Instituto Superior de Engenharia do Porto

BBeacons

INDOOR NAVIGATION BASED ON BLUETOOTH BEACONS

WOUTER VANDE VELDE & THOMAS VAN RAEMDONCK

1170215 & 1170222

Abstract

The project was a challenge for both of us. We both were new to working with Bluetooth in general and with BLE signals from beacons. Indoor navigation is also something that isn’t around for that long yet, about 3 years. But there are plenty of tutorials and libraries that offer help. We also both weren’t really familiar with Android programming either, luckily Google offers a lot of extended tutorials.

We can say we learned a lot in this project, on multiple fields of informatics.

In the end everything worked out well. We are proud to present you our project about indoor navigation with the help of Bluetooth beacons.

We would also like to thank the mentor of our project, Carlos Ferreira who helped and guided us throughout the project.

Contents

1. Introduction

The project is about geolocation inside a room. Using 4 Bluetooth beacons, a room will be covered by Bluetooth signals. The challenge is to see the location of a smartphone as accurate as possible inside that room

A few examples where this idea can be possible in real life are

- the locations of clients inside a store and their pathing
- the location of babies inside a nursery
- the location of patients with for example Alzheimer's
- …

1.1 Business

The use-case of our project is navigation/location inside a home to see where your smartphone is located within your house. The business the project will be for families who want to track a phone inside their home for example for one of following reasons

- They lost their phone
- They want to see where an elderly is
- …

The project works as following, we add a couple of rooms in the database (e.g. in the app), alongside some information about that room. This is followed by adding beacons to the database too. Now it is possible to select a room and see data gathered from the beacons posted in that specific room. This data will be processed and available in the app or on the website, such as the strength of the signals sent from the beacons or a heatmap. The final product will be that can see where a person has been inside the different rooms of a house.

1.2 Structure

The report is build in 3 main different chapters, these chapters are respectively concepts, hardware and software. In the chapter of concepts, the project describes the communication with the beacons and the trilateration. It is also talked about how these concepts are realized in practice.

The chapter of hardware talks about the hardware, mainly being the Wellcore NRF51822 beacons and how accurately the results are from these beacons.

The third and last chapter talks about the software that is realized within this project. This chapter includes the library to connect with the beacons, the structure of the database and finally the website.

In the end there is also a conclusion which talks about the problems we encountered during the project, the limitations and the future elements that can be added.

2 Concepts and related work

2.1 Indoor navigation using Bluetooth

Bluetooth communicates over small distances on the 2.4 GHz ISM frequency band. It uses 79 channels to transmit data, starting with the first channel at a frequency of 2402 MHz and continuing up to the last one at a frequency of 2480 MHz in 1 MHz increments. [1]

In order to transmit data, Bluetooth devices must first stablish a connection. One single device is capable of connecting to up to 7 devices and communicating with each one of them simultaneously. This is done by using a connection model known as “master-slave”, in which the device that initiates the connection takes the role of master over the other devices. Whenever a master and a slave establish a connection, a bond is created, enabling them to transmit and receive data. [1]

2.1.1 Communication

In the case of this project, a connection is not necessary, because there is no data exchange between the beacons and the smartphone. This means there is only a one-way communication between the two. The beacons will send out packets which are picked up by the smartphone containing all the necessary information. The most important fields found in a single packet are the name, battery percentage, temperature, RSSI, txPower (= transmission power), mac-address and UUID. We will mainly use the RSSI and txPower in our project to calculate distances using these values. A screenshot of one package is shown in figure 1, this screenshot is taken in Android Studio during debugging.

On how the communication is realized, will be discussed further on in the report.

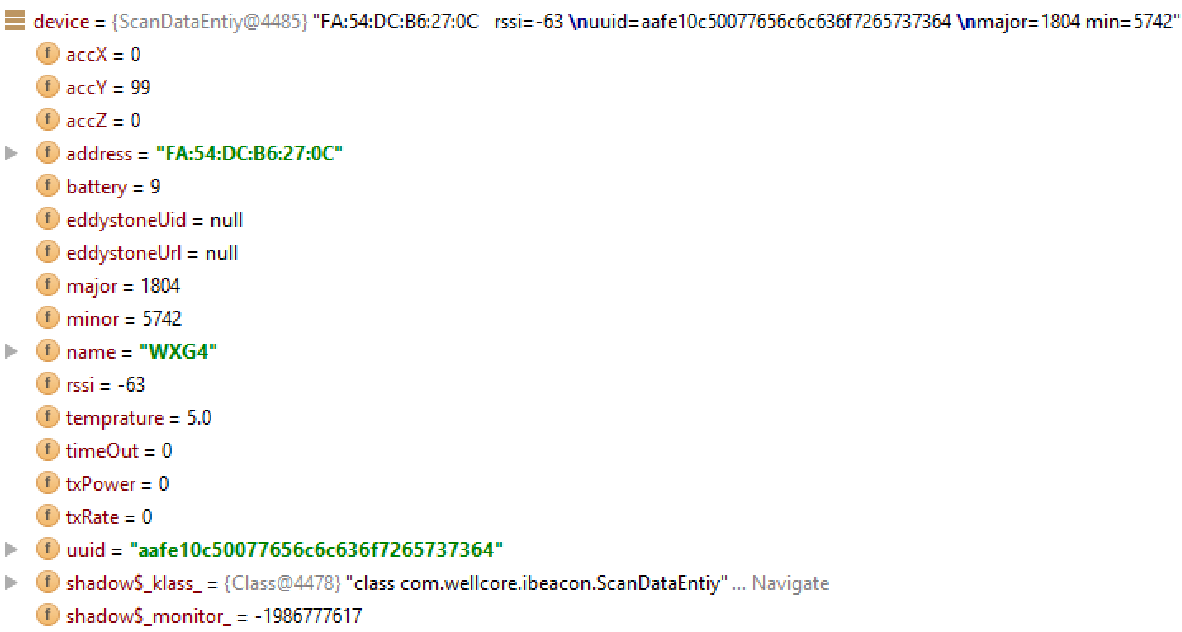

Figure 1: screenshot of the values of a single package from a Bluetooth beacon

The signals that are sent out by the beacons are not regular signals, they are BLE signals. BLE stands for Bluetooth Low Energy. The main feature of BLE is that, as the name suggests, it sends out low energy signals. This means the lifetime of one beacon can go over 2 years, depending on the transmission power and intervals between signals that is chosen on the beacons.

