

DANMARKS TEKNISKE UNIVERSITET

DesignBuild
62521

Indendørs Navigation På Hospitaler Med Beacons

Andreas Bach Berg Nielsen s205869

Jesper Clement Aaløse s205488

Naveed Imam Shah s205491

Shiv Gopal s205490

Troels Engsted Kiib s205492

December 27, 2023

Abstract

In this project we have done a complete product development cycle, from research, analysis, design, implementation and tests. We have made an Application which can be used to navigate indoors without use of GPS but with implementation of iBeacons to determine positioning of the user. During the development of the application, we found Mazemap whom has already made indoor maps and an API that we can use. However we have since found out that jetpack compose and Mazemap are incompatible so we have not been able to make a full implementation of iBeacons. Instead what was implemented was two applications running on top of each other, where the beacon will send a local notification so the user can change position manually, whereas if it was implemented fully the beacons would be able to change the position automatically. Even with the complications we found, we still managed to make a functional product which we tested. The tests resulted in 42% time saved trying to find directions and we found that 7 out of 10 people tested preferred using our product than using the signs and wallmounted map in the place of tests conducted. Therefore we can conclude that iBeacons is a viable option for indoor navigation, however the use of jetpack compose with Mazemap for mapping is not advised.

Indholdsfortegnelse

1	Formål	4
2	Indledning	4
2.1	Baggrund	4
2.1.1	Komplekse hospitaler	4
2.1.2	Økonomisk byrde	5
2.2	Problemformulering	5
2.3	Kravspecifikation	5
2.3.1	Funktionelle krav	5
2.3.2	Non-funktionelle krav	6
2.3.3	Eksklusion	6
3	Analyse	6
3.1	Tidligere Afprøvning af Mechanical Turk	6
3.2	System	7
3.2.1	Mobil Enhed	7
3.2.2	Indendørs Navigation Metode	7
3.2.3	Lagring af Data	7
3.2.4	Analyse KlasseDiagram	7
3.2.5	System beskrivelse	8
3.3	Beacon Navigation	8
3.3.1	Placering	8
3.3.2	Lokalisering	9
3.3.3	Friis ligning	9
3.3.4	Dæmpningskilder og dæmpningskoeficient	9
3.3.5	Navigations algoritme	10
3.3.6	Alternative løsningsmuligheder	10
3.4	Business Plan	10
3.4.1	Business Model Canvas	10
3.4.2	Customer Segments	11
3.4.3	Value Propositions	11
3.4.4	Key Activities	11
3.4.5	Key Ressources	12
3.4.6	Key Partners	12
3.4.7	Customer Relationships	12
3.4.8	Channels	12
3.4.9	Cost Structure	12
3.4.10	Revenue Streams	12
3.5	Estimeret omkostninger	13
3.5.1	Bluetooth	13
3.5.2	Estimote Proximity beacons	14
3.5.3	Mazemap	14
3.5.4	Firebase	14
3.6	UserExperience	14
3.6.1	User stories	14
3.6.2	Use case model	15
3.6.3	Generelle overvejelse	15
3.6.4	UI	15
4	Design	16
4.1	ModelViewViewModel	16
4.1.1	Vores MVVM	17
4.2	Beacon Lokalisering	17
4.2.1	Estimote SDK	17
4.2.2	Firebase Database	17
4.3	Mazemap API	18
4.3.1	Problem	18

4.3.2	Alternative løsninger	18
4.3.3	Vores løsning	19
4.3.4	Fremtidsperspektiv	19
4.4	Diagrammer over hele systemet	19
4.5	UI	20
4.5.1	Navigationside	20
4.6	BeaconMap	21
4.6.1	Udvalg af beacons	21
4.6.2	Proximity Beacon	22
4.6.3	Plantegning	22
5	Implementation	23
5.1	Beacon-Cloud'-Credentials	23
5.2	Beacon opsætning	24
5.3	Proximity-ZoneEventViewModel	25
5.4	Notifikation	25
5.5	FireStore - Firebase	26
5.6	MazeMap	26
5.6.1	UI	26
5.7	Webview	26
6	Afprøvning	27
6.1	Beacon Signal afprøvning	27
6.1.1	Applikation til måling af signalstyrke	27
6.1.2	Friis Ligning verifikation	27
6.1.3	Lokaliseringens afprøvning	30
6.2	Usability Test	32
6.2.1	Brugernes oplevelser	32
6.2.2	Observerede resultater	33
6.3	Effekt Afprøvning	33
6.3.1	Resultater fra effekt Afprøvningen	33
7	Diskussion	34
7.1	Estimeret gevinst	34
7.1.1	Besparelse pr. hospital	34
7.1.2	Besparelse på nationalt plan	35
7.1.3	Mulig internationalisering	35
7.2	Bæredygtighed	35
7.2.1	Sociale aspekter	35
7.2.2	Miljømæssige aspekter	36
7.3	Etiske overvejelser	36
7.4	Usability test	37
7.4.1	Valg af brugere	37
7.4.2	Resultat af test	37
7.5	Effekt afprøvning	37
7.5.1	Kvalitative observationer	38
7.5.2	7 ud af 10 foretrak at bruge appen	38
7.6	Arbejdsbeskrivelse	38
7.6.1	13-ugers	38
7.6.2	3-ugers	38
8	Konklusion	39
9	ByrdeFordeling	42
9.1	Tidsforbrug ark	45
9.2	Instagant Ark	45

1 Formål

I denne rapport vil vi løse navigationsudfordringer på hospitaler. Dette vil vi gøre ved brug af Ibeacons, Bluetooth lokalisering og et interaktivt kort. Derfor bliver vi nødt til at designe en applikation, der kan kommunikere med allerede placerede Ibeacons, og bruge denne Information til at vise lokation på et interaktivt kort, hvorefter vi så vil teste, om applikationen rent faktisk lever op til de forventninger som vi har, både teknisk men også ift. brugernes forventning til funktionalitet. Yderligere skal vi vurdere hvordan vores løsning skal sælges, samt tage forbehold for etiske og bæredygtige overvejelser. Udover dette vil vi dokumentere vores proces undervejs i form af et Gantt-diagram, en tidsplan, samt en generel beskrivelse af hvordan projektarbejdet blev struktureret og udført.

2 Indledning

I dette afsnit vil vi gennemgå baggrunden for vores rapport samt vores udarbejdede problemformulering og kravspecifikation.

2.1 Baggrund

Hospitaler er store komplekse bygninger, der kun bliver større og større. Derfor opstår der udfordringer med at finde vej og der vil skabes flere og flere udfordringer i den forbindelse på hospitalerne i fremtiden, når bygningerne bliver større og større.

2.1.1 Komplekse hospitaler

Udfordringen ved intern navigation på hospitaler, bunder kort sagt i, at det kan være besværligt at finde rundt på hospitaler, hvilket naturligvis er noget der kan gøres nemmere. Besværligheden kunne bla. skyldes størrelserne på sygehusene og kompleksiteten af hospitalerne. Der er en stor konsensus omkring gyldigheden af udfordringen om at kunne finde rundt, som har resulteret i at problematikken er blevet belyst, hvilket kan ses i citaterne nedenfor:

”Vi kan se, at hvis bare man skal gå tre skridt ned ad en gang, kan man blive i tvivl om, hvor man skal hen.” Kristine Strange, Hospitalsenheden.[1]

”Patienterne bliver usikre og enormt stressede over, at de ikke kan finde vej.” Janne Laursen, Sygeplejerske.[1] Det er dog helt forståeligt, at patienten og de pårørende bliver stressede af at navigere rundt på hospitalet, da de befinner sig i en uforudsigtelig situation. Denne uforudsigtelighed er nemlig en af de store stressorer, som skaber stress.[2] På grund af dette, bliver sygeplejerskerne dagligt nød til at vise vej til patienter og pårørende [1], hvilket både kan resulterer i en stresset hverdag for sygeplejerskerne, men også en økonomisk byrde ift. at sygeplejerskerne bruger tid på at vise vej, i stedet for at varetage deres kerneopgaver. Problematikken med komplekse hospitaler kan især ses, når der åbnes nye store sygehuse, som sygehuset i Gødstrup. Sygehuset indeholder nemlig 4600 forskellige rum fordelt på 148.000 kvadratmeter.

3 uger inden åbningen af hospitalet, var 100 patientstøttere fra Røde Kors i gang med at øve sig i at finde rundt, så de bla. kan assistere i at finde vej ved hospitalets åbning. Dette forelægger netop problematikken ift. af finde rundt på hospitaler, siden der er et stort behov for patientstøtterne, som jo netop hjælper til med vejvisning [3]. Denne udfordring kan yderligere understøttes vha. følgende citater:

”Jeg tror, det kan være overvældende at komme ind i forhallen her på sygehuset, så jeg håber, jeg kan guide folk, så de kan føle sig trygge”. Lone Boye, frivillig fra Røde Kors. [3]

”Jeg har oplevet, at når man kommer ind på et helt nyt og fremmed sted, så er det meget svært at finde vej. Derfor har jeg også fået mange positive reaktioner på, at vi står og hjælper på vej”. Sinikka Kangas Larsen, med til at rekrutterer vejviserne. [3]

Ud fra disse citationer kan man klart fornemme, at der er snakke om reelle udfordringer samt, at når man forsøger at løse udfordringen med at finde rundt, bliver patientstøtterne mødt med positiv feedback. Dette er netop med til at vise, at der er mange, som vil sætte pris på en ny og optimeret løsning, der kan gøre navigation på hospitalerne mere håndgribeligt og nemmere. Så ud fra alt dette, kan man tydeligt fornemme en problematik og udfordring ift. at finde vej på hospitaler. Det er så denne udfordring vi vil forsøge at tackle i vores projekt, og gøre det nemmere for folk at finde rundt på hospitaler. Desuden er der også en økonomisk byrde, som blev nævnt kort tidligere bl.a. sygeplejerskes tidsforbrug. Denne økonomiske byrde vil konkretiseres i følgende afsnit.

2.1.2 Økonomisk byrde

Det er svært at definere den direkte økonomiske byrde, som resultat af mangel på effektiv navigation koster. Man kan dog alligevel lave nogle kvalificerede skøn. Et studie på Emory Universitets Hospital estimerede, at personalet på regionale hospitaler bruger 4500 timer om året på at anvise vej.[4] Dette tal er dog meget lavt i forhold til en intern undersøgelse lavet på Melbourne hospital, hvor de vurderede, at personalet brugte 2,5% af deres tid på at anvise vej eller eskorterer folk rundt.[5] Dette billede ses også i Danmark, som forklaret i afsnit *2.1 Komplekse hospitaler*. Da man ikke kan finde den direkte økonomiske byrde, må vi bruge de fornævnt estimerede værdier. På baggrund af de fornævnte værdier og informationen om, at Danmark har omkring 106.000 fuldtidsbeskæftigede i sundhedsvæsenet[6], som er fordelt på over 50 sygehuse [7]. Så bliver der brugt et estimeret time antal på mellem 225.000 (1) til 424.000 (2) timer til at vejfinde.

$$50(\text{AntalSygehuse}) \cdot 4500(\text{TimerPrHospital}) = 225000(\text{NedreGrænse}) \quad (1)$$

$$2.5\% \cdot 160(\text{Fuldtidtimer}) \cdot 106.000(\text{AntalFuldtidsbeskæftigede}) = 424.000(\text{ØvreGrænse}) \quad (2)$$

Denne estimerede timeantal kan yderligere omregnes løst til penge vha. den gennemsnitlige timeløn for sundhedspersonalet i 2013 på 243 kr. i timen[8].

$$225.000 \cdot 243 = 54.675.000kr. \quad Til \quad 424.000 \cdot 243 = 103.032.000kr. \quad (3)$$

,

2.2 Problemformulering

Da hospitaler er store og komplekse, opstår der et stort behov for navigation indendørs. Dog fordi hospitaler normalt er store betonbygninger med masser af metal og andre radiobølgedæmpende materialer, er det svært at bruge GPS som navigation.

Da borgere bliver sat i en uforudsigelig situation, har borgeren svært ved at finde rundt på hospitalerne, og skal derfor ofte spørge om vej. På denne måde resulteres der ofte i spildtid for personalet og forskydning af aftaler. Derfor vil en løsning være til gavn for både personalet men også den almene borgere. Problemformuleringen for projekt er derfor formuleret således:

Hvordan kan man implementerer et indendørs navigationssystem i miljøer, hvor gps signaler ikke er en mulighed, som både er brugervenlig, hurtigt og sparar tid for personalet og borgeren?

2.3 Kravspecifikation

Ud fra vores baggrund og vores projektbeskrivelse har vi udformet de krav, vi forventer at udføre i løbet af projektperioden, samt de krav, vi ikke forventer at udføre i løbet af projektperioden. Kravene er delt op i funktionelle og non-funktionelle krav. Disse er yderligere delt op i nice-to og need-to.

2.3.1 Funktionelle krav

De funktionelle krav dækker over de krav, der stilles til funktionen af systemet fra slutbrugerens synsvinkel.

Need-to

Dækker over de krav der forventes at være udført i slutningen af projektperioden, altså minimum viable product (MVP). Need-to kravene for de funktionelle krav er som følge:

- FK1: Have et interaktivt kort, der følger brugerens egen lokation.
- FK2: Søgefunktion, hvor brugeren kan indtaste den ønskede destination.
- FK3: Rutevejledningen skal vise den hurtigste vej til destinationen.
- FK4: Kunne afbryde rutevejledning

Nice-to

Dækker over de krav der kan dækkes, når need-to kravene er udført, som vil resultere i yderligere funktionalitet i systemet.

- FK5: Bekræftelse på ankomst til endt destination
- FK6: Kunne indstille handicapvenlige ruter

2.3.2 Non-funktionelle krav

De non-funktionelle krav er infrastrukturen af systemet samt kvalitetsikringen af systemet.

Need-to-have

Dækker over de krav der forventes at være udført i slutning af projektperioden.

- NFK1: Anvende Ibeacons til det interaktive kort.
- NFK2: Benytte android mobile enheder
- NFK3: Anvende back end (Estimote Cloud, Firebase).
- NFK4: Reducere behovet for at spørge om vej.
- NFK5: Reducere tiden brugt på at finde vej.
- NFK6: God brugeroplevelse

Nice-to-have

Dækker over de krav der kan dækkes, når need-to kravene er udført.

- NFK7: Introduktion til brug og installering af applikationen.

2.3.3 Eksklusion

Eksklusion dækker over de krav der enten på forhånd eller undervejs i projektet er vurderet udenfor projektet scope, eller at kravet i løbet af projektperioden er vurderet til at tage for mange arbejdstimer i forhold til de fordele det ville have ved implementeringen. De ekskluderede krav kan derfor perspektivere til andre projektarbejder eller ved senere arbejde med systemet, hvis der ønskes en udvidelse af projektet, eller ved udarbejdelse af et helt nyt system.

- EK1: Crowdcontrol, så administrator kan indføre alternative ruter om kritiske områder.
- EK2: Logning af trafik

3 Analyse

I dette afsnit vil vi analysere hvorvidt der er et behov for vores løsning ved at udføre en mechanical turk, og analysere vores potentielle brugergruppe og deres behov i user stories. Yderligere vil vi analysere hvilke komponenter vores løsning skal bestå af på baggrund af kravsspecifikationerne og baggrunden samt undersøge beacon navigation som en helhed.

3.1 Tidlige Afprøvning af Mechanical Turk

For at analysere om vores produkt har relevans og kunne validere brugbarheden af det. Har vi udført en mechanical turk test. Det vil sige at vi har lavet en brugergrænseflade uden funktionalitet og afprøvet den hvor et menneske forestiller sig funktionaliteten. Helt konkret skulle brugeren finde en lokation med hjælp fra en prototype af applikationen og finde en lokation uden. Lokationen var DTU Lyngby campus, mere konkret 1. og 4. kvadrat - Et område der med god vilje kan minde om et hospital i forhold til størrelse og kompleksitet.

Først udførte vi testen uden hjælp fra applikationen, her gjorde testpersonen sig brug af et kort ved bygning-sområdet samt ved indgangen til bygningen, og prøvede derefter at finde vej ud fra intuition og hukommelse af kortene. Testpersonen fandt lokalet uden at gå forkert, men havde undervejs ofte brug for at stoppe op og huske den imiterede rute.

Derefter udførte vi test med hjælp fra applikationen. Brugeren forstod hurtigt hvordan applikationen virkede og navigerede hurtigt frem til destination blot ved at kigge på applikationen efter behov. Testpersonen gik direkte til destinationen, kun med en kort opbremsning ved en trappe.

På baggrund af de to test fik vi bekræftet at forsøgspersonen vil kunne bruge vores løsning effektivt og til dens tiltænkte formål. Yderligere rapporterede brugeren en mere behagelig oplevelse med mindre bekymring da hun fandt vej med applikationen.

Efter de to udførte test spurgte vi testpersonen om hendes oplevelse og kom frem til følgende erfaringer:

- Det skal tiltænkes i designet, at brugeren har intuition for vejvisning og kan både bruge applikationen delvist eller fuldkomment, i sin vejvisning. Altså brugeren kan tage mobilen op efter behov eller have sin mobil i hånden og bruge den hele vejen.

- Områder der ikke er angivet med et lokalenavn eller nummer, kan være svære at få bekræftet om det er det korrekte sted.
- Områder der kan ligne hinanden, skal det gøres tydeligere om man er det rigtige sted og ikke et andet område, der minder meget om.

3.2 System

På baggrund af de funktionelle og non-funktionelle krav stillet i kravsspecifikations afsnit 2.3 vil vi nærmere analysere hvad vores system skal bestå af. Nogle væsentlige delelementer af systemet gennemgås derfor nu.

3.2.1 Mobil Enhed

Da vores løsning skal kunne køre på en android mobil enhed ift. NFK2, kan applikationen udarbejdes i Kotlin med Jetpack Compose. Jetpack compose er androids toolkit til at lave native UI. Dette betyder dog også at vores applikation ikke vil virke til Apple devices. Jetpack compose bruger 'Model-View-ViewModel '-strukturen og vores applikation skal kunne indgå indenfor denne.

3.2.2 Indendørs Navigation Metode

Når vi kigger på indendørs navigation og ikke kan bruge GPS, er det nødvendigt at bruge bluetooth til at lokalisere og finde vej inde i et hospital. Til denne problematik kan man bruge Beacons og IBeacon protokollen. IBeacon er en protokol til at kommunikere mellem smartphones, tablets og andre devices når man er i nærheden af en beacon. Dvs. at parameteren beacons bruges til er proximity, så man kan se hvilken beacon man er tættest på og ud fra det vurdere ens umiddelbare lokation. Kort sagt er beacons nogle hardware-transmitters, lavet til IBeacon protokollen. Beacons er en type bluetooth lav energi(BLE) device, der broadcaster deres identificer til devices i nærheden.

Til at interagere med disse beacons eksisterer der en række forskellige Estimote API'er, eksempelvis Estimote Proximity SDK til applikationsudvikling og Estimote Cloud til opbevaring af Credentials til authentication af Beacons.

3.2.3 Lagring af Data

For at overholde best practice principper og ”separation of concerns”, kunne man opfylde de funktionelle krav ved fx at bruge en cloud-baseret database. Der eksisterer mange forskellige sprog, applikationer og cloud services til at håndtere ens databaser. Først og fremmest adskiller man i SQL og No-SQL, da SQL er det mest populære sprog til at håndtere relationelle databaser i. Dernæst skal databasen eksistere et sted, hvor den skal være tilgængelig gennem internettet. Dette kan opnås ved at oprette en server der indeholder databasen og en tilhørende API til at interagere med databasen eller at interaktionen kan opnås ved at bruge en cloud-service, der administrerer databasen og stiller en API til rådighed.

3.2.4 Analyse KlasseDiagram

For at analysere og anskueliggøre det overordnede domæne har vi udarbejdet følgende analyse klassediagram:

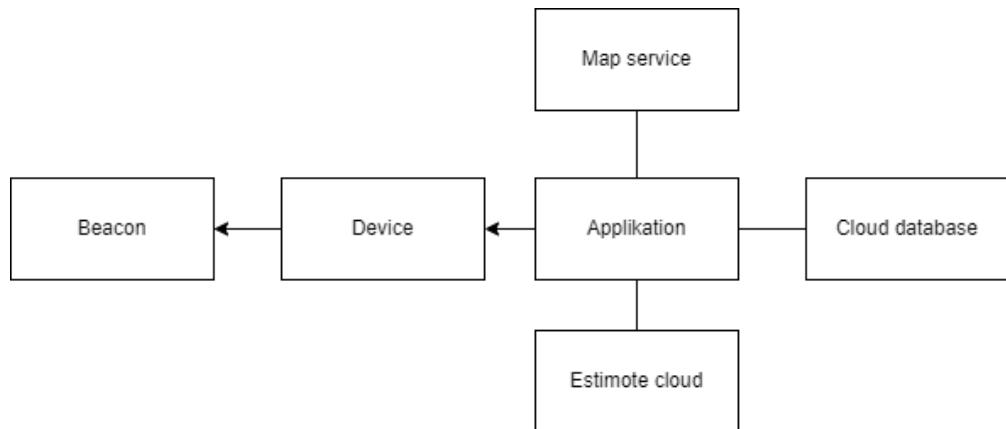


Figure 1: Analyse klasse diagram

3.2.5 System beskrivelse

For at nærmere analyserer systemets opbygning har vi udarbejdet en systembeskrivelse, hvor systemets sammenhæng og arkitektur fremgår.

