FAKULTÄT
FÜR !NFORMATIK
Faculty of Informatics

Indoor navigation with Bluetooth Low Energy in crowded places

BACHELORARBEIT

zur Erlangung des akademischen Grades

Bachelor of Science

im Rahmen des Studiums

Wirtschaftsinformatik

eingereicht von

Lukas Frank, Natasa Nikic
Matrikelnummer 1325904, 1325974

an der Fakultät für Informatik

der Technischen Universität Wien

Betreuung: Assoc. Prof. Dr. Dipl.-Ing. Hilda Tellioglu
Mitwirkung: Dipl.-Ing. Gerfried Mikusch, Projektass.
Dipl.-Ing. Michael Urbanek, Projektass.

Wien, 18. März 2018

Lukas Frank, Natasa Nikic — Hilda Tellioglu

Technische Universität Wien
A-1040 Wien ▪ Karlsplatz 13 ▪ Tel. +43-1-58801-0 ▪ www.tuwien.ac.at

FAKULTÄT
FÜR !NFORMATIK
Faculty of Informatics

Indoor Navigation with Bluetooth Low Energy in Crowded Places

BACHELOR'S THESIS

submitted in partial fulfillment of the requirements for the degree of

Bachelor of Science

in

Business Informatics

by

Lukas Frank, Natasa Nikic
Registration Number 1325904, 1325974

to the Faculty of Informatics

at the TU Wien

Advisor: Assoc. Prof. Dr. Dipl.-Ing. Hilda Tellioglu
Assistance: Dipl.-Ing. Gerfried Mikusch, Projektass.
Dipl.-Ing. Michael Urbanek, Projektass.

Vienna, 18th March, 2018

Lukas Frank, Natasa Nikic	Hilda Tellioglu

Erklärung zur Verfassung der Arbeit

Lukas Frank, Natasa Nikic
Wien

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

Wien, 18. März 2018

Lukas Frank, Natasa Nikic

Danksagung

Zunächst würden wir uns gerne bei Assoc. Prof. Dr. Dipl.-Ing. Hilda Tellioglu für die Betreuung dieser Arbeit bedanken.

Anschließend möchten wir uns bei Dipl.-Ing. Gerfried Mikusch und Dipl.-Ing. Michael Urbanek für die vielen hilfreichen Tipps während der Entwicklung des Prototyps bedanken. Außerdem haben sie uns das Testen unterschiedlicher Beacons ermöglicht und anschließend die von uns gewünschten Estimote Beacons zur Verfügung gestellt.

Zu guter Letzt möchten wir uns bei unseren Familien und Freunden für die Unterstützung im Verlauf der Arbeit bedanken.

Acknowledgements

First of all, we want to thank Assoc. Prof. Dr. Dipl.-Ing. Hilda Tellioglu for her support throughout the thesis.

Furthermore, we would like to thank Dipl.-Ing. Gerfried Mikusch and Dipl.-Ing. Michael Urbanek for their advice during the development of our prototype. Moreover, both enabled the testing of various beacons. After our decision to use Estimote beacons, they organized the number of beacons necessary in order to fulfill the requirements for our project.

Last but not least, we are thankful for the support of our families and friends during the writing of this thesis.

Kurzfassung

Heutzutage wird hauptsächlich Global Positioning System (GPS) für die Navigation verwendet, allerdings ist GPS innerhalb von Gebäuden nicht verfügbar, da die Signale fast zur Gänze von den Mauern blockiert werden. Daher wurden in den letzten Jahren viele verschiedene Techniken erarbeitet, welche eine Navigation innerhalb von Gebäuden ermöglichen können. Einige von diesen Techniken arbeiten mit Radiowellen. Diese sind allerdings sehr anfällig für Signalveränderungen oder Störungen.[QLT16]
Mit der Einführung von Bluetooth low energy (BLE) in der Bluetooth 4.0 Spezifikation, entstand eine neue Möglichkeit zur Installation eines Navigationssystems. Mithilfe von leicht zu installierenden BLE-Beacons, die weder viel kosten noch viel Strom verbrauchen, kann eine Navigationsinfrastruktur geschaffen werden.

Allerdings muss man bedenken, dass, so wie andere Wireless Technologien, Bluetooth leicht durch andere Signale, Personen oder Hindernisse, welche den direkten Weg zum Sender (Line-of-sight) blockieren, gestört werden.
Das Ziel dieser Arbeit ist diese Störungsarten und deren Auswirkungen auf ein Indoornavigationssystem zu untersuchen. Es wird ein Prototyp eines Indoornavigationssystems entwickelt, der ausschließlich mithilfe von Bluetooth Signalen funktioniert und anschließend in einem Umfeld, das dem eines belebten Bahnhofs ähneln soll, getestet. Als Störquellen werden andere Personen sowie Geräte, welche Wifi und/oder Bluetooth aktiviert haben, untersucht.

Unsere Evaluierung hat gezeigt, dass die Abweichung zwischen tatsächlicher und errechneter Nutzer Position zwar unter 2 Meter liegt, allerdings die Zeitspanne, bis diese Position angezeigt wird, bis zu 6 Sekunden betragen kann, was in einem realistischen Szenario nicht zufriedenstellend wäre.
Basierend auf einer Literaturrecherche, wird diese Arbeit verschiedene Möglichkeiten aufzeigen, wie der Prototyp erweitert werden könnte, um den Einfluss diverser Störungsarten zu verringern und die Leistung zu verbessern.

Abstract

Nowadays, GPS is widely used in outdoor environments for navigation, but it becomes unavailable in indoor environments, due to the signals not being able to go through walls. That is why many alternative indoor localization techniques have been studied and proposed in the last years. Some of them work with radio-frequency (RF) signals. Such approaches are quite sensitive to signal dynamics.[QLT16]
With the release of Bluetooth low energy (BLE) in the Bluetooth 4.0 specifications, a new way to implement such an indoor navigation system came up. With the help of easy to deploy BLE-based beacons as localization infrastructure with low power consumption and relatively low cost, it is easy to build the needed infrastructure.

However, like other wireless systems, Bluetooth suffers from signal interference causing fluctuations, which come from environmental dynamics. Furthermore, the amount of people in the room or obstacles blocking the Line-of-sight (LoS) to the beacon have an influence on the signal strength.
Therefore, the aim of this thesis is to analyze those sources of interference and their impact on an indoor navigation system.
Furthermore, the prototype will be evaluated in an environment, simulating a crowded train station, where people and other devices are blocking the Bluetooth signals.

The evaluation showed, that the position estimations were within 2 meters, which is a relatively good result compared to other solutions. During movement, the delay between the user's actual position and the displayed position was around 6 seconds which is not satisfactory in a real life scenario.
Based on a literature research this thesis presents possible improvements to the prototype to reduce the impact of interference and improve the overall performance.

Contents

CHAPTER 1

Introduction

This chapter shows how this work came into being and why this topic has been chosen. Also, the goals that this thesis wants to achieve are presented.

1.1 Motivation

In recent years, navigation solutions became more and more important, since road networks became more complex. The breakthrough of smartphones made it easy to create solutions, which can be used almost anywhere. Most of those solutions are only usable if a certain strength of GPS signals is available. This boundary makes those solutions not applicable for indoor navigation, since most of these signals are blocked by walls and housetops.

Indoor navigation could be used in many scenarios, like in malls or railway stations, where it is hard to find your way through.
This work wants to present a solution to establish such a navigation system with the help of Bluetooth Low Energy beacons. The goal is being able to navigate the user to a meeting point, with friends for example, or a point of interest.

However, Bluetooth suffers from signal fluctuations environmental dynamics, which influence the accuracy and also the latency of an indoor navigation system.
Therefore, this thesis wants to analyzes the interference on Bluetooth Low Energy signals and evaluate whether this technology is applicable in a real life scenario. Also, this thesis provides possible enhancements to improve the overall performance of such systems.

1.2 Aim of the work

This thesis aims at providing information about how beacons work in general, how they can be used to establish indoor navigation and the challenges, which have to be tackled to build such a system. This is done by a comprehensive literature research on the topic. Equipped with the knowledge gained by our research, we implement such a navigation system based on User-Centered Design. In this work, we restrict to only one source of information, the Bluetooth Low Energy signals, to be able to evaluate this technology. Nevertheless, this work also presents other possible approaches to implement such a system.

Furthermore, an evaluation is made on how precise and how useful this solution is when being used in everyday life. This evaluation includes several test cases, which should represent real life scenarios in the public transportation system, where many people are around the receiver of the Bluetooth signals and therefore interfering or even blocking the signals. Also, we want to analyze the impact of other Bluetooth or Wifi enabled devices on our prototype.
Parameters, which will be evaluated, are accuracy (the difference between the real position and the calculated position) and latency or delay (the time the app needs to show the correct position).
Also, related work, which use different technologies to overcome the problems related to indoor navigation with Bluetooth Low Energy beacons and signal interference, will be analyzed.

1.3 Structure of the work

In the first chapter 'Introduction' the motivation and goals of this thesis are presented.

The second chapter lays the theoretical foundation of this thesis. It presents information about the Bluetooth Low Energy Standard, introduces beacons and provides information on how these devices function. Additionally, different Beacon types are presented.

The third chapter deals with the state of the art of topics from this thesis. In the first section positioning techniques, with focus on indoor positioning, are presented. Furthermore, related work, which apply these techniques, is included. Since a prototype is implemented, this work also looks at interface and interaction design and its guidelines. The mobile application runs on the Android Platform, therefore the Android system is described and also the Material Design Guidelines for Android applications are presented.

The fourth chapter presents the methodologies used for this thesis. It starts with describing the literature research, goes on with the idea of User-Centered Design and also presents the evaluation process used in this thesis.

The fifth chapter describes the prototype implementation. This is done by describing the design and implementation phase based on User-Centered Design. It outlines how our prototype handles positioning and navigation, and describes the core of the prototype,

which is the Estimote-Library. This chapter also deals with the limitations of the solution presented in this thesis.
Furthermore, this chapter evaluates the prototype and examines the challenges which have to be tackled in a real life scenario. Some use cases are evaluated which should represent scenarios in everyday life when being in a public transportation system and wanting to navigate from the current position to a specific point.

The sixth chapter provides a detailed description on how to set up and how to use the prototype.

The seventh chapter shows the limitations of this thesis and gives insight on possible improvements for the application. This is done by research on related work and analysis of other solutions which work with different technologies like Pedestrian Dead Reckoning or combinations of Bluetooth and Wifi.

The eighth chapter presents the findings made during this work and provides an analysis of these findings.

The ninth chapter finally concludes this thesis with a summary of contents in this work.

CHAPTER 2

Used Technologies

In this chapter the theoretical foundation is built for this thesis. The first section presents the Bluetooth Low Energy standard and compares it with Bluetooth classic. The second section introduces beacons. That section presents how beacons work and what different kinds of beacons are available on the market.

2.1 Bluetooth BLE Standard

Early research in the direction of indoor navigation were not even considering the use of Bluetooth, since the older versions of it were very energy consuming and only point-to-point communication was supported. To be able to communicate, pairing between the sender and the receiver with each other was needed, which leads to a lot of overhead. This made earlier versions of Bluetooth not applicable for fast and cheap communication, which is needed in indoor navigation scenarios.[Blua]

This changed with the introduction of Bluetooth BLE in July 2010. Also known as Bluetooth 4.0 or Bluetooth Smart, it was first introduced by Nokia under the name of Wibree, but was adopted by the Bluetooth Special Interest Group. The goal was to make Bluetooth simpler to use and to be less energy consuming compared to standard Bluetooth is. [Blub] [KMK16]

2.1.1 Bluetooth Low Energy characteristics

Bluetooth Low Energy uses, just like standard Bluetooth does, the unlicensed "industrial Scientific and Medical" (ISM) band with 2.4GHz.[JKJC15] It uses 40 channels which split the ISM band equally, where each channel is 2MHz wide. 37 of them are data channels and the remaining 3 of these channels are used for advertising, namely: 37, 38 and 39. The allocation was made to minimize collision with the most used Wifi channels, which are 1, 6 and 11. The definition of those dedicated advertising channels was made to make device discovery much faster, since it is not needed to scan the whole frequency spectrum for devices.[DM14] The downside is, that this may lead to high signal collision rates in the advertising channels if there is a large number of BLE devices in a narrow area, since only 3 channels are available. [SyIhR16]

Furthermore, those advertisement channels enable the option to send broadcast messages. Even though the messages sent via those channels are advertisement messages, they are able to carry a payload that can be used to broadcast any information change, such as a sensor state. [FH15] This mechanism is called advertising or broadcasting mode and makes it possible to send data to devices without pairing or the establishment of a connection, which was needed when using the standard Bluetooth technology. The pairing process takes a lot of time and with that, only point-to-point communication is possible. Additionally, it was not possible to configure the rate of advertisement packages. With Bluetooth Low Energy, devices can advertise from once every 10 seconds up to once every 20 milliseconds. [DM14]

In addition to the presented advertising or broadcasting mode, there are 3 more modes in Bluetooth Low Energy: Central, Peripheral and Observer. The Observer is the opposite to the already mentioned Broadcaster and is on the receiving end of those messages. The central role is similar to the master role in the standard Bluetooth specification and is used for more sophisticated devices which initiate and manage connections to other devices. The peripheral role is typically used for simple devices, which handle one connection to a central device.

2.1.2 Bluetooth LE vs standard Bluetooth

As already described earlier, BLE is a modification to the standard Bluetooth protocol, which allows low bandwidth, low latency and an energy efficient communication. These characteristics are needed by devices, which send small amounts of data periodically via broadcast to the surrounding area. There is a large amount of devices which count to this class, for instance fitness monitors, proximity beacons for indoor navigation, or other IoT type devices which could be used in a smart home environment.
The differences between BLE and standard Bluetooth are presented in table 2.1

Major changes had to be made to reduce energy consumption of devices using Bluetooth LE for communication. First, not only was the overall number of channels reduced, but also the number of advertising channels. In Bluetooth LE scanning devices only need to search 3 channels for other devices instead of 32 in standard Bluetooth. Using only 3 channels for advertisement allows devices to scan for advertisements extremely fast. The scan is finished in 6 ms when using BLE, compared to around 100ms for standard Bluetooth.[RMBL15] [SyIhR16]

Category	Bluetooth LE	Bluetooth classic
Optimized For	Short burst of data	Data streams
Frequency	2400 to 2483.5 MHz	2400 to 2483.5 MHz
Data Channels	37	79
Advertising Channels	3	32
Encryption	AES 128 bit	64/128 bit
Throughput	$<$ 300Kbit/s	$<$ 3Mbit/s
Connection Setup Time	6 ms	$>$ 100 ms
Energy Consumption	0.01x to 0.5x	1 (reference)
Supported Topologies	Point-to-Point, Broadcast, Mesh	Point-to-Point

Table 2.1: Differences between standard Bluetooth and Bluetooth LE, [Blua] [RMBL15]

The second major difference is, that data can be sent via broadcast and not only via point-to-point connection, like in standard Bluetooth. If devices want to transfer data over standard Bluetooth, they need to establish a connection, they need to "pair" with each other. On the one hand, this enables streaming of vast amounts of data, but on the other hand it is energy consuming and takes around 6 seconds to set up. BLE does not use streams, instead it sends short burst of data via broadcast, which makes the setup time significantly lower because the pairing process is not needed. Data can be sent after around 3 milliseconds, which makes BLE applicable in scenarios where standard Bluetooth would not be usable because of the long setup time (for instance indoor navigation). The tradeoff is a significantly lower throughput rate of around 300kb/s, which is around seven times smaller than classic Bluetooth. This makes audio transmission unavailable when using Bluetooth LE without a connection between the devices. [JKJC15] [GOP12]

All these changes lead to the energy consumption being at least halved, but can go as low as 0.01 times the energy consumption of standard Bluetooth, depending on the use case of the device. [Blua]
The low energy consumption makes it possible for devices to be powered by a coin battery only. Some experiments showed, that those devices can last over a few years depending on the configuration.[GOP12]
Devices running on coin batteries are cheap, small and easy to deploy nearly anywhere since they are not bound to a power outlet like Wifi access points are. Factors, that influence the battery lifetime of an advertising device, are the configured strength with which the device should send its advertisement and the advertising interval.

It is important to keep in mind, that the presented energy consumption apply for advertising devices and not for scanning devices. In fact, smartphones in scanning mode have been the topic of many experiments. Most of them came to the conclusion, that BLE-scanning on smartphones has a very high energy overhead. In the paper [RMBL15], a detailed performance analysis has been made. Especially when the smartphone's display is turned off, the energy overhead seems to be significantly higher. The reason could be

the so called Wake Lock, which lets apps force the phone's processor to keep working even though it should sleep. Besides the scanning being in the foreground or background, there are other factors, which have impact on the power consumption, namely, how many beacons are in range of the scanning device and whether the phone performs only a scan, or if it also connects to the advertiser. Of course, the configuration of the scanning interval and the duty cycle have impact on the energy consumption too. The duty cycle is the duration of the scanning process during a scan interval. In the remaining time of the scan interval the device sleeps. [CPH+14]
This interval can be configured between 20ms to 10.24 seconds, in steps of 0.625ms, but there is a pseudo-random delay added from 0ms to 10ms which helps to reduce the possibility of collisions between advertisements of different devices.[arg]
Shorter scan intervals mean, that the BLE interface is turned on and off more frequently. Consequently, there has to be a tradeoff between power consumption and latency because the longer the scan interval, the slower the updates come in, but the faster the updates come in, the higher the power consumption. How you should approach this issue depends on the scenario. [arg]

Most works came to the conclusion, that smartphones are not benefiting from the BLE stack as much as advertising devices do. In fact, the power drain from using BLE is about the same as the drain from using standard Bluetooth. [RMBL15]

2.1.3 Problems with BLE in combination with Wifi

In this section, the problem with Wifi interfering the BLE signals is described.