2.2 Trilateration

Trilateration is a method to define a location in between 3 different points with the use of circles. It is not to be confused with triangulation which has the same goal but is reached using the measurement of angles. In this project, trilateration is used to locate a point in between 3 different Bluetooth beacons meaning there are 3 distances given. These distances are calculated trough the strength of each signal, the RSSI, of each beacon. The locating of the smartphone works in 2 steps, first the calculation of the distance, then the actual trilateration. On figure 2, it shows how 3 circles are drawn around each beacon, each with a certain distance. Through these 3 circles, the central location is derived.

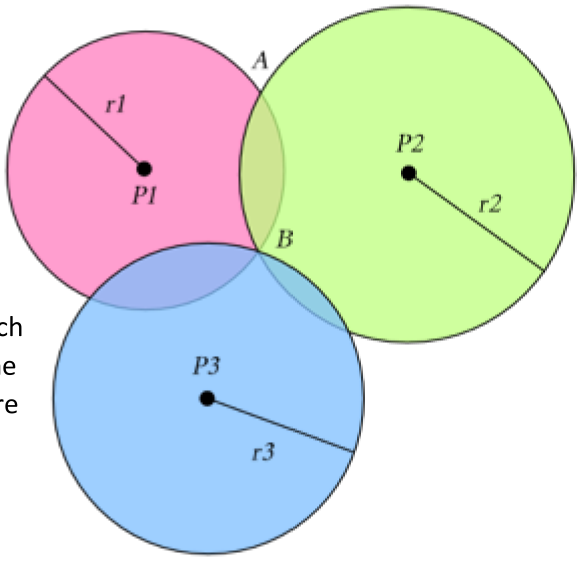

Figure 2: image of trilateration

2.3 Calculating the distance

The distance is calculated by the RSSI values that are received from the beacons. Each RSSI-value ranges from around -30 to -95 being -30 the closest and -95 the furthest.

The formula for the calculation of the distance is:

$$10^{\frac{txPower - RSSI}{20}}$$

In our case, this formula becomes following formula in the program.

```
Math.pow(10d, ((double) txPower - dataArray[i]) / (10 * 2)))
```
[2]

TxPower in the formula is the transmission power, this is a fixed value that is defined on the RSSI signal on exactly one meter. In the case of the Wellcore beacons, the default value is -59 dBm but this can be programmed into the beacons. This means that if the RSSI in *dataArray* is -59dBm, the result of the formula will be one meter. In the *dataArray* are the RSSI signals of the beacons to be found.
If the formula is plotted into a graph, the following graph is returned which is seen in figure 3. The Y-axis shows the distance, the x-axis shows the RSSI values.

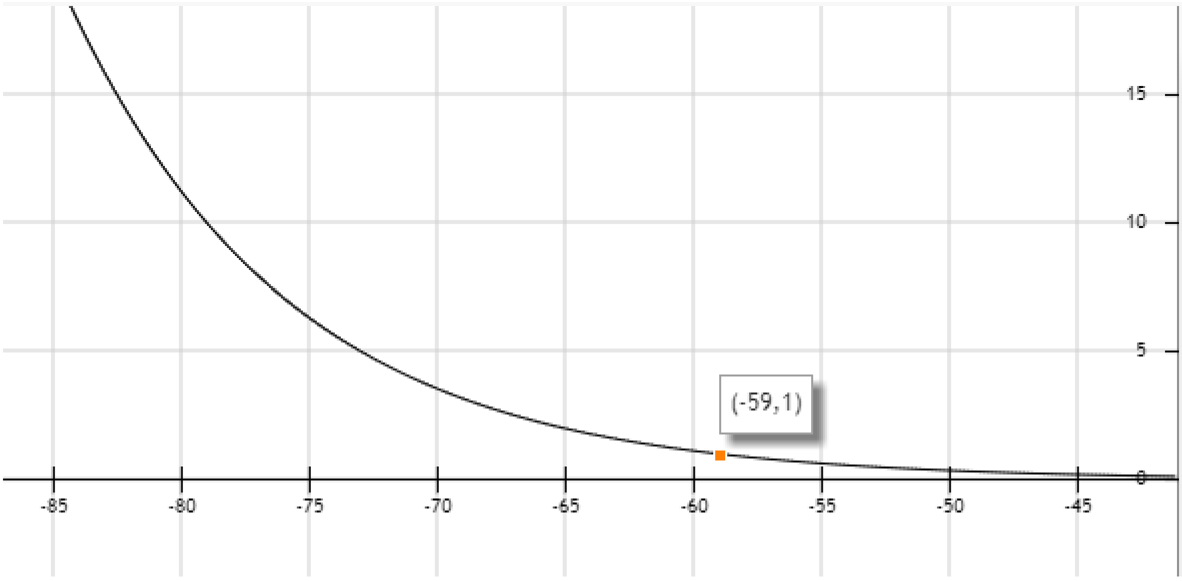

Figure 3: graph of the formula to calculate the distance of one RSSI value, y-axis is meters, x-axis is RSSI signal

The point that is highlighted on the graph shows the point (-59,1). This means the RSSI (x-axis) would be -59dBm where the y-value is 1, representing that the distance would be 1 meter. Because the graph is half a parabola show that how further/weaker the RSSI signal is, the faster the distance increases.

