BACHELOR'S THESIS 2022

Possible uses for body-worn cameras with WiFi/BT signal strength indoor positioning

Jonas Hallböök, Sebastian Forslund

ISSN 1651-2197
LU-CS/HBG-EX: 2022-06

DEPARTMENT OF COMPUTER SCIENCE
LTH | LUND UNIVERSITY

Possible uses for body-worn cameras with WiFi/BT signal strength indoor positioning

A use case-based analysis of an indoor positioning system

LUND UNIVERSITY
Campus Helsingborg

LTH School of Engineering at Campus Helsingborg
Department of Computer Science

Bachelor thesis:
Jonas Hallböök
Sebastian Forslund

LTH School of Engineering
Lund University
Box 882
SE-251 08 Helsingborg
Sweden

LTH Ingenjörshögskolan vid Campus Helsingborg
Lunds universitet
Box 882
251 08 Helsingborg

Printed in Sweden
Media-Tryck
Biblioteksdirektionen
Lunds universitet
Lund 2022

This page is intentionally left blank

Abstract

Positioning of devices has largely been associated with the use of GPS satellites and signals. While this works in an outdoor environment, the technique does not fare so well indoors. Therefore, to position a device indoors, other methods must be used. This thesis focuses on using WiFi- and Bluetooth devices to position a device in an indoor environment. There have been tests measuring the accuracy when using this method of positioning. Both WiFi- and Bluetooth devices are in abundance in most modern buildings, therefore the necessary infrastructure needed for this kind of indoor positioning is already in place. It is also investigated how Bluetooth beacons could improve the accuracy of the positioning system.

Three use cases that are linked to security guards wearing Body worn cameras (BWC) are studied. When developing use cases, information from the industry is needed. Therefore, information about current workflows was gathered by performing an interview with a security guard. The information about the workflows of security guards has been used to find possible use cases for the indoor positioning system. To further test the positioning system, features of the prototype were developed according to each use case. The features implemented are the following:

- Replacing bar codes that are currently used to report visited locations for security guards
- Performing an action when the BWC is in a specific area.
- Using recorded positioning data to search for recordings in an area.

The testing of these features show that the positioning system can position a device indoors with an average accuracy of under 10 meters. However, these features are critical to a security guards work, thus requiring a higher accuracy to be fully reliable.

Keywords: Indoor positioning, WiFi, Bluetooth, Security Guards, Geofencing.

Sammanfattning

Positionering av elektroniska enheter har till stor del förknippats med användningen av GPS-signaler. Även om detta fungerar i en utomhusmiljö, fungerar tekniken inte så bra inomhus. Därför måste andra metoder användas för att lokalisera en enhet inomhus. Detta examensarbete fokuserar på att använda WiFi- och Bluetooth-enheter i närheten för att lokalisera en enhet i en inomhusmiljö. Det har gjorts tester för att mäta noggrannheten vid användning av denna positioneringsmetod. Både WiFi- och Bluetooth-enheter finns i överflöde i de flesta moderna byggnader, så finns den nödvändiga infrastrukturen som behövs för denna typ av inomhuspositionering redan på plats. Det undersöks också hur Bluetooth-beacons skulle kunna förbättra positioneringssystemets noggrannhet.

Tre användningsfall som är kopplade till säkerhetsvakter som bär kroppsburna kameror (BWC) studeras. När man utvecklar användningsfall behövs information från den aktuella branschen. Därför samlades information om aktuella arbetsflöden in genom att göra en intervju med en ordningsvakt. Informationen om ordningsvakternas arbetsflöden har använts för att hitta möjliga användningsfall för positioneringssystemet. För att ytterligare testa positioneringssystemet utvecklades funktionerna i prototypen i enlighet med varje användningsfall. Funktionerna som implementeras är följande:

- Ersätter streckkoder som för närvarande används för att rapportera besökta platser för ordningsvakter
- Utföra en åtgärd när BWC är i ett specifikt område.
- Använda inspelad positionsdata för att söka efter inspelningar inom ett visst område.

Testningen av dessa funktioner visar att positioneringssystemet kan placera en enhet inomhus med en genomsnittlig noggrannhet på under 10 meter. Dessa funktioner är dock avgörande för en säkerhetsvakts arbete, vilket kräver en högre noggrannhet för att vara helt tillförlitlig.

Nyckelord: Inomhus positionering, WiFi, Bluetooth, Säkerhetsvakter, Geo staket.

Foreword

This thesis was made in collaboration with Axis in Lund. With the help and guidance from the people working at Axis this thesis project was made possible. We especially want to thank our supervisors at Axis; Peter Eneroth and Peter Abdelmassih Waller for their coaching and guidance. Additionally, we would like to thank Jens Olsson for helping us with the confusing world of Linux and D-Bus management when in dire need.

We also want to thank the people at Combain for letting us test their web services, answering our questions, and showing hospitality towards us by inviting us to their office for a meeting.

Another thanks to Anne-Li Rupprecht and Andreas Heder at Avarn Security, who let us do an interview with them. This interview provided crucial information, which the thesis to a large extent is based upon.

We would like to thank our supervisor Sven Gestegård Robertz at the department of computer science, LTH for giving frequent guidance surrounding the creation of the report, as well as providing helpful thoughts about various parts of the thesis.

Terminology

- Access points - A Bluetooth or WiFi device capable of wireless communication.
- AP - An abbreviation for “Access point”, either Bluetooth or WiFi ones.
- Bluetooth Beacon - An always turned on Bluetooth device made with the purpose of positioning using Bluetooth.
- BWC - An abbreviation for “Body Worn Camera”, which is the camera deviation detection was tested on.
- Geofencing - A technique used to divide a particular area into different zones and determine which zone a particular entity resides in.
- Ground truth - A piece of data that is known to be correct. In this thesis, this piece of data always consists of a coordinate associated with a location.
- OSM - An abbreviation for “Open Street Map”, which is a geographical database with an open licence. https://www.openstreetmap.org/
- RSSI - An abbreviation for ”Received Signal Strength Indicator”, which is an indicator of the signal strength from a signal. In the context of this thesis, it refers to Bluetooth and WiFi signals.
- SSID - An abbreviation for ”Service set identifier”, which is a name for a WiFi or Bluetooth device. An SSID can often be manually changed, but typically has a default value.
- Use case - A way in which an indoor positioning system could be used.
- Use case feature - A program designed to fulfill the requirements put on it by a use case. Each use case feature to be implemented is mentioned in section 3.3.3.
- Tkinter - A python GUI library used to visualise position data during the thesis. https://docs.python.org/3/library/tkinter.html
- Combain - An company providing indoor positioning based on technologies such as WiFi and Bluetooth through a web API. https://combain.com/
- Combain indoor survey - An app used to create a building model that is used to improve indoor positioning using Combain API. https://combain.com/use-cases/indoor-positioning/

Contents

1 Introduction

This section is an introduction to the thesis. This includes presenting the purpose of the thesis, the background to how it came to be, as well as the goal and the limitations of the thesis. It ends with the disposition of this thesis.

1.1 Background

Currently, most positioning systems rely on the use of GPS, but there are conditions where the GPS signal is not strong enough to accurately determine a position, such as in an indoor environment, or even more so underground. For some actors, it would therefore be beneficial to have an alternative positioning technology that works in those conditions. This thesis was done in collaboration with a company called Axis Communications that develops different network-based security cameras. During recent times, they have been manufacturing a Body-worn Camera (BWC) commonly used by security guards. Currently it is mainly used to gather evidence for potential legal disputes. Axis Communications has faced the very issue previously described; When used indoors or underground, the GPS accuracy diminishes. For this reason, they have been looking for an alternative way to perform indoor positioning. One of the interesting alternatives they found was to use an external API called Combain, which is a web service that uses MAC-addresses and signal strengths (RSSI) of nearby access points to calculate a position. Combain supports the use of a range of different wireless technologies, though this thesis is limited to using WiFi and Bluetooth. This API has been used throughout the thesis to achieve indoor positioning.

1.2 Goal and motivation

The aim of this thesis is to evaluate whether adding features that use indoor positioning to a BWC could lead to improvements for the safety or efficiency of security guards patrolling areas, which could in turn lead to safer public spaces. The result of the thesis could be used as part of an evaluation on whether indoor positioning is viable for use in other areas of work as well, since testing on the accuracy of the positioning system was also done.

1.3 Problem

In order to more accurately describe exactly what the thesis aims to achieve, a number of questions to be answered have been formulated;

1. What are the use cases for an indoor positioning system on security guards?
2. What accuracy and precision is required to satisfy these use cases?
3. What accuracy and precision is achievable?
4. How can a geographical ground truth be established and with what accuracy?
5. What information about access points is useful for indoor positioning?
6. How does the amount of access points impact the accuracy and precision?
7. Is it possible to detect which side of a wall a device is located?
8. Is WiFi or Bluetooth better in a non-prepared (environment)?
9. How can Bluetooth beacons be used to improve indoor positioning?
10. Does the indoor positioning system implemented in this thesis perform well enough for the chosen use cases?

Answering these questions has been the basis of this thesis and the answers was the foundation of what the conclusion is based on.

1.4 Limitations

Because this thesis has been created during a limited time frame, several limitations on what has been worked on in the thesis are in place. There was no speculation about how the Combain API performs the positioning; Even if such speculations could improve the accuracy of the positioning, it was predicted to be too time consuming.

When it comes to the legal parts of the indoor positioning use cases, they have not been discussed to any depth in this thesis. This is due to the authors having lack of expertise in the area. There is only a small discussion about the possible legal and ethical implications in the *Ethics* section (Section 7).

Even though the use cases were based on an interview with specific security guards, the testing of the use case features were not done in collaboration with said guards. Instead, the viability of each use case is based on a number of tests, which were designed to simulate the conditions described to the interviewers in the security guard interview.

One could imagine there are optimal ways to place Bluetooth beacons for indoor positioning. However, in this thesis, there has been no analysis about how Bluetooth beacon placements might affect the accuracy.

The way the devices scans for nearby access points are not taken into consideration any more than that they work and give the desired result. The underlying technology that performs the scans has not been investigated, only the documentation of the expected outputs from these commands was explored.

1.5 Disposition

The purpose of the *Introduction* (Section 1) is to give an introduction to the thesis work. It also formulates concrete questions, and the answers to these questions is the final result of the thesis. The *Technical background* (Section 2) provides the technical information needed to understand the rest of the thesis. *Method* (Section 3) describes the workflow used in the thesis, followed by how the use cases for indoor positioning was elicited. It also describes the different prototypes and use case features that were developed, along with some of the techniques used in them. Finally, the tests that were performed are described. The result of these tests is presented in the *Result* (Section 4). A discussion about these results section then be held in the *Analysis* (Section 5), which uncover factors that during the testing were either unknown or dismissed. These factors and other possible improvements to the positioning system is placed in the *Future work* (Section 6). In the *Ethics* (Section 7), some of the ethical questions regarding having employees use a positioning system is discussed. Finally, the *Conclusion* (Section 8) presents the answers to the questions previously formulated in Subsection 1.3 and also includes the authors thoughts about whether or not the use case features would be an improvement to the already in use systems.

2 Technical background

This section describes the technologies that have been used during the thesis. The goal of the section is to give enough introductory information about used technologies for the reader to be able to understand the rest of the thesis. First, a background to the Combain API is given. Secondly, how the access point infrastructure impacts indoor positioning using Combain is discussed. Finally, different information regarding scanning for Bluetooth and WiFi access points (AP) and their respective RSSI:s (Received Signal Strength Indicator) is presented.

2.1 Combain API

Combain positioning API is a web service that can position a device in an indoor environment [14]. It is a web-based API that calculates the position using MAC-addresses of WiFi access points and Bluetooth devices together with the RSSI from these devices. A user sends a HTTP/S-request with a list of MAC-addresses and RSSI values to the API and receives a response with information about the position. The response includes the coordinates and accuracy along with other information depending on the building.

When using Combain API, a specific model of a building is applied, which is a database containing information about the WiFi and Bluetooth access points in that building. This database can be populated using a mobile app called "Combain Indoor Survey" that performs a survey of the building [15]. The survey is done by adding a reference point on the user's current location which then scans the nearby AP:s. The reference point is manually placed on a map in the app. After a survey is done, the database is populated with the information that is later used to perform indoor positioning in that building. A general rule is that the accuracy of the response from the API depends on the size and quality of the database and adding more reference points creates a better model.

The requests sent to the Combain API are independent from each other, meaning that a previous request does not affect other request being sent. There is an option to send the state with the request, but this is not utilized in this thesis.

2.2 Infrastructure of access points

One of the main parameters that govern the success of indoor positioning is the available infrastructure in a certain environment. To achieve a desired result, there needs to be a sufficient number of AP:s, which should be positioned away from each other to improve positioning. Increasing the number of devices generally corresponds to better accuracy. The best result with indoor positioning is achieved with a carefully planned infrastructure, but this is often not the case in a facility.

Because there is often already an implemented infrastructure in a facility, there is no need to add devices to make the indoor positioning work. This makes it a portable method and can be used by many parties. However, to achieve a result with greater accuracy, one could modify the infrastructure to one's needs. When it comes to the specific area used for testing in this thesis, it has no infrastructure intended for indoor positioning, but the API can still calculate a position as presented in the *Results*. To see the impact of a modified infrastructure, some additional tests has been done by manually placed Bluetooth beacons.

2.3 Body worn camera

The body worn camera (BWC) [18] used in this thesis is a body worn camera that runs Linux. It has a network card that can communicate wirelessly which is used to send requests to the Combain API to perform indoor positioning in real time. The network card is also used to scan the area for nearby AP:s. These body worn cameras can be either in docked mode, being connected to a docking station, or operation mode being used wirelessly. In this thesis, the camera functionality was not used, the device has only been used to perform indoor positioning.

2.4 Bluetooth scanning

To perform Bluetooth scanning, the devices in this thesis used Linux terminal commands to capture the data from nearby access points. On a Linux machine, a Bluetooth scan was started by using the *Bluetoothctl* tool. Once started, the data regarding the scanned devices was captured using *Btmon* tool. This data included, but was no limited to the MAC address, address type, and the RSSI for each nearby Bluetooth access point.

