UNIVERSITETET I OSLO
Institutt for informatikk

INF5261 Prosjekt

Dataregistrering på PDA

Midtveisrapport

Omar Alvin Pettersen

Petter Bøckmann

Haakon Eikenes

Thomas Halvorsen

Ingar Vindenes

6 April 2005

Innholdsfortegnelse

Figurliste

1 Introduksjon

1.1 Innledning

I faget utvikling av mobile informasjonssystemer våren 2005 fikk vi, studentene, i oppgave å lære oss selv, de andre studentene og veileder noe innenfor dette emnet. Dette prosjektet skulle gjerne være knyttet opp mot en bedrift eller organisasjon. Vi har valgt å se på forskernes situasjon på Ullevål Universitetssykehus hvor deres informasjonssystem består av permer fulle av skjemaer og en enkel database. Disse forskerne driver i dag en av landets største undersøkelse innen psykiatri og det er store mengder data å holde styr på. Vi tror at et nytt IKT-system er det de trenger for at de skal få bedre oversikt og at vi kommer til å lære noe.

Mobilitet kan være et subjektivt begrep. Noen synes kanskje en laptop er mobil, mens andre [Hjelm] mener at slik er det ikke. Han sier at for at en enhet skal være mobil, og kunne følge en person, må den være liten nok til å passe i lommen og kunne brukes med en hånd, mens man gjør noe annet. Må en enhet også ha mulighet for å koble seg mot telenett eller internett for å kunne kalles mobil? I dette prosjektet ser vi på PDA'ens (mer om PDA senere i rapporten) muligheter for å være en mobil informasjonsinnsamlingsenhet. En PDA passer kanskje i en lomme, men man trenger iallefall begge hender for å betjene den og er kanskje ikke mobil i alles øyne. Vi mener likevel at vi kan forsvare at en PDA er mobil.

Vi registrerer at helsesektoren i disse dager driver en massiv fornyelse der de går over til IKT-basert systemer i enda større grad enn før og at det blir tatt i bruk håndholdte enheter til for eksempel elektroniske pasientjournaler. Vi opplever derfor at prosjektet vårt er dagsaktuelt.

Rapporten er bygget opp slik at vi begynner med problemstilling og spørsmålene vi vil ha svar på i første kapittel. Vi har også sett på en lignende prosjekt og vurdert dette opp i mot vårt prosjekt. Deretter følger tekniske detaljer og fakta om løsningen, som vi også ønsker å lage en prototype på i kapittel 2. I kapittel 3 drøfter vi hvordan vi kan evaluere løsningen og om den har innvirkning på pasient/helsepersonell forholdet. Mange av kildene våre har vi funnet på internett og vi har derfor skrevet litt om kildevurdering i kapittel 4. Når denne midtveisrapporten skrives har vi ennå ikke kommet til noen konklusjon, men dette faller mer naturlig inn i sluttrapporten. Sist i vedlegget ligger vår fremdriftsplan.

Omar A. Pettersen er, i tillegg til å være student, tilknyttet forskningsavdelingen ved Ullevål Universitetssykehus som prosjektet er koblet opp mot. Han har derfor en dobbeltrolle i prosjektet ved å være både student og "arbeidsgiver".

1.2 Problemstilling

Det er ønsket av arbeidsgiver, Ullevål Universitetssykehus, at permene blir byttet ut med en håndholdt PC, eller PDA, hvor svarene fra undersøkelsene blir registrert elektronisk under selve undersøkelsen. Etter endt undersøkelse tar forskeren med seg PDA'en tilbake til sykehuset hvor svarene enkelt overføres til databasen. Det blir dermed mulig å utføre spørringer mot de samme dataene uten å måtte vente til de blir tastet inn manuelt. Statistikker blir dermed også oppdatert raskere.

1.2.1 Problem- og anvendelsesområde

Dersom vi ser på dagens situasjon, dvs. bruk av papirbasert skjema, i forhold til vår visjon vil sannsynligvis verken problemområdet eller anvendelsesområdet [Mathiassen m. fl.] bli forandret. Hvis vi ser på dagens problemområde er det:

> - selve forskningsdataene som blir til når forskerne foretar intervjuene utenfor forskningskontoret, forskningsdataene når de blir arbeidet med i etterkant og generert statistikk.