2.4 Trilateration

After the calculation of the distances, the next step is to calculate the point in-between the three distances. This happens trough the following piece of code seen in figure 4:

```
public Point trilateration() {
    float beacon1X = beacon1.x;
    float beacon1Y = beacon1.y;
    float beacon2X = beacon2.x;
    float beacon2Y = beacon2.y;
    float beacon3X = beacon3.x;
    float beacon3Y = beacon3.y;
    float DistanceB1 = (float)values[0];
    float DistanceB2 = (float)values[1];
    float DistanceB3 = (float)values[2];

    double temp1 = (pow(beacon3X, 2.) - pow(beacon2X, 2.) + pow(beacon3Y, 2.) -
            pow(beacon2Y, 2.) + pow(DistanceB2, 2.) - pow(DistanceB3, 2.)) / 2.0;
    double temp2 = (pow(beacon1X, 2.) - pow(beacon2X, 2.) + pow(beacon1Y, 2.) -
            pow(beacon2Y, 2.) + pow(DistanceB2, 2.) - pow(DistanceB1, 2.)) / 2.0;
    double y = ((temp2 * (beacon2X - beacon3X)) - (temp1 * (beacon2X - beacon1X))) /
            (((beacon1Y - beacon2Y) * (beacon2X - beacon3X)) - ((beacon3Y - beacon2Y) * (beacon2X - beacon1X)));
    double x = ((y * (beacon1Y - beacon2Y)) - temp2) / (beacon2X - beacon1X);
    return new Point((int)x,(int)y);
}
```

Figure 4: Function trilateration to calculate the 2D location as a Point-object

The first 6 values are the 2D locations of each beacon, called beaconX and beaconY. DistanceB1 to DistanceB3 are the distances that were calculated before. With these values, the X and Y of the smartphone are calculated which lies in-between the 3 beacons.

2.5 Heatmap

Using the newly achieved location of the central point, it is possible to create a 2D map which can be seen in figure 5. The heatmap consists of a certain number of squares horizontally and vertically, depending on the size of the room. The size of the room is saved in the database and every square is one meter on one meter. Every 2 pixels of the screen consists of 1 cm, so every square has a ratio of 1m:200px.

The heatmap is drawn on a View-class which is drawn over the current activity. The heatmap is updated/redrawn every time the program has received a signal from each beacon, which is on average a couple of seconds. Every time the view is redrawn, the program checks in which square the coordinates of the central point are, to then retract 10 of the RGB values making the color darker.

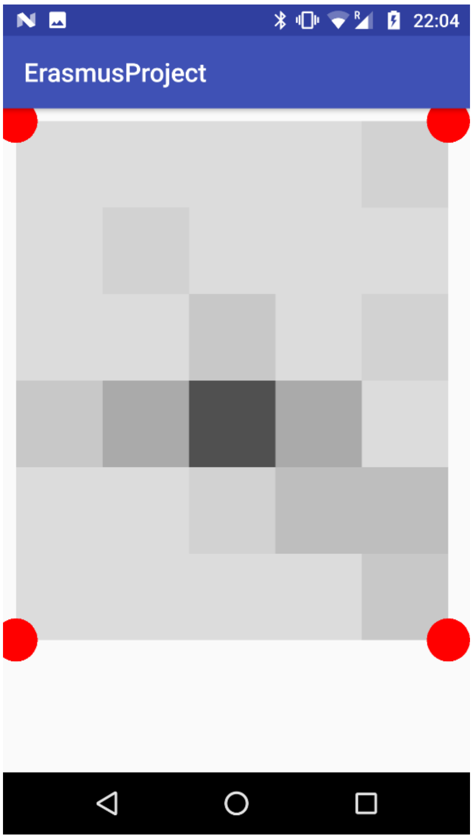

Figure 5: screenshot of the heatmap on the smartphone

2.6 Related work

2.6.1 Project 1

INDOOR NAVIGATION USING BLUETOOTH LOW ENERGY (BLE) BEACONS [3]

The project is created in 2016 by Milan Herrera Vargas. It shows the concept of indoor navigation with the help of Bluetooth beacons. The difference between our project and the described project is that the described project measures the locations in a building while our project only goes in one room. Since walls don't really affect Bluetooth, the project is very similar.

Milan measures the distance using the average RSSI measurements at predefined distances he collected himself seen in following table:

Distance	Average RSSI
1m	-71dBm
2m	-75dBm
3m	-77dBm
4m	-81dBm
5m	-87dBm

Figure 6: Average RSSI measurements at predefined distances

He then used these values to calculate the distance from the beacon to a device, the beacon sends out a signal every 30 seconds. After a test of 300 recorded values comparing the actual and the measured distances, he could conclude the results were fairly accurate. The results were quite good at distances closer than 3m and got less accurate at longer distances. He also notes that if there is no clear line of sight between the beacon and the device, the results might get affected.

Actual distance	Average calculated distance
1m	1.8538m
2m	2.4632m
3m	4.4712m
4m	6.2581m
5m	7.4569m

Figure 7: Comparison between calculated and actual distances

2.6.2 Project 2

[How to do accurate indoor positioning with Bluetooth beacons? [4]]

The project talks about how beacons cannot measure distances but that they are merely like 'lighthouses' that send out signals. That why trilateration is used, the project talks about how you need 3 or more beacons to get an accurate location. The RSSI values of each of them will help you to calculate the distance between the device and the beacons as is shown in following image.

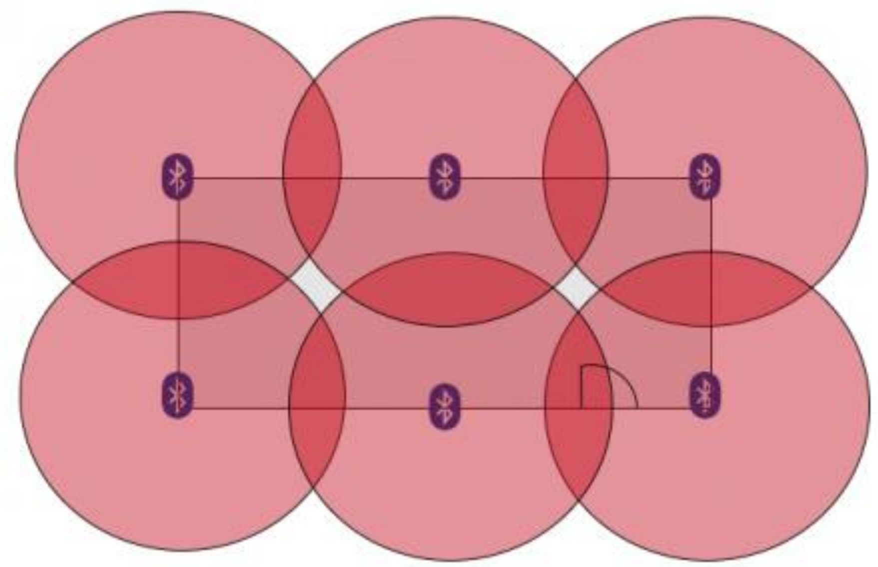

Figure 8: beacon positions in square shaped room

The best location in a room to place the beacons are on a regular and evenly basis in the environment, as shown in the picture above. It is also optimal to hang them on walls, preferably on 2m height. The article also suggests it is important to pay attention to the transmission power and interval and the maximum beacon range, it is best to adjust these values according to the environment the beacons are used in.