Since Bluetooth is used in many kinds of devices, it is necessary to sort out those that can be used for indoor positioning. In a public building, a lot of the Bluetooth devices present are cellphones, headsets, and other portable devices. These cannot be used for positioning since the devices used have to be stationary for it to work. There are a couple of methods to distinguish between different types of Bluetooth devices, where certain attributes can be characterized.

To begin with, the way MAC-addresses are assigned to Bluetooth devices differ, and can be split up in two major address types: *public* and *random* addresses. A public Bluetooth MAC-address is an address that must be registered with the IEEE and typically does not change [6]. Therefore, one can assume that the device can be uniquely identified with its MAC-address, which is critical for positioning. But not all devices have public addresses, most of the Bluetooth devices used to today use random addresses instead. One alternative to the *public* type, is to use *random static* addresses. These are either always the same or assigned during bootup, but never changes during runtime. A device of this type can be identified with its MAC-address, but it is not guaranteed. Lastly, there are *private* addresses, which can be used together with *public* or *static* ones. *Private* addresses can change during runtime and are used to protect the privacy of the device specifically to prevent tracking. In this thesis, the prototypes filtered out all *random private* addresses.

2.5 WiFi scanning

Like Bluetooth, the WiFi scanning also utilized Linux terminal. Here the *iw* tool was used to scan for nearby access points using the WLAN interface present on the device. This tool outputs the MAC-addresses and the RSSI among other data. As for WiFi access points not suited for indoor positioning, there could be mobile access points present in a building. These include for the most part mobile hotspots which uses tethering to relay a wireless signal.

The first three octets of a MAC-address are called the Organizationally Unique Identifier (OUI). The OUI part is purchased from IEEE and can be used to identify the vendor of the access point [9]. To filter the MAC-addresses, a method was used that compares the OUI part of MAC-address captured to a list of common MAC-address [8]. Since most cellphones change their MAC-address often and into random new ones, these are not likely to be a MAC-address associated with OUI:s purchased from IEEE. This list was obtained from Wireshark OUI Lookup tool [7].

3 Method

This sections describes how the thesis was conducted and the method used to achieve the results. It begins with presenting the general work flow during the thesis. After that, the way the use cases were elicited and decided upon is described. The Prototype section describes how the prototypes were implemented, what devices were used and what programming languages they were written in. It also explains what techniques were used to the filter the MAC-addresses used for positioning. In the Visualizing Data section, the GUI-application that was used to visually evaluate the results is presented. The method used when testing the positioning system is split in two sections, one describing the positioning testing and one describing how the testing of the use cases were done.

3.1 Method summary

A prototype was developed on the BWC which performs indoor positioning using WiFi and Bluetooth. To obtain a more reliable positioning data, a web service named Combain was used, which calculates a position given data about the nearby access points. The prototype was tested against a number of real-world use cases that relate to how security guards work. To validate the use cases, an interview with a security guard company has been done. Each use case resulted in a feature, which is a program created to fulfill the use case. The results from testing the prototype according to the use cases has been the basis of an analysis about whether indoor positioning seems viable to use for these applications. In the analysis, a discussion about what infrastructure and technology is necessary to get the accuracy required for the use cases is held. The prototype was first used to evaluate which ways of capturing data results in the highest accuracy. Secondly it was used to determine whether this positioning method could be used in different features, designed for use cases formed by security guards. Many indoor positioning systems uses a custom infrastructure of access points to enhance the accuracy and precision. Since the prototype is aimed to be used as a tool for security guard companies, one assumption is that the environment in which they work is not prepared with custom placed access points.

3.2 Work flow

To visualize the workflow in which we worked during the thesis, a flowchart was made (Figure 1). A square being located in the "Documentation" area indicates that documentation for the final report occured. In the squares outside of the "Documentation", documentation still occured but were merely for personal use for the thesis authors, to remember things like technical details and encountered problems.

The first part of the thesis involved planning the work ahead, getting to know how the BWC worked and how indoor positioning using Combain could be done. Some initial testing were done in Kemicentrum in Lund to get the first results. Meetings were held with the supervisors on both the company and the university each week to update on the work and discuss new ideas. This part of the work is represented as the three uppermost squares in Figure 1.

After the initial testing, there were discussions with domain-experts at the company about what use cases were suitable for the thesis. During this time, the development of the prototype and further elicitation was done in parallel. The discussions about the use cases led to the conclusions that an interview with a security guard was needed to fully shape a use case. When the interview was finished and the results from it analysed, three use cases were decided upon to use, as is further described in Subsection 3.3.3. This decision concluded the elicitation part of the thesis and the it was time to implement the features based on the use cases. A visualisation of how this part of the work was performed is seen in the "Use cases" area in Figure 1.

The development and testing of the features was done iteratively. A discussion was held about how the feature would be implemented was followed by implementing it and then testing the idea at Kemicentrum. During the analysis of the result, further ideas on how to either improve the prototype or how it was to be tested was brought up. Since this was done in an iterative process (See "Use case features" in Figure 1), there was no particular order in which the use cases features were implemented.

Lastly, during one of meetings at the company, the necessity to test the accuracy of the positioning system was brought up. This led to the implementation and testing of features that would test the accuracy and how to be able to represent the ground truth. Having done the accuracy testing and finishing the use case features, the remaining part of the thesis included analysing the results and writing the thesis paper.

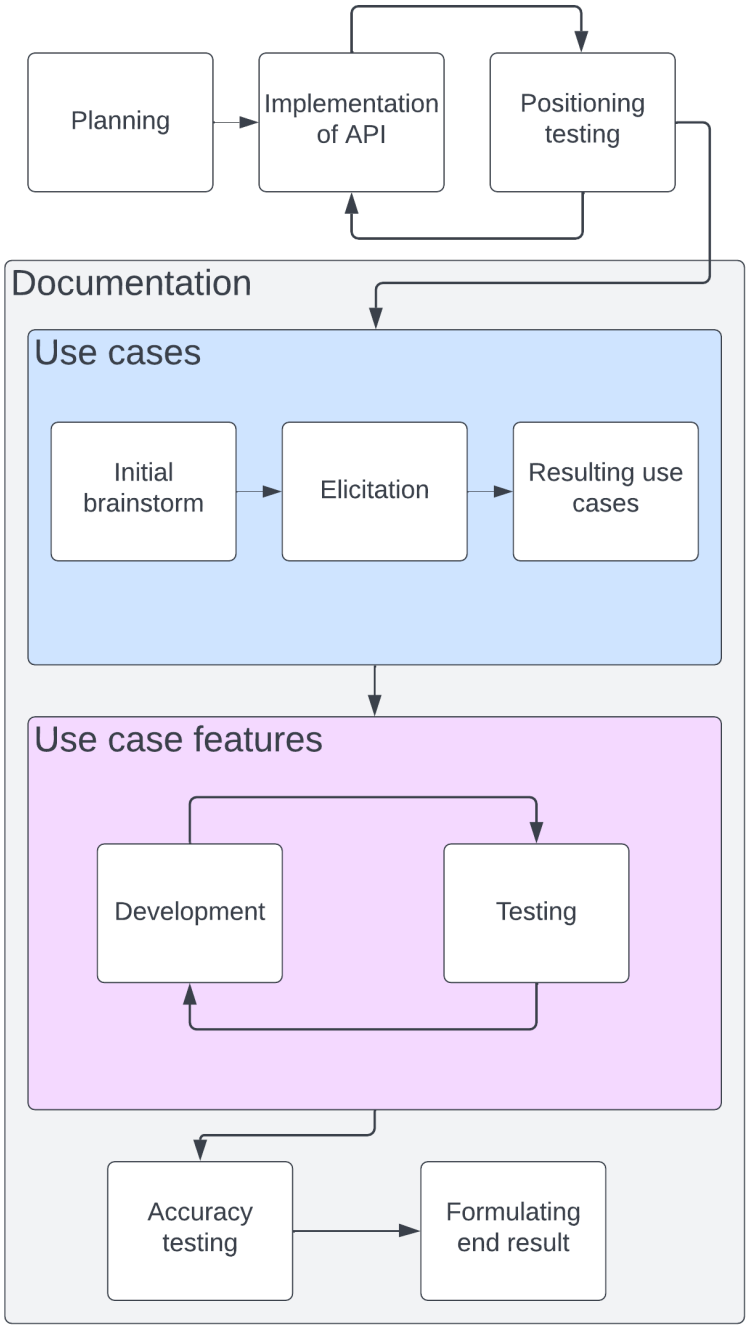

Figure 1: A visual representation of the workflow during while working on the thesis

3.3 Use Cases

This section describes the process in which the use cases that were used in the thesis were formed. It presents the elicitation process, which consisted of discussions with domain experts at the company and an interview with a security guard company. Lastly, the use cases that were decided upon are presented.

3.3.1 Elicitation

Since this thesis aims to find the possible uses of indoor positioning in real world scenarios, a decision was made to center the thesis around several concrete use cases to implement. Implementing specific use cases would make sure that what is developed has the potential to provide value for companies. The BWC has previously been used by security guards while patrolling areas. Therefore, basing the indoor positioning use cases on this was deemed suitable.

To initially form the use cases, a brainstorming session was held with a domain-expert at the company. During the brainstorming session, a lot of different ideas were found. Once the brainstorming was complete, the use cases were logically evaluated based on our assumptions about how security guards work. Based on this evaluation, some use cases were disposed of, and others kept.

After analyzing the possible candidates from the brainstorming session, three use cases were believed to be the most suitable for this thesis. The use cases were chosen by looking at three different criteria; The first is how time consuming it is to implement, since this thesis is made with a time constraint, there must be some limitations. The second is to what extent the authors and domain experts intuitively believe it could be a useful feature. And the third criteria used for the evaluation was to what extent it seems legally possible to use in the real world. Since there was no person with strong expertise in the legal area, the use cases were deemed to be more legally viable the less intrusive on privacy they seemed. At this point, these were the three use cases seen as most relevant:

- Detecting what route a guard is on, and triggering an alarm if deviations happen.
- Performing an action when in a certain area, such as turning the camera on.
- Adding positioning data to recordings in order to search for a specific recording by location.

An in-person interview with the security company was then scheduled to get more information about how security guards and security guard companies work. This information was then used to make the final decisions on what use cases the features to be implemented were based on.

3.3.2 Interviewing

The interview was conducted in a semi structured manner, where the interviewers would guide the talking points to a handful of different areas that were relevant for finding potential use cases. Exactly what things in those areas would be discussed was unknown, the only important thing was to ensure that it was information relevant to evaluating or developing the use cases. This could be things regarding a particular use case, but also general things about how the security guards work, which could give insight into what needs and wants a security guard, or a security guard company has.

To prepare for the interview, an interview questionnaire was created. Each row on the questionnaire corresponded to a certain topic, which acted as a list of topics that the interview was supposed to cover. Not knowing the time constraints of the interview, each topic was listed in order according to the priority. During the interview, the questionnaire would act as a guide for the interview to follow - if the conversation at some point stopped being relevant to finding use cases, the questions on it would be used to guide the conversation.

Something that differs between this questionnaire and a more traditional one (Like the one seen in [13]), is that it was filled out by the interviewers themselves directly after the interview. Each topic on the questionnaire was evaluated by the interviewers by either putting notes about it, or by putting a rating between "Completely incorrect" and "Completely correct". There were also notes made about topics that were not included in the questionnaire but still possibly relevant to the thesis.

3.3.3 Decided use cases

After the interview, the questionnaire created in preparation for the interview was filled in (See Appendix C). Currently, the security guard company were using a BWC from a different vendor, which in all relevant ways worked in the same way as the BWC used in the thesis. The security company were equipped with two additional devices; A radio communicator with the assault alarm feature, and a mobile phone designed for security guards with a bar code scanner attached to it.

There were several different "checkpoints" in the area they were securing. At each of these checkpoints there was a bar code, which they would scan every time they passed by it. The purpose of the bar code system was to provide a form of proof to the customers of the security company; Proof that they are patrolling the area with the desired frequency.

The security guard company also had a feature for triggering an assault alarm - whenever a security guard felt in danger, the guard would press a button and an alarm containing the guard's GPS position would be sent to the central system. From this central system, the devices on the other currently active security guards retrieved the assault alarm with corresponding GPS information. Whether their BWC or the security guard mobile phone was used to retrieve and send the GPS signal was unclear. Some of the security guards were reluctant to turn on the camera once they would start a session. To prevent this, it would be possible to detect when a camera was far away from the docking stations by using geofencing.

By using the information described above, there were 3 major use cases found:

- *Checkpoints* - Replacing the bar codes by automatically detecting what areas have been visited.
- *Location indexing* - Searching previous recordings by location.
- *Geofencing* - Detecting whether the device is in a certain area and performing an action based on it.

After having identified the use cases, an additional mail was sent to confirm that the analysis of the use cases were correct, specifically the *Checkpoint* use case. These answers cleared out what the distance between the placed bar coded usually is and where they are most commonly placed. This laid a foundation of where to place the checkpoints in the tests performed.

3.4 Prototypes

One of the goals in this thesis is to create a functional prototype for the BWC which can perform indoor positioning. To reach this goal, an iterative approach was used to develop and improve this prototype. Even though a functional prototype on the BWC is the end goal, two different prototypes were implemented, one on the BWC and one on a Linux laptop. Although they differ in both the programming language they are written in and the hardware they are based on, the general process in which the positioning was done is remains the same.

For the BWC, the prototype was written in the programming language C and executed Linux terminal commands from C to scan for both WiFi and Bluetooth access points. The BWC also ran on a custom version of Linux and the time to make changes could be long. For this reason, a prototype for the laptop was also implemented since it could be written in Python and supported the same method of scanning as the BWC. The time to make changes of the laptop was shorter and the process for testing the prototype was much less limited because of having an operating system that was easier to navigate.

