

Pramesh Dhimal

LOCATION TRACKING USING BLUETOOTH TECHNOLOGY

LOCATION TRACKING USING BLUETOOTH TECHNOLOGY

Pramesh Dhimal
Bachelor's Thesis
Spring 2019
Degree Programme in
Information Technology
Oulu University of Applied Sciences

ABSTRACT

Oulu University of Applied Sciences
Degree Programme in Information Technology

Pramesh Dhimal

Title of the bachelor's thesis: Location Tracking using Bluetooth Technology

Supervisor: Janne Kumpuoja

Term and year of completion: Spring 2019

Number of pages: 34

Tracking the location of a certain specific object has become very popular these days. There are various technologies used for location tracking like GPS, RFID, and Bluetooth. Among them, Bluetooth technology is a new and emerging technology for indoor(short-range) locations tracking. The urge to learn new cutting-edge technology inspired me to choose this topic.

The main objective of this thesis was to track some specific object using the Bluetooth Low Energy(beacon) technology and to develop an app to demonstrate its use in the real world. The application was developed with Xcode. The Estimote Notification template was used and modified to get the desired output. Different online documentation was referred to understand the Bluetooth technology and gained a brief knowledge about BLE Beacons. Finally, the application was tested using iPhone6s.

A desk-info app was developed during this project to demonstrate the use of BLE beacons for getting information about the desk owner when the user approaches the particular desk. This app can be further improved by adding some other features like triggering some action within the application.

Keywords: Location Tracking, Bluetooth Technology, Beacons

CONTENTS

ABSTRACT	3
TABLE OF CONTENTS	4
VOCABULARY	6
1 INTRODUCTION	7
2 INTRODUCTION TO INDOOR POSITIONING & NAVIGATION	9
2.1 Existing ways of Indoor Positioning and Navigation	9
2.1.1 WAN/WiFi-based indoor positioning system	10
2.1.2 RFID indoor positioning system	10
2.1.3 Bluetooth based indoor positioning system	11
3 BLUETOOTH POSITIONING	13
3.1 Beacon	13
3.1.1 iBeacon	13
3.1.2 Eddystone	14
4 IMPLEMENTATON	17
4.1 Introduction to Desk-Info Application	17
4.2 Tools and Technology Used	18
4.2.1 Mac OS	18
4.2.2 iPhone	18
4.2.3 Xcode	18
4.2.4 GitHub	18
4.2.5 Apple ID and Apple Developer ID	18
4.2.6 Estimote Beacon	19
4.2.7 Estimote Account	19
4.2.8 Estimote Cloud	20
4.2.9 Estimote iOS App	20
4.2.10 Estimote App	20
4.2.11 App ID	20
4.2.12 App Token	21
4.2.13 Estimote iOS SDK	21
4.2.14 CocoaPods	21
4.3 Setting up the beacon	21

4.4 Setting up an iOS application Project	22
4.5 Code Implementation	23
4.5.1 Enabling Bluetooth Background Mode	23
4.5.2 Setting the location services usage description	24
4.5.3 Importing the Proximity SDK	24
4.5.4 Setting up Cloud Credentials	25
4.5.5 Notification Setting	25
4.5.6 Creating a Proximity Observer Object	26
4.5.7 Creating Proximity Zone Objects	26
4.5.8 Starting the proximity Observation	27
4.6 Deploy on iPhone	28
4.7 Observation	29
5 CONCLUSION	31
REFERENCES	32

VOCABULARY

AOA	Angle of Arrival
BLE	Bluetooth Low Energy
dBm	Decibel-milliwatts
GHz	Gigahertz
GNSS	Global Positioning System
GPS	Global Positioning System
IEEE	Institute of Electrical and Electronics Engineers
LBS	Location Based Service
MHz	Megahertz
RF	Radio Frequency
RFID	Radio Frequency Identification
RSSI	Received Signal Strength Indicator
TDOA	Time Difference of Arrival
WLAN	Location Based Service

1 INTRODUCTION

Indoor location Tracking is becoming popular these days as GPS fails to work correctly inside the building due to different interference. Indoor location and tracking have now reached a new height in the present-day, but yet it is still in the evolving phase. According to a 2016 Global Research Report, by 2019 the estimated value of the indoor positioning market will increase to \$4.4 billion (IndoorAtlas, 2016). This means that there will be more use of an indoor positioning system for various purposes. Auchan, a Ukrainian hypermarket, uses beacons to enable customers to receive notifications for finding a way, and promoting as they move through the store. It gives an option to its consumers to navigate themselves through the store with the help of an indoor map. As the consumer clicks on "View on Map", the map directs them to aisles or a product they desire. (Adarsh, 2017.)

Bluetooth is one of the technologies used for Indoor Positioning and Navigation. Beacons are small Bluetooth Low Energy (BLE) devices that broadcast packets of data in short intervals. These packets contain information about the beacon, as well as telemetry readings commonly used in distance calculations. Mobile devices, such as tablets and smartphone, can be used to pick up BLE signals and read the broadcasted data, which in turn can be used on indoor navigation applications. (Herrera Vargas, 2016, 6.)

While the use of an Embedded System has also been very popular and also essential in people's day to day life, it has made people's life more accessible and comfortable. Knowing or unknowing people who are using this system in their day to day life, e.g., Smartphone, Camera, traffic lights, home security, alarm system and many more. Beacon is Bluetooth low energy devices that repeatedly broadcast an identifier and that identifier marks a vital place or an important object in a way that the user's device understands.

