 11

CONCLUSION

The Indoor Navigation application provides seamless navigation in an indoor environment for common people. Unlike traditional navigation systems, the indoor map is always available to user and the navigation to user specified destination is provided. All that the user needs is a smartphone. The app gets the user location from the nearby beacon, WiFi and geomagnetic field and navigates him to the destination. This system works according to the map of a location. It will be used especially in big industries, offices, shopping mall or college campus which is unknown to new visitors. Most systems that have been developed so far lack the kind of dynamic interaction and adaptability to changes that our system provides to the user.

CHAPTER 12

PUBLICATIONS

[1] Akshay Bhosale, Gloria Benny, Robin Jaison, Adil Khot, Sandhya Pati, ‘Indoor Navigation System using BLE Beacons’ at the 3rd Biennial International Conference on Nascent Technologies in Engineering (ICNTE) January 4-5, 2019

REFERENCES

[1] Survey of Wireless Based Indoor Localisation Technologies by Junjie Liu, junjie.liu@wustl.edu

[2] An Analysis of the Accuracy of Bluetooth Low Energy for Indoor Positioning Applications by R. Faragher, University of Cambridge, UK; R.Harle, University of Cambridge, UK

[3] Beacon Placement for Indoor Localization using Bluetooth by Sudarshan S. Chawathe

[4] Beacon implemented system for mobile content management by Richard Graves, Chris Blanz, Beat Zenerino, Kevin Huber and Greg Thornton

[5] BLE Deployment Density by R. Faragher, University of Cambridge, UK; R. Harle, University of Cambridge, UK

[6] http://web.indooratlas.com/web/WhitePaper.pdf

[7] Geomagnetic Field Based Indoor Localization Using Recurrent Neural Networks by Ho Jun Jang, Jae Min Shin, and Lynn Choi, School of Electrical Engineering Korea University, Seoul, Korea

[8] Geomagnetic Fingerprint Maps for Indoor Positioning by Xiaolong Xu and Licheng Lin , Nanjing University of posts and Telecommunications. Nanjing,China

[9] https://www.qgistutorials.com/en/docs/georeferencing_basics.html

[10] Automatic Floor Plan Analysis for Adaptive Indoor Wifi Positioning System by Xuning Zhang, Albert Kai-sun Wong

[11] https://developers.google.com/beacons/get-started

[12] https://docs.mapbox.com/studio-manual/overview/

[13] https://kontakt.io/products-and-solutions/beacon-software/

ACKNOWLEDGEMENT

We would like to thank our principal, **Dr. S.M. Khot** for giving the opportunity and permission to undertake this project.

We would also like to thank **Dr. Lata Ragha**, Head of Computer Engineering Department, for leading her full cooperation, despite her busy schedule, for the deployment of this project.

We take immense pleasure to thank our project guide **Mrs. Sandhya Pati** for her constant supervision as well as for providing necessary information regarding our project. We would like to take this opportunity to thank her for the guidance and support throughout.

We would also like to thank our project coordinator **Mrs. Rakhi Kalantri** for providing us with regular inputs about documentation and project timeline.

At last we must express our sincere heartfelt gratitude to all other teaching and non teaching staff members who helped us directly or indirectly during this course of work.

1. Akshay Bhosale 101506
2. Gloria Benny 101512
3. Robin Jaison 101525
4. Adil Khot 101537