To perform indoor positioning with WiFi and Bluetooth using the Combain API, a request containing information about nearby access points and their respective RSSI had to be sent with a HTTP-request in a JSON format. This was done by constructing a JSON-file with information captured from the device to be positioned. The information was captured by scanning the nearby area within certain intervals. This interval was set to either 2s or 4s depending on the test. When using Bluetooth, the interval was always 4s since it took at least 3s seconds to do one Bluetooth scan with the implementation of the prototype. After having sent the request, the response containing the position could be processed and the prototype evaluated. Before performing indoor positioning in a building, a survey is done using the "Combain Indoor Survey" mobile app described in Subsection 2.1.

3.4.1 Filtering MAC-addresses

A problem that may arise is that MAC-addresses from access points that are neither stationary, nor having a permanent MAC-address are sent to the API, which can cause skewed results (see figure 11). These access points could be either WiFi hotspots mobile devices or different types of Bluetooth devices such as headphones and other gadgets. Therefore, it seems necessary to filter out these access points.

When it comes to WiFi, a way to solve this is to check what company had issued the MAC-address that was captured. Many MAC-addresses can be linked to a company, which are identified using the first 3 octets of the MAC-address called the OUI part. Many cellphones, on the other hand, often change their MAC-address to random addresses which are not likely to be on associated with a reserved one. The solution is to compare the first 3 octets to a list of reserved MAC-addresses, and if it is not present, this could mean that it is an active hotspot and therefore removed from the request.

For Bluetooth, filtering could be done by using only the Bluetooth devices that had either a public or static MAC-addresses since this meant that they had a MAC-address that did not frequently change. This is more thoroughly described in Section 2

3.5 Visualizing data

In order to visualize the test results, a GUI application was built using Tkinter, which is a GUI-library for python [16]. Once a route was recorded and stored in a JSON request file on the BWC, the request file was transferred to a computer that was used to visualize the route taken. Each square represents the coordinates from a response from the API and the line between are put there to visualize the route taken. The squares are color coded from blue to red, depending on how far into the route they were captured.

To transform the coordinates into a position on the map, the relation between the length of the map in real world coordinates and the size of the picture in pixels were used. This meant that a coordinate could be mapped to a specific pixel on the screen in the GUI. To test if the plot was reliable, a coordinate was taken from google maps and plotted as a square on the map. The square was put in the same spot with just a tiny deviation which indicates that this method works well enough to visualize the result but might not be good enough for a production ready software. For the results from Kemicentrum, a map that Combain uses on their indoor platform website was used as a template and the routes taken were plotted on top of it. The map is a screenshot of Kemicentrum in Lund on Open Street Map with an indoor floor plan on top of it. Since this map is an approximation of how the floor plan looks in real life, the plot does not represent the exact location in a specific room.

Another visualization made was to convert the routes into a heat map. This was made using a JavaScript library called “heatmap.js” in a React application [17]. The reason JavaScript was chosen was that this library was easy to use, requiring just a few lines of code. The same algorithm used to convert coordinates into pixels was used to convert input data. Input data consisted of a bundle of routes taken with the same settings. The resulting heat maps were plotted to represent an average coverage of the positions captured.

3.6 Testing

The following section describes how the testing of the indoor positioning system was carried out. To begin with, it explains the process of the initial testing were the use of WiFi and Bluetooth is compared. It then describes how the ground truth was identified and how it is used in further tests. To end with, the process in which the accuracy of the system was evaluated is explained.

3.6.1 Initial testing

The initial tests were centered around learning how the Combain API should be used, finding possible bugs in the prototypes, and for the authors to get a intuition of how well the Combain API positions the camera using the laptop prototype described in Subsection 3.4. These tests were performed without specific measurements, and the results were evaluated visually with the GUI-application described in Subsection 3.5. These tests are not presented, since they resulted in no useful information for the goal of the thesis.

3.6.2 WiFi vs Bluetooth

To test the difference in accuracy when using WiFi and Bluetooth, a series of tests were done where only WiFi and Bluetooth were used. All these tests were done on the laptop prototype, to lower the time required to change the settings during the testing session. Two different settings were used during the tests, which were written in a settings file that decided how many access points of WiFi and Bluetooth to be used and the sample time of each scan.

Two Bluetooth beacons were used during the thesis, which aided in evaluating whether placing Bluetooth beacons in areas with low coverage had the potential to improve positioning results. This was done by placing Bluetooth beacons in locations where the accuracy and precision was low. If the API would place the device on this position once the Bluetooth beacon was there, it would be an indication that a Bluetooth beacon can help improve indoor positioning results. Finally, the importance of WiFi and Bluetooth AP:s is visualized by plotting two different heat maps. Each heat map contains the results of walking the route depicted in Figure 3. One of the heat maps contains data from strictly scanning WiFi AP:s, and the other strictly Bluetooth AP:s.

3.6.3 Finding the ground truth

To evaluate the positioning system, a location that is regarded as the ground truth is required to compare with the captured positions. The estimated ground truth is used later in the thesis when testing both the accuracy and the use cases. One way to determine the ground truth is to use one of the street maps that is available on the internet. In this thesis, both Google maps and Open Street Map (OSM) have been used to find the ground truth. Since Combain uses OSM, using it for finding the ground truth would be a logical choice, although, Google maps have been used as a comparison in some of the tests. To find the ground truth, these maps were used to find a location that can easily be identified in the real world. These locations could be in a corner or close to a window, making it easier to determine whether the device was positioned at the chosen location on the map. This method of finding the ground truth produces a margin of error since it is hard to position the device exactly at the chosen location. A discussion is held in the Analysis section about this method and the use of different maps.

3.6.4 Accuracy testing

To further evaluate the positioning system, it is necessary to measure the accuracy it produces. The purpose of testing the accuracy of the indoor positioning system is to get a picture of how well both the API and the prototypes function. In this thesis, the accuracy is referred to as the distance from an estimated ground truth location to the position retrieved from the positioning system. To calculate the distance between two positions, the great circle distance is calculated using the Haversine formula as seen in Formula 1 [2].

$$d = 2r * \arcsin(\sqrt{\sin^2(\frac{\varphi_2 - \varphi_1}{2}) + \cos(\varphi_1) * \cos(\varphi_2) * \sin^2(\frac{\lambda_2 - \lambda_1}{2})}) \quad (1)$$

It takes the latitude (φ_1, φ_2) and longitude (λ_1, λ_2) of both positions in radians. The radius r is the equatorial radius of the earth in meters, which has the length of $6378137m$ [3]. Furthermore, d is the distance in meters since the radius that is used is also in meters. As the Haversine uses the coordinates in radians and the coordinate from the positional system is in degrees, they are first converted accordingly. To check if this formula gives the correct distance, the result was compared to both the measure tool in Google maps [4] and an online calculator [5]. The error was about a couple of centimeters, which is acceptable since the method used for accuracy testing gives an error.

To get information about the accuracy of the indoor positioning system, a series of tests were performed to measure the distance between a retrieved position to an estimated ground truth. The ground truth position was estimated by using Google maps and Open Street Map to get the coordinates of a location inside Kemicentrum. The locations were chosen so that they could easily be identified as a real-world location, for example in a corner of the building. The tests were carried out with both the laptop and the BWC prototype, using both WiFi and Bluetooth. For the laptop, a scan was performed at the estimated ground truth location which captured five positional samples. Since the BWC had to be docked in the loading station, the test was initialized at the spot where the loading station was situated and then the BWC was carried to the ground truth location while scanning. A stopwatch was used to measure the time it took to both reach this location and the time back. With the measured time, the positional samples that were taken before having reached the location were disposed of so that only the ones that were captured when being at the ground truth location were saved. When having captured a couple of samples, the average accuracy and standard deviation was calculated at each point. Furthermore, a series of tests were conducted where the same captured position was used to compare to an estimated ground truth taken from Google Maps and Open Street Map. This test was done using the laptop as describe above.

3.7 Testing the use case features

In this section, the tests that evaluate whether the technology used for indoor positioning is viable for the chosen use cases are presented. These tests were carried out using the BWC prototype as described in Subsection 3.4. The section describes the chosen use cases in the order *Location indexing*, *Checkpoints* and *Geofencing*.

3.7.1 Location indexing

To test the location indexing use case, a series of test were performed to gather data to be analyzed. When having gathered the positional data, a rectangle area of the building was chosen and the coordinates in the corners of the rectangle were measured using the same technique as described in Subsection 3.6.3. Having the positional data and the area to compare with, a program that determined what recordings included positioning data in the given area was developed. The program would also output all the times where the area was visited. The implementation of this feature only used JSON-files and does not focus on how the positioning data could be combined with a video file.

3.7.2 Checkpoints

To test the *Checkpoint* use case, a series of test were prepared that simulates this process based on the analysis of the interviews. Each test involved walking the same route while performing indoor positioning on the BWC. Once the positioning data was collected, a program was used to test the checkpoint use case. The program would detect if a captured position was within a certain radius of a checkpoint and would then mark that checkpoint as visited. This would in turn check how many checkpoints were visited during a route.

To evaluate the performance of this feature, a couple of different checkpoint configurations were used. These configurations were a set of checkpoints, each placed on a specific spot in the testing area. When analyzing the route, the size of the radius was changed between 3, 6 and 10 meters. By using the same positioning data, the effect of different checkpoint configurations could be measured.

When performing the tests, five different checkpoint configurations were used. Using Google maps, each checkpoint was created by finding a location that suited the aim of the configuration. First, for the checkpoints to be similar to the way bar codes were used according to the Avarn Security interview, the checkpoints were placed in doorways or entrances to certain parts of a building (*Use case* configuration). The reason for this was that having visited a checkpoint, the person wearing the body worn camera would very likely have entered the room from the doorway, or the part of the building the corridor led to. In figure 7, there is an example of the checkpoint configuration that mimics the use case. Secondly, there was one where the checkpoints were placed towards the middle of hallways and areas that were being visited (*Corridors* configuration). That way the positioning system would detect whether a device was located in the middle of a room or a corridor, as opposed to a device only having been at the entrances. Thirdly, a configuration was made by placing the checkpoints were corridors intersected (*Crossroad* configuration). Another configuration was used with randomly placed checkpoints, randomly being placed in a corridor but without further consideration of the location of the checkpoint (*Random* configuration). Lastly, a configuration was made by analyzing the heat map in Figure 7 (*Heat map* configuration). By looking at the spots that were showing the most captured positions, the locations that are most covered with the best accuracy can be identified. Placing a checkpoint at these locations should yield the best result.

During the tests for the different checkpoint configurations described above, three different scanning methods were used. These methods were scanning using WiFi, WiFi and Bluetooth and only using Bluetooth. By changing the types of access points to scan, the result could be analyzed to see which performed the best. This can be seen as an addition to the WiFi and Bluetooth tests described in Subsection 3.6.2.

To further evaluate the *Checkpoint* use case, a test was prepared using different sample times between the scanning and a varying number of access points captured. Since the duration of each Bluetooth scan was around 4 seconds, it was decided that when testing varying sample times, only WiFi were to be used since it was easier to control the time between scans. The sample time is an interesting variable since it can impact things such as power efficiency for the BWC, accuracy of the positioning, or cost of using the API. Changing the number of access points was done to check the number of access points effected the result or if it were enough to use a certain amount to get a reliable result. When doing these tests, the checkpoints configuration used were the *Use case* configuration described above as well as an average for all these configurations. The sample time were chosen between 2s and 4s. The number of access points were either 20 or 40. Documentation of all tests can be find in Appendix B for all the different specifications described above. Thereafter, it is possible to estimate the success- and failure rate of each setting in the use case feature.

3.7.3 Geofencing

When it comes to testing *Geofencing*, tests were performed in order to evaluate whether the indoor positioning system is able to reliably determine whether a device is located inside a given room. The tests were performed by starting a recording, then walking towards the room being tested on. A stopwatch tracked the time when the device would enter the room as well as exit the room. Using these times and the received positioning data, a comparison was made to measure how clearly the positioning system determines if the device is located within or outside the room. The geofenced rooms have thick, brick walls, which leads to them isolating signals from outside of the rooms well. To make the model of the geofences, the same method as described in 3.6.3 was used using either google maps or Open Street Map. The rooms were modeled as rectangles, where the north, west, east, and south boundary was represented by coordinates. There were some problems when trying to model a room, since the maps only show the outline of a building. To solve this, rooms were chosen that are easily identified by the outline of the building. For instance, some of the tests were done in auditorium A in Kemicentrum. This room is perfectly outlined by the map on both Google map and Open Street Map. The other rooms that were chosen were auditorium G and F. An image showing these rooms and the fences can be seen in Figure 2. The square indicates the geofences and the letter is the name of the room. Note that the fence does not match the map where it is plotted upon. This is due to GUI used to plot the points is not entirely correct, yet the fence and points are correct relative each other.

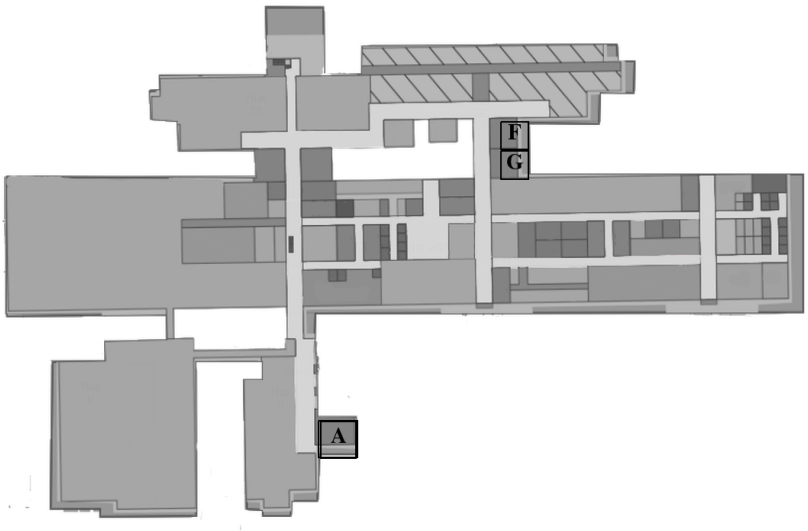

Figure 2: An image visualising the geofences used during the tests.

3.8 Source evaluation

In this subsection, each source used during the thesis will be evaluated upon how trustworthy it is. During the course of the thesis, different sources have been picked by the authors. The level of criticism each source got depended on the purpose of the citation. If a citation merely served as an example, the source was not deemed to need to be as strong as other ones. The strength of a source was mostly based on the authors of the document.

