RFID NAVIGATION

af

Mesim Bajrami

Vejleder: Robin Sharp

Informatik Civilingeniør
Eksamensprojekt

IMM-THESIS-2006-78

Danmarks Tekniske Universitet
Informatik og Matematisk Modellering
Bygning 321, DK-2800 Kongens Lyngby, Denmark

Technical University of Denmark
Informatics and Mathematical Modelling
Building 321, DK-2800 Lyngby, Denmark
Phone +45 45253351, Fax +45 45882673
reception@imm.dtu.dk
www.imm.dtu.dk

Resumé

Beregning af optimale ruter og navigering har været og er et problem i mange situationer, hvilket igennem tiden har medført forskellige løsningsmodeller. I dag forbinder man ofte et navigationssystem med GPS navigering som er meget udbredt blandt befolkningen og ikke mindst militæret. En af ulemperne ved GPS navigering er at den ikke kan anvendes indendørs pga. usikkerheden på ca. ± 10 m, samt at den kræver frit udsyn til mindst tre af de 24 GPS satellitter, der er i konstant kredsløb med jorden.

Behovet for indendørs navigering kan være til stede i større bygninger som man ikke kender, f.eks. hospitaler eller lign. Behovet vokser endda stærkt hvis det gælder personer med et handicap som f.eks. personer i rullestol. I sådan et tilfælde vil det være rart med et system som kan vise den korteste vej mellem to punkter og samtidigt undgå trapper eller elevatorer. I dag må man nøjes med at følge skiltning som ikke altid er lige præcis og nem at forstå.

I denne rapport undersøges mulighederne for at anvende RFID (Radio Frequency Identification) teknologien som et hjælpemiddel til at løse indendørs navigeringsproblemet. RFID teknologien har udviklet sig meget igennem de sidste år, og har fået en stigende interesse blandt virksomhederne der ønsker at anvende teknologien til automatisk identificering af objekter.
Rapporten er en del af et større projekt hvor mulighederne for anvendelse af RFID til formålet er blevet undersøgt i praksis, i form af test af tilgængeligt udstyr samt implementering af en prototype.

I rapporten undersøges forskellige løsningsmodeller for en eventuel løsning med henblik på at belyse fordele og ulemper. Den teoretiske baggrund for repræsentation af digitale kort, i form af matematiske grafer undersøges. Forskellige algoritmer der kan anvendes på graferne til at beregne optimale ruter mellem to punkter undersøges også.
Et krav til sådan et system er bl.a. sikkerheden, hvilket også bliver undersøgt i rapporten. Herved undersøges evt. svagheder og trusler, samt de ting som skal være til stede for systemet kan sikres.

De afsluttende bemærkninger for projektet konkluderer at systemet rent teoretisk og praktisk kan realiseres. Der er en række sikkerhedsproblemer samt stabilitetsproblemer ved RFID udstyret, som kræver nærmer analyse. Ydermere er den økonomiske del af projektet slet ikke undersøgt her, hvilket ligeledes kræver nærmere analyse inden evt. videreudvikling af projektet.

Abstract

Calculation of optimal paths and navigation has been and is still a problem in many situations, which through the years has resulted in different solution models. Nowadays one often relates a navigation system with GPS navigation which is very widely distributed among the population and especially in the military. One of the disadvantages with the GPS navigation is that it cannot be used indoor because of the uncertainty at approx. ±10 m and the need for unobstructed view for at least three of the 24 GPS satellites which are in earth orbit.

The need for indoor navigation can appear in greater buildings that one is not aware of, for instance in hospitals. The need for the system grows even more when it concerns people with a handicap, for instance people in a wheelchair. In these cases it would be comfortable to have a system that can show the shortest path between two locations and at the same time avoid stairways or lifts. Nowadays the posting of signs is applied which is not always equally well and easy to understand.

In this thesis, the possibility to use RFID (Radio Frequency identification) technology as a remedy to solve the indoor navigation problem will be investigated. The RFID technology has matured through the last period of years and it has received an increasing interest among companies who find it desirable to use the technology for auto identification.
This thesis is part of a larger project where the possibilities for the use of RFID to the purpose has been examine in practice by testing the available equipment and implementing a prototype.

This thesis examines different solutions to illuminate advantages and disadvantages. The theoretical background for the representation of the digital map in the way of mathematical graphs is examined together with the different algorithms that can be used on the graphs to calculate optimal paths between two locations.
One of the requirements for such a system is among other factors the security, which will also be examined in this thesis. Vulnerabilities and threats for this system are examined and solutions for ensuring confidentiality, integrity and availability.

The final remarks for this thesis conclude that the system can become reality in theory and practice. A number of security issues and stability problems with the RFID equipment require further investigation. Furthermore the economic part of the project has not been examined which as well require analysis on closer examination before further development of the project.

Indholdsfortegnelse

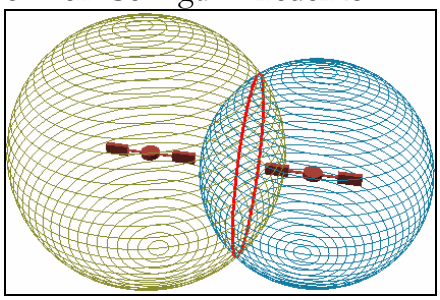
Kapitel 1

Introduktion

Problemer eller besvær med navigering er et fænomen som man altid har kendt til. For mange tusinde år tilbage anvendte man simple løsninger som afmærkninger af en rute, hvilket gjorde at man kunne finde tilbage. Senere hen stiftede man bekendtskab med astronomi og kunne anvende stjernernes position, samt solen som hjælpemiddel til navigering når man var til havs, ørknen og andre steder.

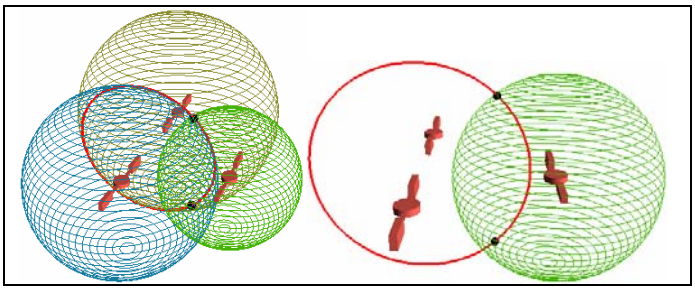
I nyere tid er skiltningen og infrastrukturen blevet meget bedre hvilket letter denne opgave. Men nye navigeringsmetoder er også opfundet. Den mest almindelige form for navigering i dag er vha. GPS (Global Positioning System) satellitter. Kort fortalt fungerer metoden ved hjælp af 24 GPS satellitter fordelt i kredsløb om jorden i seks baneplaner. Dette svarer til at der altid er seks eller syv satellitter over et punkt på jorden.

En GPS modtager kan bestemme positionen på jorden ud fra afstanden til satellitterne. Den skal således bruge afstanden til mindst 3 satellitter for at kunne bestemme positionen vha. trigeometri. Afstanden til den første satellit danner en kugle med afstanden som radius, og placeringen kan således være ethvert punkt på kuglen. Afstanden til den anden satellit danner ligeledes en kugle, og skæringen mellem de to kugler danner en cirkel som begrænser mulighederne til at positionen er et punkt på cirklen. Se Figur 1 nedenfor.



Figur 1 - Afstanden til to satellitter begrænser position til en banecirkel

Skæringen med den tredje kugle begrænser mulighederne til to punkter på cirklen, og herefter kan GPS modtageren anvende jordens overflade som den 4. kugle til at udpege hvilket punkt som svarer til positionen på jorden. Se Figur 2 nedenfor.



Figur 2 - Skæringen med den 3. kugle fra den 3. satellit begrænser positionen til to punkter

Positionen angives som et punkt i rummet, med længde- og breddegrader, samt højden over vandets overflade. I moderne navigationssystemer, som man bl.a. kender fra biler, bliver ens position vist på et kort, da det er nemmere at forstå. Dette muliggøres vha. digitale kort som associerer koordinaterne med bestemte steder på kortet.

GPS navigation har en usikkerhed på op til 10 meter, hvilket ikke normalt betyder så meget når man skal navigerer rundt på jordens overflade. Men ønsker man en mere præcis navigering, måske på cm niveau kan GPS ikke anvendes. Ydermere er GPS navigation afhængig af frit udsyn til satellitterne, hvilket betyder at det ikke kan anvendes indendørs.

Hvis man aldrig har haft problemer med at finde rundt i store bygninger kan hele idéen med indendørs navigering være svær at få øje på. Men faktum er at det kan være svært at finde rundt i store bygninger, som f.eks. hospitaler man ikke kender, og det kan især være svært at vide hvordan man kommer fra et sted til et andet sted hurtigst muligt. Behovet kan være til stede og endnu større for folk med et handicap. Personer i rullestol ønsker f.eks. den korteste rute som kun gør brug af elevator, og ikke trapper. Denne problemstilling ligger til grundlag for denne opgave, hvor vi undersøger RFID som en teknologi til indendørs navigering.

1.1 Rapportens indhold

I kapitel 2 beskrives RFID teknologierne, efterfulgt af fordele og ulemper sammenlignet med andre lignende teknologier til automatisk identificering. Der vil også være en kort opsummering af anvendelsesscenarios for RFID, således at læseren kan danne sig et indtryk af den teknologi, som resten af rapporten omhandler.

I kapitel 3 tages der udgangspunkt i problemstillingen, og forskellige løsningsmodeller for et indendørs navigationssystem fremstilles. De forskellige løsninger diskuteres med henblik på at belyse fordele og ulemper.

I kapitel 4 ser vi på grafer som kan anvendes til opbygning af digitale kort over en bygning, samt forskellige algoritmer til rutebestemmelser på en graf. Den bedste rute er ikke nødvendigvis den korteste rute, hvilket leder os til en undersøgelse af måleparametre for rutebestemmelserne.

I kapitel 5 skal vi se nærmere på sikkerheden i RFID teknologien samt i et samlet system som den der beskrives her i rapporten. I dette kapitel vil man kunne læse hvad der skal til for at systemet kan betragtes som sikkert, samt hvordan denne sikkerhed opnås hvis muligt.

I kapitel 6 beskrives den prototype som er udviklet i forbindelse med projektet. Prototypen skulle vise om hele idéen med RFID navigering kan implementeres i den virkelige verden. Detaljeret udviklingsnotaer vedlægges som bilag samt på en CD med kildekoden til prototypen.

Rapporten slutter af med en konklusion i form af afsluttende bemærkninger som kommer i kapitel 7. Her opsummeres resultaterne samt muligheden for fremtidige forbedringer.

Som bilag er der vedlagt prototypens udviklingsnota samt en oversigt for de forskellige RFID standarder. Ydermere er der vedlagt en Cd-rom som indeholder en demo video af prototypen i funktion, installationsfil til prototypen og alt kildekode.

Kapitel 2

RFID Teknologien

I de fleste industrier er der behov for automatisk identificering. Der er behov for identificering af mennesker, dyr og materielle varer. Stregkoden er den mest udbredte form for automatisk identificering af varer, primært fordi metoden er billig. Ulemperne ved denne er dog at den bl.a. kan indeholde begrænset information, samt kræver synlighed for at kunne aflæses med en stregkodelæser.

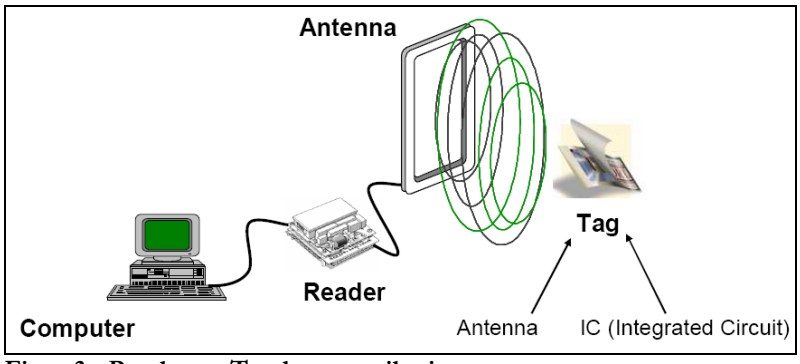
En anden måde at opbevare information på er på en chip, som vi allerede kender fra telefonkort og lign. Her er mængden af informationer større, men aflæsningen kræver kontakt mellem læseren og chippen som indeholder informationen. Den optimale måde ville være at anvende en passiv (uden intern strømkilde) og billig chip, som vil kunne aflæses på afstand uden at være synlig for læseenheden. Sådan en service kan opnås med RFID (Radio Frequency Identification), hvilket er årsagen til den stigende interesse inden for området blandt store virksomheder som ønsker sporbarhed i deres varer.

Et RFID system består altid af to dele:

- *Interrogator*, også kaldet *reader* er i stand til at læse et tag. I nogle tilfælde kan den også omprogrammere et tag.
- *Transponder,* også kaldet et *tag*, som placeres på den enhed man ønsker at identificerer.

En reader er et elektronisk kredsløb med en eller flere antenner som har til formål at læse informationer fra tags. De fleste readere kan læse flere tags på samme tid, ved at sende en forespørgsel i form af radiobølger, og derefter lytte på svar fra tags.

Ordet *Transponder* er en sammensætning af ordene TRANSmitter og resPONDER hvilket siger noget om dens funktion. Dette betyder at en transponder besvarer en forespørgsel ved at sende den information som den har i sin hukommelse.



Figur 3 - Reader og Tag kommunikation

Figur 3 viser sammenspillet mellem et tag og en reader som er tilkoblet et computersystem.

2.1 Forskellige teknologier

Alle RFID teknologier har reader og transponder til fælles og den grundlæggende måde de fungerer på er også ens. Men alligevel findes der mange forskellige RFID teknologier fra forskellige producenter. I følgende afsnit skal vi se hvordan RFID teknologierne kan kategoriseres og hvordan de adskiller sig fra hinanden, dette er også illustreret af Figur 4 nedenfor.

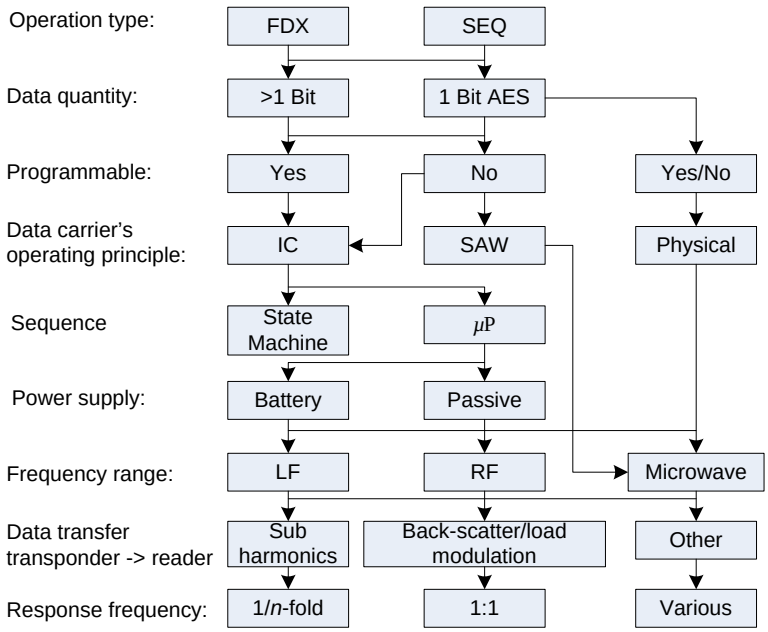
Et RFID system opererer efter et af følgende to principper:

- FDX/HDX (Full duplex/half duplex)
- SEQ(Sequential systems)

FDX/HDX systemer fungere ved at transponderen svarer tilbage så snart readerens RF felt er slået til. Det betyder at svaret kan komme på et tidspunkt hvor readerens antenne fortsat sender energi ud. Readerens eget signal kan være meget kraftigere end det signal som modtages fra transponderen. Dette kræver en bestemt procedure implementeret i kommunikationsprocessen for at readeren kan aflæse signalet fra en transponder. I praktisk forgår dette vha. *load modulation*, som kan adskille transponderens signal fra readerens eget signal. Load modulation forklares senere i rapporten.

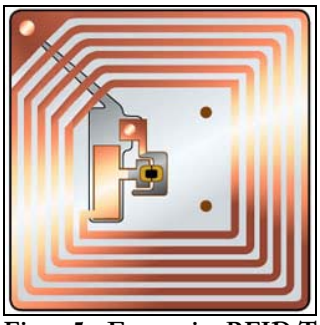
SEQ systemer opererer ved at readerens signal slås fra i faste intervaller, og dette interval udnytter transponderen til at sende sit svar i. Ulempen ved sådan et system er at den energi som transponderen ikke får i det interval hvor readeren er passiv, skal komme fra en ekstern energi kilde, som f.eks. batteriet.

Mængden af data som kan indeholdes af en transponder kan også variere fra få bytes til flere kilobytes. Men der findes også 1-bit transpondere, som er billigere at fremstille og anvendes typisk i alarm systemer på butiksvarer. Disse transpondere kan kun signalere at der findes et tag inden for readerens læsevidde.



Figur 4 - Forskellige funktioner i RFID systemer

Transponderens energikilde er også en måde at klassificere et RFID system på. En transponder kan være passiv, semi-passiv eller aktiv og består af en antenne som er forbundet til en chip.
En passiv *transponder* har ikke nogen intern energikilde, dvs. batteri eller lign., men får dens energi ved induktion fra de radiobølger som *readeren* sender med en forespørgsel.



Figur 5 - Et passivt RFID Tag

Semipassive *transpondere* minder om de passive, med den forskel at de har en lille energikilde, typisk et batteri, som gør dem i stand til konstant at være klar til at besvare forespørgsler. Det betyder, at de ikke først skal generere energi til besvarelsen og er derfor hurtigere end de passive tags.

En aktiv *transponder* har sin egen energikilde, typisk i form af et større batteri. De aktive transpondere er mere pålidelige og især i områder med væsker eller anden materiale som absorberer radiobølger. Aktive tags har

en større rækkevidde og kan derfor anvendes i områder hvor de passive tags ikke er tilstrækkelige.

Ud over energi kilden på transponderen kan man også klassificere et RFID system ud fra måden man skiver data i transponderen på. Der findes simple transpondere hvor data er forud programmeret under fremstillingen og kan ikke ændres efterfølgende. Men for de transponder som kan omprogrammeres, findes der tre typiske måde at gemme data på:

1. **EEPROMs** (electrically erasable programmable read-only memory) som er den mest udbredte metode blandt induktive RFID systemer. Ulempen ved EEPROMs er det store energiforbrug under skrive processen, samt et begrænset antal skrivecykluser (typisk 100.000 til 1.000.000).
2. **FRAMs** (ferromagnetic random access memory) er kun anvendt og testet i isoleret omgivelser. Sammenlignet med EEPROMs er skrivehastigheden 1000 gange større og strømforbruget 100 gange mindre. Men fremstillingsproblemer har forhindret dens udbredelse på markedet.
3. **SRAMs** (static random access memory) bruges især af mikrobølge systemer til at opbevare data. SRAMs er hurtig, men kræver konstant strøm fra et batteri for at fastholde data.

Transponderene produceres i forskellige former og indpakninger. Indpakningen er ofte afgørende for transponderenes anvendelse i et bestemt miljø, hvor temperatur og andre forhold kan være ekstreme. Eksempelvis fremstilles transpondere i glasindpakning, som egner sig til implantering under huden på dyr og mennesker. Plastik indpakning er også en mulighed, eller ingen indpakning men som smart label der kan klistres på f.eks. pakker.

Et af de vigtigste områder at klassificere et RFID system på er ud fra frekvens og rækkevidde, hvilket beskrives nærmere i næste afsnit.

2.2 Frekvens og Rækkevidde

Koblingen mellem transponder og reader, samt den anvendte frekvens og rækkevidde inddeler RFID systemerne i tre kategorier:

- Close coupling systems
- Remote coupled systems/HF or RF (high frequency or radio frequency
- Long-range systems

Close Coupling Systems, er de systemer som typisk kun rækker op til 1 cm. Det betyder at reader og transponder skal være meget tæt på hinanden for at kunne kommunikere, og disse systemer kan opererer i frekvensområdet fra DC til 30MHz. Pga. den tætte tilkobling kan man her bruge transpondere med energikrævende mikroprocessorer. Disse systemer anvendes ofte til løsninger hvor sikkerheden er vigtigere end rækkevidden. Det kan f.eks. være betalingssystemer, eller adgangssystemer.

Remote Coupled Systems, er de systemer som kan læse og skrive tags fra en afstand på op til 1 m. De fleste af disse systemer er baseret på en induktiv (magnetisk) kobling mellem reader og transponder, hvilket er gældende for ca. 90 % af alle RFID systemer der bliver solgt i dag[1]. Ydermere findes der kapacitiv (elektrisk) kobling, som er mindre udbredt blandt disse systemer. Remote Coupled Systems opererer typisk i frekvensområdet under 13,56MHz.

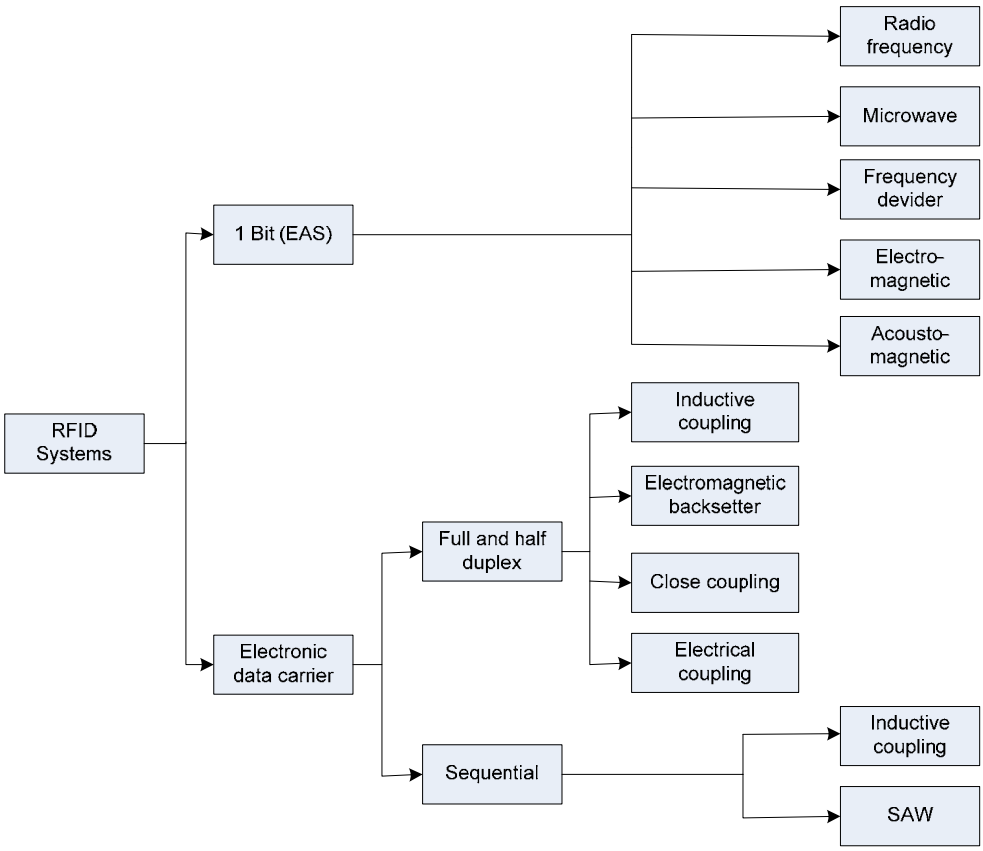
Long-range Systems, har en rækkevidde over 1m, og opererer med elektromagnetiske bølger i UHF og mikrobølgeområdet. De fleste af disse systemer betegnes som *backscatter systems* pga. deres virkemåde, hvilket vi vender tilbage til. Disse systemer opererer på UHF frekvensen 868MHz (Europa) og 915MHz (USA) og mikrobølgeområdet på 2,5GHz og 5,8GHz. Den typiske rækkevidde for passive backscatter transponder er op til 3 m, men kan komme op på 15m for aktive backscatter transponder.

Frequency ranges for RFID-Systems		
Frequency range	**Comment**	**Allowed fieldstrength / transmission power**
< 135 kHz	low frequency, inductive coupling	72 dBµA/m max
3.155 ... 3.400 MHz	EAS	13.5 dBµA/m
6.765 .. 6.795 MHz	medium frequency (ISM), inductive coupling	42 dBµA/m
7.400 .. 8.800 MHz	medium frequency, used for EAS (electronic article surveilance) only	9 dBµA/m
13.553 .. 13.567 MHz	Medium frequency (13.56 MHz, ISM), inductive coupling, wide spread usage for contactless smartcards (ISO 14443, MIFARE, LEGIC, ...), smartlabels (ISO 15693, Tag-It, I-Code, ...) and item management (ISO 18000-3).	60(!) dBµA/m
26.957 .. 27.283 MHz	medium frequency (ISM), inductive coupling, special applications only	42 dBµA/m
433 MHz	UHF (ISM), backscatter coupling, rarely used for RFID	10 .. 100 mW
865 .. 868 MHz	UHF (RFID only), Listen before talk	100 mW ERP Europe only
865.6 .. 867.6	UHF (RFID only), Listen before talk	2W ERP (=3.8W EIRP)

MHz		Europe only
865.6 .. 868 MHz	UHF (SRD), backscatter coupling, new frequency, systems under developement	500 mW ERP, Europe only
902 .. 928 MHz	UHF (SRD), backscatter coupling, several systems	4 W EIRP - spread spectrum, USA/Canada only
2.400 .. 2.483 GHz	SHF (ISM), backscatter coupling, several systems,	4 W - spread spectrum, USA/Canada only
2.446 .. 2.454 GHz	SHF (RFID and AVI (automatic vehicle identification))	0.5 W EIRP outdoor 4 W EIRP, indoor
5.725 .. 5.875 GHz	SHF (ISM), backscatter coupling, rarely used for RFID	4 W USA/Canada, 500 mW Europe

Tabel 1 - Frekvensområder for RFID teknologier [26]

Diagrammet nedenfor viser en oversigt over forskellige RFID systemer.



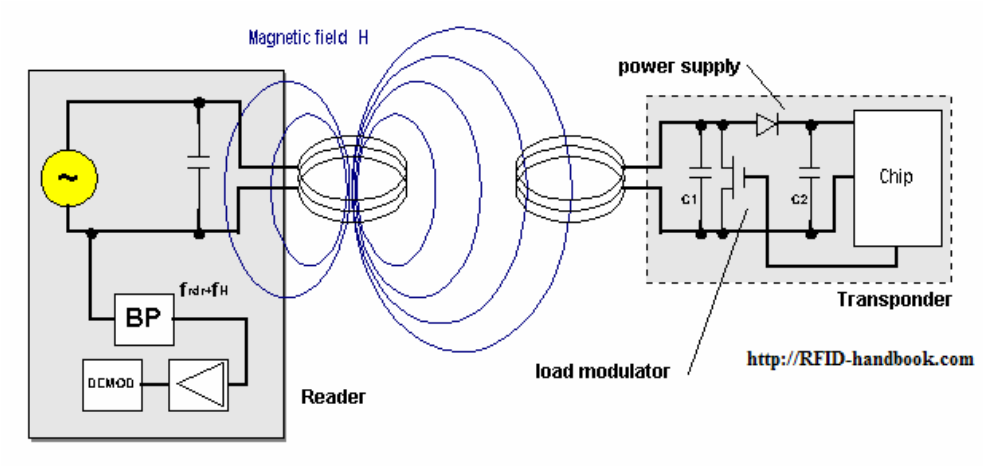
Figur 6 - Forskellige RFID systemer[1]

Ud over at klassificere transmissionsfrekvenserne i de tre hovedkategorier, så vi at man kunne underinddele i rækkevidde ud fra deres kobling: close-coupling (0-1cm), remote-coupling (0-1m) og long-range (>1m). En anden måde at inddele efter, er ud fra den procedure som transponderen anvender til at sende data tilbage til readeren med. Det kan være:

- Induktiv kobling med load-modulation, hvor readerens energifelt påvirkes af transponderen.
- Backscatter kobling, som ved hjælp af reflektering svarer tilbage med bølger svarende til readerens frekvens.

2.2.1 Induktiv Kobling

Induktiv kobling, indikerer den måde som der kommunikeres mellem reader og transponder på. Denne form for kobling kræver en bestemt type transponder som består af en chip og en spole fordelt på et større område, der fungerer som transponderens antenne. Denne form for transpondere er næsten altid passive, dvs. at chippen får alt energi fra readeren. Readerens antenne generer et stærk elektromagnetisk felt som går igennem transponderens spole (antenne). Ved hjælp af induktion genereres en spænding ved transponderens antenne, hvilket sammen med en parallel kondensator fungerer som en strømforsyning til transponderens elektriske kredsløb.

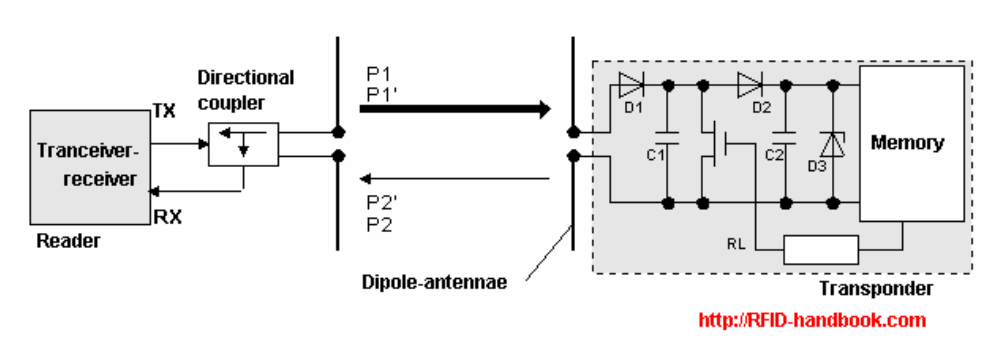


Figur 7 - Induktiv kobling mellem reader og transponder

Når sådan en transponder bringes i readerens elektriske magnetfelt, vil den svare tilbage med en resonansfrekvens, svarende til den frekvens der kommer fra readeren. Dette vil betyde at en del af den energi der er i readerens magnetfelt vil falde, hvilket readeren kan opdage som tab i spænding ved dens spole/antenne. På den måde kan man slå en belastningsmodstand til og fra i transponderens antenne og påvirke spændingen ved readerens antenne. Hvis til og frakobling af belastningsmodstanden er styret af data man ønsker at sende, kan disse data sendes fra transponder til readeren. Denne form for data transmittering kaldes også load-modulation.

2.2.2 Backscatter Kobling

Backscatter (tilbagespredning) er en anden måde at kommunikere mellem reader og transponder på. Dette kendes bl.a. for RADAR teknologien hvor elektromagnetiske bølger bliver reflekteret, når de rammer objekter større end ca. halvdelen af bølgernes bølgelængde.



Figur 8 - Backscatter Kobling mellem reader og transponder

Backscatter transpondere hører ofte til long-range systemerne som opererer i UHF frekvensområdet på 868Mhz og 915 MHz, samt i mikrobølgeområdet med frekvenser på 2,5GHz og 5,8GHz for hhv. Europa og USA. For at udnytte den energi som sendes af readeren, kræver det en anden transponder arkitektur som vist på Figur 8. Rækkevidden kan beregnes ved først at udregne *free space path loss* a_F. I beregningen indgår afstanden mellem readerens og transponderens antenne samt deres styrke. Størrelse af a_F er et mål for forholdet mellem den HF energi som går tabt i luften og den HF energi som faktisk når frem til transponderens antenne. Herefter er transponderens energiforbrug en afgørende faktor. Moderne transponder chips kan komme ned på et energiforbrug i nærheden af 5µW [1]. Beregninger [1] viser at dette giver en rækkevidde på lidt over 3 meter ved en frekvens på 868 MH, og lidt over 1 meter kan nås ved 2,45 GHz.
Ønsker man længere afstande på op til 15 meter er man nød til at have en ekstern energikilde til transponderens chip. Batteri energien anvendes udelukkende af chippen og ikke til at transmittere svaret med, den energi kommer udelukkende fra readeren.

