

# Udvikling af mobil applikation til indendørs navigering

---

**Frederik Bo Albeck - s082730**

**31-01-2012**

Kongens Lyngby 2012

IMM-B.Sc.-2012-5



**Denne rapport er skrevet af:**

Frederik Bo Albeck, s082730

**Vejledere:**

Jakob Eg Larsen  
Sune Lehmann Jørgensen

**Udgivelsesdato:**

31. januar 2012

Institut for Informatik og Matematisk Modellering  
Danmarks Tekniske Universitet  
Bygning 321 DK-2800  
Kgs. Lyngby Danmark

[www.milab.imm.dtu.dk](http://www.milab.imm.dtu.dk)

Tel: +45 45 25 33 51 / Fax: +45 45 88 26 73

E-mail: [milab@imm.dtu.dk](mailto:milab@imm.dtu.dk)



# Abstract.

---

This thesis describes the development of a mobile application, which main purpose is to work as an indoor navigation system. In the analysis part it has been examined which mobile platform and positioning method that are best suitable for this case, in developing a mobile application for indoor positioning.

A description of an useable application that works as a indoor positioning system, despite of some uncertainty, is made. This includes a description of a chosen positioning method, WLAN fingerprinting and the use of an online database.

Furthermore follows a description of the implementation of the necessary functions, the setup of the development environment, the mobile phone and the server along with the testing of the application. Several tests have been performed, which show the possible precision of a positioning and the changes in the strength of the signal from the access points.

The mobile application is developed with the aid of a DTU campus case and with the intend that it will work as an indoor navigation and as a calibration tool.

---



# Resumé.

---

Denne rapport beskriver arbejdet med udvikling af en mobilapplikation, hvis hovedfunktion er, at kunne navigere indendørs i bygninger. Ud fra analysen er der undersøgt hvilke mobilplatforme og positionerings metoder, der er til rådighed for at kunne lave en applikation til indendørs navigering.

Rapporten beskriver udviklingen af en brugbar applikation, der med en hvis usikkerhed kan udføre en positionering indendørs. Dette indebærer en beskrivelse af den valgte positionerings metode fingerprinting og brug af online database.

Derudover er beskrevet implementeringen af nødvendige funktioner, opsætning af udviklingsmiljø, telefon og server, samt test af applikation. Der er blevet udført test der viser den opnåede præcision ved positionering, samt forandringer i signalstyrkerne.

Applikationen er udviklet ud fra en DTU campus case, med henblik på at virke som indendørs navigering og kalibreringsværktøj i bygningerne på DTU.





# Forord.

---

Denne rapport er skrevet ved IMM - Institut for Informatik og Matematisk Modellering, ved Danmarks Tekniske Universitet med aflevering 31. januar 2012.

Dette er det afsluttende projekt i Bachelor uddannelsen. Projektet svarer til 15 ECTS point.

Rapporten beskriver udviklingen af en mobil applikation til indendørsnavigering. Det endelige resultat er en mobil applikation til Android, der ved hjælp af fingerprint metoden, kan fungere som positionerings- og kalibreringsværktøj.

Vejledere for dette projekt Jakob Eg Larsen og Sune Lehmann Jørgensen.

Frederik Bo Albeck



# Indholdsfortegnelse.

---

Abstract.....	iv
Resumé.....	vi
Forord.....	viii
1. Indledning.....	1
1.1. Problemformulering.....	1
1.2. Afgræsning.....	2
2. Analyse.....	2
2.1 Behovet for indendørs positionerings system.....	2
2.2 Indendørs Positionering – eksisterende løsninger.....	3
3. Teori.....	4
3.1 Bluetooth.....	4
3.2 Cellulært netværk.....	4
3.3 Wi-Fi.....	5
3.4 Access points.....	5
3.5 Fingerprinting.....	5
3.6 Triangulation og trilateration.....	6
3.7 Mobilplatforme.....	9
3.7.1 Android.....	9
3.7.2 Blackberry.....	10
3.7.3 IOS.....	10
3.7.4 Windows Phone 7.....	11
4. Design.....	12
4.1 Kravspecifikation.....	12
4.2 Arkitektur.....	12
4.3 Brugergrenseflade.....	15
5. Implementering.....	19

5.1 Udviklingsmiljø. ....	19
5.2 Applikationen. ....	19
5.2.1 Visualisering af placering. ....	20
5.2.2 Data i mellem aktiviteter. ....	21
5.2.3 Database – fingerprint. ....	21
6. Test. ....	26
6.1 Udførelse og resultater. ....	26
6.2 Problemer under testfasen. ....	29
7. Diskussion. ....	29
8. Perspektivering. ....	31
9. Konklusion. ....	33
10. Kilder. ....	34
11. Tabeller og figurer. ....	36
12.1 Bilag A – Filer. ....	37
12. 2 Bilag B - Testresultater med Funf framework. ....	38
12.3 Bilag C - Tidligere versioner af applikationen. ....	39
12.4 Bilag D - Klasse diagram. ....	40



# 1. Indledning.

---

Udviklingen af mobiltelefonen, er gennem de sidste årtier steget kraftigt, så kraftigt, at man nu kalder den for smartphone. Smartphone fordi den indeholder så meget teknologi, at den har overtaget funktionerne fra mange af vores andre elektroniske enheder. Det har den formået i form af funktioner som, E-mail, webbrowser, GPS, kamera, osv. Denne nye teknologi åbner op for en hel række nye anvendelsesområder. Disse nye anvendelsesområder, samt allerede eksisterende løsninger, er beskrevet i analyse afsnittet.

GPS er en positionsteknologi der har eksisteret siden 1980'erne, udviklet af det amerikanske militær. Denne teknologi har i mange år været en del af funktionerne i en mobiltelefon eller smartphone. Dette projekt vil undersøge mulighederne for at udvikle en mobilapplikation til en smartphone, der kan finde dens position inde i en bygning.

GPS teknologien lider under at den typisk ikke virker indendørs og det har derfor været nødvendigt at se på andre teknologier, som alternativ til GPS.

## 1.1. Problemformulering.

GPS er en udbredt teknologi til lokalisering ved hjælp af mobile enheder, men har den ulempe, at den typisk ikke virker indendørs. WiFi teknologien er blevet en udbredt infrastruktur i virksomheder og uddannelsesinstitutioner, hvorfor denne teknologi potentielt kan være en attraktiv mulighed for indendørs lokalisering af personer og objekter i forskellige anvendelser. Problemet med denne teknologi er, at den har nogle ulemper: begrænset rækkevidde indendørs, varierende signalstyrke ved interferens med andre trådløse enheder samt ved varierende belastning af netværket.

Med udgangspunkt i en konkret case (DTU Campus), ønsker jeg at undersøge følgende problemstillinger: Hvor mange access points kræves for at lave lokalisering? Hvilken grad af præcision kan opnås? Hvor meget varierer signalstyrken under forskellige forhold? Til dette formål udvikles en mobilapplikation til smartphones, der kan lokalisere personer eller enheder i en bygning. Jeg vil fokusere på at lave et kalibreringsværktøj, der kan måle og beregne signalstyrken fra access points.

## 1.2. Afgræsning.

Hovedformålet med projektet var at udvikle en mobilapplikation, der kunne fungerer som GPS indendørs. I starten af projektet var ønsket at lave en applikation til flere platforme og lave en sammenligning i forhold til præcision.

Dette blev ændret til én platform i og med at projektets omfang hurtigt kunne blive for omfattende og der derved ikke ville komme et brugbart resultat. Et andet ønske var at udvikle applikationen således, at brugeren kunne navigere rundt via applikationen. Dette blev nedprioriteret set ud fra det omfattende arbejde med at registrere destinationens data, samt den mulige vej dertil. Projektet er altså afgrænset til at udvikle en mobilapplikation, der kan finde dens egen position og vise det grafisk.

## 2. Analyse.

---

Udbredelsen af mobile enheder er vokset meget de sidste mange år og er i den vestlige verden ved at være hvermandseje. Dette medfører en lang række nye områder, som man kan udvikle nye produkter til.

### 2.1 Behovet for indendørs positionerings system.

I det følgende vil jeg beskrive, hvilket behov der kan være for denne teknologi, hvor i særdeleshed udbredelsen af location based services medfører et større og større behov for indendørs positionerings system.

Location based services (LBS) er en teknologi til at udsende en service til enheder alt efter hvor den er placeret. Dette kan være alt fra reklamer til vejrmeldinger. Et af de områder, hvor man virkelig håber på at kunne udnytte denne teknologi, er i ulykkes situationer, men også som en ekstra service der ikke er ulykkes relateret.

Et andet behov er muligheden for optimering af processer. Her kan systemet hjælpe til med at lokalisere personer eller hardware i store bygninger - eksempelvis i store virksomheder eller hospitaler.

Følgende beskriver nogle scenarier, hvor indendørs positionering og location based services kan udnyttes.

Et nyt storcenter er lige åbnet. De vil med en ny og moderne teknologi, kunne yde en ekstra service til deres kunder. En applikation designet til deres center, vil kunne hjælpe kunderne med at finde den rigtige butik. Butikkerne kan via applikationen appellere til deres kunder gennem kommunikation i form af konkurrencer, reklamer og andre tilbud.

Andre scenarier: Offentlige bygninger som uddannelsesinstitutioner, museer og hospitaler. En sådan service vil kunne hjælpe den nye studerende med at finde rundt på campus, til at finde lokale for den næste forelæsning eller finde ud af hvor den nærmeste kantine ligger. På et museum vil denne service kunne hjælpe gæsterne med at finde rundt, hvor de undervejs kan få al den information de ønsker. Det vil kunne hjælpe lægen til hurtigt at finde korrekte placering på en patient.

Ved store messer kan denne service hjælpe gæster rundt samt tilbyde dem information alt efter hvor de er placeret. Løbende kan punkter på programmet for messen, markeres på kortet.