The source in [1] is admittedly not perfectly reliable since it is a master thesis. On one hand, it is reviewed by the professors, on the other hand the thesis is written by people with a small amount of experience in the field. However, the purpose of the source is simply to give an example of the fact that positioning systems based on inertial navigation tend to drift.

The source for [2] is referenced from Wikipedia. When looking for a formula for converting distance in coordinates to meters, the Haversine formula was found in multiple sources, included on Wikipedia which sited the mentioned source. This reference is therefore deemed as reliable.

The website in [3] is a website by Nasa and is therefore deemed trustworthy.

The reference [4] is to Google maps, made by Google and is a manual on how to measure a distance using Google maps.

The website in [5] is referenced due to the calculator being used to validate if the Haversine formula [2] is working properly. It is not used to base any knowledge or tests on, only to get an idea of what values to expect from the calculations.

The source [6] is an official website by Novel Bits and the author have worked many years with Bluetooth. The amount of information about the authors make it a credible source.

The list of MAC-addresses [8] taken from [7] is used to filter out hotspots. The reliability of this method in itself is not considered very trustworthy by the authors and is further discussed in Subsections 5.2. Because of this, it was deemed that credibility of the source was not critical.

The author of [9] is IEEE, which is an institute for Electronics and electronic engineers. IEEE created many technology standards in the industry of computer science, and is therefore evaluated by the authors to have a good idea of how MAC-addresses work.

The website in [10] is an official website by the "Sveriges ingenjörer"-association. The authors deem this association likely has correct information about the code of honor for engineers.

The source of [11] is an official website ran by the EU, which means it is highly likely to be up to date with correct information.

The video in [12] is posted by the Combain youtube channel, owned by their company. It was uploaded in 2019, so some changes to the positioning API might have occured since then, but the issue with cutting corners is still present, which is what the source was meant to visualize.

The paper cited in [13] is an introduction to doing research in education and social science. It is cited by 1000 people, which lead the authors to believe it is aligned with the industry standards when it comes to questionnaires.

The website in [14] is the official website of Combain, and the technical details described there is therefore to be fully trusted. The same goes for [15].

The citation in [16] has the official python documentation website as a source. It is an official and frequently updated website, which lead the authors to believe it was reliable.

[17] cites to the personal website of the creator of the heatmap.js library. Along with the author having good reason to advertise the library, the technical details in the site are to be trusted.

4 Results

This section presents the results from the testing sessions of the prototypes and use case features described in section Testing. The information acquired is used to answer the questions described in the Problem subsection (Section 1.3). To illustrate the results, images with the captured positions are plotted as dots with lines and heat maps is used together with tables with data from the tests. All images are based on an indoor floor plan of Kemicentrum, which was the place where the tests were done. The floor plan is taken from the Combain website, but the colors have been altered to make the heat maps and plots stand out more. In figure 3, the route that has been taken for most tests is drawn as a line in black color. This route applies to most tests, if another route is used, that route is specified. The route was taken in both directions during the tests.

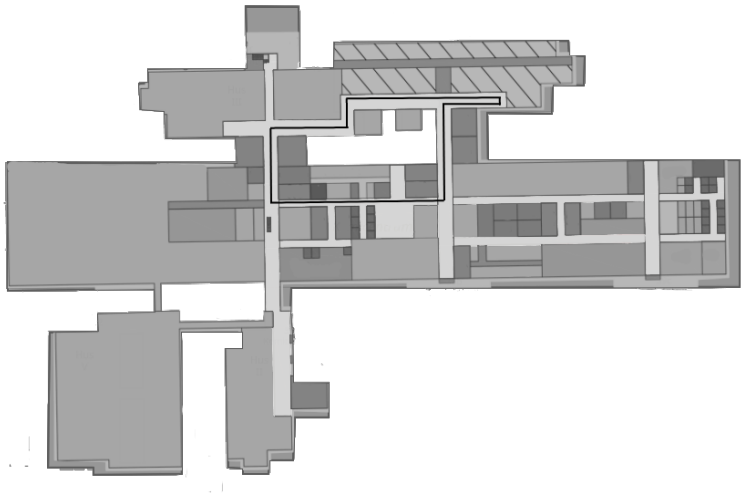

Figure 3: An image showing the route taken in most of the tests. The black line indicates the path that was walked while the prototype performed positioning.

4.1 WiFi vs Bluetooth positioning

When testing the prototypes, a comparison between using only WiFi and Bluetooth respectively was done according to section 3.6.2. The positions captured from these results are bundled up in one file and plotted as heat maps below.

Figure 4 shows the results from only using Bluetooth. The route depicted in Figure 3 was taken. Two Bluetooth beacons were placed in point 1 and 2, where it is clear that responses were received. The positions captured at the points 1 and 2 in the Bluetooth tests were not received prior to placing out the beacons there, and thus are a result of using beacons. The other spots mark areas where other Bluetooth devices used for positioning were present in the building at the time of testing. A total of two laps are used in this heat map, depicting the total amount of positions captured using only Bluetooth AP:s. As the image shows, there are not many Bluetooth AP:s available for indoor positioning. The reason for this is discussed in *Analysis* (Section 5).

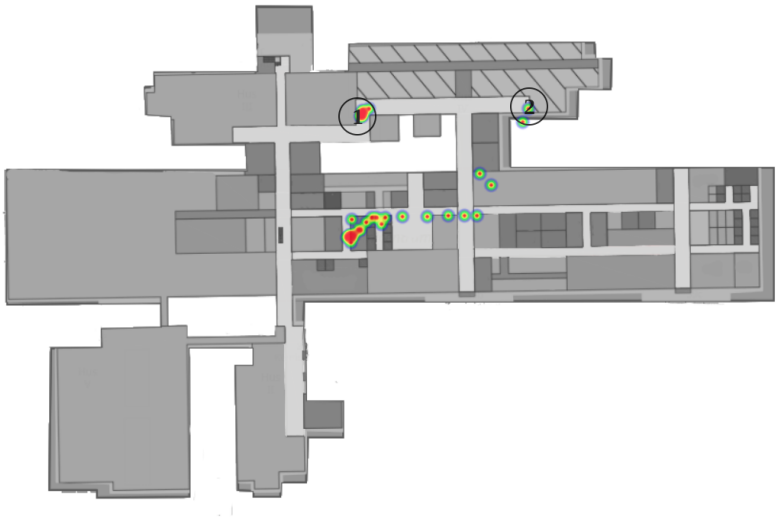

Figure 4: A heat map using all the positioning data received when scanning only Bluetooth AP:s.

Figure 5 shows the results from the same route taken as above while only using WiFi. In this heat map the captured positions of a total of 18 routes are plotted. The routes taken, with the current settings used is described in Table 1, and the routes were taken in both directions. All access points are present in the building since forehand, in other words, no additional infrastructure was added. One can see that there is more coverage compared to Figure 4. Section 5.3 discusses why this might be. One can also see that the coverage at points 1 and 2, where the Bluetooth beacons were located, has diminished.

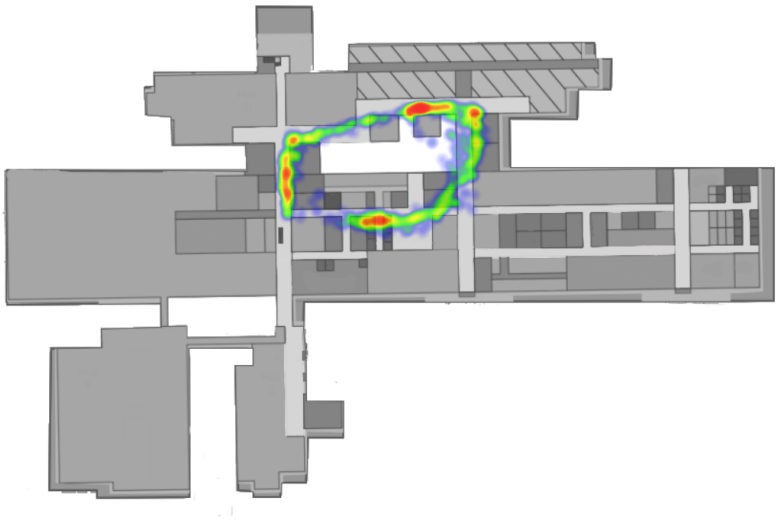

Figure 5: A heat map using all the positioning data received when scanning for strictly WiFi AP:s.

4.2 Accuracy testing results

When it comes to testing the accuracy of the API, the goal was to find an approximate number representing the accuracy and the standard deviation of the positioning system. To do this, 12 different reference points were used as ground truth locations, as shown in Figure 6. At each point, several measurements was carried out using both WiFi and Bluetooth. The average accuracy and the standard deviation for the measurements where then calculated. For a further explanation of how the tests were carried out, see Section 3.6.4.

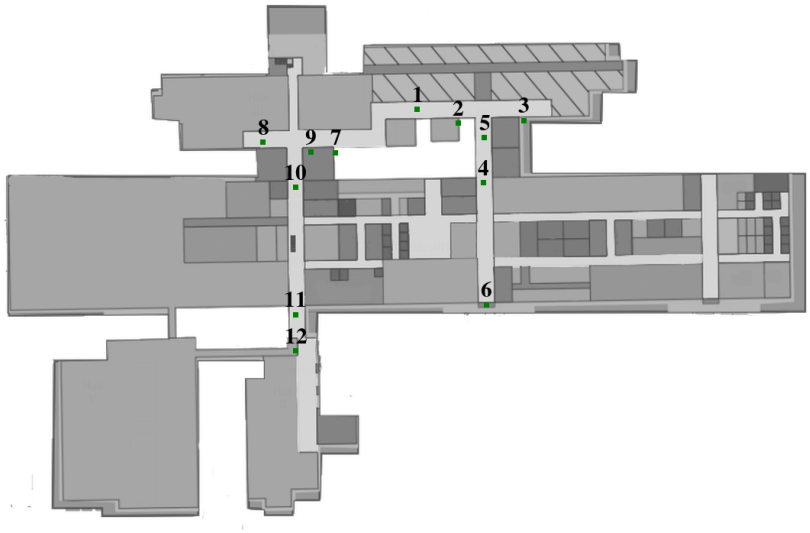

Figure 6: An image containing the different reference points used to test the accuracy of the API.

At each of these reference points, five requests were made using the laptop prototype. For each result, the differences between the ground truth position and the evaluated position in meters were calculated using the formula described in 3.6.4. The averages and standard deviations from these calculations are shown in Table 1. To check for potential differences between the different prototypes, reference points 7 to 11 were tested with the BWC prototype as well. Due to the technical implementation of our BWC prototype, a varying amount of requests were made, as opposed to five for the laptop implementation. At point 10 in 1, there is a large difference between the standard deviation of the BWC and that of the Laptop. A similar thing occurred at point 7, where there is a considerable difference between the averages and the standard deviations between the BWC and the Laptop. To see a discussion about what may have caused these differences, see section 5.8.

Table 1: Table 1 shows the results from testing the accuracy. For the raw data, see Appendix A Table 8 and Table 9.

Point	Laptop		BWC	
	Average (m)	Standard deviation (m)	Average (m)	Standard deviation (m)
1	4.21	1.85		
2	3.11	1.17		
3	18.0	16.32		
4	7.08	0.41		
5	5.45	1.7		
6	13.69	1.01		
7	8.23	0.76	12.29	4.04
8	6.91	0.62	14.41	0.63
9	4.67	0.23	6.98	1.83
10	6.32	9.8	6.43	0.41
11	8.09	4.13	6.27	1.48
12	17.31	3.8		
Averages (m)	8.59	3.48	9.28	1.68

In Table 2 below, there is the result of calculating the average error and standard deviation for both the laptop and the BWC, across all points.

Table 2: Table showing the overall average accuracy and standard deviation from the tests.

	Accuracy (m)	Standard deviation (m)
Averages BWC + Laptop	8.94	2.58

4.3 Testing the use case features

Each feature (*Location indexing*, *Checkpoints*, and *Geofencing*) has a corresponding test, each of which are presented in this sub section. This section also contains various figures and tables which are used to present various testing results.

4.3.1 Location indexing

When it comes to the testing of this feature, everything worked as expected. The program delivers the desired output with an unnoticeable delay. For an analysis on the usefulness of this feature, see section 5.9.

4.3.2 Checkpoints

The tests for the *Checkpoints* use case were carried out as explained in section 3.7.2 and used five different checkpoint configurations, one of which can be seen in Figure 7. During these tests, the route that was taken was that one depicted in Figure 3. The tests consisted of walking the route while the BWC was performing indoor positioning. After a route was taken, the positions captured during the route was saved and analyzed further afterwards. When analyzing the captured data, the positions were compared to a set of checkpoints and a percentage of checkpoints hit (detected to have been visited) was calculated. These tests were carried out with different scan settings which consisted of using only WiFi, Bluetooth, as well as both. For each setting, the radius of a checkpoint was changed between three, six and ten meters. Below in Table 3 are the results from testing the checkpoints set shown in Figure 7.

To depict a summary of all positions captured from the tests on the checkpoint feature, a heat map was made (See Figure 7). The heat maps were created with the responses captured from all WiFi and Bluetooth tests. By looking at the intensity of color in each checkpoint, one can see that some checkpoints are hit more often perform better than others. For instance, Checkpoint 1 rarely got hit. The most intense color means every recording of the route contained a response in that position. The circled numbers are the checkpoints used during testing and the used radius is 6m.