2.3 RFID og andre Auto ID systemer

RFID sammenlignes ofte med stregkoder når man taler om anvendelsesmuligheder. Men teknisk set er det en teknologi med mange anvendelsesmuligheder i mange forskellige brancher, hvor der er behov for automatisk identificering af personer eller objekter. Sammenlignet

med andre identificerings teknologier som f.eks. stregkoder, OCR, stemme genkendelse, biometri og smartcard er der en række fordele ved at anvende RFID.

En af fordelene er at læseafstanden mellem reader og tag typisk er op til 5 meter, hvor kommunikationen sker vha. mikrobølger. Det betyder også at støv og lign ikke påvirker aflæsningen, da det ikke er nødvendigt med direkte kontakt eller frit udsyn mellem tag og reader (non line of sight).

Sammenlignet med stregkoden kan RFID indeholde større mængde af information. Stregkoden kan typisk indeholde 1-100 bytes mens RFID typisk kan indeholde 16-64k bytes. En af ulemperne ved RFID er at det ikke kan aflæses af mennesker, dvs. en beskadiget RFID tag kan være en katastrofe, mens en beskadiget stregkode som ikke kan læses af en maskine i nogle tilfælde kan læses af mennesker. Stregkoder er meget billige at producerer, og ofte en del af emballagen uden ekstra omkostning mens RFID tags er en ekstra omkostning. Enhedspriserne er efterhånden kommet ned på et niveau hvor flere større virksomheder er begyndt at overveje dens anvendelse. Eksempelvis koster et *Class 1 128 bit tag* ca. 1 DKKR [27].

Tabellen nedenfor viser en oversigt som sammenligner RFID teknologien med en række andre auto-id teknologier på en række områder.

System Parametre	Stregkode	OCR[1]	Stemme genkendelse	Biometri	Smart card	RFID
Typisk data kvantitet (bytes)	1-100	1-100	-	-	16-64k	16-64k
Data densitet	Lav	Lav	Høj	Høj	Meget høj	Meget høj
Maskine læsbarhed	God	God	Dyr	Dyr	God	God
Læsbar af mennesker	Begrænset	Simpel	Simpel	Svær	Umulig	Umulig
Indflydelse af støv/damp	Meget høj	Meget høj	-	-	Mulig (kontakt)	Ingen påvirkning[2]
Indflydelse af optisk frakobling	Total ubrugelig	Total ubrugelig	-	Mulig	-	Ingen påvirkning
Indflydelse af retning og placering	Lav	Lav	-	-	Ensrettet	Ingen påvirkning
Forringelse ved brug	Begrænset	Begrænset	-	-	Kontakt	Ingen påvirkning
Omkostning ved køb af systemer	Meget lav	Lav	Meget høj	Meget høj	Lav	Middel
Omkostning ved drift	Lav	Lav	Ingen	Ingen	Middel	Ingen
Uautoriseret kopiering/ændring	Lille	Lille	Mulig (eks. lydoptagelse)	Umuligt	Umuligt	Umuligt
Læsehastighed	Lav	Lav	Meget lav	Meget lav	Lav	Høj

[1] Optical character recognition
[2] Afhænger af RFID tag og reader, da nogle tag er meget følsomme over for væsker.

inkl. data transmission	~4s	~3s	>5s	>5-10s	~4s	~0,5s
Maksimale afstand mellem data carrier og reader	0-50cm	<1cm Scanner	0-50 cm	Direkte kontakt (fingeraftryk)	Direkte kontakt	0-5m Mikrobølger

Tabel 2- RFID sammenlignet med andre auto-id teknologier [1]

Af tabellen fremgår det at RFID er et sikkert system mod både uautoriseret kopiering eller modificering men også at det ikke påvirkes af omgivelserne så længe et tag er inden for læserens læseafstand. Dette faktum afhænger af både reader og tag, samt den sikkerhed man har valgt at implementere i systemet. I kapitel 5 ser vi nærmere på sikkerheden i RFID og navigeringssystemet.

2.4 Valgkriterier for RFID teknologi

Valget af RFID teknologi afhænger af systemets formål. De mest almindelige valgkriterier for RFID teknologien kan inddeles i følgende fire grupper:

- Frekvens
- Rækkevidde
- Sikkerhed
- Hukommelseskapacitet

Frekvensen på RFID systemer bestemmer måden de kommunikerer på. Vi så at systemer i området 100kHz - 30MHz opererer vha. induktion, hvorimod mikrobølgesystemerne i området 2,45-5,8 GHz er koblet sammen vha. et elektromagnetisk felt. Absorberingsraten (*absorbation rate)* stiger i takt med frekvensen, og er ca. 100.000 gange større ved 1GHz end 100 kHz. Dette er vigtigt når man skal operer i miljøer som kan absorbere energien, f.eks. væsker i dyr. Her anvendes lavfrekvens systemer <135 kHz som bedre kan trænge igennem absorberende materialer.
Ydermere skal man tænke på følsomheden overfor elektromagnetisk interferens. Her egner de induktive transpondere sig ikke særligt godt pga. deres følsomhed, men i disse tilfælde anvender man typisk mikrobølge systemerne.
Som vi tidligere har været inde på har frekvensen også betydning for hvilket rækkevidde man opnår.

Rækkevidden skal bestemmes ud fra det ønskede formål. Det er ikke altid optimalt med en lang rækkevidde. F.eks. er det en god ide at anvende induktiv kobling med kort rækkevidde for betalingssystemer.

Her går man ud fra at brugeren står meget tæt på readeren og måske indfører transponderen i et bestemt område i nærheden af readeren, hvilket også øger sikkerheden. I andre tilfælde ønsker man at læse/skrive en transponder på længere afstand, f.eks. containere som bevæger sig igennem en port. Man ved ikke hvor på containeren transponderen er placeret, hvilket betyder at man skal kunne læse fra reader til det punkt længst på containeren. Her anvender man typisk højfrekvens eller mikrobølge systemer da de har en længere rækkevidde. I disse situationer hvor transponder, reader eller begge er i bevægelse, er man nød til at tage højde for deres hastighed i forhold til hinanden samt den maksimale afstand. Det bestemmer den tid som readeren har til at læse/skrive en transponder, hvilket er afgørende for systemets stabilitet.

Sikkerhed er også et vigtigt valgkriterium for systemet. Kravet til sikkerheden afhænger af de data som skal kommunikeres mellem reader og transponder, samt om systemet anvendes i offentligt rum eller isoleret omgivelser. Vi vender tilbage til sikkerheden senere i rapporten.

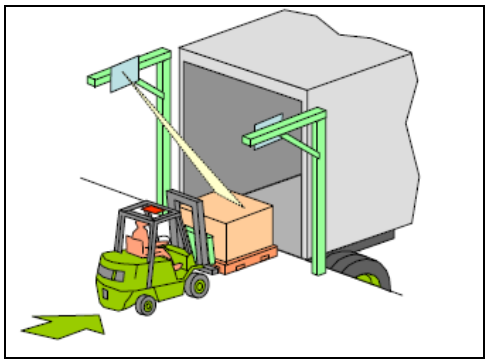
Hukommelseskapacitet er afgørende for mængden af data man ønsker en transponere skal håndterer. De forud programmerede transpondere som ikke kan omprogrammeres, indeholder typisk kun en ID. Disse kan anvendes i applikationer hvor man ikke har brug for at ændre data og prisen er afgørende. Ønsker man mere avanceret transpondere med mulighed for omprogrammering og større hukommelse skal man vælge nogle med EEPROM hukommelse, som typisk anvendes i induktive systemer og har en kapacitet mellem 16 bytes og 8 Kbytes. For mikrobølge systemerne er det typisk SRAM hukommelse som har en kapacitet på 256 bytes til 16 Kbytes.

2.5 RFID Anvendelsesscenarios

I følgende afsnit skal vi se nærmere på hvordan RFID bliver brugt i dag, for at danne et billede af anvendelsesmulighederne.

2.5.1 Port Applikationer

Mange mennesker kender RFID som stregkodens afløser og har hørt det scenarios med indkøbsvognen som man kører forbi kassen og så er alle varer identificeret og regningen står klar. Det scenario er endnu ikke implementeret på det niveau, men kun på port niveau som det ses af Figur 9. Fordelene ved sådanne applikationer er bl.a.:



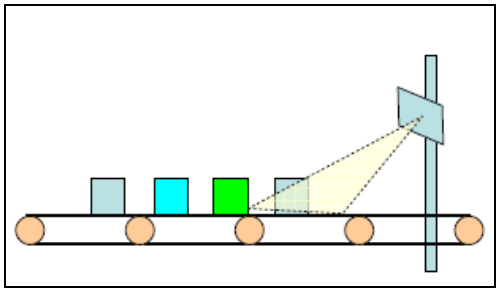
Figur 9 - RFID Port applikation

- Store porte på op til 10mx10m kan dækkes
- Automatisk modtagelse og forsendelse af varer kan registreres når de passerer RFID port
- Alarm ved forkert destination når pakker køres igennem RFID port
- Elektronisk afmærkning
- Palle/container sporing

Disse systemer kræver en læseafstand der som regel er større end en meter, her anvendes derfor typisk højfrekvens eller mikrobølge systemer.

2.5.2 Transportbånd Applikationer

RFID kan anvendes til skanning af varer på et transportbånd. Det kan være kufferter i lufthavnen, pakker på et posthus eller varer på lager eller i fabrikker.



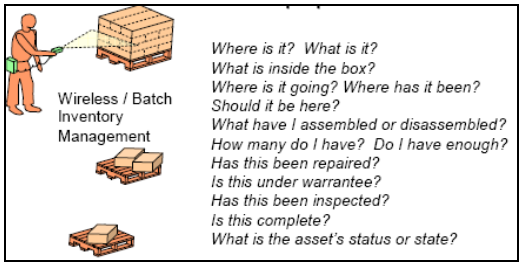
Figur 10 - RFID Transportbånd applikation

Med disse slags applikationer kan man hurtigt og nemt skanne varerne uafhængigt af deres stilling på båndet. Skanningen kan

bl.a. bruges til elektronisk registrering, afmærkning og sortering. Hvis læseafstanden er mindre end en meter kan man her anvende et induktivt systemer ellers er man nød til at anvende højfrekvens systemer.

2.5.3 Kontrol Applikationer

Ved transport, pakker man ofte varerne ned i kartonner og derefter på paller, eller i containere. Med RFID applikationer kan man trådløst identificere indholdet af en pakke, palle eller containere uden at indholdet er synligt. Ydermere kan man trække al information ud af det tag som sidder på, det kan f.eks. være holdbarhed, behandling, opbevaring, afsender, modtager osv.



Figur 11 - RFID Kontrol Applikation

RFID anvendes i dag i mange industrier og vil i fremtiden bl.a. kunne ses på pas, overvågning og som erstatning for nøgler til hus og bil. I projektet her skal vi se på anvendelsen af RFID til lokalisering af en person/objekt i en bygning og vha. en udviklet applikation navigerer rundt i store komplekse bygninger.

Til dette projekt har vi et højfrekvenssystem på 866 MHz, der ifølge producentens specifikationer kan opnå en læseafstand på op til 3 m. Reader, antenner og tags er fra *Alien Technology* med følgende specifikationer:

Reader

Name	866 MHz Reader and programmer
Model Number	ALR 8780
Operational Frequency	865.6 – 867.6 MHz (co-channel from June 2004)
Data Rate	~90 reads/sec (depends on Euro data rate)
Read Range	Meets or exceeds EPC range requirements
Radiated Power	2 watt (ERP), US equivalent is 3.28 watt (EIRP)
Anti-collision	Supported
Communications Interface	TCP/IP (RJ-45), RS-232 (DB-9 F)
Software Support	APIs, sample code, executable demo app (Alien Gateway)
Visual Indicators	Power, Serial (TX/RX), LAN (link, active), Tag (sniff, lock), RF (on)
Digital I/O	4 digital inputs, 4 digital outputs (DB-9 M)
Power Requirements	12V 5A output, 90-264 VAC input, 43-67 Hz, must be earthed
Compliance Certification	ETSI EN302-208, EN301-489, EN60950. Site licenses required in many European countries during 2004

Readerens antenne:

Model	ALR 8610-C
3 dB Beamwidth	E-plane: 69 degrees
Frequency	850MHz-875MHz
Gain (dBi)	6 dBi MAX
Polarization	Circular
RF Connector	6 meters with Reverse-Polarity TNC
Input impedance	50 Ohm –15dB return loss over full frequency range
Dimensions	11 X 8 X 1.5 inches 28 X 20.3 X 3.1 cm
Weight	27 oz 765.4 grams

Transponder:
Der anvendes Class 1 128-bit (96 user bit) NanoBlock tags. Disse tags lever op til EPC (Electronic Produc Code) åbne standard for RFID tags.

Figur 12 - ALL-9338-02

Hukommelsen er således fordelt:

	Checksum		EPC Code (or User ID Code)								Lock	PC
Byte	0	1	0	1	2	3	…	9	10	11	0	0
Bit	0-7	8-15	0-7	8-15	16-23	24-31	…	72-79	80-87	88-95	0-7	0-7

EPC koden er på 96 bit uden begrænsning for hvad der kan ligge i dette område.
Checksummen bliver beregnet og programmeret i tagget automatisk af readeren over de 96 bit EPC med CCIT-16 standarden og gemmes i de 16 første bit.
Lock og PassCode(PC) er de sidste to bytes af hukommelsen og bruges til at låse et tag med og ødelægge et tag med.

2.6 Teknologi Opsummering

Alle RFID teknologier operer efter SEQ eller FDX/HDX princippet. SEQ princippet sender kun energi i bestemte intervaller, hvilket betyder at tranponderene ikke modtager energi i readerens passive perioder. Den passive periode anvendes til at sende svaret fra tranponder til reader. De tilfælde hvor det kan være et problem, løses det ved at anvende aktive transpondere. FDX/HDX sender konstant energi ud og kræver en kommunikationsproces som f.eks. load modulation.

Frekvensen som RFID udstyret operer under er afgørende for rækkevidden og hvilket miljø det kan anvendes i. Man inddeler i tre kategorier:

- Close Coupling Systems, som har rækkevidde op til 1 cm og opererer under 3 MHz
- Remote Coupled Systems, som har rækkevidde op til 1 m og opererer under 13,56 MHz
- Long Range Systems, som opererer i UHF (868 MHz) og mikrobølge (2,5 GHz) frekvensområdet, og har en typisk rækkevidde på over en meter. En rækkevidde på 15 m kan opnås med aktive transpondere.

En transponder kan altså enten være passiv eller aktiv.

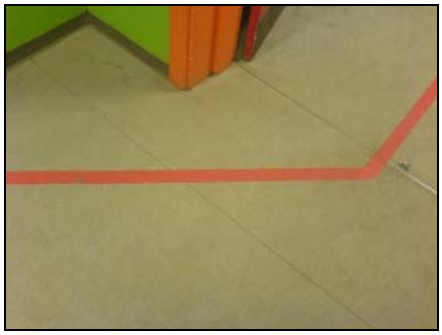
Transponderenes hukommelse er også en måde at kategorisere RFID på. Her kan man se på hukommelseskapacitet og hukommelsesteknologi, som kan være EEPROM, FRAM eller SDRAM. Teknologien er afgørende for kapacitet, strømforbrug, skrivecykluser samt afhængighed af konstant strøm for at opbevare data. Ydermere kan transponderene kommunikere med readeren vha. induktiv kobling eller backscatter kobling. Induktiv kobling er mest typisk for systemer med kort rækkevidde, mens backscatter kobling anvendes for højfrekvens systemer med længere rækkevidde.

Efter en introduktion i RFID teknologien og teknologiens anvendelses-muligheder, skal vi se nærmere på hvordan denne teknologi kan anvedes til indendørs navigering, hvilket undersøges nærmere i rapportens næste kapitel.

Kapitel 3

System Arkitektur

Navigering i større, ukendte bygninger er en kendt problemstilling, men løsningsmetoderne gør ikke brug af moderne teknik, men derimod den gode gamle skiltningsteknik. På nogle hospitaler finder man forskellige farve bånd på gulvet, som angiver at en bestemt farve fører til en bestemt afdeling. Billederne nedenfor er taget på Gentofte Amtssygehus, og illustrerer denne teknik.



Figur 13 - Rød navigations stribe fra Gentofte Amtssygehus

Figur 14 - Tre navigationsstriber som går ind i elevatoren. Gentofte Amtssygehus

Ulemperne ved denne teknik er vedligeholdelse, samt skiltning. Det er ikke så nemt at ændre en stribe hvis en afdeling flyttes eller lignende. Desuden kræver det at man ved hvor de forskellige farver fører hen, dvs. har set og forstået skiltet da man kom ind i bygningen.
En anden ulempe er at denne form for navigering ikke er så detaljeret. Hvis man ønsker at navigere fra et vilkårligt sted i bygningen til et andet vilkårligt sted i bygningen, vil denne form for navigering kræve så mange forskellige streger at det ville være en umulig opgave som vil miste sin effekt.

Vi ønsker i de efterfølgende afsnit at komme op med forskellige løsningsmodeller til denne problemstilling. Inden vi kigger nærmer på

forskellige løsningsmodeller, vil vi opstille en kravspecifikation som skal være overordnet og gældende for evt. realisering af navigationssystemet.

3.1 Kravspecifikation

Den her kravspecifikation skal danne rammerne om de forskellige løsningsmodeller som efterfølgende skal undersøges. Kravspecifikationen er også nødvendige for at bestemme hvilken løsningsmodel der bedst egner sig til en endelig løsning.

- Brugervenlighed er et krav til systemet, da brugerne ikke nødvendigvis er IT-kyndige. Til dette formål kan brugervenlighedstest fortages i designfasen, for at sikre den ønskede brugervenlighed.
- Positionen skal vises grafisk på et kort, og ikke som koordinater.
- Brugerne skal altid have mulighed for at se deres position på kortet og vælge navigation til mulige destinationer.
- Kortet skal opdateres løbende mens personen flytter sig i bygningen.
- Hver gang et RFID tag aflæses skal positionen opdateres.
- Ny rute skal beregnes hvis brugeren afviger fra den beregnede rute.
- Al kommunikation skal sikres, således at andre ikke kan aflytte eller ændre data der sendes. Det gælder både data mellem transponder og reader, og evt. netværkskommunikation mellem server og klienter.
- Systemet skal kunne beregne den korteste rute, eller anden rute efter brugerens kriterier. Eksempelvis den hurtigste rute som ikke nødvendigvis er den korteste rute.
- Mulighed for at fravælge elevatorer og trapper er en nødvendighed. Denne funktion kan man forstille sig vil blive brugt af handicappede eller personer med elevatorskræk.
- Læseafstanden mellem reader og transponder kan være op til 3 m (svarende til standard gulv-til-loft afstand)
- En transponder skal kunne aflæses ved minimum 2,5 m/s når den er længst fra readeren. Dette svarer til en hurtig bevægelse i en bygning.
- Brugeren skal operere 100 % trådløst, hvilet kræver batteristyring af de enheder der skal tilknyttes en bruger.
- Det må forventes at systemet kan have op til 2000 samtidige brugere hvilket skal kunne håndteres af systemet.
- Mulighed for at opdatere data kort og spærrede gange.

Disse krav skal være de overordnet krav til et RFID navigationssystem, og beslutningerne skal derfor tages ud fra kravenes opfyldelse.

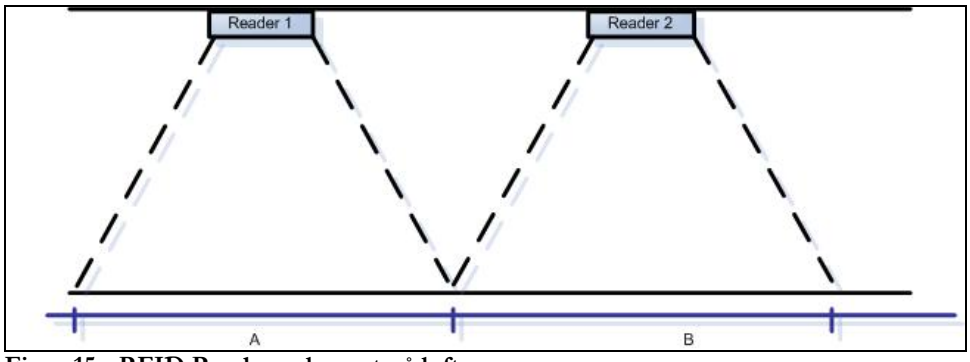
3.2 Løsningsmodeller

I følgende afsnit vil en række løsningsmodeller blive fremstillet, og fælles for alle løsningsmodeller er at de anvender RFID teknologien som et middel. Dvs. det antages at man har RFID *Reader* og *Tags* til rådighed, og så anvendes computere og evt. netværkskommunikation til at simplificere løsningen for brugere. Det nytter ikke noget at brugeren får et tal som de selv skal fortolke. Det skal være et system som kan sammenlignes med moderne GPS navigeringer, hvor en computer anvender GPS signalet til at placerer brugeren på et digitalt kort som brugeren kan forstå.

3.2.1 Tag eller Reader

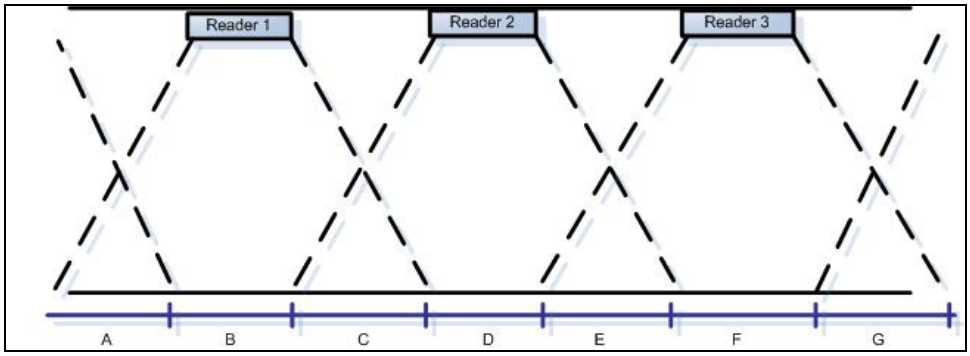
Personen eller objektet kan enten være tilknyttet en reader eller et tag. Hvis man er tag, skal der placeres readere og antenner rundt omkring, hvorimod er man en reader skal der placeres tags i bygningen.

Placeringen af readere i bygningen, minder om princippet bag GPS navigation. Her kan man placere forskellige antenner på loftet, som detekterer objekter under sig der bærer et RFID tag. Detekteringen kan sendes videre til en central server som ud fra antennens ID kan fortælle objektet hvor den befinder sig i bygningen.



Figur 15 - RFID Readere placeret på loft

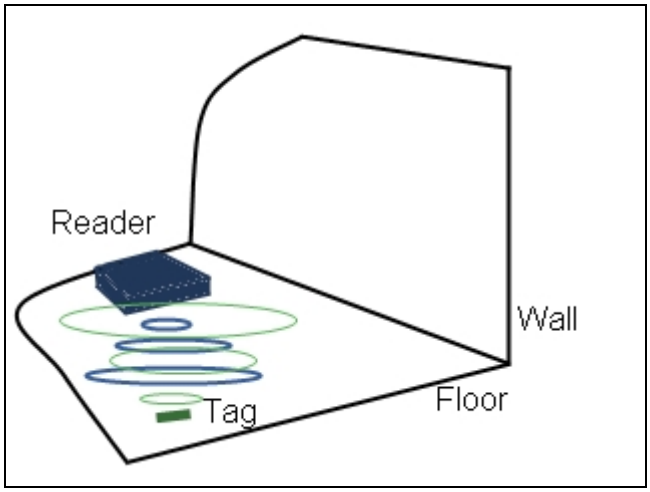
Figur 15 illustrerer to readere placeret på loftet som dækker hver deres område på gulvet. Denne løsning er mulig da RFID antennerne fra readeren kan række op til 5 meter, hvilket er større end de fleste lofthøjder. Ønsker man en mere præcis lokalisering af objekter kan man placere antennerne således at deres radiodækning overlapper hinanden.



Figur 16 - RFID Readere placeret på loft med overlap

Figur 16 illustrerer løsningen med overlap. Når et tag befinder sig i område B vil det kun være Reader1 som kan læse den, mens det i område C vil det være muligt for både Reader1 og Reader2 at læse de tags som befinder sig der. Ved at mappe de forskellige områder er det muligt at vise brugeren hvor han befinder sig ved at se hvilken antenne som kan læse det tag han er tilknyttet. Fordelen ved sådan en løsning er at man har den dyre del af RFID udstyret, readeren, monteret fast et sted utilgængeligt for uvedkommende. Ulempen er derimod at der skal bruges mange RFID readere for at dække en stor bygning, hvilket kan betyde at det ikke kan betale sig at implementere sådan et system. Set fra brugernes side er der et endnu større problem som vedrører privatlivet. Når man først er tilknyttet et tag, f.eks. på sin rullestol fra hospitalet, så er man konstant overvåget og man kan ikke selv bestemme om man skal kunne læses eller ej.

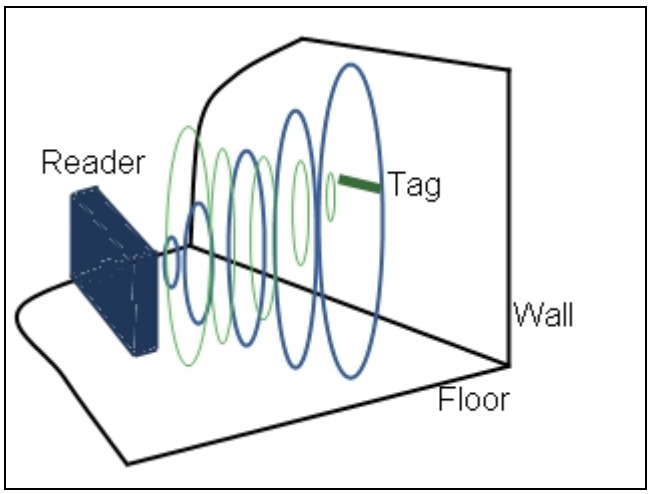
En alternativ løsning er at tilknytte den aktive del, dvs. readeren, til brugeren, og placerer tags rundt omkring i bygningen. Her kan man vælge at placere tags på gulvet, og lade en reader, der kan være monteret under en rullestol, læse disse tags. Ved at mappe tags til et kort kan man bestemme brugerens position. Figur 17 nedenfor illustrerer idéen med at have tags på gulvet som en reader kan læse.



Figur 17 - RFID Reader som læser tags placeret på et gulv

På samme måde kan man vælge at placere tags på loftet, og lade readeren pege op for at læse disse tags. Fordelen ved at have tags på loftet er at de ikke bliver beskadiget ved at folk går på dem, medmindre de er beskyttet ved f.eks. at være placeret under linoleum eller lign. Ulempen kan så være at placeringen af readeren på udstyr som rollestol kan være sværere eller direkte uhensigtsmæssig fordi radiobølgerne helst ikke skal forstyrres på vej mod loftet, af evt. absorberende materialer imellem.

Alternativt kunne man placere tags på væggene og lade readeren være placeret på siden af udstyret, som vist på Figur 18.



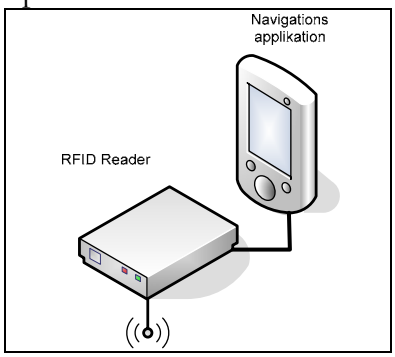
Figur 18 - RFID reader som læser tag fra side væg.

Denne løsning er enten afhængig af at man placerer tags på begge sidder af gangen eller monterer reader antenner på begge sider af udstyret, f.eks. en rullestol. Ellers risikerer man kun at kunne læse den ene vej.

Uanset hvilken af disse løsninger man vælger, så kræves der en IT løsning som kan oversætte de informationer som RFID udstyret leverer til en position. Det kræver enten at alle tags har et unikt ID, som så er mappet over i et kort over bygningen, eller også kan tagget indeholde information om den fysiske placering samt koordinaterne til et kort. Vælger man at lægge informationen i selve tagene kan man spare den del af systemet som holder styr på hvilket ID som svarer til hvilken position. Men til gengæld mister man en hel del fleksibilitet, da løsningen bliver meget statisk. Ønsker man at ændre noget, skal det enkelte tag omprogrammeres, hvilket kræver en fysisk tilstedeværelse ved selve tagget. Den anden løsning med at lade tags have en ID, som mappes et centralt sted, giver en øget fleksibilitet, men kræver at hele systemet udvikles som et distribueret system, hvilket vi vil se nærmere på. Først ser vi kort på en løsning som ikke benytter distribuerede systemer.

3.2.2 Stand-alone Applikation

Det omtalte RFID navigerings system kan implementeres som et computerprogram på det objekt som skal navigeres. Eks. kunne det være en PDA som indeholder digitale kort til beregning af ruter, som opdateres ud fra RFID readeren, som vist på Figur 19.



Figur 19 – Stand-alone løsning

Denne løsning er uafhængig af andre systemer, da den indeholder den nødvendige information selv. Dette kan være en fordel, men dynamikken i disse systemer er minimal, hvilket kan være en stor ulempe. Disse slags løsninger egner sig til applikationer som ikke ændrer sig og er uafhængige af udefrakommende information, f.eks. tekstbehandlingsprogrammer. Men hvis man ønsker muligheden for at navigerer klienterne rundt fra et centralt sted ved at give dem ruteinformation kan denne løsning ikke anvendes.
En anden stor ulempe er opdatering af data som applikationen har brug for til beregning af ruter. Hvis en gang spærres vil det være en umulig opgave at opdatere data hos de enkelte applikationer, da det vil kræve

fysisk tilstedeværelse. Det er oplagt at implementere det som et distribueret system som gør det mere fleksibelt og anvendeligt.

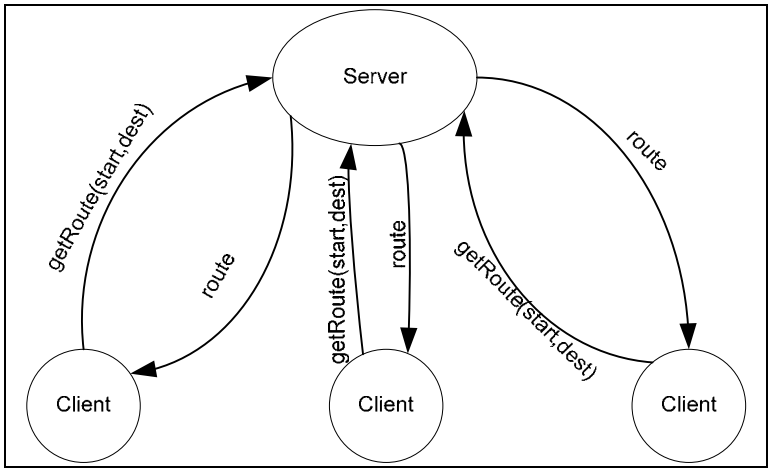
3.2.3 Distribuerede systemer

Der findes flere forskellige former for distribuerede systemer i dag og disse har visse fortrin i givne opgaver. Af disse er der nogle der er kendte for netop deres unikke måde at løse deres problem på. Fx kender mange P2P teknologien som meget effektivt kan dele data uden at have en unik server der holder referencerne eller på anden måde skal holde styr på filers placering. GRID teknologien er en anden meget kendt teknologi. Denne teknologi bruges bl.a. af SETI (Search for Extra-Terrestrial Intelligence) til deres beregninger. Disse beregninger bliver udført af folks personlige pc'er som har installeret en "SETI" screen saver der tilbyder SETI ubenyttede cpu tid. Af normale client-servere programmer som er meget vidt udbredt er msn messenger/ICQ eller mange af de andre chat systemer for bare at nævne nogle stykker. Mobile agenter er en anden måde at implementere distribuerede systemer på, her kan man lade et program vandre rundt og udfører opgaver på forskellige maskiner og måske til sidst vende tilbage til afsenderen med et resultat. Alle disse teknologier er taget i brug af mennesker fordi deres unikke egenskab viser særdeles egnet til at løse netop et specifikt problem. I de efterfølgende afsnit skal vi se nærmere på forskellige distribuerede metoder man kan anvende til RFID navigering.