Og anvendelsesområdet:

> - forskeren ute på pasientbesøk og på kontoret når de bruker de innsamlede dataene og generert statistikk.

En innføring av Personal Digital Assistent (PDA) vil ikke ha noen påvirkning av disse områdene. Det vil fortsatt være forskerne som besøker pasientene på institusjoner og lignende. Forskerne vil fortsatt arbeide med dataene og generert statistikk.

Problemene vi fokuser på ved dette systemet vil i hovedsak være: bruk av papir i motsetning til PDA med hensyn på brukergrensesnitt, sikkerhet og bruk av PDA blant forskerne.

1.2.2 Bruk av papir i motsetning til PDA

Dette vil være et omfattende problemområde i forhold til oppgaven. Nettopp fordi det er dagens papirbaserte spørreskjema som eventuelt skal byttes ut med en PDA løsning. Dagens situasjon består av at forskeren har med seg en perm, dvs. et spørreskjema per pasient. Noe som resulterer i at det blir en del permer i løpet av uken forskeren er på "intervjurunde", hvor mobilt er da 10 permer á 1 kilo?

Videre vil vi se på hvordan forskerne og pasientene forholder seg til bruk av IT-teknologi. Vil forskerne ha problemer med bruk av PDA under intervjuet? Hvordan forholder pasientene seg til en PDA fremfor et godt, gammeldags papirskjema?

Et annet moment er, hvorfor velge nettopp PDA i denne settingen? Hva skjer dersom en PDA går tom for batteri? Hvordan kan vi bruke papirflyten til å lage et bedre digitalt system?

1.2.3 Brukegrensesnitt

Flere av forskerne som skal håndtere PDA'en har liten eller ingen innsikt i tidligere PDA bruk. Utfordringen her vil da være forskerne som skal bruke systemet, og begrensingene på selve PDA'ene. En PDA i seg selv har begrensninger i forhold til skjermstørrelse og tastaturbruk.

Hvordan bør systemet organiseres på en PDA for at forskerne lettest mulig skal forstå spørreskjemaet? Hvordan kan forskerne skrive inn kommentarer og lengre tekster uten at det tar for lang tid?

1.2.4 Sikkerhet

Det ligger i denne avdelingens natur at data er sensitive. Det viktigste blir her hvordan vi skal sikre sensitive data som ligger lagret på en PDA. Det vil ikke være noen trådløs forbindelse til disse PDA'ene, dette fordi sikkerheten rundt dette prosjektet er så essensiell.

Ved innføring av en digital løsning vil de nåværende arkivskapene forsvinne, som betyr at den papirbaserte "sikkerhetskopien" vil forsvinne. Hvordan vil dette påvirke den nåværende løsningen for sikkerhetskopiering?

Videre vil selve PDA'en, som digitalt medium, være mer attraktiv å stjele enn, en perm med papir. Altså, hvordan sikre en PDA kontra papir?

1.2.5 Koding

Ved å velge en digital løsning tror vi at vi kan løse de tidligere beskrevne problemene bedre enn den nåværende papirbaserte løsningen. Dette stiller igjen nye problemer i forhold til den digitale løsningen. Vil skal derfor se på hvordan kvalitetssikre koden på best mulig måte? Hvordan håndtere inkonsistente data, i forhold til databasen?

[avgrens kode spørsmål siden vi kun har prototyp?]

1.3 PDA historie

Begrepet Personal Digital Assistent (PDA) ble opprettet 7. januar 1992 på Consumer Electronics Show i Las Vegas, Nevada [WikiPDA]. Allerede året etter, i 1993 kom Apple Computer Inc. med verdens første PDA kalt "Newton" [Handango]. Håpet var at PDA'en skulle holde rede på telefonnumre, kalender med avtaler, notater og i tillegg sende og motta data trådløst. I mars 1996 var Palm på markedet med den første ekte håndholdte datamaskinen, PalmPilot.

I den senere tid har flere aktører kommet på markedet som BlackBerry, HP iPAQ Pocket PC (Opprinnelig Compaq iPAQ inntil HP og Compaq slo seg sammen i 2002), Nokia Communicator og Dell's Axim, som vi har benyttet i vårt prosjekt. [WikiPDA].