To better understand the issues regarding the parallel use of BLE and Wifi, figure 2.1 shows the ISM frequency band, which both technologies use and therefore share. When using the same frequency bands, interference and channel blocking may occur. Recently the amount of technologies using the ISM band has been increasing significantly, leading to a more congested and crowded environment.

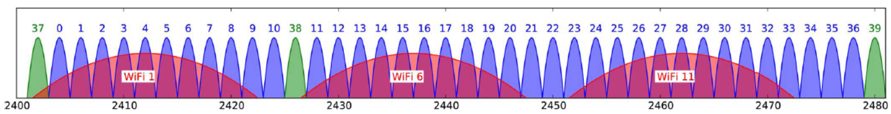


Figure 2.1: ISM Band with Wifi and BLE channels. [FH15]

Some papers like [SHNN12] analyzed the interference of Wifi with BLE. The work suggests, that the interference is higher, if the interfering device is close to the BLE receiving device. The experiment shows, that only around 60% of the packets were received by the BLE devices when the interfering Wifi device was very close. Their results also show, that only 1.5 meters seem to be enough to avoid almost all negative impact of the interfering device.

It has to be said, that the mentioned work did not make use of frequency hopping since Advanced Frequency Hopping was not implemented at that time.

BLE uses many techniques to overcome interference from Wifi.

The first and most important technique is the already mentioned Advanced Frequency Hopping. It is used to avoid congestion in both its advertising and data channels. This mechanism detects channels under interference or much use and excludes them from the hopping sequence. A hopping sequence is the sequence, which devices use when they have built a connection to communicate with each other. Those two devices do not just use one channel to communicate, they hop between channels together many times a second. The sequence of the hops is unique for each connection. The reason for frequency hopping is that the devices only spend small amounts of time on a certain channel, which minimizes the possibility of interference. [hon]

Secondly, the wide spacing of the channels, and especially the advertising channels, helps to better manage interference from Wifi, but also from classic Bluetooth or from other sources of interference, like microwaves or baby monitors. [arg]
The third technique used by BLE is Gaussian Frequency Shift Keying, but it is also used by classic Bluetooth. It is a modulation scheme, which uses a Gaussien filter to reduce sideband power. Sideband power could lead to Crosstalk, which is interference of an adjacent channel. [Max]

2.1.4 Problems with RSSI

In this subsection, interference with radio frequencies is described. This interference types may affect any wireless technology, not only Bluetooth Low Energy. The focus lies on those types of interference, which have a negative impact on RSSI values and therefore worsen the performance of our indoor navigation system.

First, we have to introduce the Received Signal Strength Indicator (RSSI). It is the strength of a signal from a Beacon, which is received by a device (a smartphone in our case). This indicator depends on the distance to the sending device and on the configured Broadcasting Power value. RSSI can be used to approximate the distance between the device and the beacon in combination with the Measured Power. The Measured Power is a factory-calibrated, read-only constant, which indicates, what the expected RSSI would be at a distance of 1 meter to the beacon. This constant is different for every smartphone model. It depends on many factors like the built-in Bluetooth chipset, but also the material the smartphone is built of has an impact. [estg]
The signals transmitted over a certain distance suffer from path loss, which means that they fade. Depending on the distance, those fades can be weaker or stronger. The following formula is the log-distance path loss model and it tries to describe the fading that a signal encounters when traveling through a building or densely populated area. [CJC+15]

$$RSSI = -20 * log(d) + A \tag{2.1}$$

where A is a reference received signal strength in dBm (measured RSSI value when 1 meter separates sender and receiver). [DD12] The d in this formula is the distance between the sender and the receiver. By solving this formula for d, a distance estimation can be calculated (equation 3.1). This approximate distance can be used to calculate a user's position, if at least 3 different RSSI values are present from different beacons and therefore 3 different distance estimations are present. The problem is that RSSI tends to fluctuate due to external factors that influence radio waves. Those external factors will be described in the following sections. [NJNR17]

Multipath fading

Multipath radio signal propagation occurs when signals travel not only the direct Line-of-sight path between sender and receiver, but on many paths. This happens because the signal does not leave the sender's antenna in just the direction of the receiver, but over many different angles. These signals may get reflected over a variety of surfaces and are still able to reach the receiver over ways other than just the direct Line-of-sight path. [Mul]
This maybe does not seem like a problem at first, but the overall signal received is the sum of all signals, which reach the receiver's antenna. When the received signal is in phase with the main signal, the signal strength will increase, but if the received signal is out of phase with the main signal, it will interfere with the main signal and therefore the signal strength will decrease. This effect is called Multipath Fading.[Mul]

This effect can be even stronger if something in this environment is moving, like a person or the receiving device or even the sending device. The movement causes changing phases in the signals, which arrive at the receiver and this will then result in the received signal strength varying.[Mul] Those multipath fades can be very strong even though there was just little movement. One study showed, that multipath fades with 30dB drops can occur on just 10cm of movement. [FH14]

Signal Absorption

As opposed to fading, where the signal strength is potentially lowered, signal absorption makes the signal very weak or lets it disappear completely. Low frequency radio waves can easily penetrate walls and stone. Very low frequencies can even travel through sea-water, but at frequencies of microwaves or higher, absorption becomes a major factor. [Wik]
For radio waves of the ISM band, water and metal have the highest potential of interference. [esth] Water does not seem like a problem when it comes to indoor navigation systems, but since the human body consist of around 60% of water, crowded places make it hard for Bluetooth signals to reach the receiving smartphone. Even the user is potentially interfering the signals, depending on the way the phone is held and in which direction the sending device is located. [FH14] [PFDZ16]

Figure 2.2 and table 5.9 show the signal behaviour when a human body is in Line-of-sight to the sender.

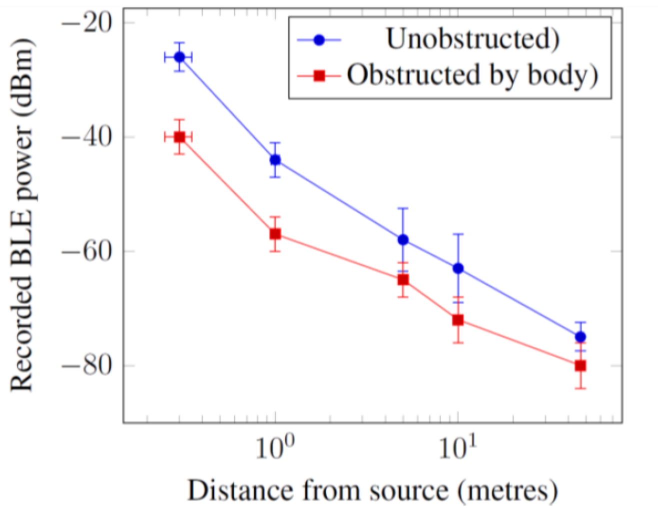


Figure 2.2: Human Body Interference, [FH14]

Other conditions

The environment is not the only factor, which has an influence on the RSSI. It also strongly depends on the battery level of sender and receiver device, the device model, or more precisely the position of the Bluetooth chip on the device and also the material the device itself is built of. [NJNR17] There is no fixed production standard regarding the designs of antennas and radio frequency front-ends, therefore many different designs exist, which also has an impact on the RSSI values.[MPB$^+$17] The RSSI differences between two different smartphones can be seen in table 5.7 and table 5.8, where a Samsung Galaxy S5 and a Nexus 5X are compared.

2.2 Beacons

In this section, beacons are introduced and the different types are presented.

Beacons are small devices which do nothing else than sending their unique identifier in a certain time interval. Those senders can be deployed nearly anywhere and can be used for many different scenarios to enhance customer experience by position based information. [Cis]
Mobile devices, like smartphones for example, can use these signals to recognize relative proximity to a certain point of interest. Another possibility would be to estimate an absolute position of the device itself and therefore of the user to be able to display it on a map.

2.2.1 BLE Beacons

BLE beacons use the BLE standard, but they only use broadcasting packets without allowing the connection of other devices. The reason for this is, that if a device would establish a connection to the beacon, advertisements would have to stop and therefore no other device would be able to find the beacon or receive the signals of it. Also, establishing a connection is more energy consuming for the Beacon itself. [arg]
BLE beacons are very attractive to retailers because they promise a long battery life (of a few years) and they have low maintenance requirements. Once deployed to a wall or an object of choice, they do not need any maintenance until the coin battery needs to be replaced. [estj]
It is important to keep in mind, that a long battery life requires low power output and/or low advertisement rates and that may affect the beacon's usefulness, depending on the scenario they are used in. [FH14]
Manufacturers, which produce BLE beacons, are Gimbal, Estimote, Kontakt and some others. [ais] A Beacon consists of a power supply (a coin battery most of the time), a Bluetooth module and a processor. Most currently available Bluetooth modules are manufactured by Texas Instruments, Bluegiga, Qualcomm and Nordic Semiconductor. [KMK16]

The format of the advertisement packets depends on the BLE communication standard which is used,. The most common ones are iBeacon and Eddystone. These will be presented in the following sections. [esti]

iBeacon

In 2013, Apple introduced their iBeacon protocol at their annual Worldwide Developers Conference. This action has drawn attention to beacons and companies started to think about applications they could develop with those small devices. [JSSN18]
This newly introduced packet structure provides the industry with a standard for application development. The structure can be seen in figure 2.3.
The payload is the interesting part of those packets. The Universally Unique Identifier (UUID) is a 128-bit value, that is typically used company-wide. The major number can be used to combine beacons to groups. For example beacons in the same building or the same floor could be configured with the same major number to put them together into groups. The minor number is used to identify the specific beacon. [KMK16][EF15]

An app, which listens for updates, could take actions depending on the beacon it receives updates from and therefore adapt its actions to the location the device is in. An example would be the information about coupons when entering a Starbucks or connect to the Wifi. [ars] This format is native only for iOS, but it is also compatible with Android.[bea]

The major drawback of iBeacon is, that the user has to have an application installed, which listens to the specific beacon, which sends the update. If we go back to the example with Starbucks, the user has to have the Starbucks app installed to be able to get information about vouchers. [ars]

(a)	Adv PDU			Payload defined by iBeacon Standard				
1 byte	4 bytes	2 bytes	6 bytes	9 bytes	16 bytes	2 bytes	2 bytes	1 byte
Preamble	Access Address	Header	MAC	iBeacon Prefix	Universally Unique Identifier (UUID)	Major	Minor	Tx Power

Figure 2.3: iBeacon Advertising Packet, [JSSN18]

Eddystone

In July 2015, Google entered the industry around beacons with their introduction of Eddystone, two years after the introduction of the iBeacon format. With another giant entering this market, the relatively new technology received even more attention.[bea] The name might sound strange, but it is said to come from the Eddystone Lighthouse in the United Kingdom. The story behind it is, that lighthouses guide ships at night and beacons are also guides in some sense. [bea]

There were several important improvements made to make Eddystone beacons be able to perform a variety of functions.

The first major improvement, in contrast to iBeacon is, that with Eddystone it is possible to send different payloads with your advertising packets. The first one, just as in the iBeacon standard, the UID, a beacon ID is sent. This ID is a unique, static ID with a 10-byte namespace component and a 6-byte instance component. [gooa]

The second payload tries to tackle the drawback of iBeacon, namely, that a user has to have a specific application installed, which listens to beacon updates. With Eddystone-URL, it is possible to send a compressed URL to the client, which, once decompressed and parsed, can be directly accessed by the client via a web browser. Additionally, this could be used to push location-specific app installs, like a guiding app for museums. If you enter the museum you get prompted to install the museum's app, with the link to the download page. The Eddystone-URL is the QR-Code version in the beacon world, with the advantage, that you do not have to find and scan the QR code because the URL is directly sent to you. [ars][gooa]

The third packet payload is Eddystone-TLM, which provides telemetry information about the beacons. Companies, which have to manage a fleet of beacons, can use this payload to receive diagnostic data and health statistics from the beacon itself. For example the remaining power of the coin battery can be examined with this payload. [ars][HQ16] The structure of the different payloads can be seen in figure 2.4.

Not only the packet formats have been changed, also improvements in the security and privacy subject were made. When using the iBeacon format, there is no feature, which allows to filter the users, which can read the beacon advertisements. The signal transmitted by a beacon is a public signal and can be received and read by any iOS devices and certain Android devices with proper specifications.[bea]

Eddystone has a solution for this issue, called Eddystone-EID. Beacons, which use EID, change their identifier every few minutes and therefore their advertisement packets change

<table>
<tr><td colspan="4">(b) Adv PDU</td><td colspan="7">Payload defined by Eddystone Standard</td></tr>
<tr><td>1 byte</td><td>4 bytes</td><td>2 bytes</td><td>6 bytes</td><td rowspan="2">UID</td><td>1 byte</td><td>1 byte</td><td colspan="2">16 bytes</td><td colspan="2">2 bytes</td></tr>
<tr><td rowspan="5">Preamble</td><td rowspan="5">Access Address</td><td rowspan="5">Header</td><td rowspan="5">MAC</td><td>Frame Type</td><td>Ranging</td><td colspan="2">UID</td><td colspan="2">Reserve</td></tr>
<tr><td rowspan="2">URL</td><td>1 byte</td><td>1 byte</td><td colspan="4">18 bytes</td></tr>
<tr><td>Frame Type</td><td>Ranging</td><td colspan="4">URL</td></tr>
<tr><td rowspan="2">TLM</td><td>1 byte</td><td>1 byte</td><td>2 bytes</td><td>2 bytes</td><td>4 bytes</td><td>4 bytes</td></tr>
<tr><td>Frame Type</td><td>TLM Version</td><td>Battery Level</td><td>Temperature</td><td>ADV_CNT</td><td>SEC_CNT</td></tr>
</table>

Figure 2.4: Eddystone Advertising Packet, [JSSN18]

every few minutes. To be able to resolve useful information out of these packets, you need to have a service, that shares a key with the beacon, the Ephemeral Identity Key (EIK). [goob]

On top of those improvements, Eddystone is not only Android and iOS compatible, it is also cross-platform and can be used by any platform that supports BLE beacons. Furthermore, it is open-source under the Apache v2.0 license, which means that businesses and developers can contribute improvements to the standard. [bea]
The whole specification is published on Github.[git]

Estimote Beacons

In this work, we used Estimote Beacons ([esta]) for our prototype. We also had a look at kontakt.io ([kona]) beacons, but with them, the Estimote software development kit was not compatible. It was important for us to be able to use the Estimote SDK because it provides an absolute user position. This position is estimated with trilateration, in comparison to the kontakt.io SDK([konb]), or the Android BLE SDK ([alt]), which only provide the beacon updates, but no estimated user position.

The Estimote SDK, we used for our system, was the Estimote Indoor SDK for Android([estc]), which needs Estimote Location Beacons to be usable. These are able to send Eddystone and iBeacon updates. [esti]
The differences between Estimote Proximity Beacons and Estimote Location Beacons are the longer battery life (from 3 years to up to 7 years) and increased range (from 70 meters to a maximum of 200 meters), but more importantly their usability in more complex deployments, where more precise location information is needed (an absolute position instead of just proximity information). [estf]

The current version of Estimote Location Beacons (F3.3) have an ARM Cortex-M4 32-bit processor, a flash memory and a BLE radio module. They are powered by a 4 x CR2477 – 3.0V lithium primary cell battery, which is replaceable. The BLE radio module works with the Bluetooth 4.2 LE Standard, has a range of up to 200 meters, the output power can be configured between -20 to +4 dBm (4 dB steps) and it provides a "Whisper mode" with -40 dBm and a "Long range mode" with +10 dBm. The sensitivity of it is -96 dBm. They are also equipped with some additional sensors, like a motion sensor (3-axis), a temperature sensor, an ambient light sensor, a magnetometer (3-axis) and a pressure sensor to be able to provide more location based information.
These beacons weight around 67g, which makes them easy to deploy. They have an already prepared backside, with which you are able to stick them nearly anywhere. [estb]

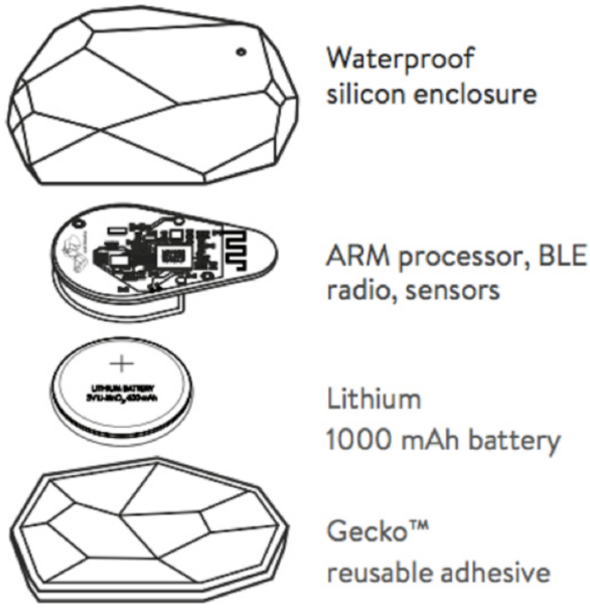


Figure 2.5: Estimote Beacon architecture, [met]

2.2.2 UWB Beacons

UWB, which stands for Ultra-wideband, is a promising technology when it comes to indoor navigation, but it has its downsides. UWB can be used in close proximity to other radio frequency signals without causing interference because the signal type and also the radio spectrum is different. The short signal pulses make it easy to filter out wrong signals, which are generated by multipath. Furthermore, the signals pass easily through walls and objects, but the strong interference of water or other liquids and metal is too much even for UWB. [LDBL07]

It has to be said, that there are some significant drawbacks when using this technology. UWB signals need a dedicated signal receiver (which are not integrated in most common smartphones), they can have a high energy consumption and they have relatively high manufacturing costs compared to other radio frequency systems. [MPB+17][NK11] Therefore, it is hard for the general public to use a system with this technology.