3.2.4 Client-Server System

Det mest almindelige ved klient servere systemer er at flere klienter kommunikere med den samme server, som de kender på forhånd, f.eks. i form af IP adresse og port. Serveren yder en fælles service for klienterne som de har brug for men ikke selv er i besiddelse af. I RFID navigations systemet kan det f.eks. være rute information. Arbejdsfordelingen mellem klient og servere skal balanceres således at flaskehalse så vidt muligt skal undgås eller minimeres. Hvis der er tale om klienter med kraftige processorer, kan man med fordel lade klienterne udføre en del af beregningerne for ikke at overbelaste serveren mens klienterne blot venter på svar fra serveren. Ligeledes kan man lade klienterne kende til statiske data eller data som sjældent ændrer sig for at spare på datastrømmen mellem serveren og klienterne og på den måde ikke overbelaste netværket.

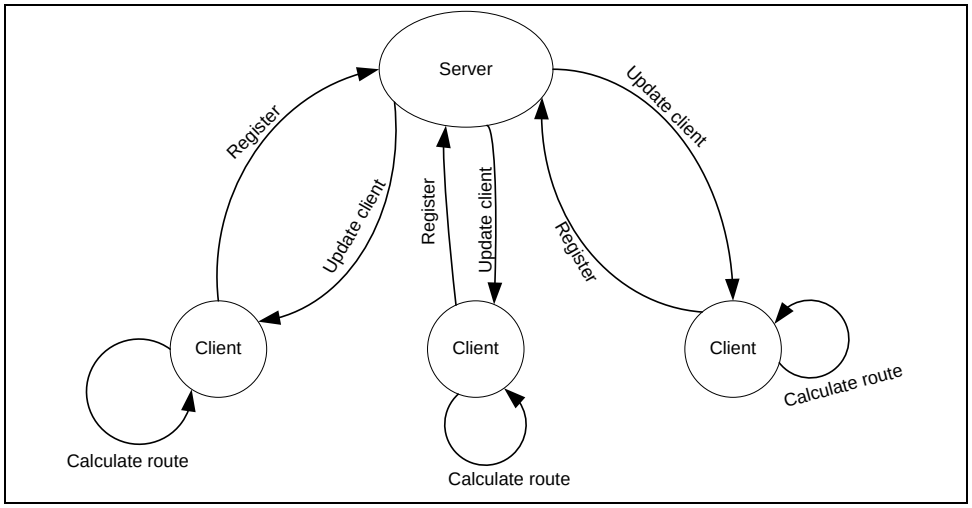
RFID navigations systemet kan implementeres med en tynd klient og en kraftig server som vist nedenfor på Figur 20.



Figur 20 – Client-Server løsningsmodel 1

En tynd klient kan bestå af en grafisk brugerflade samt noget simpel logik til kommunikation med serveren. Klienten spørg serveren om ruteinformation og viser resultatet til brugeren. Denne løsning kræver en kraftig server, da den skal fortage alle beregninger, samt konstant kommunikation mellem serveren og klienterne. En af fordelene ved denne løsning er at ændringerne i det digitale kort kun behøver at finde sted på serveren. Eksempelvis vil en blokering eller anden ændring i bygningen medføre at alle klienterne anvender denne information øjeblikkeligt hvis blot serveren kender til den.

En anden løsningsmodel for et client-server system er illustreret af Figur 21. Denne løsningsmodel anvender en mere simpel server og større klienter. Klienterne minder mere om den beskrevet stand-alone løsning, med den forskel at serveren holder klienterne opdateret når der sker ændringer, samt at klienterne kan hente konfigurationen fra serveren.



Figur 21 - Client-Server løsningsmodel 2

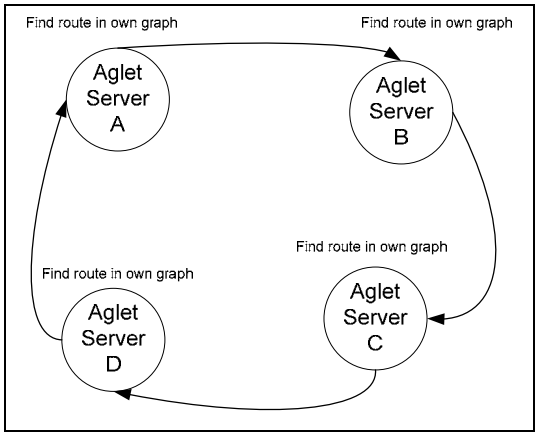
Denne løsning er mere decentraliseret fordi klienterne kan operere uafhængigt af serveren. Dette kan være en fordel hvis serveren går ned, så kan klienterne fortsætte da de selv beregner ruterne ud fra de seneste informationer fra serveren før den gik ned. I værste fald kan klienterne beregne en rute som er blokeret, men det er ikke værre end en ny alternativ rute kan beregnes når blokeringen opdages. Denne løsning anvender også klienternes processorkraft og behøver ikke en kraftig server, hvilket også kan være en fordel rent økonomisk da klienterne råder over tilstrækkelig processorkraft selv med de billigste processorer i dag.
Ønskes en endnu mere decentraliseret løsning kan man f.eks. anvende mobile agenter eller P2P systemer, som vi vil se nærmere på i de efterfølgende afsnit.

3.2.5 Mobile agenter

Mobile agenter har sin brugbarhed ved forskellige løsninger af kode som endnu ikke er skrevet. Dvs. når servicen tilbydes er der ikke lavet en specifik agent der skal køre på netværket, men denne kan blive til senere og stadig køre uafhængigt af opsætningen på de forskellige klienter.
Det er en god ting hvis man skal indsamle oplysninger eller viderebringe status på et netværk. Her er det muligt at lave en agent der besøger alle de mulige klienter hvor servicen er tilbudt og fortæller hvad status er på andre og selv finder status på den pågældende klient, for tilslut at tage videre i netværket for at fortælle andre den opsamlede information.

Et netværk af klienter som kommunikerer vha. mobile agenter er ikke helt decentraliseret fordi man er afhængig af at kende adressen på de andre klienter som den mobile agent skal vandre til. Dette problem kan løses vha. en eller flere meget simple server som blot lade en klient registrere sig når den er aktiv. Når en mobil agent skal vandre rundt til de andre klienter kan den hente en liste af klienter som den kan bruge.

Man kan lade de enkelte klienter opbygge hver deres egen graf mens de vandrer rundt. Dvs. hver gang en klient læser et RFID tag så er den ved en knude, og denne knude må være forbundet med den næste knude som den læser. Har man et stort antal klienter som flytter sig rundt i bygningen over en periode vil hver klient opbygge sin egen graf, som med tiden vil ligne hinanden. Når en klient ønsker at finde en optimal rute mellem to punkter kan den lade en mobil agent vandre rundt og lade hver enkelt klient beregne den optimale rute ud fra egen graf, og til sidst returnere et sæt af ruter hvor tidspunktet for grafen kan indgå. Denne idé er illustreret af Figur 22 nedenfor.



Figur 22 - RFID navigering med mobile agenter

Det er så op til brugeren at vælge en rute, her kan tiden anvendes til at sikre sig at ruten stadig er åben. Man vil måske foretrække en rute som er kørt for en time siden men er lidt længere end en rute som sidst blev kørt for 2 uger siden. De mobile agenter kan også anvendes til at vandre rundt og opdatere grafen (det digitale kort) hos de andre klienter, hvorefter klienterne løser opgaven som beskrevet under stand-alone applikationen.

Fordelen ved denne løsning er det digitale kort løbende holdes opdateret uden at en administrator skal opdatere en server. Men klienterne er stadig afhængig af en simpel server, og man kan ikke være sikker på at klienterne har været rundt i alle dele af bygningen. Hvis dette er tilfældet har man ikke en komplet graf, og man kan ikke være sikker på at de ruter som beregnes er de mest optimale ruter.

Ønsker man en fuldstændig decentraliseret løsning, som ikke afhænger af nogen server kan man anvende P2P teknologien som beskrevet i næste afsnit.

3.2.6 Peer-to-Peer System

P2P systemerne er mest kendt inden for fil delingstjenester fordi den giver brugerne mulighed for at dele filer med andre som man ikke kender uafhængigt af nogen server. Teknologien giver mulighed for at nå ud til mange brugere i større netværk, også Internettet, samt mulighed for at splitte den samme fil op og hente de forskellige dele fra forskellige brugere og sammensætte filen når de enkelte dele er hentet færdig.

P2P systemet kan anvendes på samme måde som beskrevet i afsnittet for mobile agenter, med den fordel at systemet kan være 100 %

decentraliseret. De enkelte klienter kan opbygge deres egen graf gennem tiden, og dele den med andre klienter som er nye i netværket. Ud over grafen kan beregnede ruter også deles, ved f.eks. at søge på ruter mellem to punkter hos de andre klienter.

Den største styrke ved dette system er decentraliseringen som sikrer at systemet altid er kørende. Men ulempen er den tvivlsomme kvalitet af de beregningerne som fortages, dvs. det er tvivlsomt om man får den mest optimale rute. Ydermere er man afhængig af andre kørende klienter, således at man har et brugbart netværk.

3.3 Fordele og Ulemper for løsningsmodellerne

De forskellige løsningsmodeller har både fordele og ulemper, hvilket også fremgår af tabellen nedenfor.

Løsningsmodel	Fordele	Ulemper
Stand-alone application	• Påvirkes ikke af andre klienters tilstand • Påvirker ikke andre klienter • Uafhængig af nogen server • Ingen netværks-afhængighed	• Svær at holde opdateret • Risiko for at beregne en blokeret rute • Meget statisk
Client-Server (Tynd klient)	• Simpel klient kræver ingen kraftig maskine • Alle klienter opdateret samtidigt • Simpel opdatering af ruter • Mulighed for beregning af optimale ruter	• Kræver en kraftig server som skal håndtere alle beregninger for klienterne • Belastning af server • Afhængig af netværk kommunikation • 100 % afhængig af server
Client-Server (Kraftig klient)	• Mere simpel server • Udnytter klienternes regnekraft og spare på serverberegninger • Nemt at opdatere klienterne • Beregner optimale ruter ud fra server-opdateringer • Kan kører uden server	• Kraftige klienter • Netværksafhængig
Mobile agenter	• Kræver kun en simpel adresse server • Intelligente klienter som selv lære ruter • Automatisk opdatering af digitale kort	• Kan ikke fungere uden en form for adresseserver • Afhængig af rute oplæring • Ingen sikkerhed for optimal rute. • Kræver kraftige klienter • Svær at ændre opsætningen

		• Kan tage lang tid før en agent kommer til den klient med de nødvendige data
Peer-to-Peer	• 100 % decentraliseret • Automatisk opdatering af digitale kort	• Ingen sikkerhed for optimale rute • Afhængig af ruteoplæring • Afhængig af antallet af aktive klienter • Svær at ændre opsætningen • Problematisk at få fat i den peer som sidder inde med ønskede information.

RFID navigationssystemet skal være sikkert og stabilt. Det betyder bl.a. at systemet skal være af høj standard således at man altid får den mest optimale rute også når der er blokerede gange eller en defekt elevator. Den mest stabile løsning opnås med en client-server løsning hvor klienterne beregner ruten ud fra en algoritme som en central server uddeler. De digitale kort over bygningen skal også uddeles af en central server, som også holder klienterne opdateret for blokeret passager. Som vi så kan denne løsning også fungere når serveren er nede, hvilket kan ske i korte perioder. De løsningsmodeller som vi hidtil har set på har udelukkende fokuseret på det samlet system og ikke på RFID udstyret som skal anvendes. RFID udstyret har også en række begrænsninger som man skal tage højde for i designet af systemet.

Efter en introduktion i RFID teknologien samt hvordan teknologien kan anvendes til at løse problemet med indendørs navigering, vil vi i næste afsnit sænke abstraktionsniveauet lidt og se nærmere på teorien omkring rutebestemmelser i navigationssystemer.

Kapitel 4

Ruteberegning

I følgende afsnit ser vi nærmere på hvordan man kan repræsentere gangene i en bygning som et digitalt kort, samt på forskellige algoritmer til beregning af optimale ruter.

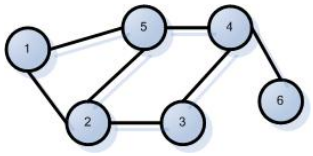
4.1 Repræsentation af kort

Et kort kan læses af mennesker, men for en computer er man nød til at lave en repræsentation som den kan forstå og arbejde med. Når man anvende *pathfinding* algoritmer, så arbejdes der med grafer i matematisk forstand. En graf består af en række knuder (*vertices*) og som er indbyrdes forbundet af linier/kanter (*edges)*.

En graf (*undirected graph*) er et par $G = (V, E)$ af sæt $E \subseteq [V]^2$ således et element af E er 2 elementer V. For V og E gælder altså:

- Elementerne af V (*vertices*) er knuderne i grafen G
- Elementerne af E er en mængde af uordnet par af knuder i V, som angiver linjerne der forbinder knuderne, kaldes kanter (*edges*).

Definitionen til en simpel graf som denne tillader ikke kanter fra en knude til sig selv (*loops)* eller dobbeltkanter, dvs. flere end én forbindelse mellem knuderne. Med andre ord er der ikke tale om en **pseudograf**[23]. To knuder x,y af grafen G siges altså at være hosliggende, eller naboer, hvis grafen indeholder en linje xy.



V = { 1, 2, 3, 4, 5, 6 }
E = { {1,2}, {1,5}, {2,3}, {2,5}, {3,4}, {4,5}, {4,6} }

Figur 23 – Simpel graf

En graf kan antage forskellige former [15]:

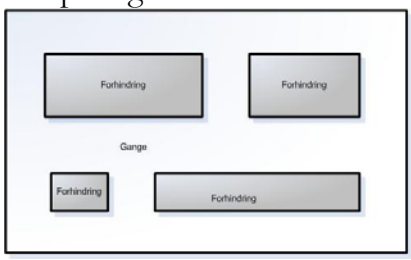
- Komplet, dvs. der er kanter mellem alle knuder $E=V^{(2)}$
- Sammenhængende, dvs. der findes en sti mellem alle knuder

- Todelt, hvis kanterne E kan deles op i to disjunkte mængder $E = X \bigcup Y, X \bigcap Y = \emptyset$, så alle kanter går mellem de to dele af grafen, $e \not\subset X$ og $e \not\subset Y$ for alle $e \in E$.
- En plangraf, hvis den kan indlejres i planet (tegnes på et stykke papir), så ingen kanter krydser hinanden,
- En skov, hvis der ikke findes cykler i grafen, der går igennem flere end 2 knuder,
- Et træ, hvis det er en sammenhængende skov.

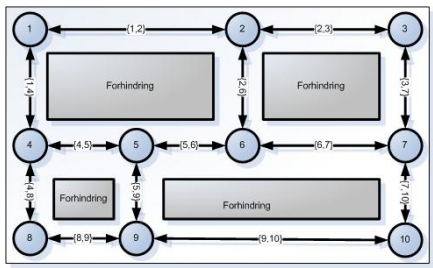
Hvis kanterne har en retning, kaldes de for pile (*Arcs)*, så er der tale om en orienteret graf eller digraf[3] $G=\{V,A\}$. Det betyder at kanterne nu har en retning så $e=\{X,Y\}$ går fra X til Y, man kan sige at Y er hovedet og X er halen.

I projektet her ønsker vi at finde rundt i en bygning, hvilket skal vises som en sti på et kort for brugeren. Ser vi kun på et 2D kort kan man lade en graf repræsentere mulige veje på kortet således at en knude repræsenterer et sted hvor vejen skilles/mødes (kryds) mens en pil angiver en gang eller et sted man kan bevæge sig på for at bevæge sig til næste knude. Anvender man en orienteret graf har man samtidigt mulighed for at angive ensretning, hvilket kan være hensigtsmæssigt ved gange hvor man kan åbne en dør fra den ene side men ikke fra den anden, eks. udgang tilladt for alle men indgang kræver nøgle eller lignende.

Figur 24 viser en skitse af en meget simpel bygning, hvor det hvide angiver områder man kan bevæge sig på (gange), mens det mørke er forhindringer (vægge). Sådan et kort kan repræsenteres af en graf som vist på Figur 25.



Figur 24 - 2D kort med forhindringer og gange



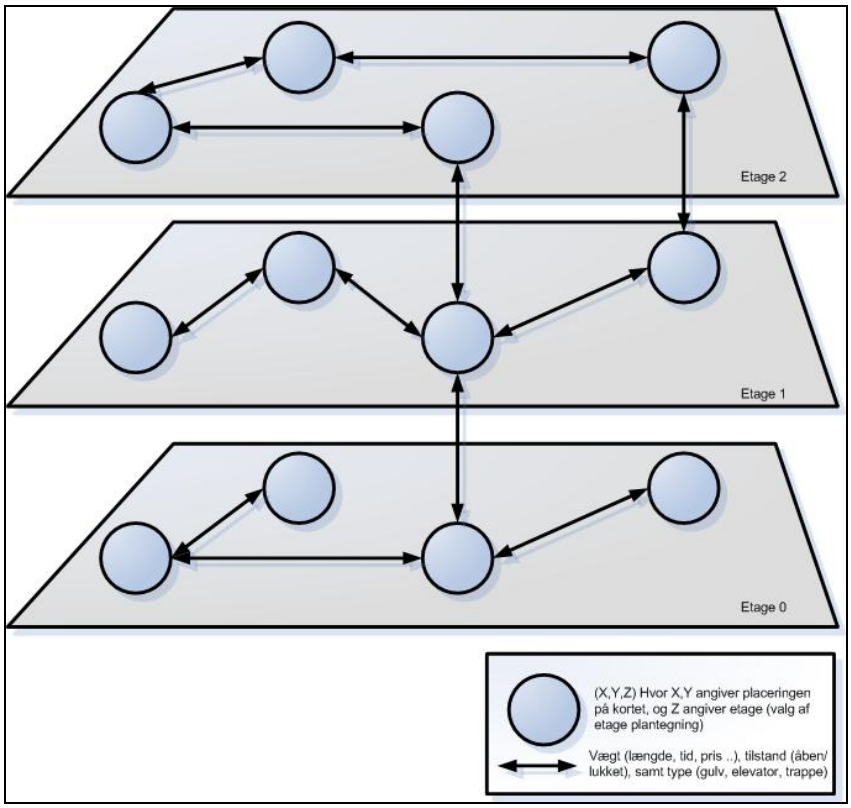
Figur 25 - 2D kort repræsenteret af en graf

En rute eller en sti beregnes ud fra grafen som repræsenterer kortet, og angiver hvordan man kommer fra et punkt på grafen til et andet punkt. I

[3] Kommer af engelsk directed graph

næste afsnit ser vi nærmere på forskellige algoritmer til at beregning af optimale ruter.

I en datamodel kan en knude være et 2D punkt $k_1 = (x_1,y_1)$ mens en pil kan repræsenteres som $p_1 = (k_1, k_2)$ hvor $k_2 = (x_2,y_2)$, men vi har brug for 3D kort da en bygning kan have flere etager. En måde man kan løse dette problem er ved at tilføre en tredje koordinat til knuderne således at de nu hedder $k_1 = (x_1,y_1,z_1)$ hvor z koordinaten angiver hvilken etager der er tale om. Datamodellen for en pil skal også udvides så den har en tilstand åben/lukket således at man kan spærre en vej. Idéen er illustreret af figuren nedenfor:

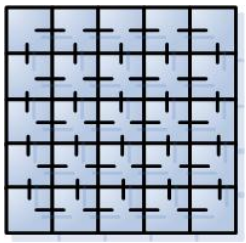


Figur 26 - 3D bygning repræsenteret vha. en graf

Fordele ved at repræsentere et kort vha. en graf er at den kan være dynamisk. Man kan flytte rundt på knuderne, oprette nye og slette knuder, grafen skal dog stadig være komplet.

I efterfølgende afsnit arbejder vi med 2D kort, da navigering i 3D bygninger mellem etager i vores tilfælde er at navigerer til trappe eller elevator på et 2D kort (aktuelle etage), og derefter navigere fra trappe eller elevator til målet på et nyt 2D kort.

Afstanden på et kort kan være nemmere at illustrere end afstanden mellem to punkter på en graf. I efterfølgende afsnit vil vi derfor se på 2D kort inddelt i celler, som vist på Figur 27.



Figur 27 - 2D kort i celler (også en slags graf)

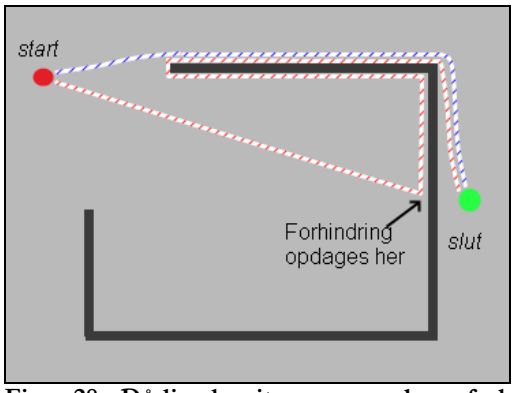
En celle kan opfattes som en knude fra en graf, mens væggen mellem to celler kan være pilen som angiver forbindelsen mellem to knuder. Da den type graf vi arbejder er en plangraf[4], kan vi godt repræsentere det som Figur 27.

4.2 Algoritmer

Som udgangspunkt for at finde den mest optimale[5] rute skal man kende den aktuelle position, samt destinationen. Når disse to punkter kendes, og der eksisterer et kort over området, som f.eks. kan være repræsenteret vha. en graf som vi så i forrige afsnit, kan man anvende forskellige algoritmer til at finde den mest optimale rute. En god algoritme adskiller sig bl.a. fra en dårlig ved at være hurtigere og mere præcis. I rute (pathfinding) algoritmer er forudsigelsen af evt. forhindringer en afgørende faktor for algoritmens hastighed.

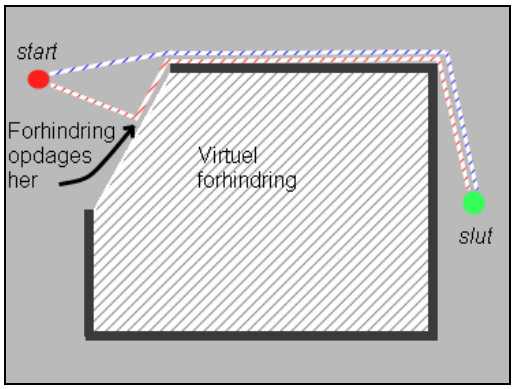
Hvis blot man bevæger sig mod målet og takler forhindringerne som de kommer hen ad vejen, så kan ruten kom til at se ud som den rød-stiplet sti på Figur 28, mens den blå-stiplet sti viser den korteste rute.

[4] Vi har en plangraf fordi gange kan ikke krydse hinanden uden at mødes inden for samme etage.
[5] Optimal rute kan være korteste, hurtigste, billigste osv. Afhængig af hvad man ønsker



Figur 28 - Dårlig algoritme som opdager forhindringen meget sent

En mulighed er at lægge information i kortet, dvs. oprette virtuelle forhindringer, således at man kun går ind i "hullerne" hvis målet befinder sig der, ellers er det en forhindring fra starten. Dette er illustreret ved Figur 29 nedenfor.



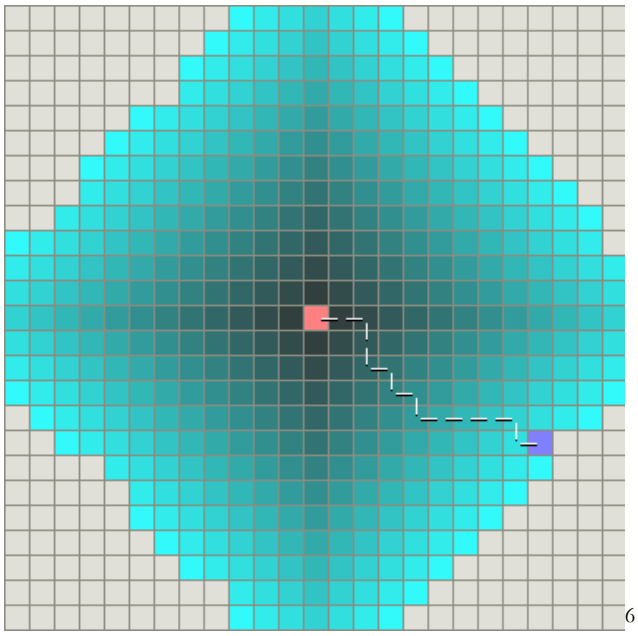
Figur 29 - Virtuelle forhindringer for at optimere ruten

I de efterfølgende afsnit skal vi se nærmere på rute algoritmer. Algoritmerne kan anvendes til at finde den optimale rute fra et punkt til et andet. Ruten skal således planlægges før den anvendes, i stedet for at flytte sig mod målet og håndtere forhindringerne som de kommer med tiden.

4.2.1 Dijkstra's Shortest Path Algoritme

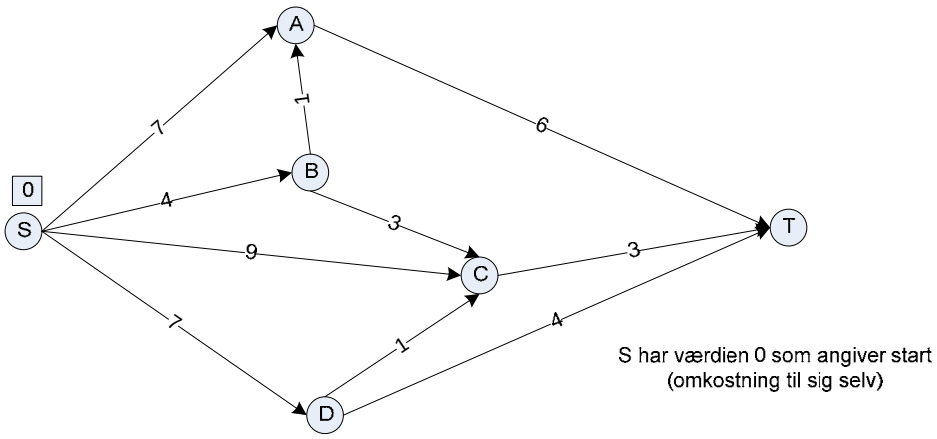
Dijkstra's algoritme er en kendt rute algoritme, som også bruges til netværksruter. Algoritmen fungerer ved at undersøge en kant ad gangen i grafen ved at starte fra et udgangspunkt. Processen gentager sig med at undersøge kanter som endnu ikke er undersøgt, og tilføjer deres kanter til en liste af kanter som skal undersøges. Algoritmen breder sig ud af fra startpunktet som vist Figur 30 indtil målet er nået ved hele tiden at undersøge den kant som er tættest startpunktet. På figuren er det lyserøde felt startpunktet, mens de mørkeblå felter er de felter som er

tættest mens de bliver lysere jo længere væk de befinder sig. Dijkstra's algoritme sikrer at man finder en af de korteste ruter, hvis alle grafens pile har en positiv vægt [14].



Figur 30 - Søgning med Dijkstra's algoritme

Følgende eksempel demonstrere hvordan algoritmen virker, på en graf som vist på Figur 31 nedenfor.

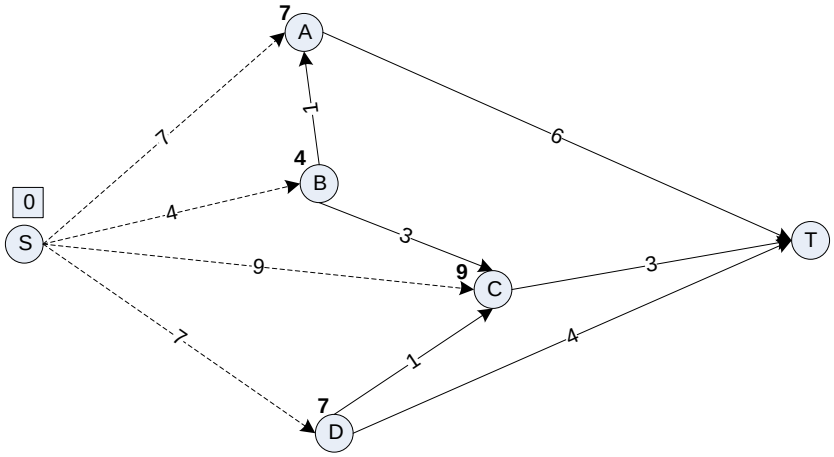


Figur 31 - En graf til demonstration af shortest path algoritmen

I eksemplet ønsker vi at finde den korteste rute fra S til T. I grafen er afstanden/omkostningen mellem to knuder angivet som et tal på pilen der forbinder knuderne. I den firkantet boks angives den hidtil korteste rute fra S, hvilket er den knude som der arbejdes videre med.

[6] Figurene 30 og 37-44 er anvendt med tilladelse fra Amit Patel [14].

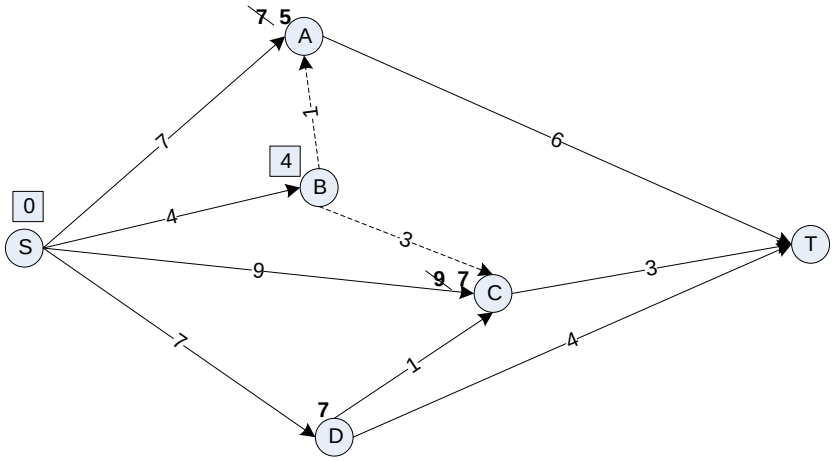
Knuderne som direkte kan nås fra S. Som næste skridt analyseres de knuder som kan nås direkte fra knudes S, dvs. der er tale om knuderne A, B, C og D. Hver af disse knuder markeres med en omkostning fra startpunktet, i det her tilfælde svarende til pilens længde fra S. Dette er illustreret af Figur 32 nedenfor.



Figur 32 - Direkte forbindelser fra S

Af grafen ses det at den korteste rute blandt SA, SB, SC og SD er SB med en omkostning på 4, hvilket markeres på grafen som den korteste rute mellem S og B. Vi ved det er den korteste rute da alle andre ruter skal gå igennem A, C eller D hvilket allerede fra første trin er længere end 4.