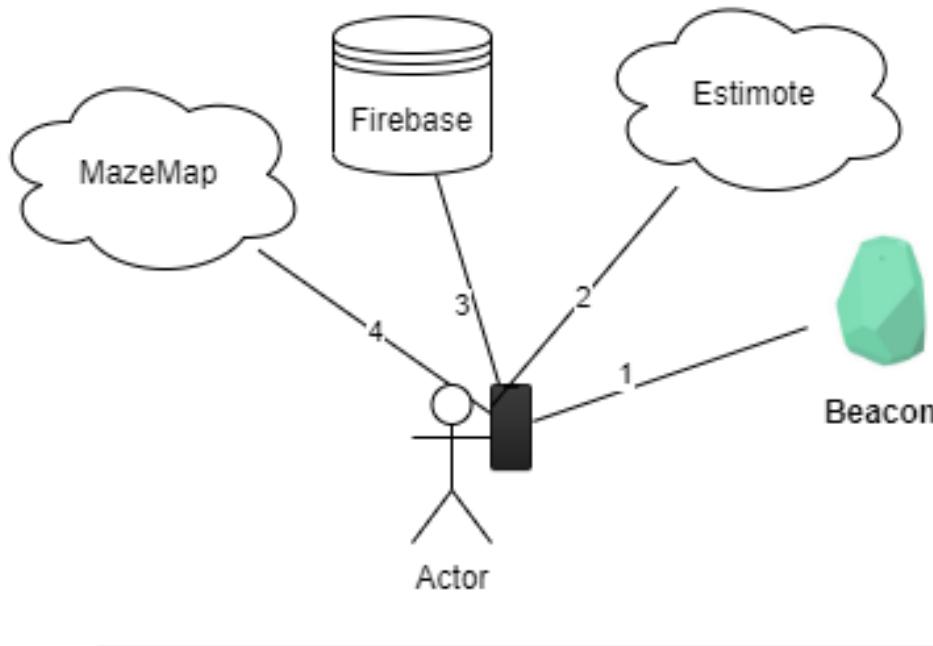


Figure 2: System beskrivelse

På figur: 2 er der vist en bruger der gennem en telefon interagerer med fire elementer.

- Nr. 1 En Beacon transmittere sin nøgle til systemet.
- Nr. 2 Systemet interagerer med Estimotes API og udveksler nøglen for et specifikt ID.
- Nr. 3 Systemet bytter ID'et for lokationsdata gennem et Query til firebase.
- Nr. 4 Systemet bruger MazeMaps API til at opdatere brugerens nye lokation.

3.3 Beacon Navigation

Når man skal lave beacon navigation bliver man nødt til at overveje følgende:

- Placeringen.
- Lokalisering.
- Friis ligning
- Dæmpningskilder og dæmpningskoeficient
- Navigationens algoritme
- Alternative løsningsmuligheder

Grunden til at følgende liste med eksempler skal overvejes, er fordi at placeringen og dæmpningen mm. påvirker brugeroplevelsen markant og i sidste ende betyder, om der er overensstemmelse med personens egen placering i bygningen og om det er reflekteret på den mobile enhed.

3.3.1 Placering

Når det kommer til placering af beacons er der en række overvejelser vi kan lave. Heriblandt proximity og eventuelle overlap mellem beacons rækkevidde. I designafsnit 4.6.3 om plantegning, udvikler vi et mere konkret beacon map, hvor vi kommer nærmere om hvor vores beacons kunne være placeret og hvilket område de repræsenterer. Det essentielle ved vores beacons er nemlig deres proximity, altså den radius hvor signalet kan opfanges. På baggrund

af størrelsen af disse proximity zoner, kan man nemlig placere dem, således at overlappet mellem zoner giver mening. Med dette menes der, at der ikke er alt for store overlap mellem vores beacons zoner, så at der ikke forekommer tvivl om hvilken zone man befinder sig i. Men samtidig også at der ikke er for store tomrum mellem vores beacon zoner, idet at det kan være misvisende hvis man står i et tomrum men kortets blue dot ikke er opdateret ift det. Derudover er det også relevant at overveje hvor vores beacons skal placeres ift lokalene. Skal de placeres inde- eller udenfor lokalet? Hvis inde i lokalet, skal de så placeres i midten af rummet eller ved døren af lokalet? Skal der også være beacons i gangene eller kun lokalene? Disse overvejelser kan jo have en betydning for user experience, ift. at hvis man ankommer til sin destination og der er en beacon i selve lokalet, så kan man nemlig direkte se at man står i lokalet. Hvis der ikke er en beacon i lokalet, så sker det netop ikke. Dog hvis man vælger både at have beacons i hvert lokale og i gangene, så skal man bruge flere beacons. På samme måde ville det kræve flere beacons hvis vi vil implementere en mere avanceret lokaliserings algoritme, da udregning ville være mere eksakt med afstande til flere beacons. At man bruger flere beacons, har jo en effekt på bæredygtighed, så ift. bæredygtighed vil man jo forsøge at bruge så få beacons som muligt. Alle disse overvejelser er noget, man kan reflektere over når det kommer til at være placering af ens beacons, hvilket gøres konkret i punkt (4.6.3).

3.3.2 Lokalisering

Lokalisering handler om at bestemme hvor noget befinner sig. I vores tilfælde kunne det handle om hvor en borger med sin telefon er placeret. Dette kan man finde ud af vha. en lokaliserings teknik. Eksempler på hvad man kan bruge til at lokalisere ultralyd, triangulær interpolation og triangulær ekstrapolation. Triangulær lokaliserering vil være den foretrukne metode til at udregne den konkrete placering af en mobil enhed i et rum. Her kender man afstanden fra den mobile enhed til placeringen af minimum tre beacons. Afstanden kan man udregne på baggrund af dBm, men kræver at man under implementering af systemet laver en grundig test og synkronisering mellem afstand og dBm. Når man kender afstand og placering af beacons, kan man opstille matricer og udregne mobilenhedens placering. Dog vil der stadig være en mængde af usikkerhed ift placering, men denne vil man reducere ned under en acceptabel størrelse.

3.3.3 Friis ligning

Bluetooth er en industriel radio-kommunikationsprotokol, dette betyder at bluetooth udgiver sit signal via radiobølger i 2.4 Ghz frekvensbåndet. Når man snakker om radiokommunikation bliver man nødt til tale om Friis ligning:

$$P_R = P_T \cdot \frac{G_T \cdot G_R \cdot \lambda^2}{(4 \cdot \pi)^2 \cdot R^2 \cdot L} \quad (4)$$

Denne ligning beskriver hvordan det modtagende signal hænger sammen med det afsendte signal samt tab ud fra en referenceafstand. Grunden til at der er et tab, er fordi vores beacon/antenne udsender sit signal i en kuppel. Denne kuppeludbredelse gør at man ikke får en linær aftagning af signalet, men derimod en aftagning der hænger sammen med den Invers-kvadratiske lov. Dette kan blive et problem i vores lokaliserings af brugeren, da jo længere væk man bevæger sig væk fra antennen/beacon, jo mindre signal modtager man og jo større bliver usikkerheden i lokaliseringen af brugeren. Derudover findes der også dæmpningskilder.

3.3.4 Dæmpningskilder og dæmpningskoefficient

Dæmpningskilder er kilder der dæmper signalet og på denne måde gør endnu mindre af signalet kommer frem til brugeren. Denne dæmpningsfaktor kan blive regnet med når man vil udregne det konkrete udbredelsestab.

$$PL(d)_{dB} = PL(d)_{dB} + 10n_{MF} \cdot \log_{10}\left(\frac{d}{d_0}\right) + \sum PAF_{dB} \quad (5)$$

Dæmpningsfaktorerne kan her ses som en dæmpningskoefficient og en summation af path attenuation faktorer i dB. Vi kan udregne vores dæmpningskoefficient ved at omskrive ligning 4 således:

$$\frac{P_R}{P_T} = K(d) \quad (6)$$

Dette kan vi gøre, da vi ikke behøver at tage forbehold for gains osv. Da vi bare kan måle et modtaget og afsendt signal, og deraf udregne vores dæmpningskoefficient. Dette hænger sammen med vurderingen, af at vi ikke har mulighed for at lave en støj og dæmpningskoefficient for alle de enkelte beacons, så derfor afprøver vi vores beacons signalstyrke imod afstand. Vi burde få et signal der mindede om denne kilde, dog med langt lavere decibel.[9] Dette forhold mellem PAF og DB kan ses på figur 3

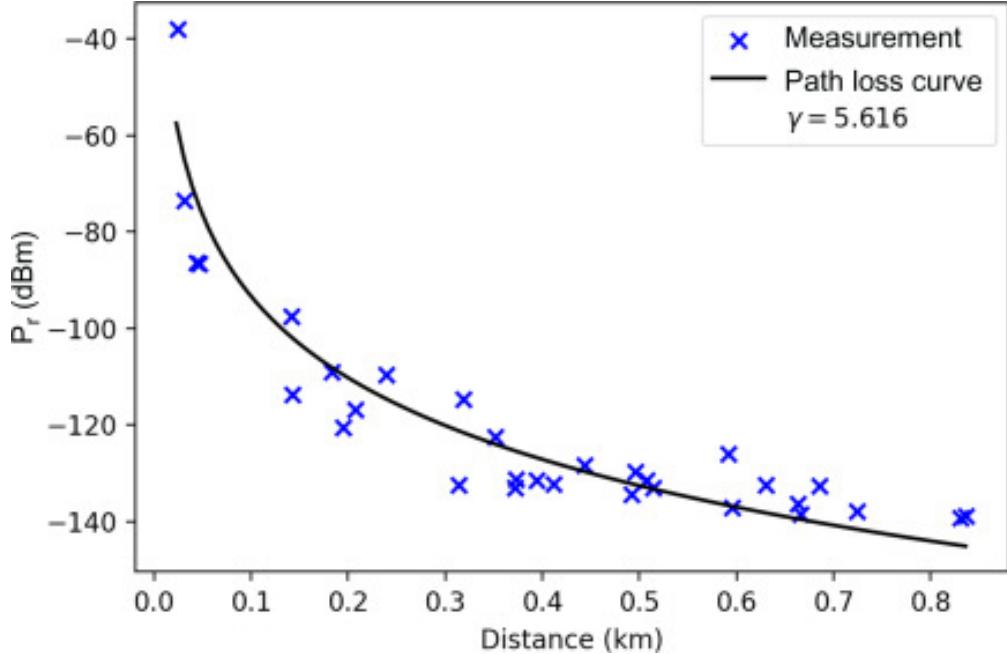


Figure 3: Path attenuation afstand [10]

3.3.5 Navigations algoritme

Når man laver en app der kan vise hvor brugeren befinder sig, er det også vigtigt, at lave et navigations algoritme lag. Denne navigationsalgoritme er en algoritme, der skal kende alle veje der kan gås på. Dette er vigtigt, så at algoritmen ikke bare anbefaler den hurtigste rute, hvor der står man skal gå igennem en væg. Så hvis algoritmen indeholder alle gyldige veje, så angives navigationsruten kun ift. dette.

Derudover er der mange forskellige ruter man kan tage fra et vilkårligt lokale A til et andet vilkårligt lokale B, derfor skal algoritmen ikke bare være i stand til at finde en rute, men også at finde den hurtigste rute. Yderligere skal der også være en algoritmetik der sørger for altid at holde ens bluedot placering i bevægelse. Dette kan selvfølgelig ikke gøres med beacon placeringer, da man så ville være nødt til at have en ny beacon omkring hver meter. I stedet kan man dog tage de fem nærmeste beacons og ud fra deres forskellige signalstyrke udregne en vektor. Gør vi dette nok gange, kan vi få en gennemsnitlig hastighed og retning man bevæger sig i. At designe sin egen velfungerende navigationsalgoritme er dog ikke en simpel opgave, så pga. omfanget af projektet ville dette ikke give mening at vi selv udvikler.

3.3.6 Alternative løsningsmuligheder

Som nævnt i et tidligere delafsnit (Navigations algoritme), så er det at lave en navigationsalgortime ikke en simpel opgave, som vi vil tage hul på i dette projekt. Derfor må vi finde en alternativ løsningsmulighed. Dvs. vi skal finde en allerede eksisterende løsning, som netop muliggør selve navigationsdelen for os. Dette kunne muligvis være Mazemap, Mapwize, Google Maps eller lignende. Når vi så har valgt en af disse løsninger, kan vi så bygge vores applikation oven i det. Det vil sige, at vi stadig selv implementere software også, som kommer til at fremgå i implementationsafsnittet.

3.4 Business Plan

For at drive en forretning, som giver økonomisk mening, er det relevant, at vi har en konkret forretningsplan, hvori vi specificerer de delelementer der udgør vores forretningsplan.

3.4.1 Business Model Canvas

I forbindelse med vores business plan har vi udviklet en Business Model Canvas, hvori vi efter diverse iterationer er kommet frem til følgende BMC, som kan ses på figur: 4 nedenfor. Nu gennemgås de relevante elementer der indgår i forretningsplanen og overvejelser deromkring.

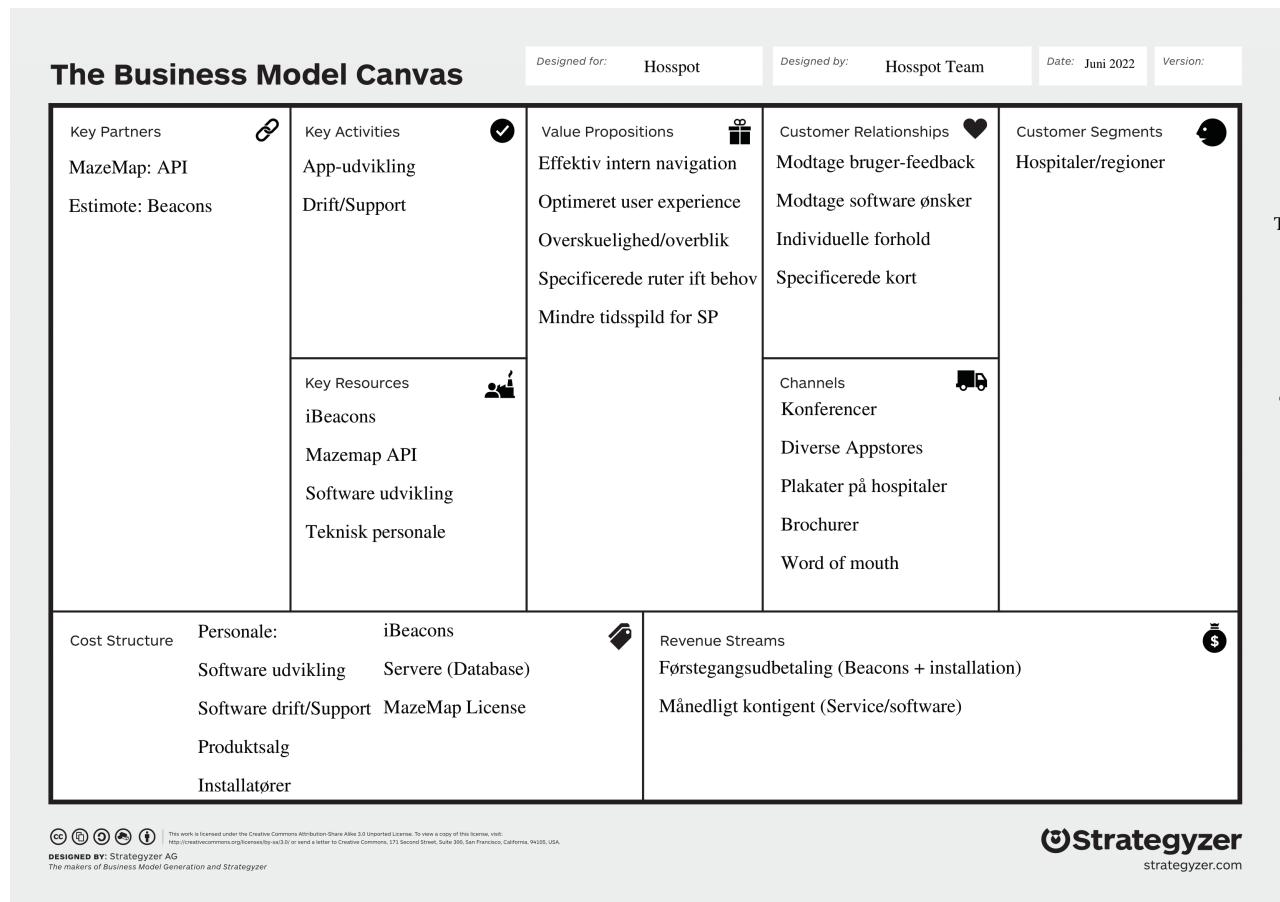


Figure 4: Business Model Canvas

3.4.2 Customer Segments

Vores produkt sælges som en pakkeløsning til hospitaler, dvs. at vores eneste kundesegment er hospitaler. Det kunne umiddelbart lyde som et begrænset marked, men som beskrevet i afsnit 2.1.2 om økonomisk byrde, er det en meget kostbar problematik vi forsøger at løse. Så på grund af dette, mener vi ikke at det er et problem, at vi udelukkende har et kundesegment.

3.4.3 Value Propositions

Det som vi egentlig sælger til hospitalerne, er vores Value propositions, som er den værdi vi skaber vha. vores løsning. Vi har udført en Mechanical Turk og kommer også til at udføre en effektvurdering, hvorudfra vi regner med at kunne se at løsningen effektiviserer den interne navigation på hospitalerne samt øger brugeroplevelsen. Bliver dette aktuelt, er disse fordele netop en del af vores value propositions. Disse value propositions opnår vi bla. ved at give et overblik over hospitalet til slutbrugerne vha. vores app, som hjælper dem i at navigere sig rundt på hospitalerne uden at være afhængige af assistance fra sundhedspersonalet. Da vi sælger vores løsning til hospitalerne, skal vi også levere noget udover, at forbedrer brugeroplevelsen for deres patienter. Dette gør vi ved at minimere den tid sundhedspersonalet bruger på at hjælpe borgere med at finde vej, hvilket kan resultere i at sundhedspersonalet kan bruge deres tid mere effektivt. På denne måde leverer vi også økonomiske fordele, således at det kan betale sig for hospitalerne at købe vores produkt.

3.4.4 Key Activities

For at realisere dette, har vi fokus på nogle konkrete nøgleaktiviteter. Heriblandt løbende App-udvikling, så vi kan leve op til slutbrugernes ønsker, markedets behov og optimere user experience samtidigt. Dog kræver dette også drift og support. Drift da vi skal sørge for den eksisterende løsning fungerer som ønsket bla. ift. databaser og software-opdateringer. Til sidst er der teknisk support ved fx nedbrud eller almen support ved fejl.

3.4.5 Key Ressources

Udførelsen af disse aktiviteter afhænger af nogle ressourcer. iBeacons bruges som noget af det mest essentielle ved vores løsning, idet at det netop er Beacons der gør den interne navigation muligt. Udover dette er der også MazeMaps Api, som er fundamentet for vores digitale kort. Derudover som tidligere nævnt, er softwareudvikling samt et veluddannet teknisk personale også altafgørende.

3.4.6 Key Partners

Vi kan ikke selvstændigt levere vores løsning, da der er visse konkrete ting som vi ikke tilbyder fx at vi ikke har vores egen navigations API. Det er netop her vores nøglepartnere kommer ind i billedet, så at vi vha. en kollaboration med andre virksomheder kan udfylde de mangler vi har. MazeMap leverer en API til os, som vi baserer vores applikation på ift kort og navigation, så det er en meget essentiel nøglepartner. En anden meget essentiel nøglepartner Estimote, som netop står for produktionen for de proximity beacons som vi gør brug af i hospitalerne.

3.4.7 Customer Relationships

For at forbedre kundeoplevelsen og optimere vores løsning går vi meget op i vores kunderelationer. Heriblandt under selve driften og den løbende support, som blev beskrevet tidligere. Et andet essentielt element er det individuelle forhold vi har med hvert hospital samt. det specificerede kort vi udvikler separat til hvert hospital. Dette gør nemlig at vi har en individuel relation med hver kunde, idet der skal en eller flere personer ud og tale med dem samtid. få lavet et beacon map, og desuden holde løbende drift. Derudover opfordres brugerne til at levere feedback samt eventuelle software-ønsker, hvilket både styrker kunderelationerne og bidrager til en videreudvikling af vores løsning. Denne bruger-feedback ville kunne resultere i en tosidig gevinst, da vi modtager konkret feedback som vi kan bruge til at forbedre vores produkt, og derved optimere applikationen som slutbrugeren gør brug af.

3.4.8 Channels

Måden vi når ud til slutbrugerne, er ved at være bredt tilgængelige og sørge for, at applikationen kan findes i diverse app stores. Udfordringen ift. at nå ud til hospitalerne, kan løses med præsentationer ved relevante konferencer, som især noget vi har tænkt os at gøre brug af for at kunne fremvise vores løsning til potentielle kunder. Vi vil derudover have plakater med QR-koder hængende på hospitaler, der beskriver og reklamerer for applikationen samtidig med at vi også laver brochures. På baggrund af den positive feedback vi har modtaget i vores test, forventer vi derudover også at "word of mouth" spreder nyheden om vores app rundt i de rigtige cirkler. Det interessante ved vores channels, er at vores kunde egentlig er hospitalerne, så det er dem vi skal skabe interessen i at opkøbe vores løsning, men hvis vi kan dokumentere at applikationen er meget populær blandt slutbrugere på andre hospitaler, kan dette også bidrage. Derfor er det også relevant at nå ud til selve slutbrugerne ude på hospitalerne, selvom det ikke er dem vi sælger produktet til. Det er netop det, som der opnås vha. de tidlige nævnte teknikker fx med plakater og brochurer.