## 2.2 Indendørs Positionering - eksisterende løsninger.

Som beskrevet er Indendørs positionering en teknologi der bliver udviklet mere og mere på og der findes i dag mange løsninger. Her gives et par eksempler på eksisterende løsninger der gør brug af indendørs positionering:

Ekahau er en af flere virksomheder, der specialiserer sig i positionering via Wifi, ved at tilbyder RTLS løsninger. RTLS står for real-time location system, der løbende laver opdatering i forhold til de målinger som systemet foretager. [1]

Google Maps - Ved dette års CES messe, har Google udviklet et Indoor Google Maps. Denne udvidet funktion i Google Maps, hjælper gæsterne til at de let kan navigere rundt på messen.<sup>1</sup> Dette er dog ikke helt nyt for Google. De har også mappet en del større amerikanske lufthavne.<sup>2</sup>

Skejby sygehus - det første sygehus i Danmark der har indført et indendørs positionerings system. Med dette system, giver det mulighed for at spore mennesker og hardware på hospitalets 147000 kvadratmeter. De har ved hjælp af fingerprinting mappet hele sygehuset trådløse infrastruktur. [2]

---

<sup>1</sup> <http://androidcommunity.com/navigate-through-ces-with-google-maps-20120108/>

<sup>2</sup> <http://latimesblogs.latimes.com/technology/2011/11/google-maps-60-indoor-mapping-airports.html>



## 3. Teori.

---

Ved indendørs positionering, hvor det ikke er muligt at bruge GPS til at finde positionen, er det nødvendigt at gøre brug af jordbaserede positionsteknologier.

I det følgende vil gennemgås forskellige teknologier der er til rådighed ved jordbaserede positionsteknologi. Satellitbaserede positionsteknologi udelades, da denne teknologi ikke er relevant ved indendørs positionering. Disse teknologier, omhandler kendte forbindelsesteknologier, som alle smartphones besidder, nemlig Bluetooth, cellulær netværk og wlan.

### 3.1 Bluetooth.

Bluetooth teknologien består af et trådløst datanet (WPAN – Wireless Personal Area Network), der tillader en hurtig dataoverførsel over korte afstande. [3] Bluetooth er en meget udbredt teknologi, hvilket understreges med et lavt strømforbrug og en lav udviklingspris, hvorfor teknologien findes i stort set alle smartphones. Bluetooth begrænses af den korte virke afstand på 10 m. alt efter forhold. Dette medfører dog ikke at denne teknologi ikke kan bruges ved indendørs positionering.

Indendørs positionering kræver dog, at der findes tilgængelige access points, hvorfra det er muligt at måle signalstyrken og derved beregne positionen. Bluetooth bruges normalt som et ad-hoc netværk til at oprette forbindelse mellem enheder og det er derfor heller ikke normalt, at man bruger denne teknologi til at forbinde enheder til internettet via bluetooth access points.

### 3.2 Cellulært netværk.

Cellulær netværket (Mobiltelefonnetværk) er et trådløst netværk til mobilkommunikation. Dette netværk tillader mobile enheder at kommunikere med hinanden. Enhederne har ikke direkte forbindelse til hinanden, men er forbundet til netværket via en af netværkets basestationer. Ved bestemmelse af position i sådan et netværk, gøres normalt brug af triangulering. Dette kræver at enheden er i nærheden af tre eller flere mobilmaster.

Præcisionen ved brug af dette netværk er ikke god nok til Indendørs positionering. I byområder kan man via triangulation, placere en enhed med en præcision ned til 50 meter. Ved indendørs positionering er denne løsning ikke en mulighed, i og med at kontorlokalerne er en del mindre end 50 meter, samt at netværkets signaler kan lave interferens med vægge og døre i bygningen og det kræver derfor højere præcision.

### 3.3 Wi-Fi.

Wi-Fi er en term for trådløse datanet, der bruges blandt andet ved WLAN (Wireless Local Area Network) netværk<sup>3</sup>. Wi-Fi omfatter IEEE standarderne 802.11, der angiver specifikationerne for netværksteknologien mellem ethernet og trådløs. Disse standarder tillader mobile enheder med en trådløs netværksenhed, som computere, smartphones, tablets, konsoller, osv., at oprette en forbindelse til internettet. Netværkets rækkevidde er alt efter brug af wifi-standarder, netværkets placering og omgivelser, fra 35 meter til 75 meter.

### 3.4 Access points.

Accesspoint eller basestationer, er bindeleddet mellem de trådløse enheder og LAN netværket. Der findes mange forskellige typer access points, med forskellige specifikationer, både til Bluetooth og Wi-Fi. Trådløse accesspoint er en let måde er tilbyde en forbindelse til mange enheder, uden at skulle trække en masse kabler. Access points sender og modtager radiosignaler med en eller flere bestemte frekvenser, alt efter hvilken 802.11 standard accesspoint og mobilenheden understøtter. De signaler som access points udsender kan måles i forhold til signalstyrken, der angives i dBm.

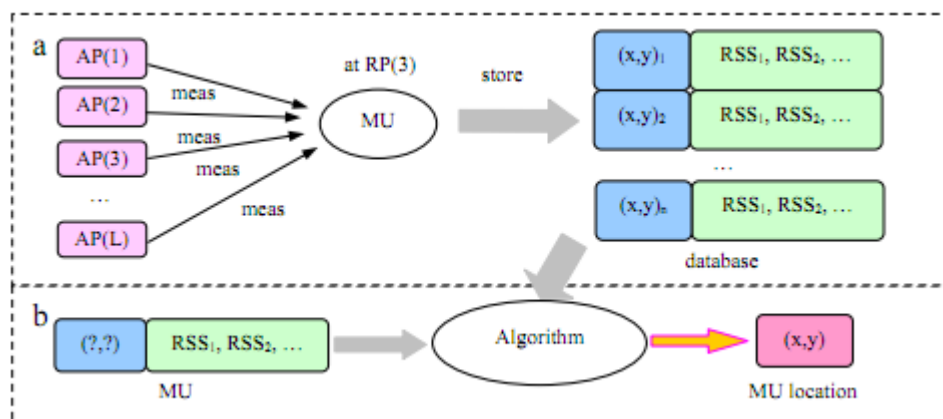
### 3.5 Fingerprinting.

Fingerprint metoden består i at der laves en måling af signalstyrkerne på en placering fra de omkringliggende access points. Dette fingerprint gemmes på en intern eller ekstern database. Det kan derefter bruges til at finde en enheds placering ved at sammenligne, de signalstyrker som enheden selv måler med fingerprintet gemt på en database.

---

<sup>3</sup> <http://searchmobilecomputing.techtarget.com/definition/Wi-Fi>

Fingerprint metoden består af to faser *a* og *b*, se nedenstående billede. [4] Fingerprint fasen *a* og positionerings fasen *b*. Fingerprint fasen, hvor selve fingerprintet registreres og uploades til en database. Positioneringsfasen, hvor enheden måler signalstyrkerne og laver en sammenligning med fingerprint databasen. De to faser er illustreret på nedenstående figur. [5]



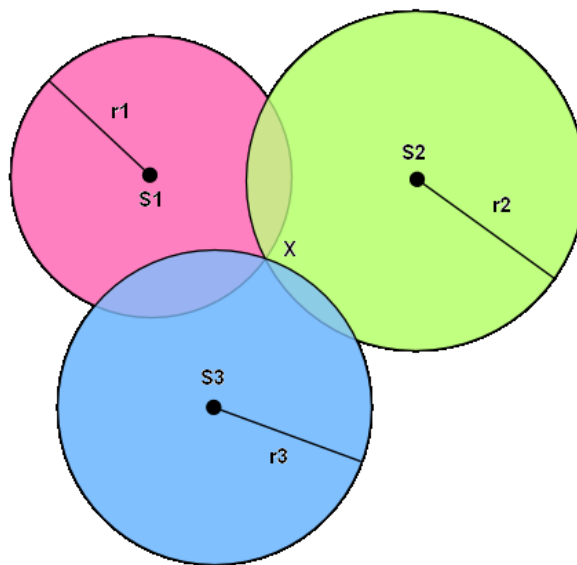
Figur 1 - Fingerprint faser

Fingerprintet der gemmes, er beregnet ud fra et gennemsnit af flere målinger, dette gør, at man undgår uregelmæssigheder i målingsresultaterne. Præcisionen af denne metode afhænger af mængden af målinger udført i fase a. En stor fordel ved denne metode er, at det er forholdsvis let at implementere, da det ikke er nødvendigt med komplekse beregningsmodeller af forringelser i signalstyrkerne.

Ulemperne ved denne metode opstår hvis der forekommer ændringer i netværks infrastrukturen. Dette vil medfører gentagne opdateringer af fingerprint databasen, hvilket er besværligt og tidskrævende.

### 3.6 Triangulation og trilateration.

Triangulation og trilateration er to metoder der kan bruges til positionering af eksempelvis mobile enheder som smartphones i forhold til kendte positioner. Triangulation fungerer ved at enheden skal være inde for rækkevidde af minimum to eller flere basestationer, for derved at kunne beregne vinklerne mellem enheden og de forskellige basestationer. Ved at beregne hvor vinklerne krydser hinanden, kan man finde positionen af enheden i forhold til mobilmasterne. Dette kræver at man har en meget præcis position af mobilmasterne.



Figur 2 - Trilateration

Trilateration fungerer ved at enheden kan måle signalstyrkerne fra minimum tre eller flere basestationer, og ved at kende afstanden mellem basestationerne beregne afstanden til enheden. Afstanden mellem basestationen og enheden angives som en sfære omkring basestationen, med afstanden som sfærens radius. Når dette gøres ved tre eller flere basestationer, kan positionen udregnes ud fra hvor, sfærerne overlapper hinanden. Dette illustreres på figur 2. hvor S er basestationer,  $r$  er afstand fra basestation til enheden og X er enhedens placering.

Udfordringen ved denne metode er bestemmelsen af afstanden ud fra signalstyrken. Det er derfor nødvendigt at gøre brug af nogle beregningsmodeller, der tager højde for forstyrrelser af signalstyrkerne. Disse forstyrrelser består blandt andet af signalernes forringelser over afstand, tab af styrke ved gennemtrængning af vægge, døre og gulve, og "multipath" refleksioner fra signalet, samt forstyrrelser fra andre enheders signaler. [6]

Free space modellen angiver de signal forringelser, der opstår når signalet har en line-of-sight rute, en direkte rute mellem sender og modtager - ingen forhindringer.

Free space modellen er dog ikke altid en mulighed. Da signal sjældent har et direkte line-of-sight, er det nødvendigt med nogle modeller der tager højde for bygningens forhindringer.

Her er der modeller som Multi-Wall-and-Floor [7] eller Multi-Wall [8] model, der gør brug af Free space modellen, men tager højde for antal vægge og gulve, signalet skal passere. Multi-Wall-and-Floor modellen kan beskrives med følgende formel.

$$L_{MWF} = L_0 + 10 n \log(d) + \sum_{i=1}^I \sum_{k=1}^{K_{wi}} L_{wik} + \sum_{j=1}^J \sum_{k=1}^{K_{fj}} L_{fjk}$$

$L_0$  = signal forringelse over en distance på én meter

$n$  = styrke forfald indeks

$d$  = afstand mellem sender og modtager

$L_{wik}$  = signal forringelse grundet væg af typen  $i$ .