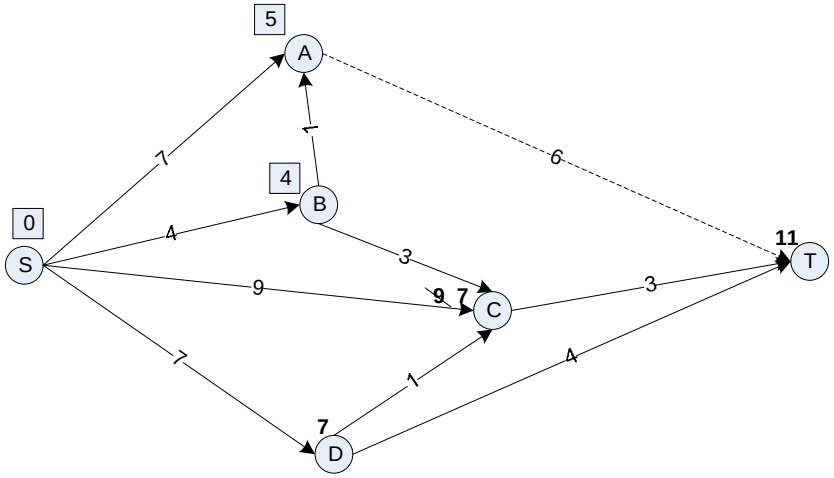
Knuder som direkte kan nås fra B. Da den korteste rute hidtil fra S er til B, vil knuderne som direkte kan nås fra B blive undersøgt som det næste. Her er tale om knuderne A og C. Hvis man går igennem B kan A nås med en omkostning på 5, mens C kan nås med en omkostning på 7, som vist på Figur 33 nedenfor.



Figur 33 - Direkte knuder fra B

Da omkostningerne til A og C er mindre igennem B end de eksisterende vil vi erstatte deres omkostninger med de nye som er 5 til A og 7 til C. Den næste knude med den korteste afstand er A, som bliver undersøgt næste gang.

Knuder som direkte kan nås fra A. T er den eneste knude som kan nås fra A, hvilet har en omkostning på 6, dvs. en omkostning fra S på 5+6=11, som vist på Figur 34 nedenfor.

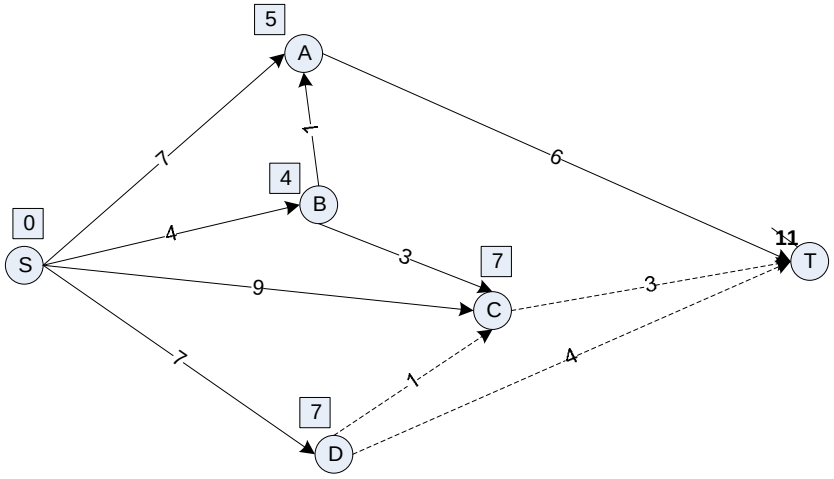


Figur 34 - Knuder som direkte kan nås fra A

Der er nu to knuder med den mindste omkostning, nemlig knuderne C og D som begge har 7. Hvilket gør at vi skal undersøge hvilke knuder som kan nås fra C og D.

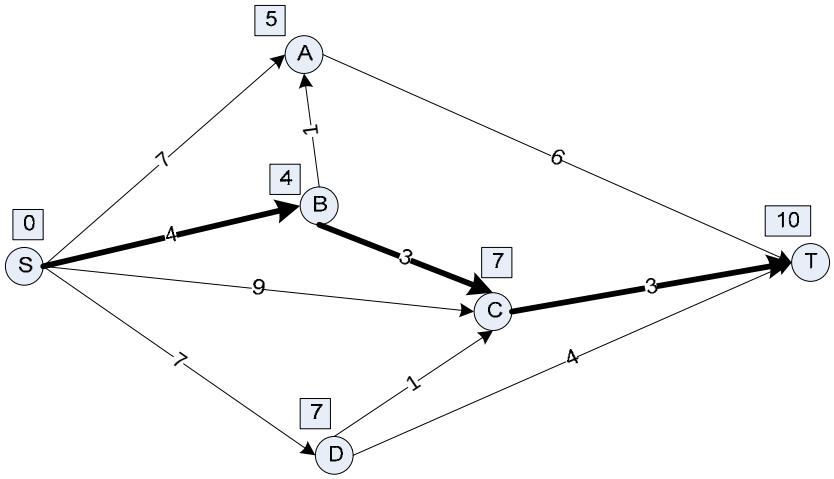
Knuder som direkte kan nås fra C eller D. Fra D kan vi nå knuderne C og T. C kan nås med en omkostning på 7+1=8 hvilket er mere end den eksisterende omkostning via B på 7. T kan nås med en omkostning

på 7+5=11, hvilket er den samme omkostning som den allerede eksisterende fra A.
Knuden C kan nå T med en omkostning på 7+3=10, hvilket er den korteste omkostning hidtil.



Figur 35 - Knuder som direkte kan nås fra C og D

Vi ved at den korteste rute fra S til T er på 10, og ruten kan bestemmes ved tilbagesporing (*backtracking*).

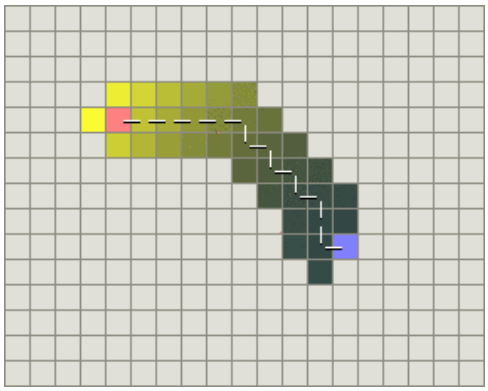


Figur 36 - Tilbagesporing af korteste rute

Tilbagesporingen starter fra knuden T og man undersøger hvor T fik den korteste værdi fra, hvilket er knuden C. C fik den korteste værdi fra knuden B, mens B kom direkte fra S. Tilbagesporingen giver os derfor den korteste rute som er S → B → C → T med en samlet omkostning på 10 [6].

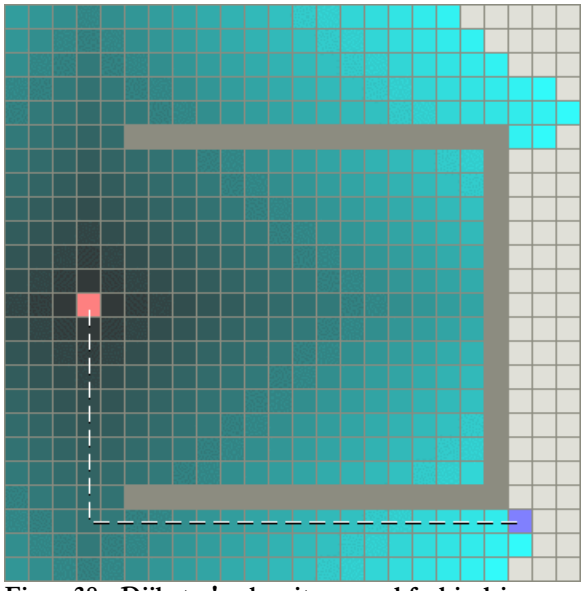
4.2.2 Best-First-Search (BFS) Algoritmen

BFS algoritmen fungerer efter samme princip som Dijkstra's shortest path algoritme men med indbygget heuristik som kan estimere hvor langt fra målet en knude er. I modsætning til Dijkstra's algoritme så undersøges ikke den knude tættest startpunktet men hele tiden den knude tættest målet. Princippet med BFS søgningen er illustreret i Figur 37, som har det lyserøde felt som start og det blå felt som mål. De felter som er mest gule er længst fra målet, mens de felter som er mørkest er tættest på målet af de undersøgte felter.

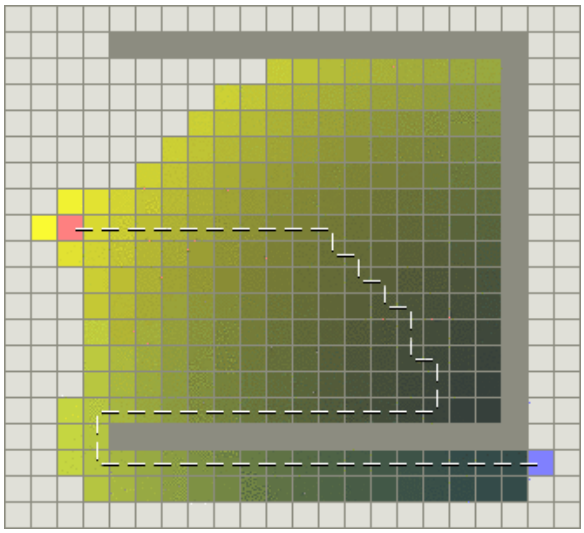


Figur 37 - Søgning med BFS

Som det ses af Figur 30 og Figur 37 så løber BFS meget hurtigere end Dijkstra's algoritme fordi den ledes i retning mod målet af heuristik, men til gengæld så finder den ikke med sikkerhed den korteste rute.
I de to tilfælde vi har set på har der ikke været nogen forhindringer, og begge algoritmer har derfor fundet den korteste rute, men BFS algoritmen har været mest effektiv hvad tid angår. Hvis der er forhindringer kan situationen se helt anderledes ud.



Figur 38 - Dijkstra's algoritme med forhindring



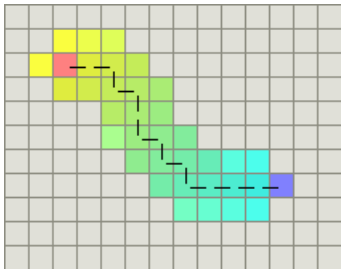
Figur 39 - BFS algoritmen med forhindring

Som vist på Figur 38 og Figur 39 kan BFS algoritmen give en rute som langt fra er den korteste rute pga. den heuristik som hele tiden fokuserer mod målet, mens Dijkstra's systematiske søgning uden heuristik sikrer den korteste ruter.
I næste afsnit skal vi se nærmere på en algoritme som både er effektiv og *kan* sikrer den korteste rute.

4.2.3 A* Algoritmen

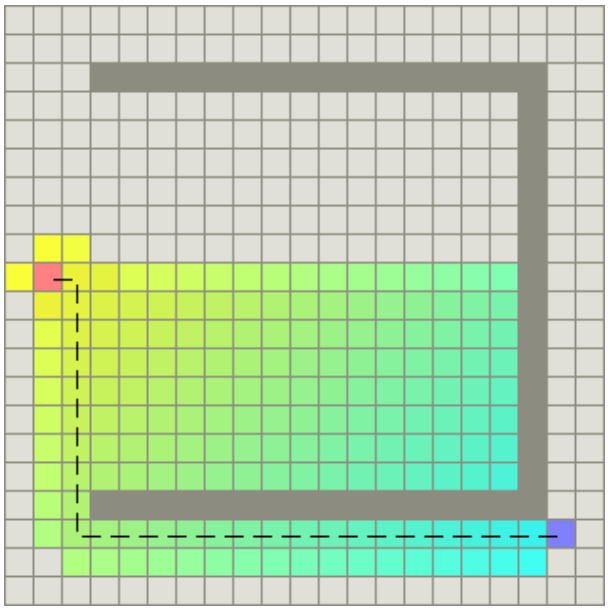
A* algoritmen er udviklet i 1968 og bruges ofte i forbindelse med kunstig intelligens programmering, f.eks. i spil hvor computeren selv styrer objekterne rundt på en bane. A* algoritmen er populær fordi den kan

sikrer den korteste rute samtidig med at den indeholder heuristik som guider den mod målet og gør den hurtig.
I det simple tilfælde uden forhindringer er A* lige så effektiv som BSF algoritmen, hvilket er illustreret af Figur 40 nedenfor.



Figur 40 - A* Algoritmen i det simple tilfælde

I et mere kompliceret tilfælde med forhindringer, finder A* algoritmen den samme rute som Dijkstra's algoritme, dvs. den korteste rute men mere effektivt. Dette er illustreret af Figur 41 nedenfor.



Figur 41 - A* algoritmen ved forhindring

Hemmeligheden i denne algoritme er at den kombinere det gode ved de to første algoritmer, dvs. den favorisere de knuder som er tættest på start knuden og kombinere det med favorisering af knuder som estimeres til at være tættest på målet vha. heuristik. I A* algoritmen har disse værdier følgende betegnelser:

- ***g(n)*** – omkostningen fra startknude til en knude *n*
- ***h(n)*** – heuristik som er estimeret omkostning fra en knude *n* til målet

A* algoritmen balancerer disse to værdi mens den bevæger sig mod målet ved at vælge den knude med laveste omkostning ***f(n)*** ved at beregne:

$$f(n) = g(n) + h(n)$$

Den første faktor i additionen kan beregnes eksakt mens den anden er et estimat bestemt vha. heuristik, og denne værdi kan være god eller dårlig afhængig af hvor god en heuristik funktion man anvender. Heuristik funktionen kontrollerer hvordan algoritmen opfører sig, og man kan styre den på følgende måde:

- Sættes ***h(n)*** til nul vil funktionen kun have ***g(n)*** og dermed opfører sig som Dijkstra's algoritme uden heuristik.
- Hvis ***h(n)*** altid er mindre eller lig med den faktiske omkostning fra en knude *n* til målet så er man sikret at algoritmen finder den korteste rute. Jo tættere på nul ***h(n)*** er desto mere spreder algoritmen sig og dermed bliver langsommere. I tilfælde hvor heuristikfunktionen underestimere kalder man A* algoritmen for simpel A algoritme.
- Hvis ***h(n)*** er den eksakte omkostning ved at komme fra en knude *n* til målet så kan man altid beregne den bedste rute uden at algoritmen vil sprede sig ud mere end nødvendigt. Problemet er at man ikke altid kan have en perfekt heuristik funktion, men algoritmen sikrer at være optimal i de specielle tilfælde hvor heuristikken er perfekt.
- I de tilfælde hvor heuristikfunktionen ***h(n)*** er større end omkostningen fra en knude *n* til målet så kan man ikke være sikker på at algoritmen finder den korteste rute men til gengæld er den hurtig.
- I den sidste ekstremme situation er tilfældet hvor heuristikfunktionen ***g(n)*** er nul eller meget lille i forhold til heuristik funktionen således at ***f(n)*** kun styres af ***h(n)*** så vil A* algoritmen opfører sig udelukkende ud fra heuristik som vi så ved BFS algoritmen [14].

Algoritmen er meget fleksibel da man selv kan balancere mellem hastighed og præcision ved at stille på ***h(n)*** og ***g(n)***.

4.3 Måleparametre og Heuristik

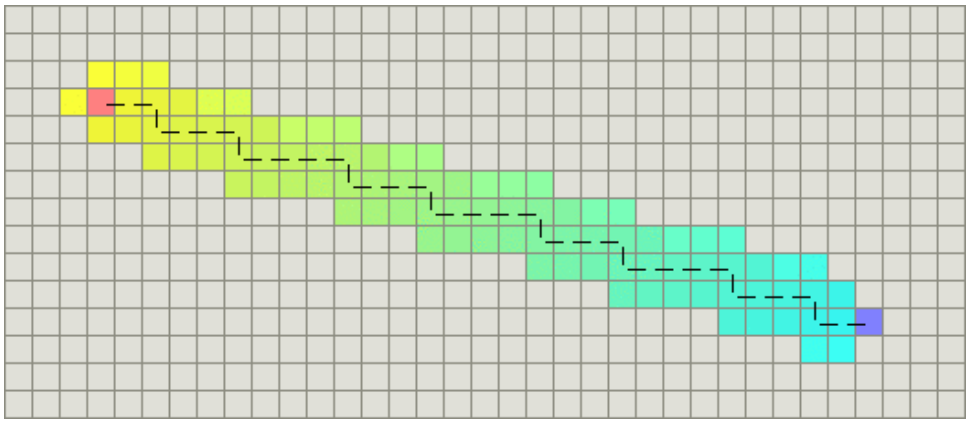
Den korteste vej er ikke altid den mest optimale vej. Eksempelvis vil en person med gangbesvær måske foretrække en længere rute som ikke involverer trapper frem for en kort rute med trapper. I navigeringsalgoritmer vægter man linkene mellem knuderne således at den optimale rute kan beregnes. F.eks. kan et link mellem to forskellige etager som repræsenterer en trappe, vægtes højere end et link for to knuder på gulvet. Her tager man ikke kun længen i betragtning når ruten beregnes men også tiden. Omkostningen for en rute skal vælges så den passer til det man ønsker at måle på hvad enten det er pris, længde, tid, trafik eller lignende.

I navigationssystemet som vi arbejder på er afstanden en oplagt faktor at arbejde med. Ydermere skal elevatorer og trapper være en valgmulighed for brugerens og kun anvendes når det er nødvendigt med etageskift. Når algoritmerne anvendes på en graf skal man beregne omkostningen ud fra en fælles måleenhed. For A* algoritmen findes der en række kendte heuristik funktioner som kort vil blive gennemgået her:

Manhattan distance

Ser man på afstanden som summen af afstanden i x-retningen og afstanden i y-retningen.

```
H(n) = D * (abs(n.x-goal.x) + abs(n.y-goal.y))
```



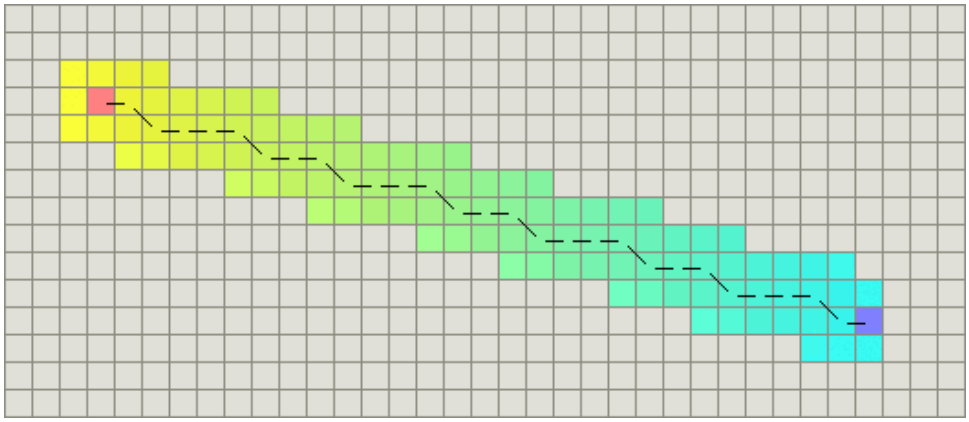
Figur 42 - Udbredelse af A* med manhatten distance heuristik

Diagonal distance

Nogle gange kan diagonale bevægelser være korter. Eksempelvis kan Manhatten funktionen resultere at en bevægelse på 4 mod nord og 4 mod øst i alt koster 8*D, men med Diagonal funktionen kan nøjes med

4*D i nordøst retningen, hvis diagonale bevægelser koster det samme som horisontale og vertikale bevægelser altså D [14].

```
H(n) = D * max(abs(n.x-goal.x), abs(n.y-goal.y))
```



Figur 43 - Udbredelse af A* med diagonal distance heuristik

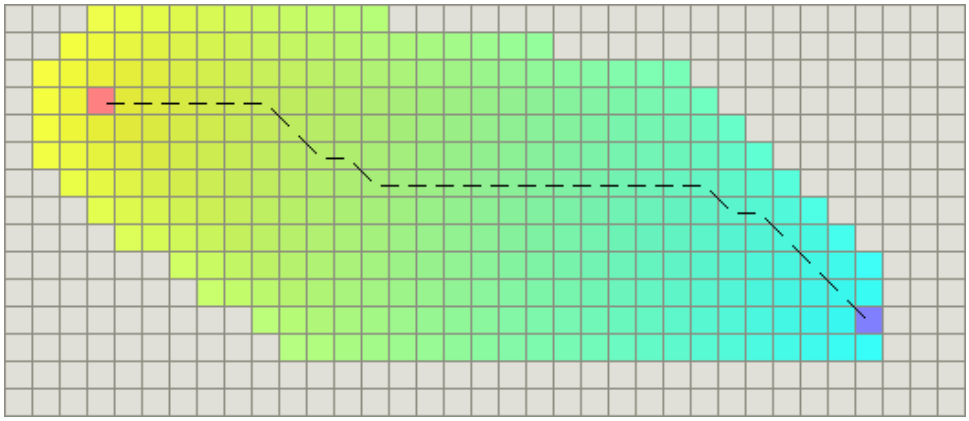
Diagonalen er ofte længere end de horisontale og vertikale bevægelser. Dette kan løses ved at anvende en anden omkostning for diagonale bevægelser, f.eks. $D2 = \sqrt{2} * D$. Da vil heuristikfunktionen være som følgende [14]:

```
h_diagonal(n) = min( abs(n.x-goal.x) , abs(n.y-goal.y) )
h_straight(n) = ( abs(n.x-goal.x) + abs(n.y-goal.y) )
h(n) = D2*h_diagonal(n)+D*(h_straight(n)- 2*h_diagonal(n)))
```

Euclidean distance

Hvis det er muligt at flytte sig i hvilken som helst retning men ikke kun i gitter retning så kan man anvende en lige linje mellem punkterne som kan beregnes vha. Pythagoras. Denne heuristik funktion kaldes Euclidean distance:

```
H(n) = D * sqrt((n.x-goal.x)^2 + (n.y-goal.y)^2)
```



Figur 44 Udbredelse af A* med euclidean distance heuristik

Euclidean distance funktionen er mere krævende pga. kvadratroden som skal beregnes [14].

Til tider kan heuristik funktionen komme ud for at flere muligheder med den korteste rute. Her er man nød til at vælge vha. en *tie-breaker* funktion til A* algoritmen.

A* algoritmen er blevet en af de mest populære pathfinding algoritmer og beskrevet i litteratur om kunstig intelligens [18]. Algoritmen er også meget diskuteret på Internettet med implementeringer i forskellige programmeringssprog som C#, C++, Java og andre kendte sprog.

4.4 Opsummering

I dette kapitel har vi set på matematiske grafer, samt hvordan man vha. flere plangrafer kan repræsentere en bygning. Ydermere så vi set på 3 algoritmer der kan anvendes til at bestemme ruter i grafen.
Den første vi så på var Dijkstra's Shortest Path algoritme, som altid giver den korteste rute men kan være tung og kræve mange beregninger. Den anden algoritme vi så på var BFS algoritmen, som fokuserer på målet og kan være mere effektiv end Dijkstea's algoritme. Men til gengæld kan man ikke være sikker på at den finder den korteste rute. Den sidste algoritme som blev undersøgt, er en AI algoritme som gør brug af heuristik og kaldes A* algoritmen. Denne algoritme kombinere principper fra de to første algoritmer hvilket gør den effektiv i tid og kan sikre den korteste rute.

Nu hvor vi har RFID teknologien på plads, samt den teoretiske viden for at implementere navigationssystemet, vil vi kaste et blik mod sikkerheden i sådan et system hvilket er emnet for næste kapitel.

Kapitel 5

Sikkerhed

Når et computersystem/netværk siges at være sikkert, så mener man som regel at tre betingelser er opfyldt, forkortet til **CIA**:

C – Confidentiality (fortrolighed)
I – Integrity (integritet)
A – Availability (rådighed)

Det handler om at sikre sig mod uautoriseret brug af systemet, samt aflytning og indblanding når systemet anvendes af autoriseret brugere. Man har brug for at kunne identificere de parter som kommunikationen sker imellem, således at man ikke giver fortroligt information til personer/computere som angiver sig for at være andre. Når kommunikationen etableres er det samtidigt vigtigt at sikrer sig mod ændring af beskederne undervejs således at information går tabt, eller aflytning så andre får indsigt i kommunikationen. Ydermere skal systemet være til rådighed for autoriseret brugere når de skal bruge det [7].

I følgende afsnit skal vi se nærmere på hvordan navigation systemet kan sikres. I afsnit 3.2 så vi på forskellige løsningsmodeller til navigeringssystemet, og kom frem til et distribueret system i form af klient-server vil være en af de mest optimale løsninger. Sådan et system indebærer kommunikation som skal sikres imellem klienterne og serveren samt sikkerhed i kommunikationen mellem RFID readeren og tags. Sikkerheden bliver derfor beskrevet i to dele hvor den første omhandler sikkerheden i RFID teknologien, mens den anden del beskriver sikringen af hele systemet.

5.1 Sikkerhed i RFID teknologien

Mens RFID teknologien udvikles og anvendes i større grad, undersøges den heldigvis af eksperter for manglende sikkerhed, hvilket i sidste ende også gavner den videre udvikling. RFID teknologiens sikkerhed afhænger en del af udstyret man anvender samt den måde systemet implementeres på. Af kendte sikkerhedsproblemer for RFID teknologien kendes:

1. **Sniffing.** Et problem man kender fra computerverden når man kommunikerer over netværk med en usikker forbindelse. Andre kan vha. software lytte med i kommunikationen og få indsigt i alle ikke-krypterede data. RFID tag er designet til at blive læst af forskellige readere inden for samme standard. Som standard vil andre kunne lytte med i de data som et RFID tag svarer tilbage med vha. en antenne eller en tilsvarende reader.

2. **Tracking.** RFID tag kan læses på afstand uden direkte kontakt mellem tag og reader. Det kan være et problem hvis folk bærer rundt på et tag eller genstande med RFID tag, da men uden deres viden eller accept kan spore hvor de bevæger sig hen eller f.eks. hvad de har købt ind i supermarkedet.

3. **Spoofing.** Dette er også et kendt problem fra computerverden som går ud på at sende falske informationer der udgiver sig for at være en anden computer/person. Vha. sniffing kan man læse RFID tags, og disse data kan man vha. en reader programmere i et tomt programmerbart tag. Dette tag kan derefter misbruges da det angiver det objekt som det blev kopieret fra. Dette blev vist ved et eksperiment [20] som viste at det er muligt at kopiere et krypteret tag og anvende det i stedet for det originale, altså vha. spoofing.

4. **Replay attack.** Undersøgelser har vist det muligt vha. en falsk reader og udstyr til at transmittere signalet over længere afstande end RFID kan række, er det muligt at retransmittere et svar fra et RFID tag. Dette gøres ved at læse et RFID tag som er lagt fra readeren med en falsk reader, transmitterer data over til en modtager tæt på readeren, som videresender RFID data til readeren som om det tag var i nærheden [21]. Dette kan misbruges til at snyde pas systemer, betalingssystemer, adgang til bygninger eller lignede systemer. Sikkerheden kan forbedres ved at udfordre svaret med det software som ligger bag den originale reader.

5. **Denial of Service (DoS).** Et af punkterne for sikkerhed som kapitlet blev introduceret med var **Availability**, hvilket er et sårbart område for RFID. Man kan forhindre systemer i at læse RFID tags ved at indkapsle dem i bestemte materialer som blokerer eller absorbere radiobølger, eller man kan ødelægge signalet helt ved Jamming. Angriber man et pas system, kan man måske i nogle tilfælde tvinge personalet til at undersøge passene manuelt uden brug af RFID chippen, hvilket fjerner den ekstra sikkerhed mod kopiering som man tilføjer med RFID chippen.

Disse punkter er nogle af de mest kendte områder hvor RFID angribes på eller kan angribes på. En undersøgelse viser at RFID kan bære virus, som kan ødelægge det bagvedliggende system. De fleste RFID systemer har noget software med en database som bruges til registrering eller identificering af tags som læses. Undersøgelsen[19] fra universitetet i Amsterdam viser i detaljer hvordan man kan inficere et RFID tag med virus som fortager et SQL-injection angreb på det bagvedliggende system. Rapportens pointe er bl.a. at man ikke skal tage sikkerheden i RFID for givet og heller ikke kun tænke på sikkerheden mellem tag og reader, men også hvordan virusinficeret RFID tags kan påvirke det bagvedliggende system som altså kræver øget omhu under design og implementering. Selvom man endnu ikke har set eller tænkt så meget på RFID vira, så er der en række faktorer som gør RFID systemer attraktive for hacker:

Store mængder af kildekode. RFID systemerne er ofte avanceret computersystemer som styres af specielt designede applikationer med mange linjers kode. Koden kan ikke være fejlfri og det er disse fejl og mangler ved implementeringen som hackere vil forsøge at udnytte.

Kendte protokoller. For at spare tid og penge i udviklingen anvendes ofte kendte protokoller som kendes fra internettet. Dette kan være en fordel, men man skal huske på at man også tager alle sårbarhederne med som man kender fra internettet. Her vil hacker også forsøge at knække systemet hvis ikke det er implementeret med korrekte og sikre protokoller.

Databaser. Som regel indgår der også en database i et RFID system som anvendes til at gemme og læse data fra RFID aflæsninger. Databasen kan i sig selv udgøre en stor sikkerhedsrisiko som beskrevet i sikkerhedsrapport for RFID fra universitetet i Amsterdam [19].

Data med høj værdi. RFID tags anvendes på objekter med høj værdi og kan derfor være attraktive for kriminelle. Det kan være data som giver adgang til virksomhedens økonomi eller handel. Men også data af betydning for et lands sikkerhed, som f.eks. data i de nye pas med RFID tags. RFID angreb kan påvirke folk og ting i den virkelige verden direkte og ikke kun computer systemer.

Falsk tryghed. Man siger at ingen sikkerhed er bedre end dårlig sikkerhed. For med ingen sikkerhed er man klar over at man skal være forsigtig, hvorimod dårlig sikkerhed kan give en falsk tryghed som kan udnyttes af hackere. Virksomheder der implementerer RFID systemer til brug internt i virksomheden og ikke online, tænker ofte ikke over risikoen for at deres system inficeres med virus eller orm fra et tag. Bedre viden inden for risikoen kan øge sikkerheden i de systemer som udvikles. RFID producenterne udvikler hele tiden nye metoder for at sikre tags og reader mod uautoriseret brug. F.eks. har Philips udviklet næste generation af RFID chips med password beskyttelse som de kalder *ICODE SLI S*. Denne chip kræver et password for at ændre visse sektioner af hukommelsen, samt mulighed for at låse et tag så den ikke kan omprogrammeres. Chippen er primært tænkt til brug af biblioteksystemer, men egner sig godt til navigations systemet som vi har beskæftiget os med. Uden lås på tags, vil hackere kunne omprogrammere tags på gulvet således at brugeren bliver kørt af en forkert rute eller i værste fald slet ikke kan finde en rute. Philips chippen virker som en god ide men noget af fleksibiliteten ryger, hvis man nu ønsker at omprogrammere nogle tags. Ydermere er sikkerheden ikke blevet undersøgt af uafhængige eksperter.

Sniffing, Spoofing og Tracking er muligt fordi der anvendes RFID tags med svag eller ingen kryptering. SkyTek [22] har annonceret en ny type RFID reader og tag med indbygget kryptering som skulle sikre chippens data. Der er tale om ekstra hukommelse med mulighed for AES (Advanced Encryption Standard) som er en krypteringsstandard fra NSA (National Security Agency), samt DES (Digital Encryption Standard) og TripleDES standarden. Deres avanceret AURA reader arkitektur giver mulighed for SHA og MD5 hashing, hvilket giver mulighed for at udvikle RFID systemer med sikkerhed som man allerede kender fra sikker netværkskommunikation.