2.6.3 Project 3

[Wayfinder: indoors routing guided by Beacons [5]]

This project is less relevant since it talks more about the pathing inside a building with the use of beacons. In our project we will mainly focus on the location and less on the pathing. The app created in the project is called Onyx Beacon's Wayfinder. It shows the most optimal route in a building according to the points of interest principle and it will offer step by step indication for indoors navigation. The user will get an overview of the route and individual instruction on next steps to reach the destination. The distances between the beacons are 25 to 40 meters.

2.6.4 Project 4

[Indoor positioning with beacons and mobile devices [6]]

The method used in the project for estimation position using distance is trilateration.

In a N dimensional world, there are N+1 reference points needed: in a 2D area there are 3 beacons needed to get an accurate position as is shown in following image:

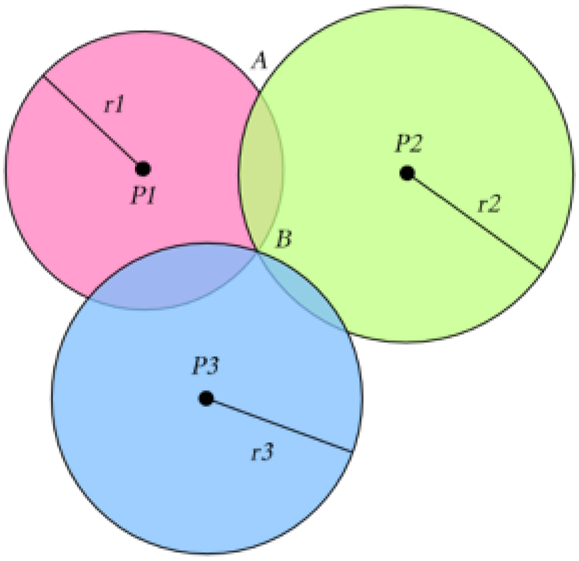

Figure 9: trilateration

The experiment that was conducted in the project shows that is hard to find a satisfactory solution. Therefore, they made use of heatmaps and a large set of data to get a probability area that tells where the device may be found. This is shown in following image:

The room that is used is a big hall of 8m x 18m and the beacons were placed around 2.5m high on the walls.

The article suggests using a signal as strong as possible to get the most accurate signals, it also says to put the beacons as high as possible and the more beacons the better.

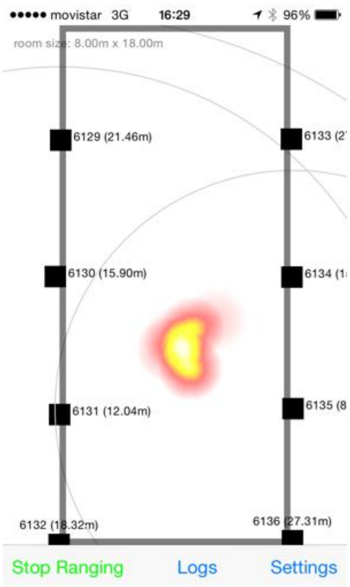

Figure 10: heatmap of a room

3 Hardware

3.1 NRF51822 Wellcore Beacon

The beacons used in this project are “NRF51822 Bluetooth beacons” by the company Wellcore. They were bought on the following link. [7]

The beacons are little CPU’s from ARM® Cortex™ M0 with a build in NRF51822 SoC. The NRF51822 chip is made for Bluetooth low energy and 2.4GHz ultra low-power wireless applications. [8]

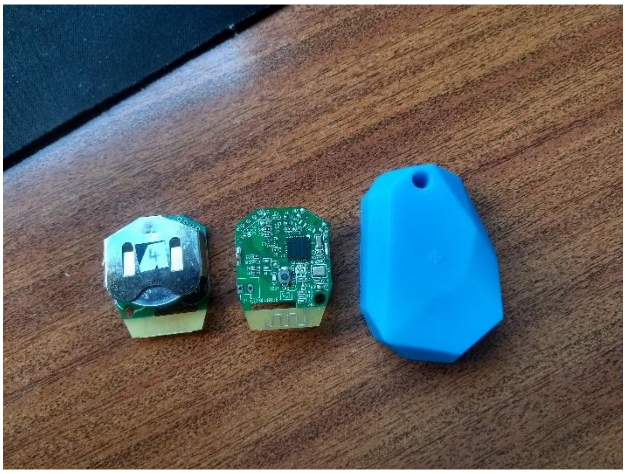

Figure 11: image of the back, the front and a cover of a beacon

On figure 11, there are 3 beacons shown. The first beacon shows the backside, exposing the battery which is a CR2477. The second shows the frontside of the beacon. The third item is the cover that was found around each beacon.

From the results that were gathered, it seemed like the cover had a little impact on the strength of the signal, so they were removed leaving the beacons a little more exposed, just to be sure.

As was mentioned in the preface, the beacons are bought in China. This means the quality of the beacons were not guaranteed beforehand since items bought in Chinese can either be good or bad, in our experience. In this case, the beacons are decent, but not that well suited for indoor location. This is caused by the fact that the RSSI signals can be unstable which leads to some inaccurate results. The main reason for buying the beacons in China was the price. In table 1, you can see a comparison of Estimote beacons and the ones used in the project, Wellcore Beacons.

Table1:

Estimote	**Wellcore**
€16/piece (bought per 3)	€10/piece (bought per 5)
SDK included	SDK included
A lot of documentation provided (e.g. example projects, code examples)	Fewer documentation provided
They promise accurate signals	The signals are found inaccurate

3.2 Accuracy

As an example to show how accurate the beacons really are, the program ran for a little time to record 300 values on different distances. The distances on which the RSSI signals were recorded are 1, 2, 3 and 5 meters. The time between each value is not fixed because the program gets closer values faster and further values slower. The program saves the values as soon as a signal has been received from each of the beacons positioned on different distances.

In figure 12, a graph is shown of the 300 values that were measured by android studio. These values are explained in more detail below, in figures 13 to 15.