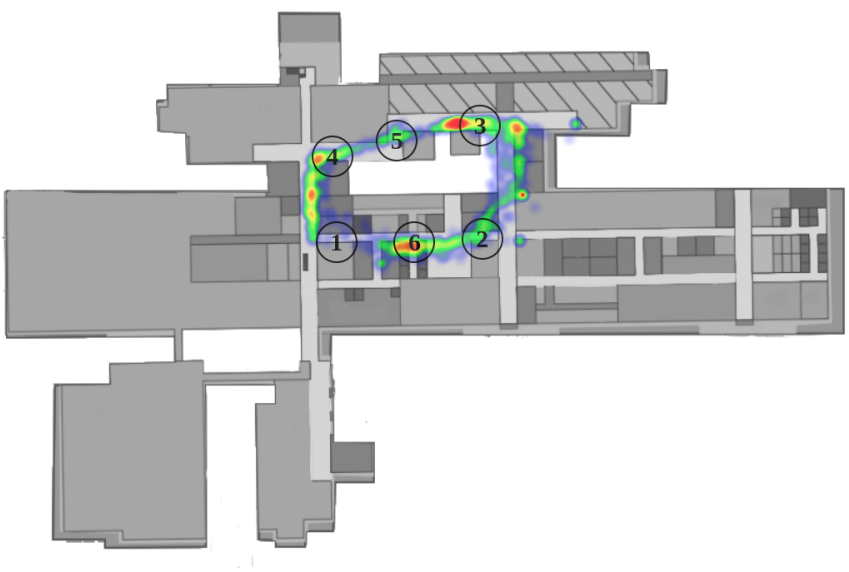

Figure 7: The checkpoints in a configuration mimicking the use case (*Use case* configuration), with a heat map of all responses gathered during all these tests overlayed.

For each of the five checkpoint configurations, an average of checkpoints hit was calculated to show the difference in performance for different scan- and checkpoint settings. The results from the use case configuration, which had highest chance to detect each checkpoint, is shown in Table 3 below. In this table, Checkpoints hit shows the average proportion of checkpoints hit per recording. To see the rest of the data for each specific checkpoint configuration, see Appendix B.

Table 3: Results from testing the checkpoint configuration mimicking the use case with different radii for the checkpoints. The checkpoint locations for this configuration is shown in Figure 7.

Checkpoints: *Use case* configuration			
Mode	Laps	Radius (m)	Checkpoint hit chance (%)
WiFi	10	3	68.33
		6	88.33
		10	98.33
WiFi + Bluetooth	4	3	54.17
		6	87.5
		10	95.83
Bluetooth	2	3	41.67
		6	41.67
		10	41.67

Table 4 shows an average over all configurations further described in Appendix B. One thing to note is that the hit rate is around 90% for both WiFi and WiFi + Bluetooth with 10m radius. Since this is the average of a large number of different checkpoint placements, this is an indication that this is the expected hit rate for a randomly placed checkpoint using this positioning system, at least in Kemicentrum which was the testing area for this thesis.

Table 4: Table showing the average hit rate on checkpoints with varying radius.

Checkpoints: Average all configurations			
Mode	Laps	Radius (m)	Checkpoint hit chance (%)
WiFi	10	3	49.33
		6	76.0
		10	90.38
WiFi + Bluetooth	4	3	38.1
		6	73.1
		10	90.12
Bluetooth	2	3	28.33
		6	37.62
		10	39.29

The results from testing the different settings for the sample time and amount of access points is shown below in Table 5. The table shows the average hit rate over all checkpoint configurations. The sample time is the interval between the scanning for access points. When using 2s sample time, the amount of WiFi access points were also changed between 20 and 40.

Table 5: Table showing the results from checkpoint testing using WiFi with varying sample time.

Checkpoints: All configurations using WiFi, varying sample time			
Mode	Laps	Radius (m)	Checkpoint hit chance (%)
WiFi 4s	3	3	49.68
		6	73.02
		10	89.21
WiFi 2s 20AP	2	3	54.76
		6	78.81
		10	93.57
WiFi 2s 40AP	5	3	46.95
		6	76.67
		10	89.81

To show a common pattern that emerged for the WiFi tests, visualizations of a WiFi-only test are shown in Figure 8. Many of the checkpoints are missed with small margins, showing that this method of determining that a device have been at these locations can be unreliable. It also depicts that checkpoint 2 is never visited, possibly because it is not covered with any WiFi access points.

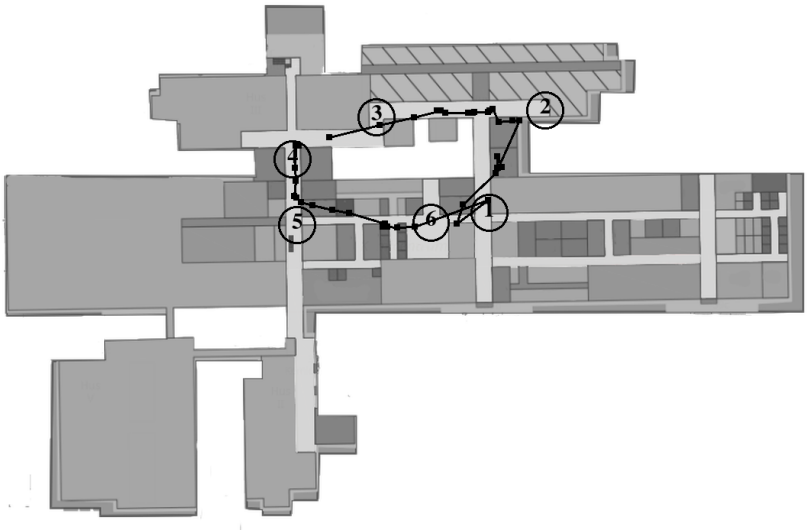

Figure 8: An example of a test with 3/6 checkpoints hit. The checkpoint configuration used is the *Random* configuration. Results from testing this configuration can be seen in Appendix B, Table 13.

When using only Bluetooth, only a few requests gave a valid position (See Figure 9). This small number of valid positions indicates a lack of Bluetooth AP:s that can be used to position the device. Because of the configuration used (*Random*), and having checkpoints where the Bluetooth beacons were positioned, the hit rate was higher than other configurations when using only Bluetooth (About 58% compared to the average 39% when using a radius of 10m).

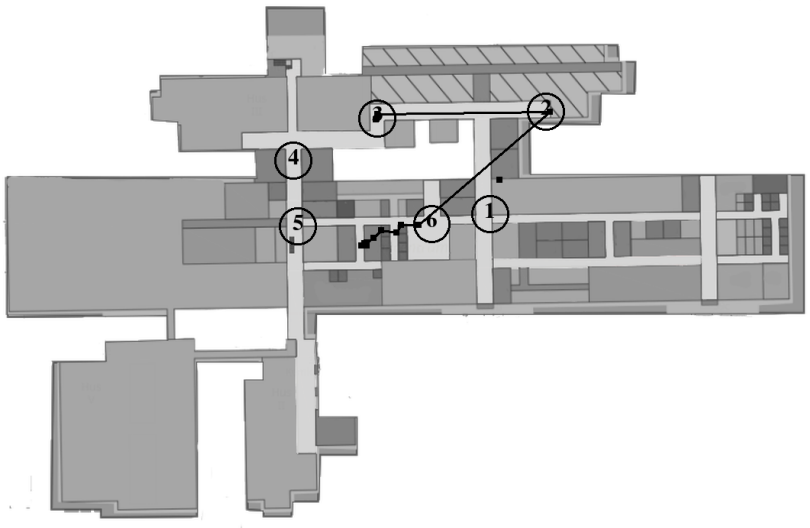

Figure 9: The results from walking the route described in Figure 3 when using Bluetooth only and two beacons placed at checkpoint 2 and 3. The checkpoint configuration used was the *Random* configuration.

4.3.3 Geofencing

The tests concerning *Geofencing* used a rectangle shaped model of a room and checked whether a captured position was inside it. To see the rooms being geo fenced, see Figure 2. To get a result, a log file containing the time stamp when the modeled room first was entered was written. This was then compared to the timestamps recorded with a stopwatch during the routes. The hit rates of the *Geofencing* tests can be seen in Figure 6. A fence is visited if a captured position during the route taken is within the area of the geofence. For room A, two different geofences for the same room was used, one taken from Google maps and one from Open Street Map. For rooms F and G, the rooms were split up in two geofences using Google maps.

Table 6: The rate in which the geofences were visited.

Recording	Room	Hit rate %
12:47	A	2/2
12:44	A	2/2
12:42	A	2/2
12:39	A	2/2
12:20	FG	2/2
12:13	FG	1/2
12:10	FG	1/2
12:01	FG	2/2
11:56	FG	2/2
Total	Both	16/18

A comparison between the stopwatch times and the API response times can be seen in Table 7. When a device entered and exited a room, there was a consistent discrepancy. This can be seen on the averages and standard deviations in Table 7. Since the requests sent to Combain are independent from another, the problem with the delay has to do with the way the scanning for access points is performed. A discussion about this topic is held in Section 5.

Table 7: The delays between reaching and exiting the rooms. The 12:42 and 12:39 recordings seem to have considerably lower delays, for reasons unknown to the thesis authors.

Recording	Room	Reach delay (s)	Exit delay (s)
12:47	A	22	23
12:44	A	31	1
12:42	A	0	1
12:39	A	9	3
12:20	FG	21	-12
12:13	FG	24	-15
12:10	FG	26	19
12:01	FG	26	6
11:56	FG	28	32
Average	Both	20.8	6.4
Standard deviation	Both	9.4	14.7

In Figure 11, there is a figure visualizing an example of the *Geofencing* tests. This test was one of the tests with large delays. One can see the enter and exit delays by observing the large distances between the points for entry and exit. The square indicates one of the geofences used during the tests. Note that the fence does not match the map where it is plotted upon. This is due to the GUI used to plot the points not being entirely accurate, yet the fence and points are correct relative each other.

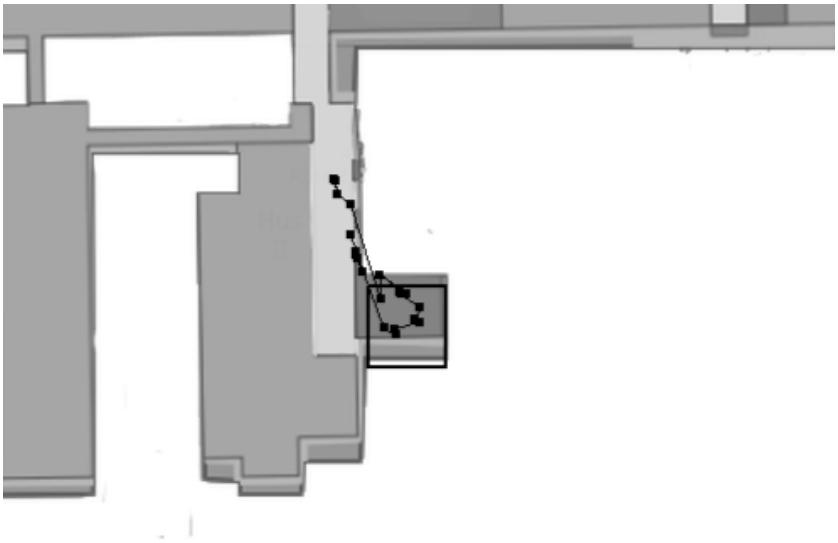

Figure 10: An image visualizing a geofencing test with a room.

4.3.4 Hotspot on cellphone phone active

When doing one of the tests, the hotspot on one of the authors' cellphones was active. This hotspot was also added to the building model through a survey and acted like a normal access point. When the survey was done, the hotspot was active around checkpoint 1, and that was the position it was modelled as. In Figure 11, one can see a recording disturbed by this hotspot. When starting to walk the route, the hotspot was in the same position as the starting point around checkpoint 3. The RSSI received from the hotspot was as such high, so the position received was close to where the hotspot was added in the building model. This made the route skewed; And only when the RSSI from the hotspot became low enough, the correct position was received from the API. When walking back to starting position, the same pattern can be seen. Instead of the end of the route being at checkpoint three, the route ends at checkpoint 1.

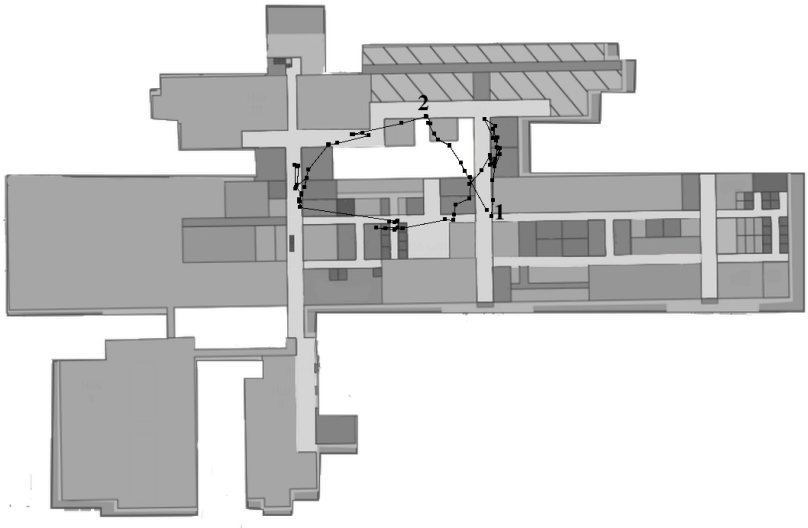

Figure 11: Visualizing the route taken where the positioning is skewed.

5 Analysis and discussion

The purpose of the analysis section is partly to analyze and evaluate the different decisions made during the thesis project, and partly to analyze the results from the testing of the features. Firstly, an analysis of the way use cases were shaped is presented. Then a discussion on the different techniques used during the thesis is held, involving; The way the scanning was done, the difference between using WiFi vs Bluetooth, the problems with mobile hotspots on cellphones, how displaying the captured position on heat maps can be useful and how the different maps were used to find the ground truth. It continues to analyze the accuracy results. Lastly, the features implemented for the use cases are analyzed.

5.1 Use cases

The elicitation of the use cases was conducted through a face-to-face interview, where the interviewers would fill in a questionnaire after the interview. Working with a questionnaire in this manner could be problematic, since it may not reflect the exact opinions of the interviewee, but rather the impressions the interviewers perceived. This could lead to the result of the interview being slightly skewed.

On the other hand, having an interviewee fill out a form after the interview was also considered to be a problematic alternative. A questionnaire might be seen as an annoyance by the interviewee, which could lead to the interviewee answering the questions hastily in order complete it as fast as possible. Using this method, there would be more focus on the discussions instead.