For at opnå sikkerhed i RFID delen af navigationssystemet skal de tag som anvendes kunne låses således at data ikke kan ændres. Dette skal forhindre uautoriseret brugere i at omprogrammere tags og dermed gøre systemet ubrugeligt. Anvender man samtidigt en stærk kryptering som AES eller DES sammen med hashing algoritmer som SHA og MD5 kan

man sikre at data mod sniffing og spoofing. Det kan diskuteres om det er brugbart for hackere at få læst tag informationer, da de alligevel skal slås op i en database for at finde den tilsvarende fysiske position. Men hvis data ikke er beskyttet kan spoofing ikke forhindres, da hacker kan sende falske informationer til en bruger i rulle stol. Det vil medføre at brugeren tror at han/hun befinder sig et andet sted hvorefter ruterne vil blive beregnet med det forkerte sted som udgangspunkt. Udsyret er med andre ord afgørende for den ønskede sikkerhed i RFID kommunikationen. Navigationssystemet består ikke kun af RFID aflæsning og programmering, men det bagvedliggende system som vi selv implementerer, skal også sikres for at hele systemet kan antages at være sikker. Dette gennemgås i efterfølgende afsnit.

5.2 Sikkerhed i Systemet

Det software som skal styre logikken bag navigationssystemet er lige så afgørende for systemets sikkerhed som selve RFID delen. Når man snakker om sikkerhed i et computersystem taler man ofte om sårbarhed i systemet samt en eksisterende trussel som kan udnytte denne sårbarhed.

Sårbarheden (vulnerability) i systemet er en svaghed i sikkerheds-systemet som følge af dårligt design eller implementering der kan udnyttes til at skade systemet med.

En **trussel** (threat) mod et computersystem er et sæt af omstændigheder/forhold der har potentiale til at skabe skade eller tab på systemet.

Sårbarheden i navigationssystemet kan forekomme hvis man vælger at anvende ikke sikre protokoller, eller ikke sikre sig imod evt. åbne bagdøre i systemet. Det er vigtigt at have et godt design og retningslinjer for udviklingen som skal sikre en vis kvalitet af koden. Eksempelvis kan man have en fast udviklingspolitik om ikke at anvende *pointers* i C++ fordi det kræver stor omhu at rydde op efter sig for at undgå memory leaks. En måde at sikre kode kvaliteten på er at have code-reviews på den kode som udvikles. Det giver flere udviklere mulighed for at læse den samme kode igennem og på den måde forøge chancen for at finde usikker kode der kan udnyttes af hackere. Det kunne f.eks. være manglende check der kan medføre til *buffer-overflow*. Sådan en fejl kan udnyttes af hackere til at eksekvere egen kode på computeren hvilket kan give dem fuldt kontrol over computeren.

For at sikre systemet er vi nød til at se på hvordan det kan angribes, og ud fra det tage nogle design og implementeringsbeslutninger som skal optimere sikkerheden. Der er fire måder systemet kan angribes på:

Opsnappe. Omdirigere kommunikationen så den forkerte modtager informationen.

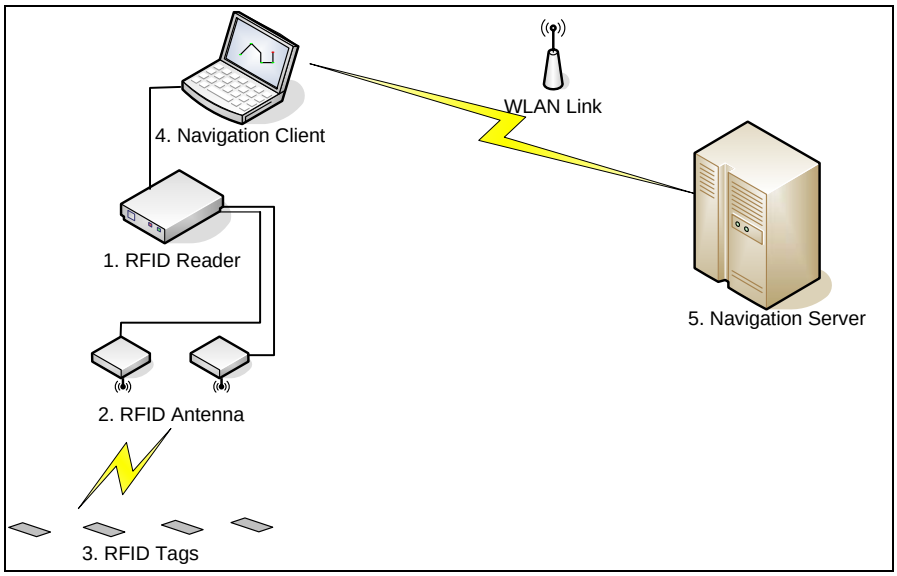
Standse. Afbrydning af kommunikationen således at data ikke modtages af den ønskede modtager.

Ændring. Modificering af data således at de har en anden betydning end da de blev sendt.

Fremstilling. Fabrikere falske data således at modtageren tror de er korrekte og kommer fra en afsender man kender.

Vi kan se på navigationssystemet og hvordan de fire angrebsmetoder kan udgøre en sikkerhedstrussel mod systemet hvis det indeholder sårbarheder.

Navigationssystemet opbygning skal som sagt bestå af en klient som fysisk er tilkoblet en RFID reader, der trådløst kan aflæse RFID tags. Sikkerheden i RFID delen så vi på i forrige afsnit. Klienten skal løbende kommunikere med en server for at holde sig opdateret, dette kan ikke gøres med en fysisk tilkoblet forbindelse, men er nød til at foregå trådløst, f.eks. med WLAN som vist på Figur 47 nedenfor.



Figur 45 - RFID Navigation opsætning

Risikoen for angreb ligger i klient-server kommunikationen som bør sikres. En hacker kan lytte med og få indsigt i oplysninger der kan krænke brugerens privatliv. Hackere kan enten lade den rigtige kommunikation fortsætte eller også kan data standses således at de aldrig når serveren eller klienten. Dette lægger op til at indholdet af data skal sikres vha. kryptering, men det sikre os ikke i at data ikke ændres undervejs eller at kommunikationen sker vha. fabrikerede data med en forkert server.

En måde at løse dette sikkerhedsproblem på er ved at anvende en sikker TLS/SSL protokol til kommunikationen mellem klienten og serveren. TLS er en forkortelse for Transport Layer Security og er udviklet til at yde sikkerhed for kommunikation i usikre netværk som f.eks. internettet. TLS er senere videreudviklet til SSL version 3.0 af Netscape Corporation.

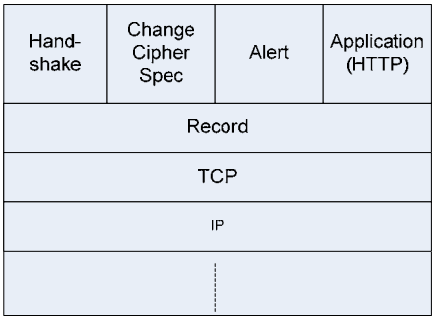
TLS sikkerhedsprotokollen er placeret mellem *Application Protocol Layer* og *TCP/IP Layer*, hvor den kan sikre og sende applikationsdata til transport laget. Protokollen giver serveren og klienten mulighed for at opdage følgende sikkerheds risici:

- Manipulation/ændring i beskederne
- Opsnapning af beskeder
- Fabrikation af falske beskeder fra en falsk afsender

TLS protokollen er inddelt i to lag. Det første lag indeholder applikationslaget samt tre *Handshake* underprotokoller:

1. Handshake Protocol
2. Change Cipher Spec Protocol
3. Alert Protocol

Det andet lag indeholder så Record Protocol, som vist på Figur 46 nedenfor:



Figur 46 - TLS protokollens relation til internetmodellen for kommunikationssystemer

Handshake protokollens funktion er at sørge for klient-server autencitet, samt udveksle sessions ID, krypteringsalgoritme, komprimeringsalgoritme, samt hemmelige data til generering af nøgler.

Change Cipher Spec protokollen opgave er at vælge den krypteringsalgoritme som Handshake protokollen valgte samt krypteringsnøgler som skal anvendes.

Alert protokollen har til opgave at alarmere fejl som f.eks. når forbindelsen lukkes, ikke gyldig besked modtages, besked ikke kan dekrypteres eller en bruger afbryder operationen.

Hver APDU som skal transmitteres gennemgår følgende processer:

1. Fragmentering af data til blokke mindre eller lig med 2^{14} bytes.
2. Komprimering uden tab (valgfrit)
3. Tilføje MAC ved at anvende en delt hemmelig MAC nøgle. MAC algoritmen kan være MD5 eller SHA-1.
4. Kryptering ved brug af delt hemmelig krypterings nøgle. Algoritmen er typisk Data Encryption Standard (DES), Triple DES (3-DES), RC2, RC4, eller Advanced Encryption Standard (AES).
5. Tilføje en header som angiver den anvendte applikations protokol: Handshake, Change Cipher Spec, Alert eller andre applikationer [17].

Inden udveksling af data sker der en klient-server autencitet som Handshake protokollen sørger for. Der anvendes *public key* også kaldet asymmetrisk nøgle kryptering til autencitet og nøgleudveksling mens der anvendes symmetriske nøgler til kryptering af resten af kommunikationen som beskrevet i punkt 4 ovenfor.
Asymmetrisk kryptering anvender et nøgle-par som beregnes samtidigt ud fra en kompliceret matematisk proces. Den ene nøgle offentliggøres i et certifikat, som typisk opbevares af en troværdig CA. Den anden nøgle, privat nøglen, holdes hemmeligt og forbliver ukendt for alle andre end ejeren. Nøglerne fungere som hinandens inverse funktioner, dvs. hvis man krypterer noget med den offentlige nøgle kan det kun dekrypteres med den private nøgle, og noget som krypteres med den private nøgle kan kun dekrypteres med den offentlige nøgle. Dette giver mulighed for at alle kan sende data krypteret med den offentlige nøgle, men kun indehaveren af den private nøgle kan dekryptere det. Samtidig kan indehaveren af den private nøgle sende data ud krypteret med den

private nøgle, og alle med den offentlige nøgle kan dekryptere det, med andre ord verificere at det må være indehaveren af den private nøgle som krypteret indholdet. Asymmetrisk kryptering anvendes også i nøgleudveksling af en symmetrisk nøgle til den efterfølgende symmetriske kryptering. Den digitale signatur sikrer **autencitet**, mens protokollens tilføjelse af MAC også sikre **integritet** i data.

Hash processen skaber en slags fingeraftryk for beskeden. Dvs. det er en proces som tilføjer data som er unikt for beskeden, og kan genberegnes med den samme hash-algoritme men man kan ikke gå tilbage fra en hash værdi til den oprindelige besked. Hashværdien er mindre end selve værdien, f.eks. er hash værdien på 128 bit når man anvender MD5 mens den er på 160 bit når SHA-1 anvendes. Når data modtages og dekrypteres, kan modtageren beregne en ny hash værdi og sammenligne den med den der er modtaget. Hvis værdien er den samme kan man være sikker på at data ikke er modificeret undervejs.

Ved at anvende en protokol som TLS kan vi altså sikre **autencitet** og **integritet** i kommunikation. Men hvis forbindelsen er en trådløs forbindelse er der yderligere sikkerhedsforanstaltninger som bør tages. Der bør sættes filtre op som kun tillader de rigtige klienter at koble op på netværket, samtidigt med man bør beskytte netværket med f.eks. WPA (eller WPA2) som til en vis grad kan sikre det trådløse netværk mod uautoriseret brug.

Det sidste punkt i vores sikring af systemet (C-I-A), er *Availability*, dvs. sikre at systemet er til rådighed for autoriseret brugere når de har brug for det. En ting er at have et godt design samt en fornuftig implementering således at systemet ikke går ned af sig selv, men en anden ting er at sikre sig mod at andre sætter systemet ude af drift. Men hvorfor skulle nogen hacke et venligt system som RFID navigationssystemet og hvornår kan de gøre det?

For at en hacker kan udgøre en trussel mod systemet skal han besidde MOM (Method-Opportunity-Motive).

Method. Den nødvendige viden, udstyr og evne til at udføre et angreb.

Oppertunity. Tiden og adgang til at gennemføre angrebet.

Motive. En grund til hvorfor han skulle gennemføre sådan et angreb.

Det er svært at få øje på motivet til evt. hackere ville angribe et fredeligt system som dette. Men hvis systemet hænger sammen med resten af

hospitalets system, eller hackere tror den gør det, kan de forsøge at knække systemet for at få adgang til personfølsomme data der i nogle tilfælde kan have en høj pris. Eksempelvis ses det ofte at når regeringsledere eller personer med særlig offentlig interesse bliver indlagt med alvorlige skader eller sygdomme holdes informationen tilbage så længe det er muligt af sikkerhedsmæssige årsager. Nogle hackere vil måske forsøge at hacke systemet for blot at se om det er muligt, mens andre vil udnytte maskinerne til at udføre andre angreb med.

Historien har lært os at når mange hackere angriber en server kan de næsten altid få den til at gå ned. Ofte anvendes andre maskiner til at sprede noget kode som til sidst kører på så mange maskiner verden rundt at når de kontakter serveren samtidigt, så kan serveren ikke håndtere alle forbindelser og går ned eller kan ikke besvare korrekte henvendelser. Dette er kendt som Distributed Denial of Service angreb, og er svære at sikre sig helt imod, selvom der findes en række mekanismer man kan anvender på sin server for at øge sikkerheden. En helt anden måde som systemet kan sættes ude af drift på er ved at jamme signalet, dvs. både RFID signalet med også WLAN signalet. Hvis dette kan opnås er hele systemet ubrugeligt. En måde at gøre det på er ved jamming af signalet, dvs. ødelægge kommunikationen mellem de trådløse enheder. Ved hjælp af en mikrobølgeovn uden metalskærm udenom kan man skabe sig en jammer som de fleste har råd til. Eksemplet viser at det stort set er umuligt at sikre systemets rådighed, men det er et beslutningsområde hvor man er nød til at tage stilling til om man kan leve med den svaghed i systemet.

Andre sikkerhedsproblemer kan opstå hvis bølgerne eller partikler fra f.eks. hospitalets røntgen maskiner påvirker signalet eller maskinens hukommelse således at data ikke er korrekte mere. Klienten skal derfor designes og implementeres med løbende check og validering af de data der arbejdes med.

Ydermere kan man ikke forestille sig at radiobølgerne fra RFID læseren og tags samt fra WLAN kommunikationen er sikre i et hospital. Det kan f.eks. påvirke deres øvrige systemer og apparater, hvilket også kræver nærmere analyse. Eksempelvis må man i nogle hospitaler ikke anvende mobiltelefon af sikkerhedsmæssige årsager, må man så anvende WLAN eller RFID?

Sikkerhedsspørgsmålet stiller krav til udstyr og udvikling, hvilket koster penge og kan derfor være afgørende for projektets gennemførsel. Men belysning af sikkerhedsproblemerne sikrer at man ikke udvikler et system

som man derefter finder ud af ikke er sikker og det ikke er noget man kan leve med.

5.3 Opsummering

Kapitlet her analyseret sikkerheden i sådan et system, hvilket resulteret i en følgende punkter som skal være opfyldt for at øge systemets sikkerhed:

- Anvende RFID Reader med mulighed for kryptering
- Anvende RFID Tags som kan låses og håndtere kryptering
- Anvend sikker kommunikation ved f.eks. at anvende sikker protokol som TLS
- Krypter evt. WLAN kommunikation med WPA eller WPA2
- Omhyggelig design af systemet
- Implementer med retningslinjer for sikkerhed
- Lav sikkerhedspolitik for anvendelse af systemet
- Adskil systemet fra andre systemer i separat netværk
- Sørg for OS altid er opdateret
- Anvend anti virus program, firewall og IDS

Hvis disse punkter overholdes er man godt sikret mod de fleste angreb, men som sagt er det svært eller umuligt at sikre systemet 100 %. Det er primært DoS angreb som man ikke kan sikre sig imod. Ydermere kan vi ikke for prototypen sikre kommunikationen mellem transponder og reader, da udstyret ikke tillader det.

På nuværende tidspunkt ved vi hvordan RFID teknologien virker samt hvordan det kan anvendes til indendørs navigering. Vi har set på den teori der skal til at udvikle sådan et system, samt hvilke overvejelser der skal ligge til grund for designet hvis man ønsker at sikre det. Inden vi kan udtale os om realiseringen af sådan et systemer vi nød til at udvikle en prototype, hvilket bliver emnet for næste kapitel.

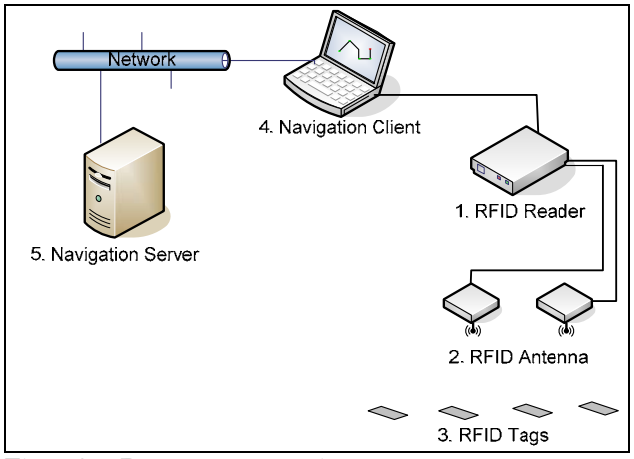
Kapitel 6

Prototypen

En del af projektets formål har været at udvikle en prototype som skal demonstrere at RFID kan anvendes til indendørs navigering. I kapitel 3 blev forskellige løsningsmodeller diskuteret. Den mest optimale løsning til det ønskede system var en client-server løsning med en *tyk* klient som selv håndterer beregningerne mens serveren sørger for opdateringer og konfiguration af klienterne. Denne løsningsmodel vil være udgangspunktet for prototypen, hvortil der anvendes følgende udstyr:

1. RFID Reader (ALIEN Technology ALR-8780[7])
2. 2 stk. Reader antenner (ALIEN Technology Circularly Polarized Antenna ALR-8610-AC)
3. RFID Tags (ALIEN Technology ALL-8338-02 med 128 bit hvoraf 96 bit brugerdata)
4. 1 Bærbar PC med Windows XP SP2
5. 1 Stationær PC med Windows XP SP2

Disse udstyr skal sammen med det navigerings software som udvikles udgøre en funktionsdygtig prototype, som illustreret på Figur 47.



Figur 47 - Prototype opsætning

[7] http://alientechnology.com/products/alr8780.php

I de efterfølgende afsnit ser vi på afgrænsningen til prototypen samt opstilling af kravspecifikationen. Ydermere ser vi på designet af prototypen samt resultatet af implementeringen.

6.1 Afgrænsning & Kravspecifikation

For at prototypen kan anvendes som en god indikator er der minimum et sæt krav som den skal opfylde. Prototypen vil adskille sig fra en evt. endelig løsning ved at være mindre detaljeret. Prototypen afgrænses derfor kun til at dække følgende områder:

- Læse RFID tags placeret på et gulv mens Readeren er i bevægelse (2,5 m/s som minimum).
- Vise brugeren en position på et kort over bygningen ved at anvende de læste RFID tags.
- Beregne den korteste rute mellem den aktuelle position og en ønskede destination.
- Håndterer etageskift med elevatorer og trapper.
- Sikker kommunikation mellem klinterne og serveren.

Disse punkter afgrænser prototypen, og vil til sammen indikere hvordan den endelige løsning kan implementeres og hvordan den vil fungerer.

Hele systemet deles op i tre applikationer:

1. **RFID NaviServer** som holder styr på konfiguration, kort, spærringer og opdatering af disse informationer hos klienterne.

2. **RFID NaviClient** som anvendes til at beregne ruter ud fra brugernes input. Holder sig opdateret med serveren via en sikker forbindelse.

3. **RFID NaviMaster** har samme funktionalitet som RFID NaviClient samt ekstra funktionalitet som giver en administrator mulighed for at oprette digitale kort samt fortage ændringer. Ændringerne sendes til serveren via en sikker forbindelse.

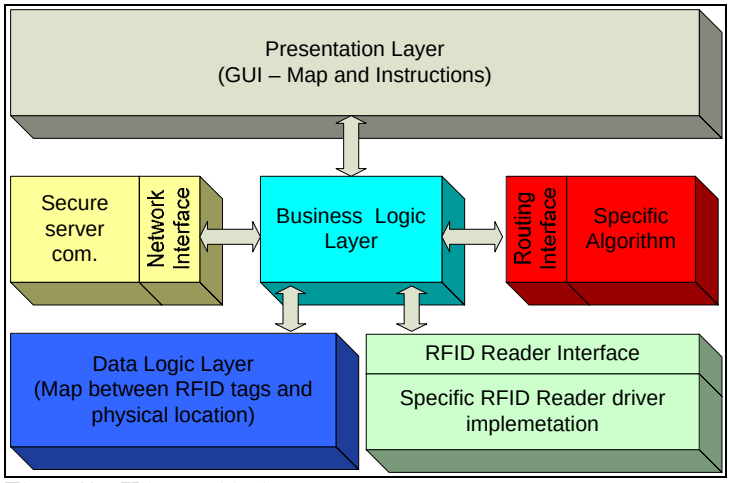
Tabellen nedenfor viser en mere detaljeret kravspecifikation for de tre applikationer:

RFID NaviServer	RFID NaviClient	RFID NaviMaster
- Lyt til administrator efter nye data - Gem seneste data og holde version styring - Lyt til klient login - Oprethold forbindelserne - Hold klienterne opdateret over en sikker forbindelse	- Konfigurerbar uden genkompilering: ▪ RFID reader driver ▪ Rute algoritmer ▪ Kort data ▪ Plantegninger ▪ Etager - Vis nuværende position - Vælg destination - Beregn korteste rute - Genberegn nye ruter - Håndtering af trapper - Håndtering af elevator - Automatisk skift mellem etager mens person bevæger sig. - Grafisk fremstilling af ruten - Zoom ind og ud på kort - Implementere en driver til Alien Tech RFID reader - Implementere en algoritme til beregning af ruter	- "Opbyg en bygning" med etager, gange mm. - Indlæs plantegninger til etager - Ændre eksisterende plantegninger - Forhåndsvisninger af plantegninger - Opret graf som repræsentation af gange i bygningen - Rediger graf - Fysisk map mellem rfid tags og graf knuder - Specificering af trapper og elevatorer - Trapper og elevator sammenhæng i etagerne - Gemme grafen i en fil - Åben og luk passager i gange - Send data til serveren over en sikker forbindelse

6.2 Design & Implementering

I følgende afsnit skal vi se nærmere på applikationens arkitektur som skal opfylde kravspecifikationen. Et af kravene til hele applikationen er at den skal være fleksibel med mulighed for bl.a. at ændre plantegninger, graf, algoritme og RFID reader uden at genkompilere hele applikationen. Dette kan opnås ved at benytte interfaces til de dele som skal kunne ændres løbende. At udvikle med interfaces giver ydermere øget fleksibilitet under udvikling da man ikke behøver at have en færdig implementering af alle dele men man kan nøjes med interfaces.

Figur 48 nedenfor viser klientarkitekturen, hvor de enkelte blokke repræsenterer hver deres ansvarsområde, mens pilene imellem angiver hvilke dele af applikationen som kan kommunikere sammen.



Figur 48 - Klient arkitektur

Klientarkitekturen har en logisk opdeling i 6 blokke og tre interfaces:

Presentation Layer er det lag som sørger for den grafiske repræsentation af ruter og plantegninger som digitale kort. Dette lag kan også ses som et interface mellem brugeren og applikationen, da dette lag fanger alle brugerens input. De input som kræver beregning eller anden form for logisk håndtering sendes videre til Business Logic laget som sender svaret tilbage. Business Logic Layer er det eneste lag som dette lag kender til.

Business Logic Layer er centrum for applikationen og det lag som kender alle andre komponenter i applikationen. Eksempelvis vil et ønske om beregning af en rute komme fra præsentationslaget, hvorefter dette lag vil anvende de andre komponenter til at samle et svar og returnere det tilbage til præsentationslaget. Med andre ord så kender dette lag til "forretningslogikken" og er derfor i stand til at behandle en forespørgsel og sammensætte svaret korrekt vha. de andre komponenter.

Routing Interface er et interface til forskellige algoritmer som kan beregne optimale ruter. Dette interface giver os mulighed for at udskifte algoritmerne så længe de implementerer interfacet.

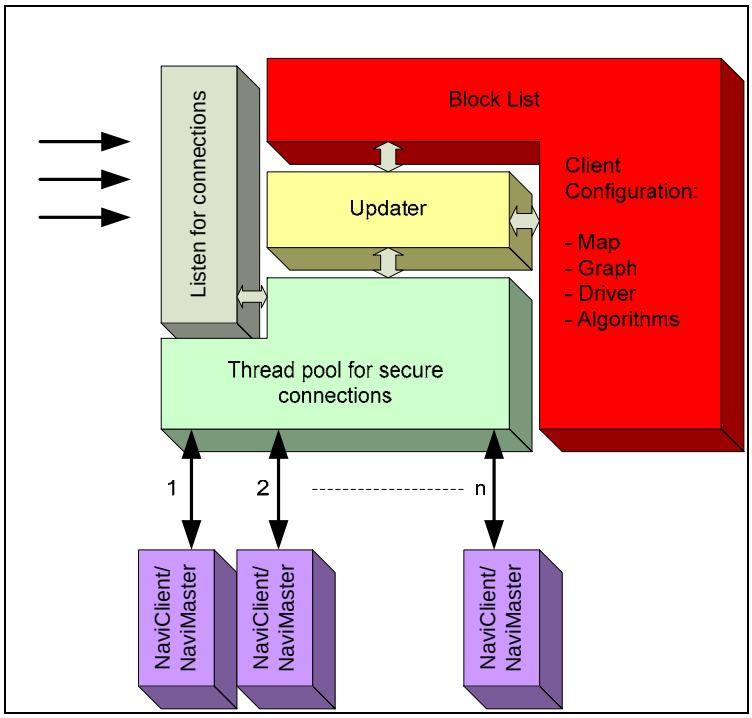
Network Interface er et interface som giver mulighed for at kommunikerer med serveren. Implementationen kan derfor implementeres som man ønsker det, f.eks. sikker kommunikation.

Data Logic Layer er det lag som håndterer grafrepræsentationen af bygningen samt sammenhængen mellem fysiske RFID tags og logisk positionen. Eksempelvis vil dette lag kunne oversætte læsningen af et RFID tag til en position på skærmen.

RFID Reader Interface er et interface som giver mulighed for at udskifte RFID Reader uden at genkompilere hele applikationen. Den nye reader kræver højst en ny implementering af interfacet, svarende til at man installerer en driver til det nye hardware.

RFID NaviMaster som er administrationsklienten har samme arkitektur som beskrevet for RFID NaviClient ovenfor, med den forskel at præsentationslaget indeholder ekstra funktionalitet som giver brugeren mulighed for at rediger i grafen for de digitale kort og sende opdateringer til serveren.

RFID NaviServeren har en anden arkitektur som er illustreret af Figur 49 nedenfor.



Figur 49 - Server arkitektur

Serveren er opbygget af fire hovedkomponenter som udgør serverens kernefunktionalitet. De fire komponenter indeholder følgende funktionalitet:

Listener delen er en tråd for sig som konstant lytter efter nye indkommende forbindelser fra klienterne. Når en forbindelse er oprettet behandles den af en anden komponent for at frigøre sig selv og lytte efter næste indkommende forbindelse.

Thread Pool er den del af applikationen som sørger for at behandle alle aktive og passive forbindelser. Denne del af serveren sørger for at aktiverer trådene, som hver er en forbindelse til en klient, når der sker en ændring som skal sendes ud til klienterne. Komponenten virker også som en aflastning for serveren da trådene her kan genbruges og man kan spare tiden det kræve at oprette en ny tråd.

Updater delen modtager ændringerne fra en RFID NaviMaster og sørger for at meddele de dele som skal bruge denne information.

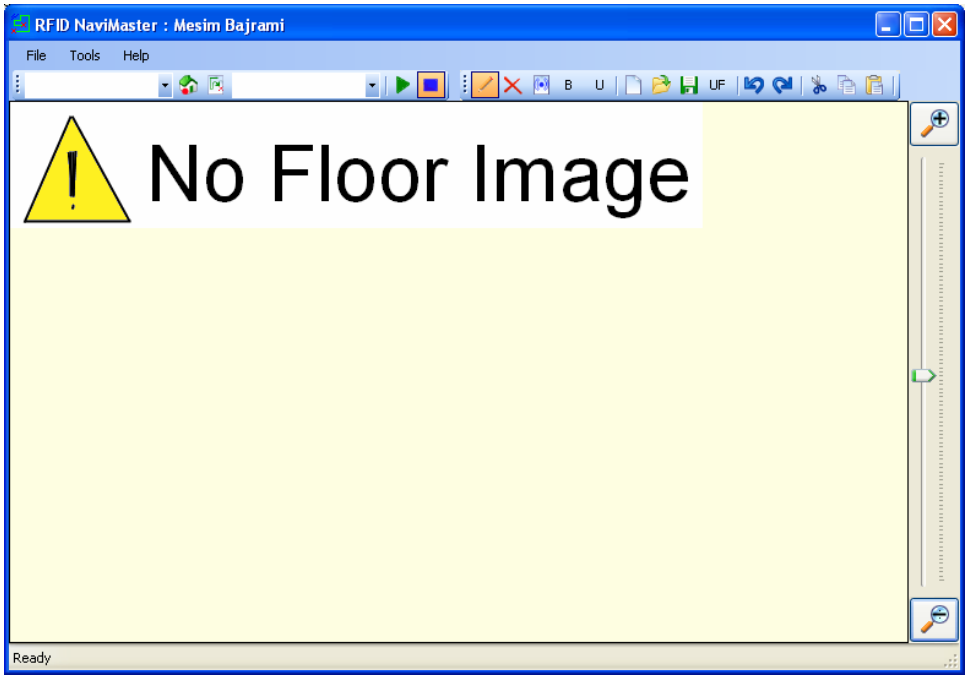
Block List er den del af applikationen som indeholder information om blokerede gange i bygning som er modtaget fra en administrator. Denne del af applikationen svarer til et data lag som indeholder alle de centraliserede data som klienterne har til fælles.

Hele prototypen er udviklet i MS .NET v2.0 teknologi, men kunne lige så godt være udviklet i Java, C++ eller et helt andet udviklingssprog.

Den grafiske del af applikationen er udviklet med standard Windows komponenter samt en række brugerdefineret komponent som selv er udviklet for at kunne tegne på samt vise ruten grafisk på skærmen.

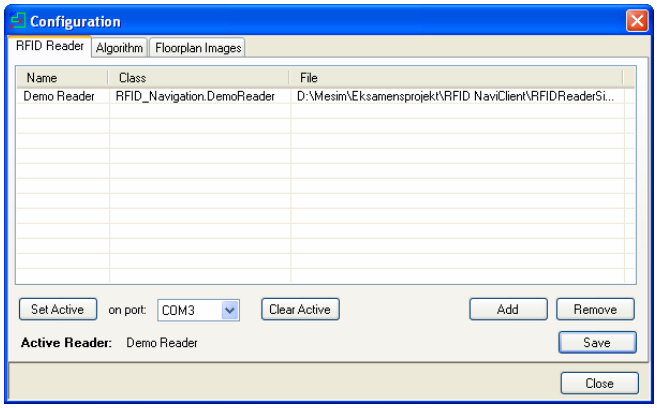
Til prototypen udvikles der kun én algoritme samt en driver til det RFID udstyr som instituttet råder over. Begge dele er udviklet som eksterne selvstændige projekter og indføres applikationen efter kompileringen vha. *reflection*. Dette er tilstrækkeligt for at vise den ønskede fleksibilitet som blev beskrevet under kravspecifikationen.