Det finnes også en rekke hybrider mellom telefoner og PDA, de såkalte "smart phones".

1.3.1 Utvikling

De senere årene har det skjedd stor utvikling på PDA fronten. Etter hvert som markedet har økt har nye funksjoner blitt lagt til PDA'ene som fargeskjerm, touch screen, mer minne og raskere prosessor. Dessuten har flere trådløse grensesnitt blitt lagt til som infrarødt (IR), "Bluetooth", trådløst nettverk (WLAN), 2G og 3G.

Per i dag er PDA markedet delt i to: PDA'er som kjører Palm OS og de som kjører Pocket PC. Vi vil konsentrere oss om PDA'er som kjører Pocket PC da det er denne løsningen vi har valgt i vårt prosjekt.

1.3.2 Prosessor

I tråd med utviklingen har også PDA'ens prosessorkraft økt betraktelig. Dette har medført til at mer ressurskrevende programmer kan kjøres. Dessuten har minne (RAM) tilgjengelig for brukeren økt kraftig noe som har resultert i at større filer og programmer kan lastes og kjøres.

Det er også nyttig å merke seg at forskjellige minnekort er støttet av PDA – noe som betyr at dataene ikke blir borte ved strømtap (flatt batteri?).

1.3.3 OS

Innen Pocket PC verdenen har det kommer flere versjoner av Pocket PC. Her har Microsoft kommet med forskjellige versjoner fra den første Pocket PC, Windows CE 3.0, Windows Mobile 2002/2003 og Smartphone 2002/2003. Alternative OS er Palm OS (som kjører på Palm), BlackBerry og Linux for å nevne de andre mest populære alternativene.

Siden OS er lagt i et ikke skrivbart minne (ROM) så er det vanskelig å skifte ut, og av samme grunn vanskeligere å legge til nye drivere.

1.3.4 Vår PDA

I prosjektet har vi benyttet oss av:

Dell AXIM X50
Windows Mobile™ 2003 Second Edition (Build 1460.2.0.5)
Prosessor: Intel® PXA270 520MHz
Minne: 64 MB
Språk: Engelsk

Denne PDA'en har forhåndsinstallert Windows Mobile 2003 SE, så å benytte dette operativsystemet var et naturlig valg.

1.4 Andre prosjekter

1.4.1 Medicom Patient-Response-Module (PRM)

Et av hovedprosjektene på Høgskolen i Oslo våren 2003 synes vi det var verdt å skrive om i rapporten, nemlig ”Medicom Patient-Response-Module” [Medicom]. Dette prosjektet har mange likhetstrekk med vårt eget prosjekt. Slik vi forstår det har studentgruppen på Høgskolen fått en relativt klar kravspesifikasjon og hovedfokuset har ligget på implementering og ”deployment”. Vi jobber ikke ut ifra en kravspesifikasjon men heller ut ifra en oppgave om å forbedre en situasjon. Hvilke komponenter systemet bør bestå av og hvilken effekt disse får på situasjonen er vi veldig kritiske til.

	Vårt prosjekt ”Dataregistrering på PDA”	Medicom Patient-Respons-Module
Oppdragsgiver	Forskningsavdelingen ved Ullevål universitetssykehus Vil forbedre intervjuarbeid ved hjelp av PDA.	Medicom A/S Firma som utvikler og produserer tekniske hjelpemidler for helsevesenet.
Prosjektbeskrivelse	Utvikle en datainnsamlingsenhet	Utvikle en datainnsamlingsenhet
Hovedbrukere (hovedforskjell)	Forskere	Pasienter Helsepersonell
Hovedfunksjoner	Forskere skal kunne foreta intervju med forhåndsdefinerte spørreskjema. Enkel og sikker overføring av data til server. Innsamlet data skal kunne administreres.	Pasienter skal selv kunne besvare spørreundersøkelser. Det skal kunne lages spørreskjemaer på vanlige PC’er med internett tilgang Innsamlet data skal kunne administreres.
Utviklings plattform	PDA .NET compact framework MS SQL 2000 C# UML	PDA .NET compact framework MS SQL 2000 C#, ASP.NET UML CVS Nunit og MS ACT
Bruksområde (fysisk)	Behandlingssteder (sykehus, poliklinikker o.l.)	Privat (hjemme hos pasient og lege) Sykehus
Motivasjon	Redusere papirarbeid for administrativt personell. Redusere feilkilder. Bedre og raskere oversikt over data.	Redusere papirarbeid for administrativt personale. Ønske om daglig/hyppige informasjonsinnsamling, noe som kan gi bedre oppfølging. Direkte tilgang til pasientdata