The topic of the thesis was given from OUAS. The primary objective of this bachelor's thesis was to study the use of BLE Beacons for Indoor Positioning and develop a simple mobile application to demonstrate its use in the real

world. This document also contains a brief description on Indoor Positioning System, different techniques used in the indoor positioning system, different types of the beacon and its uses.

2 INTRODUCTION TO INDOOR POSITIONING & NAVIGATION

GPS is widely used for positioning and navigation outdoor. But when it comes to indoor, it fails because of physical barriers and other interference. An Indoor Positioning System is a solution to indoor positioning and navigation. An Indoor Positioning System (IPS) is a system to locate objects or people inside a building using radio waves, magnetic fields, acoustic signals, or other sensory information collected by mobile devices (Wikipedia, Indoor Positioning System, cited 15.07.2018). There are different ways and technology used for indoor positioning and navigation, and the most commonly used ones are WiFi positioning, Bluetooth(BLE) or positioning based on magnetic fields.

An Indoor Positioning System is beneficial in a larger commercial or public building where it is complicated to locate and track specific objects or people. For example, IPS can be used in the hospital to track and locate dementia patients. It can also be used in the University for locating the conference hall and lecture rooms. It can also be used in the supermarket for finding different products.

The accuracy of the indoor positioning system depends on the chosen positioning technology. The type of building being mapped also influences the accuracy of the indoor positioning system.

2.1 Existing ways of Indoor Positioning and Navigation

There are different ways and technology used for indoor positioning and navigation, but there is no standard for an IPS system. IPSes use different technologies, including distance measurement to nearby anchor nodes(node with known positions, e.g., WiFi access points or Bluetooth beacons), magnetic positioning, dead reckoning. (Wikipedia, Indoor Positioning System, cited 15.5.2018.)

2.1.1 WAN/WiFi-based indoor positioning system

There are numerous commercial applications for Indoor Positioning available on the market that are based on the WAN/Wi-Fi technology. Some of the applications combine multiple technologies, such as Wi-Fi, GPS, mobile phone signal and Bluetooth. But they all claim to use Wi-Fi technology as their main approach.

“WLAN (Wireless Local Area Networks, IEEE 802.11 standard; ‘Wi-Fi’ is used interchangeably or as a superset of IEEE 802.11 and denotes the registered trademark of the Wi-Fi Alliance) can be used to estimate the location of a mobile device with this network. The use of WLAN signals is a tempting approach since WLAN access points are readily available in many indoor environments and it is possible to use standard mobile hardware devices. The range of 50 m to 100m which is typically covered by WLAN outreaches that of Bluetooth or RFID, Another advantage of using WLAN is that line of sight is not required, ToA, TDoA or AoA methods are less common in WLAN due to the complexity of time delay and angular measurements. The most popular WLAN positioning method is to make use of RSSI (Received Signal Strength Indicators) which are easy to extract in 802.11 networks and can run on off-the-shelf WLAN hardware. Therefore, the WLAN positioning system has become the most widespread approach for indoor localization.”(Mautz, 2012, 57.)

2.1.2 RFID indoor positioning system

“RFID (Radio Frequency Identification) is a common term for radio-frequency technologies used for detecting, identifying products and things. The technology is based on storing information on the RFID tag and its wireless reading through an RFID reader using radio waves. Simpler RFID tags are wireless memory devices. The idea of the RFID system is simple: RFID tags stores data content, they are attached to the desired targets and are read by RFID readers that typically transmit information to the background system. If necessary, the tag information can be updated using RFID reader.”(RFID lab, cited 14.5.2018.)

“There are passive, semi-passive and active RFID-tags. Passive tags have no own power source, e.g., a battery. For that reason, the tags can be made very small to fit smaller products. The passive tags receive the operational power, e.g., from the corresponding interrogator. Active tags typically have an internal battery and a transmitter or a transceiver. They have a long range, and that is why they can be used for real-time location system and indoor positioning.”(Alatalo 2012, 25.)

2.1.3 Bluetooth based indoor positioning system

Bluetooth is a wireless communication technology that allows electronic devices to communicate over small distances. Bluetooth frequency is allocated within the 2.4 GHz ISM band. Despite operating on an open band, the Bluetooth technology is regulated by the "Bluetooth Special Interest Group" (SIG) and was standardized by IEEE as IEEE 802.15.1, even though it no longer maintains the standard. Some common examples of Bluetooth devices are speakers, mobile headsets, game controller, wireless mouse, and keyboard.(Herrera Vargas. 2016,13.)

Bluetooth uses 79 channels to transmit data. The ISM band typically extends from 2400 MHz to 2483.5 MHz (i.e., 2.4000-2.4835 GHz). The Bluetooth channels are spaced 1 MHz apart, starting at 2402 MHz and finishing at 2480 MHz. This can be calculated as $2401+n$, where n varies from 1 to 79. (Poole. Electronics notes. cited 30.08.2018.) In order to avoid interference from other RF signals, a technique known as frequency hopping is used, where data is transmitted over one of the available channels for a small period of time and resent over another channel in case of an interference(Herrera Vargas. 2016, 13).

Bluetooth devices are low power devices and use battery power to operate over relatively short distances. The transmitter powers for Bluetooth are quite low. Three different classes are used to classify Bluetooth devices according to their maximum output power and their corresponding maximum range.