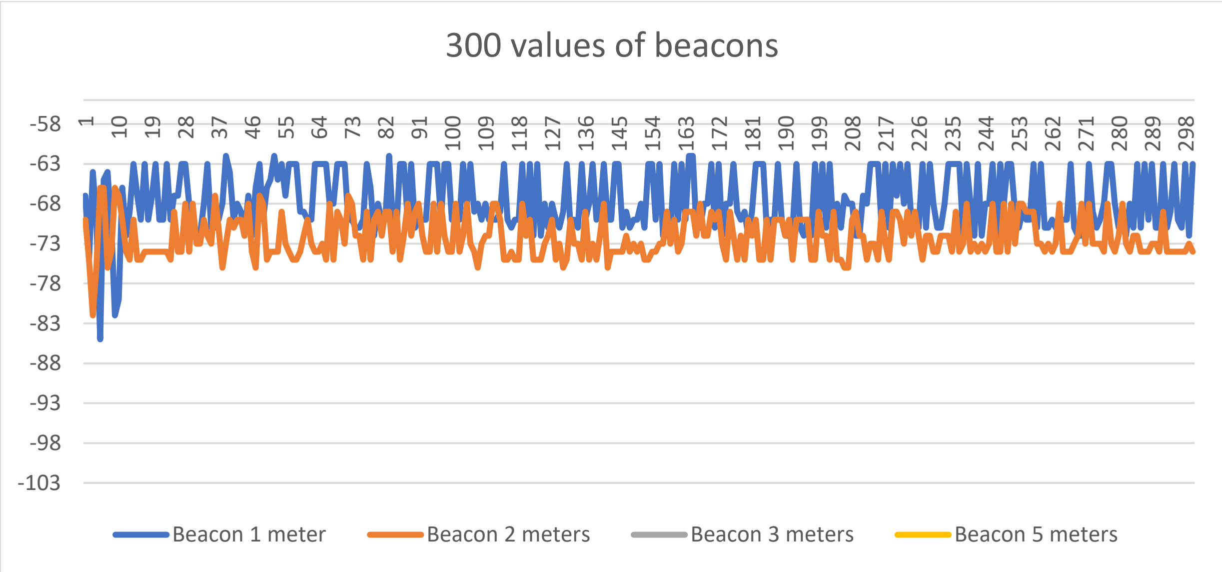

Figure 12: graph of 300 RSSI values on distances respectively 1,2,3 and 5 meters

As seen in the graph on figure 13, it is shown that the signals overlap already on one and two meters. These values should be the most accurate because the closer the beacons are, the stronger thus the more stable the signal should be. Although it is also shown that the average of the values still shows a significant difference.

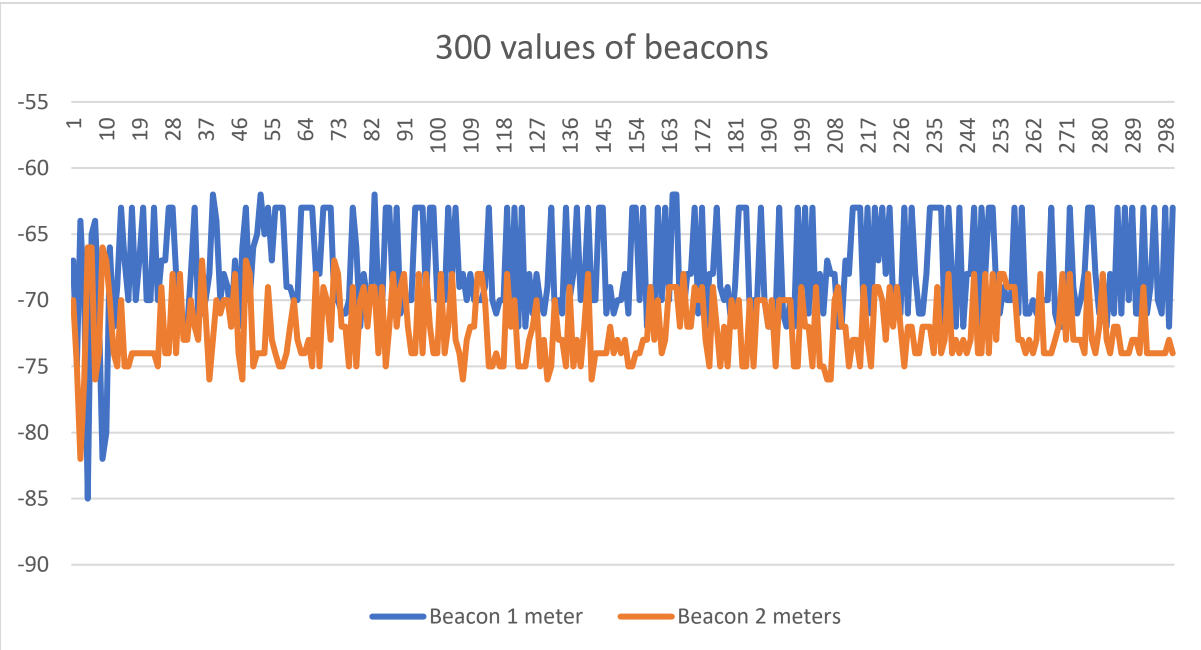

Figure 13: graph of 300 RSSI values on one and two meters

As shown on figure 14, at 3 meters, which is the yellow line, the values start to get really inconsistent. They jump over those of 2 meters and over the ones of 5 meters. There are also weird spikes to be seen around for example 73, 172 and 262.

When we get into 5 meters, the grey line, the values are much worse. The values should be a little higher than the yellow line, which they are sometimes. But the main issue is how unstable they are. It is hard to accurately calculate the distances from these RSSI values.