1.4.2 Vurdering av PRM

Hovedforskjellen i de to produktene (vårt slikt vi tenker oss det ferdig og PRM) ligger i hvem som bruker det, ikke bruksmåte eller funksjoner, som er tilnærmet like. Ved bruk av begge systemene dannes først spørreskjema på en arbeidsstasjon og overføres til PDA. PDA'en brukes til å samle inn svar på skjemaene. Svardata blir overført til en database og brukt av helsepersonell. Vårt tenkte produkt brukes av forskere for å foreta intervju av mange pasienter (og hver pasient kun opptil 5 ganger) og i etterkant forske på dataene, mens PRM går ut på at pasienter selv svarer på samme/flere spørreskjema hyppig, opptil daglig, uten kontakt med helsepersonell, og disse dataene overvåkes/behandles fortløpende.

Produktene er så like at de nesten kunne vært brukt om hverandre, men ved forskningsavdelingen på Ullevål universitetssykehus har de ikke bestemt seg for å ta i bruk et nytt system. De trenger en forbedring eller omgjøring av dagens system. Ved å la en studentgruppe prøve å utvikle eller finne en eller flere alternative løsninger på problemet kan de få en gratis og sannsynligvis god vurdering. Spesielt med tanke på det den tekniske delen. I tillegg kan de få muligheten til å prøve en prototyp i praksis.

1.4.3 Konklusjon

Medicom sitt produkt dekker behovene til forskerne slik vi har forstått dem og dersom det er innenfor de økonomiske rammene til avdelingen er Medicom PRM et alternativ vi kan formidle videre.

2 Valgte løsninger

2.1 Teknologi

Vi ønsker å utvikle et system hvor forskere kan utføre oppgavene sine ved hjelp av en PDA og måtte derfor velge en utviklingsplattform som gav god støttet for dette. Valget falt på .NET compact framework. Denne plattformen er kun støttet av Windows CE, et av de to mest utbredte PDA operativsystemene, hvorav Palm OS er det andre. Windows CE er det mest fleksible av disse to. Jason Perlow [Perlow] sammenligner i artikkelen ”Comparing Windows CE with Palm OS” Windows CE med en joggedress og Palm OS med en dress. Han sier at Palm OS sitter skreddersydd til sine funksjoner og sitt bruk (mest almanakk/tidsplanlegging), mens Windows CE ikke nødvendigvis sitter like støpt til disse oppgaver, men er samtidig åpen for mange andre bruksmåter. Han gjør også et poeng ut av at enhetene med Palm OS ofte har svart/hvit skjerm, mens enhetene som kjører Windows CE har fargeskjermer. I og med at vi legger mye vekt på brukergrensesnittet støttes vårt valg av OS opp av dette poenget.

Valg av programmeringsspråk var enkelt. C# støtter .NET plattformen fullt ut og alle kunne en del java fra før, noe C# er til forveksling likt, dermed falt valget på C#.

Når det gjelder databasen vi ønsker å lagre dataene i var vi mer i tvil. MySQL, Access og MS SQL var alternativene. Grunnen til at andre alternativer ikke ble vurdert er pris og at vi hadde trengt lengre tid på å sette oss inn i disse. MySQL falt bort fordi databaseserveren som brukes ved forskningsavdelingen er access basert, men mest fordi det ikke ligger noen støtte for MySQL innbakt i .NET plattformen.

Vi ble først enige om å bruke det eksisterende databaseverktøyet Access. Dette var mest i hensyn til at vi har begrenset tid på oppgaven og fordi vi kanskje kunne bruke noe av strukturen som allerede var laget. Men rett etter programmeringsstart valgt vi bort Access til fordel for MS SQL.