3.4.9 Cost Structure

At have denne virksomhed kørende er dog ikke gratis, så der er naturligvis nogle omkostninger. Omkostningerne udgøres af personale til diverse opgaver, opkøb af beacons, licens til MazeMaps Api samt Bluetooth og adgang til en cloud-service til datalagring. Personaleomkostninger opstår under udviklingen af selve applikationen, den løbende drift, support af applikationen, salg af produktet og til sidst at få installeret produktet i form af placering og plantegninger af beacons.

3.4.10 Revenue Streams

Disse cost structures finansieres af vores indtægtskilder der udgøres af en førstegangsdubtaling og et månedligt kontingent. Førstegangsdubtalingen dækker prisen på selve beacons, udvikling af beacon map til det specifikke hospital samt installationen af beacons ude på hospitalerne. Med udvikling af beacon map til det specifikke hospital, menes der, at de individuelle beacons geografiske lokationer på hospitalet placeres på baggrund af hospitalets plantegning. Dog er det ikke bare nok at betale førstegangsdubtalingen og få installeret beacons, og derfor kommer det månedlige kontingent ind i billedet. Det månedlige kontingent bruges til at finansiere vores service. Dette vil sige, at den dækker adgang til vores applikation, løbende support, drift og opdateringer. Adgang til vores applikation skal forstås som, at hospitalet fremgår i applikationen.

3.5 Estimeret omkostninger

For at komme nærmere nogle af omkostninger til produktet i tal, er det vigtigt allerede nu at kunne sætte en estimeret pris på de omkostninger der er i forbindelse med slutproduktet. Det er rigtig svært i en tidlig fase at afgøre de præcise omkostninger til et slutprodukt, men her er der angivet priser der er nogenlunde realistiske. Udgifter til et antal årsværk med en pris det koster at udvikle et produkt er et eksempel, hvor det er urealistisk at komme med estimat.

3.5.1 Bluetooth

Da vi bruger Bluetooth som kommunikation, kræver det en kvalifikation af produktet. For at kunne kvalificere vores produkt, er der fees i den forbindelse. Se figur 5 for fees.[11]

Fees	Explanation	Associate Member	Adopter Member
Declaration Fees			
Declaration Fee	A Declaration fee must be paid for any new, changed, used or branded Bluetooth product	\$4,800	\$9,600
GATT-based Profile Client (app)	Applications that support the client role only and implement one or more Bluetooth SIG adopted GATT-based Profiles are qualified as a Profile Subsystem product type.	\$120	\$120
Recognition Fees			
Bluetooth Qualification Test Facility (BQTF)	A Bluetooth Qualification Test Facility (BQTF) is recognized by the Bluetooth SIG as competent to perform "Category A at or below the HCI layer" qualification test cases identified in the Test Case Reference List (TCRL).	\$9,600 Annually	\$9,600 Annually
Bluetooth Recognized Test Facility (BRTF)	A Bluetooth Recognized Test Facility (BRTF) is recognized by the Bluetooth SIG to perform testing only on its own behalf.	\$2,400 Annually	\$2,400 Annually
Profile Tuning Suite (PTS) Fees			
PTS Runtime Cost	The Profile Tuning Suite (PTS) is a specification-based testing tool that automates protocol and profile interoperability testing.	Free	Free
PTS Radio Module	The PTS radio module is required for full testing of a physical device.	\$99	\$99

Figure 5: Bluetooth qualification fees

På figur 5 ses en række fees, som kan regnes op i førstegangsudgifter, løbende årlige udgifter og udgifter i forbindelse med ændring af produktet. Udgifterne kan se i følgende figur 7.

Fees	Førstegangsydelse	Løbende årlige udgifter	Udgifter i forbindelse med ændringer
Declaration fee	9600\$	0	9600\$ / 67912,80 kr
GATT-based Profile Client	120\$	0	0
Bluetooth Qualification Test Facility	9600\$	9600\$	0
Bluetooth Recognized Test Facility	2400\$	2400\$	0
PTS Radio Module	99\$	0	99\$ / 700,35 kr
I alt	21819\$	12000\$	-
I DKK (kurs = 707)	154353,06 kr	84891,00 kr	

Figure 6: Bluetooth qualification fees cost

Trods de beacons vi bruger i vores løsning, der allerede er markedsført og forventet testet af Estimote Cloud, kan der stadig forventes at være udgifter i forbindelse med test af vores løsning. Derudover er der også omkostninger i

forbindelse med at kunne markedsføre et produkt der benytter bluetooth, altså et årligt medlemskab af Bluetooth.

Annual Membership Fee		
For companies whose annual revenue is < \$100M USD	\$0	\$9,000
For companies whose annual revenue is > \$100M USD	\$0	\$42,000

Figure 7: Annual membership fee

Da projektet forventes at være en mindre startupvirksomhed vil de årlige omkostninger til at kunne markedsføre sig med bluetooth være 9000\$ altså 63668,25 kr til kurs 707. Altså når man skal lancere sit første produkt med bluetooth, vil der være en omkostning lige under 220.000kr for medlemskab af bluetooth og kvalificering af produktet.

3.5.2 Estimote Proximity beacons

Selv beaconsdevices vi bruger i vores løsning, har også en pris. Hvis man som virksomhed gerne vil købe 4 styks, kan man købe den for 123,32€, 917,34kr til kurs 743,87. Når man skal udregne en løsning til et hospital, vil det være mere overskueligt og forventeligt at regne prisen for 1000 beacons, da en løsning formegentlig vil kræve op mod de 1000 beacons. 100 beacons koster lige under 230.000kr. Hvis man køber beacons direkte fra producenten, vil det forventeligt være lavere og hvis der bliver indkøbt samlet, vil det der forventeligt også være en rabat ved større ordre. [12]

3.5.3 Mazemap

Når vi i vores løsning gerne vil bruge mazemap som kortudbyder, har de på deres hjemmeside en pris for 5000kvm på 49000kr for første år og 30000kr for de følgende år. Dette er inklusiv hosting, support og opdatering. [13]

3.5.4 Firebase

Til at lagre informationen fra beacons, kræver det en cloudbaseret serverplads. Her har firebase en pris for 1 GiB (1.078 GB) på 0.108\$. Hvis man estimerer at en beacons kræver 1kB, vil et hospital med 1000 beacons kræve 1MB. Altså at omkostningerne til serverpladsen for beacons til et hospital er ubetydelige. [14]

3.6 User Experience

Da vi gerne vil give brugeren en brugeroplevelse, bliver vi nødt til at drøfte alt, som har med brugerens oplevelse og interaktion med vores produkt. Dette gøres for at danne os et overblik over vores overvejelser og valg, og om det er den ønskede interaktion, der er i fokus.

3.6.1 User stories

Vi har lavet nogle user stories, som kan findes i bilag 2-4, som vi bruger til at finde ud af hvilke behov, vores potentielle brugere, vil have. Disse user stories, har vi lavet på baggrund af vores valg af mulige brugere. Dette er sket således, at vi får en bred målgruppe, hvis omfang er skærpet nok til at lave et målgruppespecifikt produkt, men stadig bred nok, til at majoriteten vil få gavn af den. På baggrund er deres udsagn, kan vi så lave en masse overvejelser, om hvordan vores produkt skal se ud. Disse overvejelser kommer konkret fra de brugerudsagn vi har fra user stories, hvilket kan ses her:

- ”så jeg vil ønske der var en APP hvor jeg bare kunne indtaste min onkels stue, hvorefter en god rute derhen fremvises”
- ”Så jeg ville have sat pris på hvis der var en nem og intuitiv måde som jeg kunne finde forskellige madsteder på.”
- ”så hvis jeg får et lokale nummer, kan jeg indtaste det, og finde den hurtigste rute derhen.”

3.6.2 Use case model

På baggrund af vores UserStories, har vi lavet en usecase af vores program, som viser hvor meget brugere skal gøre, for at bruge vores APP korrekt. Dette hjælper os med at danne et overblik over, hvor simpelt programmet er og hvor mange input der skal haves fra brugerens side. Usecasen kan ses på figur: 8:

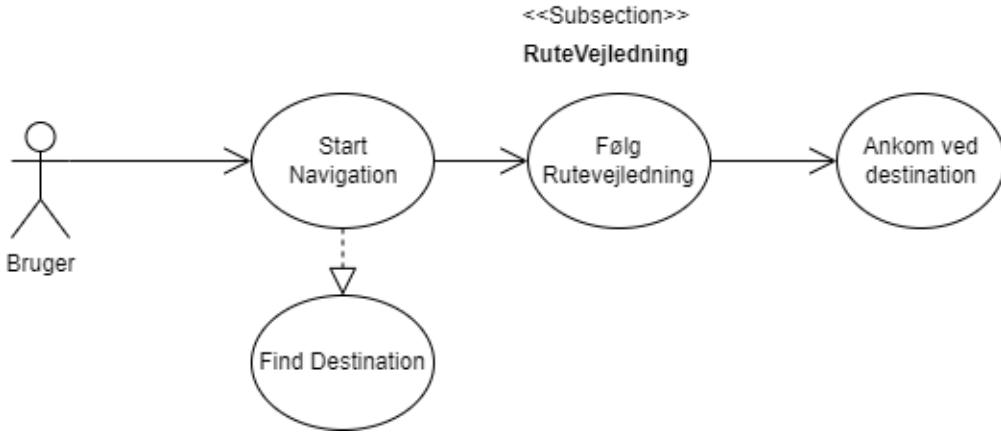


Figure 8: Use-Case Diagram

3.6.3 Generelle overvejelse

Vores målgruppe indeholder både ældre, men også unge. Både Patienter men også besøgende. Både patienter og besøgende med godt syn, og dem som er lidt til svært seende. Derfor har vi en meget bred målgruppe, og skal derfor lave et produkt, som tager hensyn til de forskellige bruger og deres behov.

Det første er flowet af vores program. Hvordan vil vi gerne have at vores brugere bliver ført igennem vores program? Dette bruges til at dikttere designet af UI, så vi nudger brugeren, til at følge vores ønskede forløb.

Vi ved, på baggrund af brugerudsagn, at der ønskes et let, hurtigt og forbeholdstagende program. Dette betyder, at der skal være størst mulig funktionalitet, med mindst mulig kompleksitet. Vi skal så vidt lave et program, som kun beder brugeren om de ting som er nødvendige. Dette kunne f.eks. være destination. Start position er ikke noget som brugeren skal indtaste, dette vil være spild af tid, da vi kan få programmet til automatisk at finde det.

Dette betyder at vi kan gøre det så enkelt som at have en startside, hvor de skriver destination og forhold, dernæst klikker start, også begynder deres navigation. På denne måde, vil vi kunne overholde at have så få knapper, eller bruger inputs, men stadig opnå den ønskede funktionalitet.

Det næste vi skal tænke over, er brug af farver og fonte. Da det er en bred målgruppe, kan der forekomme farveblinde, eller personer med nedsat syn, som også vil bruge vores app. Derfor skal vi vise hensyn, når vi vælger vores farver. Måden dette gøres er ved at sørge for høj kontrast, og store klare tekstrfonte, som gør det nemt at læse. Dog samtidig skal vi også være korte og præcise i vores tekst. Der er ingen grund til at have en masse tekst. Stikord eller en kort sætning, der forklarer brugeren, hvad de skal gøre, eller hvad en funktion gør.

3.6.4 UI

I denne sektion, vil vi analysere, hvert enkelt aspekt af vores program, som brugeren vil se eller interagere med.

Layout Vi bruger Mazemaps API, derfor vil det være det brugeren hovedsageligt kigger på, også være deres kort af hospitalet. Dog kan vi stadig implementere vores eget design og layout, af alt andet end kortet. Derfor skal vi have et layout, som ikke viser en masse overflødig, men kun fremviser den viden, som brugeren vil være interesseret i. Derfor kan man overveje, at der under navigationen, ikke fremvises andet end kortet med ens rute og en afslut-knap eller lignende. Dette kan man gøre, ved at have en separat side, hvor man f.eks. indtaster sin destination.

Knapper Vores bruger skal møde så få knapper som muligt, der skal være en knap til at starte ruten og en knap til at afslutte ruten. Der er ikke brug for flere knapper end disse, da det kun vil overkomplificere vores app. Brugeren vil åbne appen, klikke på søgefeltet og klikke start. Derefter, når vedkommende når sin destination, vil vedkommende lukke appen. Hvis brugeren skal afbryde navigationen, skal de også klikke på en knap. Det vil sige, at vi i vores program, muligvis kan skærpe ned, til kun tre knapper eksklusive søgefelter mm.

Etage-fremvisning Da der er flere etager på et hospital, skal man også kunne fremvise tydeligt, hvilken etager man befinder sig på. Derfor skal dette fremstå tydeligt i appen. Med inspiration fra andre Kort-Apps. Er der en trend til at have det i højre hjørne. Dette betyder, at hvis brugeren er kendt med andre maps, så vil de søge højre hjørne for etagerne, derfor skal vores også være der. Den skal fremstå tydeligt og med god kontrast, så selv de svært seende, vil kunne se det.

Søgefunktion Søgefunktionen skal være det vigtigste, når de åbner appen, fordi dette er hvad de bruger den for. Jo nemmere søger funktionen er at bruge, desto nemmere bliver det for brugeren. Derfor er det også søgefeltet, som er det første de ser, når de åbner, hvor ved de kan indtaste lokation, og klikke start.

Under Rute Når man er igang med ruten, kigger man ikke altid på skærmen, derfor skal farverne være meget tydelige og teksten stor, så man kan få den nyttige viden, ved et hurtigt blik. Det skal det under ruten, være klart hvor man er, med en stor fed cirkel i en klar farve. Dernæst skal hele ruten også være i en nogenlunde samme farve, og fremstå tydeligt, så man ikke går på vildspor ofte.

Ankomst Vores program skal være simpelt, derfor skal vi ved ankomst ikke have en kompliceret side, hvor brugeren for en masse information eller lign. Derfor er der en afslut knap, som er vist på skærmen, så brugeren kan afslutte når de vil, om de er færdige med ruten eller ej.

4 Design

For at kunne implementere vores applikation, bliver vi nødt til at designe, hvordan strukturen af programmet skal se ud. Deraf vil der i dette kapitel indgå:

- Overgående DesignPattern
- Forklaring af IBeacon lokaliseringen
- Brugen af Mazemap API
- Sekvens- og Designklasse diagram over systemet
- Placering af beacons i et beaconMap

4.1 ModelViewViewModel

Vi bruger Model-View-ViewModel (MVVM) til at lave en separation imellem business og præsentationslogikken fra applikationens user-interface. Der er primært tre elementer i et MVVM pattern:

- View
- View Model
- Model [15]

Der kan ses på figur 9:

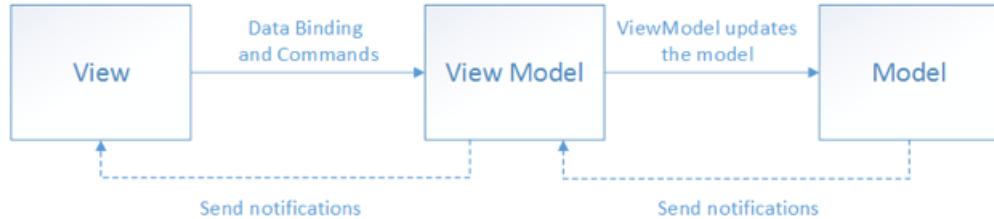


Figure 9: MVVM Pattern

View er ansvarlig for strukturen, layout og udsende af hvad brugeren ser og indeholder idealt ingen business logik.

View Model er ansvarlig for at implementerer funktionalitet til View, således at de funktionaliteter, som View Model udbyder, definerer funktionalitet af brugergrænsefladen, imens at View sørger for, hvilken funktionalitet der skal fremvises på brugergrænsefladen. View Model er også ansvarlig for at koordinerer Views' interaktion med de forskellige modeller, samt notificerer View Model om evt. opdateringer.

Model er ansvarlig for at indeholde systemets data, samt indholde business og validations logik.[15]

4.1.1 Vores MVVM

Vi har nogenlunde opdelt vores struktur som beskrevet i diagrammet på figur: 10

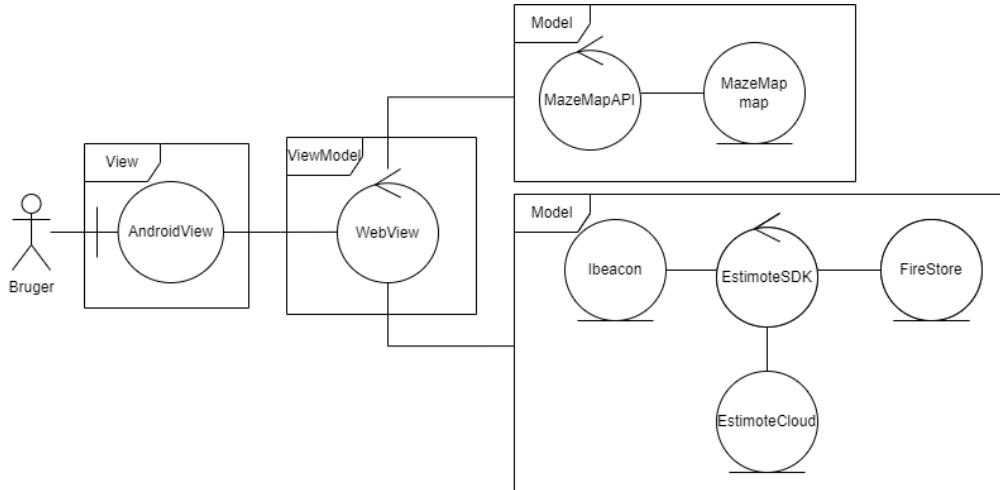


Figure 10: Vores Model

4.2 Beacon Lokalisering

Vi bruger Ibeacons fra estimote til at lokaliserer brugen vha. Bluetooth broadcasts. Typen af Ibeacons fra estimote kommer med sine egne overvejelser og bliver gennemgået i afsnit: 4.6 om beaconmap. For at få Beacon lokaliseringen til at fungere godt, gør vi brug af Estimotes SDK, og en cloud baseret database. Når vi så har fået lokaliseret brugeren, kan vi overføre denne lokation videre til Mazemap Api 'en, og på denne måde interaktivt vise hvor brugeren befinder sig.

4.2.1 Estimote SDK

Den Estimote SDK som vi vælger at bruge, er proximity zone. Servicen består grundlæggende af 4 elementer:

- Observer: starter og stopper monitoreringen af ens zoner.
- Zoner: repræsenterer et areal af beacons med samme tag.
- Proximity context: en beacon med dets tags osv.
- Callbacks: Hver zone har 3 typer af callbacks triggers:
 - Når man går ind i en zone.
 - Når man går ud af en zone.
 - Når antallet af observerede proximity contexts ændrer sig.

Grunden til at vi ikke brugte dBm og derudfra udregnede positionen kan ses i afsnit 6.1.3. Ved hjælp af disse elementer har vi mulighed for at registrere, om man befinner sig i en zone, og deraf kan vi bruge Estimotes iBeacons til at bluetooth broadcaste deres unikke tags, som derved kan bruges til at få deres unikke beaconID igennem Estimote cloud service. Når vi så har den unikke beaconID kan vi Query vores database for at få de specifikke koordinater for vores beacon.

4.2.2 Firestore Database

Til vores database bruger vi en NoSQL-database - mere specifikt Firestore. Vi vælger en NoSQL database, fordi vi gerne vil have en schema-less database, dvs. en database uden specifikke databaserestriktioner. En anden grund er, at en NoSQL database er nemmere at lave iterationer på, da vi meget hurtigere kan tilføje nye felter til vores database, uden at skulle lave det hele retrospektivt hvis vi f.eks. tilføjede flere funktionaliteter til vores løsning. En NoSQL database har dog også ulemper som f.eks. at duplikere data, men fordelene er bedre for os, da vi får en rigtig hurtig read time. Strukturen af vores database skal bestå af en Collection, der består af dokumenter der består af felter.[16] Deraf ville vi få noget der minder om en træ lignede struktur, som afspejler vores forskellige beacons med unikke ID'er, der hver har sin egen prædefinerede longitude og latitude.