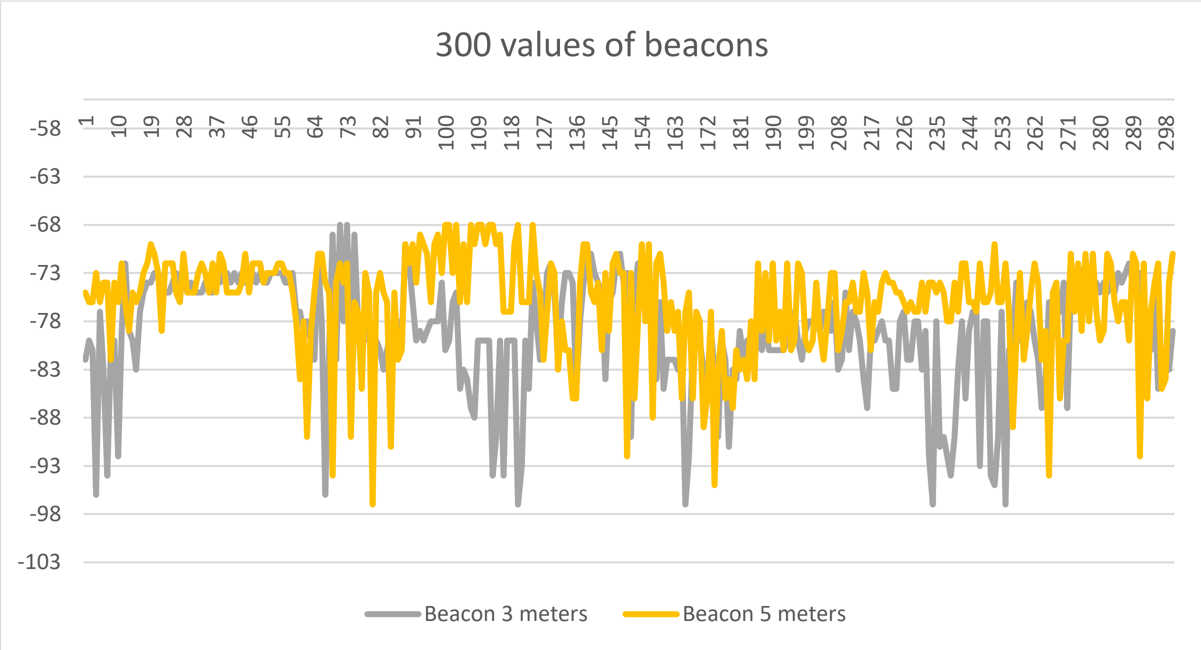

Figure 14: graph of 300 RSSI values on 3 and five meters

In figure 15, a graph is shown where the values 49-103 are enlarged to give an example on how inaccurate they can be. The values of one meter and 5 meters even overlap around 27.

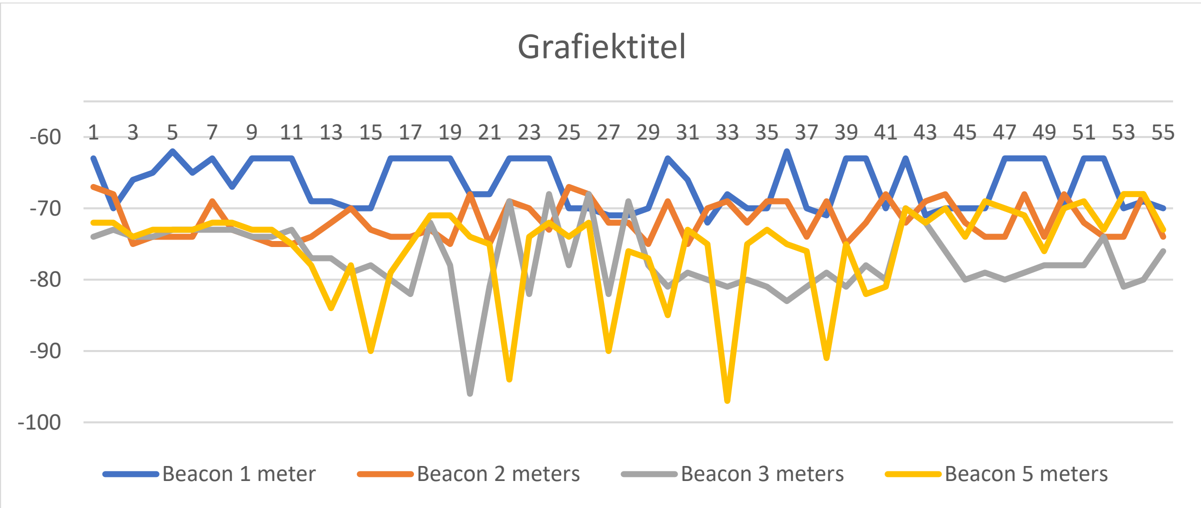

Figure 15: a zoomed in graph of values 49 to 103 on all 4 distances

Several variables can impact the strength of the signal, but the most common issue is something that is in-between the beacon and the smartphone. The most optimal location to put the beacons is then on a high place, preferably on a wall. The signals that were gathered in previous graphs were gathered in an open environment where nothing was in-between the beacons and the smartphone.

4 Software

As said earlier in the report, the communication between the beacons and the smartphones is only in one direction. The beacons broadcast a signal and the smartphones catch that signal using their Bluetooth connection. We use the library AltBeacon to retrieve the data from the beacons. [9] [10]

4.1 Altbeacon

Altbeacon is a library created for the purpose of detecting beacons on Android. Altbeacon allows Android devices to use beacons much like iOS devices do. An app can request to get notifications when one or more beacons appear or disappear. An app can also request to get a ranging update from one or more beacons at a frequency of approximately 1Hz.

The library makes the communication really easy, the first thing to do is to implement the *BeaconConsumer* interface. This is followed by the abstract methods that are added, one of those methods is *didRangeBeaconsInRegion.* This method creates a collection of Beacon objects of all the beacons that are in range. The only thing that is left to do is to iterate trough these beacons, filter out the correct ones, and use the values the objects contain. Once the functions have been ran, the program will be scanning for beacons in the background making use of a backgroundworker, to continuously update the values.

4.2 Database

he database is a MySQL database and is hosted locally on a Wamp-server. It consists of 4 tables being devices, rooms, values and users.

The table Devices contains all the beacons, this contains the name and the room in which the beacon can be found.

In the table Rooms are all the room saved alongside some information about each room such as the width and length and a small description.

In the table values are the values being saved that are taken on a certain moment in a certain room.

Finally, the table Users which contains all the users, the point of the table users is more to check connection to the database from the smartphone. The project doesn't focus on security, so this is not implemented thoroughly in the site and Android application.

On figure 16, the design of the database is shown.