$L_{fjk}$  = signal forringelse grundet gulv af typen  $j$ .

$I$  = antal af væg typer

$J$  = antal af gulv typer

$K_{wi}$  = antal vægge af type  $i$  signalet skal igennem.

$K_{fj}$  = antal gulve af type  $j$  signalet skal igennem.

Fordelene ved trilateration metoden er, at det modsat fingerprint metoden, ikke er nødvendigt at opbygge en fingerprint database, der løbende kræver opdateringer. Denne metode er bedre egnet til ændringer i netværks infrastrukturen.

Ulemperne ved metoden er, at det kræver præcis placering over access points samt brugen af komplekse beregningsmodeller der tager højde for access points omgivelser.

Set ud fra denne analyse, og det forhold at bygningerne på DTU er udstyret med WLAN access points, er det eksisterende WLAN netværk derfor vurderet som den oplagte løsning til brug ved positionering i applikationen.

Positioneringsmetoden fingerprint blev foretrukket i forhold til triangulation og trilateration. Triangulation og trilateration blev udeladt på grund af problemer med udregning af vægtning af wifisignaler i forhold til access points på flere etager og derved uregelmæssigheder i målingerne på grund af signalernes forringelser igennem vægge og gulve. Som beskrevet krævede dette nogle ret komplekse beregningsmodeller.

### 3.7 Mobilplatforme.

De mest kendte og udbredte platforme til smartphone telefoner er: Android, Blackberry, IOS, og Windows Phone 7. Hvor Blackberry, Symbian og Windows Mobile (i dag Windows Phone 7) var de største i starten af smartphone bølgen, er de nu blevet overhalet af de nyere styresystemer Android og IOS, der i dag er de to største "spillere" på området.[9]

I dette projekt havde det være ønskværdigt at udvikle til flere platforme og lave en sammenligning, men for at begrænse projektet til det mere overkommelige, har jeg valgt kun at udvikle til en platform, hvilket jeg vil komme ind på.

I det følgende vil jeg beskrive og sammenligne de fire styresystemer Android, Blackberry, IOS og Windows Phone 7, og ud fra dette redegøre for mit valg til dette projekt, samt hvilke alternativer der har været mulige. Da applikationen der skal udvikles til platformen, skal fungere ved at gøre brug af signalstyrker fra de omkringliggende trådløse netværk, stiller det et krav til styresystemet, at det skal være muligt at tilgå enhedens trådløse netværks API.

#### 3.7.1 Android.

Android er en del af "the Open Handset Alliance" ført an af Google. Google udkom med første version af Android i 2008. Android er i dag det mest udbredte styresystem til håndholdte enheder, set i forhold til deres markedsandele. [10]

Android er et Open Source system, der gør blandt andet medfører, at der er åbent for deres API'er, hvilket betyder at det nemt at udvikle applikationer der kan interagerer med telefonens software.[11] En androids telefons netværksenhed er let tilgængelig via Android bibliotekets WifiManager. [12] Denne klasse udgør API'en til at håndtere og overvåge trådløs forbindelse.

Udvikling til android foregår i udviklings sproget er Java. Til udvikling findes pakkerne Android SDK (Software Development Kit) og AVD (Android Virtual Device). SDK der indeholder de fornødne værktøjer og biblioteker, og AVD der kan oprette en virtuel mobil til test. Derudover findes der et ADT plugin til Eclipse IDE (Android development tool), der gør Eclipse til det foretrukne

udviklingsmiljø. Udgivelse af applikationer til android kræver at man registrerer en udvikler konto og et engangsgebyr på 25\$, der fungerer som deres sikkerhed for bedre produkt kvalitet.

### 3.7.2 Blackberry.

Blackberry OS er RIM (Research in Motion) styresystem til deres smarthphones. Blackberry har været en af de større "spillere" på markedet, men gennem de seneste år, har det været nedadgående med deres markedsandele. Applikationer til Blackberry kan udvikles i blandt andet Java og web-udvikling. Support af udviklingsmiljøer udgør blandt andet Eclipse og Microsoft Visual Studio via Blackberrys WebWorks plugin. Ved udvikling til Blackberry, er der adgang til mange af deres API'er, hvilket giver mulighed for at tilgå information fra telefonens netværksenhed via WLANinfo. [13]

For at udgive applikationer til Blackberry App World, [14] kræver det "kun" en registrering, modsat Android Market er der ikke noget gebyr.

### 3.7.3 IOS.

IOS er Apples styresystem til deres iPhones. IOS udkom første gang i 2007 og blev hurtigt utroligt populært. IOS har længe været det mest udbredte styresystem, men må i dag se sig slået i af android styresystemet til mobile enheder. Udvikling til IOS foregår i deres eget udviklingsmiljø xCode, som man får adgang til efter en registrering.

Det kræver et gebyr at få lov til at udvikle og udgive applikationer til IOS, de har dog et gratis University Program[15], der gør det muligt for blandt andet studerende at arbejde med.

Apple tillader ikke at man kan tilgå IOS wifi API, dette er dog muligt ved at gøre brug af 3. parts kode.[16] Udgivelse af applikationer til kan foregå på forskellige måder, men for at applikationen skal udgives på Apples App Store, skal den gennemgå Apples godkendelse. Dette sikre deres brugere at applikationerne de kan anskaffe på App Store, virker og ikke er skadelige. Samtidig betyder det at applikationer der strider i mod deres godkendelses regler ikke kan udgives. Dette indebærer blandt andet, at applikationer der gør brug af 3. parts API'er ikke kan godkendes.

### 3.7.4 Windows Phone 7.

Windows Phone 7 er Microsofts styresystem til smartphones, der er en efterfølger til deres Windows Mobile. Da Windows Phone 7 er meget nyt styresystem i forhold til de andre, er ikke i nærheden af at være lige så udbredt som Android og IOS. Applikationer til WP7 udvikles i C# og gør brug af frameworket Silverlight og XNA-toolset. Udvikling til WP7 foregår i Visual Studio.

På nuværende tidspunkt er der ikke nogen API der gør det muligt at tilgå netværksenheden i WP7 styresystemet. Da dette er essentielt for udvikling af applikationen, medfører det at WP7 platformen ikke kan bruges til projektet.

For at udgive en applikation, på Microsofts Marketplace, kræver det en registrering igennem Microsofts App Hub, samt et årligt gebyr på 99\$.<sup>4</sup> Derudover skal applikationen igennem en udgivelses proces der er lidt lige som Apple, med godkendelse af applikationen, for at sikre Marketplace mod forbudte apps. [17]

Valget af mobilplatform til projektet var ikke låst til en platform. Som beskrevet var der muligheder for at gøre brug af andre platforme, dog endte valget med Android styresystemet, hovedsageligt på baggrund af de mange muligheder der er ved udvikling til android, set ud fra deres brug af Open Source samt deres integration med Eclipse udviklingsmiljøet.

---

<sup>4</sup> <http://create.msdn.com/en-US/home/membership>



## 4. Design.

---

### 4.1 Kravspecifikation.

For at applikationen kan betegnes som funktionel, er det nødvendigt at den opfylder nogle krav:

1. Angive enhedens position med en rimelig præcision. Ned på bygning, etage og rum-niveau.
2. Applikationen skal være enkel og let at bruge.
3. Ved fingerprint, skal nødvendig information uploades til en database.
4. Fingerprint koordinater kan angives med tryk på skærmen.
5. Ved bestemmelse af position skal den rette information downloades fra database, ud fra de angivne søgekriterier.
6. Applikation skal angive en visualisering af positionen på kort.
7. Det skal være muligt at flytte og zoom ind og ud på kortet.
8. Applikationen skal med få ændringer kunne tilpasse til flere bygninger.

### 4.2 Arkitektur.

Applikationen er opdelt i fire aktiviteter:

Ip\_touch

Denne klasse indeholder hovedaktiviteten der angiver en startmenu, med de valgmuligheder brugeren får stillet når applikationen startes. Position, der starter en ny aktivitet som viser et kort over bygningen og positionen for enheden. Fingerprint, der starter en ny aktivitet der giver brugeren mulighed for at vælge en bygning og en etage for hvor man ønsker at lave et fingerprint.

Wifiscanner

Denne klasse er til for at udføre en måling af de omkringliggende access points signalstyrke. Denne del er implementeret i programmet ud fra en tutorial, der laver en generel skanning og lister dem. Denne kode virker ved at gøre brug af Androids wifimanager class, der giver mulighed for at få information om tilgængelige trådløse access points. Dette indebærer den nødvendige information i form af access points mac adresse og signalstyrke.

## Location

Denne klasse giver brugeren mulighed for at vælge hvilken bygning og på hvilken etage man ønsker at lave et fingerprint. Da denne applikation er i en tidlig fase og udviklet til brug på DTU, er der ikke mange valgmuligheder i dette vindue. Ved valg open map startes en ny aktivitet med klassen map, hvor de valgte muligheder sendes med.