4.3 Mazemap API

Da vi ikke vil opfinde den nye tallerken, vælger vi at bruge Mazemaps API. Dette betyder dog, at vi må arbejde med Mazemaps API, som den er udviklet. Dette betyder at vi skal bruge Android View webView til at køre vores JavaScript, da den skal køre på en Native Android jetpack compose app. For at dette skal kunne fungere godt, skal vi overføre vores koordinater løbende igennem webView således, at vi kan ændre vores lokation via. JavaScript. Vi har lavet et sekvensdiagram, der beskriver dette forløb på figur: 11

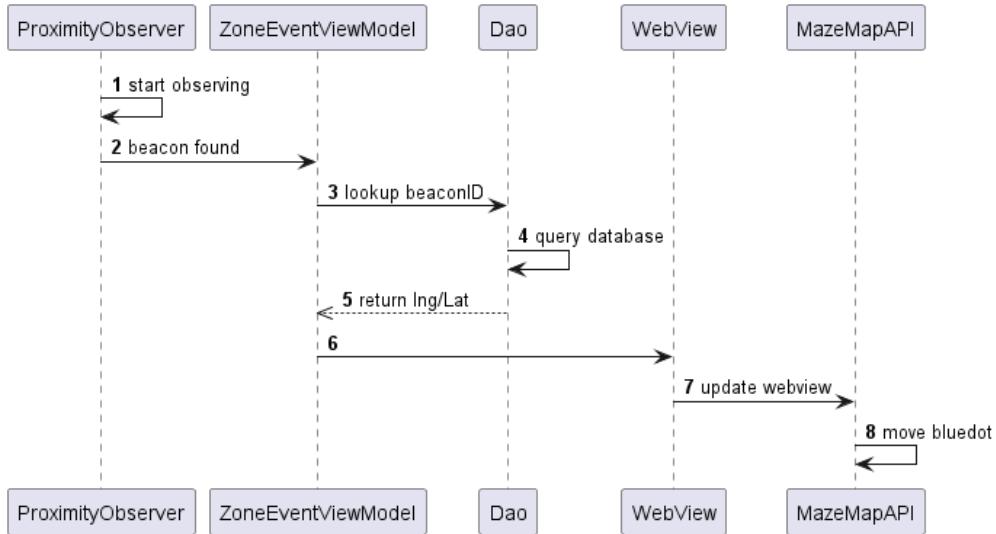


Figure 11: Planlagt sekvens af overførsel af koordinater

4.3.1 Problem

Vi må konkludere at Android View WebView ikke kan bruges til at evaluate javascript, da WebViewet skal konstrueres i en lambdafunktion i en composeable. Dette medfører at man ikke kan tilgå WebViewet og kalde evaluate javascript, som ville opdatere javascript filen og dermed give os mulighed for at flytte bluedotten.

Da vi ikke kan tilgå WebViewet, og vi ikke kan konstruere det et andet sted og initialisere det i androidView, da denne også skal have en helt præcis syntaks, er vores eneste mulighed at loade hele webView om. Her kan vi dog godt give koordinaterne med, men dette ville resultere i, at vi reloadede hele webviewet, hver kan man nærmere sig en beacon, og derved fjernede ruten og gøre applikationen ufattelig langsom og nærmest ubrugelig.

4.3.2 Alternative løsninger

Vi kan umiddelbart komme frem til to alternative løsninger, der begge kommer med sine egne fordele og ulemper. Den første løsning er, at vi skriver applikationen i React Native og den næste løsning er, at vi laver falsk bluetooth koblingen.

React Native

Fordelen ved at lave applikationen i React Native er, at vi får en cross-platform applikation der kan køre på både Android, men også Apple produkter. Dette ville være en mere brugervenlig løsning, da vi ikke har lyst til kun at virke på en Android mobil. Ulemperne er f.eks. at man får en langsommere load time, som egentlig ikke ville være et stort problem i vores tilfælde. Den store ulempe bliver at vores Implementation bliver meget længere, da vi ikke allerede har færdighederne til at kunne udføre følgende Implementation.

Falsk bluetooth kobling

Vi ville også kunne lave en fake bluetooth kobling, set som at vi kører en todelt løsning i stedet for en samlet løsning. Disse to løsninger ville fungere således, at den første skal køre i baggrunden og kommunikere via bluetooth med Ibeacons og firebase, og på den måde lokaliserer brugerden. Når den så har lokaliseret brugerden, skal den sende en notifikation med en besked om hvilken zone man bevæger sig ind i, således at man kan vælge denne lokation i den del. Den anden del skal køre Mazemap API 'en i androidview webView, som skal have mulighed for at vælge den

iBeacon, som svarer til den notifikation man har fået fra den anden del. På denne måde ville vi kunne simulere, at app'en selv overførte lokationen til mazemap og derved vise, at de enkelte dele godt virker.

4.3.3 Vores løsning

Vores løsning til dette problem, bliver løsningen med Mechanical turk, da vi er presset på omfanget af tid givet i projektet, som vi kan bruge til Implementation. Dette betyder at vi må indføre fejlkilden i vores konklusion, om hvorvidt vores produkt kan det, som vi siger den kan, og hvor godt den gør det.

For at komme igennem disse fejlkilder, bliver vi nødt til at planlægge vores afprøvning således, at vi tilser alle fejlkilder så godt som muligt. For at tydeligt beskrive hvordan dette nu ændre, hvordan sekvensen for overførslen af koordinaterne ser ud, har vi lavet et opdateret sekvensdiagram, som ses på figur: 12

Diagrammet viser hvordan man nu overfører et beacon nummer i stedet for koordinater, hvor beacon nummeret nu er associeret med en prædefineret koordinat. Systemet bliver også afhængig af en aktiv bruger, hvilket ikke er optimalt.

4.3.4 Fremtidsperspektiv

Hvis vi kigger nogle år frem, kan der være udviklere, som gør det muligt at lave vores løsning i et Kotlin Jetpack compose miljø. Disse udviklinger kan f.eks. indeholde:

- Videreudvikling af androidview WebView.
- Udvikling af Jetpack compose javascript kompatibilitet.
- Udvikling af Mazemap API til kotlin og jetpack compose.
- Erfaring med React native.
- Udvikling på google maps indoor API således at man selv kan lave kort.
- Lave hele kort delen selv.

4.4 Diagrammer over hele systemet

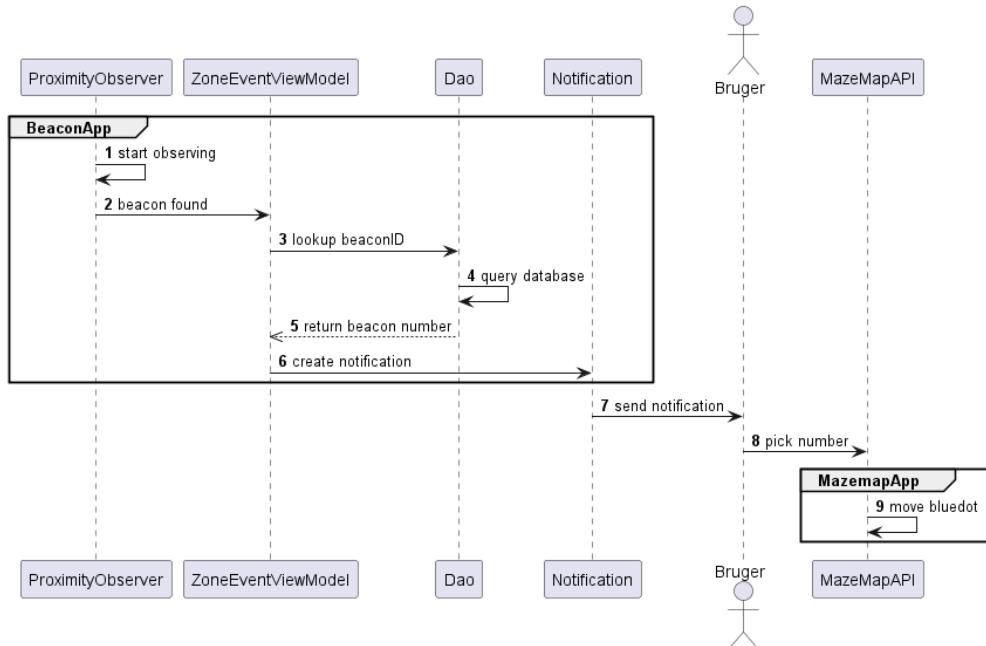


Figure 12: Opdateret sekvens af overførsel af koordinater

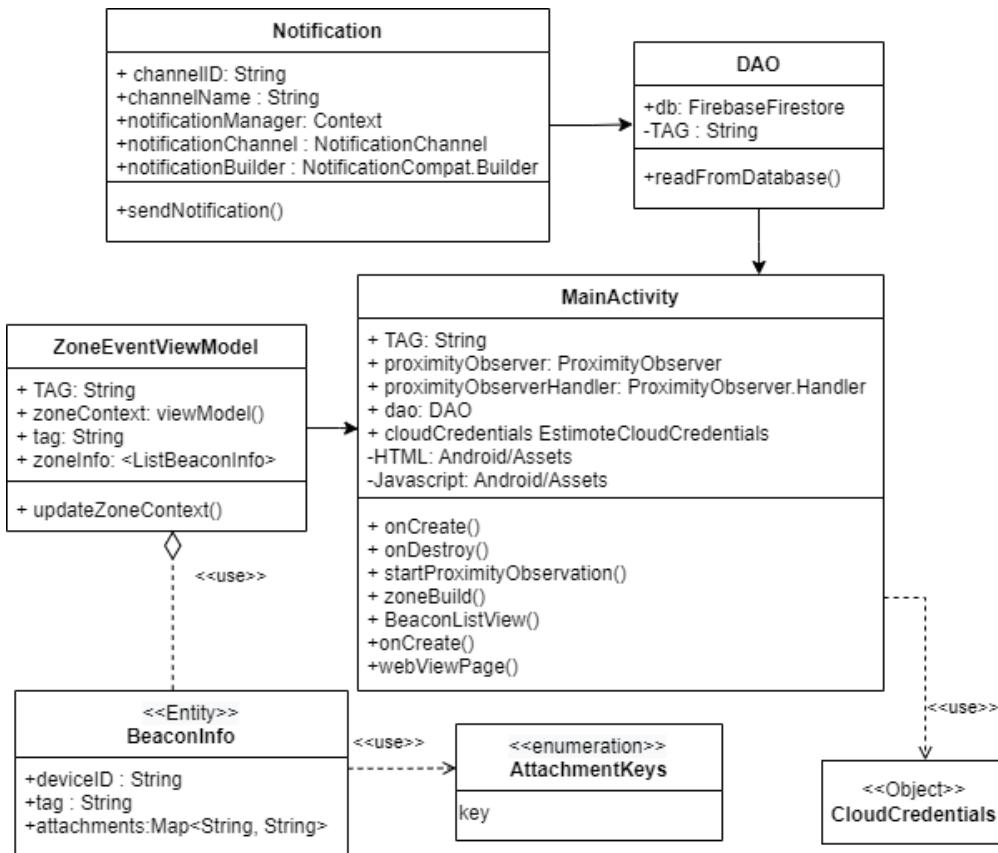


Figure 13: Design Klasse Diagram

4.5 UI

Vi har designet vores UI i forhold til vores analyse og UX. Der er en side, som indeholder forskellige elementer, i forskellige stadier. Disse elementer og deres design vil blive drøftet i dette afsnit.

4.5.1 Navigationside

Dette er den eneste side, som bliver vist for brugeren. Den Består af alle elementer som brugeren skal interagere med, såsom en start knap, et søgefelt mm. Dog er det hovedsageligt Mazemaps API der er i fokus, som designet er blevet lavet ud fra.

MazeMap

Mazemaps API kan ikke ændres, så designet er som de har lavet det. Deres API er et kort, som hele tiden fremvises på skærmen. Det er dette kort, som brugeren gør brug af, til at navigere rundt eller følge rutevejledningen. Det er også under Mazemaps API, at vi har vores etage fremvisning, dette er også implementeret gennem API'en, så den ligger som Mazemap har sat.

Søgefelt

Dette er et felt, hvor brugeren kan søge efter et lokale, i stedet for at finde det på kortet. Dette sættes på toppen af appen, da det er dér, man normalt har søgefeltet. Så for at gøre det nemt og intuitivt, så gør vi det samme.

Start Navigation

Denne knap er til for at starte sin rutevejledning. Den sættes på bunden, med stor fokus omkring den, så brugeren uden tvivl kan finde frem til den.

Skift Position

Så har vi en skiftposition del, som består af nogle knapper man kan klikke på, for at skifte position. Dette bliver sat til i bunden, så man nemt kan bruge sin tommelfinger til det.

Eksempel på navigationsiden

På baggrund af disse designbeslutninger har vi konstrueret følgende eksempel på vores brugergrænseflade

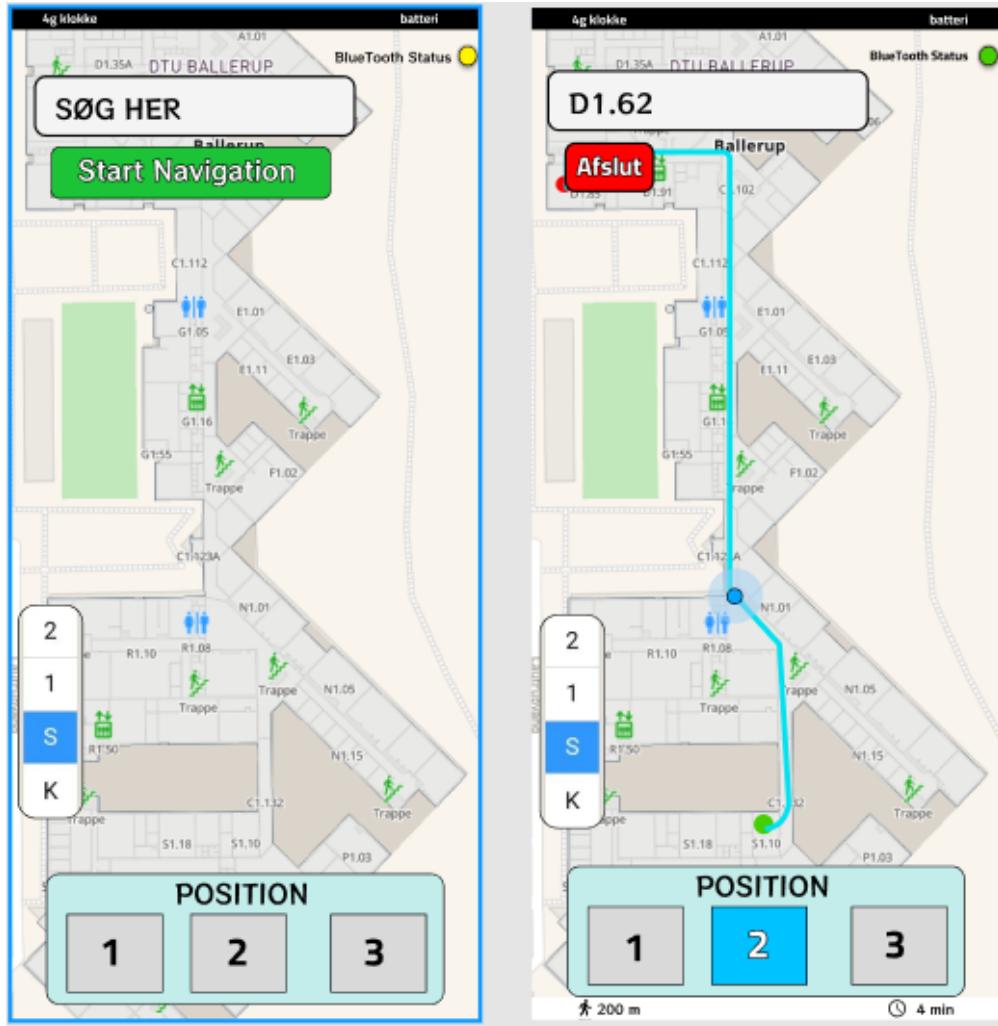


Figure 14: Eksempel på brugergrænsefladen

4.6 BeaconMap

For at kunne programmere det interaktive kort med Beacons, er det nødvendigt at undersøge hvilke typer beacons der skal benyttes, hvor mange beacons der er behov for og hvordan beacons fysisk skal placeres på den ønskede lokation. Derfor er det essentielt først at undersøge hvilket udvalg der eksisterer på markedet af beacons, hvilke muligheder og begrænsninger de har, så de på bedst mulige måde opfylder kravene for systemet.

4.6.1 Udvalg af beacons

Estimote har udviklet en række beacons, der er udviklet med forskellige formål. På figur 15 ses en oversigt med relevante specifikationer for diverse beacons.

							
Built-In Radios	Bluetooth 4.2	Bluetooth 4.2	Bluetooth 5.0	Bluetooth 5.0	Bluetooth 5.0 UWB	Bluetooth 5.0 WiFi	Bluetooth 5.0 LTE-M/NB-IoT GPS/GALILEO/ GLONASS
Default battery life	1 year	2 years	3 years	3 years	3 years	USB	2 years
Maximum battery life	1 year	3 years	5 years	5 years	5 years	USB	USB-C
Maximum Bluetooth range	7 meters	70 meters	100 meters	150 meters	200 meters	10 meters	200 meters
Supported use-cases	asset tracking	proximity, asset tracking	proximity	indoor location	indoor location	digital signage	asset tracking, vehicle tracking, proximity gateway
Thickness	6 mm	15 mm	25 mm	24 mm	27 mm	14 mm	16 mm
Length	29-66 mm	55 mm	69 mm	63 mm	75 mm	90 mm	90 mm
Width	28-44 mm	38 mm	46 mm	41 mm	50 mm	60 mm	90 mm
Weight	8-12 g	30 g	86 g	67 g	98 g	72 g	57 g
Recommended SDK	Estimote SDK	Proximity SDK	Proximity SDK	Indoor SDK	Indoor SDK	Mirror SDK	Web IDE
Fleet management	Estimote App	Estimote SDK	Estimote SDK	Estimote SDK	Estimote SDK	Cloud via WiFi	Cloud via LTE
Programming language	Objective-C Swift Java Kotlin	Objective-C Swift Java Kotlin	Objective-C Swift Java Kotlin	Objective-C Swift Java Kotlin	Objective-C Swift Java Kotlin	Objective-C Swift Java Kotlin JavaScript	JavaScript

Figure 15: Udvalg af beacons fra estimote [17]

Her er der tre mulige kandidater til systemet baseret på kravene i forhold til bluetooth rækkevidde. De tre kandidater er henholdsvis Proximity Beacon, Location Beacon og Location UWB Beacon.

4.6.2 Proximity Beacon

Proximity Beacon har en bluetooth rækkevidde på 100 meter og fungerer efter hvorvidt den mobile enhed er tæt på, i nærheden eller langt væk. For yderligere forklaring se afsnit 5.3 om Proximity-ZoneEventViewModel. Dette er tilstrækkeligt for at kunne udvikle systemet, og de andre to kandidater er på baggrund af dette valgt fra, trods af at de er designet til indendørs lokalisering. Desuden har Proximity Beacon en forventet batterilevetid på 3 år, hvilket vurderes til at være rigeligt batterilevetid. Yderligere specifikation for den udvalgte beacon kan findes på estimotes egen hjemmeside.[18]

4.6.3 Plantegning

Her skal der tages højde for, hvor mange beacons der skal placeret i de forskellige lokaler for at kunne være sikker på, at den mobile enhed er placeret i det rigtige lokale, modsat beacons placeret i andre lokaler/gangarealer. På gangarealerne skal der placeres nok beacons, så man kontinuerligt kan følge sin rute fra a til b, uden at der kommer store hop. Her er det vurderet til, at der skal være cirka 10 meter mellem hver beacon. Steder hvor der er mange små rum, kan der også placeres én beacon, for at markere et område. Desuden skal der placeres beacons ved trapper og elevatorer. Tager vi udgangspunkt i en bygning, i dette eksempel et afsnit på DTU Ballerup Campus på figur 16, er der givet et kort med placeringer af en række beacons, der skal kortlægge og give et indtryk af, hvor mange beacons der skal benyttes på et givent område for, at understøtte funktionaliteten af systemet.



Figure 16: Placering af beacons i mappet

På figuren er der tegnet blå områder, der skal markere hvor der er placeret en beacon og dens rækkevidde. Beacons radius/rækkevidde er omrent 5 meter. Det har ikke været muligt at skaffe en plantegning angivet med dimensioner, så der er her brugt et kort fra Mazemap. Desuden er det værd at bemærke, at der ikke er taget højde for svækkelse af signet ved forhindringer såsom vægge, interiør eller andre genstande der kan påvirke signalrækkevidden.