Another thing to consider about the interviews, was that there was only a single interview conducted. Only having one interview might not have given enough information about the workflows of the security guards in order to make a definite conclusion about whether or not the use case features would be an improvement to their current way of work. Another method to shape the use cases in a more reliable way would be to perform tests with security guards. In this way they could share their knowledge about the domain in a more natural way.

5.2 Scanning

The way the scanning was performed plays a central role in the performance of the positioning system and can be further discussed. As mentioned in Subsection 4.3.3 on *Geofencing*, there is a delay in the system. This could be caused by the underlying technology behind the terminal commands used. The output for these commands could be older values which are not updated in time or must be reset manually. Another possibility is that the difference between the RSSI captured is not large enough to make a difference on the position. An alternative way of processing the data captured from the access points could be to use an average RSSI value over a couple of scans. This could lead to a more accurate result but would require a longer interval between the scans.

When filtering WiFi MAC-addresses, a method was used that looks up the vendor of the access point based on OUI part of the MAC-address. This method assumes that the MAC of a cellphone is randomly assigned due to privacy issues and is not likely to be in the list used to look up common vendor owned OUI:s. This assumption could not be fully assessed during the thesis and was only briefly tested on two cellphones to evaluate its reliability. First, it is not confirmed by the authors that cellphones are in fact being assigned random

MAC-addresses. Secondly, if assigned random MAC:s, it is not confirmed that they cannot be assigned the same OUI as one associated with an access point vendor by chance. However, since it worked for the cellphones used during the tests for surveying and taking time, it was determined a sufficient solution.

5.3 WiFi vs Bluetooth

The results of the tests from Kemicentrum shows that using WiFi in an unprepared infrastructure is much more reliable than using only Bluetooth. The building has a larger coverage with WiFi access points and in many cases, only using WiFi is sufficient. A reason for this could be that many of the Bluetooth devices present in a building are filtered out due to their MAC addresses being either non static or public type as discussed in Section 2. It could also be that the API itself filters it out, but since the API is considered a black box in this thesis, no definite conclusions can be drawn about this theory.

5.4 Hotspot on cellphone

When conducting the first survey in Kemicentrum, the hotspot on the cellphone used was active by accident. The hotspot was recognized as a WiFi access point and was added to the Combain building model. In a later test, together with the BWC which in turn led the positioning system to believe it was not moving at all. The hotspot had a fixed position in the model so the results given was that the BWC was still at that fixed position since the RSSI from that hotspot was much larger compared to other access points. During another test, the cellphone was placed on a table where the route started from, but the hotspot was surveyed to be in another place. This meant that in the start of the route, the positioning systems thought that the BWC was close to the place the hotspot was surveyed to due to receiving strong RSSI when being close to the phone. This phenomenon can be seen in Figure 11.

The fact that hotspots or other movable access points can be added to the building model during a survey could be a problem for the reliability of the positioning system. If a mobile access point happens to have been added to a building model, any time that access point is in the area of a device being positioned by the positioning system, it causes disruptions.

5.5 Heat map and low accuracy areas

During the thesis, a couple of heat maps were created combining the captured positions from multiple routes taken. When looking at the heatmap from all the recordings put together (see Figure 7), one can see some colourless spots. Since the same route is always taken, these colourless spots must indicate biases in positioning data. The explanation for these spots could be due to the signal strengths from the access points being poor in those areas. There could be different reasons for the signal strength data being poor, such as lack of access points to scan, or an environment with bad conditions for WiFi and Bluetooth connectivity. To address this problem, a heatmap can be generated of a certain building to get a picture of how well the default infrastructure can support indoor positioning. The areas that are not well covered can then be complemented with low energy Bluetooth beacons such as was done in Figure 4.

5.6 Map accuracy and Ground truth

The quality of the map being used while plotting routes could affect the resulting positioning. On one hand, one might think an inaccurate map could lead to discrepancies in the positioning system, since the coordinates put into the system are not entirely correct. On the other hand, if the same map is used as the one in the Combain surveys, these discrepancies would be the same for both maps. This results in a sort of local coordinate system, where the positions received from the API have the same offset as the ones plotted on the map. For these reasons the authors are unsure about whether the positioning accuracy is negatively affected by inaccurate maps of buildings. Since the way the routes and the heat map are painted, by having the corners represent real world coordinates, the plots are always correct in relation to the window. The map used on the other hand cannot be verified as showing the real building. In the map used, both corridors and rooms are present, but this is just a general representation of how it looks like in the real building.

This discrepancy was at first deemed to not affect results, since whether the shape was intact in relation to the map was believed to matter the most. However, when starting the accuracy testing, a realization about the implications of map inaccuracies were made. When measuring accuracy, one must know the coordinates that the device is currently located at (ground truth), and then compare those coordinates to the ones received from the API when performing a scan. The way the ground truth was retrieved in this thesis, was by using the same map as the positioning API and retrieving the coordinates of a point. To find the real-world equivalent of these points, easily discernible locations were used, such as corners of rooms. But because of the map inaccuracies, there was no way of knowing if the coordinates for these points are the actual ground truth, or if there was an offset.

5.7 Use case problem: Does everyone have a map of their building?

When it comes to using maps to place checkpoints or geofences, it could be problematic. Some buildings are old, and thus do not have an accurate map. There are also buildings whose layouts are secret for security reasons, such as the layout of a bank. This could make it necessary to find an alternative way to find the approximate coordinates of certain areas if this use case was to be implemented in such an area.

5.8 Analysing the accuracy testing results

When doing the accuracy testing on the laptop, there were some unexpected results. The testing was performed by measuring the captured position to an estimated ground truth at a point several times. For point 10 as seen in Table 1, there was a large standard deviation (9.8m) compared to other points as the average standard deviation was 3.48m. When looking at Table 8, the measurements for point 10 are around 1m for all except one, which is around 50m. The reason for is not fully understood. To visualise these recordings, Figure 12 was made. The red squares show the estimated ground truth for points 3 and 10. The blue squares are the captured positions when located at the respective points.

Figure 12 and Figure 13 show a possible relationship between the positions captured from Bluetooth scanning and the positions captured from point 3 and 10. The squares in Figure 12 that are far away from the ground truth are located near the hot spots on the heat map. Indicating that each of the highly deviated results were believed by the positioning system to have been located at one of the Bluetooth access points in Figure 7.

Figure 12: Image showing the positions captured from accuracy testing. During both measurements, one position is considerably farther away from the ground truth than the others.

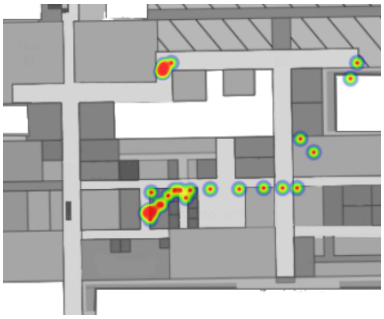

Figure 13: Image showing the same heat map is in Figure 4 but cropped to a smaller size.

5.9 Location indexing

Searching the recording JSON-files for positions in a certain area resulted in a correct result without complications. Despite this, there are two points of concern when it comes to this feature. First, the tests that were performed in this thesis used a small amount of data (<2MB). This means it is unknown whether this feature would search recordings fast enough for it to be useful in a real-life scenario, where multiple guards record their positions for hours per day. Secondly, it is unknown to the authors how large areas would be searched when using this feature. Based on the accuracy testing done in section 4.2, if the area is 10 meters or larger, it is reasonable to believe the search function is reliable enough. However, if one is searching for positions in for instance a small bathroom, the accuracy could lead to the search feature filtering out recordings that should not be.

5.10 Checkpoints

During the thesis, a couple of tests were performed to check whether the bar codes used to verify that a security guard have visited a location could be replaced by automatically detecting a position within a checkpoint. The fact that none of the tests resulted in a 100% hit rate means that this method could not be trusted in a real-world application. A security guard patrolling an area needs to have a correct proof that they have been there to show the customer. If they have been in an area but the positioning system misses the checkpoint, then the system cannot be trusted. This method needs to be fully functional, by having a hit rate of 100% to be able to be used.

There is a comparison that can be made between the accuracy testing and testing the *Checkpoint* use case. The average accuracy when testing with the BWC was 9.28m, which is slightly below the radius used in the 10m configuration for the checkpoints. This would mean that on average, all checkpoints would be visited using 10m radius. However, the rate averages to 90.12% using both Bluetooth and WiFi, which is what was used during the accuracy testing. When looking at Figure 7, one can see that Checkpoint 1 is often missed, which could be why the hit rate is not closer to 100 %. The reason for missing Checkpoint 1 could be that the specific area does not have a good coverage of access points. It could also be due to a common pattern that emerged during the recordings - when sharp turns were made, the corners were often cut. This can be seen in the heat map (Figure 7), for instance at checkpoint 5 or 1. The same phenomena can be seen in the demo video of the positioning API made by Combain themselves [12].

One way to assure that the hit rate for the checkpoints is as high as possible is to place them in areas where the accuracy and precision is high. This can be achieved by analyzing the heat maps created with all captured positions in a building or by analyzing the accuracy in a specific location with the method described in Subsection 5.8. The result from testing this theory is showed in Table 14 in Appendix B. This gives a hit rate of 100% when using the same method used to create the heat map which was analyzed (WiFi and Bluetooth). However, this result was achieved when having a checkpoint radius of 10m. Since these bar codes are often placed in doorways, one could argue that this requires a radius of about 1m or even less to fully mimic this way of working.

The result from testing different sample time intervals and varying amount of access points can indicate what settings are best to use when performing indoor positioning. However, for these tests, there were not a large difference between the use of different settings. There was no clear indication on what performed the best. Since the number of laps taken during these tests are low, it is hard to draw a conclusion whether one functions better than another.

5.11 Geofencing

The thesis explored the use of geofences to determine whether the device was located inside a room or not. When performing the tests for *Geofencing*, there was on average a delay of 20.8 seconds for detection on entering a geofence, and an average delay of 6.4 seconds when exiting one. Because of the scanning interval being set to 4 seconds, a delay of detection is expected. One can imagine a scenario where a scan is finished just before entering a room, which would lead to the worst-case detection time being two sample intervals, which is 8 seconds. Because entering a room never took more time than one sample interval, larger delays than this must therefore be a result of a lack of positioning accuracy.

There are a couple of reasons for these delays; First, it could be caused by how the Bluetooth beacons were set up in the rooms - perhaps a placement that were closer to the entrance and exits would have reduced the delays due to there being a more sudden change of signal strengths for the Bluetooth beacons. Secondly, since the requests sent to the API are independent from one another, the problem could be in the way the scanning for access points is performed. The underlying technology in which the way the scanning is performed is not explored much as explained in the Technical background Section 2 and in the Limitations under Section 1. A possibility for the delay could be that data from older scans are still captured from the output of the terminal commands used. The older values could be stored and the command does not show the new values if not manually reset and updated.

There can be discussions when it comes to how the environment the geofencing takes place in affects the results. If the walls isolate a room being geofenced from outside signals, there might be an improvement in the reliability of that geofence since signal strengths are impacted more when entering or leaving. The authors believe it is likely that there are additional properties about a room that could affect the reliability a geofence, such as the area, the surrounding infrastructure, the layout, or the layout of the building it resides in. However, the authors have chosen not to look further into this issue, since the authors do not believe they could achieve reliable results due to their lack expertise in how radio waves function.

There is an important discussion to be made surrounding whether it is reliable enough to be put into real world use. The answer to the question depends on what requirements are put on the geofence. What are the acceptable chances for false positives and false negatives occurring? Is there a strict constraint on the delay of a detection? In what conditions does the geofence reside in? The authors do not possess the information needed to answer these questions, therefore the judgement of the usability of the use case were simply based on how reliably it detected if a device was located inside or outside the geofences.

When it comes to judging the reliability of the *Geofencing*, the authors believe the test results show that using an indoor positioning system like it is

currently implemented is not reliable enough to be used for geofencing. For this to work in a more reliable way, the way the scanning is performed needs to be reviewed. With a more reliable method, the delay could be less of a factor, resulting in a better performing geofencing feature. Although the geofences were often all marked as visited, when plotting the testing of the G and H rooms, the positioning system does not detect a clear distinction between when the device is inside a room and when it is outside of it. For the use case to be fulfilled, there would need to be a clear distinction between if a device is located inside a room or in a corridor next to a room.

The tests that were performed in this thesis only measured the chance of a correct positive. Additional tests regarding how often false positives occur could result in useful information, since having a false positive at an unfortunate timing could lead to important moments not being recorded.

6 Future work

This section contains things that the authors think have a chance of being useful to implement in the future to improve the indoor positioning system. These things include covering different use cases, different techniques, improvement to current features, and improvements to the AP-fetching algorithm. There are different reasons these things were not developed, such as lack of expertise, it being outside of the scope of the thesis, or lack of reward to time ratio.

6.1 Improvements to the positioning system

One could imagine some possible algorithms to improve the accuracy of the positioning system. As previously mentioned, a request to the Combain API does not depend on a previous one. To more accurately depict a human walking, perhaps there are some ways to filter the responses from the API. An example of one of those filters would be a Kalman filter, which out of multiple less accurate samples derives a more accurate final sample.

Another possible improvement could be to more accurately model the environment in which it is used. For example, if there is a graph depicting the different possible ways to walk in each building, that graph could be used to help determine where a person is located by assuming a person cannot be inside of a wall. This could for example be most useful when it comes to the problem of corners being cut while turning.

To achieve an improved positioning accuracy, the authors believe it could be possible to incorporate inertial navigation by using for instance gyroscopes and accelerometers. In a masters thesis by Markovska M and Svensson R [1], they measured an accuracy drift of 0.2 meters per meter traveled. The authors believe this drift could be remedied by frequently calibrating the inertial navigation system when a given device is located where the accuracy of the indoor positioning tends to be high.

6.2 Additional features using indoor positioning

Since it is often the case that guards tend to walk the faster routes between the different checkpoints, it would be interesting to recommend new routes when a specific route has been walked too often. This would however require a lot of effort since it would either require the program to save previous routes or to have a graph containing the different ways to navigate a location.