Access hadde nok kunnet dekket alle behov for oss siden vi kun skal lage en prototyp, men blir systemet tatt i bruk har SQL mange fordeler fremfor Access. MS SQL er mer rettet mot klient/server enn Access er, som er mer enn filserver. Dette kan bedre sikkerheten med bedre innloggingsrutiner og bedre støtte for flere brukere/PDA’er. Denne klient/server utførelsen bidrar også til økt effektivitet da den kun vil sende data du ber om hvis du kjører en spørring, der filbasert utførelsen til Access sender alle data i alle tabeller man ber om utdrag fra.

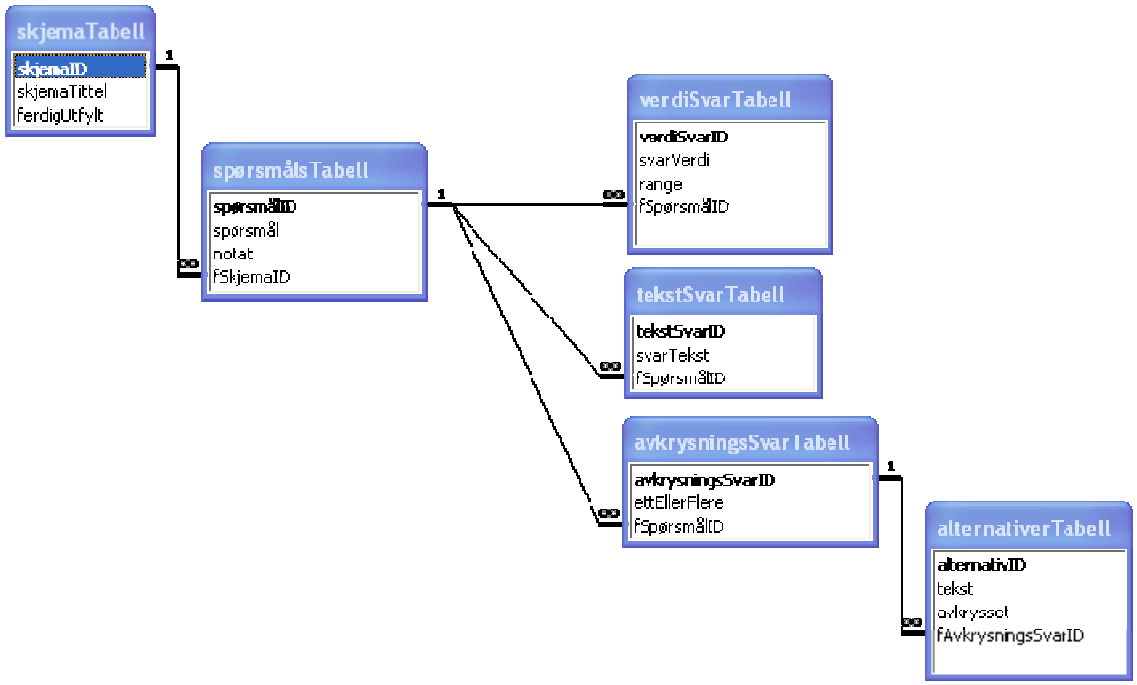
Når datamengden blir stor og man ikke sitter med databasen lokalt på egen arbeidsstasjon blir dette en trang flaskehals å komme gjennom og er det vanligste problemet med hensyn på ytelse. Når du jobber mot en MS SQL database blir det du leser ut eller skriver inn tatt hånd om av en ”intelligent” datastyrer, mens bruker du en MS Access database skriver du rett inn i selve databasefila [Aldex].

Dersom en overføring går galt, en nettverksfeil oppstår eller en klient kræsjer vil selve Access databasen (ikke kun data man holder på med) henge i en tynn tråd, uten mulighet for ”roll-back”. En MS SQL database skjønner at noe har gått galt dersom en overføring plutselig slutter og ikke starter igjen og kjører en ”roll-back” operasjon. ”Roll-back” vil si at data som er blitt endret nylig gjenopprettes til tidligere verdier. Microsoft sier selv at Access ikke egner seg til flerbrukermiljøer og at det bidrar til korrupsjon av data og ustabilitet[Aldex].