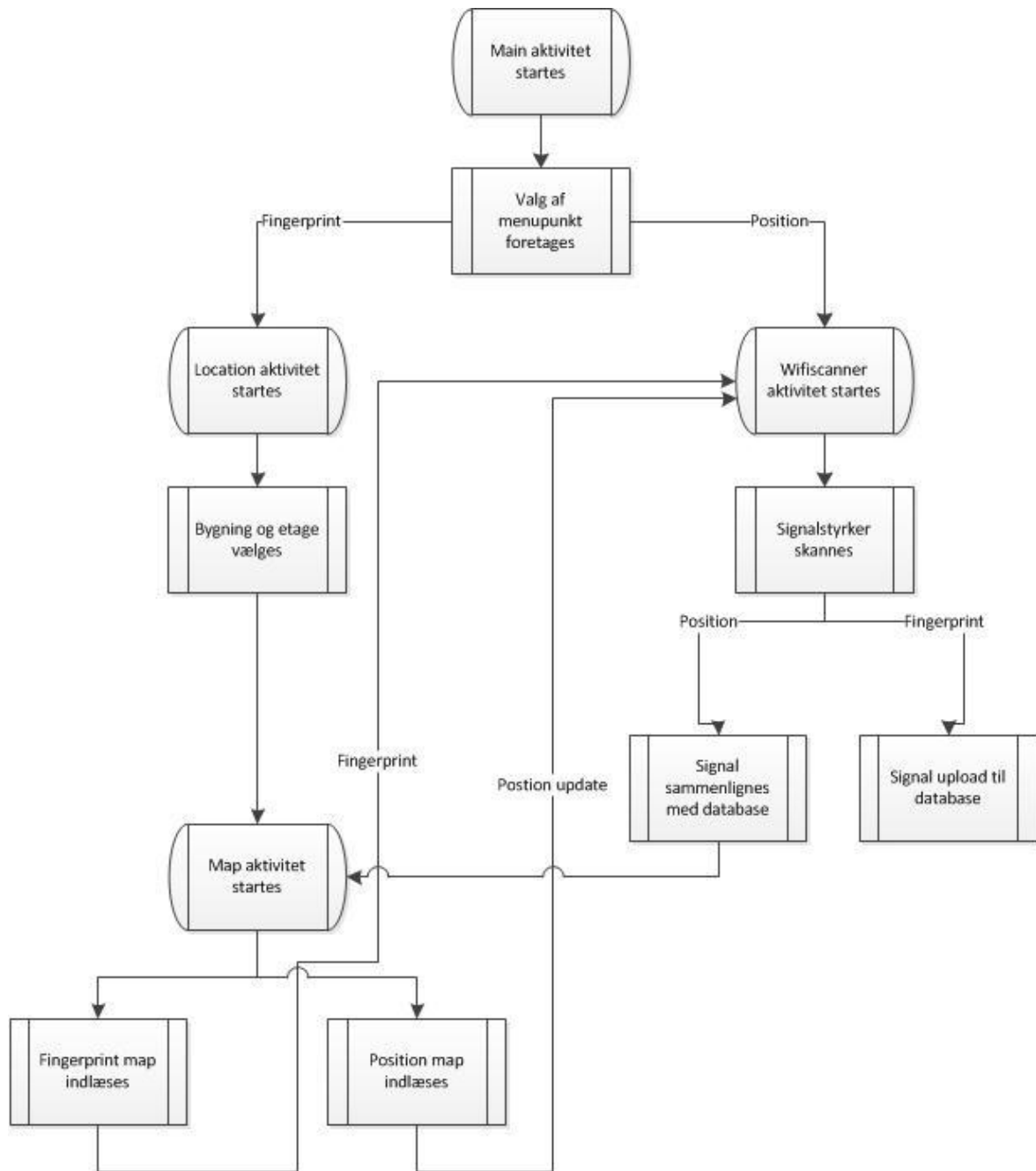
## Map

Denne klasse består af en aktivitet, der angiver et kort, i form af en plantegning, over en etage i en given bygning. Kortet bestemmes ud fra hvilke informationer der er blevet sendt til denne klasse.

I denne aktivitet, er der implementeret androids Gesture funktioner, der tillader blandt andet brugen multi-touch. Dette har tilføjet applikationen translate og zoom funktioner, der er med til at gøre oplevelsen af applikation bedre.

Ved fingerprint, er det muligt at sætte positionen man ønsker at lave fingerprintet for, ved at trykke en enkelt gang på skærmen. Via en menu vælges fingerprint, og derved uploades informationen til serveren. Ved position, er dette ikke muligt, her er der vist et punkt, der angiver enhedens position. Menupunktet er udskiftet med punktet Refresh, der ved aktivering, indlæser en ny måling af signalstyrkerne og anmoder om ny placering fra serveren.

Følgende flow diagram, viser hvordan aktiviteterne fungerer sammen og hvordan flowet ser ud. For yderligere information, se klasse diagram i bilag d.



Figur 3 - Flow diagram

### 4.3 Brugergrænseflade.

I det følgende vil jeg præsentere brugergrænsefladen for den udviklede applikation samt dens funktioner. Brugergrænsefladen for tidligere versioner af applikationen kan ses i bilag B.

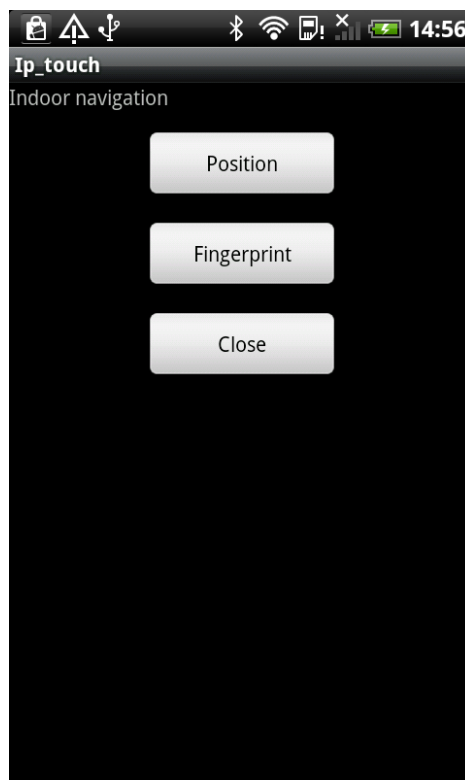
Applikationens brugergrænseflade er udviklet til at være simpel og enkel, og derved gøre applikationen så let overskuelig som muligt.

Opstartsvinduet giver brugeren mulighed for at vælge mellem nogle menupunkter.

Position, der åbner et kort og angiver enhedens placering.

Fingerprint, der lader brugeren opdatere databasen med fingerprint.

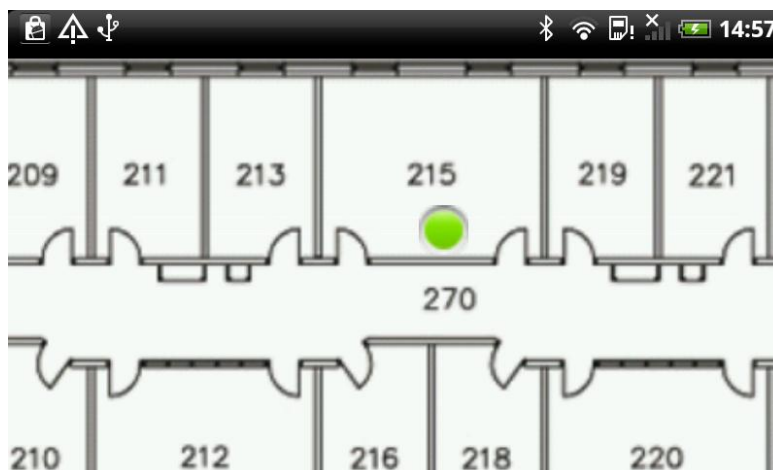
Close, der lukker applikationen.



Figur 4 - Startvindue med menu

Ved tryk på position, vises et kort over den beregnede position i map aktiviteten. Enhedens position vises med en grøn "prik".

I dette vindue (figur 5) har brugeren mulighed for at flytte og zoome ind og ud på kortet, der viser etagen og bygningen hvor enheden er placeret.



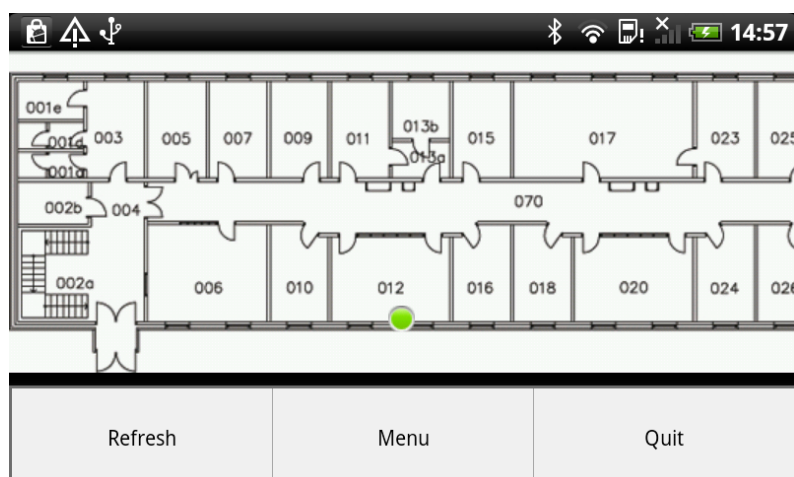
Figur 5 - Position map

Brugeren har ved tryk på telefonens menuknap, mulighed for at få en options menu frem (se figur 6). Denne menu indeholder punkterne:

Refresh: opdatere placeringskoordinaterne fra databasen.

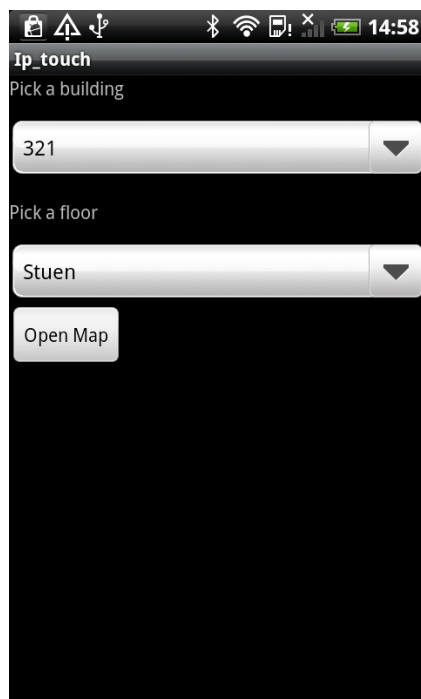
Menu: gå tilbage til startaktiviteten, for eventuelt at vælge fingerprint.

Quit: lukke applikationen ned og vende tilbage til telefonens Home-vindue.



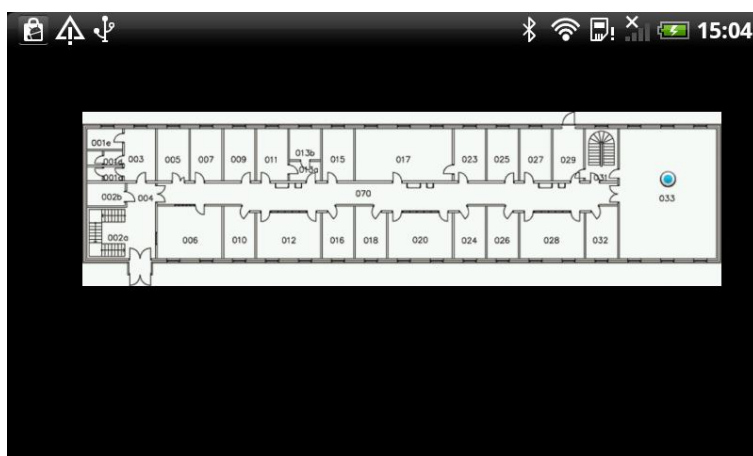
Figur 6 - Position map menu

Ved tryk på fingerprint i startvinduet, åbnes en ny aktivitet, hvor valg af bygning og etage foretages (se figur 7).