TABLE 1. Bluetooth Classes and their corresponding range (Herrera Vargas. 2016,14)

Class	Maximum Output Power	Range
1	20 dBm	Up to 100 m
2	4 dBm	Up to 10 m
3	0 dBm	10 cm

The technology developed within Bluetooth made it possible to use its technology for indoor navigation. During the last decade, Bluetooth sensors were designed to only broadcast data without the need of connection have become very popular, especially in the fields of healthcare and sports. In addition to these kinds of devices, a new communication trend known as the Internet of Things(IoT) began encouraging the use of small sensors equipped on everyday objects, enabling them to communicate with each other. Smaller sensors running on smaller power sources started emerging and the need for even lower power consumption option become evident. In answer to this need, a new Bluetooth standard named Bluetooth Low Energy(BLE), also known as Bluetooth Smart, was introduced. (Herrera Vargas. 2016,14,15).

According to the Bluetooth Special Interest Group, Bluetooth is all about proximity, not about the exact location. Bluetooth was not intended to offer a pinned location like GPS, however it is known as a geo-fence or micro-fence solution which makes it an indoor proximity solution, not an indoor positioning solution. Micro-mapping and indoor mapping has been linked to Bluetooth and the Bluetooth LE-based iBeacon promoted by Apple Inc. A large-scale indoor positioning system based on iBeacon has been implemented and applied in practice (Wikipedia, Indoor Positioning System, cited 15.5.2018).

3 BLUETOOTH POSITIONING

3.1 Beacon

Beacons are small, often inexpensive devices that enable more accurate location with a narrow range than GPS, cell tower triangulation and Wi-Fi proximity. Beacons transmit small amounts of data via Bluetooth Low Energy (BLE) and as a result are often used for the indoor technology, although beacons can be used outside as well. (Stroud. cited 22.5.2018.)

There are different types of beacons developed by different companies for the commercial use. Most of the beacons implement both iBeacon and Eddystone protocols which are used by iOS and the Android platform.

3.1.1 iBeacon

iBeacon is a protocol developed by Apple and introduced at the Apple Worldwide Developers Conference in 2013. Since then, various vendors have made iBeacon-compatible hardware transmitters. (Wikipedia, cited 16.12.2018.) iBeacon is just an advertising protocol that tells what data, and in what format, a Bluetooth beacon needs to advertise. iBeacon has native, built-in support in iOS, via iOS Location Services and the Core Location framework. The Core Location API allows to detect nearby iBeacon devices, as well as listens for enter/exit events, fired when the user enters and exists the range of an iBeacon device. (Estimote Inc. cited 16.12.2018.)

An iBeacon advertisement provides the following information as shown in the figure 1 below via Bluetooth Low Energy:

Field	Size	Description
UUID	16 bytes	Application developers should define a UUID specific to their app and deployment use case.
Major	2 bytes	Further specifies a specific iBeacon and use case. For example, this could define a sub-region within a larger region defined by the UUID.
Minor	2 bytes	Allows further subdivision of region or use case, specified by the application developer.

FIGURE 1. *iBeacon advertisement information (Apple Inc. 2014,3)*

The UUID, major and minor values provide the identifying for the iBeacon. This information is hierarchical in nature with the major and minor fields allowing for subdivision of the identity established by the UUID. UUIDs can be generated by using the `uuidgen` command line utility in OS X, or programmatically using the `NSUUID` Foundation class. (Apple Inc. 2014, 3.)

3.1.2 Eddystone

Eddystone is a protocol specification released by Google in July 2015 that defines a Bluetooth Low Energy (BLE) message format for proximity beacon messages. It describes several different frame types that may be used individually or in combinations to create beacons that can be used for a variety of applications. The Eddystone protocol has four different packet frame formats unlike iBeacon protocol where only one format of packet is transmitted. The different kinds of packets are Eddystone-UID, Eddystone-TLM and Eddystone-EID. The frame types are identified by the byte value in a packet and parsed accordingly. The figure 2 shows the byte values of the frame types.

Frame Type	High-Order 4 bits	Byte Value
UID	0000	0x00
URL	0001	0x10
TLM	0010	0x20
EID	0011	0x30
RESERVED	0100	0x40

FIGURE 2. Byte values of different Eddystone frame types (Google Inc. Eddystone cited 15.12.2018)

The Eddystone-UID frame broadcasts an opaque, unique 16-byte Beacon ID composed of a 10-byte namespace and a 6-byte instance. The namespace is intended to ensure ID uniqueness across multiple Eddystone implementers and may be used to filter on-device scanning for beacons. An instance ID can function like the UUID, major and minor numbers in iBeacon protocol. (Google Inc. Eddystone-UID, cited 15.12.2018.)

The Eddystone-URL frame broadcasts a URL using a compressed encoding format in order to fit more within the limited advertisement packet. The Eddystone-URL frame forms the backbone of physical Web. After decoding the encoded format it redirects the user to the decoded URL with access to the internet. The figure 3 shows the frame type specification.(Google Inc. Eddystone-URL, cited 15.12.2018.)

Byte offset	Field	Description
0	Frame Type	Value = 0x10
1	TX Power	Calibrated Tx power at 0 m
2	URL Scheme	Encoded Scheme Prefix
3+	Encoded URL	Length 1-17

FIGURE 3. Eddystone-URL frame type specification (Google Inc. Eddystone-URL, cited 15.12.2018)

The Eddystone-TML broadcast information about the beacon. This information consists of different information about beacons such as device temperature, battery level and other sensor data. The Eddystone-TML can also be encrypted before broadcasting and only trusted device or user can decrypt the information. (Google Inc. Eddystone-TML, cited 15.12.2018.)