5 Implementation

I dette afsnit vil vi forklare hvordan vi har implementeret følgende:

- Beacon-CloudCredentials
- Beacon opsætning
- Proximity-ZoneEventViewModel
- Notifikation
- FireStore - Firebase
- MazeMap
- Webview

5.1 Beacon-Cloud'-Credentials

Inden selve koden gennemgås, forklares hvordan vi fik adgang til vores beacons. Først fik vi givet 3 beacons af vores underviser. Derefter oprettede vi en konto på cloud.estimate.com, hvorefter vi så informerede vores underviser om den anvendte mailadresse så vi kunne få tilknyttet ejerskab over de 3 beacons i vores konto. Således kunne vi vha. cloud.estimate.com tilgå vores beacons. Inde på cloud.estimate.com har vi således, angivet tags til vores beacons hvor man desuden kan se de individuelle identifiers, som bruges til at tilgå vores beacons i vores applikation:

- 507 Identifier: ed79379663c0ebea0a839dd870ce1f32
- 508 Identifier: 6e9ea9f6936b18d66f9782e89c95431e

- 509 Identifier: 2058e68b57b5a9c1b4ac0ced49600813

Desuden forbindes applikationen via. hjemmesiden. Dette gøres ved konkret vha. et App ID og en App Token. Begge oprettes via. cloud.estimate.com. Under Apps trykkes "Add New App", hvor man så vælger "Your Own App". Således fik vi oprettet følgende applikation:

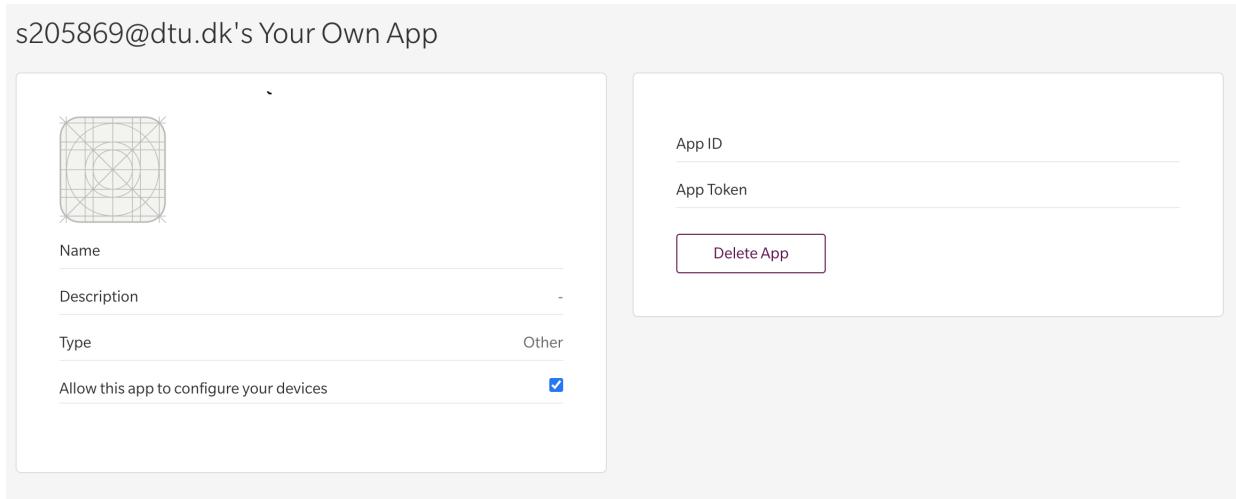


Figure 17: Skærbillede af vores applikation inde på cloud.estimate.com.

Således har vi fået oprettet et App ID og en App Token. Derefter tilføjede vi de relevante dependencies i vores gradle filer. Herefter har vi under klassen ved navn "CloudCredentials", angivet vores App ID og App Token. Disse CloudCredentials benyttes så når man vil noget med vores beacons, fx i funktionen "startProximityObservation()", så bruges til at opstarte proximity observation fra vores 3 beacons.

5.2 Beacon opsætning

Da vi startede med at benytte vores beacons havde de en default opsætning, dog opdagede vi en udfordring ift. proximity/signalstyrke. På daværende tidspunkt var vores proximity range sat til near-range, hvilket vi ikke havde bemærket. Dette forsøgte vi at tackle, ved at ændre broadcasting power, og det er netop her beacon opsætningen kommer ind i billedet. For at kunne ændre opsætningen, loggede vi ind via. Estimote Cloud, hvori vores beacons kan tilgås. Her valgte vi Beacon 509, og ændrede opsætningen på denne. Broadcasting Power kan ændres for følgende pakker: Connectivity packet, Estimote Telemetry Packet,iBeacon Packet, Eddystone-UID Packet, Eddystone-URL Packet og Eddystone-TLM Packet. Nedenfor kan der ses et skærbillede af et eksempel af Beacon opsætning på Cloud Estimote.

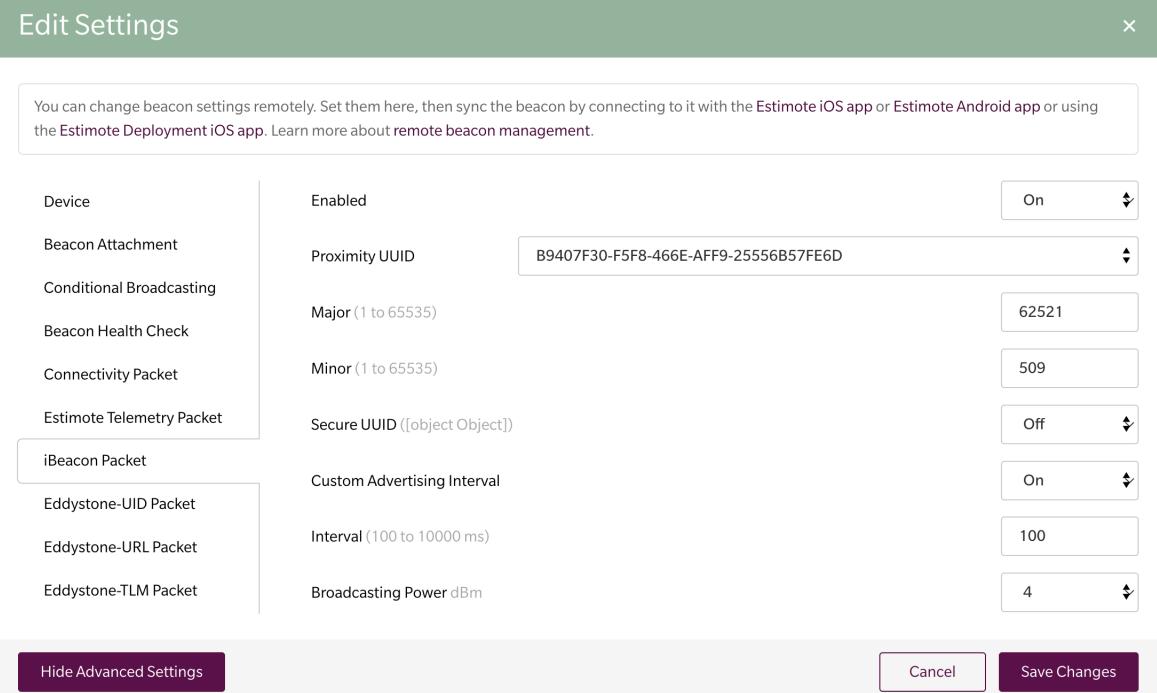


Figure 18: Skærbillede af et eksempel af Beacon opsætning på Cloud Estimote.

Disse Beacon-indstillinger kan foretages via Estimote Cloud hjemmesiden, dog anvendes indstillingerne først efter de nye indstillinger er blevet synkroniseret med vores beacons, hvilket gøres ved at forbinde til dem via Estimote Android app. Dog ændres disse parametre ikke i vores program, idet vi holder dem på deres default indstillinger. Det som vi gør på at justerer på vores proximity, er bare at ændre hvilken range vi bruger i selve koden, om det er immediate, near eller far range, hvilket uddybet i næste afsnit.

5.3 Proximity-EventViewModel

Den grundlæggende beacon parameter der gøres brug af i vores applikation, er proximity. Den måde det kodemæssigt kommer til udtryk er først ved at se på funktionen med navnet "startProximityObservation" i MainActivity.kt. Her påbegyndes den aktive observationer fra vores 3 beacons, hvilket sker ved at der i vores proximityZones tilføjes zoneBuild() for hver beacon. zoneBuild() er endnu en funktion i samme klasse, som bruges til at specificere hvad der skal ske når man går ind i en zone, når man går ud af zone og bare når der sker en kontekst forandring, som både kan være at man går ind eller ud. Ved indgang i en zone, afhentes tag fra databasen. Ved udgang sker der ikke noget, og ved kontekstændring sker der en opdatering af zonekonteksten. I samme funktion angives der at det er in.Far.range() der benyttes. Der findes nemlig 3 typer ranges, som er immediate, near og far. Immediate er indenfor et par centimeter, near er indenfor et par meter og far er mere end et par meter. Hvor vi så har valgt at benytte far-range, pga. årsager som er redejort for i afsnit 4.6 om Beacon Map.

5.4 Notifikation

Først laver vi en NotificationChannel, NotificationManager og en NotificationBuilder. Dernæst sætter vi vores notificationmanager til at lave en notificationchannel, som vi sætter som vores notificationChannel. Dernæst opsætter vi vores notifikation via, NotificationCompat i vores NotificationBuilder. Til slut laver vi en notifikation igennem notificationManager med parametrene som vi har sat via vores NotificationBuilder. Alt dette sker i funktionen "sendNotification()" hvor notifikationen bliver konstrueret og man får mulighed for at angive en titel og besked.[19]

Dette gøres fordi vi har med en todelt løsning at gøre, så når slutbrugeren modtager en notifikation, omkring hvilken zone vedkommende befinder sig i, kan personen så angive dette således at Blue Dot opdateres. Hermed fortsættes navigationen.

5.5 FireStore - Firebase

Vi har opsat vores database således at vi ville have en abstraktion over vores beacons. Denne struktur skulle bestå af en overkategori over alle beacons, og dernæst en underkategori over de enkelte beacons.

De enkelte beacons skulle dernæst af 3 felter, 2 til koordinater og 1 til et nummer, der hang sammen med vores brugergrænseflade, da vi jo laver en falsk bluetooth opkobling. Strukturen vises herunder:

- Collection: Beacons
 - Document: UnikBeaconID
 - * Field: Lat: String
 - * Field: Lng: String
 - * Field: nummer: String

Til at læse data fra vores database lavede vi et 'get'-kald. Grunden til at vi lavede et 'get'-kald, var fordi at et 'get'-kald, fungerede bedre end en snapshotlister i vores program. Dette var fordi, at vores databases data egentlig ikke ændrer sig, medmindre der tilføjes flere beacons til en lokation, samtidig med at dataene ikke skal bruges når den opdateres, men i stedet skal bruges når man træder ind i en ny zone. Derfor kunne vi lave et 'get'-kald til et specifikt dokument inde i den altid samme Collection. Derefter kontrollerede vi om dokumentet eksisterede og hvis det gjorde det, smed vi en notifikation.

5.6 MazeMap

Da vi skulle lave vores Implementation af mazemap, brugte vi Mazemaps dokumentation og omskrev den lidt så den bedre passede til vores løsning.[20] Deraf lavede vi et HTML dokument med et stylesheet og tilhørende javascript. Vi tilføjede 3 ekstra knapper til at simulerede de placerede beacons og gav disse beacons prædefinerede destinationer. Knapperne fungerer således, at når man trykker på dem, vil man flytte Blue Dot der beskriver hvor man er befinner sig. Vi har lavet nogle små ændringer til mazemap-APIen's UI, dette har vi gjort, da vi under vores analyse og design afsnit har fundet nogle løsninger der passer bedre til netop vores løsning.

5.6.1 UI

Disse løsninger består primært af

- At have positionsknapper i bunden
- At have en meget stor knap i toppen, til at starte og afslutte rutevejledning.
- At have et søgefelt lige over startknappen, så navigation er i fokus et enkelt sted
- Kun at have en enkel side, hvor ens start position er angivet på forhånd.

Så vi har sørret for at de vigtige tekster, har en border, til at highlighte dem. Samt har vi valgt farver der matcher. Så start knappen er Grøn, Afslut knappen er Rød og knapper til position er blå, da det er en Blue Dot, man ændrer. Derfor kan vi, ligesom i analysen vise, at vores UI, kun består af tre elementer, så det er så simpelt som muligt. Det resterende da vises på skærmen, såsom kortet, zoom ind og ud, eller Etage-fremvisning, er implementeret gennem Mazemaps API.

5.7 Webview

For at kører vores mazemap api bliver vi nødt til at bruge webView, sådan at vi kan køre HTML, CSS og javascript dokumenterne i et kotlin miljø. Vi har opsat dette således at vores app indeholder et Html og et Javascript dokument i sin assets folder, som vi derved kan åbne igennem Webview oprettet som en composeable. Grunden til at vi bruger webView, er fordi at webView er en browser engine som vi kan åbne via kotlin og jetpack compose og derved kører vores HTML/Javascript dokumenter.

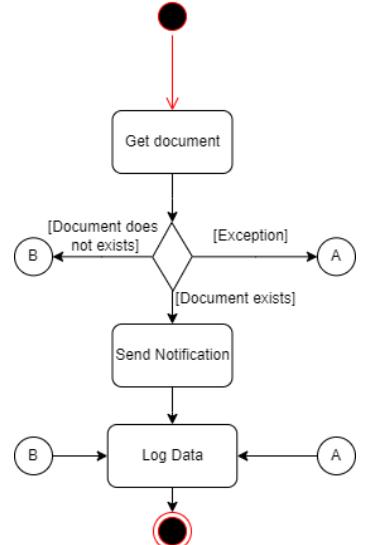


Figure 19: Database Read.

6 Afprøvning

I dette afsnit vil vi udføre følgende afprøvning:

- Beacon Signal afprøvning
- Lokalisering afprøvning
- Effekt Afprøvning
- Usability Test

6.1 Beacon Signal afprøvning

I dette afsnit vil vi gerne undersøge vores beacons signalstyrke. Til dette har vi konstrueret en applikation der kan måle signalstyrken i dBm. Med denne applikation vil vi gennemgå en verifikation af Friis ligning, samt et eksempel på en triangulation via. Rstudio.

6.1.1 Applikation til måling af signalstyrke

Da vi gerne vil måle forskellige beacons dBm, var det nemmeste at lave en android applikation der modtog signalerne og registrerede deres dBm. Da vi gerne ville udføre nogle beregninger til triangulering, lavede vi en applikation der kunne vise 3 beacon samtidigt. Den måde vi lavede dette, var ved at bruge "estimote 101" eksemplet og modificere det så det passede til vores forsøg. Eksemplet fungerer ved at modtage alle signaler og filtrerer dem fra, som ikke indeholder en specifik nøgle, og deraf vises de ikke-filtrerede beacons på en brugergrænseflade i en row. Disse vises med et navn og dBm værdi. Vi indsatte de korrekte identifikationsnøgler og omskrev brugergrænsefladen til at vise 3 filtrede beacon på samme tid med deres dBm. Et eksempel på hvordan vores afprøvnings applikations er, ses i figur: 20

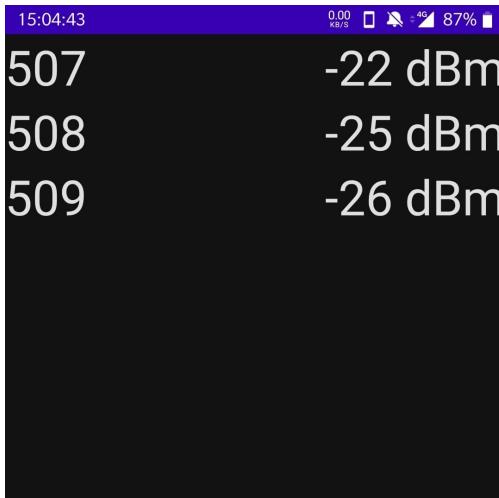


Figure 20: Afprøvning applikation

6.1.2 Friis Ligning verifikation

En verifikation af Friis ligning, vil vi gerne udføre fordi vi endnu ikke ved om det vi kom frem til følger Friis ligning. Vi har også tidligere nævnt Friis ligning i 4 og nu vil vi så verificere den i denne afprøvning i forhold til vores beacons. Det som vi konkret gør, er at vi har 3 beacons og telefon som opfanger signalstyrken i dBm fra vores beacons. Derfor står vi med forskellige afstande mellem vores beacons og telefonen, hvorefter vi til sidst så har en række gennemsnitlige observationer, over 15 afrundede observationer på samme afstand. Således har vi en specifik afstand samt den gennemsnitlige signalstyrke der blev opfanget på den afstand. Disse to parametre plottes så i et scatter plot med afstand på x aksen og signalstyrke på y. Ifølge Friis ligning bør vi få plottet en graf som ligner 3. I tabel: 1 ses de værdier vi fik fra vores afprøvning og i figur: 21 har vi fået plottet de tilsvarende grafer

Afstand	Beacon1 dBm	Beacon2 dBm	Beacon3 dBm
3	-60	-70	-70
6	-61	-68	-68
9	-59	-68	-68
12	-69	-76	-76
15	-73	-83	-82
18	-75	-83	-85
21	-80	-88	-89
24	-83	-93	-93
27	-86	-93	-93
30	-84	-91	-96
33	-86	-97	-96
36	-82	-95	-95
39	-85	-92	-94
42	-90	-96	-100
45	-89	-99	-95

Table 1: Afprøvnings data

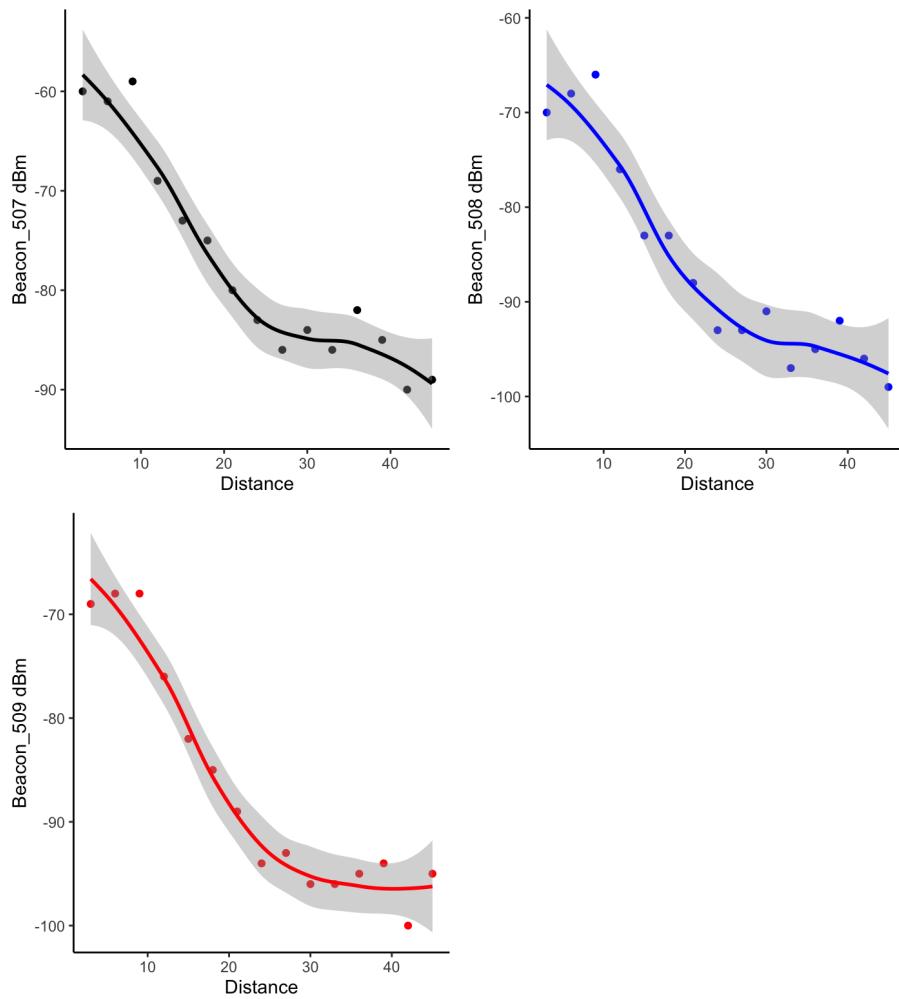


Figure 21: Scatterplots af hhv. beacon nr. 507, 508 og 509 ift. afstand med tilnærmede linjer

Ud fra visuel aflæsning kan vi se, at de 3 grafer til en vis grad ligner det som vi bør få ifølge Friis ligning 4, som er vist i figur 3. For at konkretisere dette vil vi undersøge dæmpningskoefficienten 6. For at udregne K(d) bliver vi nødt til at omskrive dette igen.

$$P_R = P_T \cdot \frac{K}{d^n} \quad (7)$$

Denne skal nu igen omskrives, da vi gerne vil repræsentere vores resultat på en lineær graf.

$$\log_{10}(P_R) = \log_{10}(P_T \cdot K) - 10 \cdot n \cdot \log_{10}(d) \quad (8)$$

Deraf skulle vi få en lineær graf, hvorfaf vi kan lave en lineær regression og aflæse hældningen, da denne vil være vores n også kaldet dæmpningskoefficient. Dvs. at vi skal transformere vores distance med 10 tals logaritmen. Dette resultat kan ses på figur: 22