Figur 7 - Fingerprint position

Dette vindue viser to dropdown menuer, hvor det er muligt at vælge den etage hvorpå der skal foretages fingerprint. Ved tryk på "Open Map", startes en ny aktivitet.

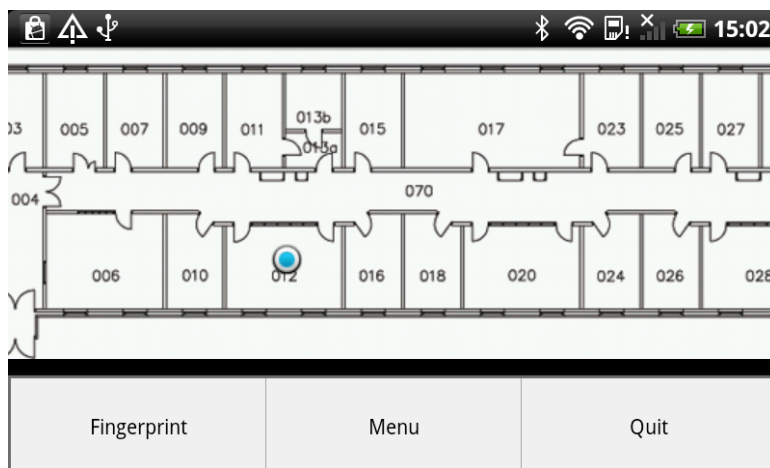


Figur 8 - Fingerprint

Figuren ovenover, viser et kort over stue etagen i bygning 321, hvor billedet er flyttet ind på midten og zoomet ud.

Ved tryk på menu knappen, fremkommer Options menu, hvor brugeren får vist tre menupunkter, lige som ved bestemmelsen af position. Menupunktet Refresh er i dette vindue skiftet ud med menupunktet Fingerprint.

Fingerprint, der gemmer placeringen af den blå "prik" og starter en ny aktivitet, der uploader den angivende placering til databasen.



Figur 9 - Fingerprint menu

Udviklingen af applikationen har budt på mange ændringer undervejs i projektforsløbet, dette er også gældende for brugergrænsefladen og måden hvorpå fingerprint position blev gemt på databasen. I bilag A kan ses tidligere versioner af applikationen.

# 5. Implementering.

---

## 5.1 Udviklingsmiljø.

Valget af udviklingsmiljø faldt på Eclipse IDE, hvilket var nemt set ud fra analysen af de mobile platforme. Android gør brug af udviklings sproget Java, samt det at Eclipse bliver understøttet af Google, i form af Android Development Tools (ADT) der gør det muligt for Eclipse at arbejde sammen med Androids Application Programming Interface (API) og Android Virtual Device (AVD). [18]

Test af applikationen undervejs i projektet, har hovedsageligt foregået på telefonen. Dette har været nødvendigt i og med at AVD android emulator ikke kan tilgå Wifi-enheden.

## 5.2 Applikationen.

I det følgende vil jeg beskrive implementeringen af essentielle dele af applikationen og hvordan de fungerer.

Opsætning af wifimanager

For at kunne skanne signalstyrkerne fra de trådløse netværk, har det været nødvendigt at få adgang til WifiManager API. Ved hjælp af en tutorial, har det været muligt at implementere en wifi-scanner. Følgende kode initierer Android wifimanager.

```
mainWifi = (WifiManager) getSystemService(Context.WIFI_SERVICE);
receiverWifi = new WifiReceiver();
registerReceiver(receiverWifi, new
IntentFilter(WifiManager.SCAN_RESULTS_AVAILABLE_ACTION));
```

Dette giver os mulighed for at få oplysninger om netværk i form af blandt andet, BSSID og level, der angiver netværkets accesspoint mac adresse og signalstyrke.



### 5.2.1 Visualisering af placering.

Map klassens hovedfunktion i denne applikation er at lave en visualisering af placeringen. Dette gøres i form af nogle plantegninger over etagerne i en bygning. En "smart" feature ved denne del af applikationen, er mulighed for billedmanipulation. Dette er muligt ved brug af touch-funktioner. Til denne del er der fundet hjælpende kode på nettet og herefter implementeret i applikationen.

Berøring af skærmen kontrolleres af metoden:

```
public boolean onTouchEvent(MotionEvent ev)
```

Forskelle på berøring, kontrolleres ved:

```
switch (action & MotionEvent.ACTION_MASK) {  
    case MotionEvent.ACTION_DOWN: {  
  
    case MotionEvent.ACTION_MOVE:
```

Hvilket giver mulighed, for at lade applikationen ændre kortets tilstand på forskellige måder.

Kortet kan manipuleres yderligere via scalelistenere, der gør at man kan zoome ind og ud på kortet.

```
private class ScaleListener extends ScaleGestureDetector.SimpleOnScaleGestureListener
```

Hvis forrige klasse har indlæst data i koden til at starte en ny aktivitet, indlæses dette, hvilket indvirker på hvordan map aktiviteten opfører sig. Dette indebærer hvilke plantegninger der skal indlæses og hvis map aktiviteten er indlæst i forbindelsen med positionering, skal aktiviteten, visualiserer denne placering.

For angivelse af fingerprint positionen, er implementeret funktioner som visualisere enkelt tryk på skærmen, samt gemmer koordinaterne for trykket. For at dette skulle fungere samtidig med at have muligheden for at flytte og skalerer plantegningen, har det være nødvendigt at implementere en kode der tager højde for ændringerne.

```
int CposX = (int)((((posX - (posImage.getIntrinsicWidth()/2)) -  
    mPosX)/mScaleFactor);  
  
int CposY = (int)((((posY - (posImage.getIntrinsicHeight()/2)) -  
    mPosY)/mScaleFactor);  
  
posImage.setBounds(CposX, CposY, CposX+posImage.getIntrinsicWidth(),  
    CposY+posImage.getIntrinsicHeight());
```

Hvor `CposX` og `CposY` er de nye beregnede koordinater for billedet der angiver positionen for hvor der er trykket på skærmen. Her tages der højde for om

Viewet er blevet rykket ved `posX` og `posY` eller hvis det er skaleret ved `mScaleFactor`.

### 5.2.2 Data i mellem aktiviteter.

Denne applikation består af flere aktiviteter, der kommunikerer med hinanden på forskellige måder. Som vist i arkitekturen af applikationen, har det været nødvendigt at kunne håndtere, hvordan der skulle sendes mellem aktiviteter.

Fra wifiscanner aktiviteten til map aktiviteten sendes parametrene "building", "floor", "x\_coordinate" og "y\_coordinate", med følgende kode:

```
Bundle bundle = new Bundle();
bundle.putString("building", xbuilding);
bundle.putInt("floor", xfloor);
bundle.putBoolean("bool", true);
bundle.putInt("x", x0);
bundle.putInt("y", y0);
Intent i = new Intent(wifiscanner.this, map.class);
i.putExtras(bundle);
startActivity(i);
```

Da location aktiviteten også sender data til map i form af parametrene "bulding" og "floor", er det nødvendigt at tage højde for, hvordan denne data skal modtages. Dette gøres ved at sende en Boolean med der angiver om der skal indlæses koordinater.

### 5.2.3 Database - fingerprint.

Denne klasse styrer forbindelsen mellem enheden og serveren, som beskrives senere i kapitlet. Klassen består af to metoder, der henholdsvis sender og modtager data fra serveren. Der oprettes en ny forbindelse til serveren hver gang der skal sendes eller modtages data. Forbindelsen oprettes ud fra *org.apache.http* biblioteket, hvor fra det er muligt at lave en http forbindelse:

Alt data om fingerprintet, som bygning, etage, koordinater, mac-adresser og signalstyrker indlæses i en ArrayList og sendes via:

```
HttpClient httpClient = new DefaultHttpClient();
HttpContext localContext = new BasicHttpContext();
HttpPost httpPost = new HttpPost(db_url);
httpPost.setEntity(new UrlEncodedFormEntity(request));
```

Hvor URL adressen angiver hvilket php-script på serveren, der kaldes. Dette php-script behandler kald fra android enheden. Ved setdata oprettes forbindelse til databasen:

```
mysql_connect("localhost","root","pass");  
mysql_select_db("android");
```

Data indlæst i ArrayList, læse ud fra de angivne navne i database klassen.

```
$building = $_POST['building'];  
$floor = $_POST['floor'];
```

Data indlæses i database tabellen ved følgende INSERT MySQL queiry:

```
mysql_query("INSERT INTO "tabelnavn" (kolonne1, kolonne2, ...)  
VALUES(værdi1, værdi2...)");
```

Når der skal hentes position fra databasen, oprettes forbindelse på samme måde som ved set data. Kaldet til databasen sendes til et php-script der behandler den ønskede anmodning. php-scriptets funktion er at finde et passende fingerprint, enhedens måling sendes via ArrayList, på samme måde som ved set data. Dette fungerer ved at scriptet, laver en søgning i tabellen efter den bedste match, ud fra access points mac-adresse og signalstyrke:

```
mysql_query("SELECT building, floor, x_coordinate, y_coordinate FROM table  
WHERE ap_mac1 = ap_mac1 AND (ap_strength1 <= ap_strength1+5 AND  
ap_strenght1 >= ap_strength1-5)
```

Hvor understreget er mac-adresse og signalstyrke målingerne fra telefonen. Hvis dette ikke giver et match, matches næste signalstyrke med samme mac-adresse. Hvis ikke søgningen giver et resultat, udvides søgning så den ikke kræver at alle accesspoint passer. Se php-script i bilag. Dette medfører en usikkerhed i præcisionen af positionen.

Output fra php-scriptet er formateret som Json objekt. Den relevante data fra Json objektet indlæses i et returnValue objekt, som derefter returneres:

```
JSONObject json = new JSONObject(result);  
returnValue[0] = json.get("building");  
returnValue[1] = (String) json.get("floor");  
returnValue[2] = (String) json.get("x_coordinate");  
returnValue[3] = (String) json.get("y_coordinate");
```

Applikationen er opbygget, så den let kan udbygges til omfatte flere bygninger. Dette vil kun kræve en ændring af placeringsinformationerne og valg af plantegning over bygningerne.