Eddystone-EID is the latest addition to the types of frame which was introduced in mid-2016. The Eddystone-EID frame broadcasts an encrypted ephemeral identifier that changes periodically at a rate determined during the initial registration with a web service. The broadcast ephemeral ID can be resolved remotely by the service with which it was registered, but to other observers appears to be changing randomly. This frame type is intended for use in security and privacy-enhanced devices. (Google Inc. Eddystone-EDI, cited 15.12.2018.)

4 IMPLEMENTATION

4.1 Introduction to Desk-Info Application

Desk-info is a simple iOS mobile application built during this project to demonstrate the use of Bluetooth BL beacons to get information through the notification about the zone user enters. In a large corporate or offices where there are large numbers of employees, user can get information about the desk owner with the help of this app through a notification as shown in the figure 4.

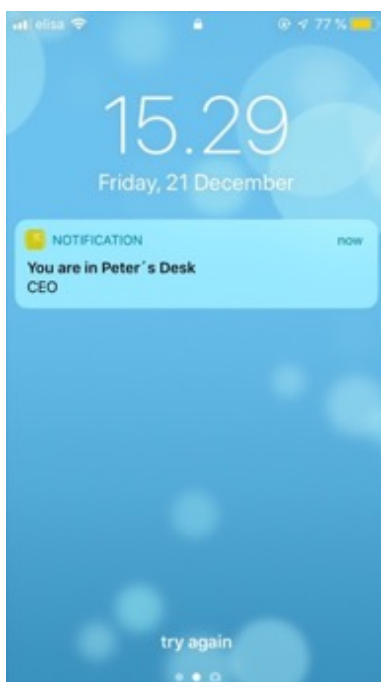


FIGURE 4. Notification when a user enters the zone.

This app was developed using tools and technology as mentioned in the section 4.2. As it is a native iOS application, it works only on the iOS platform, i.e., on iPhone and iPad. It also works when the mobile is locked, but the user should give permission to access when the screen is locked while using the app for the first time or the user can also allow access from the device setting. This app helps the user to get the right information about the desk owner.

4.2 Tools and Technology Used

4.2.1 Mac OS

It is the Operating System developed by Apple for its Macintosh Computer. The desk-info app was developed using Mac OS operated Macbook.

4.2.2 iPhone

iPhone is a smartphone manufactured by Apple that uses its own iOS operating system. iPhone 6s was used for running and testing the desk-info app during this project.

4.2.3 Xcode

Xcode is an Integrated Development Environment(IDE) for macOS. It contains a suite of software development tools developed by Apple for developing software for Apple products like macOS, iOS, watchOS, and tvOS. Xcode is available in Mac App Store free of charge for macOS high Sierra and macOS Mojave users. (Wikipedia XCode, cited 15.12.2018.) It was used for developing desk-info iOS application using the Swift programming language.

4.2.4 GitHub

GitHub is a web-based version control application. Many developers can work on the same project simultaneously using the GitHub repository. It was used to get different tools from the GitHub repository while developing the Desk-info application.

4.2.5 Apple ID and Apple Developer ID

Apple ID is an authentication system for accessing Apple's product. The user has to create an Apple account to get access to Apple product like iWork, iTunes Store and Apple Store. Creating an Apple ID is free and can be created by registering on the official webpage of Apple.

Apple Developer ID is also an authentication system for iOS developer who wants to develop and release their application in the App Store. The user has to

enroll and pay a certain amount to get an Apple developer account. The user can enroll as an individual if he/she is a sole developer or he/she can enroll as an Organization where there can be many developers. In the case of Organization, the team leader can invite another developer through email. Once the developer accepts the request, they can start developing and testing the application.

For this project OUAS developer account was used for running the Desk-info application in iPhone. After accepting the request as the Apple developer team member from OUAS, OUAS as a team was selected in Xcode as shown in the figure 5.

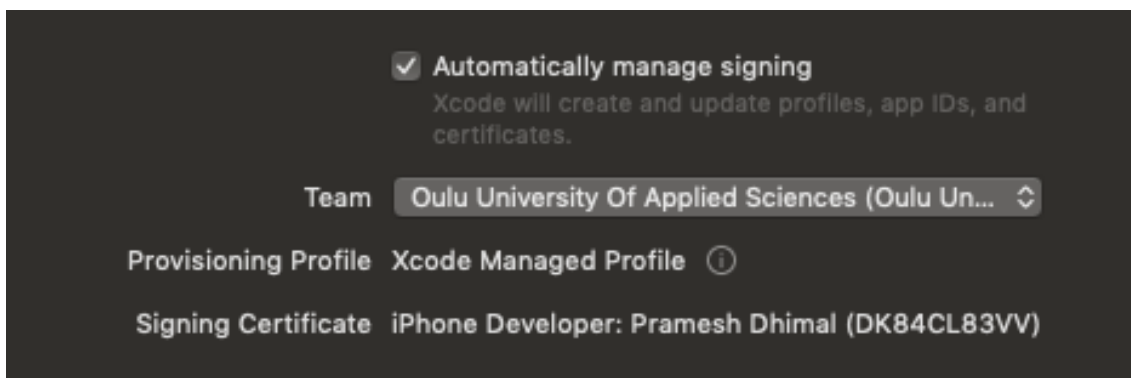


FIGURE 5. Signing as a team member.