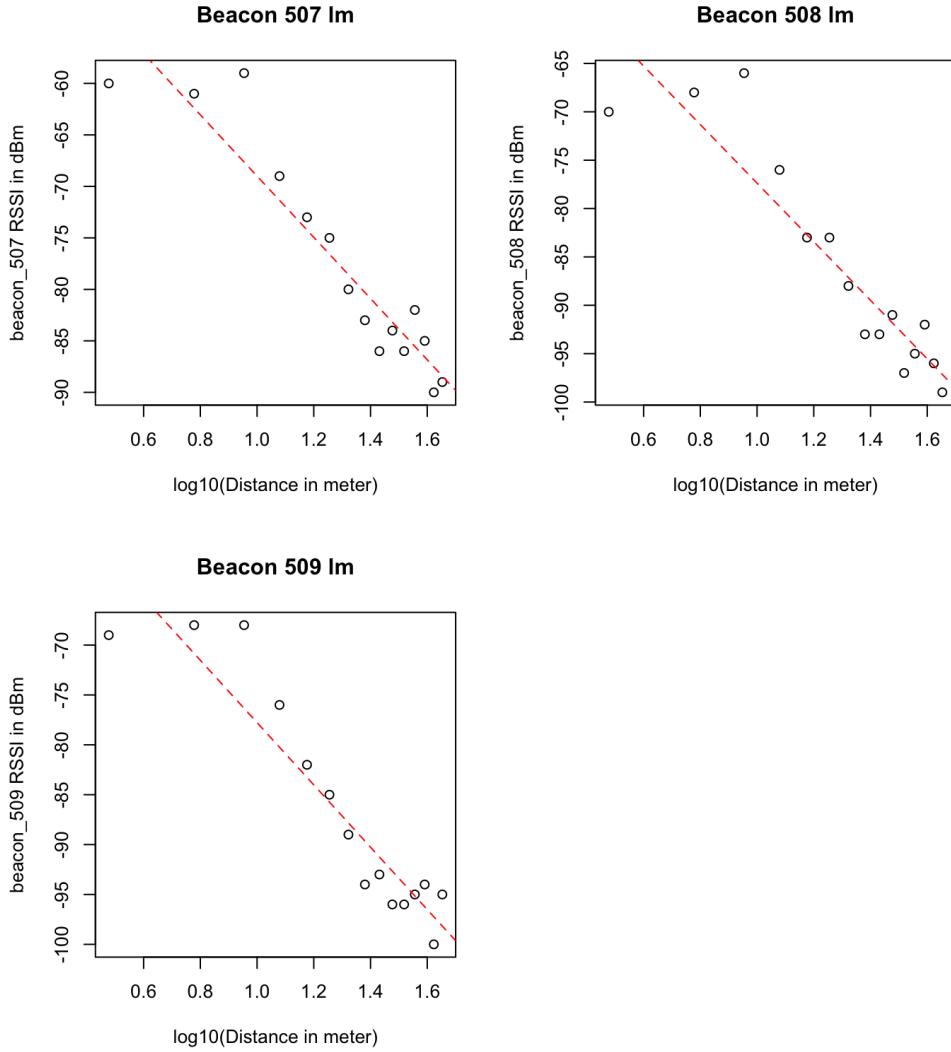


Figure 22: Logaritme plot af RSSI og meter

Regressionsværdierne følger formel i ligning: 9 og kan ses i tabel: 2

$$y = A + Bx \quad (9)$$

Beacon nr.	A	B
Beacon_507	-39.26	-29.93
Beacon_508	-47.10	-30.27
Beacon_509	-46.54	-31.23

Table 2: Koefficienter af regression

Heraf kan vi aflæse vores hældning af de 3 forsøg og deraf tage gennemsnittet og bruge denne som vores hældnings, vi skal dog lige huske at transformerer den tilbage fra logaritmen og $(\cdot(-1))$ dvs. at vi skal gøre følgende med vores resultat:

$$\frac{hældning}{-10} = dæmpningskoefficienten \quad (10)$$

Deraf kan vores hældnings gennemsnit aflæses på ligning: 11 og dæmpningskoefficienten aflæses på ligning: 12

$$\frac{1}{3} \sum hældning = -91,43 \quad (11)$$

$$\frac{hældning}{-10} = 3,048 \quad (12)$$

6.1.3 Lokalisering afprøvning

Da vi gerne vil bruge så få beacons som muligt, vil vi gerne prøve at lokaliserer en person ved brug af 3 beacons. Den måde vi vil afprøve dette, er ved at placerer 3 beacons i et aksesystem således at vi kan ændre på placeringen af brugerens og triangulerer placeringen via Rstudio. Det koordinatsystem vi oprettede, havde beacons placeret i (10,10), (10,40), (40,25) og en bruger placeret som på figur: 23. Deraf brugte vi vores dBm målings applikation til at måle de forskellige signalstyrker for de forskellige positioner. Disse målingsværdier kan findes på i tabel :3

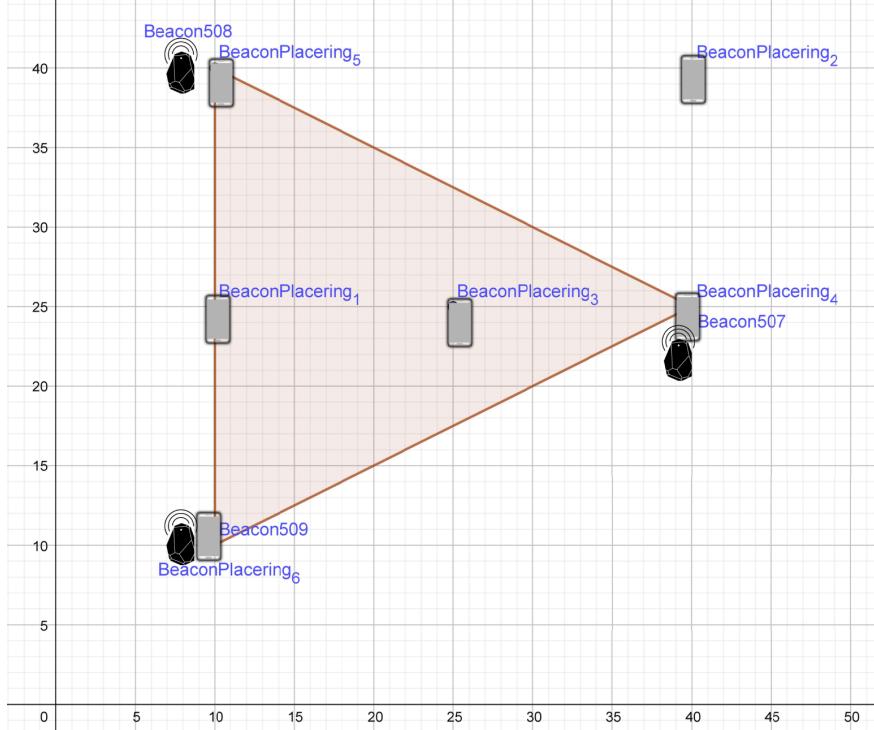


Figure 23: Koordinat system

Placering	Beacon 507 dBm	Beacon 508 dBm	Beacon 509 dBm
1	-79	-83	-78
2	-84	-85	-94
3	-81	-85	-84
4	-47	-82	-89
5	-87	-62	-88
6	-76	-89	-49

Table 3: Afprøvnings værdier Triangulering

Disse værdier kan vi nu skrive ind i Rstudio og prøve at udfører triangulering. Vi kan med den før udarbejde hældning udregne afstanden således:

$$d = 10 \cdot \left(\frac{RSSI - (A)}{-10 \cdot n} \right) [21] \quad (13)$$

Her beskriver A vores RSSI på en reference afstand af 1 meter samt n der beskriver hældningskoefficienten for den enkelte beacon.[21] Dette kan ses i en tabel på tabel:

Beacon Nr.	dBm 1 m	n
507	-63	2.993
508	-71	3.027
509	-72	3.123

På denne måde kunne vi omregne RSSI værdien til distance som svarer til vores graf hvor forholdet et 5:1 dvs. at 5 på grafen svarer til 1 meter.

Ud fra dette kan 3 ligninger opskrives i et ligningssystem, idet vi har koordinaterne samt afstandene altså RSSI.[22] De 3 ligninger opstilles på følgende måde:

$$(x - x_1)^2 + (y - y_1)^2 = d_1^2 \quad (14)$$

$$(x - x_2)^2 + (y - y_2)^2 = d_2^2 \quad (15)$$

$$(x - x_3)^2 + (y - y_3)^2 = d_3^2 [22] \quad (16)$$

Og da vi allerede kender koordinaterne for beacons og kan udregne distancen til de forskellige beacons, kan man trække ligning 16 fra 14 og 15. Således at man nu har et ligningssystem med 2 ligninger uden eksponenter. Ud fra dette kan man opstille en matrix og løse den vha. Gauss Elimination. Efter man har gjort dette, får man netop 2 værdier. Man får et x-koordinat og et y-koordinat. Dette koordinatsæt sammenligner vi så med den egentlige placering på figur: 24. Dette har vi udført I Maple og i bilag 1 figur:28 kan vores Maple script ses.

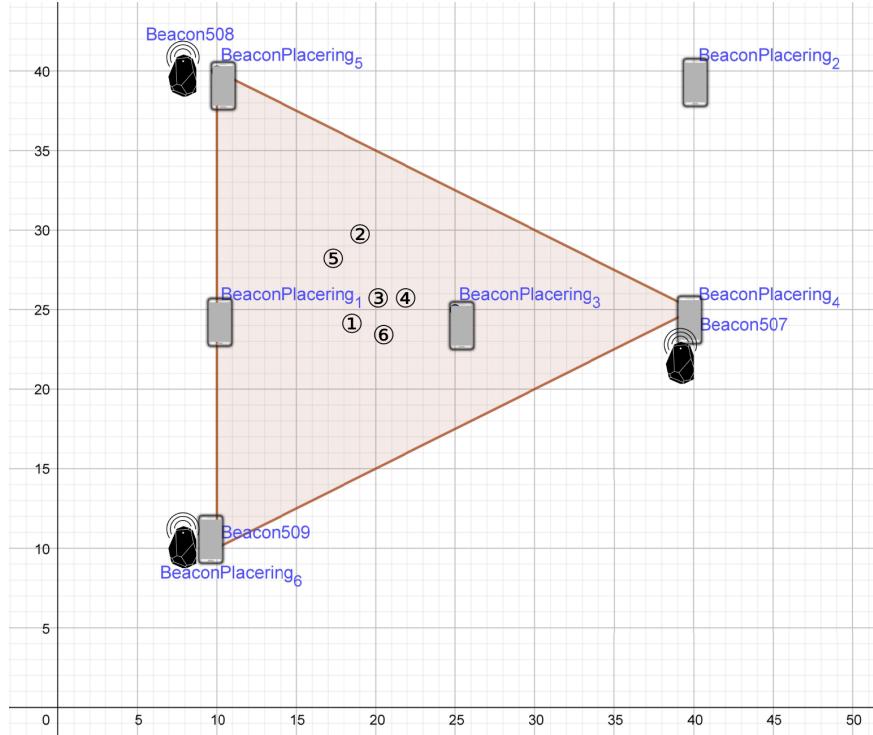


Figure 24: Udførte triangulering

Efter at have udført beregningerne, sammenlignede vi de beregnede koordinatsæt med de egentlige koordinatsæt. Ud fra dette kom vi frem til at der er en relativ stor variation. Dette kan skyldes nogle fejlkilder, som at antennerne måske er bøjet eller beskadiget, eller en forskel på transmitteret signalstyrke.

Derfor har vi vurderet at med de beacons vi har, ville det ikke give mening at udføre en triangulering og dBm, så derfor ville det give mere mening at bruge estimote zoner.

6.2 Usability Test

Vi vil udføre en opgavebaseret usability test med følgende case:

Case

- 1. Tag din android mobil enhed frem.
- 2. Åben Kort App.
- 3. Stå ved v1.01.
- 4. Indtast destinationen: X1.32.
- 5. Få en rute frem.
- 6. Følg navigationen frem imod destinationen.
- 7. Ankom ved X1.32 og afbryd navigationen.

Undervejs observeres forsøgspersonen, hvorefter forsøgspersonen spørges ind til selve bruger oplevelsen under udførelsen af den opgavebaserede usability test.

6.2.1 Brugernes oplevelser

Vi fik følgende respons fra brugerne.

- Bruger 1: ”Jeg glemte at bruge kortet og havde svært ved at forstå applikationen”. ”Jeg blev også forvirret over at den blå plæt ikke bevægede sig som jeg var vant til fra google maps”.
- Bruger 2: ”Jeg synes layoutet var flot og kunne godt lide applikationen”.
- Bruger 3: ”Jeg synes applikationen var nem at forstå og intuitiv. Dog forventede jeg at en rute blev foreslæbt ved indtastning af lokale nr. uden behov for at trykke på start knappen”.

6.2.2 Observerede resultater

Vi observerede følgende fra brugerne.

- Bruger 1: Forvirring ved indtastning af destination i søgerfeltet og hun tog lang tid om at indtaste og starte rute. Hun havde svært ved at forstå hvilken retning hun skulle gå, fordi den blå prik ikke bevægede sig og skærmen ikke orienterede sig i den retning hun skulle bevæge sig. Dog efter at have afprøvet at gå i en forkert retning, korrigerede hun og fandt den rigtige vej.
- Bruger 2: Under vej findingen trykkede hun på skærmen hvilket gjorde at ruten forsvandt, dette skabte forvirring, men hun løste hurtigt problemet selv. Udover dette havde hun ikke meget besvær med brug af applikationen og fandt hurtigt frem til sin destination.
- Bruger 3: Hun missede fuldstændig den store grønne ”Start rute” knap under søgerfeltet og skulle bruge hjælp for at fortsætte. Dog da hun fik rutevejledningen i gang, så havde hun nemt ved at finde hvilken vej hun skulle gå. Hun ankom til destinationen uden besvær, dog så stoppede hun op lidt over halvvejs igennem, da Blue Dot ikke bevægede sig overhovedet fra start positionen. Dog fulgte hun fint ruten og ankom til destinationen.

Alle testpersonerne fandt succesfuldt frem til deres destination ved brug af applikationen, dog var der få tilfælde hvor de skulle bruge hjælp til at forstå appen. Ved flere tilfælde fik testpersonerne ikke en notifikation når de gik ind i en zone og derfor ikke opdateret deres lokation.

6.3 Effekt Afprøvning

Da vi gerne vil undersøge hvorvidt vores løsning forbedrer noget, vil vi lave en effekt afprøvning. Denne afprøvning vil fungere ved at en forsøgsperson går en rute med vores løsning og en rute uden vores løsning. Disse ruter skal selvfølgelig være sammenlignelige. Vi vil primært måle på 2 parametre:

- A3. Tid.
- A4. Bruger oplevelsen.

Vi har tænkt os at gøre det på følgende måde:

- A3. Vi sammenligner tiden det tager at komme fra punkt A til punkt B med og uden vores løsning. Dvs. vi foretager egentlig en fortsættelse af det som vi testede med vores mechanical turk, bare hvor vi nu har udviklet et konkret produkt der kan bruges.
- A4. En rangering fra 1 til 10 om hvordan oplevelsen var, bliver testpersonerne spurgt om både ved brug af vores app og da de ikke brugte vores app. Dette kan så bruges til at sammenligne user experience

6.3.1 Resultater fra effekt Afprøvningen

Efter udførelsen af testen kom vi frem resultaterne som kan ses i tabellen nedenfor.

1. Rute uden App: V1.05 - T1.14 - 73 m
2. Rute med App: V1.05 - X.1.32a - 79 m

Testperson	Tid 1	Bruger Oplevelse 1	Tid 2	Bruger Oplevelse 2
1	1:15	10	0:54	7
2	1:25	5	1:19	7
3	1:03	10	1:24	7
4	1:58	3	1:23	8
5	2:35	2	0:55	10
6	2:28	3	1:35	7
7	2:36	5	1:24	10
8	1:24	8	1:10	7
9	1:33	4	1:09	9
10	1:16	7	1:09	9

Table 4: Effekt Afprøvning

Ud fra disse værdier har vi konstrueret følgende grafer i Rstudio figur: 25. Figurerne viser at vi både sparer tid og opnår en højere oplevelst brugeroplevelse med vores applikation end uden vores applikation i afprøvningen. Præcisist

sparrer vi 29,54 procent af tiden og opnår en 42,11 procent højere oplevet brugeroplevelse, disse udregninger kan man se i ligning: 17 og ligning: 18.

Effektudregning af sparret tid i sekunder på baggrund af gennemsnittet af de to test.

$$\left(\frac{105.3}{74.2} - 1 \right) \cdot 100 = -29,54\% \quad (17)$$

Effektudregning af effekt på brugeroplevelse på baggrund af gennemsnittet af de to test.

$$\left(\frac{5.7}{8.1} - 1 \right) * 100 = 42.11\% \quad (18)$$

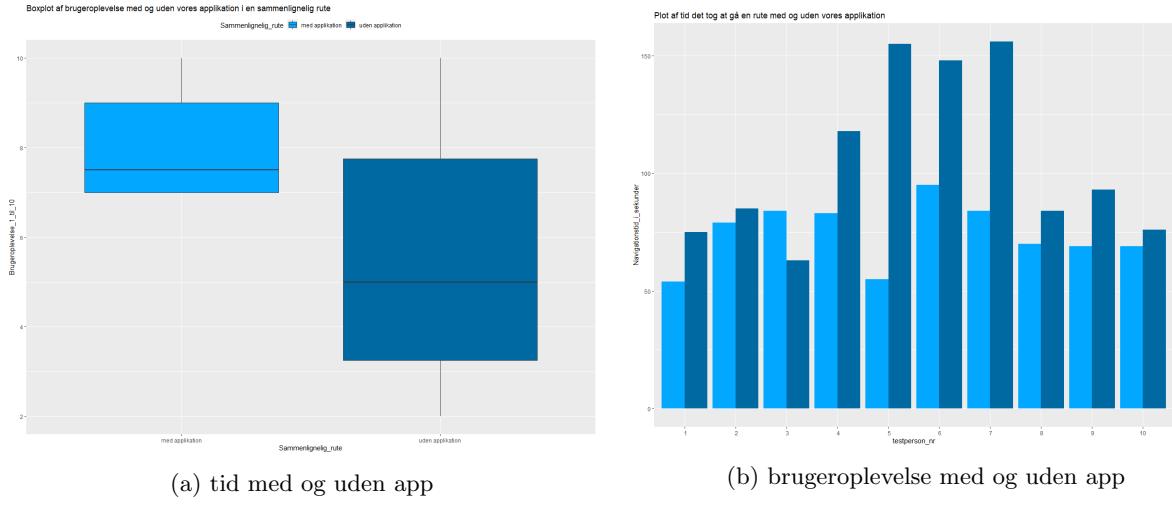


Figure 25: Plots fra effekt afprøvning

7 Diskussion

I dette afsnit vil vi diskutere samtlige aspekter af vores projekt. Dette indebærer alt fra de enkelte elementer i vores produkt, til omkostningerne for vores produkt.

7.1 Estimeret gevinst

Da vi i afsnit 2.1.2 om økonomisk byrde estimerede, at det danske sundhedsvæsen årligt bruger omtrent mellem 55,5 til 103 millioner kroner på vejvisning, vil vi vurdere hvilken eventuel økonomisk gevinst vi kan levere. Dette vil vi regne ud fra de 54.675.000kr, som er baseret på at sundhedspersonalet benytter 4500 timer på vejvisning årligt.

7.1.1 Besparelse pr. hospital

Når det kommer til en eventuel besparelse, er compliance en relevant parameter, idet vi naturligvis ikke forventer at applikationen opnår 100% compliance, og at vi frigør alle 4500 timer sundhedspersonalet årligt bruger på vejvisning. Dog er det heller ikke muligt at vise hvad vores compliance helt konkret bliver på nuværende stade, og vi kunne heller ikke finde nogle valide kilder omkring lignende situationer og deres compliance rater. Derfor har vi således valgt at antage en compliance rate på 50% på baggrund af positive resultater fra effektvurdering og knap så positive resultater fra usability testen. Således har vi en compliance rate at regne ud fra med. Dvs. at beregningerne skal tages med et gran salt. I øjeblikket benyttes der gennemsnitligt 4500 arbejdstimer på et sygehus ift. vejvisning, som nævnt tidligere, med en timeløn på 243 kr., hvilket vil sige at der årligt pr hospital bruges følgende beløb på vejvisning:

$$4500 * 243kr. = 1.093.500kr. \quad (19)$$

Dette vil sige på baggrund af tidligere nævnte informationer, bruges der over 1 million kroner pr hospital i form af vejvisning. På baggrund af den antagne compliance rate, kan vi også estimere, hvor stor en andel af vejvisningen på vores produkt vil overtage:

$$100\% - 50\% = 50\% \quad (20)$$

Denne procentsats kan nu bruges til at beregne at antallet af timer, som sundhedspersonalet bruger på vejvisning i et hospital falde til:

$$4.500 \text{ timer} * 0,50 = 2.250 \text{ timer} \quad (21)$$

Dette vil ud fra beregningerne, resulterer i en økonomisk besparelse på følgende, inden vi selvfølgelig medregner prisen for vores løsning:

$$4.500 * 243kr. - 2.250 * 243kr. = 546.750kr. \quad (22)$$

Dvs. på baggrund af vores estimer og antagelser, er vi kommet frem til at et hospital kan spare over en halv million kroner årligt ift. vejvisning. Dog skal førstegangsydelsen af vores løsning og 12 gange det månedlige kontingent også inkluderes. Førstegangsydelsen er afhængig af faktorer som antal beacons og installationsomkostninger, hvilket er meget forskelligt fra hospital til hospital. Denne varians kan skyldes alt fra arealstørrelsen på selve hospitalet til om de fx kun ønsker beacons i specifikke rum eller om de vil have beacons i gangene osv. Det månedlige kontingent afhænger også af diverse faktorer, som bla. lønninger samt drift og support. Dette vil sænke den økonomiske besparelse. Dog sigter vi naturligvis mod at de finansielle omkostninger, ved vores løsning, ikke vil påvirke besparelsen på en betydningsfuld måde. Vi er jo netop klare over at hospitalerne ønsker at se en signifikant besparelse for at de vil investere i vores løsning. Derfor efterstræbes dette naturligvis. Så dette kan resultere i at de 2250 spade arbejdstimer pr hospital, kunne bruges på noget som fx at forbedre borgerens ophold, ellers kunne besparelsen på en halv million bruges på noget andet. Den eventuelle økonomiske gevinst er helt klart også noget, som kan bidrage til salg.