It is possible to combine the UWB technology with BLE signals to get an overall better result. [Kol17] present the concept of a hybrid Bluetooth Low Energy and Ultra-wideband positioning system. The approach is to calculate the location with the help of BLE RSSI measurements because BLE has a lower power consumption. The measured values from UWB are used to periodically improve the BLE RSSI based localization. They claim, that their approach allows a high location update rate because of BLE, while still having a high accuracy and low energy consumption.

Their tests show, that including UWB measurements into the localization calculation leads to a better outcome of more than 60%, in comparison to using only RSSI values measured with BLE. Their solution is a trade off between accuracy and energy consumption, but the energy consumption can be lowered by lowering the UWB measurement rate with still having significantly better accuracy. The 60% increased accuracy was reached with an UWB update rate of 5Hz. With an update rate of 0.5Hz, the increased accuracy was still 30% better than without using UWB measurements.

[JS17] evaluated a use case, where the system should help to find things of interest in a museum. Not only UWB was evaluated, but also BLE. It was found, that even under close to constant LOS, the measured range differed in 0.4 meters from the straight LOS path. The cause of this ranging error were probably some people in the same room during the testing phase, which shows that UWB is also effected negatively by interference caused by water, which is in line with the literature.[LDBL07]
They say, that UWB radio technology can obtain a ranging performance of 0.2 m when Line-of-sight is present and that the technology is capable of a maximum range of more than 100 meters. In apartment type scenarios, the maximum measurable range is 10-15 meters. They also claim, that UWB is the most accurate ranging system currently available for indoor scenarios with multiple reflecting surfaces or obstacles.
The same scenario was tested with BLE beacons. When using only 1 BLE beacon, the range estimations were not very accurate, showing an error of around 3 meters in 60% of the cases. With 6 beacons and with the enhancement of PDR (= Pedestrian Dead Reckoning), the results were almost as good as the results of the tests with UWB.

The results show, that the UWB technology has a better accuracy than BLE most of the time, but with some relatively easy enhancements, the BLE results can be as good as the ones from UWB, without having the downsides of extra costs and the need of dedicated receivers.

2.2.3 Wifi

Wifi was the technology to go when it came to indoor localization before the introduction of BLE in 2010. [KMK16] Even though there are no Wifi beacons, beaconing, which is the periodical sending of information, is still possible because Wifi access points can be used instead.[FH15][ZYL+16]
This might seem convenient at first, since most bigger facilities, like train stations or shopping malls, offer Wifi already, so no additional infrastructure would be needed.[ZXM+14]
However, those Wifi access points are deployed to provide good coverage over the facility for communication purposes, which means minimal range overlap. Typically, there is no consideration for Wifi access point geometry for user positioning. [FH15][JSSN18][ZXM+14]

Another reason why BLE became the center of interest in indoor positioning research is, that BLE beacons are easy to deploy. The beacons are way smaller than Wifi access points and can last a few years with coin batteries, whereas most Wifi access points need

a power outlet.[JS17][ZYL+16] In addition, BLE beacons are inexpensive in comparison to Wifi access points and therefore dense deployments are feasible.[ZXM+14]
Furthermore, one major limitation of Wifi is the low scan rate. When using a smartphone to scan for Wifi signals, the time span between the received signals can be as large as 4 seconds. On top of that, Wifi buffers readings and provides the information as a single batch, which further increases the time between updates. This means, that the update rate of Wifi could be below 0.25 Hz. In comparison, BLE allows for advertisement rates up to 10Hz.[JS17] Also, the Wifi advertisements increase the network traffic and therefore reduce the Wifi throughput of the network. [FH15]

Since Wifi is already a few years old, newer mechanisms, like channel hopping, are not available. As a result, the RSSI of BLE is "cleaner" and less prone to interference. Also, the lower transmission power of BLE contributes to the better performance because it reduces multipath effects in some scenarios. On the other hand, the lower transmission power results in stronger fading of RF signals, which reduces the range of usable signals, hence the beacons need to be deployed closer to each other. [FH14][ZXM+14][CAAG16]

All those points are reasons, why BLE took over the market for indoor positioning, but there are still some solutions which use Wifi, some in combination with BLE, like [KMK16], others are based on Wifi fingerprinting. Fingerprinting will be explained in section 3.1.2.

CHAPTER 3

State of the art

3.1 Positioning techniques

Roughly speaking, there are two approaches to implement BLE signals into a indoor positioning system: proximity based and location based algorithms.[este]
The main difference between these two approaches is, that proximity based algorithms use relative location information based on the proximity to a reference point (beacon), whereas location based algorithms use an absolute position within a certain environment. [JSSN18]
The third approach is pedestrian positioning. Systems which use this technique are not based on radio signals, but on sensor data, which is collected during movement.[NK11]

3.1.1 Proximity based algorithms

The proximity based approach works with beacons placed at fixed positions within a building. When a person with a smartphone enters the building, it recognizes the signals from the beacons and the strongest, and therefore hopefully the closest, beacon is considered when calculating the user's proximity. This means, that the user is believed to be close to the corresponding beacon.[NK11]
Other ways to use proximity information are event-triggering thresholds. The device constantly measures RSSI and if the RSSI passes a certain configured threshold a proximity report can be triggered. Such a proximity report indicates, that a target device is in the coverage area of a reference network node (beacon). Such a scenario is called cell identifier positioning, where a user's position is associated to the fixed position of a beacon or access point. [YZG15]

Figure 3.1 pictures the proximity zones of three different reference points. The vectors show, whether the position is in proximity to the respective beacon (1) or not (0). For example the vector [1,1,0] means that the position is in proximity to reference point 1 and 2.

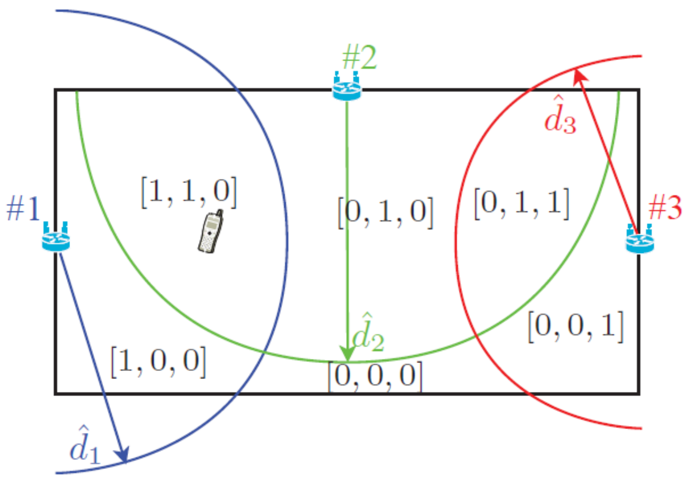


Figure 3.1: Segmentation for fingerprinting, [YZG15]

Thresholds can be configured for more than one range or area around a reference point. When using the Estimote proximity API ([estd]) for example, you can use 4 predefined zones or even configure more than those 4. The 4 predefined zones are immediate (very close to the beacon), near (about 1-3 m from the beacon), far (further away or the signal is fluctuating too much to make a better estimate) and unknown.[estg]

In some scenarios, it might be better to use a proximity based indoor navigation solution instead of finding an accurate position estimate, which comes with higher costs. The system can prompt the user (possibly in real time) when the zone is reached where the target of interest is in, or when the user is about to enter or leave it.[YZG15]

Another use case could be marketing. Imagine a visitor with a BLE enabled smartphone is in close proximity to your shop or restaurant and you want to attract their attention. It would be easy to send them a voucher or special offers when having a proximity based system installed in your facility. [KMK16]
Another scenario would be the information supply at a bus stop or train station. As soon as a passenger arrives at the station, they get notified with the transit times, or if the waiting time is longer, some suggestions for restaurants or promotions for shops could be provided. In the same way museums could promote exhibitions, when customers enter the building.[ars] [KMK16]

3.1.2 Location based algorithms

In this section, location based algorithms are presented. These algorithms provide an absolute position within a location, which means that those algorithms can provide a position estimation with x and y coordinates.

Trilateration

Trilateration is one of the earliest and most used approaches for indoor positioning. [RGR+17] When using trilateration, at least three reference points (beacons or base stations) must be in range of the scanning device. Around these three or more reference points, circles are calculated with the radii of the distances to the reference points. The estimated position is the intersection points between these circles.
To calculate the position, a system of at least three circle equations has to be solved with two unknown values, which are the coordinates. To be able to solve this problem, you have to have at least three equations, which is why you need at least three reference points.
When having three reference points in range, the next step is to measure the distance to these reference points. There are a few ways to calculate the distance to the reference points. These are presented in the following. The combination of the knowledge of the actual position of the reference points, together with the calculated distances, form the basis for trilateration. [DM14]

One way to calculate the distance between the sender and the receiver is Time-of-Arrival (TOA). The idea is to measure the one-way propagation time by using the relationship between the speed of light an the frequency of the signal. The duration is the difference between the sending of the signal to the reception of the signal. [DM14] An illustration can be seen in figure 3.2. However, this approach requires precisely synchronized clocks because an error of 1.0µs in time equals an error of 300 meters in distance.[CJC+15] The second issue is, that a timestamp has to be sent in the advertising packet. To accomplish this, the payload of the packet has to be altered accordingly.[LDBL07] TOA can not be used with cheap devices because a high accuracy clock has its price. Therefore, TOA is not applied very often.[CJC+15]

Time Difference of Arrival (TDOA) works quite similar to TOA. The difference is, that signals of two or more senders are used to calculate the distance. More specifically speaking, the time between the reception of signals from multiple transmitting devices is measured, rather than the absolute time of arrival as in TOA. [LDBL07] The upside of this approach is, that only the clocks of the transmitters have to be synchronized. The receiving device does not need to a synchronized clock. [DM14]

The two approaches above have some drawbacks. In indoor environments, there will not be many scenarios were Line-of-sight to the transmitters is present. Radio propagation in such environments is interfered from multipath effects, which has influence on the received signal strength, but also on the time of arrival. Therefore, the accuracy of the estimated position would decrease or even be too bad to be usable at all.[LDBL07]
In this paper, we are going to use an alternative approach. The idea is to estimate the distance of the receiving device by exploiting the attenuation of the transmitted signal

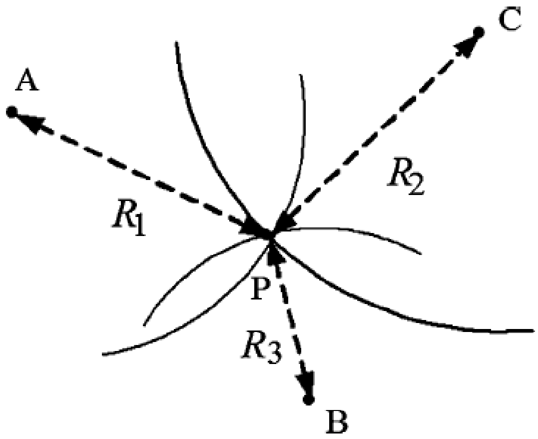


Figure 3.2: Trilateration with Time-of-Arrival, [LDBL07]

strength. Signal attenuation-based approaches use theoretical and empirical path loss models to translate the difference between the emitted signal strength and the received signal strength into a distance estimation. Figure 3.3 presents this approach. LS1, LS2 and LS3 indicate the path loss from the respective sender to the receiver P.
Based on the log distance path loss model, which is already explained in section 2.1.4, the formula for the distance can be derived from the RSSI formula. The distance can be calculated with:

$$d = 10^{\frac{A-RSSI}{20}} \tag{3.1}$$

where A is a reference received signal strength in dBm (measured RSSI value when 1 meter separates sender and receiver). [DD12]
Due to multipath and signal fading, those models do not always hold and the parameters used in those models are sometimes site specific, which means that they are dependent on the environment.[LDBL07]

The results of the paper [EF15] are promising if the location provides a clear Line-of-sight environment. If LOS is given, their achieved precision is around 1m, but if something blocks the signal transmission, the precision of their solution drops to around 5m. They find, that their solution is only applicable in larger facilities, like airports or train stations, where high precision is not mandatory. They also say, that their results are not feasible for indoor localization when high precision is needed, like in office buildings, where 5m inaccuracy could lead to navigation errors and that their solution needs improvement through further applications. Therefore, they think that hybrid approaches are needed for indoor positioning solutions.

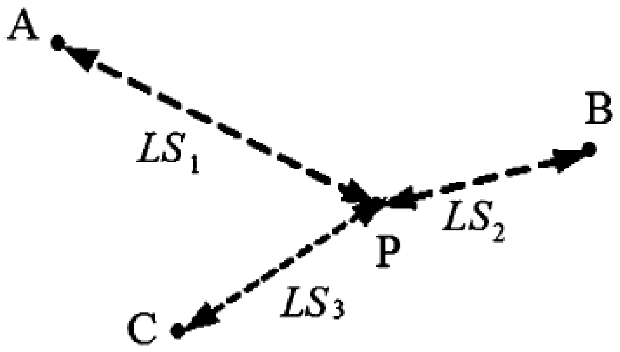


Figure 3.3: Trilateration with RSSI, [LDBL07]

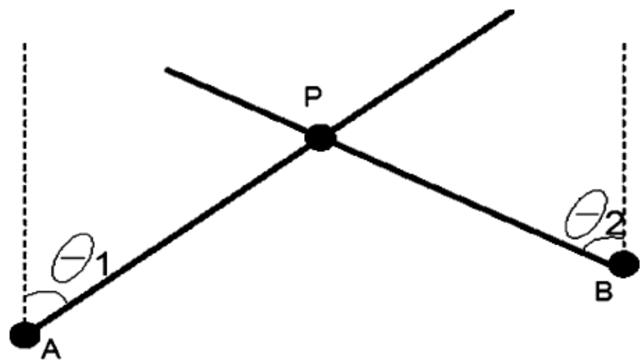


Figure 3.4: Triangulation with angle of arrival, [LDBL07]

Triangulation

Instead of estimating the position based on the distances to reference points, triangulation works with the angle of arrival (AOA). This means that positioning is done by computing the angles relative to multiple reference points.[RGR+17] The position can be found by the intersection of several pairs of angle lines, where each is formed by the circular radius around a beacon or a base station to the receiver. As presented in figure 3.4, at least two known reference points (A and B), and therefore two angles are needed to derive a location of the receiver P.[LDBL07]
To obtain the information about the arriving angle, either goniometers, gyroscopes or a compass can be used. However, this mechanism requires the use of isotropic antennas, which can be quite expensive. [DD12] Another issue is, that Line-of-sight between the receiver and each of the respective beacons, against which angle of arrival is to be calculated, is necessary, which makes this approach less applicable in realistic indoor scenarios. [DM14]

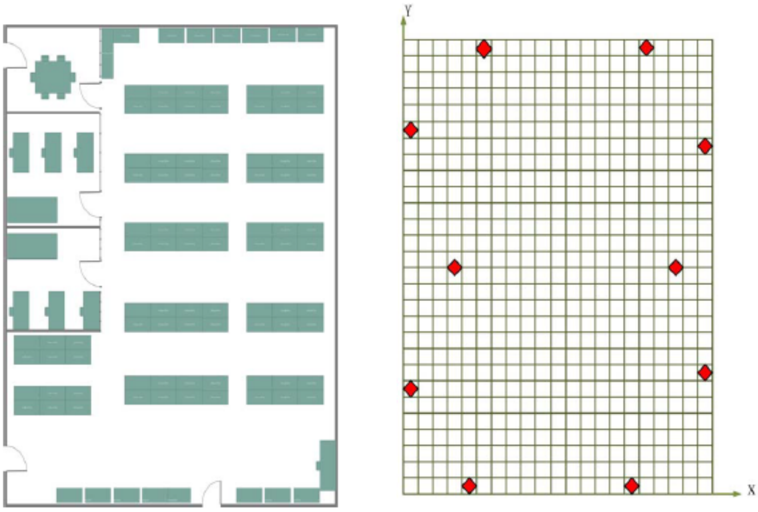


Figure 3.5: Segmentation for fingerprinting, [PFDZ16]

Fingerprinting

Fingerprinting is very different to the approaches presented above. Fingerprinting, or scene analysis, refers to the algorithms that first collect features (fingerprints) of a scene, stores them into a database and then estimates the location of the user by matching the present received signals against the vectors stored in the database. The signal measurements are commonly based on RSSI.[LDBL07]
This approach involves the division of the map of the building into multiple segments or a grid. (Figure 3.5) The algorithm consists of two stages. During the offline phase, or training phase, a site survey is performed. Each segment or cell on the grid is labeled with the location coordinates and on each of those positions, some unique attributes are collected to be able to distinguish each cell from another. In the context of wireless positioning, these attributes could be the signal strengths (RSSI), from nearby beacons or base stations, measured and stored as a vector into a database. [DM14] If RSSI is used as attribute, the average RSSI is stored for each cell or segment. [RGR+17] Each fingerprint is a vectors of RSSI averages and is labeled with the actual position of the map segment or grid cell. Afterwards, those fingerprints are stored in the database.
In the second stage, or online stage, the position estimation takes place. The user to be positioned uses a smartphone to measure the signals currently in range. With those signals, a fingerprint of the place where he/she currently is can be built. This fingerprint is compared with all fingerprints recorded in the offline phase. The fingerprints with the highest similarity are searched. By the combination of the fingerprints with the highest similarity, you can estimate the position of the user. [KMK16]

There are several algorithms, which try to find the similarities between those fingerprints. Some of them are: k-nearest-neighbor (kNN), neural networks,support vector machine

(SVM), and smallest M-vertex polygon (SMP).[LDBL07] Another approach is taken by [ZYW+15]. They propose a solution based on Vector Cosine. With this technique, the angle between two vectors can be calculated to find similar fingerprints.