4.2.6 Estimote Beacon

Estimote Beacons are the beacon developed by Estimote Inc. They have released different kinds of beacon that are available in the market. They can be bought by ordering through their official webpage. For this project two Location UWB beacon was used. Location UWB supports both iBeacon and Eddystone protocols.

4.2.7 Estimote Account

Estimote Account is an authentication system for accessing Estimote's product and managing beacons with the help of the Estimote Cloud or Estimote iOS or Android Application. It contains a username and password. Estimote Account

was created by registering in the official webpage of the estimate i.e.,

<https://cloud.estimate.com/#/signup>

4.2.8 Estimate Cloud

Estimate Cloud is the cloud platform maintained and developed by Estimate Inc. It is a web-based platform for managing beacons. Estimate ID is required to get access to Estimate Cloud. It allows the user to remotely access settings of beacons and locations saved with Indoor Location SDK. (Anderson. 2017, cited 20.12.2018.) It also helps to create and manage Estimate apps. It was used for the ownership transfer of beacons and also creating and managing desk-info app.

4.2.9 Estimate iOS App

Estimate iOS App is an application used for the management of Estimate Beacons. It connects over Bluetooth to a device to manage the setting or access the built-in sensors. This app works with the Estimate Proximity Beacons, Location Beacons with UWB, Mirror, and Stickers. This application is available in App Store. Estimate iOS App was installed on iphoe6s and was used for updating firmware and managing beacons.

4.2.10 Estimate App

Estimate apps are the app created on top of Estimate SDK. Using Estimate API, developers can build proximity or indoor location applications and trigger pre-programmed actions or send relevant messages. To use Estimate Cloud API requires an App ID and App token which is created while making an app through Estiomote Cloud or Estimate iOS app.

4.2.11 App ID

Estimate App Id is log in to Estimate Account so that Estimate Cloud can authenticate the user and authorize access. App Id can be created through estimate cloud Apps section by login with Estimate Account.

4.2.12 App Token

Estimote App Token is the password to Estimote Account so that Estimote Cloud can authenticate the user and authorize access. When creating an estimate app App Token is automatically generated.

4.2.13 Estimote iOS SDK

Estimote iOS SDK is a complete Software Development Kit for developing Estimote applications. The desk-info app was developed using Estimote Monitoring and Proximity SDK

4.2.14 CocoaPods

CocoaPods is an application level dependency manager for the Objective-C, Swift and any other languages that run on the Objective-C runtime, such as RubyMotion, that provides a standard format for managing external libraries (Wikipedia CocoaPods, cited 20.12.2018). CocoaPods is built with Ruby and is installable with the default Ruby available on macOS. CocoaPods was installed using the following code as shown in the figure 6.

```
Prameshs-MacBook-Pro:~ Pramesh$ sudo gem install cocoapods
```

FIGURE 6. Cocoapods installation code.

4.3 Setting up the beacon

This project was developed with two UWB Estimote beacons which were provided by the school along with their full ownership. A user can only get access to beacon setting with the full ownership of those beacons. First of all, Estimote Account was created as mentioned in the section 4.2.7. Similarly, iOS Estimate app was also downloaded and installed in the device through the app store. With the help of estimate app, those two UWB beacons firmware was updated and checked if those beacons were broadcasting Estimote Monitoring and iBeacon protocol.

After updating firmware and confirming that it is broadcasting estimote monitoring and iBeacon protocol, tags and attachments were set up with the following steps.

- Login to Estimote Cloud with Estimote Account
- Then one of the beacon was selected.
- To set up a tag: Click on “Tags”, “Create New Tag”, enter “desk”, and click “Add”.
- To set up attachment: On the left side, select “Beacon Attachment”. Then, add a “desk-owner” key with a value set to “Peter”.

For the second beacon, the same steps were followed by using the same “desk” tag but changing the “desk-owner” value to “Alex”.

Now beacons were attached to the physical desk and are also ready for broadcasting packets.

4.4 Setting up an iOS application Project

First of all, the latest version of Xcode was downloaded from the app store. A new app called “desk-info” app was added choosing the Notification app template by logging in to Estimote Cloud. During adding this app swift programming language was chosen as a development platform and those two UWB beacons was also associated with "desk-info" app. The template was downloaded to a local repository. App ID and App token were also created. Estimote app ID and App token are used to connect to Estimote Cloud API and they work as a sort of login and password in Estimote SDK. The “desk-info” app contains the following file hierarchy as shown in the figure 7.

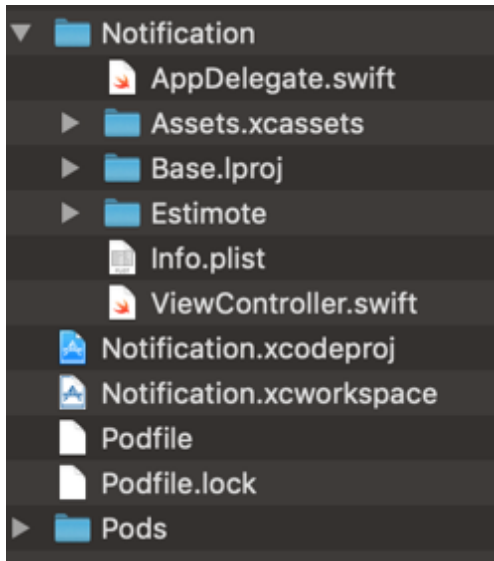


FIGURE 7. File hierarchy of Desk-info app (Estimote Inc. cited 16.12.2018)

Notification.xcworkspace was opened in Xcode and the iOS application project was ready for further code implementation.