Prototypen er bygget op på en måde som viser den ønskede brugervenlighed ved en evt. endelig version af systemet. Figur 50 nedenfor viser hvordan klienten er opbygget. På skærmbilledet ses at fokus primært er lagt i plantegningen samt muligheden for zoom ind og ud. Funktionaliteten er lagt i menu øverst, samt i værktøjslinje for de mest anvendte funktioner. Figur 50 viser administratorens skærmbillede ved en helt frisk installation uden konfiguration af nogen art.

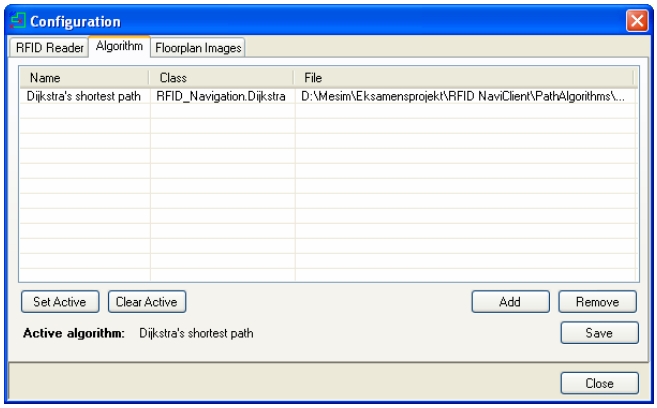


Figur 50 - RFID NaviMaster før en administrator har konfigureret noget

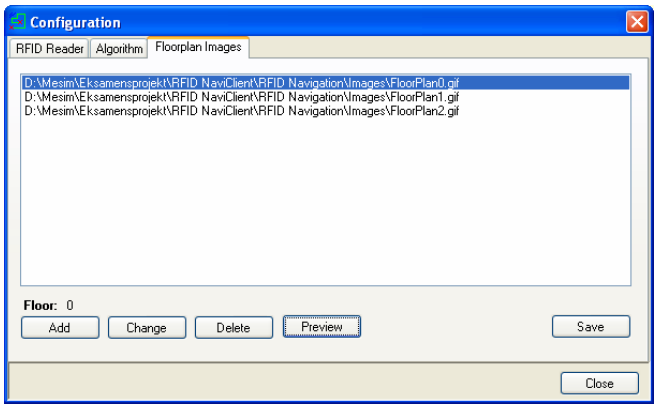
En administrator kan konfigurere applikationen ved at vælge en RFID reader, en algoritme til beregning af rute samt plantegninger for bygningen. Fremgangsmåden fremgår af skærmbillederne fra prototypen nedenfor. Se Figur 51, Figur 52 og Figur 53 nedenfor.



Figur 51 - Konfiguration af RFID reader



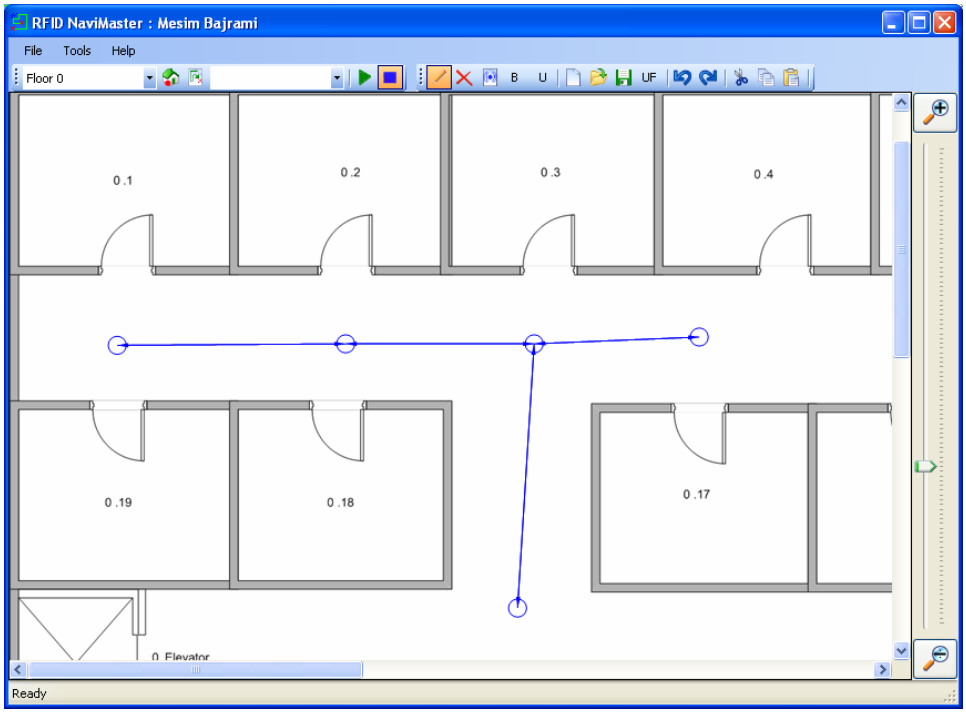
Figur 52 - Konfiguration af rute algoritme



Figur 53 - Konfiguration af plantegninger

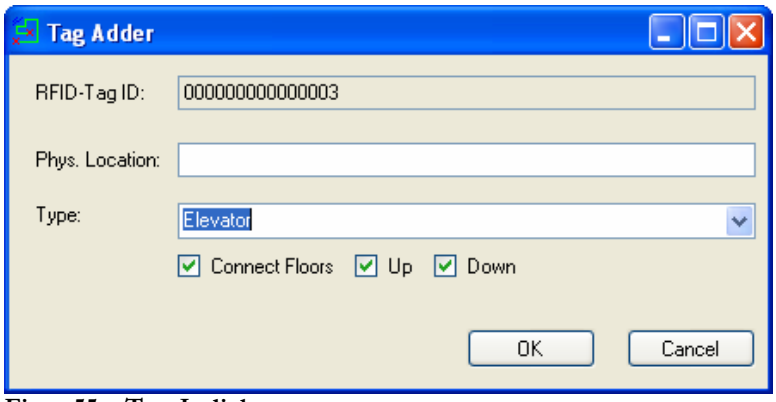
RFID Readeren og Algoritmen vælges som en dll fil. Denne fil undersøges herefter af applikationen vha. *reflection* og alle implementeringer som opfylder interfacet listes op således at de kan vælges af brugeren.

Når plantegningen og resten af applikationen er konfigureret har administratoren mulighed for at tegne en graf som skal repræsentere det digitale kort, som beskrevet under afsnit 4.1. Administratoren bestemmer selv hvor detaljeret dette kort skal tegnes, hvilket gøres ved at anvende en pegeenhed. Grafen ses på skærmen mens den tegnes som vist på Figur 54 nedenfor. Administratoren kan ændre i grafen løbende, samt kopiere og klippe, hvilket gøre arbejdet lettere hvis bygningen indeholder flere etager med samme konstruktion og den samme graf kan anvendes i flere etager.



Figur 54 - Administrator tegner en graf

Efter administratoren har tegnet grafen kan knuderne i grafen associeres med RFID tags placeret rundt i bygningen. Dette gøres ved at sætte applikationen i "Tag-It mode" som læser RFID tags og åbner en ny dialog som vist på Figur 55. Her får administratoren mulighed for at associere et RFID tag med en fysisk placering, eksempelvis *Afdeling E5.3*, samt hvilken knude på grafen det svarer til. Dette giver mulighed for at vise brugeren hvor man står når man kører rundt og skanner RFID tags, samt mulighed for at vælge en destination ud fra logiske afdelings nr.



Figur 55 – Tag-It dialog

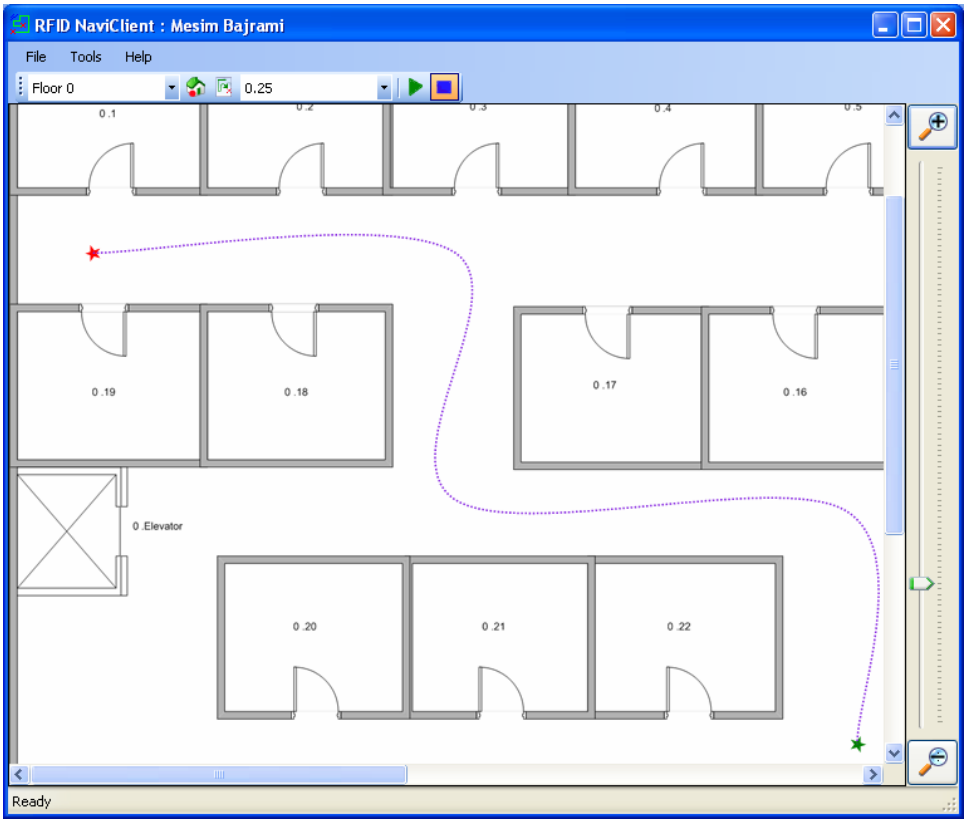
Når administrator tagger en knude på grafen er der mulighed for at bestemme hvilken slags knude der er tale om, dvs. gulv, elevator eller trappe. Når en knude er en trappe eller en elevator kan man ydermere forbinde knuden med andre knuder fra de andre etager, dvs. man

sammensætter graferne fra de forskellige etager. Se Figur 55 ovenfor. Når man forbinder flere etager sammen vil applikationen tilføje en knude med samme position, men forskellige Z koordinat svarende til etage, hvis knuderne ikke allerede findes.

Når alle data er konfigureret kan en klient anvende RFID navigationen med RFID NaviClient applikationen som vist nedenfor. Denne applikation minder om den administratoren anvender, men har begrænset funktionalitet til kun at beregne ruter.
Figur 56 viser et skærmbillede fra en klient som har beregnet en rute. Den aktuelle position vises med en rød stjerne, mens destinationen vises med en grøn stjerne. Når klienten er i bevægelse vil den løbende læse nye RFID tags, som applikationen oversætter til positioner på kortet og dermed flytter den røde stjerne. På den måde kan brugeren se hvor han befinder sig i bygning i forhold til den ønskede destination. Destinationen vælges ud fra en liste øverst i værktøjslinjen, og applikationen kan løbende beregne nye ruter mellem position og destination, hvilket giver brugeren mulighed for ikke at følge den beregnet rute men tage en anden vej hvorefter applikationen vil beregne en rute derfra. På samme måde som man allerede kender det fra bilernes GPS navigation.

Demo applikationen indeholder ikke stemmestyring som guider brugere igennem ruten men består af de to markeringer for position og destination som forbindes med en stiplet linje for ruten som vist på Figur 56 nedenfor.

Når ruten beregnes vil resultatet være en liste af knuder som man skal igennem. Disse knuder kan ses som punkter for en graf, som er den stiplede linje. Punkterne forbindes med en blød kurve vha. en matematisk *spline* funktion[16]. Her anvendes en *Cardinal Spline*, som forbinder punkterne ved at tage højde for punktet før og efter de to punkter som forbindes. Ved at tage højde for punkterne før og efter to punkter som forbindes vil alle punkterne blive forbundet med en blød kontinuert kurve som vist på Figur 56



Figur 56 - RFID navigationsklient som har beregnet den korteste rute

Alle pilene i grafen har også en status parameter som angiver om linket er åbent eller lukket. Klienten modtager status for lukket links fra serveren, hvilket betyder at algoritmerne kun anvender de åbne links til at beregne ruter. Brugeren kan derfor også selv undgå bestemte steder, f.eks. trapper ved at angive disse links som lukket, hvorefter algoritmen ikke vil medtage disse links i beregning af ruten.

Al kommunikation mellem klienterne og serveren sker via en sikker SSL forbindelse. Forbindelsen er implementeret vha. .NET SslStream som anvender TLS protokollen som beskrevet under afsnit 5.2. Kommunikationen kræver at serveren har et gyldigt certifikat, som klienten kan verificere.
Hvis man ønsker at verificere klienterne også, kan de også have et certifikat som de kan give serveren, eller de kan logge ind med brugernavn og password over den sikre forbindelse.
Prototypen er implementeret til at opfylde de ønskede krav og viser mulighederne i RFID navigation.

Der gås ikke i detaljer på kodeniveau for implementeringen, da det ville være en hel bog i sig selv med de mange tusinde linjers kode som er

skrevet. Alt kildekoden til prototypen er vedlagt en Cd-rom, som også indeholder en video der demonstrerer prototypen.

6.3 System test og kvalitetssikring

I følgende afsnit skal vi se på, hvordan man kan kvalitetssikre systemet vha. test og retningslinjer for design og implementering. Dette system kan ligesom alle andre IT systemer ikke undgå at indeholde fejl, men som designere og udviklere kan man gøre en hel del for at minimere antallet af fejl i design og udviklingsfasen.

Design fasen kan optimeres ved at have en klar kravspecifikation inden man går i gang med at designe systemet. For de fleste projekter ændrer kravspecifikationen sig løbende, hvilket kræver ændring i design og implementering. Disse ændringer har gjort SPU modellen et populær værktøj. Hvis RFID navigationssystemet skal realiseres, vil kravene sandsynligvis også ændre sig løbende, men derfor kan og skal de vigtige overordnede krav være på plads. Mange fejl kan undgås på den måde, da man kan nøjes med mindre ændringer. Eksempelvis kan man godt på forhånd vælge en klient-server løsningsmodel for systemet. Herefter kan man bygge et framework op for systemet som egner sig til den løsningsmodel. Når man langt inde i implementeringsfasen laver store ændringer, der påvirker hele frameworket, er der ikke tid og råd til at lave hele designet og implementeringen om. Dette medfører et *quickfix* her og der i applikationen som i sidste ende resulterer i en lappeløsning med masser af fejl. Ydermere kan man anvende design-patterns for forskellige problemstilligner, da disse sikre ensartet implementering i systemet.

Implementeringen er en anden fase som også kræver faste retningslinjer der skal sikre kvalitet og ensartethed i koden. Disse retningslinjer kan i nogle tilfælde begrænse udviklernes kreativitet, men er også med til at gøre systemet mere sikkert. Eksempelvis skal man ikke bare sammensætte et SQL statement ud fra en række tekst *strings* med brugerens input da dette kan medføre et *SQL-injection* angreb, og i værste fald ødelægge hele databasen. Retningslinjerne skal også klarlægge hvordan udviklerne anvender *frameworket*, og undgår *memory leaks* og *buffer overflows*.

Reviews er en vigtig del af et projekt, som desværre ofte bliver sparet væk pga. tidspres eller økonomiske årsager. Idéen med reviews er at få flere udviklere til at læse hinandens kode igennem for fejl og mangler. Her kan man fange mange svagheder som måske ikke er en fejl her og nu men udgør en risiko på sigt hvis systemet kommer i en bestemt

tilstand. Ligeledes skal designer gennemgå designs med andre udviklere for at sikre den mest optimale løsning vælges.

Unittest kræver udvikling af *testcases,* hvilket er tidskrævende. Men fordelen er at de løbende kan køres under udviklingen og sikrer at noget som virker fortsat virker efter en anden del af applikationen er ændret. For RFID navigations systemet kan man eksempelvis udvikle *unittests* som tester rutealgoritmen. Her kan man anvende en testgraf, hvor alle knuder og kanter er kendte samt afstanden mellem knuderne beregnet på forhånd. Testen kan være at give algoritmen den kendte graf og beregne forskellige ruter, som er kendte på forhånd. Hvis de beregnede ruter stemmer overens med de kendte, kan man antage at den del af applikationen fungere som forventet. Denne test kan naturligvis ikke teste alle scenarios men giver alligevel en vis form for sikkerhed for systemet fungerer. Unittest kan udvikles til alle dele af applikationen og køres hver gang man bygger systemet, hvilket kan være en automatisk rutine i det daglige byg.

Fysisk test er krævet for RFID delen af systemet. Efter implementering af systemet som skal lytte efter nye RFID tags, er man nød til at teste det med en fysisk opstilling. Her skal man teste om den ønskede rækkevidde mellem reader og transponder kan realiseres. Det er nødvendigt at teste om transponderen kan bevæge sig med minimumshastigheden i forhold til readeren og stadig blive læst. Dette skal gentages mange gange, måske 1000 gange for at få en idé om systemets stabilitet. Hvis systemet ikke er 100 % stabilt, kan det reddes af at placerer flere transpondere på gulvet. Dette sikre at de står tættere på hinanden, og hvis readeren overser én, vil den fange den næste. Men den skal stadig kunne læse over 90 % af transponderene for at systemet kan fungere flydende. Denne test skal udføres i en tidlig fase, og er afgørende for valg af udstyr.

Bruger-interface skal testes ved at kører de forskellige funktioner som applikationen tilbyder. Her skal systemet reagere som forventet, dvs. man er nød til at have *test-cases* som beskriver hvad der skal gøres, hvad det forventede resultat er som sammenlignes med det faktiske resultat. Denne del af testen skal også sikre at systemet kan håndterer forkerte bruger inputs, og ikke gå ned af det.

Brugervenlighed skal testes i en tidlig fase, måske ud fra prototypen. Her skal en brugervenlighedstest forberedes af en med kendskab til denne slags test, og udføres af brugere med kvalifikationer der minder om slutbrugernes.

Netværks test indebærer en simulering af flere klienter som kobler op til samme server. Testen her skal vise at serveren kan håndtere flere klienter samtidigt og holde dem opdateret når der sker ændringer. Netværkskommunikationen er sikret vha. en SSL forbindelse, som er en anerkendt måde at oprette sikker kommunikation over netværk på.

Black-box-test er den sidste test som bliver udført, og indeholder alle dele af applikationen. Her oprettes et kort og en matematisk graf over kortet. Transpondere fordeles på gulvet og mappes til grafen/kortet. Denne test skal vise at hele systemet fortsat fungere efter hensigten når alle dele af samlet.

Prototypens resultater har været positive i den forstand at systemet ser ud til at fungere efter hensigten. Prototypen kan på nuværende tidspunkt:

- Kommunikere med det tilgængelige RFID udstyr
- Oprette nye kort
- Redigere eksisterende kort
- Oprette/redigere matematiske grafer som repræsenterer de digitale kort
- Vise positionen når et RFID tag aflæses
- Beregne korteste rute til en destination
- Håndtere etageskift
- Håndtere trapper og elevatorer
- Løbende opdatering af positionen under bevægelse
- SSL kommunikation til en *multithreaded* server.
- Serveren kan opdatere klienterne når en passage spærres
- Klienterne anvender ikke spæret passager men beregner ruter udenom spærringerne.

Kapitel 7

Afsluttende bemærkninger

Interessen for RFID som auto identificering er stigende fordi den indeholder klare fordele sammenlignet med andre auto ID systemer. Denne sammenligning blev foretaget i rapportens kapitel 2 og viste bl.a. følgende fordele:

- EEPROM hukommelse på op til 8 Kbytes og SRAM hukommelse på op til 64 Kbytes.
- Genanvende programmerbare transpondere
- Tilføje data løbende til sporbarhed
- Tilføje information som f.eks. modtager, afsender, ejer, dato, indhold, osv.
- Non line-of-sight system
- Læseafstande fra 0-15 m afhængig af transponder og reader type

I rapporten så vi også behovet for at sikre systemet, og hvad man kan gøre for at sikre et RFID navigationssystem. Kryptering af data mellem reader og transponder er nødvendig samt sikring af netværks-kommunikation ved f.eks. at anvende en TLS protokol.

Forskellige arkitekturer blev diskuteret, hvor klient-server løsningen ser ud til at være den mest optimale løsning for systemet. Den sikre fleksibilitet i systemet som gør det muligt at opdatere data hos klienterne fra et centralt sted. For at sikre systemets drift når serveren er nede, kan man lade klienterne have kendskab til nødvendige data for at køre uden serveren.
Ydermere er den optimale løsning at lade brugerne bære rundt på readere mens transponderene placeres rundt omkring i bygningen. Dette sikre brugernes privatliv så de ikke kan overvåges, og ydermere kan det være en økonomisk fordel at lade den billige del af RFID, nemlig transponderen være spredt ud over hele bygningen.

I afsnit 6.1 blev følgende krav sat op for prototypen:

- Læse RFID tags placeret på et gulv mens Readeren er i bevægelse (2,5 m/s som minimum).
    - Test af udstyret viste at det er muligt at bevæge readeren med ca. 3 m/s og ca. 3 m fra en transponder, og stadig kommunikere mellem reader og transponder.
- Vise brugeren en position på et kort over bygningen ved at anvende de læste RFID tags.
    - Prototypen giver mulighed for at anvende plantegninger for forskellige bygninger. Ydermere kan man oprette matematiske grafer der kan associeres med RFID tag, således at brugerens position vises som en stjerne på et kort.
- Beregne den korteste rute mellem den aktuelle position og en ønskede destination.
    - Prototypen beregner den korteste rute ud fra Dijkstra's shortest path algoritme.
- Håndterer etageskift med elevatorer og trapper.
    - Prototypen anvender plangrafer for de forskellige etager. Plangraferne er koblet sammen med en kant, som repræsenterer en trappe eller elevator. Når en bruger bevæger sig fra en etage til en anden skifter prototypen selv plantegninger/kort. Det er muligt at beregne korteste rute mellem forskellige etager.
- Sikker kommunikation mellem klinterne og serveren.
    - Netværkssikkerheden er sikret vha. en SSL forbindelse, men der er ingen RFID sikkerhed da udstyret ikke tillader det.

Prototypen blev udviklet efter klient-server modellen, hvor klienterne operer efter de data som serveren giver dem. Serveren holder styr på kort over bygningen, den matematiske graf, samt spærret gange. Klienten kan herefter køre selvom den mister forbindelsen til serveren, men kan naturligvis ikke blive opdateret med nye data.
Systemet viste sig dog at være lidt ustabilt til tider hvilket i værste fald betød at 10 % af gangene kunne en transponder ikke aflæses.

Prototypen sammen med analysen viste, at idéen med RFID navigation teknisk set kan realiseres. Man skal dog være opmærksom på at systemet ikke kan sikres 100 %, da man f.eks. ikke kan sikre sig mod en jammer der kan sætte hele systemet ude af drift.

7.1 Fremtidige forbedringer

Rapporten ser kun på den tekniske side af projektet og beskæftiger sig slet ikke med økonomien. Men RFID teknologien er i dag meget udbredt og anvendes i bl.a. pas, billetsystemer, betalingssystemer og i større detailkæder som Wal-Mart og Metro Group. Den stigende interesse har både forbedret teknologien men også presset priserne ned. Dette kan indikere at systemet også rent økonomisk kan hænge sammen, men for at sige dette med sikkerhed er det nødvendigt med en nærmer analyse inden for området.

Projektet er kun testet med én reader, derfor kan test af andre reader og måske andre RFID teknologier være en god idé. Den teknologi som prototypen er bygget op omkring giver ikke mulighed for kryptering af data, og yder næsten ikke noget sikkerhed. Dette er ikke tilfredsstillende og kræver derfor en bedre RFID teknologi som kan sikres.

Prototypen kan videreudvikles så den understøtter flere readere, og algoritmer til beregning af ruter. I rapporten viste vi at A* algoritmen egner sig specielt godt til formålet pga. dens effektivitet. Denne algoritme kan med fordel implementeres i systemet.

Brugerfladen kan forbedres, ved at køre en brugervenlighedstest på den og finde den bedste måde at tilgå funktionerne på.

Ydermere vil det være en fordel at udvikle systemet så den kan køre på små bærbare enheder, som f.eks. en PDA eller lignende. Dette vil være tætter på en virkelig løsning end en bærbar pc.

Referencer

[1] Klaus Finkenzeller: *RFID Handbook 2nd edition*, Wiley 2003

[2] Simon Garfinkel & Beth Rosenberg: *RFID Applications, Security and Privacy*, Addison-Wesley 2006

[3] Thomas Hjorth: *Supporting Privacy in RFID Systems*, IMM-Thesis 2004

[4] George F Luger: *Artificial Intelligence Structure and Strategies for Complex problem Solving 5th edition,* Addison Wesley 2005

[5] Pierre Berlioux & Philippe Bizard: *Algorithms 2 Data Structures and Search Algorithms,* John Wiley & Sons 1990

[6] Alan Dolan and Joan Aldous: *Networks and Algorithms*, John Wiley & Sons 1993

[7] Charles P. Pfleeger & Sahri Lawrence Pfleeger, *Security in Computing 3. edition,* Prentice Hall PTR 2003

[8] Greogery R. Andrews, *Multithreaded, Parrallel and Distributed Programming,* Addison-Wesley 2000

[9] William R. Cheswick, Steven M. Bellovin & Aviel D. Rubin, *Firewalls and Internet Security 2. edition,* Addison-Wesley 2003

[10] Cliff C. Zou, *PCB: Physically Changeable Bit for Preserving Privacy in Low-End RFID Tags,* Whitepaper fra University of Central Florida (http://www.rfidjournal.com/whitepapers/download/124)

[11] Jim Harper, *RFID Tags and Privacy ,* 2004 Whitepaper from www.cei.org

[12] Stephen August Weis, *Security and Privacy in Radio-Frequency Identification Devices,* Whitepaper at Massachusetts Institute of

Technology 2003.
(http://www.rfidjournal.com/whitepapers/download/45)

[13] Madhav Pappu, Ph.D., Rohit Singhal, Ben Zoghi, Ph.D., RFiD IN HOSPITALS: ISSUES AND SOLUTIONS, Whitepaper 2004
(http://www.rfidjournal.com/whitepapers/download/20)

[14] Amit J. Patel (Phd), *Pathfinding*, Whitepaper from Stanford University 2006
(http://theory.stanford.edu/~amitp/GameProgramming/)

[15] Grafteori fra Wikipedia.org,
(http://da.wikipedia.org/wiki/Grafteori)

[16] Lars Eldén, Linde Wittmeyer-Koch og Hans Bruun Nielsen, *Intoduction to Numerical Computationk*, The Authers and Studentlitteratur 2004.

[17] Robin Sharp, *Principles of Protocol Design Draft 2. edition,* DTU 2004

[18] Stuart Russel, Peter Nordvig, *Artificial Intelligence: A Modern Approach 2. edition,* Prentice Hall

[19] Melanie R. Rieback, Bruno Crispo, Andrew S. Tanenbaum, *RFID Security Whitepaper*, Vrije Universiteit Amsterdam
(http://www.rfidvirus.org/papers/percom.06.pdf)

[20] Stephen C. Bono, Matthew Green, Adam Stubblefield, Ari Juels, Aviel D. Rubin, Michael Szydlo, *Security Analysis of a Cryptographically-Enabled RFID Device,* 14th USENIX Security Symposium, Baltimore, Maryland, USA, July-August 2005. USENIX.
(https://www.usenix.org/publications/library/proceedings/sec05/tech/bono/bono.pdf)

[21] Ziv Kfir, Avishai Wool, *Picking Virtual Pockets using Relay Attacks on Contactless Smartcard Systems*, Tel Aviv University 2005 (http://eprint.iacr.org/2005/052.pdf)

[22] SkyTek krypterings tag
(http://www.skyetek.com/ABOUTSKYETEK/Newsroom/PrivacyPR/tabid/243/Default.aspx)

[23] Reinhard Diestel, *Graph Theory 3rd edition,* Springer 2005 (Electronic version at: http://www.math.uni-hamburg.de/home/diestel/books/graph.theory/GraphTheoryIII.pdf)

[24] John Howard, *Hjælp til oprettelse af digitale certifikater i wondows*: http://blogs.technet.com/jhoward/archive/2005/02/02/365323.aspx

[25] Andre Azevedo, *Secure Asynchronous Socket Server and Client,* http://www.codeproject.com/cs/internet/AsyncSocketServerandClien.asp

[26] Klaus Finkenzeller: RFID Handbook (http://www.rfid-handbook.de/rfid/frequencies.html)

[27] Alien Technology (http://www.alientechnology.com/)

9 BILAG B – Prototypens Udviklingsforløb

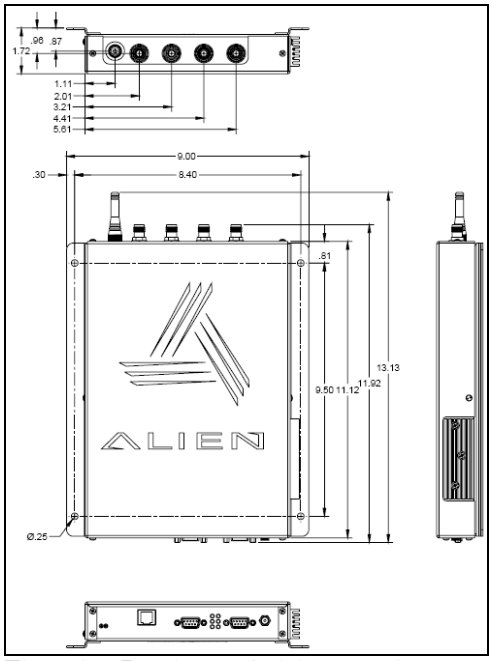
9.1 RFID Udstyret

Følgende afsnit beskriver udviklingen af prototypen, samt testen af udstyret. RFID udstyret var helt nyt udstyr på DTU uden nogen havde kendskab til hvordan det fungerede. Dette krævede en undersøgelse af udstyret for at finde ud af hvordan det virker og hvad det kan. Udstyret består af:

RFID Reader af mærket Alien Technology model ALR-8780. Denne reader er godkendt til det europæiske marked, og er designet til at læse og programmere Class 1 128-bit (96 user bit) NanoBlock tags. Readeren sender events til en computer som kan forbindes med RS-232 seriel kabel eller netværks kabel når der læses tags. Readerens specifikationer fremgår af Tabel 3 nedenfor, mens readerens fysiske design ses af Figur 57.

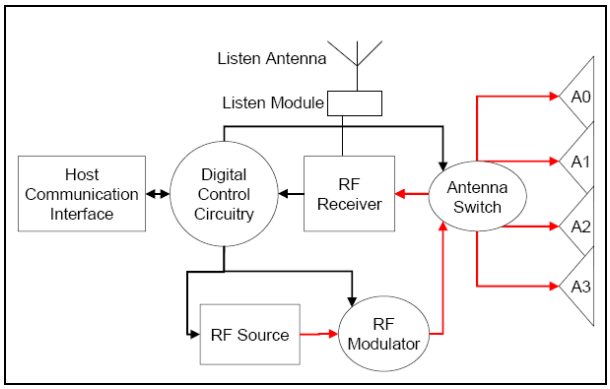
Name	866 MHz Reader and programmer
Model Number	ALR 8780
Operational Frequency	865.6 – 867.6 MHz (co-channel from June 2004)
Data Rate	~90 reads/sec (depends on Euro data rate)
Read Range	Meets or exceeds EPC range requirements
Radiated Power	2 watt (ERP), US equivalent is 3.28 watt (EIRP)
Anti-collision	Supported
Communications Interface	TCP/IP (RJ-45), RS-232 (DB-9 F)
Software Support	APIs, sample code, executable demo app (Alien Gateway)
Visual Indicators	Power, Serial (TX/RX), LAN (link, active), Tag (sniff, lock), RF (on)
Digital I/O	4 digital inputs, 4 digital outputs (DB-9 M)
Power Requirements	12V 5A output, 90-264 VAC input, 43-67 Hz, must be earthed
Compliance Certification	ETSI EN302-208, EN301-489, EN60950. Site licenses required in many European countries during 2004

Tabel 3 - Reader ALR-8780 specifikationer



Figur 57 - Readerens fysiske størrelse

Readerens arkitektur fremgår af Figur 58 nedenfor.