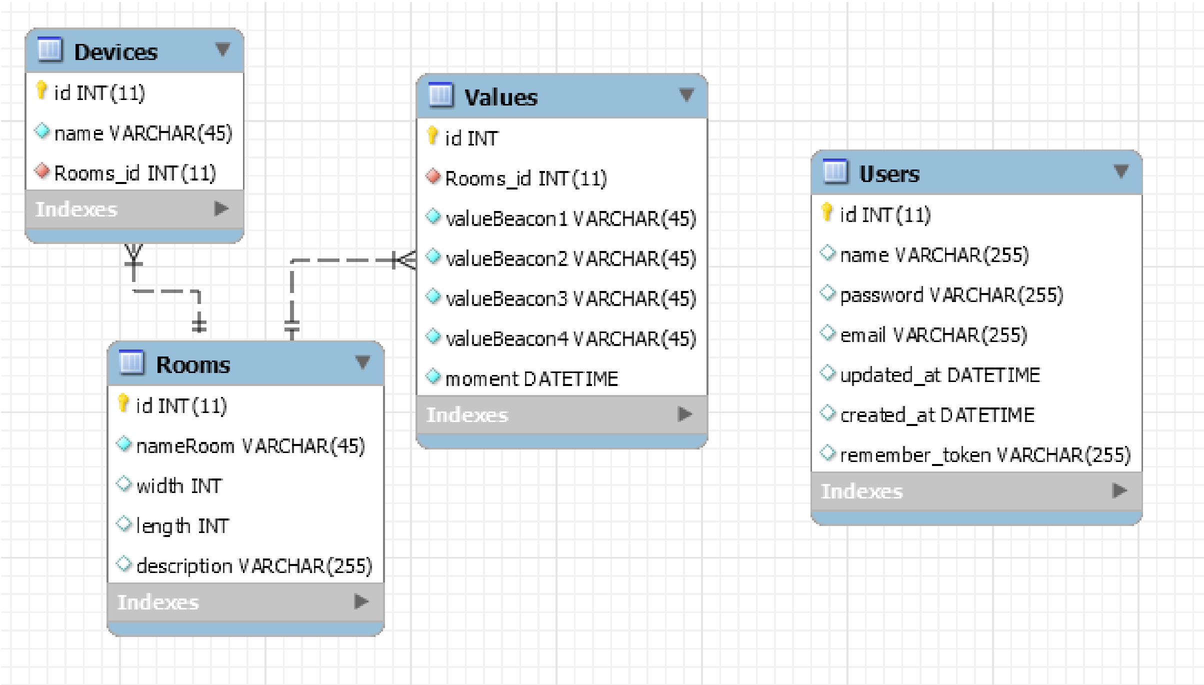

Figure 16: the design of the database

4.2.1 Database connection from Android application

The android application consists of 6 activities, one of these activities is for communicating with the beacons. The other 5 have to make a connection to the database. Because the connection happens on a smartphone to a different ip-address, this process is resource heavy. The program therefore uses background-workers, otherwise the program would stop working for a couple of seconds.

The connection with the database from the android application is done by PHP scripts. The Android application calls a different script for each query it wants to use. So the application doesn't send the query's itself. It uses a HTTP connection. The application writes values in the query to the server (localhost). [11]

The Android app sends the parameters to the server. Here the PHP scripts receive the parameters and sets them in the query. When the query is executed the PHP script returns the outcome to the android application.

4.2.2 Database connection from the website

The connection with the database is done by the Laravel package called Eloquent ORM, this is talked about in 4.3.1.

4.3 Website

The website is created in the PHP web framework Laravel. Laravel is intended for the development of web applications following the model-view-controller pattern. Following features are some of the integrated ones in the website: Restful controllers, Blade, Form request and Eloquent ORM.

The website contains of 3 pages. The homepage, which shows an overview of all the rooms in the house. Then there is also a page which shows all the values of the beacons of the rooms in the house. When you select one of the rooms from the list of rooms, you are directed to the third page. This is a detailed page of the rooms with all the data such as width and length and the beacons in the room accompanied with the data of that room.

4.3.1 Eloquent ORM

Eloquent ORM is a package provided by Laravel and helps with the connection to the database. ORM stands for Object Relational Mapper, this means that objects are linked to each other just as tables in a database are linked to each other. To achieve this, there is a PHP-file made for each table in the database. These files are then linked to each other by methods such as oneToMany or belongsTo. An example of the table rooms is shown in figure 17.

```php
class Rooms extends Model
{
    protected $table = 'rooms';
    protected $fillable = ['nameRoom'];
    public function devices()
    {
        return $this->hasMany('App\Devices', 'rooms_id');
    }
    public function values()
    {
        return $this->hasMany('App\Values');
    }
}
```

Figure 17: Rooms table programmed as an object class

After that the classes are correctly linked to each other, the relations can be queried to get the data out easily. In figure 18, the first line shows a query that gets out all the rooms. Thanks to Eloquent ORM, it is possible to just reference the table devices to the resultset from the first query to create a simple inner join. This means *$room* contains the room that has the id *$id*. *$devices* contains all the beacons that can be found inside that room.

```php
public function detail($id)
{
    $room = Rooms::findOrFail($id);
    $devices = $room->devices;
    return view('roomDetail', array('room' => $room, 'beacons' => $devices));
}
```

Figure 18: query to get out a room and its containing beacons out of the database

5 Diagrams

The diagrams are created in Visual Paradigm. In this project, there is a deployment diagram. This deployment diagram was created to remove the doubts on how we would design the project, mainly how we would send the data from the phone to the server and where the processing of the data would be done.

5.1 Deployment diagram

This diagram shows the hardware and the software of the whole website as can be seen in figure 19. There are 3 distinctive parts in this diagram. The left part shows the website and server, the middle part shows the android application on the smartphone and the right part shows the beacons.