4.5 Code Implementation

4.5.1 Enabling Bluetooth Background Mode

Enabling Bluetooth Background Mode in Xcode project allows the app to run in the background when in a range of beacons. The Notification template already had Bluetooth Background Mode enable in it. The Bluetooth Background Mode can be enabled with the following steps.

- Select the project in the left sidebar, and on the right, go to “Capabilities”.
- Flip the Background Mode to “ON”.
- Expand the Background Modes selection and check the “Use Bluetooth LE accessories” option.

4.5.2 Setting the location services usage description

Proximity SDK requires iBeacon Location Services to work in the background. Location services use were described in the info.plist file as shown in the figure 8.

```
<key>NSLocationAlwaysAndWhenInUseUsageDescription</key>
<string>We'll show you notifications when entering and leaving proximity of
your beacon. (You can change this message in the Info.plist file.)</string>
<key>NSLocationAlwaysUsageDescription</key>
<string>We'll show you notifications when entering and leaving proximity of
your beacon. (You can change this message in the Info.plist file.)</string>
<key>NSLocationWhenInUseUsageDescription</key>
<string>We'll show you notifications when entering and leaving proximity of
your beacon. (You can change this message in the Info.plist file.)</string>
```

FIGURE 8. : Location Service uses description (Estimote Inc. cited 16.12.2018)

- NSLocationWhenInUseUsageDescription: It describes how the app uses Location Service when it is in use.
- SNLocationAlwaysUsageDescription: It describes how the app uses Location Service when in use and in the background. This description is only for the app with iOS 10 or earlier. The user can agree to the “always” authorization or disable Location Services in the app.
- NSLocationAlwaysAndWhenInUseUsageDescription: It describes how the app use the Location Service both when in use and in the background. This description is only for the app with iOS 11. The user can select between the “always” or “only when in use” authorizations, or disable Location Services in the app.

4.5.3 Importing the Proximity SDK

The desk-info app uses Proximity SDK. Proximity SDK was imported with the code as shown the figure 9.

```
import UIKit
import EstimoteProximitySDK
```

FIGURE 9. Use of AppDelegate and AppDelegate (Estimote Inc. cited 16.12.2018)

4.5.4 Setting up Cloud Credentials

Cloud Credentials allows estimate SDK to communicate with Estimote Cloud. It will enable changing the beacon setting in the cloud without breaking in the app code. App Id and App token are used to connect to Estimote Cloud API. App Id and App token were used in the following way as shown in the figure 10.

```
let estimoteCloudCredentials = CloudCredentials(appID:
    "infoapp-985", appToken: "4a30fc7468ef87a68fd1592ccf62edd3")
```

FIGURE 10. Use of AppId and AppToken

4.5.5 Notification Setting

The notification is the core function of desk-info app. Before using the notification function, it has to be imported by with the following code as shown in the figure 11.

```
import UserNotifications
```

FIGURE 11. Importing Notifications

Now to show the notification, the app has to ask permission from the user which was performed with the following code as shown in the figure 12.

```
let notificationCenter = UNUserNotificationCenter.current()
notificationCenter.delegate = self
notificationCenter.requestAuthorization(options: [.alert, .sound])
{ granted, error in
    print("notifications permission granted = \(granted), error = \(
        error?.localizedDescription ?? "(none)")")
}
```

FIGURE 12. Code for permission request (Estimote Inc. cited 16.12.2018)

4.5.6 Creating a Proximity Observer Object

The Proximity Observer defines the zone where some action will be performed when predefined conditions are met. Each zone is built around a tag. One or more beacons can have the same tags associated, making a zone consist of more than one beacon. (Estimote Inc. Proximity Observer, cited 21.12.2018.)

Proximity Observer was defined by adding a property to hold the Proximity Observer as shown in the figure 13.

```
var proximityObserver: ProximityObserver!
```

FIGURE 13. Code for defining Proximity Observer

And the Proximity Observer was created with the following codes as shown in the figure 14.

```
proximityObserver = ProximityObserver(credentials:
    estimoteCloudCredentials, onError: { error in
    print("ProximityObserver error: \(error)")
})
```

FIGURE 14. Code for creating Proximity Observer

4.5.7 Creating Proximity Zone Objects

Proximity zone objects were created associating the desks tag. This “desks” tags and its attachments were already set in the section 4.3 during the beacon setup. The Proximity Zone Object was created with the code as shown in the figure 15.

```
let zone = ProximityZone(tag: "desks", range: .near)
```

FIGURE 15. Code for creating Proximity Zone Object

In the above code, the range is set to “.near” which is predefined and its value is 1 meter. The user can also define their own range. Now as the app had already connected with the Estimote Cloud, the attachments associated with the tags “desks” were imported from the cloud and used in the onEnter event as shown in the figure 16.

```
zone.onEnter = { context in
  let content = UNMutableNotificationContent()
  let deskOwner = context.attachments["desk-owner"]
  let dpost = context.attachments["post"]
  content.title = "You are in \(deskOwner ?? " " )'s Desk"
  content.body = "\(dpost ?? " " )"
  content.sound = UNNotificationSound.default()
  let request = UNNotificationRequest(identifier: "enter", content:
    content, trigger: nil)
  notificationCenter.add(request, withCompletionHandler: nil)
}
```

FIGURE 16. Code for onEntry event (Estimote Inc. cited 16.12.2018)