The accuracy of the position estimation depends on the algorithm used on one hand and on the other hand on the quality of the fingerprints stored in the database. The quality of the fingerprints could be affected by radio interference. [KMK16][LDBL07]

These algorithms can be improved to be more resistant against radio interference as shown in [PFDZ16]. They propose the usage of an iterative weighted KNN algorithm instead of the traditional KNN approach. They evaluated their approach and it improved their accuracy by around 2 meters compared to their tests with traditional KNN. They say, that the use of iterative weighted KNN can reduce the impact of bad RSSI measurements caused by sudden interference of indoor environments, for example, if people are walking beside the user.

However, there are some more issues when using scene analysis for indoor localization. The time needed for the necessary training phase can be quite long since the measurement has to be done in each part of the facility. If changes in the infrastructure occur, for example the beacon deployment changes, this training phase must be performed again, since every fingerprint could have other measurements now. [RGR+17]
Another issue is the "smearing" of fingerprints during movement. In comparison to Wifi advertisements, which are propagated as a bulk internally in a smartphone, BLE advertisements are propagated immediately. On the one hand this means faster initial localization, but on the other hand it means that there has to be a time window in which RSSI measurements are taken to build the fingerprint. The choice of the width of this time window is crucial since it has to be big enough to ensure sufficient samples are taken to mitigate interference effects, but also small enough to minimize spatial smearing from movement. [KMK16][FH15]
The choice of window width depends on many factors. A few of them are movement rate (if the receiver is moving during the fingerprint collection, which means that the fingerprint is formed from measurements at different spatial locations), the presence of interference in the indoor environment (which is site specific) and the beaconing rate (since we do not want to miss advertisements). [FH15]
Fingerprints are also affected by the position of the user's body during measurements, as the human body is typically a good attenuator for radio frequencies. [FH14]

Another point to keep in mind is, that the results between Wifi fingerprinting and BLE fingerprinting can be quite different. [FH15] evaluated the same location with an existing Wifi infrastructure and afterwards with different densities of BLE beacons. They also discussed the parameters, which have an influence on the accuracy of the system. The most important ones are deployment density, advertising interval and transmission power levels. In their evaluation, they deployed 19 beacons in an office area (one beacon per 30 m2) and achieved tracking accuracies of <2.6m 95% of the time. When using a lower density beacon deployment (1 beacon per 100 m2), they achieve accuracies of <4.8m. Afterwards, they tested Wifi fingerprinting in the same location and achieved accuracies

of only <8.5m 95% of the time.
They mention, that they used the existing Wifi infrastructure, with access points that are not positioned optimally for indoor localization. The beacons have the advantage that they can be located in a way, in which they provide good signal geometry.
An interesting finding of them is, that beacon density decreased the positioning error, but only up to a certain threshold of around 8-10 beacons in range of the receiver. Beyond that threshold, there was no further improvement measurable. [FH15]

Another interesting approach is to combine BLE and Wifi fingerprinting. [KMK16] did exactly that and evaluated their solution. They used a weighted k-nearest-neighbors approach to compare the fingerprints. During their evaluation, they collected 680 fingerprints with 115.511 individual RSSI samples, which shows the workload of such an approach. They first evaluated their accuracy when only using Wifi and after that, they combined Wifi an BLE to improve their precision. Their accuracy of their Wifi only approach was 1 around meter and with the combination of BLE, they increased their accuracy to around 0.77 meters.

All in all, it can be said that this approach can provide a high precision if the environment does not change. [RGR+17] Fingerprinting can mitigate Non-Line-of-sight errors effectively, but is limited by the heavy workload in the offline phase, where fingerprints have to be acquired.[CJC+15]

3.1.3 Pedestrian positioning

Pedestrian positioning is a completely different approach than the other presented positioning techniques. With this approach, you can avoid the use of any kind of radio frequency signal to increase scalability, installation costs, maintenance and you mitigate interference effects. [EF15] Instead of deploying the positioning system in the building, the user is carrying the localization sensors. One way to use the sensor data is Dead Reckoning. When using Dead Reckoning as a navigation method, the user starts the navigation from a well-known location. The system adds the position changes to the initial position and calculates the new position and so on. In Pedestrian Dead Reckoning (PDR) the speed of movement and the direction of the movement is recorded.[NK11]
The sensors are activated by accelerations from translational and rotational movements of the smartphone. The sensors, which can be used for this approach, are accelerometers and gyroscopes whereas accelerometer-based mechanisms have been shown to be accurate, robust and practical for object monitoring.[JS17] [DD12]

There are several works that use PDR for localization. Most of them are in combination with other technologies to initialize the starting position of the user. [WH08] for instance used an inertial sensor that is mounted to the foot of the user, who wants to get positioned. They used a detailed facility map and were able to provide an absolute position with their approach. They were also able to handle stairs and multiple floors with their solution. They used Wifi fingerprinting to initialize the user's starting position.

[CAAG16] is another work that used PDR, but they used it to enhance their BLE fingerprint positioning system to be able to achieve a higher accuracy. Their fusion method yields and increased accuracy in all their test cases with a mean positioning error of 0.9 meters compared to 2.6 meters when only using BLE fingerprinting. This work shows how powerful PDR can be and what impact it can have on the accuracy of the indoor positioning system.

However, when having sensor data only, measurement errors can be a serious problem. The issue with PDR is that this error can cumulate over time since the position changes are always aggregated with the position before. This error could even have cubic growth in time, which is called drift. The survey in [Har13] showed, that PDR can provide good positioning under certain circumstances, but that periodical position corrections from other technologies, like Wifi or BLE, are needed to be able to ensure correct long-term operation. [ZCJ+17] tackled the drifting problem with the extension of BLE.

3.2 Interface and Interaction design

When developing a mobile application, it is important to take a look at the interface and interaction design. Apps are different from desktop-applications and therefore need to be designed in another way. It is important to think about the screen size, the users and their interaction with the app and many other aspects. In the next section some of these design-concepts and rules will be discussed.

To ensure that users interact correctly and efficiently with mobile applications (and any other kind of application), a higher level of usability is needed. Both, the user and the developer want the app to be easy to use, which is why they can help each other in the developing process, by testing the app together in the early application design phase. [Low13] [M.C13]

When developing an application, simplicity is a needed factor to obtain a high usability.

> „less is more“- Mies van der Rohe

Therefore, strategies for simplicity are needed, such as removing features which are not necessary at certain contexts. This is not always possible and can make the application even more complex.
Another point is hiding information which is not needed that often. Context-menus can be used to accomplish this.
For better understanding of the context, organizing the information into logical groups by placing related features closer to each other is helpful. Those groups can reduce complexity and ensure a high level of usability.

Ergonomics also play an important role in the designing process. Elements that are related should be aligned on a invisible grid with each other. Through that the screen looks more organized and overlooking.

Colors also have an impact on the usability. It has to be considered that every color looks different depending on the brightness settings of the smartphone. They should not look to bright. They should also be used with caution since too many colors can make the content look overloaded and distract the user.
To avoid those issues and the usage of not-matching colors, different tools can be used to get a user-friendly coloring scheme. Harald Küppers introduced a color chart in 1958, which could be used. Nowadays, there are many tools available online, such as the Material Design from Google. [gooc][Kü81][Bab17]

Icons or symbols are a very important topic in interaction design of mobile applications. Complex factors have to be taken into consideration when choosing them.
The user should not have to think much about what might be the functionality behind the icon. It is recommended to use graphics that have a reference on already known symbols and their meaning. No additional text should be needed to understand the meaning behind the used icon. There are some exceptions in which a label can help to make the usage more efficient and understandable.
When choosing the icon, the alignment, color and size have to be considered as well as the matching with the whole application layout and design.
To help making the usage of icons easier, there are already some that have an established context and will lead to a certain user expectation. Google's Material Design for example, is a common library for developing Android applications and can be used to ensure that the used icons will be recognized by the user in a certain context. [gooc][GSKJ13]

The application needs to give the user feedback, after executing an action. This can have different forms, such as visual, acoustic or haptic events. Visual events for example would be if something changes on the screen, like a notification that pops up.
It is important to ensure that the feedback is in the user's language and that the user's knowledge of the is considered because novice users might have a different understanding of the app than the developers do. The feedback is not helpful if the user does not understand its meaning. [Sav07]

3.3 Android-Platform

The Open Handset Alliance launched the Android platform in 2017. Mobile devices, which use a modified Linux kernel, can use this platform. Many apps that are developed for the Android platform are programmed in Java.

Newer Android devices (version 5.0 or higher) run all apps in their own process and with their own instance of the Android Runtime (ART). ART has many features, the major ones are the improved debugging support, more detailed diagnostic exceptions and that it is now possible to set watchpoints for specific monitoring.
Android additionally provides a set of runtime libraries that have lots of features, which is well-known from Java.[and]

To create an Android app, lots of APIs are provided. They are mostly written in Java and support core-reuse, modular system components and services. Those include an Activity Manager, which usage was mandatory for the late developed prototype.[and]

3.3.1 Material Design

Material Design was invented to combine newest technology with the classic principles of design.
Its main goal is to make user experience across platforms unified with a single underlying system.
Also, the developers of this guideline want others to expand it easily and therefore provide methods for expanding it. [mat]

> „Unlike real paper, our digital material can expand and reform intelligently. Material has physical surfaces and edges. Seams and shadows provide meaning about what you can touch.“- Matías Duarte

This proves that the whole language is inspired by the real world and its textures. This includes the reflection of light and shadows.
By applying shadows to each element a visual hierarchy comes into being. This generates the impression of elements lying above/underneath each other. [mat] Figure 3.6 shows some examples of how shadows should or should not be applied to objects.

Material Design architecture supports developers by building a strategy for the creation of a mobile application on the Android platform. [mat]

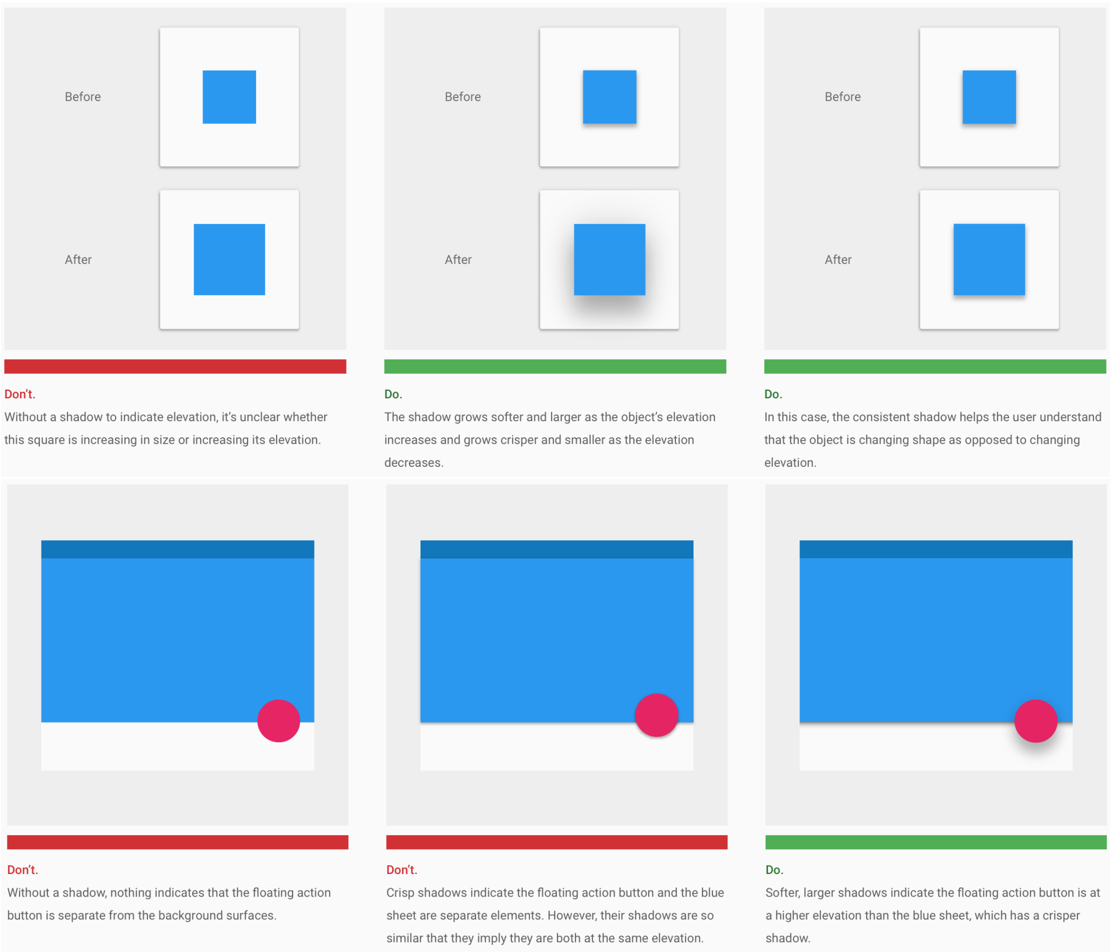


Figure 3.6: Dos and don'ts for applying shadows [mat]

CHAPTER 4

Methodology

This chapter describes the used methodologies. Different methods can be used for designing applications. This thesis will use the User-Centered Design method because an indoor navigation app will be developed, which should be simple an easy to use.

4.1 Literature research

At first, it was necessary to achieve knowledge about the used technologies referred to in this paper. Therefore, a comprehensive literature research was carried out.

It was important to lay a foundation for the development of an indoor navigation application, which in our case means the usage of beacons. To be able to do so, lots of papers on different beacon technologies and their usage have been analyzed.

One of the goals was to develop a functional prototype, which later on could evaluate whether BLE Technology is enough for precise indoor navigation or not. To do so, it was necessary to build knowledge about the Android Framework to be able to implement such a system.

For all those researches online academic libraries such as IEEE Xplore Digital Library [IEE] were used, as they offer many different papers, eBooks and articles, that lay the foundation for this thesis. The previous chapters (2 and 3) have mostly been influenced by those papers.

4.2 User-Centered Design

User-Centered Design integrates the user into the design process and helps to find issues with the design of an application early on. Since one of our goals was to develop an Android-App, it was important for us that this app can be used easily and therefore user feedback was important during the development phase.

It is suggested to consider the following factors when creating a User-Centered Design application:[Geo08]

- Learnability: It is easy to perform basic tasks as a new user and it does not take much to figure out the more challenging tasks.
- Effectiveness: It is easy to use the product.
- Efficiency: After the user learned the design, it can be used more quickly and at a higher pace. This is important for users that use the application frequently.
- Memorability: This aspect is essential for users that use the design infrequently.
- Attitude, satisfaction: The product fulfills the user's needs and has a pleasant and appealing design.
- Errors: Define how easy it is for the user to get back to a normal status if an error occurs. Also, the error message should be easy to understand.
- Usefulness: Does the system do what the user wants it to do? Does it behave as expected

After considering the above mentioned factors, it is necessary to carry out user evaluations. Those help improving the usability of the product. It is important to integrate the user in the design process because they will be the ones using the product in the end.