Kort

Plantegninger til visualisering på telefonen, er venligst udgivet af DTU. Disse plantegninger var ikke fuldt opdaterede, idet et par store kontorer var delt op til mindre kontorer.

Telefon:

Telefonen der blev benyttet til dette projekt er en HTC Desire med Android version 2.2., der er udlånt fra DTU.

Server opsætning:

Til projektet har det være nødvendigt med en server til at gemme data, blandt andet fra fingerprinting af access points samt placering af access points. Serveren er opsat med henblik på at fungere som database, hvor telefonen via en http webservice opretter en forbindelse.

Denne server består af en Lenovo Thinkpad bærbar med Windows 7 SP1 professionel som operativ system. Hvorpå er følgende software installeret:

- Apache 2.2.21
- PHP 5.2.12
- MySql 5.5
- PhpMyAdmin

Apache er en HTTP webserver fra Apache Software Foundation, der er Open Source software. [19] webserveren er opsat således, at den angiver stien til mappen "webroot", hvorpå telefonens http forbindelse har adgang til. Dertil er scriptsproget PH [20] og databasestyresystemet MySQL [21] installeret og konfigureret til at samarbejde med Apache. For at lette arbejdet med MySQL databasen, er der installeret PhpMyAdmin på webserveren, der et administrationsværktøj til MySQL. [22]

I denne mappe er der opsat to php-script, der opretter forbindelse til den ønskede mysql database og henholdsvis indsætter data fra http forbindelsen,

fra telefonen eller downloader data, i form af bygning, etage og koordinater fra fingerprint sammenligningen.

I denne del har jeg oplevet en del problemer. Mysql installationen har nogle svagheder, som i den grad kom til syne i dette projekt. Hvor jeg i sidste ende, måtte tage den lette udvej og vælge en anden computer, på grund af problemer med "Current root password", fra tidligere sql installation. Serveren er, af praktiske og økonomiske grunde, placeret på en privat computer i hjemmet. Denne computer er tilsluttet en 100/100 mbit forbindelse uden et max forbrug.

Databasen består af en tabel, hvor på de relevante data kan gemmes. Nedenstående tabel angive tabelstrukturen i databasen.

Felt	Type	Eksempel på data	Note
Id	Int(10)	136	Auto genereret
Timestamp	Timestamp	2012-01-23 13:34:53	Auto genereret
Building	Varchar(5)	321	Bygning angivet ved fingerprint
Floor	Varchar(5)	1	Etage angivet ved fingerprint. Bruges ved valg af plantegning.
x_coordinate	Varchar(5)	355	X koordinate angivet ved fingerprint. Bruges ved angivelse af position
Y_coordinate	Varchar(5)	72	y koordinate angivet ved fingerprint. Bruges ved angivelse af position
Ap_mac1	Varchar(45)	00:22:6b:95:37:7e	Accesspoint mac adresse

Ap_strength1	Int(5)	-68	Accesspoint signalstyrke
--------------	--------	-----	-----------------------------

Tabel 1 - Database struktur

Ovenstående tabel er "kun" et udsnit af databasen, for at vise strukturen. Databasen indeholder felter for mac-adresser og signalstyrke, for op til 10 access points.

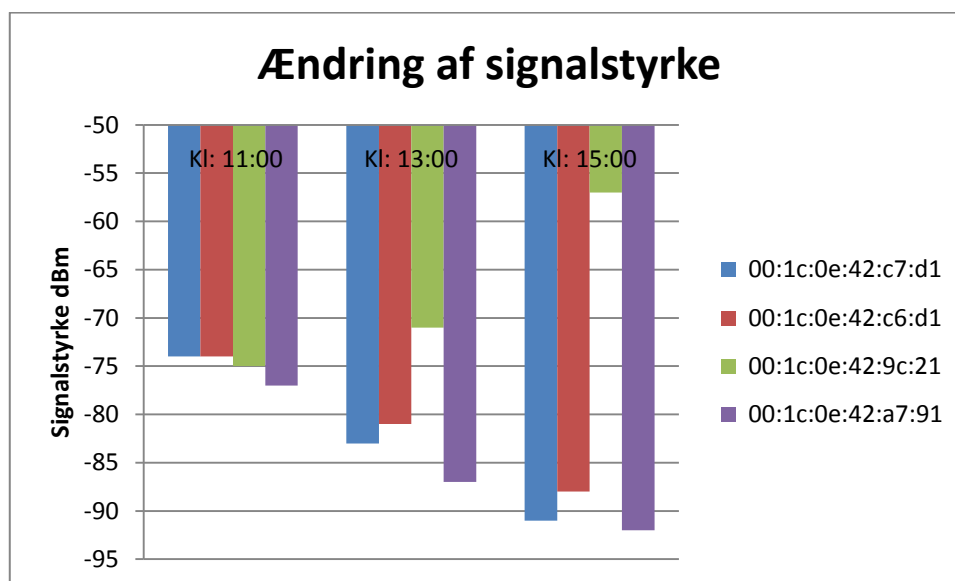
Databasen er opsat således, for at fingerprintet kan indeholde så mange access points som muligt. Test bygningen indeholdte "kun" fire access points, men hver udsendte de op til tre forskellige mac-adresser, hvilket har gjort at det har været nødvendigt at kunne håndtere mere end fire access points, for at undgå scenariet hvor man kun får signaler fra et eller to accesspoint. Dette gør yderligere, at applikationen er lettere at tilpasse til andre bygninger på DTU.

## 6. Test.

### 6.1 Udførelse og resultater.

Test af applikationen er blevet udført i bygninger på DTU, hvor det hovedsageligt er foregået i bygning 321. Derudover er der lavet test løbende i udviklingen. Bygning 321 er en tre etagers bygning med mange små kontorer og ingen store auditorier. Bygningen har fire access points, to i stueetagen, én på første sal og én på tredje sal.

Fingerprint fasen kan give os et overblik over hvor meget udfald, der er på signalstyrken fra access points på forskellige tidspunkter. Fingerprint målingerne er blevet foretaget flere gange på forskellige tidspunkter af dagen og derved med en forskellig belastning. Data fra målingerne blev gemt på databasen og derfra er de blevet analyseret. Målet med dette var at vise, hvordan forholdene ændrer sig på forskellige tidspunkter på en hverdag i den samme bygning.



Figur 10 - Diagram over ændring af signalstyrke

Figur 10 angiver måling af signalstyrkerne fra samme position flere gange på samme dag. Dette viser at belastningen på nogle netværk falder og andre stiger, set ud fra signalstyrkerne. Der skal dog medregnes fysiske forhindringer for signalerne, så som lukkede og åbne døre, andre mennesker samt retning af telefonen. Alle disse forhold kan medvirke til ændringer i signalstyrkerne. Det skal altså ikke kun forstås som, at der er større

belastning på netværket senere på dagen. Det er erfaret at dette vil kræve mange flere målinger, for at kunne præcisere tydeligere. Denne graf skal vise de uregelmæssigheder der opstår ved målingen, foretaget flere gange på samme position.

Positioneringstesten foregik på samme måde som fingerprint fasen, ved at gå rundt på 18 tilgængelige kontorer og lade telefonen bestemme dens placering. Denne test er blevet udført på kontorer, hvor der er foretaget mindst tre fingerprint. Dette har medført, at det ikke er alle kontorer der er blevet testet, da det ikke har været muligt at få adgang til dem.

Følgende data angiver testresultater fra denne test, med faktisk placering og telefonens beregnede placering ved de tre bedste resultater.

<b>Rum</b>	<b>Position 1</b>	<b>Position 2</b>	<b>Position 3</b>	<b>Resultat</b>
<b>133</b>	133	133	133	5
<b>132</b>	132	133	128	3
<b>129</b>	127	127	132	2
<b>128</b>	128	126	123	3
<b>127</b>	127	127	125	4
<b>126</b>	126	127	127	3
<b>120</b>	120	120	120	5
<b>119</b>	119	117	120	3
<b>117</b>	117	116	116	3
<b>116</b>	113	113	111	1
<b>115</b>	116	111	111	1
<b>113</b>	106	109	109	1
<b>112</b>	113	113	109	2
<b>111</b>	111	111	107	4
<b>107</b>	107	107	104	4
<b>106</b>	106	106	104	4
<b>104</b>	104	103	107	3
<b>103</b>	103	103	104	4

Tabel 2 - Position testresultater



Kolonnen Resultat angiver på en skala fra 1 til 5 hvor præcis det positionering har været. Hvor 5 angiver godkendt position og 1 for ikke godkendt position.

Et godkendt resultat er, hvor applikationen placerer enheden i det rigtige rum ved alle positionsmålinger. Et ikke godkendt resultat er, hvor applikationen ikke kan finde den rigtige position.

Fingerprint er foretaget flere gange over flere dage, og test af positionering derefter. Dette har betydet at det er forskellige rum, der har været tilgængelige ved fingerprint og test, hvilket afspejles i ovenstående tabel.

Til yderligere test af applikationen er der gjort brug af Funf software pakken. Funf er en open source sensing framework.<sup>5</sup> [23] Denne software pakke indeholder en applikation til android der kan indsamle målinger fra telefonens hardware. Denne data kan sendes til en computer og analyseres. Denne del af testen er udført således:

Applikationen er sat til at måle efter 'nearby Wifi devices' undervejs i en fingerprint fase. Denne data sendes til en computer der analyserer data og med denne data får koordinaterne for hver måling. Disse koordinater sammenlignes med den rute hvorpå fingerprint fasen foregik.

Data fra målingen med Funf software er blevet analyseret og tegnet som en rute på en plantegning over etage. Denne rute er sammenlignet med den rute, hvorpå der blev foretaget målinger. Denne test var ikke nogen succes set ud fra sammenligning. Her viser ruten fra Funf målingen en meget upræcis rute i forhold til den faktiske rute. Se bilag B.

Undervejs i testen af positionering, opstod der situationer, hvor applikationen ikke kunne finde nogen matchende position. Dette opstod hovedsageligt i omkring trappen i bygningen. (Rum 104). På denne position, var det begrænset hvor mange access points applikationen kunne finde. Dette kan ses på inputtet på databasen, hvor der nogle gange kun er uploadet informationer for ned til kun to access points.