7 Ethics

The ethics section discusses some of the potential ethical and legal implications of the developed indoor positioning system. This includes things such as GDPR, privacy issues, and potential use of positioning data as court case evidence.

When thinking about the security guards that would use indoor positioning, one can imagine that they would think their privacy is at risk. Since the device continuously gathers positioning data, there might be concerns about what their employer could be using that data for; One could imagine that an employer with this data would be able to compare the efficiency of different workers using the indoor positioning data. There could be some form of agreements between the security guards and employers to solve the problem. These agreements would guarantee that the employers do not use the positioning data in a way that is not purely beneficial for the security guards. However, this would go against the first principle in the Swedish engineers' code of honor; [10] "Engineers in their professional capacity ought to feel personally responsible for technology being used in a manner that benefits humanity, the environment and society.". To take personal responsibility that this technology does not cause detriments for the security guards, the companies implementing a positioning system like in this thesis should make sure the system is used strictly for its intended purpose.

When it comes to secrecy surrounding videos by BWC, there are additional secrecy issues. Recorded video is not accessible to the public and is according to the interview with the security guard usually only handled by law enforcement. A question that needs answering is whether positional data falls under the same category as recorded video. The answer to that question shapes the use cases in which indoor positioning can be applied to. This is because there is less use to perform indoor positioning if the data is not available for security companies to use due to legal constraints. The same things apply to performing real time monitoring of the position, which, when ignoring this potential legal issue, could lead to widely applicable use cases. Legal questions:
1. Is it possible to enable real time positioning?
2. Who is allowed to access positional data after recordings?

Before a private company starts recording any form of positioning data, a clear answer to the legal questions and concrete measures against privacy concerns should be in place. Even if the employees agree to being subject to the positioning system, the company using it would have to be mindful of the different laws that could interfere. An example of a set of such laws would be GDPR [11], which puts restrictions on how personal data can be processed.

8 Conclusion

This section presents the conclusions made based on the analysis. To start off with, the questions presented in the introduction are answered in order. It then summarizes the work done in this thesis and to conclude with, a concise presentation of the conclusion made as a whole.

8.1 Answers to the list of problems

The list of problems that were formulated in the beginning of the thesis is answered below:

What are the use cases for an indoor positioning system on security guards?

There were three use cases that were elicited from the security guard in the face-to-face interview, which are presented below.

The idea of the first use case was to use the positioning system to replace a system made to report to customers of the security company that an area was being patrolled often enough. This system used bar codes placed in different locations in an area being patrolled, and when a security guard visited one of these areas, they would scan the bar code to document they visited that area. To replace this system, the areas with the bar codes could be exchanged for digital checkpoints. If a BWC were to be positioned in a certain radius of one of these checkpoints, it would be reported to have visited that area. In order for this system to be able to replace the existing one, it would have to recognize that a BWC is in a checkpoint accurately enough, without having to make the checkpoints too big for the resulting data to become uninteresting for the customer.

For the use case *Geofencing*, the idea was to evaluate whether a program was able to determine if the BWC was located inside a room. This would be useful since certain actions could then be made depending on the position of the device; For instance, to make sure the camera always is on during a patrol, the camera could be turned on after leaving the room where it is being docked and unused.

The last use case, position indexing, involved associating a recording with positioning data. This could be useful for when law enforcement wants to view recorded footage, since it could reduce the time it takes to find the right recordings.

What accuracy and precision is required to satisfy these use cases?

To answer this question, a discussion about what use case requires the highest accuracy can shed a light on what the bottleneck is. From having analyzed the results from the interviews and discussion during the thesis, the *Checkpoint* use case is probably the one requiring the highest accuracy. If the idea is to replace bar codes being placed in doorways, the accuracy needs to be high. To know for certain that a security guard have been in the same position as when scanning the bar code, the accuracy would need to be less than a meter. However, one can determine that a guard have passed through a passage using two checkpoints. This requires a model of the building to determine where the checkpoints are to be placed. If a guard is to walk through a corridor, having one checkpoint in the end and in the beginning can give enough information to tell if they have passed through the doorways leading there. The accuracy needed to satisfy this requirement depends on the building but having a checkpoint radius of 3 meters should be sufficient.

What accuracy and precision is achievable?

From testing the accuracy and the precision of the indoor positioning system using both the laptop and the BWC, an average of 8.9 meters in accuracy was measured. The average standard deviation from the tests was 2.6 meters (See table 1). The accuracy differs depending on the area where the positioning takes places. An area where the coverage of both WiFi and Bluetooth is high, the accuracy is higher, ranging from 3-7 meters or even down to 1-2 meters. In areas where the coverage is low, additional Bluetooth beacons can be used as a complement to make the accuracy and precision higher in these locations.

This achieved accuracy is however specific for Kemicentrum, the place the tests were conducted in. Additional testing environments would yield a more general result.

How can a geographical ground truth be established and with what accuracy?

To establish a geographical ground truth, the same map was used as the one used for the survey done for the Combain API. A coordinate on the map was picked at a point that is easily recognizable in real life. The chosen coordinate would be considered ground truth when standing on the corresponding real-life location.

What information about access points is useful for indoor positioning?

For performing indoor positioning, the most important things are MAC-addresses and RSSI:s, since the MAC address is an identifier for an access point, and the RSSI gives an indication as to how far away that access point is. The OUI part on the MAC-addresses can be used to identify the vendor of the access point and therefore make conclusion about what type of access point it is. The data captured from scanning for Bluetooth access points also tells what type of MAC it is. These can be either public or private MAC:s, where the private can be either static or random. By identifying the type of address or the OUI, the MAC:s of the access points not suited for indoor positioning can be filtered out.

How does the amount of access points impact the accuracy and precision?

There needs to be more testing done to reach a definitive answer to this question. The results in section 4.3.2 suggest no meaningful difference between the different number of AP:s, apart from the fact that Bluetooth performs poorly on its own.

Is it possible to detect which side of a wall a device is located?

The results show that the positioning system is consistently able to detect it. However, on average it takes 20 seconds until the detection occurs. How reliable the detection is, is also not clear, further testing would have to be done to test the false positive detections.

Is WiFi or Bluetooth better in a non-prepared environment?

From the testing at Kemicentrum, it was clear that WiFi performed better. This could be due to the API filtering out most Bluetooth AP:s, or because there are few of them in that house.

How can Bluetooth beacons be used to improve indoor positioning?

By placing Bluetooth beacons, the chance of getting a better accuracy in the direct vicinity of the beacon is improved. However, they do not seem to improve the overall accuracy much when looking at the results from these tests. When performing tests using the beacons, positions near the beacons were consistently captured with very high accuracy, as can be seen in Figure 4.

Does the indoor positioning system implemented in this thesis perform well enough for the chosen use cases?

As discussed in question 9.1.2, the accuracy required to satisfy the *Checkpoint* use case is around 1m if it should be able to mimic the way bar codes are scanned. However, if they are placed in a thought through manner, one could be able to infer that the security guards have taken a specific route.

The results from the *Geofencing* tests show that the positioning system is able to detect whether the BWC is inside a room. However, there is a delay until it detects a position inside the room and until it leaves. The reason for this is believed to be due to how the scanning on the prototype is implemented by not getting updated RSSI values. Furthermore, there were no tests performed when walking outside the room because there was not much time left when this was discussed. With these additional tests, a conclusion could be made whether the system reported false positives regarding being in a room when in fact being outside. To conclude, the positioning system can detect when the device is in the room but with a delay and perhaps false positives.

Indexing recordings by position worked as expected. A user can put coordinates as an input to the program, and as an output the user receives the recordings that contain a response in that location. The thesis authors believe the size of the area being searched through could affect how useful the use case would be, because of the accuracy of the positioning.

8.2 Final conclusions

Using WiFi and Bluetooth, a position can be calculated in a non-prepared environment. The positioning system performs well enough to determine a position with less than a 10-meter error on average. With this performance, the positioning system can reliably tell what part of a building the BWC is located.

The positioning system does not perform well enough to exactly mimic the use of bar codes used by security guards to track the route taken. However, with logically placed checkpoints, the route taken by a guard can be inferred. With the use of heat maps, the expected accuracy in an area can be measured and used to determine in what areas a captured position can be trusted.

When it comes to *Geofencing*, the BWC is not able to reliably determine if it is located in an area of a building or in a large room. Using the results from the accuracy testing, the authors would not recommend using geofencing if area being geofenced has a radius of less than 10 meters. The chances for a false positive to occur have not been examined.

Searching for a recording located in an area is proven to be possible. As with the previous use cases, the reliability of this feature is tightly coupled with the accuracy of the system. Since this use case does not require the accuracy and precision as the other once, it can more reliably be implemented.

Judging from the interview with a security guard, a way to position a BWC would be useful for the industry. However, the authors believe the positioning system in this thesis has a too low accuracy to be able to fully fulfill the requirements put on it by the use cases, especially *Geofencing* and *Checkpoints*. It is critical that these features are reliable since they are used as either a verification of a security guards work (*Checkpoints*) or controlling whether the camera is recording (*Geofencing*).

There could be various improvements to indoor positioning in the future, which leads the authors to believe it is possible to reach an adequate accuracy for the use cases elicited in this thesis. If these developments happen, and legal and privacy questions are cleared up, the authors believe indoor positioning has the potential to add value to the security guard industry.

References

[1] Markovska M, Svensson R. Evaluation of Drift Correction Strategies for an Inertial Based Dairy Cow Positioning System [degree project on the Internet]. Stockholm, Sweden: Kungliga tekniska högskolan; 2019 [2022/06/02]. Available from: https://www.diva-portal.org/smash/get/diva2:1366127/FULLTEXT01.pdf%20exjobb%20som\%20jobbar%20med%20positioning%20med%20inertial%20navigation

[2] Gade K. *A Non-singular Horizontal Position Representation.* Journal of Navigation. Cambridge University Press; 2010;63(3):395–417.

[3] Williams DR. Earth Fact Sheet [Internet]. Maryland, USA: NASA; 2022 [2021/12/21;2022/06/02]. Available from: https://nssdc.gsfc.nasa.gov/planetary/factsheet/earthfact.html

[4] Google. Measure distance between points [Internet]. Google; 2022 [;2022/06/02]. Available from: https://support.google.com/maps/answer/1628031?hl=en&co=GENIE.\Platform%3DDesktop

[5] Williams E. Great Circle Calculator [Internet]. [2022/4/11;2022/06/02]. Available from: http://edwilliams.org/gccalc.htm

[6] Afaneh M. Bluetooth Addresses Privacy in Bluetooth Low Energy [Internet]. :Novelbits 2020 [2022/04/11; 2022/05/20]. Available from: https://www.novelbits.io/bluetooth-address-privacy-ble/

[7] Wireshark. OUI Lookup Tool [Internet]. [;2022/06/02]. Available from: https://www.wireshark.org/tools/oui-lookup.html

[8] Wireshark. [Internet]. [;2022/06/02]. Available from: https://gitlab.com/wireshark/wireshark/-/raw/master/manuf

[9] IEEE. Guidelines for Use of Extended Unique Identifier (EUI), Organizationally Unique Identifier (OUI), and Company ID (CID) [Internet]. 2017[2017-08-03;2022/06/02]. Available from: https://standards.ieee.org/wp-content/uploads/import/documents/tutorials/eui.pdf

[10] Sveriges Ingenjörer. Hederskodex [Internet]. [2022/03/07; 2022/06/02]. Available from: https://www.sverigesingenjorer.se/om-forbundet/organisation/hederskodex/

[11] European Union. General Data Protection Regulation [Internet]. [;2022/06/02]. Available from: https://gdpr-info.eu/

[12] Combain. Indoor Positioning Combain Office [Internet]. 2019 [2019/05/10;2022/06/02]. Available from: https://www.youtube.com/watch?v=CaO5ffE78so

[13] Bell J. Doing Your Research Project 4/e: A guide for first-time researchers in social science, education and health [Internet]. United Kingdom: Open University Press; 2005. 4th edition. [2022/06/02]. Available from: https://www.amazon.com/Doing-Your-Research-Project-researchers/dp/0335215041

[14] Combain. Locate everything everywhere [Internet]. [;2022/06/02]. Available from: https://combain.com/

[15] Combain. Indoor positioning [Internet]. [;2022/06/02]. Available from: https://combain.com/use-cases/indoor-positioning/

[16] Python Software Foundation. tkinter — Python interface to Tcl/Tk [Internet]. [2022/06/02;2022/06/02] https://docs.python.org/3/library/tkinter.html

[17] heatmap.js : Dynamic Heatmaps for the web [Internet]. 2022 [;2022/06/02] Available from: https://www.patrick-wied.at/static/heatmapjs/

[18] Body worn solutions, Axis communications [Internet]. 2022 [;2022/06/02] Available from: https://www.axis.com/solutions/body-worn-solutions

Appendices

A Accuracy testing

Point	Measurements				
	1(m)	2(m)	3(m)	4(m)	5(m)
1	4.63	3.52	7.07	1.33	4.49
2	3.6	4.44	1.94	4.07	1.5
3	50.63	9.66	9.86	9.91	9.92
4	6.32	7.0	7.29	7.42	7.37
5	2.69	6.33	4.55	6.01	7.66
6	11.9	14.95	13.53	13.9	14.16
7	9.68	7.74	7.53	8.1	8.12
8	5.95	6.99	6.73	7.89	7.0
9	4.48	4.91	4.44	4.97	4.54
10	1.38	1.56	1.7	25.92	1.05
11	7.29	5.42	16.07	4.53	7.12
12	14.47	13.82	21.61	14.38	22.28

Table 8: This table shows the complete data from accuracy testing with the laptop as seen in Table 1. The points represent each location the measurement was taken. For the laptop testing there were always 5 measurements for each point. As seen in point 3 and 10, there is sometimes a large discrepancy between the measurements, showing that the positional system can sometimes give a bad result.