Incorporating the users throughout the whole process of the application development is important for the usability of the product. Therefore, continuous user testing is necessary.

> „Quality over quantity“

is a known phrase and can also be used for User-Centered Design. It is important to have an amount of representative users, rather than many different user-groups. Another essential decision to make is which method should be used for the user evaluations. In this thesis methods like the Cognitive Walkthrough 4.3.1 will be applied.

When involving users into the design process, it should not be forgotten that the users are not always right and that they are not designers.

User-Centered Design is iterative and therefore problems can be identified in each phase of the design and/or development process. The figure above (Figure 4.1) shows the steps of the User-Centered Design process.

All in all it can be said, that User-Centered Design has boundaries and opportunities. One disadvantage would be the fact that it is not suitable for radical innovations. It is also not usable for fundamental research and technological breakthroughs. An advantage would be practicing in unknown problem-fields. It is also useful for incremental improvements.

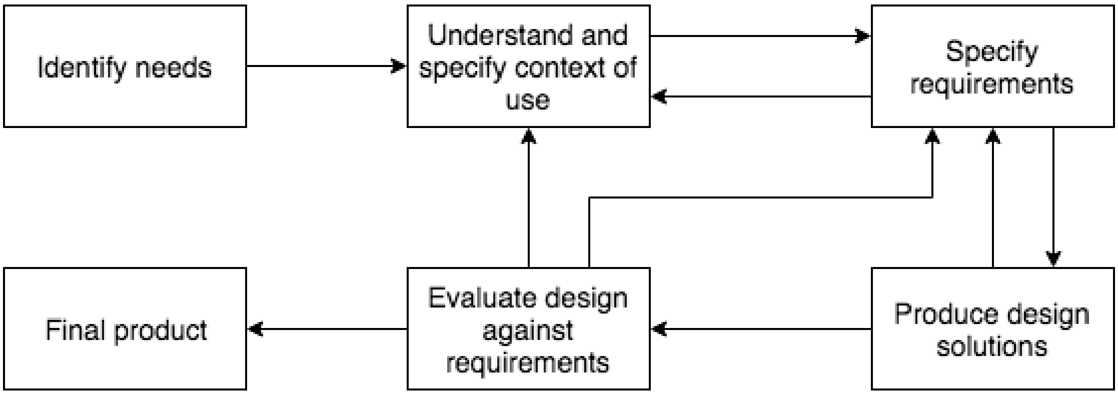


Figure 4.1: User-Centered Design Process by ISO 9241-210

4.3 Design Evaluation process

Based on financial and temporal trade-offs an evaluation method should be chosen. Also, the characteristics and quality of those methods play a major role in the selection. In the following section the used evaluation methods for this thesis are described. Later on these methods will be used to evaluate the developed prototype.

4.3.1 Cognitive Walkthrough

At first, evaluations were made with the Cognitive Walkthrough method. This method does not need test-users for testing. Developers acting like a novice user can perform these tests to ensure that the product is intuitive. The purpose of the Cognitive Walkthrough method is to make sure the elements and functions are understandable for the user. [RFR95] [MSK10]

To start the evaluation method, relevant user tasks have to be defined. Next, the easiest way to achieve a task has to be explained. Afterwards, testers follow this path and evaluate the results considering four criteria: [MSK10]

1. Which step is the next one to achieve the desired goal?
2. Is the user interface element for the next step perceivable?
3. Is there a connection between the element and its effect? Can other elements be considered to achieve the same effect?
4. Is the system responding correctly?

For each task these four questions have to be answered. Every deviation has to be commented and analyzed in order to find a solution for uncertainties or even issues.

The Cognitive Walkthrough is used by many developers because it is a reasonable way to analyze the user interface and fix problems in the early phase of the development. It is preferable because the tests can be performed by the developer and it saves costs since the evaluation takes place in early development phases. [RFR95] [MSK10]

4.3.2 Thinking-Aloud method

To evaluate the usability of the prototype, the Thinking-Aloud method was used. Usability tests confront the user with the application and how they think the app works. While observing the user, their behaviour can help identifying misunderstandings of the product and its cause. In order to better understand why users act the way they do, they are encountered to comment on every step they make and explain their thoughts. Users are also asked to give improvement ideas. All of the feedback is then documented by a test-observer and can be evaluated afterwards.

Different methods can be used to get the user's thoughts on different tasks. The test-observer can either note every word that has been said by the users or record them and transcript the recordings afterwards. Either way, the observer should collect the results and make plans for improvements.

CHAPTER 5

Prototype

The previously mentioned methods and the researched state of the art have been combined into the development of a prototype. This chapter will give a closer look into the implementation of an indoor navigation application prototype.

The following sections will give impressions on how the prototype GUI was designed – more precisely which user-survey lead to which decisions. The implementation of the app will be described and questions such as how the communication between the layers works will be answered. It will also be explained how the Estimote Library works with the prototype and why this library was chosen for the application. Limitations of the prototype and the used library will also be explained in a separate section. Finally, the prototype will be evaluated and the results will be presented.

5.1 Design

As mentioned in the chapter before (Chapter 4), User-Centered Design was used for the prototype design.

At first, a mock-up was designed. It was based on the Google Maps app[gma]. The first sketch was made by hand and is shown in the figure below. (5.1)

This first draft was then redrawn in Adobe Illustrator[ail]. Then the different possible states have been made and finally combined with Adobe XD [axd]. The generated clickable mock-up can be found online. [moc]

The evaluation was made with the above mentioned mock-ups (see Figure 5.2). For the tests, the previously explained methods were used. Five test persons have been chosen to evaluate the design of the prototype. The testers have been chosen from age 12 to 46 because in our opinion everyone with a smartphone should be able to use the app regardless of the age. It was also interesting to see the differences between the ones who already had contact to similar apps like Google Maps and those who did not. (Table 5.1)

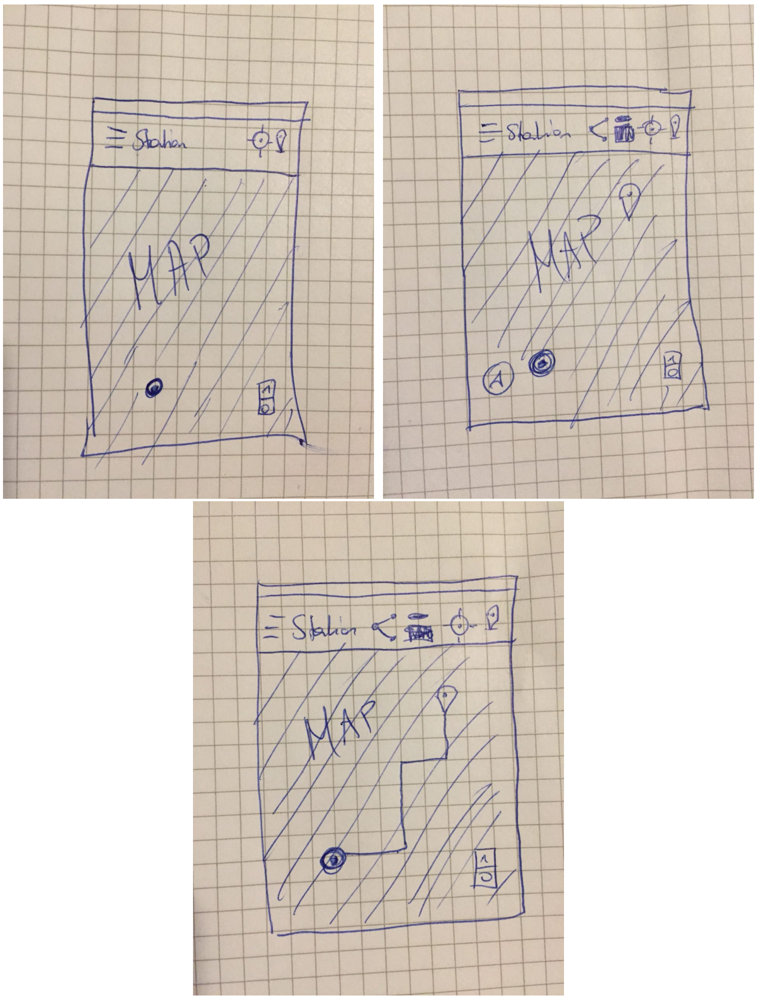


Figure 5.1: Sketches - Positioning, Set Destination, Navigate

Each user was given the same tasks (Tables 5.2, 5.3, 5.4) and the test observer took notes while watching the testers.

The results of the evaluations have shown that those users who have not used navigation apps like Google Maps before did not complain about the navigation-button being on the left side of the screen. Those who used such apps before said it would be easier to use the application if the buttons for the level-change would be switched with the one used for navigation.
Another point that the experienced users had doubts about was the position of the „show my current location“button. They said it would be easier to use if it the positions were like the ones in the Google Maps app (bottom-right). The users who have never used an navigation app before, did not even knew what this button was used for. They used the „floor-changing“buttons instead.

Figure 5.2: Mock Ups - Positioning, Set Destination, Navigate, Share Location

Nr.	Gender	Used Similiar App	Age
1	female	yes	24
2	male	no	46
3	male	yes	28
4	female	no	43
5	male	yes	12

Table 5.1: Test-Users

At last, the share-location pop-up was not as clear as thought by the developers. Many users were not sure if the „check“sign was just copying the link or already opening another application to share it.

The perceptions from those tests had influences on the developed prototype. That can be seen in figure 5.3.

Task: Set Location	Description
Exercise	Set a location on the first floor
Precondition	App is opened and the permissions (use current location) have been given
Exercise Final State	The destination icon is set on the first floor
Description for the test user	You are at a Subway station and want to set a destination which is on the first floor.

Table 5.2: Task 1 - Set Destination

Task: Navigate	Description
Exercise	Navigate from the current position to the set destination.
Precondition	The destination has been set.
Exercise Final State	A path is drawn from the current position marker to the set destination icon. Should be checked in both floors.
Description for the test user	You are at a Subway station and have set a destination somewhere on the first floor. The problem is that you are not sure how to get there so you let the app navigate you. You should make sure the path is drawn in both floors. (The one you are positioned at currently and the one where the set destination is.

Table 5.3: Task 2 - Navigate

Instead of showing the level by numbers, now an icon is used which symbolizes layers. After tapping on this button the different levels are shown. Also the position moved from right to left.
The button for locating the user moved to the right side. (Switched places with the layers-button.) This was a result from the user-evaluation. All of the testers were right-handed, which is probably why it was easier for them to have the buttons they use the most on the right side.
After setting the destination, the navigation-button appears on the right side - which seems to be more intuitive as previously discussed. In addition to that, the share-button and the delete-button are visible on the right side of the menu.
The accept and decline buttons of the pop-up, which are shown after clicking on the share-icon, were changed to make the usage easier. As mentioned above, the users were not sure what the symbols meant, which is why text was used for the prototype instead.

Task: Share Location	Description
Exercise	Share a set destination
Precondition	The destination has been set.
Exercise Final State	A pop-up is shown with a link and the possibility to copy it.
Description for the test user	You have chosen a destination where you would like to meet with a friend which has never been on the subway station at which you are right now. Therefore you would like to share a chosen point at the station to meet there.

Table 5.4: Task 3 - Share Location

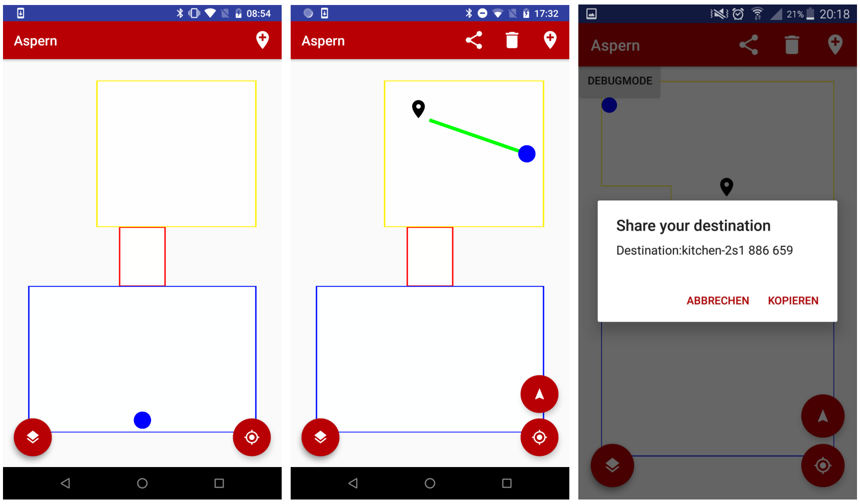


Figure 5.3: Prototype - Positioning, Navigate, Share Location

5.2 Implementation

The implementation was carried out with a layered architecture. This means that every layer can only communicate with its „neighbours“. An illustration can be seen in figure 5.4. [Ric15]

The prototype had the Estimote-Cloud as its Database Layer. Information like the user-position, location and the beacon-positions have been stored in this database.
Since the developed prototype needed more than just showing the user position and the location, it was necessary to build separate Location-Objects, which had additional parameters, such as which room is next to the current one and at what position does the room has a wall, etc. This was implemented in the Persistence Layer.
In the Business Layer the actual services such as navigation, user location and share

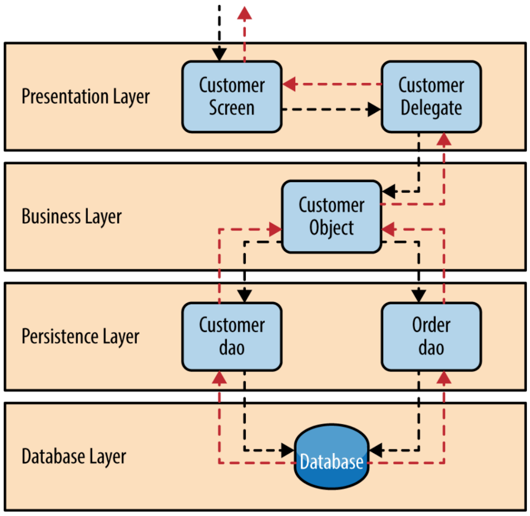


Figure 5.4: Layered Architecture, Software Architecture Patterns by Mark Richards (Chapter 1)

location takes place.
Since Android applications have Activities and Views, it was necessary to split the Presentation Layer. The Main-Activity is coordinating the app. In this class, it is defined what methods should be called on different button-pushes. Additionally, it is also used as interface between the Business Layer and the View. In other words, the Activity defines which view should be used depending on what information has been received by the Business Layer.
For the communication between the Activity and the Business Layer Broadcasts have been used. Such a Broadcast had only the needed information in it and nothing more. The other layers communicated over the in general used Java-Interfaces.

Another important aspect of the implementation were the authorizations. To use a navigation app it is necessary to get the approval from the user that the Bluetooth signals can be used. Before anything happens in the app, it has to be checked if the permission has been given. If not, a pop-up has to be shown to get those.
Not just the permission for the use of the location has to be checked. It is important to make sure that Bluetooth is activated in order to allow the user to be located with the help of the beacons.

5.3 Estimote-Library

The Estimote-Library uses a Cloud to manage beacons and locations. A location can be defined in two different ways. One is to use the Estimote Indoor-App for iPhones. The other one is to write a Swift- or C#-script and use an iPhone emulator to execute the code. (more about that can be found in Chapter 6)

It is possible to organize all the beacons in the Cloud but the adjustments that have been made to the beacons have to be applied to the beacons by using the Estimote App. With this app, the nearest beacons can be found and changes like the beacon's transmission power can be applied to the beacon. To be able to do that, Bluetooth has to be activated.

A main advantage of the Estimote-Library, in comparison to others like kontakt.io, is that Estimote provides an absolute position. This is helpful for further processing and the displaying of the user position on the screen.
Another advantage is the Estimote Cloud. With that, the developer does not need to gather information about the location, he simply connects to the cloud and is able to download the locations, which have been configured by the user before. Also, the position estimation algorithms are already provided, in comparison to other libraries, where only the beacon signal updates are provided.

All the above mentioned advantages led to the use of this library for the prototype development.

5.4 Limitations

Since the developed application is a prototype it is implicated that the app has some limitations.

First of all, in the mock-ups it is shown that it is possible to switch between floors. This functionality is not provided by the generated application. The GUI for this feature was already made but the back-end is not implemented.
Another point is that the app was only tested with nine beacons, which means that there was a limitation on resources and therefore the evaluation took only place in two-three rooms.
One major restraint is that the locations have to be drawn manually in addition to the one in the cloud. It is also necessary to know the exact position of each room at one place. This information also needs to be added manually.
Also, in order to show the location and the user position in the room, it is necessary to have an internet-connection. This is because the prototype needs to download the location from the cloud and afterwards exchange the information about the user position with the cloud. This means that for people, which are using this application as tourists in a different country than their own, there should be internet provided for free.

Unfortunately, some limitations came with the used Estimote-Library. The placing of the beacons for example is restricted. It is not possible to initialize a location which has beacons deployed to the roof of the room. Estimote needs at least one beacon for each wall deployed in a room. It would have been interesting to evaluate the performance of our system with beacons placed on the roof instead of walls, which would be the way of deployment in public facilities.
Another disadvantage of the used library is that the positioning can only be done in one room at a time. This means that in a location with multiple rooms the switching of

rooms has to be done differently. For this prototype the Altbeacon Library was used to find out which beacon is the closest and to which room it belongs. If the closest beacon is not in the room in which the positioning is currently happening, the Estimote library is informed and it changes the room accordingly.