Figur 58 - Readerens arkitektur

Antenner: Readeren leveres med to antenner af typen ALR 8610-C med specifikationerne som vist i Tabel 4 nedenfor.

Model	ALR 8610-C
3 dB Beamwidth	E-plane: 69 degrees
Frequency	850MHz-875MHz
Gain (dBi)	6 dBi MAX
Polarization	Circular
RF Connector	6 meters with Reverse-Polarity TNC
Input impedance	50 Ohm –15dB return loss over full frequency range
Dimensions	11 X 8 X 1.5 inches 28 X 20.3 X 3.1 cm
Weight	27 oz 765.4 grams

Tabel 4 - Antenne specifikation

Tag: Der anvendes Class 1 128-bit (96 user bit) NanoBlock tags. Disse tags lever op til EPC (Electronic Produc Code) åbne standard for RFID tags.

Figur 59 - ALL-9338-02

Hukommelsen er således fordelt:

	Checksum		EPC Code (or User ID Code)								Lock	PC
Byte	0	1	0	1	2	3	…	9	10	11	0	0
Bit	0-7	8-15	0-7	8-15	16-23	24-31	…	72-79	80-87	88-95	0-7	0-7

EPC koden er på 96 bit uden begrænsning for hvad der kan ligge i dette område.
Checksummen bliver beregnet og programmeret i tagget automatisk af readeren over de 96 bit EPC med CCIT-16 standarden og gemmes i de 16 første bit.
Lock og PassCode(PC) er de sidste to bytes af hukommelsen og bruges til at låse et tag med og ødelægge et tag med.

Kravene til udstyret kan fungere er:

- PC Windows 98 eller højere, med RS-232 seriel port.
- Strømforsyning 100V-240V AC på 1.5A (50Hz-60Hz)
- Host software
- RFID Tags EPCglobal Class 1

Vi anvender en computer med Windows XP SP2, RS-232 seriel port, netværks stik, P4 3.4 Ghz processor, 2GB RAM, hvilket lever op til udstyret minimumskrav.

9.2 Opsætning af udstyret

Opsætning af udstyret sker på følgende måde:

1. Forbind RS-232 kablet til readeren
2. Forbind RS-232 kablet til Pc'en
3. Forbind strømforsyningen til readeren
4. Forbind antennerne til readeren, også de lille lytter antenne
5. Sæt strøm til udstyret
6. Tænd computeren og start evt. det software som skal bruges.

Figur 60 - Eksempel på opsætning af udstyret

9.3 Test af udstyret

Til at starte med blev udstyret testede med det software der følger med udstyret. Med dette software kan er det muligt at:

- Læse et til flere tags
- Konfigurere readeren
- Omprogrammere tags
- Låse tags
- Ødelægge tags (hvis man ønsker at gøre et tag ubrugeligt)

1. Test resultat

Problem: Lamperne på readeren lyser som de skal, softwaren på computeren kører som den skal men ingen tag kan læses.

Løsning: Efter flere forsøg lykkedes det ikke at læse et tag. Antennen blev udskiftet med den anden antenne, men det hjalp ikke. Til sidst blev begge antenner sat på, hvilket hjalp. Også selvom kun den ene antenne blev valgt vha. softwaren og den antenne ikke peget nogen steder mod de tags som skulle aflæses. Det fremgår ikke af dokumentationen nogen steder at to antenner skal være tilkoblet før udstyret virker, tværtimod står der at man kan anvende 1,2,3 eller 4 antenner med denne reader efter behov. På internettet var der heller ikke noget hjælp at hente. Efter en henvendelse hos leverandøren fik vi besked om at begge antenner skal være på ellers risikere man at ødelægge udstyret. Og at årsagen skulle være at den ene sender energien til et tag mens den anden antenne lytter. Denne forklaring stemmer ikke helt overens med de erfaringer som er tilegnet under forsøgene, men da vi ikke har købt noget support hos leverandøren må vi leve med den forklaring og have begge antenner på. Leverandørens forklaring ses her:

```
-----Original Message-----
From: s010224@student.dtu.dk [mailto:s010224@student.dtu.dk]
Sent: 28. marts 2006 10:44
To: sg@betatechnic.dk
Subject: RFID Reader ALR-8780
Importance: High


Hej Sune Granzow,

Mit navn er Mesim Bajrami og jeg er Civilingeniørstuderende ved DTU. Jeg er i gang med mit
eksamensprojekt hvor jeg anvender noget RFID udstyr som min vejleder, Christian Damsgaard Jensen, har
købt hos jer for et par måneder siden. Det drejer sig om en ALR-8780 Reader fra Alien Technology med
to antenner. Jeg skriver til dig fordi jeg oplever at udstyret er ualmindelig ustabilt, forstået på
den måde at jeg kan få det til at virke men det er svært at læse RFID-Tags stabilt (dvs. hver gang jeg
kører readeren over et tag).

Jeg skal sætte begge antenner på for det virker, selvom jeg kun anvender en af dem. Er dette normalt
og hvorfor? Det fremgår nemlig ikke af de officielle dokumenter fra producenten.

Jeg skal have sat RFID tags på et linoleumsgulv hvor underlaget er beton.
Dette virker nogen gange men for det meste kan jeg ikke læse de tags.
Hvorfor? Kan jeg finde et sted hvor der er en god forklaring på hvor man kan sætte tags og læse dem?

Jeg har også forsøgt på en computer (metal) og det virker heller ikke specielt godt? Ydermere har jeg
```

forsøgt på plastik, træ og papir.

Jeg har selv konfigureret Readeren, samt forsøgt med standard indstillingerne og det er stadig et problem. Jeg forsøger også med det software som følger med, men uden den store succesoplevelse.

Det er første gang jeg arbejder med RFID teknologien, og ud fra det jeg kan læse om det så burde det virke bedre end det jeg oplever. Jeg håber du kan hjælpe mig med at afklare om det er udstyret der er noget galt med eller om det generelt er teknologien som er ubrugelig.

Venlig hilsen
Mesim Bajrami
s010224@student.dtu.dk

Hej Mesim,

1. UHF teknologien som du sidder med i hænderne er stadig under udvikling.
Dog vil jeg sige at det burde være mere stabilt end det du der beskriver.
Derfor vil jeg anbefale at du opgraderer til den seneste firmware.
Gatewayprogrammet kan også opgraderes. Jeg har vedhæftet begge filer. .jar filen er til gatewayprogrammet, du skal bare lægge der hvor den gamle .jar fil ligger, og .zip filen er til at opgradere læserens firmware med.

2. Systemet virker KUN ved at begge antenner er monteret og peger mod taggen. Rent faktisk kan læseren blive beskadiget ved kun at anvende en antenne, et faktum jeg har også indskærpet overfor Chr. Damsgaard. Det virker på den måde at energien bliver sendt af den ene antenne mens den anden lytter.

3. Det er HELT sikkert at UHF teknologien er mere følsom end f.eks LF
(125KHz) og HF(13,56 MHz). Den store fordel ved UHF er læseafstanden og tagpriserne ved store kvantiteter.
Det kan være svært at tagge visse materialer. Træ f.eks kan give store udfordringer, idet det kan indeholde fugtighed. UHF er følsomt overfor fugt idet antenneenergien bliver afsat i vandet istedet for i tagantennen. Metal kan også have en negativ virkning på radiofeltet. Man kan dog anvende specialicerede tags som har til formål at blive monteret på metal. Du siger at du prøver at læse på tags der sidder ovenpå et betongulv. I betongulve er der jo jernarmering, men dog mener jeg at det skal kunne lade sig gøre.

Mvh
Sune Granzow
BETA technic

Problem: Udstyret er meget ustabilt forstået på den måde at den af og til ikke kan læse tags. Man skal helt tæt på eller vifte med antennen flere gange for at den ser et tag.

Løsning: Der blev forsøgt forskellige materieller som baggrund for et tag:

- Træ
- Plastik
- Metal
- Flamingo
- Stof
- Papir

Fugtige ting virker meget dårligt med det her udstyr fordi det absorberer radiobølger hvilket gør læsningen besværlig/umulig. Men løsningen til problemet var ikke baggrunden men mere en opdatering af firmwaren på readeren hvilket hjalp.

Efter firmware opgraderingen virkede udstyret nogenlunde, dvs. man skulle omhyggeligt vælge tags som virkede. Også helt nye og ubrugte tags virkede ikke nogen gange, og det skyldtes at disse tags er meget skrøbelige og at antennen hurtigt kan rive sig fri fra chippen. Så de tag som virkede, skulle behandles ordentligt hvis man ønsker at de fortsat skal fungere.

Herefter kunne et tag:

- Programmeres
- Omprogrammeres
- Låses

Låsen er meget usikker da den består af en kode på 8bit, og man har ubegrænset forsøg til at finde koden. Dette giver 2^8 muligheder, altså 256 kombinationer hvilket man dårligt nok kan kalde for sikkerhed. Dette skal nok mere ses som en sikkerhed mod fejlagtig omprogrammering af tags, men ikke en sikring mod uautoriseret brug.

Efter test af udstyret var det tid til selv at udvikle den nødvendige software.

Udviklingens faser er sket som følgende:

Fase 1: Test af udstyret for at få kendskab til mulighederne og begrænsningerne.

Fase 2: Udvikling af driver som beskrevet ovenfor.

Fase 3: Udvikling af datastruktur til graf repræsentation.

Fase 4: Udvikling af en grafisk brugerkomponent som tillader oprettelse og ændring af en graf.

Fase 5: Udvikling af en algoritme, som kan beregne den korteste rute mellem to punkter.

Fase 6: Sammensætning til en fælles klient, med alle nødvendige funktioner.

Fase 7: Udvikling af sikker kommunikation vha. SSL mellem en klient og en server.

Fase 8: Udvikling af en server, der anvender parallelle tråde samt en sikker SSL kommunikation til klienten.

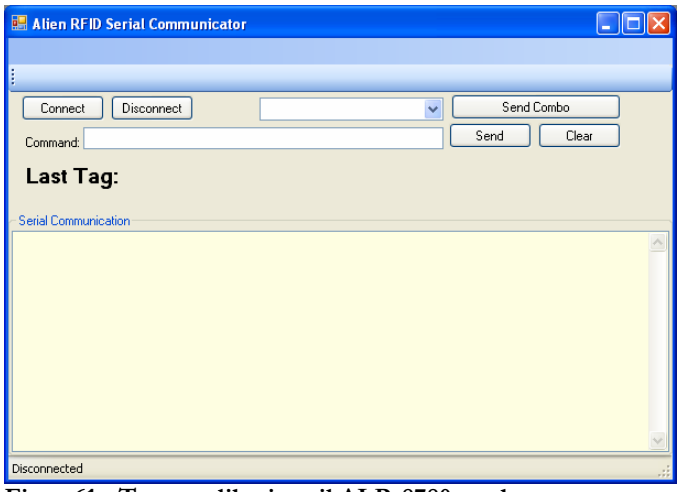
Fase 9: Opdeling af Klienten til en master og slave.

Fase 10: Test af det samlet system og evt. fejlrettelser.

9.4 Driver Udvikling

Prototypen skal have et RFID reader interface der tillader forskellige RFID readere at blive anvendt med den udviklede applikation. Dette betyder at der skal implementeres et modul som opfylder interfacet og kan anvendes som en "driver" til det RFID udstyr som vi har til rådighed.

Driveren blev udviklet sammen med et testprogram for at se om alle kommandoer kunne sendes til Readeren korrekt, samt om readeren svaret som ønsket. Testprogrammet ses på figuren nedenfor:



Figur 61 - Test applikation til ALR-8780 readeren

Denne applikation kan oprette en forbindelse til readeren, og sende kommandoer hvorefter readeren svarer tilbage, og kommunikationen kan følges i den store tekstboks nederst i applikationen.
Driveren bliver udviklet som et bibliotek (dll fil) således at den kan anvendes af navigationssystemet herefter.

Interfacet mellem PC og reader foregår via. et RS-232 kabel og sker ud fra følgende kommandoer:

General Commands

Command	Description
Help ("h")	List reader commands.
Info ("i")	List current reader settings.
!	Repeat last command.
Save	Save current settings to flash memory.
get|set Function	Change the operation from "Reader" to "Programmer" and back.
get|set ReaderName	Associate an arbitrary name with the reader.
get ReaderType	Get a description of the reader type.
get ReaderVersion	Get the reader software versions.
get Uptime	Return the time in seconds since the last reboot.
get|set Username	Get and Set the username used for the network-based access control.
get|set Password	Get and Set the password used for the network-based access control.
get MaxAntenna	Returns the maximum addressible antenna port number.
get|set AntennaSequence	Get and Set the antenna port sequence the reader should use.
get|set RFAttenuation	Get and Set the amount of digital attenuation to apply to the emitted RF signal.
FactorySettings	Reset the reader to its original factory settings.
Reboot	Reboot the reader.

TagList Commands

Command	Description
get TagList ("t")	Get the current list of active tags from the reader.
get|set PersistTime	Get the time a "stale" tag remains on the TagList.
get|set TagListFormat	Get and Set the format for how TagLists are returned (Text, XML, Terse, Custom).
get|set TagListAntennaCombine	Specify whether to combine reads of a tag from different antennas into one TagList entry.
get|set TagListCustomFormat	Specify a custom format for TagLists.
get|set AcquireMode	Get and Set how tags are acquired (Global Scroll or Inventory).
Clear TagList	Clear the list of active tags on the reader.
get|set AcqCycles	Specify the number of acquisition cycles to perform during each tag read action.
get|set AcqCount	Specify the number of reads to perform in each cycle.
get|set AcqEnterWakeCount	Specify the number of times to Wake tags before each acquisition cycle.
get|set AcqExitWakeCount	Specify the number of times to Wake tags at the end of each acquisition cycle.
get|set AcqSleepCount	Specify the number of times tags are slept, as they are read.
get|set TagType	Specifies which RFID tag types the reader should look for.
Wake	Wake tags.
Sleep	Sleep tags.
get|set Mask	Specify a Mask for addressing tags.

Notify Commands

Command	Description
get|set NotifyFormat	Get and Set the format for notification messages (Text, XML or Custom).
get|set NotifyHeader	Turn the header/footer lines of notification messages on and off.
get|set NotifyAddress	Get and Set the address to push TagLists to.
get|set NotifyTime	Get and Set the time interval for automatically pushing TagLists.
get|set NotifyTrigger	Get and Set the trigger for pushing TagLists (Add, Remove, Change, True, False, TrueFalse).
get|set NotifyKeepAliveTime	Get and Set the time the reader leaves its Notification network socket open.
get|set MailServer	Get and Set an SMTP mail server. Only required if notification email messages used.
get|set MailFrom	Get and Set the "From" email address for email notifications.
get|set NotifyMode	Get and Set Notify Mode state (On / Off).
get|set NotifyRetryCount	Get and Set how many times an unsuccessful network notification is tried before failing.
get|set NotifyRetryPause	Get and Set how long to wait before retrying an failed network notification.
NotifyNow	Send an immediate message via the notification system.

Network Configuration Commands

Command	Description
get MACAddress	Return the reader's unique hardware identifier.
get|set DHCP	Turn DHCP on or off. If DHCP is on, the reader automatically configures itself for the network on power-up.
get|set IPAddress	Get and Set the IP address of the reader. If DHCP is enabled, this is set automatically.
get|set Gateway	Get and Set the network Gateway. If DHCP is enabled, this is set automatically.
get|set Netmask	Get and Set the Subnet Mask. If DHCP is enabled, this is set automatically.
get|set DNS	Get and Set the Domain Name Server. If DHCP is enabled, this is set automatically.
get|set NetworkTimeout	Get and Set the idle seconds before network connections timeout.
get|set CommandPort	Get and Set the port number to listen on for commands issued over the network.
get|set HeartbeatAddress	Get and Set the IP address to send heartbeat messages to.
get|set HeartbeatPort	Get and Set the port number to send heartbeat messages to.
get|set HeartbeatTime	Get and Set the time interval, in seconds, between successive heartbeats.

Time Commands

Command	Description
get|set Time	Get and Set the reader's local time.
get|set TimeZone	Get and Set the time zone offset from UTC.
get|set TimeServer	Get and Set the address of a network timeserver.

External I/O Commands

Command	Description
get ExternalInput	Get the external input pin values.
get|set ExternalOutput	Get and Set the external output pin values.
get|set InitExternalOutput	Get and Set the initial state of external outputs.
get|set InvertExternalOutput	Turn the inversion of output states on and off.
get|set InvertExternalInput	Turn the inversion of input states on and off.

AutoMode Commands

Command	Description
get|set: AutoTrueOutput AutoFalseOutput AutoWaitOutput AutoWorkOutput	Get and Set the value of the digital output pins when in the states of AutoMode. (Waiting, Working, Eval True, Eval False).
get|set: AutoStartTrigger AutoStopTrigger	Get and Set the values of the start and stop triggers for the Working state.
get|set: AutoStopTimer AutoTruePause AutoFalsePause	Get and Set the delays for various AutoMode states.
get|set AutoAction	Set the action to be performed while working
AutoModeReset	Turn AutoMode off and reset parameters.
AutoModeTriggerNow	Force AutoMode to start, if waiting for a trigger.
get|set AutoMode	Get and Set AutoMode state (On / Off).

Readeren kan ud fra disse kommandoer sættes op til at operere i forskellige tilstande. Vi ønsker at den skal kører i automatisk tilstand hvor readeren konstant sender energi ud og lytter om der er et tag som svarer tilbage. Hver gang den læser et tag skal den sende resultatet til Pc'en. Denne opsætning programmeres i driveren således at opsætningen kan ske automatisk og evt. tidligere opsætninger nulstilles. Opsætningen sker med følgende kommandoer:

AutoModeReset
set NotifyFormat = Terse
set AutoAction = Acquire
set AutoStartTrigger = 0,0
set AutoStopTimer = 0
set AutoTrueOutput = 1
set AutoTruePause = 50
set AutoFalseOutput = 2
set AutoFalsePause = 50
set NotifyAddress = serial
set NotifyTrigger = Add

```
set NotifyMode = On
set AutoMode = On
```

Koden C# (.NET 2.0) til kommunikationen med readeren ses nedenfor:

```
private void connect()
{
    this.serialPortAlien.PortName = "COM1";
    this.serialPortAlien.BaudRate = 115200;
    this.serialPortAlien.Parity = Parity.None;
    this.serialPortAlien.DataBits = 8;
    this.serialPortAlien.StopBits = StopBits.One;

    try
    {
        this.serialPortAlien.Open();
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.Message);
    }
}
private void disconnect()
{
    if (this.serialPortAlien.IsOpen)
    {
        this.serialPortAlien.Close();
    }
}

private void sendCommand(string message)
{
    try
    {
        this.serialPortAlien.WriteLine(message);
        this.textBoxCommunication.AppendText("\r\nAlien  >  "  +  message  +
"\r\n");
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.Message);
    }
}

private        void        serialPortAlien_DataReceived(object        sender,
SerialDataReceivedEventArgs e)
{
    this.Invoke(new EventHandler(ReadData));
}

private void ReadData(object s, EventArgs e)
{
    try
    {
        tempMessage += serialPortAlien.ReadExisting();

    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.Message);
    }

    if (tempMessage.EndsWith("\0"))
    {
        textBoxCommunication.AppendText(tempMessage);
        tempMessage = "";
    }

}
```

9.5 Grafrepræsentation

Grafrepræsentationen ses på klassediagrammet nedenfor.

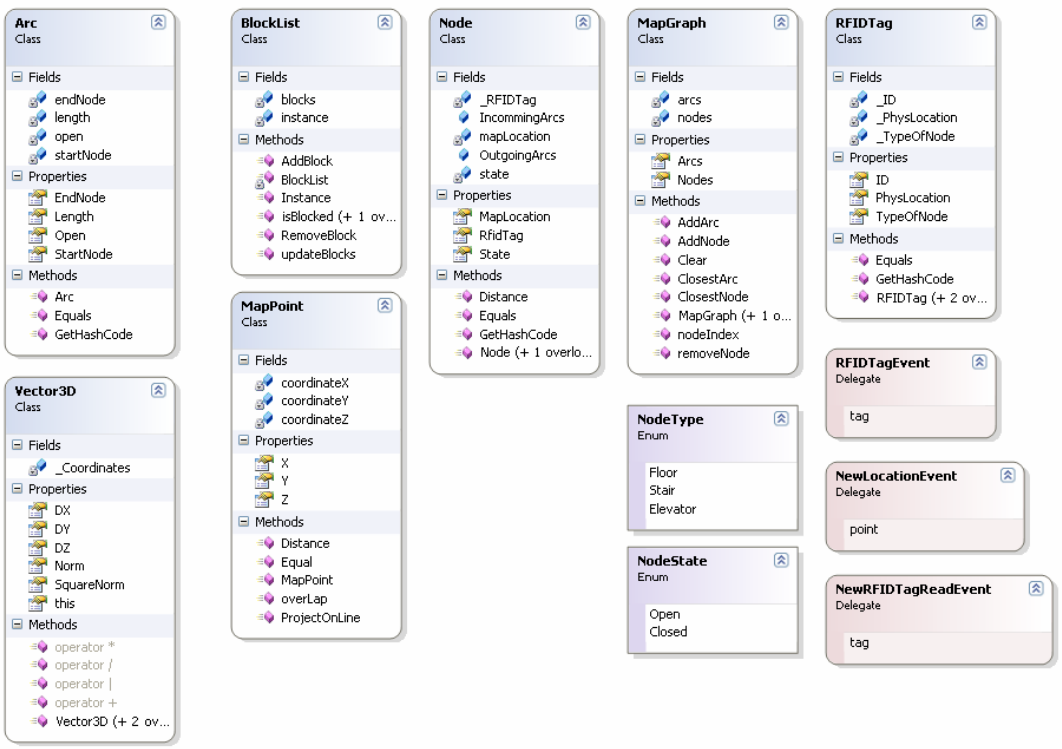
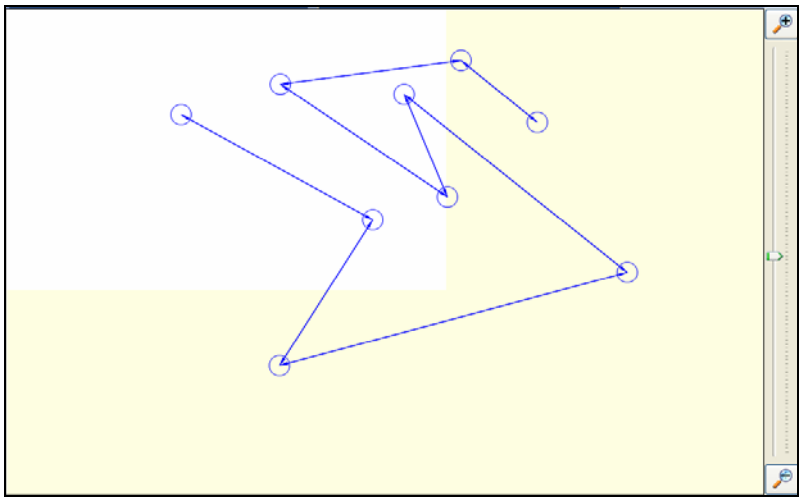


Diagram 1 - Klassediagram for den matematiske grafrepræsentation af kort

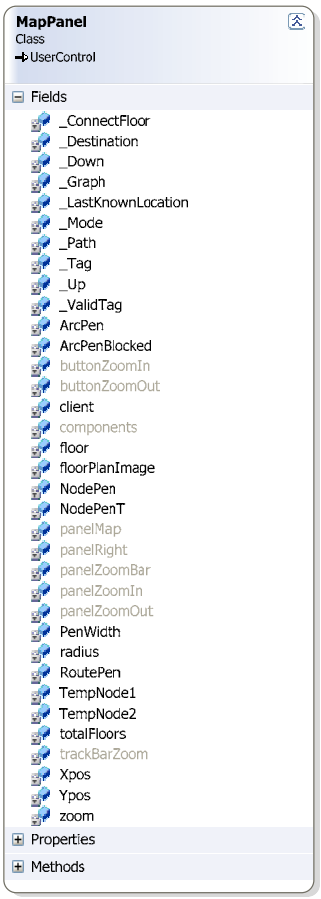
9.6 Grafisk komponent til grafdesign

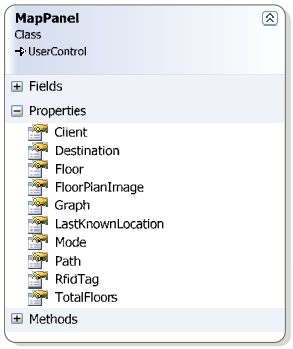
Den grafiske komponent kan anvendes i alle dele af applikationer og har forskellige funktioner:

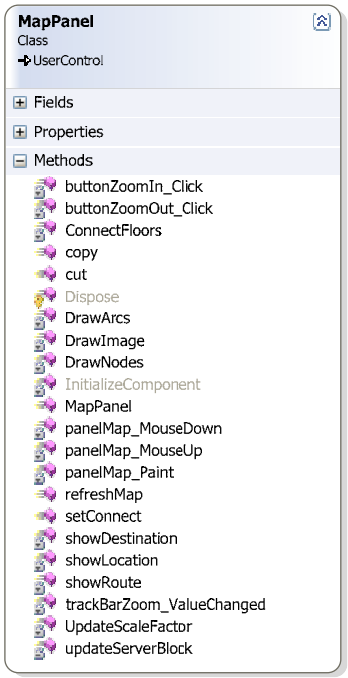
- Klik med musen opretter en knude
- Klik ned → flyt musen og slip et andet sted opretter to knuder med en forbindelse imellem
- Klik på en knude og slip musen over en anden knude forbinder knuderne (hvis de ikke allerede er forbundet)
- Slet en knude/forbindelse
- Udføre specielle opgaver på en knude ved at sætte kontrollen i bestemt mode.
- Zoom ind og ud ved at klikke til højre eller bruge scrolleren på musen.



Figur 62 - Grafisk komponent til grafdesign

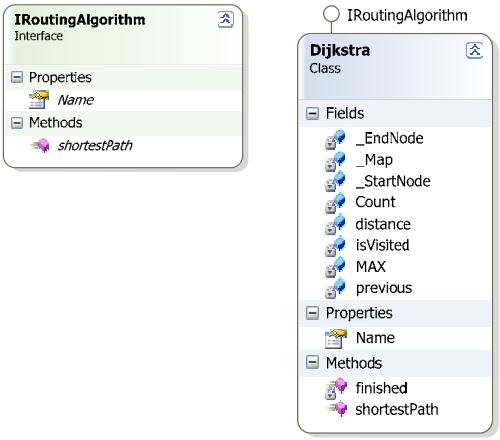

MapPanel
Class
UserControl
Fields
_ConnectFloor
_Destination
_Down
_Graph
_LastKnownLocation
_Mode
_Path
_Tag
_Up
_ValidTag
ArcPen
ArcPenBlocked
buttonZoomIn
buttonZoomOut
client
components
floor
floorPlanImage
NodePen
NodePenT
panelMap
panelRight
panelZoomBar
panelZoomIn
panelZoomOut
PenWidth
radius
RoutePen
TempNode1
TempNode2
totalFloors
trackBarZoom
Xpos
Ypos
zoom
Properties
Methods


MapPanel
Class
UserControl
Fields
Properties
Client
Destination
Floor
FloorPlanImage
Graph
LastKnownLocation
Mode
Path
RfidTag
TotalFloors
Methods

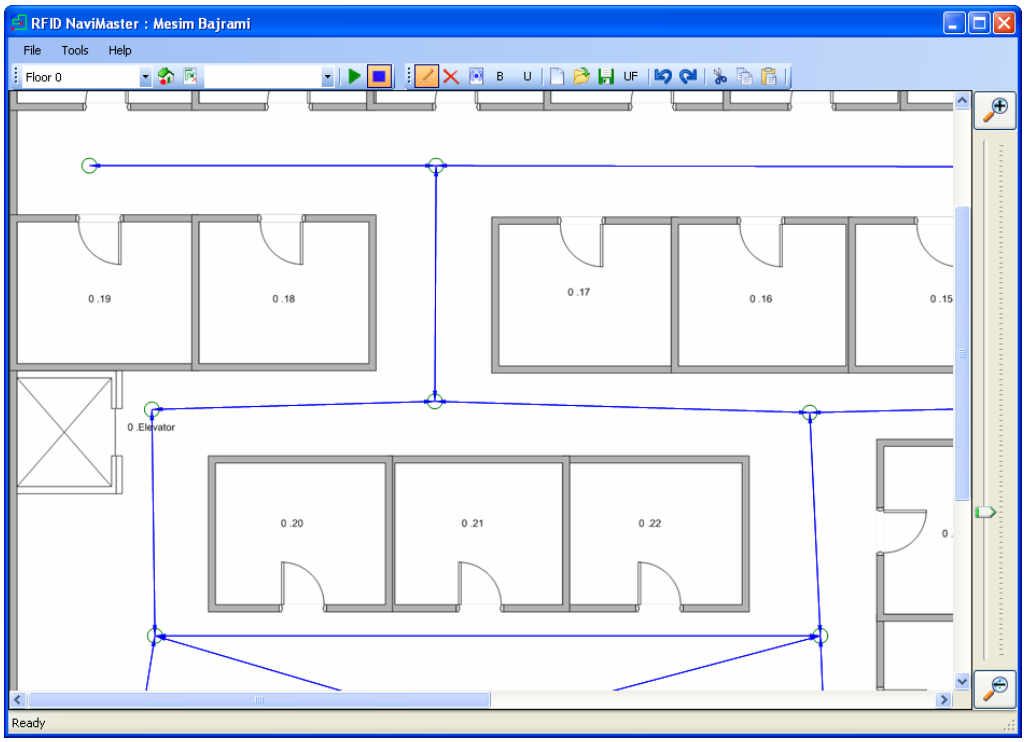

MapPanel
Class
UserControl
Fields
Properties
Methods
buttonZoomIn_Click
buttonZoomOut_Click
ConnectFloors
copy
cut
Dispose
DrawArcs
DrawImage
DrawNodes
InitializeComponent
MapPanel
panelMap_MouseDown
panelMap_MouseUp
panelMap_Paint
refreshMap
setConnect
showDestination
showLocation
showRoute
trackBarZoom_ValueChanged
UpdateScaleFactor
updateServerBlock

9.7 Shortest Path Algoritmen

Som algoritme til prototypen er udviklet Dijkstra's shortest path algoritme. Teorien for denne algoritme er gennemgået i rapporten. Algoritmen er udviklet som et bibliotek der implementerer IRoutingAlgorithm interfacet:



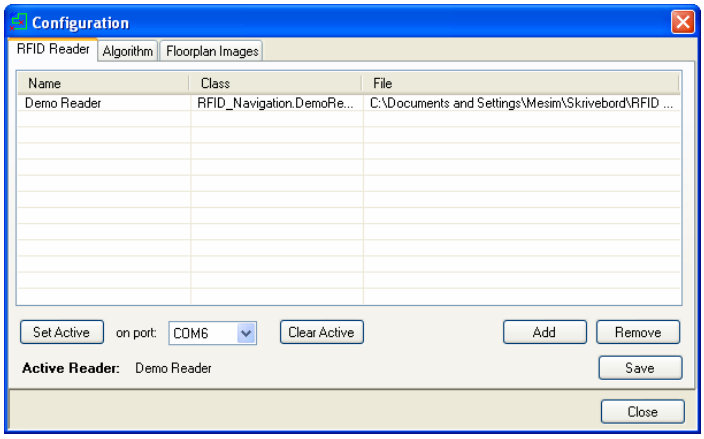
9.8 Sammensætning af RFID Navigation Klient



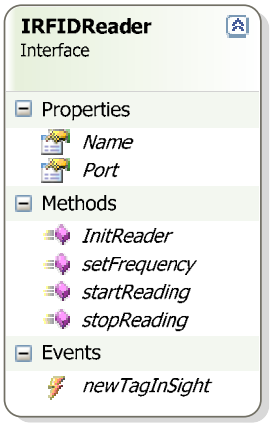
Klienten kan deles op i fire dele:

- **Menubar**: Menubaren giver adgang til de funktioner man ikke bruger så ofte. F.eks. at ændre applikationens opsætning.
- **Toolbar**: Værkstøjslinjen giver adgang til applikationens mest anvendte funktioner. Den er delt op i to dele, hvor den ene benyttes til at anvende applikationen som navigation, mens den anden anvendes til at tegne og redigere grafen med.
- **Grafiske kort**: Denne del af applikationen er bruger komponenten som viser kort og ruter, samt mulighed for at zoom ind og ud.
- **Status bar**: viser evt. meddelelser for applikationens status.