---

<sup>5</sup> <http://funf.media.mit.edu/>

## 6.2 Problemer under testfasen.

Under test af applikationens fingerprint funktioner, fremkom der mindre problemer med svartiderne fra serveren, hvilket betød at fingerprinting delen "hang" meget og forsinkede arbejdet. Dette kan højst sandsynligt skyldes serverens placering på en privat adresse.

Et andet problem, der kom frem under testfasen, var, at signalstyrkerne ikke bare ændrede sig alt efter belastning og placering i forhold til access points, men at der ved skanning af netværket var udefrakommende access points. Det er den store ulempe ved fingerprint metoden, hvilket viste sig ved uregelmæssigheder i netværksstrukturen. Disse access points, har kunnet forringe kvaliteten af positioneringen eller slet ikke givet nogen positionering, da match algoritmen ikke tager højde for nye access points. Dette problem kan opstå ved at blandt andet bærbare computere der befinder sig i bygningen, er opsat til at fungere som wifi hotspot, eller i værste tilfælde ved ændring i wireless infrastrukturen hvor fingerprintet er foretaget.

Selve dataindsamling blev besværliggjort af den begrænsede adgang til de enkelte kontorer. Derfor indeholdt den endelig test ikke data fra alle kontorer, men "kun" de kontorer der var adgang til.

## 7. Diskussion.

---

Der er mange punkter i dette projekt der har kunnet været udført anderledes og bedre. Men set i lyset af at dette projekt har krævet tilegnelsen af en masse ny viden omkring udvikling af mobilapplikation og server med database kan det udledes, at målet med dette projekt har været sat højt.

Ved ændringer i planen - som ofte sker i ethvert andet projektforsøg opstod der udfordringer for projektet og forhindrede at projektet overholdt

tidsplanen og det tiltænkte endelige produkt. Selv om det er svært at gardere sig i mod uforudsete forhindringer, har en bedre planlægning, kunne give bedre resultater og der af færre forhindringer.

Resultatet af dette projekt, blev en applikation, der ved brug af fingerprint metoden, kan indsamle fingerprint data og udføre en positionering, dog med en hvis usikkerhed. Denne usikkerhed, der beskrives i test afsnittet, skyldes blandt andet manglende fingerprint data, samt det lille antal af tilgængelige access points.

Et bedre resultat kunne have været opnået med en større fingerprint database, men på grund af opgaven størrelse, har det ikke været muligt på samme tid, at lave en meget stor fingerprint database.

Testresultaterne angiver, at der forekommer, en del upræcise positioneringer, hvor applikationen i 13 ud af 18 målingspositioner opnåede at få den rigtige position, mindst en gang. Derudover opnåede kun to ud af de 18 målingspositioner at få den rigtige position ved tre ud af tre forsøg.

Selve fingerprint metoden, hvor fingerprint informationerne indsamles, blev ændret undervejs i forløbet. Fra først, at blive angivet med information omkring bygning, etage og rum, der viste sig at være en mindre god løsning, blev ændret til bygning, etage og koordinater. Denne ændring blev udført for at lette arbejdet med visualisering af positionering, samt muligheden for implementering af trilateration metoden.

Bygningen hvor testen blev udført, var "kun" udstyret med fire access points på flere etager og der kan argumenteres for at det er i underkanten for vellykket indendørs positionering. Fingerprint metoden kan, som vist ved applikationen, fungere til en hvis grad, men ved flere tilgængelige access points samt flere målinger, kunne det give en bedre præcision. Andre metoder som triangulation og trilateration, kræver mindst tre tilgængelige access points og deraf kan der med fire access points i bygningen, opstå situationer, hvor der er dårligt eller slet ingen signal fra access points.

Testen viste, at der var en forskel i signalstyrkerne på forskellige tidspunkter på dagen. Den viste en ændring på op til 20 dBm, hvilket viser nødvendigheden af en stor fingerprint database.

## 8. Perspektivering.

---

I det følgende vil der gennemgås hvilke forbedringer der kan gøres ved positioneringsmetoden. Desuden ved feature work for applikationen, der beskriver ønskede funktioner.

For at få det bedste resultat ved fingerprint metoden, kræver det en større dataindsamling ved fingerprint fasen for derved at kunne dække flere forskellige belastninger på netværket.

Future work for applikationen.

Udviklingen af applikationen har budt på en del udfordringer, og mængden af funktioner er blevet nedjusteret flere gange. Der har været mange ønsker for applikationen, der beskrives i det følgende.

Navigationens funktionalitet - Det skal være muligt at indtaste placeringen man ønsker at blive navigeret hen til. Dette er en omfattende funktionalitet, set ud fra hvis applikationen skal kunne tilpasses flere bygninger. Dette kræver en destinationsdatabase, med navne og kontonr, der hver indeholder bygning, etage og koordinater.

Navigationen skal foregå i real-time og vise afstand. Dette vil kræve en mindre ændring for den eksisterende wifiscanner kode, men samtidig en omfattende ændring omkring database. Et alternativ til dette vil være at gøre brug af trilaterations metoden til løbende beregning af placering.

Find venner - Mulighed for at give andre mulighed for at vise hvor man er placeret i bygningen. Dette kræver dog en del, da man er nød til at lave en Bruger-funktion, evt. med login for sikkerhed. Dette vil gøre det lettere for personer at finde hinanden i store bygninger. Dette kan integreres med Check-ind funktionen kendt fra Google map.

Location based messaging. - Mulighed for brugere, at sende beskeder rundt til andre brugere med samme placering. Dette kan eventuelt være grupper man er tilmeldt, hvor en administrator, kan sende fælles meddelelser ud.

"Hvad er der her" funktion – Funktion kendt fra Google maps, der giver brugeren mulighed for at se, hvad der er placeret omkring en. Vist på liste og på kort. Dette vil være muligt at implementere i den eksisterende kode, dog vil det kræve en omfattende database, der løbende vil skulle opdateres.

## 9. Konklusion.

---

Dette projekt har beskrevet udviklingen af en applikation til indendørs positionering. Projektet har budt på mange udfordringer, der har medført en del ændringer undervejs i projektet. Ønsket var at udvikle en applikation, der kan fungere som indendørs navigation og kalibreringsværktøj. Applikationen skulle gøre brug af nogle af de positioneringsmodeller beskrevet i analysen og sammenligne positionerne med fingerprint metoden. Dette viste sig at være en meget stor opgave og målet blev nedjusteret til en applikation, der fungerer udelukkende via fingerprint metoden.

Den endelige applikation, der blev udviklet, er gjort så simpel som mulig, hvad brugergrænsefladen angår. Der er dog implementeret muligheden for touch-input fra brugeren til at angive og visualisere position, ved fingerprint og positionerings faserne.

Afslutningsvis kan det konkluderes, at det er lykkedes at udvikle en mobilapplikation, der kan fungere som indendørs positionerings system og som et kalibreringsværktøj. Dette projekt viser nogle metoder, hvorpå dette kan udføres, samt mulighederne for udvikling på flere platforme. Testresultaterne viste, at der var en forholdsvis stor variation i signalstyrkerne, hvilket har haft indflydelse på at positioneringstesten gav nogle resultater, med ringe præcision. Som beskrevet kunne signalstyrkerne over en periode, varierer med op til 20 dBm.

Præcisionen af positioneringen, afhænger i høj grad af hvor mange access points, der er tilgængelige ved skanningen. Den udviklede applikation, har problemer med at finde matchende fingerprint, når der ikke er tilstrækkelig data til sammenligning. Det kan deraf afledes, at jo flere access points der er tilgængelige, jo bedre en præcision kan opnås.

## 10. Kilder.

---

- [1] Ekahau Wifi Tracking Systems, RTLS and WLAN Site Security  
<http://www.ekahau.com/products/real-time-location-system/overview.html> (sidst verificeret 30-01-2012)
- [2] Skejby Sygehus, indendørs lokaliseringssystem.  
[http://atea.dk/download/atea\\_cases2009/Aarhus%20Universitetshospital.pdf](http://atea.dk/download/atea_cases2009/Aarhus%20Universitetshospital.pdf)  
(sidst verificeret 30-01-2012)
- [3] The Bluetooth Special Interest Group (SIG) - About the Technology  
<http://www.bluetooth.com/Pages/About-the-Technology.aspx> (sidst verificeret 30-01-2012)
- [4] Kamol Kaemarungsi and Prashant Krishnamurthy -  
*"Modeling of Indoor Positioning Systems Based on Location Fingerprinting."*  
School of Information and Science, University of Pittsburgh
- [5] Binghao Li, James Salter, Andrew Dempster, Chris Rizos -  
*"Indoor Positioning Techniques Based on Wireless LAN"*  
School of Surveying and Spatial Information System, University of New South Wales, Sydney and  
School of Computer Science and Engineering, University of New South Wales, Sydney
- [6] Binghao Li, Andrew Dempster, Chris Rizos, Joel Barnes -  
*"Hybrid Method for Localization Using WLAN"*  
School of Surveying and Spatial Information System, University of New South Wales, Sydney
- [7] Matthias Lott and Ingo Forkel -  
*"A Multi-Wall-and-Floor Model for Indoor Radio Propagation"*  
Siemens AG, Information and Communication Mobile and RWTH Aachen Communication Networks
- [8] Nicola Lenihan - *"A Local Optimal User Position System for Indoor Wireless Devices"*, University of Limerick
- [9] Android leads US market share, iOS may have stopped growing, RIM is still falling -  
<http://www.engadget.com/2011/12/14/shocker-android-grew-us-market-share-after-q2-ios-was-static/> (sidst verificeret 30-01-2012)
- [10] Gartner : Android OS Rose to Account for More Than 50 Percent of Smartphone Sales  
<http://www.gartner.com/it/page.jsp?id=1848514> (sidst verificeret 30-01-2012)
- [11] Android Open Source Project - Philosophy and Goals  
<http://source.android.com/about/philosophy.html> (Sidst verificeret 30-01-2012)

- [12] Android developers - WifiManager  
<http://developer.android.com/reference/android/net/wifi/WifiManager.html> (sidst verificeret 30-01-2012)
  
- [13] Blackberry API reference WLANInfo  
<http://www.blackberry.com/developers/docs/7.1.0api/net/rim/device/api/system/WLANInfo.html> (sidst verificeret 30-01-2012)
  
- [14] BlackBerry App World Vendor Support  
<http://us.blackberry.com/developers/appworld/> (Sidst verificeret 30-01-2012)
  
- [15] Apple developer - <http://developer.apple.com/> (sidst verificeret 30-01-2012)
  