7.1.2 Besparelse på nationalt plan

Det tidligere nævnte estimat er lavet pr. hospital. Hvis man på landsplan også gøre brug af tidligere nævnte antagelse ift. compliance af produktet, kan vi endda komme frem til en estimeret økonomisk besparelse i hele landet. Så hvis vi antager at 50 procent af arbejdstimerne der bruges på vejvisning kan erstattes ude på landets over 50 hospitaler, kan vi ud fra den antagelse beregne at den samlede estimerede økonomiske besparelse vil lægge på følgende beløb:

$$546.750kr. \cdot 50 = 27.337.500kr. \quad (23)$$

Dvs. på baggrund af vores estimer og antagelser, er vi kommet frem til at på nationalt plan kan der bespares over 27 millioner kroner årligt ift. vejvisning. Dette er et beløb der sagtens kunne bruges på andet. De spade arbejdstimer kunne også bruges på andre relevante aktiviteter, når vores løsning netop går ind og erstatter en andel af timerne brugt på vejvisningen. Dog er det igen vigtigt at bide mærke i at dette beløb ikke inkluderer omkostningerne ift. vores løsning. Men som sagt tidligere sigter vi imod at omkostningerne ikke har en betydningsfuld effekt på den estimerede besparelse.

7.1.3 Mulig internationalisering

En eventuel økonomisk gevinst på nationalplan af denne størrelsersart er muligvis også noget, som kan benyttes ift. at dokumentere løsningens validitet med de udførte test. Dette kunne være gavnligt, hvis vi på sigt vil forsøge at sprede løsningen ud over landets grænser fx til Norge og Sverige. Der vil et lignende regnskab muligvis komme os til gode.

7.2 Bæredygtighed

Vi stræber mod at udvikle en bæredygtig løsning. Derfor er der nogle kerneprincipper, som vores projektarbejde tager udgangspunkt i. Disse kerneprincipper er baseret på FN verdensmål. Følgende verdensmål er relevant for dette projekt:

- 3. Sundhed og trivsel.
- 9. Industri, innovation og infrastruktur.
- 12. Ansvarligt forbrug og produktion.
- 13. Klimaindsats.

Tilgangen vi har til disse verdensmål, bliver konkretiseret ift. de relevante bæredygtige aspekter for vores projektidé, som gennemgås nu.

7.2.1 Sociale aspekter

Før Ved den nuværende løsning kan en gæst på hospitalet henvende sig ved receptionen og få information, om hvor de skal gå hen for at finde deres familie/venner. Dette giver en lille menneskelig kontakt, og giver også den besøgende mulighed for at stille andre spørgsmål eller komme af med bekymringer.

Vores løsning Løsningen vil kunne tilgås gennem receptionen, en plakat eller et skilt. Dette kunne fjerne noget menneskelig kontakt men det kunne også bidrage til mere tryghed ved den instruks man får ved receptionen, da man har noget at læne sig op ad hvis man glemmer eller begår fejl. Derved mener vi, at det til en vis grad er med til at bidrage til sundhed og trivsel (Mål 3), idet vi forbedrer borgernes oplevelse når de tager på et hospital pga. enten sin egen eller en pårørendes sundhedstilstand.

Økonomisk Derudover er der er også et andet aspekt udover det sociale, altså det økonomiske. Da der ikke er en eksisterende løsning der udkonkurreres, vil den primære økonomiske effekt være, at hospitalet skal betale for denne løsning og derfor kan der laves et argument for, at ressourcerne skulle placeret på noget vigtigere, såsom at redde menneskeliv eller bedre arbejdsvilkår.

7.2.2 Miljømæssige aspekter

Før Den nuværende navigations løsning på hospitalet har ikke rigtig nogen indflydelse på miljøet. Der kunne dog tales om et arbejdsmiljø eller trafik miljø. Altså gangene på hospitalet deles af gæster, patienter, sygeplejersker, håndværkere, teknikere og læger. der alle skal fra a til b.

Vores løsning Trafik miljøet/gang miljøet på hospitalet kunne have færre personer der ikke kan finde vej, og som er nødsaget til at forstyrre og spørge om vej. Yderligere kunne gæster bevæge sig rundt med større autonomi og endda hjælpe hinanden med at finde vej. Dette mener vi lever op til mål 9, idet vi innoverer pålidelig infrastruktur.

Klima Vores løsning vil kræve servere og beacons, som tilsammen vil have et mindre co2 aftryk i form af strøm til serverne og brug af ressourcer forbundet med konstruktionen af beacons. Yderlige hvis en større dataindsamling laves ved hjælp af vores projekt ide, vil dette bidrage til den store energi byrde, som datacentre har. Dette vil have indflydelse på mål 13, og derfor vil det være vigtigt at vi er opmærksomme på at minimere antal beacons og omfanget af dataindsamling. Derudover vil vi sætte krav til vores leverandør, om at produktionen skal foregå så grøn, som realistisk muligt, hvilket lever op til mål 12. Samt vil vi med vores løbende service og drift, forsøge at få hospitalerne til at beholde de samme beacons i så lang tid som muligt, hvilket igen bidrager til mål 12.

7.3 Etiske overvejelser

Det er essentielt for nye løsninger, specielt løsninger på det sundhedsfarlige område, at få kortlagt de relevante etiske problemstillinger der skulle foreligge sig. Det er vigtigt at gøre sig nogle overvejelser herom, så der ikke opstår uhen-sigtsmæssigheder. Under udarbejdelsen af projektet har der været en række etiske overvejelser undervejs.

Den første overvejelse er, om der er nogle befolkningsgrupper man taber i udviklingen af løsningerne til udfordringerne om at finde vej. Da applikationen har til hensigt at blive benyttet på hospitaler, kan der forventes at være mange borgere med diverse grader af handikap eller funktionsbegrænsninger nævnes. Svagtseende der kan have svært ved at se kortet, hvor der kunne udarbejdes en svagtseende mode, der tager hensyn ved at forstørre tekst, billeder etc. Blinde der bruger mobil med stemmeføring, hvor applikationen kan fungere i samarbejde med Google assistance etc. Personer der har nedsat funktionsevne, der kan have en række forskellige behov og skal have mulighed for enten pauser undervejs, navigering igennem elevatorer i stedet for trapper eller navigering ved brug af diverse transportmuligheder eller services hospitalet stiller til rådighed. Der er også en tendens til at den ældre del af befolkningen ikke er vant til brug af mobile enheder, eller slet ikke ejer nogen, og at de derfor vil blive tabt, hvis mobile enheder delvist erstatter de løsninger de er vant til at bruge. Spørgsmålet i projektet har været, om hvor vidt disse overvejelser skulle implementeres i vores løsning og hvilken effekt de vil have på de målgrupper de er rettet imod. For at kunne implementere de 3 nævnte eksempler, vil det også kræve en yderligere undersøgelse om, hvor stor en målgruppe eksemplerne drejer sig om. Om det er problematisk for disse målgrupper at finde vej, eller om der allerede findes effektive løsninger for disse målgrupper og at derfor ikke er nogle reelle udfordringer for disse.

Lagring af personfølsomme data er også et relevant etisk spørgsmål. Hvis personer der har til hensigt at misbruge dataene, kan få fat på viden omkring folks ageren på hospitaler og måske misbruge den viden - er spørgsmålet så, hvilke konsekvenser det vil have for ofrene og deres omgivelser? Desuden er det også et spørgsmål, om hvor vidt man overhovedet skal logge dataene og helt undgå misbrug af data. Det er også en vejning om der skal bruges ressourcer på at imødegå misbrug af dataene, hvis man vælger at logge data. Det er desuden heller ikke en del af projektrammen, at skulle hav fokus på IT-sikkerhed.

Der eksisterer altid et etisk spørgsmål når der udvikles støtteværktøjer, nemlig giver dette værktøj brugerne mere autonomi og empowerment eller bliver de gjort afhængige af værktøjet. Hvis vi tager udgangspunkt i brugere som har daglig gang på et hospital, kunne det derfor forestille sig at vedkommende egentlig på den lange bane bruger mere tid på at bruge appen og derefter finde vej, i stedet for bare at bruge sin intuitive navigering og hukommelse.

Et andet etisk aspekt er også, om appen bidrager til, at brugere holder deres fokus ned på deres telefon og ikke er opmærksomme på deres omgivelser. Det kan tænkes at der også vil opstå situationer, hvor folk kommer til at stå i vejen eller placere sig uheldigt med deres mobil i hånden. Et hverdagseksempel kunne være at folk stødte ind i hinanden og nogen kom til skade. Et mere ekstremt eksempel kunne være at et hjertestophold bliver bremset af brugeren, fordi vedkommende står i vejen.

7.4 Usability test

Vi har lavet en opgavebaseret usability afprøvning og kom frem til en række forskellige resultater vi gerne vil diskutere.

7.4.1 Valg af brugere

Da vi havde mulighed for et begrænset udvalg af brugere, måtte vi vælge brugere fra vores egen klasse, der også arbejder med beacons, til at repræsentere vores brugere. Derfor har de kendskab til hvordan beacons-systemet burde virke, hvilket ikke er repræsentativt for den bruger vi gerne vil ramme. Undervejs brugte vores valgte brugere deres energi på at undre sig over Implementation af beacons, hvilket umiddelbart ikke kan forventes af vores ønskede brugere. Yderlige påvirkes den generelle feedback fra de valgte brugere af at disse har venskabelig relation til projektgruppen, hvilket betyder at der kan introduceres et yderligere antal af biasens.

7.4.2 Resultat af test

Overordnet kan vi sige at testen gik godt da alle tre forsøgspersonen kom frem til destinationen uden større besvær, vi kom dog frem til en række interessante forbedringspunkter til vores applikation.

Det første man kunne sige er, at vores resultat viser stor forvirring i at det er et todelt produkt. At beacons appen ikke samarbejder med kort appen, men at man er mellemleddet mellem de to. Dette betyder nemlig at notifikationen fra beacon appen, ikke har noget med kort appen at gøre, og hvis man ikke observere denne notifikation, vil man ikke ændre position, og så vil man ikke kunne bruge vores produkt som tilsigter.

Dernæst forventede testpersonerne i flere tilfælde at ruten automatisk ville opdatere sig selv da de trykkede på et lokale eller indtastede et lokale nummer. Dette gjorde de pga. deres forståelse for interaktive kort, der kommer fra f.eks. Google Maps. Når applikationen ikke stemmer overens med deres intuition, troede de der var noget galt med applikationen eller at de havde gjort noget forkert. Dette viser en stor mulighed for forbedring da vi kan tilføje den funktionalitet de forventer uden at ofre andre elementer af applikationen.

Det er et centralt element af vores applikation at vise brugerens placering med den blå prik på kortet. Dog har testpersonerne, som tidligere nævnt, en forståelse for hvordan denne skal opføre sig pga. tidligere erfaringer fra lignende applikationer. Vi har delvist implementeret funktionalitet på samme måde som lignende applikationer, den fungere dog ikke lige så flydende. Dette bliver højest sandsyndeligt forbedret ved at bruge flere beacons, da vi i vores tilfælde kun brugte tre. Yderligere kunne en mere kompleks algoritme gøre bevægelsen af den blå prik mere brugervenlig. Til sidst kunne vi også tilføje en orientering af kortet, så det peger den vej man skal bevæge sig.

Det næste vi oplevede var, at nogle brugere havde svært ved enten at finde nogle elementer, eller forstå hvad de forskellige elementer gjorde. Derfor kunne vi forbedre vores layout og UI til at være mere præcis og tydelig i dets funktion. Noget af dette vil blive løst.

7.5 Effekt afprøvning

På baggrund af resultaterne fra effektafprøvningen, har vi fundet at vores applikation giver en 42 procent forbedring i tiden det tager at navigere rundt og 29.5 procent forbedring i den tilhørende brugeroplevelse. På trods af det lave antal forsøgspersoner, vil vi mene at disse resultater er markante.

Udførelsen af afprøvningen gik som planlagt, der var dog stor variation i testpersonernes præstation, tre forsøgspersoner var to og et halvt minut om at navigere 73m uden brug af vores app, hvilket selvfølgelig betydeligt har bidraget til

vores positive effekt. Man kan diskutere om dette er et heldigt tilfælde, eller om det afslører en problemstilling og en potentiel målgruppe, som netop vores applikation kan hjælpe.

Den generelle fremgangsmåde, brugt af testpersonerne, til at gå den første rute uden applikationen var at finde det nærmeste kort, orientere sig, memorere ruten og så navigere. Der var dog et par testpersoner som blindt gik forbi det 2 meter brede og 3 meter høje vægkort for derefter at bruge eliminationsmetoden til at finde vej. Enkelte personer var så stressede, at de i øjeblikket ikke kunne lokalisere lokalet på kortet, dog lykkedes det alle med at finde vej.

Når vi sammenligner dette med den store tidsforsk, mellem de to ruter, kan vi argumentere for, at der eksisterer et potentiale for forbedring pga. variation i menneskelig adfærd og evne.

Ved ruten, med hjælp fra applikationen, så vi en mindre variation i tid, men også den hurtigste tid og det hurtigste gennemsnit. Næsten alle testpersoner fulgte ruten perfekt, som applikationen vejledte, og kom hurtigt afsted efter at have orienteret sig. Der er en række faktorer der kunne medvirke til disse positive resultater, og det kan derfor diskuteres om disse har haft effekt under vores afprøvning. Ruterne var forskellige, og selvom vi sørgede for at ruten de gik med vores applikation var længst, kan der stadig være ukendte uligheder mellem de ruterne. Den første test kunne også fungere som en form for opvarmning for testpersonerne, hvilket mentalt kunne forbedre deres retningssans. Der kunne også eksistere ukendte bias og motivationer blandt testpersonerne, der kunne påvirke deres præstation.

7.5.1 Kvalitative observationer

Under udførelsen af effektafprøvningen, som egentlig er en kvantitativ afprøvning, kunne vi observere et par interessante ting. Først og fremmest gik samtlige testpersoner langsommere når de navigerede med telefonen. Dvs. på trods af den langsomme ganghastighed, fandt majoriteten af testpersonerne alligevel hurtigere frem til deres destination med hjælp fra applikationen. Samtlige brugere havde en stærk intuition om hvordan en kort/navigations app som vores fungere, og var derfor exceptionelt hurtigere til at forstå og bruge den. Her skal det dog tages med, at det er unge studerende på DTU, som er testpersonerne.

7.5.2 7 ud af 10 foretrak at bruge appen

Under registreringen af brugeroplevelsen fandt vi ud af, at der var 3 forsøgspersoner som registrerede en højere brugeroplevelse uden vores applikation, end med. Dette er selvfølgelig ikke et godt resultat. Dog vil vi mene, at hvis vi kigger dybere på disse personers resultater, var det primært personer som havde rigtig nemt ved at finde rundt uden nogen form for vejledning. Forsøgspersonerne havde generelt en lavere navigationstid end gennemsnittet, og gav ikke vores applikation en lav brugeroplevelse, de havde simpelthen bare ikke brug for nogen form for vejledning i navigationen. Dette har vi dog også taget med i vores estimeret gevinst afsnit: 7.1, da vi her kun regner med en 50% compliance, da der vil være en befolkningsgruppe der ikke har behovet for applikationen.

7.6 Arbejdsbeskrivelse

I dette afsnit vil vi beskrive hvordan vi har struktureret projektet og om vores arbejdsproces.

7.6.1 13-ugers

Under 13-ugers forløbet har fokuset været på at skabe et overblik over projektet og undersøge en løsning til en problemstilling vi vil arbejde med i 3-ugers perioden. Her er tiden blevet brugt på at undersøge, designe og udvikle en applikation, der ved brug af Ibeacons kan lave indehørs navigation med henblik på at blive brugt på hospitaler. Derudover er tiden blevet brugt på at forberede fremlæggelserne til Conceive Design Seminar og Design Build Seminar, hvor vi har fået kritisk revision/feedback af vores undervisere og en virksomhedsrepræsentant.

7.6.2 3-ugers

Under 3-ugers perioden har vi kunnet tage ordenligt hul på projektet og kunne få det ført projektet fra ”ide til virkelighed”.

Gruppekontrakt

I den spæde start af 3-ugers projektarbejdet, var alle gruppemedlemmer enige om at bruge godt med tid til at udarbejde en gruppekontrakt. Gruppekontrakten har til formål at skabe dialog om gruppearbejdet samt fastsætte de

rammer vi som gruppe gerne vil arbejde under. Håbet er at skabe et arbejdsmiljø, der fremmer arbejdsprocessen og konstruktivitet i projektgruppen, samt reducere gnidninger og frustrationer imellem gruppemedlemmerne. Et arbejdsmiljø hvor der er plads til uenigheder og hvor gruppemedlemmerne skal være villige til at indgå kompromiser. Gruppekontrakten indeholder blandt andet emnerne: ambitionsniveau, arbejdstider, konsekvens ved forsinkelse og scrummøder.

Scrummøder

Under udarbejdelsen af gruppekontrakten var der konsensus blandt alle gruppemedlemmer om at have et dagligt scrummøde om morgenen mellem 8:30-9:00 alle dage undtagen weekend. Scrummøderne har til formål at skabe en rød tråd gennem projektarbejdet og give projektmedlemmerne mulighed for at opdatere sig på gruppens arbejdsstatus. For at skabe en effektiv mødegang, er der blevet udarbejdet en skabelon for dagsordnen til scrummøderne for at skabe struktur og forudsigelighed. Derudover er der til hvert scrummøde forudpeget en scrummøde-torvholder, der har ansvar for at scrummødet følger dagsordnen samtidigt med at mødeprocessen er effektiv og at tiden ikke løber fra sig.

Registrering af tid, tidsstyring og planlægning af projektet

For at sikre at projektet kommer i mål til tiden, har vi ved brug af Instagantt lavet et gantt-diagram. Diagrammet er en oversigt over de delopgaver, der udgør det færdige projekt samt den tid de forskellige opgaver kræver for at blive udført. Overordnet er diagrammet delt op i tre hovedopgaver: en struktur over udarbejdelsen af selve projektrapporten, gruppearbejdet som helhed og udarbejdelsen af posteren. Igennem diagrammets tre hovedprocesser er der lavet en kritisk vej, så vi under projektet hele tiden kan revurdere om projektet når i mål til tiden. Undervejs i projektet skal arbejdstimerne registreres dagligt for de enkelte gruppemedlemmer, samt angive hvilke opgaver de daglige arbejdstimer er blevet brugt på.

Uddeling af opgaver

Vi er i gruppen blevet enige om, at gruppemedlemmer selv aktivt skal tage opgaver til sig. Gruppemedlemmerne er begrænset til kun at sidde med én opgave af gangen. Instagantt-diagrammet bliver brugt som værktøj til at holde styr på hvem der har hvilken opgave, hvor gruppemedlemmerne selv kan skrive sig på, hvor langt man er med opgaven og hvilken status opgaven har. Dog er der stadig konsensus blandt gruppemedlemmerne om, at hver gruppemedlem individuelt også skal huske at konferere med de andre gruppemedlemmer om sin opgave undervejs.

8 Konklusion

Vi har hermed bevist at man ved implementering af vores løsning kan implementere et indendørs navigationssystem i miljøer hvor gps signaler ikke er en mulighed, som både er brugervenlig, hurtig og sparre tid 6.3. Vi har yderligere også opfyldt alle ”need to have” krav fra vores kravspecifikation og disse krav er blevet fremvist under afprøvningerne.

Opfyldte funktionelle krav	Opfyldte non funktionelle krav	Ikke opfyldte nice to have
FK1: i afsnit: 6.2	NFK1: i afsnit: 6.3	FK5.
FK2: i afsnit: 6.2	NFK2: i afsnit: 6.2	FK6.
FK3: i afsnit: 6.2	NFK3: i afsnit: 6.2	NFK7.
FK4: i afsnit: 6.2	NFK4: i afsnit: 6.3	
	NFK5: i afsnit: 6.3	
	NFK6: i afsnit: 6.3	

Table 5: Sammenligning med Krav

Vi kan med denne rapport konkludere at vores løsning ikke kan løse navigationsudfordringer på hospitaler, i denne iteration. Vi kan dog konkludere på baggrund af vores usability- 7.4 og effektafprøvning 7.5, at man ved implementering af kompasfunktionalitet, flere beacons og automatisk indsætning af beacon koordinater kan løse navigationsudfordringerne. Dette vil dog være en dyr løsning på baggrund af vores estimeret gevinst 7.1 og estimeret omkostninger 3.5. Dog er vi som gruppe alt i alt tilfredse med vores samlede arbejde samt løsningen, som vi nu er kommet frem til.