5.5 Evaluation

In this section the evaluation of the prototype is documented. The test cases try to mock a real life scenario at a busy train station, where many people are around the signal receiving device and there is a lot of radio frequency interference by other devices, which may have Bluetooth or Wifi enabled. Not only are the devices from other people interfering, but also their body is blocking signals because of the amount of water in the human body. [bes] The influence of the beacon placement and the influence of the way the phone is held on the received signal strength (RSSI) were also part of this evaluation. The received signal strength depends on the distance to the beacon and some environmental factors. [IEE17][IEE10]
All measurements were taken three times and the mean values are presented to mitigate measurement errors.

5.5.1 General distance tests

Our first tests dealt with finding out which phone we should use for our test scenarios. We had a Samsung Galaxy S5 and a Nexus 5X. Also, the influence of the orientation of the beacon itself on the received signal strength was tested. For these tests, we used the Android app Locate from Radius Networks.[loc]
To get valid measurements, we had to stand still at the same position until the measurements were smooth and more importantly valid ones (this took around 10 seconds) because some were far beyond realistic distance measurements.

The test setup was a near to empty room with direct Line-of-sight between smart phone and beacon. These tests were made with a Samsung Galaxy S5 using Android 6.1. The beacon broadcasting power was set to -4dBm (which means that the beacon has a maximum range of about 40 m).
At first, we tried to figure out what phone position works best. The results of this test are presented in table 5.5.

This test showed us that the best position for the phone is to be held in the users hand with the top of the phone pointing directly to the beacon.

The next test was about finding the right position and orientation for the beacon itself. The results are shown in table 5.6.

The results show that the signals are better if the beacon is on the chair instead of the ground, probably because the wooden floor blocks too many signals.

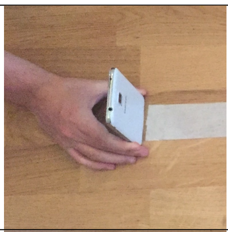
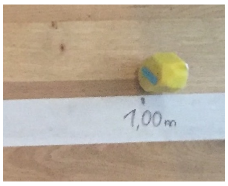
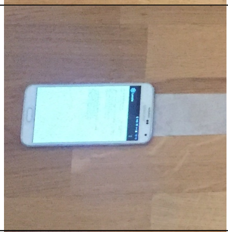
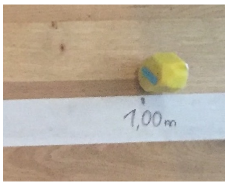
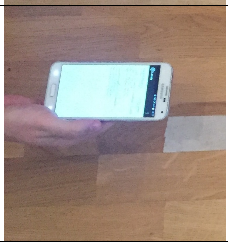
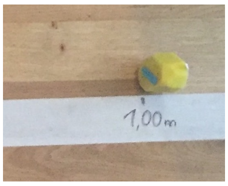
Phone position	Beacon position	Actual distance	Measured distance
		1 m	1.4 m
		1 m	1.9 m
		1 m	1.3 m

Table 5.5: Testing different phone positions

After these tests we know how we should place the beacons and how the phone should be held to achieve the strongest RSSI. 5.7

To see if another phone will have different measurements due to different Bluetooth chips, we repeated the distance tests with a Nexus 5X using Android 8.1. 5.8

As we can see the distances are not as accurate as with the Samsung phone, but the RSSI values are better than the ones that we measured with the Samsung phone. Better because the received signal is stronger, which is important for absolute indoor positioning.

The human body is also said to be a source of disturbance for radio frequency waves because of the high percentage of water. [bes] In a scenario where you are at a train station and want to get navigated to a point of interest, usually many people will be around you and therefore interfering with the signals your smartphone receives. To examine this interference, we had a person standing between the phone and the beacon. 5.9

We can see that the closer the person stands to the receiving device or the sending device, the stronger the loss of signal strength gets. This is not surprising since more signals get blocked that way.

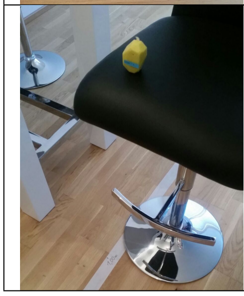
Beacon position	Actual distance	Measured distance
	1 m	1.1 m
	1 m	1.0 m

Table 5.6: Testing different beacon positions

Actual distance	Measured Distance
1 m	1.0 m, RSSI: -74
2 m	1.5 m, RSSI: -78
3 m	1.7 m, RSSI: -80
4 m	1.86 m, RSSI: -84

Table 5.7: Different distances - Samsung Galaxy S5

Actual distance	Measured Distance
1 m	0.1 m, RSSI: -52
2 m	0.26 m, RSSI: -59
3 m	0.6 m, RSSI: -64
4 m	1.0 m, RSSI: -74

Table 5.8: Different distances - Nexus 5X

5.5.2 Tests with our prototype

In the following section, we present the results of the test with our prototype. The tests aim to evaluate our prototype in a real life scenario where indoor navigation is needed, for example in a crowded area with people and devices interfering with your smartphone. The test setup is one room and the prototype was running on a Nexus 5X with Android 8.1. The room plan can be seen in figure 5.5. We set reference-positions, which are marked with an X. At these, we measured the delay between the person reaching the reference point and the time it takes the application to update the location of the user

Actual distance	Person standing at (measured from position of phone)	Measured Distance
3 m	0.5 m	1.2 m, RSSI: -78
3 m	2.5 m	1.05 m, RSSI: -70
3 m	1 m	1.05 m, RSSI: -73
3 m	holding hand in front of the phones Bluetooth receiver	1.5 m, RSSI: -84

Table 5.9: Different distances with human body as interference

to the reference point, then took the average value of a few measurements. Also the deviation of position estimation was analyzed.

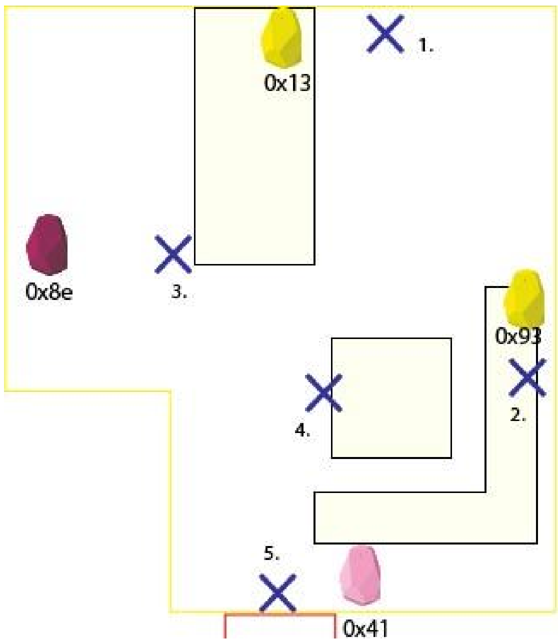


Figure 5.5: Setup for prototype tests

The first measurements of the accuracy of positioning and delay have been made in a room with no other person than the one testing.

In table 5.10, it can be seen that there have been delays of around two seconds, but up to three seconds, which is not a good result since this test has been carried out in an almost optimal environment. Another finding was that there have been some positioning inaccuracies. The reason might be the phone-position and/or the user blocking some signals with the body.

Reference point	Actual position (x, y)	Measured position (x, y)	Delay
1.	x: 3.3 y: 0.3	x: 3.0 y: 0.0	2 seconds
2.	x: 5.0 y: 3.2	x: 5.0 y: 2.7	3 seconds
3.	x: 2.5 y: 1.4	x: 2.5 y: 1.8	2.5 seconds
4.	x: 2.9 y: 2.5	x: 2.5 y: 4.5	3 seconds
5.	x: 3.2 y: 5.5	x: 2.5 y: 5.5	1 seconds

Table 5.10: Prototype tests, no active interference

Reference point	Actual position (x, y)	Measured position (x, y)	Delay
1.	x: 3.3 y: 0.3	x: 3.0 y: 0.0	2.5 seconds
2.	x: 5.0 y: 3.2	x: 5.0 y: 2.5	5 seconds
3.	x: 2.5 y: 1.4	x: 2.5 y: 2.0	4.5 seconds
4.	x: 2.9 y: 2.5	x: 2.5 y: 4.5	6 seconds
5.	x: 3.2 y: 5.5	x: 2.5 y: 5.5	1.5 seconds

Table 5.11: Prototype tests, BLE and Wifi interference

The table 5.11 shows tests, which were made with interfering signals. It can be seen that the last reference point had very short delay. This might be because this reference position was very close to a beacon, which helped the system to estimate the position faster.

Our test included four people being in the room and having Bluetooth and/or Wifi activated on their phones.
When people were standing in the room and Wifi was activated (Bluetooth deactivated), the positioning in the room had a delay of around 3 seconds, but up to 6 seconds. There have also been some fault-signals, which lead to the position to be shown wrong. For example, the user was standing on one side of the room, but the app would show him being on the opposite side, around four meters away. Also, we found some beacons being recognized better by the API than others even though the broadcasting power was set equally for each of them. This could be caused by factors like battery levels of the beacons. [IEE17]
When Bluetooth was activated, a similar behaviour occurred. Combining Bluetooth and Wifi did not change the delay (there was no delay increase as some might expect).
When all phones had Bluetooth and Wifi deactivated (except for the one which was using our prototype) we had better results, but there still have been some fault-signals. The reason for the fault-signals might be the presence of many people in the room blocking signals.

Estimote suggests to put the beacons on the walls at a certain height to overcome Line-of-sight issues. Therefore, we deployed our beacons on the walls at a height of around two meters. In that way, the signal attenuation of bodies seemed a little bit lower, resulting in an overall better accuracy and less delay. The results are shown in table 5.12.

Reference point	Actual position (x, y)	Measured position (x, y)	Delay
1.	x: 3.3 y: 0.3	x: 3.1 y: 0.0	2.2 seconds
2.	x: 5.0 y: 3.2	x: 5.0 y: 2.7	4 seconds
3.	x: 2.5 y: 1.4	x: 2.5 y: 1.8	4 seconds
4.	x: 2.9 y: 2.5	x: 2.5 y: 3.5	5.2 seconds
5.	x: 3.2 y: 5.5	x: 2.5 y: 5.5	1 seconds

Table 5.12: Prototype tests - beacons 2 m above ground

All in all, the positioning performance decreased by having interfering signals, such as devices with Wifi and/or Bluetooth enabled near to the user's phone and people being in the same room. The deployment of the beacons higher up on the walls reduced the signal attenuation caused by human bodies, but they were still present and measurable.

The errors of the position estimations were smaller than two meters most of the time, which is a reasonable result. The estimation errors were smaller for reference points closer to beacons. For reference number 4 for example, the estimation errors were sometimes quite big because the shown position would jump to the nearest beacon, which was at the right wall even though the reference point's position was in the middle of the room. The delay until the estimated position reaches the real position could be a problem. A delay of around 3 seconds means that a walking human with a walking speed of around two meters per second could be 6 meters apart from the position, which is shown on the map. That result is certainly not satisfactory and needs improvement. Possible improvements are presented in chapter 7.

CHAPTER 6

User Guide

In this chapter we will shortly explain the functionality of our prototype and how we implemented the use case.
The use case was divided into five steps, which will be discussed in the following sub-chapters.

6.1 Map location

Estimote ([esta]) offers a cloud in which locations and beacons can be managed. After ordering the beacons it was necessary to define which beacon should be in which location. This can be done in two different ways.

The first method includes downloading the Estimote Indoor Location App, which is only available for iPhone 4 or higher. (For Android the Indoor Location - functionality is not available.) Bluetooth should be activated on the phone because only then the beacons can be recognized by the app.
After downloading the app, the beacons can be placed in the room (it is recommended that each wall has at least one beacon). The user has to walk close to the walls beginning in one corner of the room until the app recognizes a beacon and directs the user to move the next beacon and so on until the user gets back to the first corner.
Afterwards, the app may suggest some re adjustments of some beacons for better geometry, to be able to provide more precise location performance in the room.

The other method for adding a location into the Estimote Cloud would be to write a small iOS application in Swift (or Objective-C).
Locations were created with Xcode[xco], which is a program to generate apps for iOS or MacOS. The code was written in Swift and the Indoor Location SDK from Estimote was used.

Within the locationBuilder, the corners of a room can be defined and afterwards the locationBuilder can be build.
Every boundary point connects to the previous one (the last point is automatically connected to the first one). In this way the walls are built. The order in which the „corners“are defined is important.
The x and y-coordinates are defined in meters.

```
let locationBuilder = EILLocationBuilder()
locationBuilder.setLocationName("room")
locationBuilder.setLocationBoundaryPoints([
    EILPoint(x: 0.00, y: 0.00),
    EILPoint(x: 5.00, y: 0.00),
    EILPoint(x: 5.00, y: 5.00),
    EILPoint(x: 0.00, y: 5.00)])
```

With addBeaconWithIdentifier, the beacons can be set to a particular wall. The identifier can be found either on cloud.estimote.com or in the Estimote-App, which is available for Android and iOS.
With the atBoundarySegmentIndex parameter, the beacon is set to a wall. The indexes start with „0“for the first wall, meaning the first defined by the first and the second corner. With index „1“, the second wall would be taken, defined by the second and the third point and so on.
With the inDistance parameter and the from parameter, it is possible to define where exactly beacons are placed on a particular wall. When standing in front of the wall and looking straight at it (left is leftSide and right is rightSide).

```
locationBuilder.addBeacon(withIdentifier: "d2e5a99f0a6157b4e89bb709977d6716",
                              atBoundarySegmentIndex: 0, inDistance: 2.5,
                                  from: .leftSide)

locationBuilder.addBeacon(withIdentifier: "b8d5a0caabab6dd856073a3a37d99526",
                              atBoundarySegmentIndex: 1, inDistance: 3, from:
                                   .rightSide)

locationBuilder.addBeacon(withIdentifier: "440bb6986feb036ae8a78eeca764123c",
                              atBoundarySegmentIndex: 2, inDistance: 2.5,
                                  from: .leftSide)

locationBuilder.addBeacon(withIdentifier: "e0dfa3041770efc1ddeb00687bac2c39",
                              atBoundarySegmentIndex: 3, inDistance: 2.5,
                                  from: .leftSide)

let location = locationBuilder.build()
```

Last but not least, the meta-data for the Estimote-Cloud has to be defined (AppID and an AppToken). This meta-data can be found on cloud.estimote.com. After adding an App online, the App-ID and App-Token are generated and can be inserted into the code-lines below.

```
ESTConfig.setupAppID("<App ID>", andAppToken: "<App Token>")
```

```
let addLocationRequest = EILRequestAddLocation(location: location!)
addLocationRequest.sendRequest { (location, error) in
    if error != nil {
        NSLog("Error when saving location: \textbackslash(error)")
    } else {
        NSLog("Location saved successfully: \textbackslash(location?.
            identifier)")
    }
}
```

After executing the code, the location is stored in the Estimote-Cloud and can be displayed as shown in the image above.

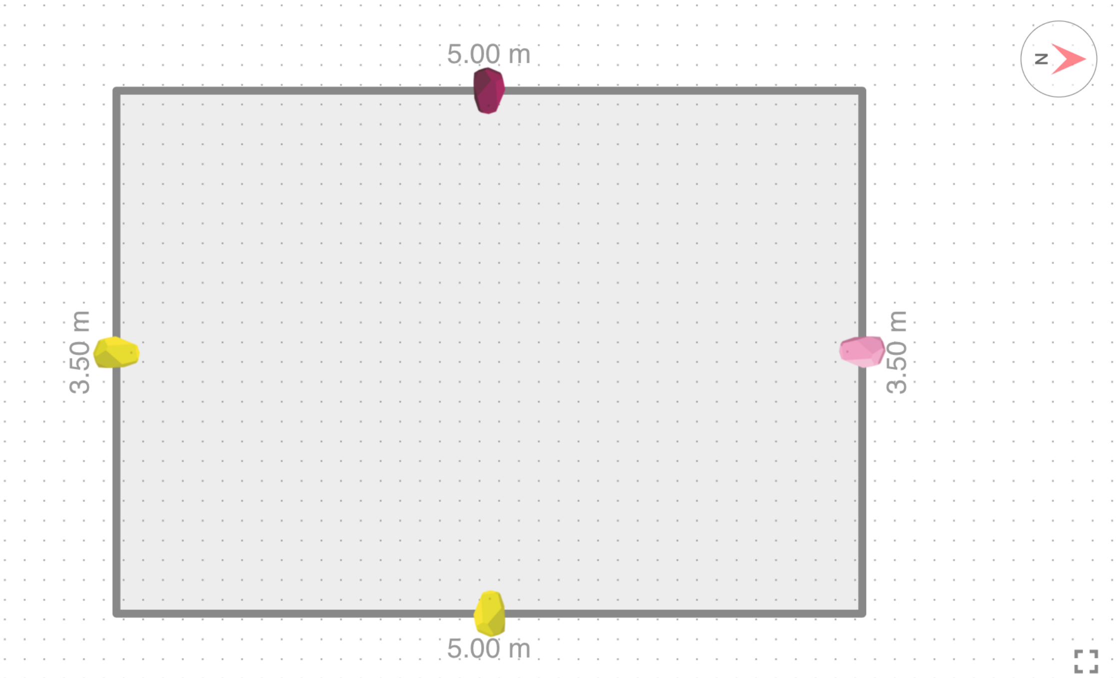


Figure 6.1: How the location is shown in the Estimote-Cloud

We prefered the second method because it was easier to adjust the parameters for each location after writing the program-code once. It was not necessary anymore to walk through the whole room from beacon to beacon. Another advantage was that the self written program did not need an actual phone because it could be executed by an iOS emulator.