Når applikationen skal konfigureres sker der igennem menuen Tools→Settings som åbner følgende dialog:



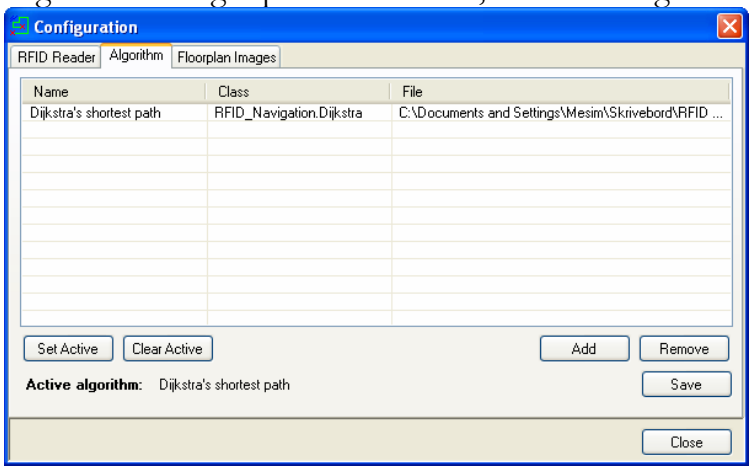
Her kan man sætte op hvilken reader, algoritmer og plantegninger der skal anvendes. Readeren indsættes ved at vælge et bibliotek, som undersøges for match med IRFIDReader interface klasse:



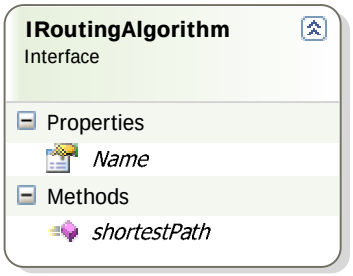
Det bibliotek som indlæses, undersøges vha. .NET reflection som vist nedenfor:

```
string filename = dlg.FileName;
Assembly objAssembly = Assembly.LoadFrom(filename);

foreach (Type t in objAssembly.GetTypes())
{
    try
    {
        if (Activator.CreateInstance(t) is IRFIDReader)
        {
            IRFIDReader rd = (IRFIDReader)Activator.CreateInstance(t);
            ListViewItem itmp = new ListViewItem(rd.Name);
            ListViewItem.ListViewSubItem sub1 = new
ListViewItem.ListViewSubItem(itmp, t.FullName);
            ListViewItem.ListViewSubItem sub2 = new
ListViewItem.ListViewSubItem(itmp, filename);
            itmp.SubItems.Add(sub1);
            itmp.SubItems.Add(sub2);

            listViewReaders.Items.Add(itmp);
        }
    }
    catch { }
}
```

Algoritmen vælges på samme måde, men fra "Algorithms" fanebladet:

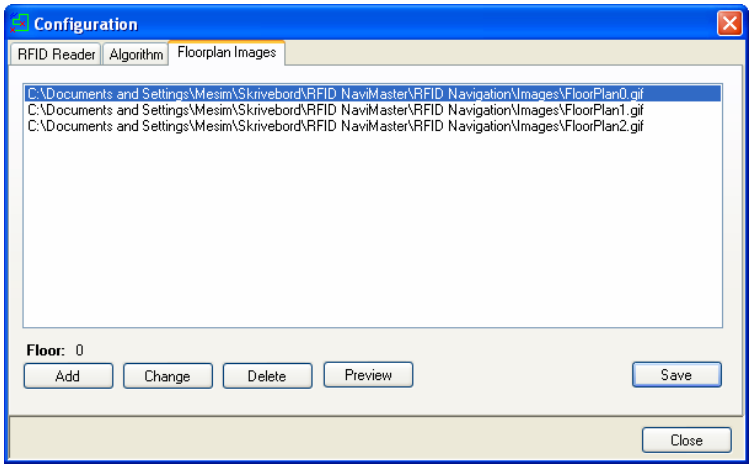


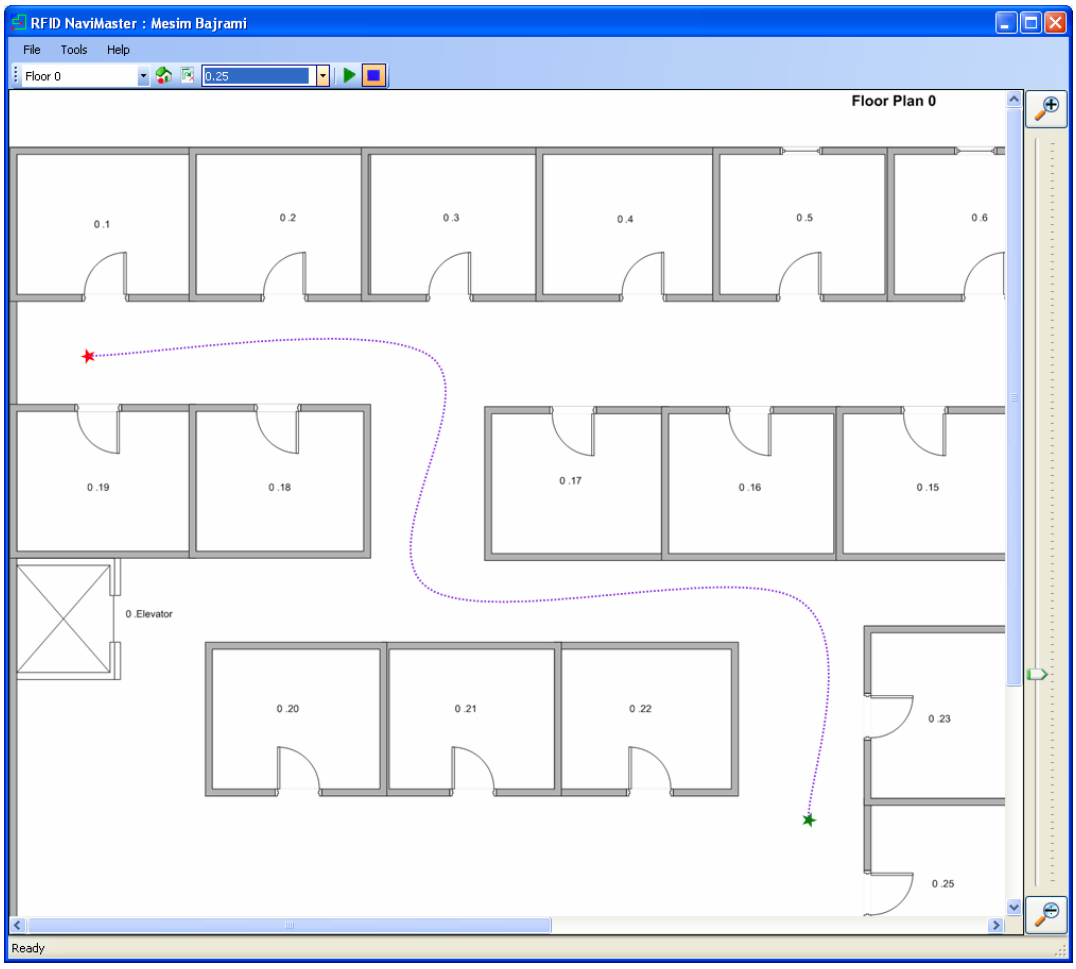
Algoritme biblioteket undersøges om den opfylder




```
string filename = dlg.FileName;
Assembly objAssembly = Assembly.LoadFrom(filename);

foreach (Type t in objAssembly.GetTypes())
{
    try
    {
        if (Activator.CreateInstance(t) is IRoutingAlgorithm)
        {
            IRoutingAlgorithm al =
(IRoutingAlgorithm)Activator.CreateInstance(t);
            ListViewItem itmp = new ListViewItem(al.Name);
            ListViewItem.ListViewSubItem sub1 = new
ListViewItem.ListViewSubItem(itmp, t.FullName);
            ListViewItem.ListViewSubItem sub2 = new
ListViewItem.ListViewSubItem(itmp, filename);
            itmp.SubItems.Add(sub1);
            itmp.SubItems.Add(sub2);

            listViewAlgorithms.Items.Add(itmp);
        }
    }
    catch { }
}
```

Det er også muligt at konfigurere bygningen etager. Antallet af etager afhænger af plantegninger som ligger i systemet. Det er muligt at tilføje og slette plantegninger fra systemet, og disse plantegninger bliver vist som baggrund for det digitale kort under navigationen.





10 BILAG B – RFID Standarder

Kilde: http://rfid-handbook.de/rfid/standardization.html

- 26. BImSchV: „Sechsundzwanzigste Verordnung zur Durchführung des Bundes-Immissionsschutzgesetzes – Verordnung über elektromagnetische Felder“, mit Erläuterungsteil; in: Wolfgang Kemmer, „Die neue *Elektro-smog-Verordnung*“, H. Hoffmann GmbH Verlag, Berlin 1997, ISBN 3-87344-103-9.
- AIAG ARF-1 „Application Standard for RFID Devices in the Automotive Industry“.
- AIAG B-11 „Tire and Wheel Identification Label Standard“
- ANSI/INCITS 256 „Radio Frequency Identification (RFID)“, NCITS 256 defines a standard for Radio Frequency Identification (RFID) for use in item management. This standard is intended to allow for compatibility and to encourage interoperability of products for the growing RFID market in the United States.
- ANS/INCITS 371: „Information Technology – Real Time Locating Systems (RTLS).“
    - Part-1: 2.4 GHz Air Interface Protocol
    - Part-2: 433 MHz Air Interface Protocol
    - Part-3: Application Programming Interface
- ANSI/MH 10.8.4 „RFID for Returnable Containers.“
- AWWA IMT61457 „The Use of Mobile and RFID Data and Field Force Integration in a Major Water Utility“
- CEPT T/R 60-01: „Low-power radiolocation equipment for detecting movement and for alert“ (*EAS*). Technical Recommendation. http://www.ero.dk
- CEPT T/R 22-04: „Harmonisation of frequency bands for Road Transport Information Systems (*RTI*)“ (Mautsysteme, Frachtidentifikation). Technical Recommendation. http://www.ero.dk
- ECMA-340: siehe ISO/IEC 18092 (NFCIP-1)
- ECMA-352: siehe ISO/IEC 21481 (NFCIP-2)
- ECMA-356: siehe ISO/IEC 22536 (NFCIP-1; RF Interface Test Methods)
- ECMA-362 siehe ISO/IEC 23917 (NFCIP-2; Protocol Test Methods for NFC)
- EN 50061: „Sicherheit implantierbarer *Herzschrittmacher*“. Vorschriften zum Schutz vor Fehlfunktion durch elektromagnetische Beeinflussung (entspricht VDE 0750). http://www.etsi.org
- EN 300 220: „Electromagnetic compatibility and Radio spectrum Matters (ERM); Short Range Devices (*SRD*); Radio equipment to be used in the 25 MHz to 1 000 MHz frequency range with power levels ranging up to 500 mW;“ http://www.etsi.org
- Part-1: Technical characteristics and test methods
    - Part-2: Supplementary parameters not intended for conformity purposes
    - Part-3: Harmonized EN covering essential requirements under article 3.2 of the R&TTE Directive
- EN 300 330: „Electromagnetic compatibility and Radio spectrum Matters (ERM); Short Range Devices (*SRD*); Radio equipment in the frequency range 9 kHz to 25 MHz and inductive loop systems in the frequency range 9 kHz to 30 MHz“; http://www.etsi.org
    - Part-1: Technical characteristics and test methods
    - Part-2: Harmonized EN under article 3.2 of the R&TTE Directive
- EN 300 440: „Radio Equipment and Systems (RES); Short Range Devices, Technical characteristics and test methods for radio equipment to be used in the 1 GHz to 25 GHz frequency range with power levels ranging up to 500 mW“; http://www.etsi.org

- ETS 300 683: „Radio Equipment and Systems (RES); ElectroMagnetic Compatibility (EMC) standard for Short Range Devices (*SRD*) operating on frequencies between 9 kHz and 25 GHz"; http://www.etsi.org
- EN 300 761: „Electromagnetic compatibility and Radio spectrum Matters (ERM); Short Range Devices (SRD); *Automatic Vehicle Identification* (AVI) for *railways* operating in the 2,45 GHz frequency range";
  http://www.etsi.org

Part-1: Technical characteristics and methods of measurement
Part-2: Harmonized standard covering essential requirements under article 3.2 of the R&TTE Directive

- EN 300 674: „Electromagnetic compatibility and radio spectrum matters (ERM); Road Transport and Traffic Telematics (RTTT); Technical characteristics and test methods for Dedicated Short Range Communications (DSRC) transmission equipment (500 kbit/s / 250 kbit/s) operating in the 5,8 GHz Industrial, Scientific and Medical (ISM) band";
  http://www.etsi.org
- EN 301 489: „Electromagnetic compatibility and radio spectrum matters (ERM); ElectroMagnetic Compatibility (EMC) standard for radio equipment and services";
  http://www.etsi.org
    - Part-1: Common technical requirements
    - Part-2: Specific requirements for radio paging equipement
    - Part-3: Specific requirements for Short Range Devices (SRD) operating on frequencies between 9 kHz and 25 GHz
    - Part-4: Specific requirements for fixed radio links and ancillary equipment and services
    - Part-5: Specific requirements for Private and Mobile Radio (PMR) and ancillary equipment (speech and non-speech)
    - Part-6: Specific conditions for Digital Enhanced Cordless Telecommunications (DECT) equipment
    - Part-7: Specific conditions for mobile and portable radio and ancillary equipment of digital cellular radio telecommunications systems (GSM and DCS)
    - Part-8: Specific requirements for GSM base stations
    - Part-9: Specific conditions for wireless microphones and similar Radio Frequency (RF) audio link equipment
    - Part-10: Specific conditions for First (CT1 and CT1+) and Second Generation Cordless Telephone (CT2) equipment
    - Part-11: Specific conditions for FM broadcasting transmitters
    - Part-12: Specific conditions for Earth Stations operated in the frequency ranges between 4 GHz and 30 GHz in the Fixed Satellite Service (FSS)
    - Part-13: Specific conditions for Citizens' Band (CB) radio and ancillary equipment (speech and non-speech)
    - Part-15: Specific conditions for commercially available amateur radio equipment
    - Part-16: Specific conditions for analogue cellular radio communications equipment, mobile and portable
    - Part-17: Specific requirements for Wideband data and HIPERLAN
    - Part-18: Specific requirements for Terrestrial Trunked Radio (TETRA)
    - Part-19: Specific conditions for Receive Only Mobile Earth Stations (ROMES) operating in the 1,5 GHz band providing data communications
    - Part-20: Specific conditions for Mobile Earth Stations (MES) used in the Mobile Satellite Services (MSS)
    - Part-22: Specific requirements for VHF aeronautical mobile and fixed radios
- ERC/DEC 92-02: „CEPT/ERC Decision on the frequency bands to be designated for the coordinated introduction of Road Transport Telematic Systems";
  http://www.ero.dk
- ERC/DEC 97-10: „CEPT/ERC Decision on the mutual recognition of conformity assessment procedures including marking of radio equipment and radio terminal equipment"; http://www.ero.dk

- ERC/DEC 01-01: „CEPT/ERC Decision: Non-specific short range devices in 6765 – 6795 kHz and 13.552 – 13.567 MHz“; http://www.ero.dk
- ERC/DEC 01-02: „CEPT/ERC Decision: Non-specific short range devices in 26.957 – 27.283 MHz“; http://www.ero.dk
- ERC/DEC 01-03: „CEPT/ERC Decision: Non-specific short range devices in 40.660 – 40.700 MHz“; http://www.ero.dk
- ERC/DEC 01-04: „CEPT/ERC Decision: Non-specific short range devices in 868.0 – 868.6 MHz, 868.7 – 869 .2 MHz, 869.4 – 869.65 MHz, 869.7 – 870.0 MHz“; http://www.ero.dk
- ERC/DEC 01-05: „CEPT/ERC Decision: Non-specific short range devices in 2400 – 2483.5 MHz“; http://www.ero.dk
- ERC/DEC 01-13: „CEPT/ERC Decision: Short range devices for inductive applications in 9 – 59,750 kHz, 59 .750 – 60.250 kHz, 60.250 – 70 kHz, 70 – 119 kHz and 119 – 135 kHz“; http://www.ero.dk
- ERC/DEC 01-14: „CEPT/ERC Decision: Short range devices for inductive applications in 6765 – 6795 kHz, 13,553 – 13.567 MHz“; http://www.ero.dk
- ERC/DEC 01-15: „CEPT/ERC Decision: Short range devices for inductive applications in 7400 – 8800 kHz“; http://www.ero.dk
- ERC/DEC 01-16: „CEPT/ERC Decision: Short range devices for inductive applications in 26.957 – 27.283 MHz“; http://www.ero.dk
- ERC/REC 01-06: „CEPT/ERC Recommendation: Procedure for mutual recognition of type testing and type-approval for radio equipment“; http://www.ero.dk
- ERC/REC 70-03: „CEPT/ERC Recommendation 70-03 relating to the use of Short Range Devices (*SRD*)“; http://www.ero.dk
- ETSI TS 102 190: siehe ISO/IEC 18092 (NFCIP-1)
- ETSI TS 102 312: siehe ISO/IEC 21481 (NFCIP-2)
- ETSI TS 102 345: siehe ISO/IEC 22536 (NFCIP-1; RF Interface Test Methods)
- ISO/IEC 6346 „Freight containers – Coding, identification and marking“
- ISO/IEC 7810: „Identification cards – Physical characteristics“
- ISO/IEC 7816: „Identification cards – Integrated circuit(s) cards with contacts“
    - Part-1: Physical characteristics
    - Part-2: Dimensions and location of the contacts
    - Part-3: Electronic signals and transmission protocols
    - Part-4: Interindustry commands for interchange
    - Part-5: Registration system for applications in IC Cards
    - Part-6: Interindustry Data Elements
    - Part-7: Interindustry commands for Structured Card Query Language (SCQL)
    - Part-8: Security architecture and related interindustry commands
    - Part-9: Enhanced interindustry commands
    - Part-10: Electronic signals and answer to reset for synchronous cards
    - Part-11: Card structure and enhanced functions for multi-application use
    - Part-12: Cryptographic information application
- ISO/IEC 8824-1: „Information technology – Abstract Syntax Notation One (ASN.1) – Specification of basic notation.“
- ISO/IEC 8825-1: „Information technology – ASN.1 encoding rules – Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER).
- ISO/IEC 9798: „Information technology – Security techniques – Entity authentication“. Grundlagen und Beschreibung von Authentifizierungsverfahren.
    - Part-1: General
    - Part-2: Mechanisms using symmetric encipherment algorithms
    - Part-3: Mechanisms using digital signature techniques
    - Part-4: Mechanisms using a cryptographic check functions
    - Part-5: Mechanisms using zero knowledge techniques

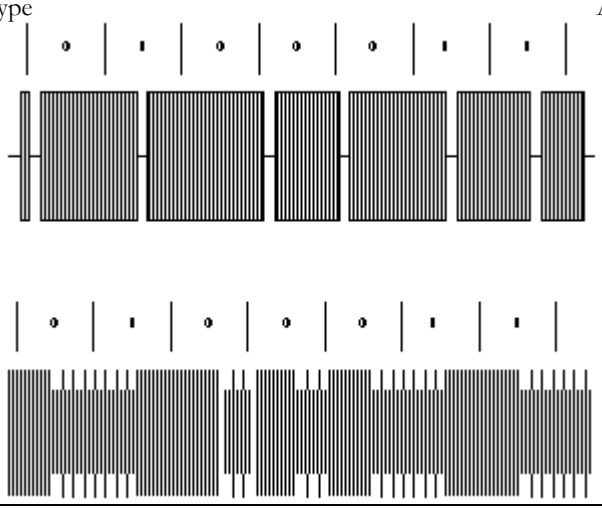
- ISO/IEC 9834-1: 1993/Amd.2: 1988 „Information technology – Open Systems Interconnection – Procedures for the operation of OSI Registration Authorities: General Procedures."
- ISO/IEC 10373: „Identification Cards – Test methods". Prüfmethoden für „Plastikkärtchen"; zum Prüfen des Kartenkörpers und der eingebauten Kartenelemente (Magnetstreifen, Halbleiterchips). Die Norm besteht aus folgen-den Teilen:
  - Part-1: General
  - Part-2: Magnetic strip technologies
  - Part-3: Integrated circuit cards (kontaktbehaftete Chipkarten)
  - Part-4: Contactless integrated circuit cards (close-coupling)
  - Part-5: Optical memory cards
  - Part-6: Proximity cards (kontaktlose Chipkarten nach ISO/IEC 14443)
  - Part-7: Vicinity cards (kontaktlose Chipkarten nch ISO/IEC 15693)
- ISO/IEC 10374: „Container – Automatische Identifizierung" („*Freight containers – Automatic identification*"). Automatische Identifizierung von Fracht-Containern durch ein 2,45 GHz-Transpondersystem.
- ISO/IEC 10536: „Identification cards – Contactless integrated circuit(s) cards". Kontaktlose Chipkarten in Close-coupling-Technologie. Die Norm besteht aus folgenden Teilen:
  - Part-1: Physical characteristics
  - Part-2: Dimensions and location of coupling areas
  - Part-3: Electronic signals and reset procedures
  - Part-4: Answer to reset and transmission protocols
- ISO/IEC 11784: „Radio-frequency *identification of animals* – code structure"; Identifizierung von Tieren durch RFID-Systeme. Beschreibung der Datenstruktur.
- ISO/IEC 11785: „Radio-frequency *identification of animals* – technical concept"; Identifizierung von Tieren durch RFID-Systeme. Beschreibung der RF-Übertragungsverfahren.
- ISO/IEC 14223: „Radio-frequency *identification of animals* – Advanced Transponders":
  - Part-1: Air Interface
  - Part-2: Code and command structure
- ISO/IEC 14443: „Identification cards – Proximity integrated circuit(s) cards":
  - Part-1: Physical characteristics
  - Part-2: Radio frequency interface
  - Part-3: Initialization and anticollision
  - Part-4: Transmission protocols
- ISO/IEC 14816: „Road Traffic and Transport Telematics – Automatic Vehicle and Equipment Identification – Numbering and Data Structures"
- ISO/IEC 15459: „Information technology – Automatic identification and data capture techniques – Unique identifiers for item management"
  - Part-1: Unique identification of transport units
  - Part-2: Registration procedures
  - Part-3: Common rules for unique identification
  - Part-4: Unique item identification for supply chain management
  - Part-5: Uniqe identification of returnable transport items (RTIs)
  - Part-6: Unique identification for product groupings in material lifecyle management
- ISO/IEC 15693: „Identification cards – contactless integrated circuit(s) cards – Vicinity Cards"
  - Part-1: Physical characteristics
  - Part-2: Air interface and initialisation
  - Part-3: Protocols
  - Part-4: Registration of Applications/issuers
- ISO/IEC 15961: „Information technology – RFID for *Item Management* – Data protocol: application interface".

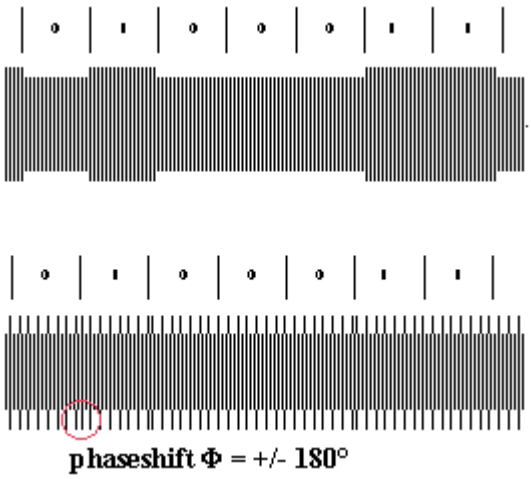
- ISO/IEC 15962: „Information technology – RFID for *Item Management* – Data protocol: data encoding rules and logical memory functions“.
- ISO/IEC 15963: „Unique Identification of RF tag and Registration Authority to manage the uniqueness“.
    - Part-1: Numbering System
    - Part-2: Procedural Standard
    - Part-3: Use of the unique identification of RF tag in the integrated circuit
- ISO/IEC 17358: „Supply chain application for RFID – Application requirements“.
- ISO/IEC 17363: „Supply chain application for RFID – Freight containers“.
- ISO/IEC 17364: „Supply chain application for RFID – Transport units“.
- ISO/IEC 17365: „Supply chain application for RFID – Returnable transport items“,
- ISO/IEC 17366: „Supply chain application for RFID – Product packaging“.
- ISO/IEC 17367: „Supply chain application for RFID – Product tagging“.
- ISO/IEC 18000: „RFID for *Item Management*: Air Interface“.
    - Part-1: Generic Parameter for Air Interface Communication for Globally Accepted Frequencies
    - Part-2: Parameters for Air Interface Communication below 135 kHz
    - Part-3: Parameters for Air Interface Communication at 13.56 MHz
    - Part-4: Parameters for Air Interface Communication at 2.45 GHz
    - Part-5: Parameters for Air Interface Communication at 5.8 GHz
    - Part-6: Parameters for Air Interface Communication - UHF Frequency Band (868 / 915 MHz)
- ISO/IEC 18001: „Information technology – Radio frequency identification for item management – Application requirements profiles“.
- ISO/IEC 18046: „RFID Tag and Interrogator Performance Test Methods“.
- ISO/IEC 18047: „Information technology – Radio frequency identification device conformance test methods“. Testmethoden für ISO/IEC 18000
    - Part-3: Test methods for air interface communications at 13.56 MHz
    - Part-4: Test methods for air interface communications at 2.45 GHz
    - Part-7: Test methods for air interface communications at 433 MHz
- ISO/IEC 18092: „Near Field Communication (NFC) Interface and Protocol-1 (NFCIP-1).“
- ISO/IEC 18185: „Freight containers – Radio frequency communication protocol for electronic seal“.
    - Part-1: Communication protocol
    - Part-2: Application requirements
    - Part-3: Environmental characteristics
    - Part-4: Data protection
    - Part-5: Sensor interface
    - Part-6: Message sets for transfer btw. seal reader and host computer
    - Part-7: Physical layer
- ISO/IEC 19762: „Information technology AIDC techniques – Harmonized vocabulary“.
    - Part-1: General terms relating to Automatic Identification and Data Capture (AIDC).
    - Part-2: Optically readable media (ORM).
    - Part-3: Radio frequency identification (RFID).
- ISO/IEC 21007: „Gas Cylinders – Identification and Marking Using Radio Frequency Identification Technology“
    - Part-1: Reference Architecture and Terminology
    - Part-2: Numbering Schemes for Radio Frequency
- ISO/IEC 21481: „Near Field Communication (NFC) Interface and Protocol-2 (NFCIP-2).“
- ISO/IEC 22536: „Near Field Communication (NFC) Interface and Protocol-1 (NFCIP-1); RF Interface Test Methods.“
- ISO/IEC 23389: „Freight containers – read write radio frequency identification (RFID)“.

- ISO/IEC 23917: „Near Field Communication (NFC) Interface and Protocol-2 (NFCIP-2); Protocol Test Methods for NFC."
- ISO/IEC 24710: „Information technology AIDC techniques – RFID for Item Management – ISO/IEC 18000 Air Interface Communications – Elementary Tag license-plate functionality for ISO/IEC 18000 air interface definitions".
- ISO/IEC 24729: „Information technology – Radio frequency identification for item management – Implementation guidelines."
    - Part-1: RFID-enabled labels and packaging
    - Part-2: Recyclability of RF tags
    - Part-3: RFID interrogator/antenna installation
- ISO 69873: „Werkzeuge und Spannzeuge mit Datenträgern – Maße für Datenträger und deren Einbauraum".
- S-918-00: AAR Manual of Standards and Recommended Practices Railway Electronics, S-918: „Standard for Automatic Equipment Identification" Adopted: 1991; Revised: 1995, 2000
- VDE 0848: „Sicherheit in elektromagnetischen Feldern" (Teil 2 – Schutz von Personen im Frequenzbereich 30 kHz bis 300 GHz, Teil 4A2 – Schutz von Personen im Frequenzbereich 0 Hz bis 30 kHz)
- VDE 0750: Siehe EN 50061
- VDI 4470 - Teil 1: „*Warensicherungssysteme* – Kundenabnahmerichtlinie für Schleusensysteme"; Ermittlung der Erkennungsrate und Detektionsrate bei der Inbetriebnahme von EAS-Systemen vor Ort.
- VDI 4470 - Teil 2: „*Warensicherungssysteme* – Kundenabnahmerichtlinie für Deaktivierungsanlagen"; Prüfung von Deaktivierungsanlagen für EAS-Systeme.
- (August 2006)

ISO 14443 (ISO SC17/WG8) - proximity cards

Part 1: Physical Characteristics
Part 2: Radio frequency power and signal interface
Part 3: Initialization and anticollision
Part 4: Transmission protocol

Common parameters	Power Supply: 13,56 MHz, inductive coupling Field strength: 1,5 .. 7,5 A/m
Type A: 	Downlink: ASK 100%, modified Miller Code, 106 kBit/s Uplink: Loadmodulation with 847 kHz Subcarrier ASK-modulated, Manchester Code, 106 kBit/s Anticollision: Binary search tree

Type B:  	Downlink: ASK 10%, NRZ Code, 106 kBit/s Uplink: Loadmodulation with 847 kHz Sucarrierer BPSK (bi-phase shift keying) modulated, NRZ Code, 106 kBit/s Anticollision: Slotted Aloha

ISO 15693 (ISO SC17/WG8) - vicinity cards

Part 1: Physical Characteristics

Part 2: Radio frequency power and signal interface

Part 3: Anticollision and transmission protocol

Power supply:	13,56 MHz inductive coupling, 0,15 .. 7,5 A/m
Downlink (data transmission reader > card)	Modulation: 10% ASK, 100% ASK (card supports both) Bit coding: “1 out of 256”, “1 out of 4” (card supports both) Baud rate: 1.65 kBit/s, 26.48 kBit/s (card supports both)
Uplink (data transmission card > reader)	Modulation: Load modulation with subcarrier Bit coding: Manchester, the subcarrier is ASK (423 kHz) or FSK (423/485 kHz) modulated Baud rate: 6.62 kBit/s, 26.48 kBit/s (selected by reader)