2.2 Vår løsning

[midlertidig avsnitt som presenterer grensesnitt og løsning]

Dette avsnittet innholder foreløpig kun to bilder. Det første viser oppbyggingen av databaseløsningen, mens det andre viser hvordan vi tenker oss at et spørsmål blir presentert på PDA’en.



Figur 1 Skjematisk oversikt av databaseløsningen.

Figur 2 Eksempel på spørsmål.

2.3 Lovverk

Etter å ha vært i kontakt med datatilsynet så anbefaler de kryptering av dataene med DES 128 eller bedre. Det står ikke spesifikke krav til tekniske løsninger i lovverket så det er derfor enda viktigere å dokumentere dette. Ellers refererer vi til lov om personopplysninger og forskrifter for personopplysinger (se vedlegg)

2.3.1 Dokumentasjon

Det er viktig at rutiner rundt bruken av systemet (her PDA) skal dokumenteres. Dette skal lagres i 5 år etter at gammel versjon er erstattet av ny (Se §2-19 [ForskriftPerson]).

2.3.2 Mulige løsninger

§2-10 er vanskelig å oppfylle da PDA'en ikke kan være innelåst til enhver tid.

§2-11 kan passordbeskyttelse og kryptering av data benyttes for å sikre innsyn av data. Vi kommer tilbake til måter å kryptere data på og valg av løsning.

§2-13 Det kan være ønskelig å opprette ny kopi av forrige brukte database når endringer lages – hvor den nye databasen blir utstyrt med timestamp og versjonsnummer slik at man kan gå tilbake og se historien til det enkelte spørreskjema. Vi tar opp dette i et eget scenario. Det vil være naturlig å passordbeskytte databasen som helhet eller de enkelte data som er lagret i basen.

§2-14 Innlogging på datautstyret må loggføres. Både vellykkede og ikke vellykkede innlogginger (mulig hacking) Loggen skal lagres i minst 3 måneder (se §2-16)..

I følge § 28 i ”Lov om behandling av personopplysninger (personopplysningsloven)” skal det ikke lagres personopplysninger lenger enn nødvendig – men åpner allikevel for å kunne lagre informasjonen for historiske, statistiske eller vitenskapelige formål.

2.4 Kryptering

Det er mulig å benytte seg av flere forskjellige krypteringsalgoritmer. Vi vil her gå igjennom noen av de mest vanlige måtene å kryptere på og i denne sammenheng belyse våre valg av krypteringsalgoritme.

Ifølge e-post fra datatilsynet var det naturlig å benytte en krypteringsalgoritme tilsvarende DES 128, og det er denne retningslinjen vi har fulgt når det gjelder valg av krypteringsalgoritme.

2.4.1 Nøkler

Ved kryptering benyttes nøkler for å kryptere / dekryptere dataene. Disse nøklene må på en eller annen måte være forbundet med hverandre både hos den som krypterer og dekrypterer. Man kan ha en felles nøkkel hos begge parter (symmetriske nøkler). Krypteringsalgoritmer som benytter denne typen nøkkel er som regel enklere og raskere enn de som benytter seg av offentlige og private nøkler.

Ved bruk av offentlige og private nøkler genereres nøkler hos sender og andre nøkler hos mottaker. Det ene settet kalles offentlig nøkkel og er nøkkelen som alle kjenner til og benytter for å kryptere data. Det andre settet med nøkler, private nøkkel benyttes for å dekryptere dataene når de har blitt mottatt [WebopediaPKC]. Algoritmene som benytter seg av Public keys krever mer resurser ved kryptering / dekryptering [WikiPubKey].

2.4.2 DES

Data Encryption Standard (DES) benytter seg av en enkelt symmetrisk nøkkel.
Krypteringen foregår ved at dataene som skal krypteres deles opp i forskjellige biter (blokker) som så krypteres vha. nøkkelen [WikiDes][Han m.fl]. Nøklene som benyttes er på bare 56 bit og kan knekkes raskt.

2.4.3 Trippel DES

Trippel DES (også kalt 3DES) er en modifikasjon av DES. Her benyttes det tre (2 effektive) symmetriske nøkler med en total lengde av 168 bit (3 DES nøkler). Dataene blir først kryptert med den første nøkkelen, så blir de krypterte dataene kryptert med den andre nøkkelen og til slutt kryptert med den første nøkkelen igjen. Siden den første og siste nøkkelen er lik så er den effektive nøkkellengden 112 bit [Wiki3DES]!