Our goal was to show the user an overview of all rooms on the same floor of the whole location. The Estimote library comes with a special location view, but this view is only able to display one room and not the whole facility. So we had to program this view ourselfs. Therefore, we need an additional Xml-file, where all rooms are defined.

With pathData it is possible to define the paths, which represent the walls of a location. By setting fillColor differently, it is easier to distinguish the rooms from each other later on.

- „M“moves the cursor to a position (x,y) on the screen at which the drawing should start.
- „v“and „h“describe in which direction the line should be drawn. (vertical or horizontal)
- „z“closes the path - a line is drawn from the last cursor position to the first one.

```
<path
    android:name="natsKitchen"
    android:strokeColor="#FFF000"
    android:strokeWidth="3"
    android:fillColor="#FFFEFF"
    android:pathData="M200,75 v500 h350 v-500 h-350  z"
    />
```

6.2 User-Positioning

The User-Positioning is done with the Estimote-IndoorSDK, which offers a ScanningIndoorLocationManager that is able to locate a person in a specific location.
For the ScanningIndoorLocationManager to work properly, it is needed to load a location with the IndoorCloudManager and to add a positionUpdateListener to the locationManager.
The calculation of the position in the specific room is done by the Estimote-IndoorSDK. It had to be slightly adapted by us, so that the position was shown correctly on the GUI, for different smartphones with different display resolutions.

When saving a location into the Estimote-Cloud, the location gets an unique ID and this can be used afterwards to load the location from the cloud.
If the room is found, an indoorManager can then be set with the location as parameter. In the cloudCredentials we saved our Estimote-AppID and AppToken. The setOnPositionUpdateListener allows us to get a position within the current location.

```
IndoorCloudManager cloudManager = cloudManager.getLocation("room-841", new
    CloudCallback<Location>() {

    public void success(final Location location) {
        ScanningIndoorLocationManager indoorManager = new
            IndoorLocationManagerBuilder(BeaconService.this,location,
            cloudCredentials).withDefaultScanner().build();
        indoorManager.setOnPositionUpdateListener(new
            OnPositionUpdateListener() {

            public void onPositionUpdate(LocationPosition position) {
```

```
                    BeaconService.this.position = position;
                }
            });
        }
});
```

To avoid constant switching between rooms whenever a beacon from a wrong room is perceived as the closest one, we had to add a counter. After every signal scan cycle, the closest beacon is searched. If the same beacon was the closest one in 3 scans in a row, the room will be switched.

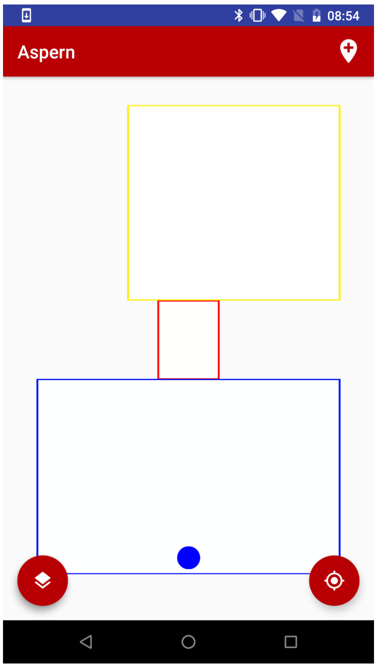


Figure 6.2: Show user position

6.3 Set Destination

To set a destination, the user has to tap on the icon in the top right corner of the display. After that, the user should tap on a point within the location and then the icon will be set there.

If the user wants to change the destination, the destination-icon in the menu has to be tapped again and the icon will be relocated to the next tapped position.

The user can also completely delete the destination-icon from the locations - by tapping the trashcan, which appears in the menu after a destination was set.
Since we are drawing the destination-icon on a bitmap, we had difficulties with the delete-option because it is not possible to just delete the icon from the bitmap. Therefore, we

needed to automatically create a new bitmap when the destination deleted. Afterwards, we had to draw a new bitmap with the old information in it - such as locations, zoom and user position.

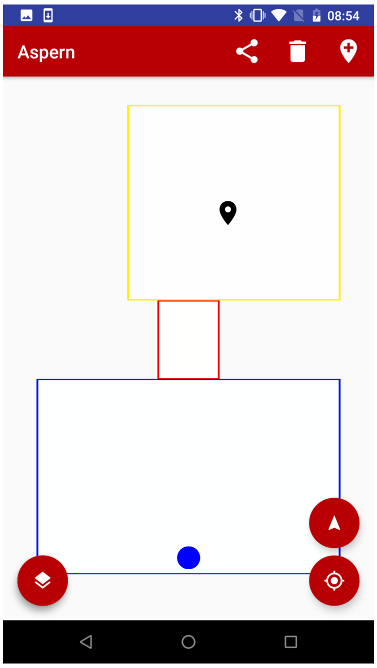


Figure 6.3: Set a destination

6.4 Share Location

After setting a destination, the user has the option to share it. The share-button opens a small window in which a link to the destination is created. This link can then be copied and sent via messaging-apps.

The person to which the link was sent has to have the app installed to proceed. If the app is installed, the user should be able to open the same view of the location (with the destination set), as the one who sent the link.

6.5 Navigation

The navigation was implemented in two steps. At first we implemented the navigation within a location (destination and user-location are in the same room). This was easier because the navigation-line is drawn between the destination-icon and the current user's position-point.

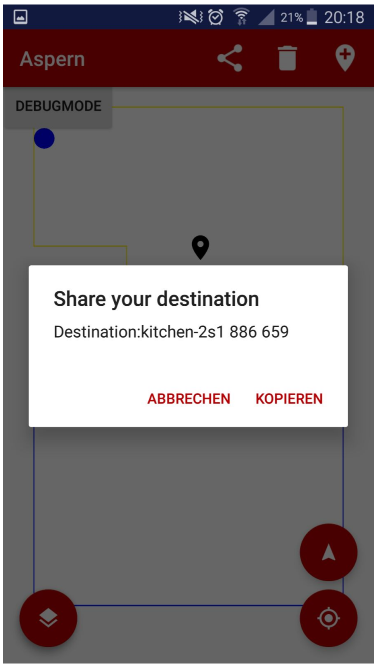


Figure 6.4: Share destination

The second step was to allow the navigation through different rooms. Therefore, it was necessary to store x- and y-coordinates of the doors of each room and the location which is behind that door. Then the navigation-line can be drawn from the destination to the first door, from there to the next door (if needed) and finally to the user-position.

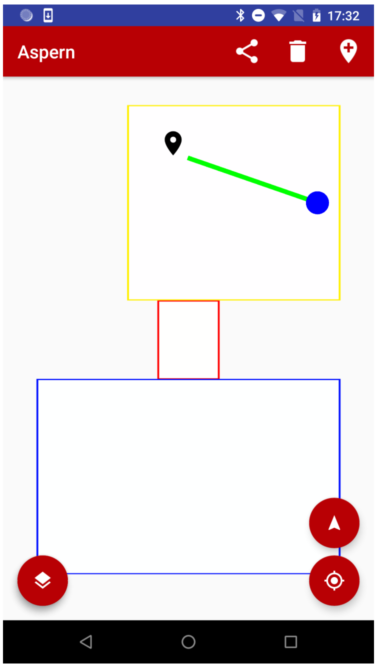


Figure 6.5: Navigating

CHAPTER 7

Limitations and future work

In this chapter the limitations of our solution are presented. Also, possible extension to our prototype are presented to improve the accuracy and reduce the delay of the positioning process.

Our prototype uses only one source of information, namely the Bluetooth signal. To process this information, we used the Estimote Indoor API ([estc]). This library comes with many advantages, but also with some limitations, which are presented in chapter 5.4.

Depending on the use case, on the surrounding area and especially on the presence of interference, RSSI only solutions are able to provide satisfactory results, up to 1 meter precision in optimal situations. An optimal environment would be present, if Line-of-sight is given at all times. If LOS is not given, the accuracy decreases tremendously (up to 5 meter errors).[EF15] In our scenario, where many people are around the sender and receiver of the Bluetooth signals, Line-of-sight can not be guaranteed. Furthermore, interference from other Bluetooth or Wifi enabled devices is present, which further increases the error, reduces the accuracy and usability.

We evaluated the performance of our RSSI-based indoor positioning system in such an environment. We had people blocking the Line-of-sight to some beacons and their phones had BLE and/or Wifi enabled. We found that not only the precision but also the delay of our prototype was too high to be usable in a real life scenario. This corresponds to the literature.[EF15][DD12]
Our solution would only be usable in big halls or buildings, where high precision is not necessary, but most indoor positioning scenarios require high accuracy to be able to generate added value for the user.

There are some ways to improve the accuracy of indoor navigation systems. The key is to combine multiple technologies. There are some works which use Bluetooth in combination Wifi. This approach can increases the accuracy up to 30% [KMK16]. Another way could

be the combination with UWB, which would lead to high accuracy, but also to increased costs and not all phones would be able to receive the UWB signals since special hardware is needed.
The work [Kol17] presents the concept of a hybrid positioning system with Bluetooth Low Energy and Ultra-wideband. The idea is to calculate the position by combining the BLE RSSI measurements and UWB measurements. The advantages would be that you get fast update rates, due to BLE, but also high accuracy because of UWB. BLE has a lower power consumption, so BLE measurements are taken more often. UWB updates are used to periodically improve the position estimations from BLE.
Their evaluation shows, that including UWB measurements into the localization calculation leads to a better outcome of more than 60% in comparison to their test when only using BLE as information source. Their solution is a tradeoff between accuracy and energy consumption since UWB has a relatively high energy consumption. The energy consumption can be lowered by reducing the UWB measurement rate with still having significantly better accuracy. The 60% increased accuracy was achieved with an UWB update rate of 5Hz. With an update rate of 0.5Hz the increased accuracy was still 30% better than without using UWB measurements at all.

The drawback of these hybrid approaches is that all of those suffer from the same kind of radio frequency interference. To overcome this problem, an information source which is not based on radio frequency signals is needed.

Most currently available commercial solutions are using positioning systems that are based on sensor fusion, which means that they use data from different sources like BLE, Wifi and internal sensors.
One example would be indoo.rs. This company was founded in 2010 and since then provides state-of-the-art indoor positioning systems. Their current system works with either Bluetooth Low Energy signals, more precisely iBeacon, or Wifi signals. The positioning technique they are using is fingerprinting.
Their focus is to provide a high precision real-time positioning system. To achieve this, sensor fusion is implemented. The Radio frequency signals are combined with a second source of information, sensor data. This data is obtained by the smartphone's compass or the accelerometer.[ind]
The company guarantees a precision of 5 meters in 95% of the cases (during movement!), but they also say that the precision can be easily increased to around 2 meters, by simply adding more beacons to the infrastructure.

Another company, which provides indoor positioning systems, is SenionLab [sen]. Their positioning system uses proprietary beacons, called SenionBeacons. The SenionBeacon is a Bluetooth Low Energy chip, which uses the iBeacon format. The user has to use a calibration app to collect data samples from the deployed beacons before the indoor positioning system can be used. Together with a map, the data is sent to SenionLab, which then provides the customer with an app and an API to be able to further develop and customize the app, depending on the needs of the customer. Just like indoo.rs, SenionLab can also provide a solution that is based on Wifi fingerprinting, if needed.

CHAPTER 8

Analysis and findings

This chapter presents the findings of our work and analyses them based on our literature research.

Bluetooth Low Energy has some major advantages over other technologies which could be used for indoor positioning. The most important one is probably the low energy consuming beacons. BLE beacons are powered by a small coin cell battery, which makes the device small and easy to deploy nearly anywhere. On the other hand Wifi access points are bound to power outlets. Furthermore, the position update rate can be higher if BLE is used in comparison to Wifi, which is important when movement during the positioning process is present. If the update rate is too low during movement, signal smearing may appear. Ultra-wideband is also a promising technology in this sector, but is coupled with high hardware costs and UWB signal receivers are not available when using most recent smartphones.

It is important to keep in mind, that a high update rate results in a higher power consumption. There is a tradeoff between energy consumption and update rate and therefore the accuracy of the positioning system. But not only the update rate or advertising interval has impact on the energy consumption, also the configured transmission power has impact on the energy used. [EF15]

The positioning technique we used in our prototype was Trilateration. The reason is, that Trilateration is not as vulnerable to positioning inaccuracy caused by movement or Line-of-sight issues, like Triangulation or Fingerprinting. Triangulation needs a Line-of-sight environment at all times and that is not given in a real life scenario. Fingerprinting has the need of collecting fingerprints in a learning phase. During the positioning phase, the measurement is done during movement and is interfered by the user's environment, which could cause different signals than the fingerprints in the database have. This may causes positioning inaccuracy.

We tested our solution extensively in different scenarios. We wanted to evaluate the impacts of different sources of interference. The interference types we tested were other Bluetooth devices, Wifi devices and the presence of people in the same room.

The evaluation showed, that the biggest enemy of our system is the human body. We analyzed the interference of the human body and found that even one person in the same room increases the delay of our system. The interfering person does not even have to block the Line-of-sight to a beacon, just the presence was enough to reduce the system's performance.
We proceeded to deploy our beacons on the walls at a height of about 2 meters and even then the human body in the room had a negative impact on the performance. The reason could be a combination of the multipath propagation and the human body interference coming from the water portion of it.

Another finding was, that the delay even without an additional person in the room is too high for everyday use. When the user was standing still the device had time to compute the position out of different RSSI values, but during movement, this computation took too long to be relevant. The delay during movement was around 3 to 3.5 seconds, which does not seem much at first, but an average human walks at around 1.5 to 2 meters per second, which means that the person already walked 4.5 to 7 meters further till the prototype shows the old position. The problem is, that even when standing still RSSI values are fluctuating very much and therefore the prototype needs a few good measurements before it is able to give an estimation of the real position. During movement every measurement is different and therefore the estimation is way off or way behind.

Another factor, which has influence on the RSSI, are the devices you use. Therefore, we analyzed the RSSI values measured with two different devices and the outcome showed, that the differences were quite big. The distance estimations were completely different between those two devices, which shows how hard it is to build an indoor navigation system based on RSSI for a wide variety of devices because every model has a different Bluetooth chip, a different antenna and is built out of different materials, which have different interference potentials.

After this evaluation and literature research, we conclude, that BLE RSSI as only source of information is not sufficient for indoor positioning systems because of the high fluctuation of RSSI, due to radio frequency interference. [DD12] The main problem we see, which is not mentioned in the literature, is the position delay, which makes the solution even less applicable for real life situations.

Since our solution is not applicable for real life scenarios, we searched for possible improvements for our prototype. There are several ways to combine different technologies together to get more sources of position data, like the combination of BLE with Wifi([KMK16]) or UWB ([Kol17]).
The problem is, that all of those technologies suffer from radio frequency interference. So we searched for something else and found sensor fusion, which is the common way for indoor positioning in currently available commercial solutions. The idea is to combine

the data gathered from an accelerometer, gyroscope or compass, which can be found in most currently available smartphones, and use that as a second source of information. The data collected by those sensors can be used to analyze the motion of the user and to combine this information with signals received from beacons to get an overall better performance, but most importantly a more responsive solution.
The reported performances seem promising. The work [CAAG16] for example presents such an approach. They have reached a positioning accuracy of 0.9 meters when using sensor fusion compared to an error of 2.6 meters when using BLE only.
[ind] claim that they can reach an accuracy of around 2 meters, if a dense beacon infrastructure is present, which would be a satisfactory result for most indoor navigation scenarios.

CHAPTER 9

Conclusion

Indoor navigation systems have been extensively researched in the last few years. Bluetooth Low Energy brings major advantages in comparison to other technologies, but especially when it comes to crowded places, BLE has its drawbacks.
The target of this work was to evaluate, whether an indoor navigation system, which uses only Bluetooth signals for position estimations, is usable in a real life scenario with a lot of interference nearby.

To achieve this, we implemented a prototype based on the Estimote Indoor SDK with 9 Estimote beacons. The SDK provides a position estimation based on the received signal strength. The position can be calculated with RSSI Trilateration, where distance estimations are used to estimate the user's position.

To test our prototype's performance when used in an environment with a lot of interference, we had other smartphones with Bluetooth and/or Wifi enabled close to our receiving device and other people staying in the room and blocking Line-of-sight to some beacons.