References

- [1] Thomas Bach Poulsen. Mona Aaberg. Viborg er som alle andre sygehuse - det er umuligt at finde rundt: Nu skal lokale naturperler gøre det lettere. https://www.dr.dk/nyheder/regionale/midtvest/viborg-er-som-alle-andre-sygehuse-det-er-umuligt-finde-rundt-nu-skal?fbclid=IwAR03RaRsseF038oVPbcVX5501C8omiWaC_bMRZes9a3n73iK04QRMk8UDzw. (accessed: 03-06-2022).
- [2] Mette Schilling. *Psykologi i sundhedsfag*. munksgaard., 2012. (Side 249 til 263).
- [3] Marie Linddahl. Orienteringsløb i goedstrup: Vejvisere øver sig i at finde rundt på det nye kæmpesygehus. https://www.tvmidtvest.dk/herning/orienteeringsloeb-i-goedstrup-vejvisere-oever-sig-i-at-finde-rundt-paa-det-nye-kaempesyghehus?fbclid=IwAR3za02NFgAz-25KnwEeFtSYzr4Q_Pr4Sd7ymrDlnLJ1TQeegT-20f_XCHU. (accessed: 03-06-2022).
- [4] Craig Zimring. The costs of confusion: monetary and non-monetary costs of the emory university hospital wayfinding system." georgia institute of technology paper, 1990.
- [5] Michel Verheem. Id-lab: Wayfinding made simple. beautifully simple.
- [6] sundhedsdatastyrelsen. Over 106.000 fuldtidsbeskæftigede på offentlige sygehuse i 2018. https://sundhedsdatastyrelsen.dk/da/nyheder/2019/beskaeftigede_ved_sygehuse_10042019. (accessed: 03-06-2022).
- [7] sundhed.dk. Kommuner, regioner og sygehuse. <https://www.sundhed.dk/borger/guides/sundhedsvaesenet/regioner-sygehuse-kommuner/>. (accessed: 03-06-2022).
- [8] DST. Befolkningsens løn befolkningenslØndanmarksstatistik. <http://www.dst.dk/pukora/epub/upload/19581/befloen.pdf>. (accessed: 03-06-2022).
- [9] Theodore S. Rappaport. *Wireless Communications*. Pearson, 2002. (Chapter 4).
- [10] Friis equation. <https://www.sciencedirect.com/topics/computer-science/friis-equation>. (accessed: 10-06-2022).
- [11] Bluetooth. Bluetooth. <https://www.bluetooth.com>. (accessed: 17-06-2022).
- [12] Botland. Estimote proximity beacons. <https://botland.store/bluetooth-modules/12201-estimote-proximity-beacon-4pcs-5904422340797.html>. (accessed: 17-06-2022).
- [13] Aroscloud. Mazemappris. Kilde:[https://aroscloud.dk/aros-cloud-leverer-wayfinding-til-skoler-med-mazemap-og-viaso?fbclid=IwAR2W2wxSl3RsBfaQjL7htIEr13IA5-1qsE_9W6P_FMCggxKi3R5WYsFDD0](https://aroscloud.dk/aros-cloud-leverer-wayfinding-til-skoler-med-mazemap-og-viaso). (accessed: 17-06-2022).
- [14] Firebase. Firebase. Kilde:<https://firebase.google.com/pricing>. (accessed: 17-06-2022).
- [15] DavidBritch nchonni-conceptdev JohnCOborne. The model-view-viewmodel pattern. <https://docs.microsoft.com/en-us/xamarin/xamarin-forms/enterprise-application-patterns/mvvm>. (accessed: 04-06-2022).
- [16] FireBase. What is a nosql database? how is cloud firestore structured? get to know cloud firestore. https://www.youtube.com/watch?v=v_hR4K4auoQ&ab_channel=Firebase/. (accessed: 04-06-2022).
- [17] Danimex. Estimote beacons. <https://danimex.com/products?ProductID=PROD1311>. (accessed: 08-06-2022).
- [18] Estimote. Technical specification of estimote beacons and stickers. <https://community.estimote.com/hc/en-us/articles/204092986-Technical-specification-of-Estimote-Beacons-and-Stickers>. (accessed: 08-06-2022).
- [19] DrVipin. Jetpack compoe -1- android notification — jetpack compose tutorials — dr vipin classes. <https://www.youtube.com/watch?v=je3km6-FZmc>. (accessed: 08-06-2022).
- [20] MazeMap. Mazemap js api 2. <https://api.mazemap.com/js/v2.0.66/docs/#intro>. (accessed: 08-06-2022).
- [21] Yapeng Wang. Xu Yang. Yutian Zhao. Yue Liu. Indoor positioning system by using triangulation algorithm. <https://www.ijitee.org/wp-content/uploads/papers/v5i8/H2255015816.pdf>. (accessed: 14-06-2022).
- [22] Ebru Alp. Tamer Dag. Taner Arsan. Bluetooth positioning using rssi and triangulation methods. https://www.researchgate.net/publication/261056426_Bluetooth_positioning_using_RSSI_and_triangulation_methods. (accessed: 14-06-2022).

9 ByrdeFordeling

Indholdsfortegnelse	Lavet af:
Abstract	Naveed
1 Formål	Jesper
2 Indledning	
2.1 Baggrund	Shiv og Troels
2.1.1 Komplekse hospitaler	Shiv
2.1.2 Økonomisk byrde	Troels
2.2 Problemformulering	Jesper, Naveed og Troels
2.3 Kravspecifikation	Alle
2.3.1 Funktionelle krav	Jesper
2.3.2 Non-funktionelle krav	Jesper
2.3.3 Eksklusion	Jesper
3 Analyse	
3.1 Tidligere Afprøvning af Mechanical Turk	Andreas og Jesper
3.2 System	Andreas
3.2.1 Mobil Enhed	Andreas
3.2.2 Indendørs Navigation Metode	Andreas
3.2.3 Lagring af Data	Andreas
3.2.4 Analyse KlasseDiagram	Andreas
3.2.5 System beskrivelse	Andreas
3.3 Beacon Navigation	Shiv og Troels
3.3.1 Placering	Shiv
3.3.2 Lokaliserering	Shiv og Jesper
3.3.3 Friis ligning	Troels
3.3.4 Dæmpningskilder og dæmpningskoeficient	Troels
3.3.5 Navigations algoritme	Shiv
3.3.6 Alternative løsningsmuligheder	Shiv
3.4 Business	Shiv
3.4.1 Business Model Canvas	Shiv
3.4.2 Customer Segments	Shiv
3.4.3 Value Propositions	Shiv
3.4.4 Key Activities	Shiv
3.4.5 Key Ressources	Shiv
3.4.6 Key Partners	Shiv
3.4.7 Customer Relationships	Shiv
3.4.8 Channels	Shiv
3.4.9 Cost Structure	Shiv
3.4.10 Revenue Streams	Shiv
3.5 Estimeret omkostninger	Jesper
3.5.1 Bluetooth	Jesper
3.5.2 Estimote Proximity beacons	Jesper
3.5.3 Mazemap	Jesper
3.5.4 Firebase	Jesper
3.6 UserExperience	Naveed
3.6.1 User stories	Naveed
3.6.2 Use case model	Naveed
3.6.3 Generelle overvejelse	Naveed
3.6.4 UI	Naveed
4 Design	Troels
4.1 ModelViewViewModel	Troels
4.1.1 Vores MVVM	Troels

4.2 Beacon Lokalisering	Troels
4.2.1 Estimote SDK	Troels
4.2.2 Firestore Database	Troels
4.3 Mazemap API	Troels
4.3.1 Problem	Troels
4.3.2 Alternative løsninger	Troels
4.3.3 Vores løsning	Troels
4.3.4 Fremtidsperspektiv	Troels
4.4 Diagrammer over hele systemet	Troels
4.5 UI	Naveed
4.5.1 Navigationside	Naveed
4.6 BeaconMap	Jesper
4.6.1 Udvalg af beacons	Jesper
4.6.2 Proximity Beacon	Jesper
4.6.3 Plantegning	Jesper
5 Implementation	
5.1 Beacon-Cloud'-Credentials	Shiv
5.2 Beacon opsætning	Shiv
5.3 Proximity-ZoneEventViewModel	Shiv
5.4 Notifikation	Shiv Troels
5.5 FireStore - Firebase	Troels
5.6 MazeMap	Troels
5.6.1 UI	Troels og Naveed
5.7 Webview	Troels
6 Afprøvning	
6.1 Beacon Signal afprøvning	Shiv og Troels
6.1.1 Applikation til måling af signalstyrke	Shiv og Troels
6.1.2 Friss Ligning verifikation	Shiv og Troels
6.1.3 Lokaliserings afprøvning	Shiv og Troels
6.2 Usability Test	Andreas, Naveed, Shiv og Troels
6.2.1 Brugernes oplevelser	Andreas, Naveed
6.2.2 Observerede resultater	Andreas, Naveed
6.3 Effekt Afprøvning	Andreas, Naveed, Shiv og Troels
6.3.1 Resultater fra effekt Afprøvningen	Andreas, Troels
7 Diskussion	
7.1 Estimeret gevinst	Shiv
7.1.1 Besparelse pr. hospital	Shiv
7.1.2 Besparelse på nationalt plan	Shiv
7.1.3 Mulig internationalisering	Shiv
7.2 Bæredygtighed	Andreas og Shiv
7.2.1 Sociale aspekter	Andreas og Shiv
7.2.2 Miljømæssige aspekter	Andreas og Shiv
7.3 Etiske overvejelser	Jesper
7.4 Usability test	Andreas og Naveed
7.4.1 Valg af brugere	Andreas og Naveed
7.4.2 Resultat af test	Andreas og Naveed
7.5 Effekt afprøvning	Andreas
7.5.1 Kvalitative observationer	Andreas
7.5.2 7 ud af 10 foretrak at bruge appen	Troels
7.6 Arbejdsbeskrivelse	Jesper
7.6.1 13-ugers	Jesper
7.6.2 3-ugers .	Jesper

8 Konklusion

	Naveed, Shiv og Troels
Beaconkode	Andreas, Naveed, Shiv og Troels
NotifikationKode	Shiv og Troels
FirebaseKode	Shiv og Troels
MazeMapKode	Naveed og Troels
Udførsels af Usability Afprøvning	Andreas og Naveed
Udførsels af Effekt Afprøvning	Andreas, Naveed, Shiv og Troels
Udførsels af Friss ligning/Triangulering	Naveed, Shiv og Troels
Udarbejdning af poster	Naveed, Shiv og Troels
Udarbejdning af grafer til Effekt i R	Andreas
Udarbejdning af grafer til Friis i R	Shiv
Udregning af triangulering	Shiv og Troels

9.1 Tidsforbrug ark

	Navn	Troels	Siv	Jesper	Andreas	
Dato	Antal timer	Type af arbejde	Antal timer	Type af arbejde	Antal timer	Type af arbejde
02-05-2022	Torsdag					
03-05-2022	Fredag	6 rapportkravning	5,75 Planlægning	5,75 Planlægning	5,75 Planlægning	
04-05-2022	Lørdag		5,75 Baggrund, design	5,75 Baggrund, BMC analyse, Gemmerlæring	5,75 Arbejde med arbejdsprocess og kravspecifikation	5,75 Panoram
05-05-2022	Søndag	5 UX Analyse	6 Design	4,5 BMC, Økonomisk gevinst, foretakningsplan analyse	0 Passer bæn..	2,5 Rette design, Estimere/beacons
06-05-2022	Mandag	5 UX Analyse og Design	5 Notifikation, gemmerlæring	5 Beregnygtighed, Notifikation, gemmerlæring, UX	5 Specifikation, Beacons	5 Beregnygtighed, software
07-05-2022	Tirsdag	6,5 Design UX, Figma, Applikation	6 Implementering	6,5 Implementering af software, Beacons	6 Beaconsmap	6 Beacons, arbejde med software og vejledning
08-05-2022	Onsdag	5 Implementering af UI	5 Skrive Implementering	5 Skrive implementering, Beacons	6 Etiske aspekter - Rapportstruktur - Rapportlæsning fra stat	5 Retelse og gennemgang af kode
09-05-2022	Torsdag	5,5 Implementering af UI	5 Aftavvning forbedrelse, Diagrammer	5 BMC testing, Absolvering forbedrelse, Diagrammer	5 Rapportlæsning : Design og ned	5 Diagramme og etiske overvejelser rettelser
10-05-2022	Fredag	7 Aftavvning	7 Beacon navigation analyse, beacon sprørring	7 Beacon navigation analyse, beacon sprørring	5 Aftavvning	5 Aftavvning
11-05-2022	Lørdag					
12-05-2022	Søndag	2 Mulig kompas implementation				
13-05-2022	Mandag	3,5 Usability	7,5 Fris ligeing, lokalisering, R	7,5 Fris ligeing, lokalisering, R	6 Generellessig	4 Usability
14-05-2022	Tirsdag	6 Trængelinge, Effektprøvning	7,5 Trængelinge, Fris ligeing, Poster	7,25 Trængelinge, Fris ligeing, Poster	3,5 Generellessig	4 Effektprøvning
15-05-2022	Onsdag	6,5 Effekt, Poster, nette Usability	10 Poster, Refleks, Fjernede 2 app til 1 app	5 Etatnet gevinst, Poster researchvideoer, Instagant	6,5 Poster	0 Syg
16-05-2022	Onsdag	4,5 Poster, Effektprøvning	9,25 Poster, Effektprøvning	5 Poster, Effektprøvning	0 Passer syg neve	5 Effektudregning og databehandling
17-05-2022	Fredag	5,5 Poster	5,5 Poster	5,5 Poster	6 Komiktor	4 Effekt grader og plots
18-05-2022	Lørdag					
19-05-2022	Søndag					
20-05-2022	Mandag	6 korrektur	6 korrektur	6 korrektur	6 korrektur	6 korrektur
21-05-2022	Tirsdag	5 Poster	5 poster	5 poster	3,5 poster	3,5 poster
22-05-2022	Onsdag	4 Poster Øve	4 Poster Øve	4 Poster Øve	4 Poster Øve	4 Poster Øve
23-05-2022	Onsdag					
Samlet tidsbrug	81,5	102,75	92,75	75,5	73	

Figure 26: Tidsforbrug

9.2 Instagant Ark

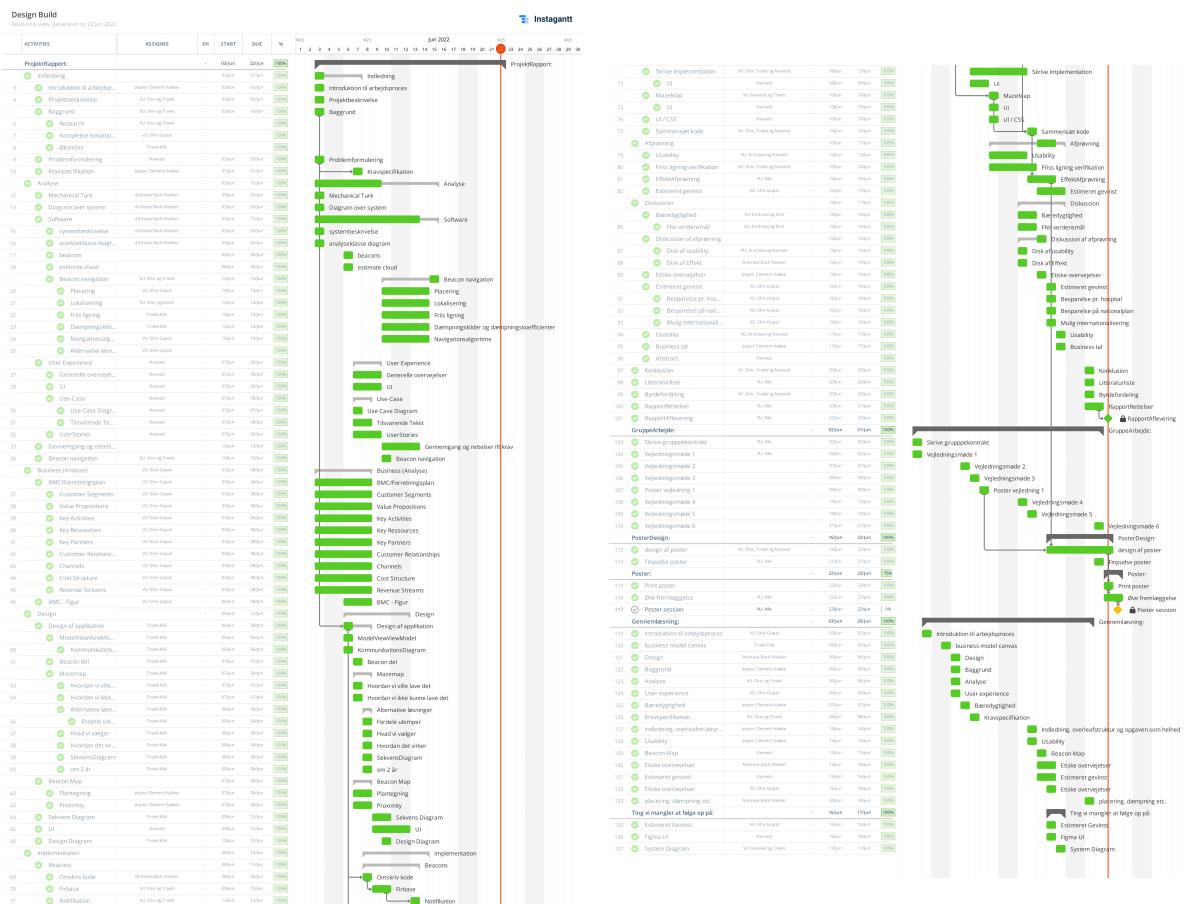


Figure 27: Instangtt

Bilag

Bilag 1

```

[> with(LinearAlgebra):
[> A1 := 63:
[> A2 := 71:
[> A3 := 72:
[> db1 := 79:
[> db2 := 83:
[> db3 := 78:
[> d1 := 10· $\left(\frac{db1 - A1}{12.91}\right)$ 
[> d1 := 12.39349341
[> d2 := 10· $\left(\frac{db2 - A2}{13.15}\right)$ 
[> d2 := 9.12547529
[> d3 := 10· $\left(\frac{db3 - A3}{13.56}\right)$ 
[> d3 := 4.42477876
[> (x - 40)2 - (y - 42)2 = db12:
[> (x - 10)2 - (y - 40)2 = db22:
[> (x - 10)2 - (y - 10)2 = db32:
[>
[> Ligning1 := ((x - 40)2 + (y - 25)2 = d12) - ((x - 10)2 + (y - 10)2 = d32)
[> Ligning1 := (x - 40)2 + (y - 25)2 - (x - 10)2 - (y - 10)2 = 134.0200118
[> simplify( Ligning1 )
[> -60 x - 30 y + 2025 = 134.0200118
[>
[> Ligning2 := ((x - 10)2 + (y - 40)2 = d22) - ((x - 10)2 + (y - 10)2 = d32)
[> Ligning2 := (y - 40)2 - (y - 10)2 = 63.69563220
[> simplify( Ligning2 )
[> -60 y + 1500 = 63.69563220
[>
[> B := 
$$\begin{bmatrix} -60 & -30 & (d1^2 - d3^2 - 2025) \\ 0 & -60 & (d2^2 - d3^2 - 1500) \end{bmatrix}$$

[> B := 
$$\begin{bmatrix} -60 & -30 & -1890.979988 \\ 0 & -60 & -1436.304368 \end{bmatrix}$$

[> LinearAlgebra:-LinearSolve( B )
[> 
$$\begin{bmatrix} 19.5471300666667 \\ 23.9384061333333 \end{bmatrix}$$

[>

```

Figure 28: Bilag 1

Bilag 2

User-Stories

Persona 1:

- Navn: Martin Nielsen.
- Alder: 34 år.
- Køn: Mand.
- Rolle: Pårørende der er på besøg.
- Teknologiforståelse: Lav.
- User story:
 - Jeg skal til Herlev Hospital, for at besøge min onkel som har fået et svært KOL anfall. Sidst jeg var på Herlev Hospital, var det meget svært at finde rundt, så jeg vil ønske der var en APP hvor jeg bare kunne indtaste min onkels stue, hvorefter en god rute fremvises.



-
- <https://focusedcollection.com/179011394/stock-photo-doctor-talking-man-visiting-patient.html>

Persona 2:

- Navn: Maria Carlsen.
- Alder: 19 år.
- Køn: Kvinde.
- Rolle: Patient der gerne vil have snacks, udover hospitals-måltiderne.
- Teknologiforståelse: Høj.
- User story:
 - Sidst jeg var på indlagt på Herlev hospital, havde jeg haft et astmaanfall og spiste udelukkende hospitalsmad i 1 uge. Jeg vidste ikke, hvor man ellers kunne købe snacks og jeg havde ikke energien til at lede efter et sted. Så jeg ville have sat pris på hvis der var en nem og intuitiv måde som jeg kunne finde forskellige madsteder på.

Bilag 3



- <https://www.istockphoto.com/search/2/image?phrase=woman+patient+in+hospital+bed>

Persona 3:

- Navn : Bo Karl Jensen
- Alder : 46 år
- Køn : Mand
- Rolle : Kørestols-Patient som skal til undersøgelse
- Teknologiforståelse: Middel.
- User Story:
 - Jeg har tidligere været til undersøgelse, hvor jeg skulle finde frem til et lokale, som jeg ikke kendte, så jeg henvendte mig til et kort. Kortet tog ikke forbehold for mit behov, og jeg endte med selv at skulle lede efter en rute, som ikke involverede trapper og andre forhindringer. Det ville derfor være rart med et kort, hvor jeg kan angive at jeg er på kørestol, og kunne få en rute som tager forbehold for det.



- <https://da.depositphotos.com/63582921/stock-photo-depressed-disabled-man-in-wheelchair.html>

Bilag 4

Persona 4:

- Navn : Jeanette Lisbeth Juelsberg
- Alder : 32 år
- Køn : Kvinde
- Rolle : NyAnsæt Sygeplejerske.
- Teknologiforståelse: over-middel.
- User Story:
 - Jeg arbejder på Herlev Hospital, og jeg er sygeplejerske på Diabetes-afdelingen. Den afdeling er meget stor, og skal ofte frem og tilbage mellem stuegange. Jeg er stadig ikke fuld orienteret omkring layoutet, og ønsker derfor en meget hurtig og effektiv måde at kunne finde rundt. så hvis jeg får et lokale nummer, kan jeg indtaste det, og finde den hurtigste rute derhen.



* <https://www.colourbox.dk/billede/nurse-billede-27874207>