2.4.4 XTEA

XTEA (Extended Tiny Encryption Algorythm) bruker i likhet med DES og trippel DES symmetrisk nøkkel for å kryptere ned dataene. Her benyttes det også 64 bit blokker, men 128 bit nøkkel for kryptering / dekryptering [WikiXTEA][Hernandez m.fl.].

2.4.5 RSA

Denne krypteringsformen benytter private og offentlige nøkler som er blitt generert matematisk. Sikkerheten her ligger i at man benytter seg av enormt store tall for å generere nøklene og at det ikke er samme nøkkelen som trengs for å dekryptere dataene som nøkkelen for å kryptere dataene.

2.4.6 Sikkerheten til de forskjellige krypteringsalgoritmene:

Vi ser på forskjellen mellom de forskjellige krypteringsalgoritmene:

- En algoritme for å cracke 9-round DES med 215.8 kjente plaintexts har en $2^{29.2}$ "time complexity" (Biham et al, 2002).
    - DES nøkkel har blitt funnet I løpet av 50 dager på 12 HP 9735 arbeidsstasjoner.
- XTEA: $2^{20.5}$ chosen plaintexts and a time complexity of $2^{115.15}$ [Ko m. fl.].
- RSA: her er det avgjørende på hvor stor nøklene er for tiden det trengs for en utenforstående å dekryptere dataene.
    - For en nøkkel på 512 bit trenger 50 arbeidsstasjoner 7 måneder og 6 dager[Robshaw].

Vi kan konkludere md at RSA krypteringen er en relativt sterk kryptering, men som nevnt tidligere trengs det mer ressurser å i det hele tatt å kryptere og dekryptere data. DES krypteringen er relativt svak men blir noe sterkere i trippel DES da denne bruker 2 forskjellige nøkler. XTEA gir heller ikke den perfekte krypteringen men er bedre en DES.

En fast regel er at jo lenger nøkler en bruker jo sikrere er krypteringen. Siden trippel DES og XTEA benytter seg av lengre nøkler er disse sikrere. RSA krypteringen kan ha forskjellig lengde på nøklene (128 – 2048 bit nøkler), og ved å øke disse økes også sikkerheten.

2.4.7 Valg av krypteringsalgoritme

[Denne delen må uansett endres når vi har kommet lengre i prosjektet. Vi har vel egentlig ikke blitt helt enige om hva vi skal velge?]

Vi har en del elementer vi har måtte ta hensyn til. Krypteringen av dataene skjer på PDA – en enhet som kjører på begrensede resurser. Dessuten har plattformen den kjører på begrensede muligheter med tanke på innebygde funksjoner og muligheter (Dette må legges til i etterkant ved behov). Valget falt på XTEA. Dette er en raskere krypteringsmåte enn RSA, og gir med sine 128 bit nøkler en grei nok kryptering for prototypen vi utvikler. Det var ikke mange krypteringsmuligheter for NET Compact Framework plattformen programmet kjører på heller, og XTEA var en av få muligheter vi hadde. Vi må også ta forbehold med "Scenario 1" for å ikke

overbeskytte systemet slik at vi hindrer naturlig drift. Ved videreutvikling av programvaren kan naturligvis krypteringsalgoritmen byttes ut med en kraftigere algoritme når dette er gjort tilgjengelig for .NET plattformen eller en krypteringsmodul er tilgjengelig.

2.4.8 Scenario 1

Dataene vil ligge kryptert ned i databasen. Hvis DES eller trippel DES var benyttet ville det være større sannsynlighet for at dataene skulle bli dekryptert – da det er en og samme nøkkel som benyttes av både den som krypterer og dekrypterer.

Hvis en Public Key benyttes er det bare den som skal dekryptere som skal kunne gjøre dette. Svakheten er dermed at hvis helsepersonellet skal gjøre endringer i basen i etterkant vil det bli vanskeligere da PDA'en ikke har mulighetene til å dekryptere dataene før bruk.