- [16] Google Code - iphone wireless  
<http://code.google.com/p/iphone-wireless/> (sidst verificeret 30-01-2012)
  
- [17] Microsoft Application Certification Requirements for Windows Phone  
[http://msdn.microsoft.com/en-us/library/hh184843\(v=vs.92\).aspx](http://msdn.microsoft.com/en-us/library/hh184843(v=vs.92).aspx) (sidst verificeret 30-01-2012)
  
- [18] Android developer - Installing SDK  
<http://developer.android.com/sdk/installing.html> (Sidst verificeret 30-01-2012)
  
- [19] Apache HTTP Server Project  
[http://httpd.apache.org/ABOUT\\_APACHE.html](http://httpd.apache.org/ABOUT_APACHE.html) (Sidst verificeret 30-01-2012)
  
- [20] Php script language  
<http://php.net/> (sidst verificeret 30-01-2012)
  
- [21] MySQL - Open Source Database  
<http://www.mysql.com/> (sidst verificeret 30-01-2012)
  
- [22] phpMyAdmin - MySQL administration tool  
[http://www.phpmyadmin.net/home\\_page/index.php](http://www.phpmyadmin.net/home_page/index.php) (Sidst verificeret 30-01-2012)
  
- [23] Funf - Open source sensing Framework  
<http://funf.media.mit.edu/> (Sidst verificeret 30-01-2012)



# 11. Tabeller og figurer.

---

Figur 1 - Fingerprint faser .....	6
Figur 2 - Trilateration .....	7
Figur 3 - Flow diagram .....	14
Figur 4 - Startvindue med menu.....	15
Figur 5 - Position map.....	16
Figur 6 - Position map menu.....	16
Figur 7 - Fingerprint position .....	17
Figur 8 - Fingerprint .....	17
Figur 9 - Fingerprint menu.....	18
Figur 10 - Diagram over ændring af signalstyrke.....	26
Figur 11 - Start vindue    Figur 12 - Upload fingerprint.....	39
Figur 13 - Klasse diagram.....	40
Tabel 1 - Database struktur .....	25
Tabel 2 - Position testresultater .....	27

## 12.1 Bilag A – Filer.

---

Her er en liste over de filer der følger med rapporten på medfølgende cd-rom.

ip\_touch\_v1.0.apk - installations filen til mobil applikationen.

/ip\_touch/ - Indeholder Eclipse projektet.

- /src/ - Kildekoden til applikationen

setdata.php - php script der indlæser fingerprint fra telefonen.

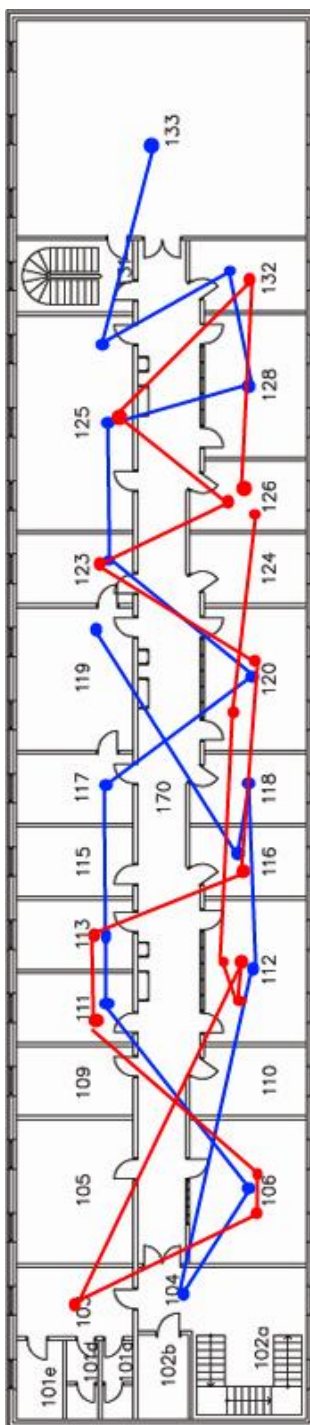
getdata.php - php script der sammenligner og returnerer fingerprint.

fingerprint\_touch\_tabel.xlsx - Excel fil med data fra fingerprint databasen.

## 12. 2 Bilag B - Testresultater med Funf framework.

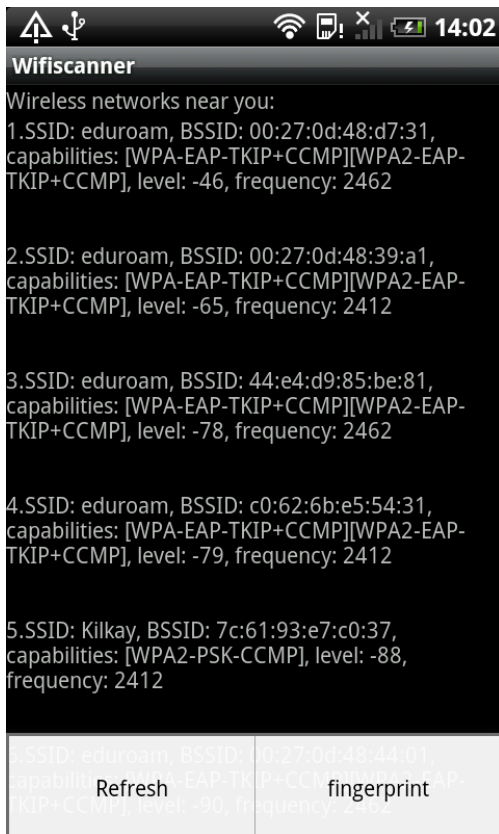
---

Sammenligning mellem den faktiske rute (blå) og ruten baseret på resultatet fra Funf målingerne (Rød). Start i rum 133.

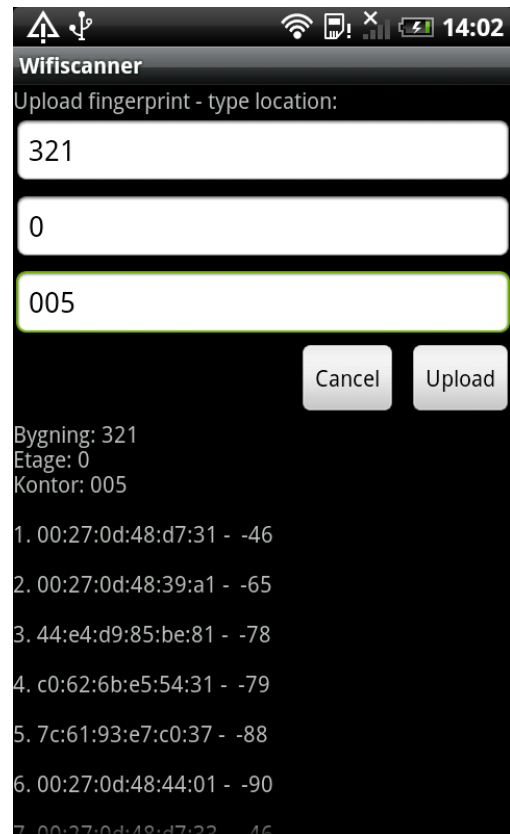


## 12.3 Bilag C - Tidligere versioner af applikationen.

Nedenstående screenshots viser tidligere versioner af applikationen.

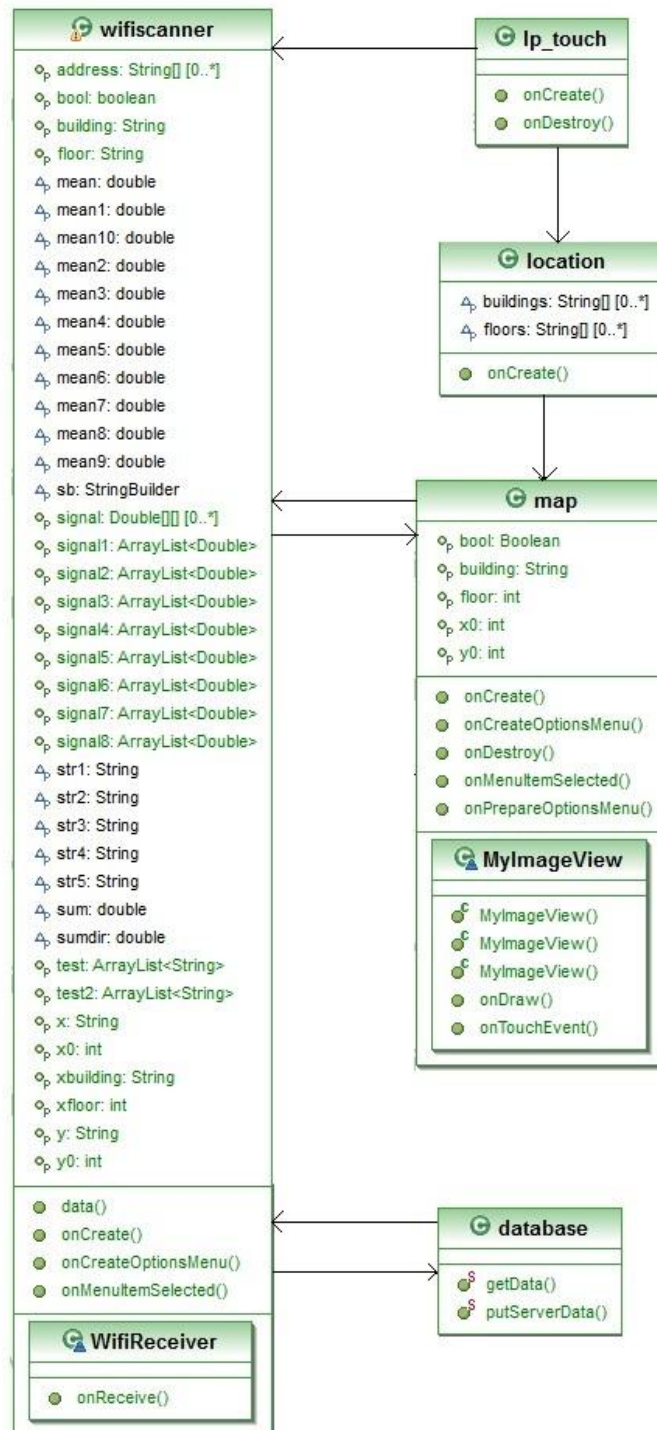


Figur 11 - Start vindue



Figur 12 - Upload fingerprint

## 12.4 Bilag D - Klasse diagram.



Figur 13 - Klasse diagram