Similarly, when the user exists the desk “zone.onExit” event was used to display the goodbye notification when the user leaves the desk as shown in the figure 17.

```
zone.onExit = { context in
  let content = UNMutableNotificationContent()
  content.title = "Bye bye"
  content.body = "Hope to see you again."
  content.sound = UNNotificationSound.default()
  let request = UNNotificationRequest(identifier: "exit", content:
    content, trigger: nil)
  notificationCenter.add(request, withCompletionHandler: nil)
}
```

FIGURE 17. Code for onExit Event (Estimote Inc. cited 16.12.2018)

4.5.8 Starting the proximity Observation

After doing all the above steps, proximity observation was started with the following line of code shown in the figure 18.

```
proximityObserver.startObserving([zone])
```

FIGURE 18. Code for starting Proximity Observation (Estimote Inc. cited 16.12.2018)

Now, the project was ready for build and run on an iOS phone.

4.6 Deploy on iPhone

The Application developed on Swift programming language platform cannot directly run on iOS devices. The user should have an app developer ID or should be a member of some development team. More detail about the app developer ID is mentioned in the section 4.2.5. OUAS developer account was used in this project which is set up as mentioned in the section 4.2.5.

The USB lightning cable was used to connect iPhone6s and Mac. The desk-info app was built and deployed into the iPhone6s. When the app was opened, the home screen was displayed as shown in the figure 19.



FIGURE 19. Home screen of the desk-info app

4.7 Observation

Beacons were placed around 5 meters apart. The app was opened and the device was locked. When the device was taken near around 1 meter, it shows the notification as shown in the figure 20. And when the device was moved far from the beacon, it displayed Goodbye notification as shown in the figure 21.

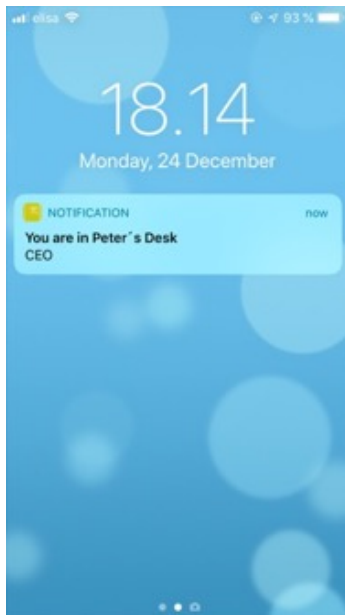


FIGURE 20. Notification when user enter Peter's desk

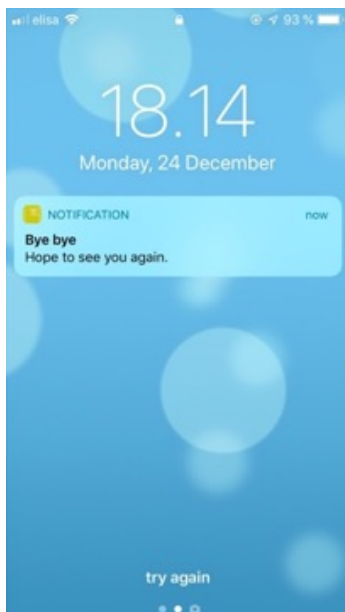


FIGURE 21. Goodbye Notification when a user leaves the Peter's desk

When the device was moved next to the second, it displayed the desk info about the particular desk and the goodbye message was shown as shown in the figure 22 and in the figure 23.

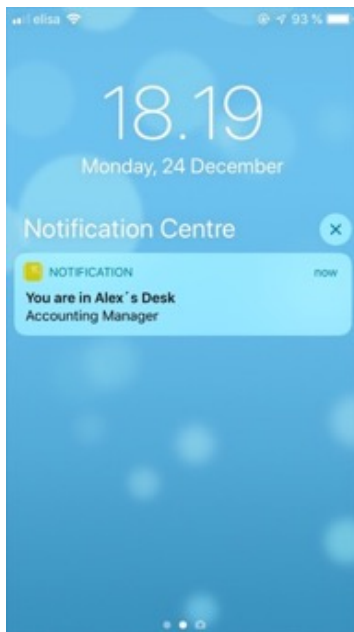


FIGURE 22. Notification when a user enters Alex's desk.

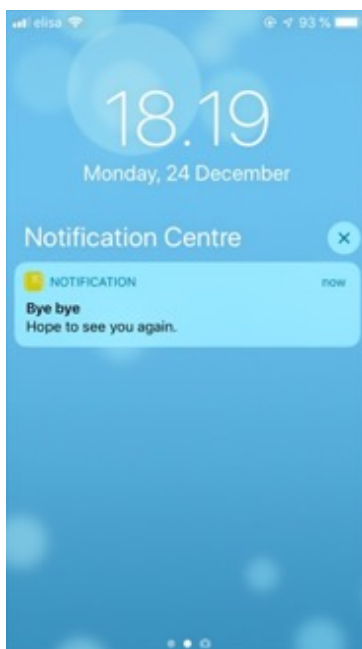


FIGURE 23. Goodbye Notification when a user leaves Alex's desk.

5 CONCLUSION

The BLE beacon technology is new and still in the evolving phase since it was first introduced. Even during the development process of this thesis Estimote introduced new LTE Beacons designed primarily to seamlessly locate assets and vehicles when they move between an indoor and outdoor environment (Estimote Inc. cited 23.12.2018).