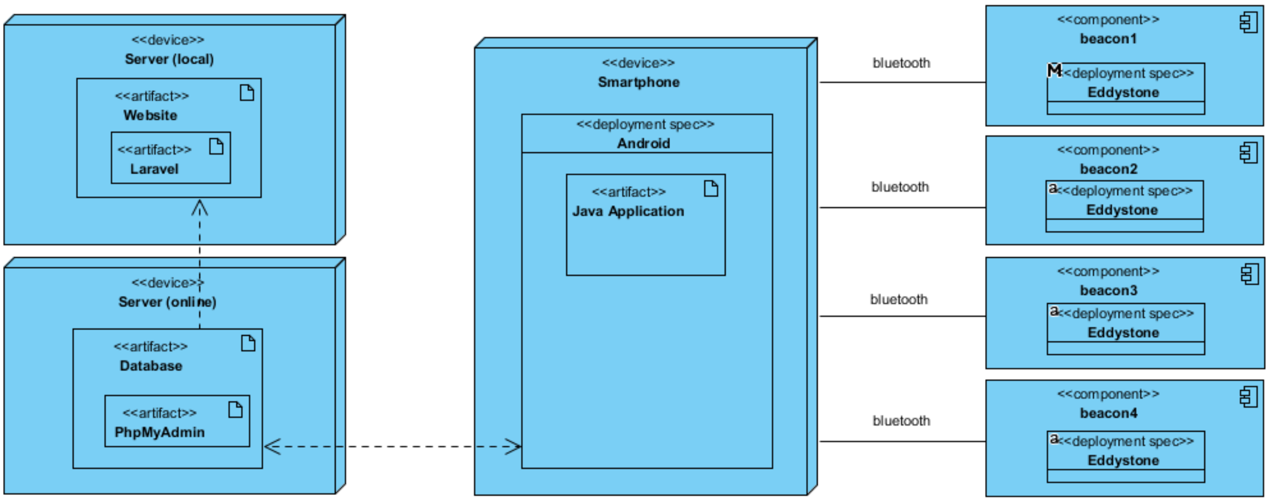

Figure 19: deployment diagram

6 Conclusion

6.1 Problems

The biggest problem was buying the beacons. We chose the buy our beacons in China since the quality should not be that much worse and the price is more than half compared to beacons from companies like Estimote or Kontakt. This also meant the delivery time of the beacons was around 20 to 40 days.

The first time we ordered beacons was around the 10th of October. These beacons arrived in the airport in Porto on the 17th of October but were not cleared by the customs. They requested the passport of the buyer, but we both only have an identity card. We couldn’t get them cleared so the beacons where eventually sent back on the 21th of October.

This meant we had to order new beacons, which we did on the 25th of October. These ones we finally got at the end of November. This means we had lost almost 2 months’ time before we could start working with the beacons.

When the second set of beacons was on the way, we also ordered a new set from a different manufacturer in case the other beacons were sent back again. These beacons were also “nRF51822 Bluetooth beacons” but from the company Skylab. We ordered 2 sets of 3 beacons because we needed at least 4 beacons. When these ones arrived, we noticed they only sent one set of 3 beacons which meant we couldn’t fill a room with them since we wanted 4 beacons. When we sent them, they said it was an accident and we could get 6 new ones for the price of 3…

6.2 Future of the project

In the future, it will be possible to add houses to the project so that a user can not see data from houses other than the ones he is allowed in. Save authentication would be required also then.

It would also be possible to see a heatmap on the website and more additional data.

Bibliography

[1] Anonymous, “Bluetooth low energy beacon,” Wikipedia, 12 January 2018. [Online]. Available: https://en.wikipedia.org/wiki/Bluetooth_low_energy_beacon.

[2] P. Uttarwar, “How do I calculate distance (in meters/km/yards) from rssi values (in dBm) of BLE in android?,” 4 June 2016. [Online]. Available: https://www.quora.com/How-do-I-calculate-distance-in-meters-km-yards-from-rssi-values-in-dBm-of-BLE-in-android/answer/Pravin-Uttarwar.

[3] V. Milan, “INDOOR NAVIGATION USING,” Theseus, 2016. [Online]. Available: http://www.theseus.fi/bitstream/handle/10024/105619/Herrera%20Vargas_Milan.pdf?sequence=1&isAllowed=y.

[4] Kalle, “How to do accurate indoor positioning with Bluetooth beacons?,” Proximi, 28 July 2017. [Online]. Available: https://proximi.io/accurate-indoor-positioning-bluetooth-beacons/.

[5] D. Supeala, “Wayfinder: indoors routing guided by Beacons,” Onyx, 25 november 2015. [Online]. Available: https://www.onyxbeacon.com/wayfinder-indoors-routing-guided-by-beacons/.

[6] L. Floreani, “Indoor positioning with beacons and mobile devices,” citrusbytebits, 24 February 2015. [Online]. Available: http://bits.citrusbyte.com/indoor-positioning-with-beacons/.

[7] Wellcore, “5pcs/lot NRF51822 ibeacon Module BLE 4.0 bluetooth beacon eddystone beacons,” Aliexpress, [Online]. Available: https://www.aliexpress.com/item/5pcs-lot-NRF51822-ibeacon-Module-BLE-4-0-bluetooth-beacon-eddystone-beacons/32795069285.html.

[8] nordicsemi, “nRF51822,” nordicsemi, [Online]. Available: https://www.nordicsemi.com/eng/Products/Bluetooth-low-energy/nRF51822. [Accessed 17 January 2018].

[9] D. G. Young, “Android Beacon Library,” Altbeacons, 2015. [Online]. Available: http://altbeacon.github.io/android-beacon-library/.

[10] D. G. Young, “How to detect IBeacon in android without using any library,” Stackoverflow, 8 May 2014. [Online]. Available: https://stackoverflow.com/questions/23546371/how-to-detect-ibeacon-in-android-without-using-any-library.

[11] Stackoverflow, “How do I connect to a MySQL Database in Python?,” Stackoverflow, 16 December 2008. [Online]. Available: https://stackoverflow.com/questions/372885/how-do-i-connect-to-a-mysql-database-in-python.

[12] Infsoft, “Indoor Navigation,” Infsoft, [Online]. Available: https://www.infsoft.com/solutions/indoor-navigation. [Accessed 25 January 2018].

[13] M. b. editor, “Path Guide: A New Approach to Indoor Navigation,” Microsoft, 14 July 2017. [Online]. Available: https://www.microsoft.com/en-us/research/blog/path-guide-new-approach-indoor-navigation/.

[14] Anonymous, “Indoor positioning system,” Wikipedia, 20 January 2018. [Online]. Available: https://en.wikipedia.org/wiki/Indoor_positioning_system#Bluetooth.

[15] Estimote, “Indoor navigation,” Estimote, [Online]. Available: https://estimote.com/. [Accessed 2 October 2018].