Point	Measurements							
	1(m)	2(m)	3(m)	4(m)	5(m)	6(m)	7(m)	8(m)
7	17.33	16.16	15.17	14.24	13.29	6.69	6.41	9.0
8	15.49	14.2	14.01	13.93				
9	8.89	9.45	8.33	6.83	7.31	3.61	5.27	6.17
10	6.41	6.83	6.64	6.08	5.73	6.9		
11	6.08	7.12	5.0	5.52	4.57	7.81	9.07	5.03

Table 9: This table shows the complete data from accuracy testing with the BWC as seen in Table 1. The points represents each point the measurement was taken and are the same points in Table 8. For the testing on the BWC, the amount of measurements varied since the tests were done during a random time period.

B Checkpoint testing

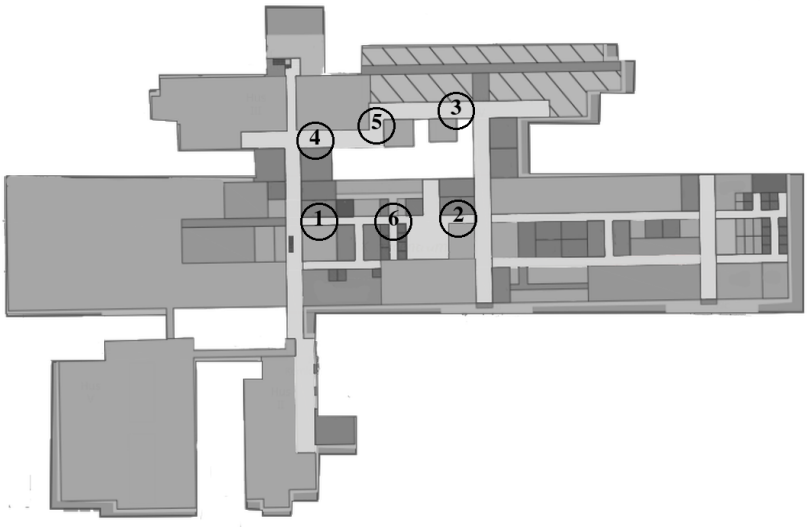

Figure 14: Checkpoint configuration *Use case*

Checkpoints: *Use case* configuration			
Mode	Laps	Radius (m)	Checkpoint hit chance (%)
WiFi	10	3	68.33
		6	88.33
		10	98.33
WiFi + Bluetooth	4	3	54.17
		6	87,5
		10	95.83
Bluetooth	2	3	41.67
		6	41.67
		10	41.67

Table 10: Table showing the result from checkpoint testing were checkpoints are placed in opening of doorways and into corridors. This configuration was made to mimic the result from the interviews.

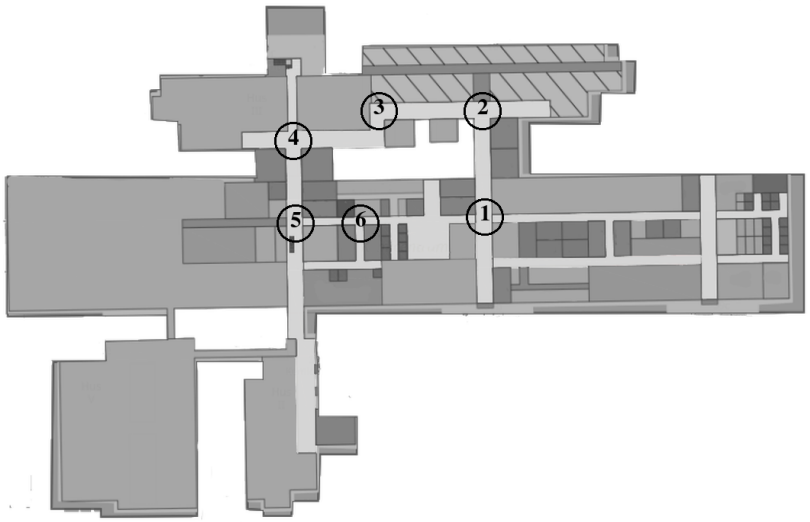

Figure 15: Checkpoint configuration *Crossroad*

Checkpoints: *Crossroad* configuration			
Mode	Laps	Radius (m)	Checkpoint hit chance (%)
WiFi	10	3	26.67
		6	68.33
		10	91.67
WiFi + Bluetooth	4	3	25.0
		6	58.33
		10	87.5
Bluetooth	2	3	33.33
		6	33.33
		10	41.67

Table 11: Table showing the result from checkpoint testing were checkpoints are placed were corridors intersect. This configuration was made to check whether a guard walks through a crossing between corridors.

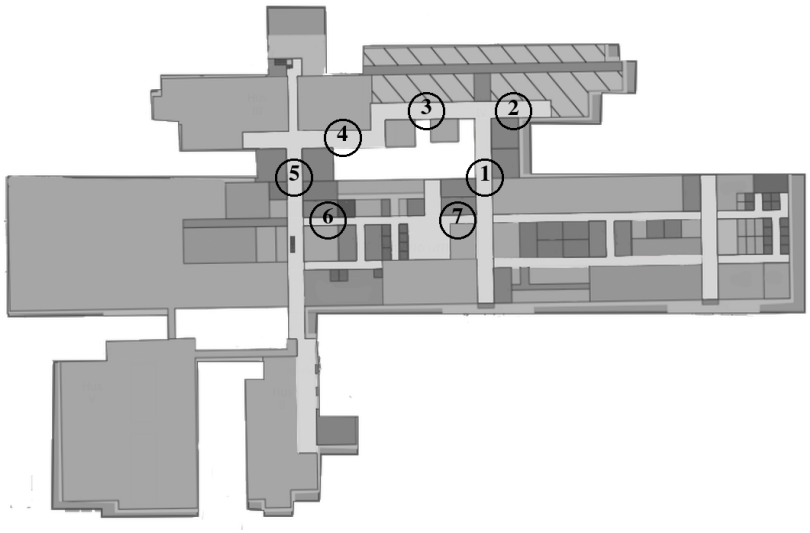

Figure 16: Checkpoint configuration *Corridor*

Checkpoints: *Corridor* configuration			
Mode	Laps	Radius (m)	Checkpoint hit chance (%)
WiFi	10	3	50.0
		6	80.0
		10	88.57
WiFi + Bluetooth	4	3	32.14
		6	82.14
		10	96.43
Bluetooth	2	3	0.0
		6	21.43
		10	21.43

Table 12: Table showing the result from checkpoint testing were checkpoints are placed in the middle of a corridor. This configuration was made to check whether a guard have walked in a specific corridor.

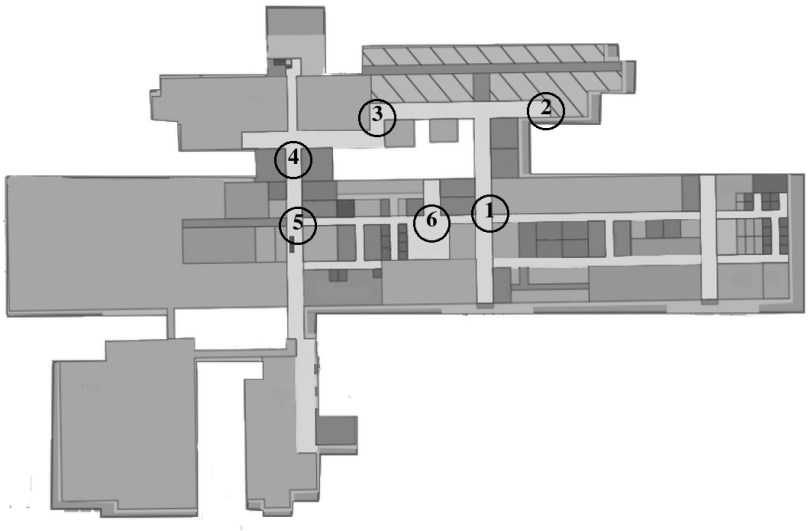

Figure 17: Checkpoint configuration *Random*

Checkpoints: *Random* configuration			
Mode	Laps	Radius (m)	Checkpoint hit chance (%)
WiFi	10	3	21.67
		6	50.0
		10	76.67
WiFi + Bluetooth	4	3	12.5
		6	45.83
		10	70.83
Bluetooth	2	3	16.67
		6	41.67
		10	58.33

Table 13: Table showing the result from checkpoint testing were checkpoints are placed at random locations inside a corridor. This configuration was made to see what the result is when the checkpoints are chosen at random.

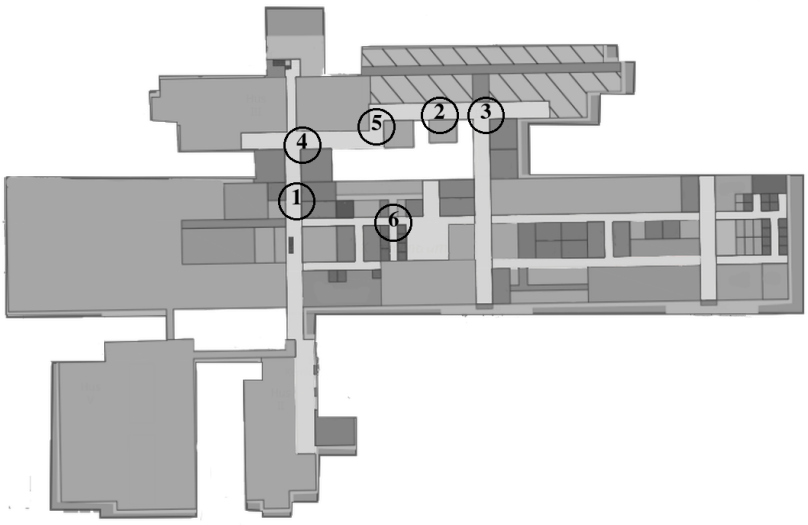

Figure 18: Checkpoint configuration *Heat map*

Checkpoints: *Heat map* configuration			
Mode	Laps	Radius (m)	Checkpoint hit chance (%)
WiFi	10	3	80.0
		6	93.33
		10	96.67
WiFi + Bluetooth	4	3	66.67
		6	91.67
		10	100.0
Bluetooth	2	3	33.33
		6	33.33
		10	33.33

Table 14: Table showing the result from checkpoint testing were checkpoints are placed at places where the heat map in Figure 7 shows the most filled spots. This configuration was made to check what happens if the checkpoints are placed in areas with good coverage and accuracy. The heat map was made with the responses captured from using both WiFi and Bluetooth. As seen in the WiFi + Bluetooth row with a radius of 10m, the hit rate is 100%.

Checkpoints: Average three configurations			
Mode	Laps	Radius (m)	Checkpoint hit chance (%)
WiFi	10	3	48.33
		6	78.89
		10	92.86
WiFi + Bluetooth	4	3	37.1
		6	75.99
		10	93.25
Bluetooth	2	3	25.0
		6	32.14
		10	34.92

Table 15: Table showing the average hit rate on checkpoints with varying radius. The three configurations used were *Use case* (Figure 14), *Crossroad* (Figure 15) and *Corridor* (Figure 16).

Checkpoints: Average all configurations			
Mode	Laps	Radius (m)	Checkpoint hit chance (%)
WiFi	10	3	49.33
		6	76.0
		10	90.38
WiFi + Bluetooth	4	3	38.1
		6	73.1
		10	90.12
Bluetooth	2	3	28.33
		6	37.62
		10	39.29

Table 16: Table showing the average hit rate on checkpoints with varying radius. The table shows an average over all configurations used. One thing to note is that the hit rate is around 90% for both WiFi and WiFi + Bluetooth with 10m radius. This is an indication that this is the expected hit rate for a random placed checkpoint using this positioning system, at least in Kemicentrum which was the testing area for this thesis.

Checkpoints: All configurations using WiFi, varying sample time			
Mode	Laps	Radius (m)	Checkpoint hit chance (%)
WiFi 4s	3	3	49.68
		6	73.02
		10	89.21
WiFi 2s 20AP	2	3	54.76
		6	78.81
		10	93.57
WiFi 2s 40AP	5	3	46.95
		6	76.67
		10	89.81

Table 17: Table showing the results from checkpoint testing using WiFi with varying sample time. The table shows the average hit rate over all checkpoint configurations. The sample time is the interval between the scanning for access points. When using 2s sample time, the amount of WiFi access points were also changed between 20 and 40

Checkpoints: Use case configuration using WiFi, varying sample time			
Mode	Laps	Radius (m)	Checkpoint hit chance (%)
WiFi 4s 20AP	3	3	66.67
		6	83.33
		10	100.0
WiFi 2s 20AP	2	3	75.0
		6	91.67
		10	91.67
WiFi 2s 40AP	5	3	66.67
		6	90.0
		10	100.0

Table 18: Table showing the results from checkpoint testing with the use case configuration (Figure 14) using WiFi with varying sample time. The sample time is the interval between the scanning for access points. When using 2s sample time, the amount of WiFi access points were also changed between 20 and 40.

C Interview questionnaire

Assumptions	Completely incorrect	Incorrect	No clear response	Correct	Completely correct
Is a BWC currently in use during work?					x
What tasks does the BWC perform?	The camera is constantly recording. When a button is pressed, the previous 60 seconds are saved and the future recording is saved until the button is pressed again. barcodes are scanned in order to prove to the customers of the security company that they have been securing those regions.				
Do you have any use for GPS positioning?					x
What are those GPS uses?	They send the GPS coordinate of a guard to the xxx when he is being assaulted.				
Are you in need of indoor positioning?			x		
Of what accuracy?	It would be optimal to be able to tell the difference on which train station platform a person is located on.				
Could geofencing be interesting?		x			
Is it useful to perform an automatic action based on geofencing?		x			
What would one of those actions be?	He was unable to come up with a scenario where it would be useful.				
Do you see any issues with storing positioning data from a secrecy standpoint?		x			
Would there be problems surrounding secrecy when it comes to using realtime positioning data?	x				
Do you feel okay with being positioned in realtime in terms of privacy?					x
Do you ever walk in set routes?		x			
Are there similar jobs that use routes? Do you have any information about the process in which they do so?	Guards usually use more specific routes than they do, but he does not have much information about their process.				
Would it be useful to see where along a route someone has been?		x			
Would it be useful to perform an action if someone does not reach a position at a certain time?		x			

Table 19: The different questions and answers put into the questionnaire. Some questions are put as being different levels of agreement, and some questions are a summary of the thoughts of the security guard. The questions are ordered by how important they were felt to be.