The original idea of the thesis topic was to use robotics for location tracking and positioning using the BLE technology. As all the technology was new, it was difficult to implement so the original topic was dropped out and only the use of BLE beacon was finalized. Now the main objective was to study the use of the Bluetooth LE technology for location tracking and positioning and to develop a simple mobile application to demonstrate its use in the real world. A simple desk-info app was built to give the information about the desk owner as a notification when the user enters into the zone of the desk-owner.

This app can be further improved by triggering some action when the user gets into the app. At this moment there is only one static home screen which does not perform any task. This app can be modified and also used in the supermarket for promotional of some products.

To conclude, BLE beacons are very popular these days for indoor location tracking and positioning. One of the use cases was successfully demonstrated through a desk-info app. It is easy to predict that in the future many large offices and buildings are going to implement the BLE beacon for indoor navigation. BLE beacons are not only limited to indoor navigation, they can be used for multiple purposes such as promotional products, locating product or objects and many more.

REFERENCES

Adarsh Monika. 2017. 11 Incredible beacon use cases from 2017 every marketers should know in 2018, <https://blog.beaconstac.com/2017/12/11-incredible-beacon-use-cases-from-2017-every-marketer-should-know-before-2018/>

Alatalo Ilari. 2012. Technology Behind Real-Time Location System, http://www.theseus.fi/bitstream/handle/10024/45658/Alatalo_Ilari.pdf?sequence=1&isAllowed=y

Anderson Jess, 28.02.2017. What is Estimote Cloud? Cited 20.12.2018, <https://forums.estimote.com/t/what-is-estimote-cloud/7076>

Apple Inc. Started with iBeacon. 02.05.2014. Version 1.0, <https://developer.apple.com/ibeacon/Getting-Started-with-iBeacon.pdf>

Bekkelien Anja. 2012. Bluetooth Indoor Positioning. University of Geneva, Master's thesis

Dru, M. & Saada, S. 2001. Location-based mobile services: The Essentials. Alcatel Telecommunications Review, 71-76.

Estimote Inc. Announcing LTE Beacon for asset tracking. Cited 23.12.2018, https://blog.estimote.com/post/177348177680/estimote-lte-m-beacon?gclid=Cj0KCQiAgf3gBRDtARIsABgdL3lOrq29sHYynKx5cLHSf4_ItD-DxOFhyYILaKVU5nwzzZHL0BcW0ElaAtntEALw_wcB

Estimote Inc. Proximity Observer, Cited 21.12.2018, https://estimote.github.io/Android-Proximity-SDK/com.estimote.proximity_sdk.api/-proximity-observer/index.html

Estimote Inc. What is iBeacon? Cited 16.12.2018, <https://en.wikipedia.org/wiki/IBeacon>

Google Inc. Github. Eddystone-EDI, Cited 15.12.2018,
<https://github.com/google/eddystone/tree/master/eddystone-eid>

Google Inc. Github. Eddystone Protocol Specification. Cited 15.12.2018,
<https://github.com/google/eddystone/blob/master/protocol-specification.md>

Google Inc. Github. Eddystone-TML. Cited 15.12.2018,
<https://github.com/google/eddystone/tree/master/eddystone-tlm>

Google Inc. Github. Eddystone-URL. Cited 15.12.2018,
<https://github.com/google/eddystone/tree/master/eddystone-url>

Google Inc. Github. Eddystone-UID. Cited 15.12.2018,
<https://github.com/google/eddystone/tree/master/eddystone-uid>

Herrera Vargas Milan, 2016, Indoor Navigation Using Bluetooth Low Energy (BLE) Beacons, Bachelor's Thesis

IndoorAtlas. 2016. The Rise of Indoor Positioning. Cited 03.06.2018,
<http://www.indooratlas.com/wp-content/uploads/2016/09/A-2016-Global-Research-Report-On-The-Indoor-Positioning-Market.pdf>

Jiang, Landu. 2012. A WLAN Fingerprinting Based Indoor Localization technique in the University of Nebraska, Master's thesis.

Mautz Rainer. 2012. Indoor Positioning Technologies. Habilitation Thesis submitted to ETH Zurich. Cited 03.06.2018, <https://www.research-collection.ethz.ch/bitstream/handle/20.500.11850/54888/eth-5659-01.pdf>

Miller Paul. The new Raspberry Pi has 5 GHz and Bluetooth 4.2. Cited 22.05.2018, <https://www.theverge.com/circuit-breaker/2018/3/14/17117446/raspberry-pi-3-model-b-plus-dual-band-wi-fi>

Poole Ian. Bluetooth radio interface, modulation & channels. Cited 30.08.2018, <https://www.radio-electronics.com/info/wireless/bluetooth/radio-interface-modulation.php>

RFID-lab, What is RFID? .Cited 14.5.2018, <http://www.rfidlab.fi/rfid-teknologia/mita-on-rfid/>

Stroud Forrest. Beacon. Cited 18.05.2018, <https://www.webopedia.com/TERM/B/beacon.html> 18.05.2018

Wikipedia. CocoaPods. Cited 20.12.2018, <https://en.wikipedia.org/wiki/CocoaPods>

Wikipedia. iBeacon. Cited 16.12.2018, <https://en.wikipedia.org/wiki/iBeacon>

Wikipedia. Indoor Positioning System. Cited 15.5.2018, https://en.wikipedia.org/wiki/Indoor_positioning_system

Wikipedia. Xcode. Cited 15.12.2018, <https://en.wikipedia.org/wiki/Xcode>