Our tests showed, that our prototype is able to achieve an accuracy of around 2 meters, which is a reasonable result. BLE and/or Wifi enabled devices increased the delay from between 1 to 3 seconds to between 1 and 5.2 seconds during the positioning process, but did not influence the estimated position as much as we expected.

We do not see the position inaccuracy as our main problem. The major issue is the delay between the real position and the time the system needs to show that position. Interestingly, the literature talks mostly about positioning inaccuracy but not of delay. Our tests, which included movement during the positioning, resulted in a delay of around 1 to 3 seconds in an optimal environment and between 1.5 and 6 seconds if strong interference was present. If a person walks with 2 meters per second, and the delay is 3.5 seconds, the position estimation is already up to 7 meters behind, which is too inaccurate for most indoor positioning scenarios.

We conclude, based on these tests and the literature, that a BLE only approach is not applicable for indoor navigation scenarios because the accuracy is not very good compared to other technologies, and additionally, the delay makes the performance of such systems even worse.

To overcome this problem, we researched and presented hybrid approaches, which are necessary to provide satisfactory indoor position estimations. The key is to combine technologies to get data from different sources. There are approaches, which combine BLE with Wifi or UWB, but all these technologies suffer from the same radio frequency interference. Therefore, we suggest the combination of BLE with PDR, which uses sensor data to provide additional positioning data.
BLE in combination with PDR achieves promising accuracy of about 2 meters during movement and 0.9 meters when standing still during measurement. [ind] [CAAG16] These results are accurate enough for most indoor positioning scenarios.

List of Figures

List of Tables

Bibliography

[ail] Adobe illustrator. https://www.adobe.com/at/products/illustrator.html. Accessed: 2018-06-25.

[ais] Aislelabs, beacon guide. https://www.aislelabs.com/reports/beacon-guide/. Accessed: 2018-04-29.

[alt] altbeacon, altbeacon sdk github. https://github.com/AltBeacon. Accessed: 2018-06-15.

[and] Android platform. https://developer.android.com/guide/platform/. Accessed: 2018-06-15.

[arg] Argenox, a ble advertising primer. http://www.argenox.com/bluetooth-low-energy-ble-v4-0-development/library/a-ble-advertising-primer/. Accessed: 2018-06-07.

[ars] arstechnica, meet google's "eddystone"—a flexible, open source ibeacon fighter. https://arstechnica.com/gadgets/2015/07/meet-googles-eddystone-a-flexible-open-source-ibeacon-fighter/. Accessed: 2018-06-02.

[axd] Adobe xd. https://www.adobe.com/at/products/xd.html. Accessed: 2018-06-25.

[Bab17] Nick Babich. The underestimated power of color in mobile app design. 2017. Accedded: 2018-06-15.

[bea] beaconstac, ibeacon vs eddystone: Which one works better for your pilot project? https://blog.beaconstac.com/2016/01/ibeacon-vs-eddystone/. Accessed: 2018-06-01.

[bes] Best practices for installing estimote beacons. https://community.estimote.com/hc/en-us/articles/202041266-Best-practices-for-installing-Estimote-Beacons. Accessed: 2018-05-27.

[Blua] Bluetooth, radio versions. https://www.bluetooth.com/bluetooth-technology/radio-versions. Accessed: 2018-04-28.

[Blub] electronicdesign, bluetooth low energy finally earns some respect. http://www.electronicdesign.com/energy/bluetooth-low-energy-finally-earns-some-respect. Accessed: 2018-04-28.

[CAAG16] Vivek Chandel, Nasimuddin Ahmed, Shalini Arora, and Avik Ghose. InLoc: An end-to-end robust indoor localization and routing solution using mobile phones and BLE beacons. In *2016 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*. IEEE, oct 2016.

[Cis] Cisco, indoor wifi location and beacons: Better together. https://blogs.cisco.com/wireless/indoor-wifi-location-and-beacons-better-together. Accessed: 2018-04-15.

[CJC+15] Hosik Cho, Jianxun Ji, Zili Chen, Hyuncheol Park, and Wonsuk Lee. Accurate distance estimation between things: A self-correcting approach. 1:19–27, 08 2015.

[CPH+14] Keuchul Cho, Woojin Park, Moonki Hong, Gisu Park, Wooseong Cho, Jihoon Seo, and Kijun Han. Analysis of latency performance of bluetooth low energy (BLE) networks. *Sensors*, 15(1):59–78, dec 2014.

[DD12] Qian Dong and Waltenegus Dargie. Evaluation of the reliability of rssi for indoor localization. https://www.rn.inf.tu-dresden.de/dargie/papers/icwcuca.pdf, 2012.

[DM14] Erik Dahlgren and Hasan Mahmood. Evaluation of indoor positioning based on bluetooth r © smart technology. 2014.

[EF15] Marcel Estel and Laura Fischer. Feasibility of bluetooth ibeacons for indoor localization. In *DEC*, 2015.

[esta] Estimote. https://estimote.com. Accessed: 2018-05-10.

[estb] Estimote, beacon specifications. https://community.estimote.com/hc/en-us/articles/204092986-Technical-specification-of-Estimote-Beacons-and-Stickers Accessed: 2018-04-14.

[estc] Estimote, estimote indoor sdk. https://github.com/Estimote/Android-Indoor-SDK. Accessed: 2018-06-06.

[estd] Estimote, estimote proximity sdk. https://github.com/Estimote/Android-Proximity-SDK. Accessed: 2018-06-06.

[este] Estimote, physical-world context 101: Proximity vs location. http://blog.estimote.com/post/148730623715/physical-world-context-101-proximity-vs-location. Accessed: 2018-04-14.

[estf] Estimote, proximity beacons vs location beacons. https://forums.estimote.com/t/proximity-beacons-vs-location-beacons/6947. Accessed: 2018-04-14.

[estg] Estimote, what are broadcasting power, rssi and other characteristics of a beacon's signal? https://community.estimote.com/hc/en-us/articles/201636913-What-are-Broadcasting-Power-RSSI-and-other-characteristics-of Accessed: 2018-05-20.

[esth] Estimote, what are potential sources of wireless interference. https://community.estimote.com/hc/en-us/articles/200794267-What-are-potential-sources-of-wireless-interference-. Accessed: 2018-05-11.

[esti] Estimote, what is a beacon protocol. https://community.estimote.com/hc/en-us/articles/208546097-What-is-a-beacon-protocol-. Accessed: 2018-05-20.

[estj] Estimote, what's the battery life of estimote beacons? https://community.estimote.com/hc/en-us/articles/202552866-How-to-optimize-battery-performance-of-Estimote-Beacons-. Accessed: 2018-05-03.

[FH14] R Faragher and R Harle. An analysis of the accuracy of bluetooth low energy for indoor positioning applications. 1:201–210, 01 2014.

[FH15] Ramsey Faragher and Robert Harle. Location fingerprinting with bluetooth low energy beacons. *IEEE Journal on Selected Areas in Communications*, 33(11):2418–2428, nov 2015.

[Geo08] Carole A. George. *Usability Evaluation Methods*. 2008.

[git] Google, eddystone github. https://github.com/google/eddystone. Accessed: 2018-06-15.

[gma] Google maps app. https://play.google.com/store/apps/details?id=com.google.android.apps.maps&hl=de. Accessed: 2018-06-25.

[gooa] Google, eddystone. https://developers.google.com/beacons/eddystone. Accessed: 2018-06-15.

[goob] Google, eddystone eid. `https://developers.google.com/beacons/eddystone-eid`. Accessed: 2018-06-15.

[gooc] Google, material design. `https://developer.android.com/design/`. Accessed: 2018-06-15.

[GOP12] Carles Gomez, Joaquim Oller, and Josep Paradells. Overview and evaluation of bluetooth low energy: An emerging low-power wireless technology. *Sensors*, 12(9):11734–11753, aug 2012.

[GSKJ13] Syed Ghayas, Suziah Sulaiman, Muzafar Khan, and Jafreezal Jaafar. The effects of icon characteristics on users' perception. pages 652–663, 2013.

[Har13] Robert Harle. A survey of indoor inertial positioning systems for pedestrians. *IEEE Communications Surveys & Tutorials*, 15(3):1281–1293, 2013.

[hon] Honeywell, what is bluetooth adaptive frequency hopping (afh)? `https://support.honeywellaidc.com/s/article/What-is-Bluetooth-Adaptive-Frequency-Hopping-AFH`. Accessed: 2018-04-28.

[HQ16] Katrin Hartmann and Andrew Quirnn. How useful are BLE beacons for mobile guides? In *2016 5th International Conference on Wireless Networks and Embedded Systems (WECON)*. IEEE, oct 2016.

[IEE] Ieee xplore digital library. `https://ieeexplore.ieee.org/Xplore/home.jsp`. Accessed: 2018-06-13.

[IEE10] IEEE. *P802.11 - IEEE Draft Standard for Information Technology - Telecommunications and Information Exchange Between Systems - Local and Metropolitan Area Networks - Specific Requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. 2010.

[IEE17] IEEE. *Optimized indoor positioning for static mode smart devices using BLE*. 2017.

[ind] indoo.rs. `https://indoo.rs/`. Accessed: 2018-07-10.

[JKJC15] Myungin Ji, Jooyoung Kim, Juil Jeon, and Youngsu Cho. Analysis of positioning accuracy corresponding to the number of BLE beacons in indoor positioning system. In *2015 17th International Conference on Advanced Communication Technology (ICACT)*. IEEE, jul 2015.

[JS17] A.R. Jimenez and F. Seco. Finding objects using UWB or BLE localization technology: A museum-like use case. In *2017 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*. IEEE, sep 2017.

[JSSN18] Kang Eun Jeon, James She, Perm Soonsawad, and Pai Chet Ng. BLE beacons for internet of things applications: Survey, challenges, and opportunities. *IEEE Internet of Things Journal*, 5(2):811–828, apr 2018.

[KMK16] Pavel Kriz, Filip Maly, and Tomas Kozel. Improving indoor localization using bluetooth low energy beacons. *Mobile Information Systems*, 2016:1–11, 2016.

[Kol17] Marcin Kolakowski. Kalman filter based localization in hybrid BLE-UWB positioning system. In *2017 IEEE International Conference on RFID Technology & Application (RFID-TA)*. IEEE, sep 2017.

[kona] kontaktio. `https://www.kontakt.io`. Accessed: 2018-06-15.

[konb] kontaktio, kontaktio sdk github. `https://github.com/kontaktio`. Accessed: 2018-06-15.

[Kü81] Harald Küppers. *Basic Law of Color Theory*. Barron's Educational Series Inc.,U.S., 1981.

[LDBL07] Hui Liu, Houshang Darabi, Pat Banerjee, and Jing Liu. Survey of wireless indoor positioning techniques and systems. *IEEE Transactions on Systems, Man and Cybernetics, Part C (Applications and Reviews)*, 37(6):1067–1080, nov 2007.

[loc] Locate beacon app. `https://www.radiusnetworks.com`. Accessed: 2018-05-27.

[Low13] Jonas Lowgren. Interaction design - brief intro. *The Encyclopedia of Human-Computer Interaction, 2nd Ed.*, 2013.

[mat] Material design. `https://material.io/design/introduction/#principles`. Accessed: 2018-06-15.

[Max] Maximintegrated, gaussian-frequency-shift-keying. `https://www.maximintegrated.com/en/glossary/definitions.mvp/term/Gaussian-frequency-shift-keying/gpk/519`. Accessed: 2018-04-29.

[M.C13] John M.Carroll. Human computer interaction - brief intro. *The Encyclopedia of Human-Computer Interaction, 2nd Ed.*, 2013.

[met] Metmuseum, beacons: Exploring location-based technology in museums. `https://www.metmuseum.org/blogs/digital-underground/2015/beacons`. Accessed: 2018-06-27.

[moc] Clickable mock-up. `https://xd.adobe.com/view/521014da-4aae-4d9c-88bb-1420cb62b8bd/`. Accessed: 2018-06-27.

[MPB+17] Zixiang Ma, Stefan Poslad, John Bigham, Xiaoshuai Zhang, and Liang Men. A BLE RSSI ranking based indoor positioning system for generic smartphones. In *2017 Wireless Telecommunications Symposium (WTS)*. IEEE, apr 2017.

[MSK10] Thomas Mahatody, Mouldi Sagar, and Christophe Kolski. State of the art on the cognitive walkthrough method, its variants and evolutions. *International Journal of Human–Computer Interaction*, 26(8):741–785, 2010.

[Mul] Multipath propagation. ://www.radio-electronics.com/info/propagation/multipath/multipath-propagation-basics-tutorial.php.

[NJNR17] Quang Huy Nguyen, Princy Johnson, Trung Thanh Nguyen, and Martin Randles. Optimized indoor positioning for static mode smart devices using BLE. In *2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*. IEEE, oct 2017.

[NK11] Klaithem Al Nuaimi and Hesham Kamel. A survey of indoor positioning systems and algorithms. In *2011 International Conference on Innovations in Information Technology*. IEEE, apr 2011.

[PFDZ16] Yiran Peng, Wentao Fan, Xin Dong, and Xing Zhang. An iterative weighted KNN (IW-KNN) based indoor localization method in bluetooth low energy (BLE) environment. In *2016 Intl IEEE Conferences on Ubiquitous Intelligence & Computing, Advanced and Trusted Computing, Scalable Computing and Communications, Cloud and Big Data Computing, Internet of People, and Smart World Congress (UIC/ATC/ScalCom/CBDCom/IoP/SmartWorld)*. IEEE, jul 2016.

[QLT16] Jun-Wei Qiu, Chien-Pu Lin, and Yu-Chee Tseng. BLE-based collaborative indoor localization with adaptive multi-lateration and mobile encountering. In *2016 IEEE Wireless Communications and Networking Conference*. IEEE, apr 2016.

[RFR95] John Rieman, Marita Franzke, and David Redmiles. Usability evaluation with the cognitive walkthrough. In *Conference Companion on Human Factors in Computing Systems*, CHI '95, pages 387–388, New York, NY, USA, 1995. ACM.

[RGR+17] Ioana Radoi, Gabriel Gutu, Traian Rebedea, Cristian Neagu, and Marius Popa. Indoor positioning inside an office building using BLE. In *2017 21st International Conference on Control Systems and Computer Science (CSCS)*. IEEE, may 2017.

[Ric15] Mark Richards. *Software Architecture Patterns*. O'Reilly Media, Inc., 2015.

[RMBL15] Meera Radhakrishnan, Archan Misra, Rajesh Krishna Balan, and Youngki Lee. Smartphones and BLE services: Empirical insights. In *2015 IEEE 12th*

International Conference on Mobile Ad Hoc and Sensor Systems. IEEE, oct 2015.

[Sav07] Braiterman J. Savio, N. *Design sketch: The context of mobile interaction. In Mobile HCI*. 2007. pp. 284-286.

[sen] senion. `https://senion.com/`. Accessed: 2018-07-10.

[SHNN12] Matti Siekkinen, Markus Hiienkari, Jukka K. Nurminen, and Johanna Nieminen. How low energy is bluetooth low energy? comparative measurements with ZigBee/802.15.4. In *2012 IEEE Wireless Communications and Networking Conference Workshops (WCNCW)*. IEEE, apr 2012.

[SyIhR16] Gaoyang Shan, Sun young Im, and Byeong hee Roh. Optimal AdvInterval for BLE scanning in different number of BLE devices environment. In *2016 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. IEEE, apr 2016.

[WH08] Oliver Woodman and Robert Harle. Pedestrian localisation for indoor environments. In *Proceedings of the 10th International Conference on Ubiquitous Computing*, UbiComp '08, pages 114–123, New York, NY, USA, 2008. ACM.

[Wik] Radio propagation. `https://en.wikipedia.org/wiki/Radio_propagation#Absorption`. Accessed: 2018-05-18.

[xco] Xcode. `https://developer.apple.com/xcode/`. Accessed: 2018-05-10.

[YZG15] F. Yin, Y. Zhao, and F. Gunnarsson. Proximity report triggering threshold optimization for network-based indoor positioning. In *2015 18th International Conference on Information Fusion (Fusion)*, pages 1061–1069, July 2015.

[ZCJ+17] Han Zou, Zhenghua Chen, Hao Jiang, Lihua Xie, and Costas Spanos. Accurate indoor localization and tracking using mobile phone inertial sensors, WiFi and iBeacon. In *2017 IEEE International Symposium on Inertial Sensors and Systems (INERTIAL)*. IEEE, mar 2017.

[ZXM+14] X. Zhao, Z. Xiao, A. Markham, N. Trigoni, and Y. Ren. Does btle measure up against wifi? a comparison of indoor location performance. In *European Wireless 2014; 20th European Wireless Conference*, pages 1–6, May 2014.

[ZYL+16] Yuan Zhuang, Jun Yang, You Li, Longning Qi, and Naser El-Sheimy. Smartphone-based indoor localization with bluetooth low energy beacons. *Sensors*, 16(5):596, apr 2016.

[ZYW+15] Aiguo Zhang, Ying Yuan, Qunyong Wu, Shunzhi Zhu, and Jian Deng. Wireless localization based on RSSI fingerprint feature vector. *International Journal of Distributed Sensor Networks*, 11(11):528747, jan 2015.