En mulig løsning er at man benytter seg av DES eller trippel DES men har nøkkelen passordbeskyttet slik at eget passord må benyttes for å få tak i nøkkelen før bruk. Man kan også la passordet og nøklene være det samme, men da med en minimumslengde. Man har dermed muligheten til å ha dataene nedkryptert og kunne gjøre endringer i disse ved behov – uten å måtte gå igjennom hele spørreundersøkelsen på nytt.

2.4.9 Scenario 2

Ved nye endringer av databasen vil man alltid ha muligheten til å gå tilbake i historikken til databasen og se hvilke endringer som er blitt gjort når. De forskjellige endringene til et svaralternativ kan holdes atskilt ved at hvert datafeltet har et tilhørende timestamp.

Dette kan være nødvendig for å kunne oppdage feilaktige endringer eller misbruk av databasen.

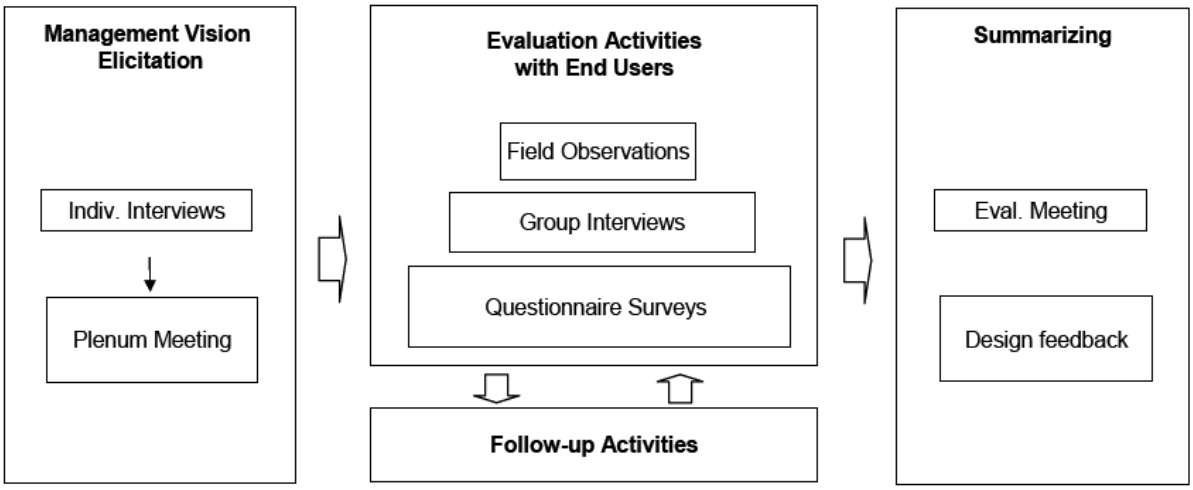
3 Drøfting

3.1 Evaluering av målsetninger

Målet med dette prosjektet har vært å effektivisere et papirbasert system, ved å bruke ny teknologi. Vi har derfor som en del av prosjektet utviklet et system basert på en database hvor PDA'er blir fronten mot brukeren i felten. En evaluering av det nye systemet vil være nødvendig for å kunne si hvor vellykket prosjektet er.

3.2 Forundersøkelser (case studies)

Artikkelen "A Low Cost Design For User-Centered Pilot Studies" [Rahlff m. fl.] tar for seg hvordan bruk av forundersøkelser kan gi svar på spørsmål rundt det utviklede systemet på en kostnadseffektiv måte. En forundersøkelse eller "pilot case" blir definert som en nedskalert, objektiv undersøkelse hvor utvalgte brukere tester systemet i praktisk bruk innen gitte rammer. Forundersøkelsen har som hovedmål å finne ut om systemet er bra nok til en full utrulling i organisasjonen. Dette målet nåes ved å undersøke den praktiske nytten i organisasjonen, hvor fornøyde brukerne er, identifisere krav til infrastruktur og brukerstøtte, samt estimere kostnadseffektiviteten til det nye systemet. En forundersøkelse består fem trinn [Glass]: Planlegging, design, utførelse, evaluering og bruk.



Figur 3 [Rahlff m. fl.] viser de ulike elementene i en forundersøkelse.

Rahlff m. fl. fokuserer også på spesielle problemstillinger knyttet til det å evaluere et mobilt system. Det er hovedsakelig fire områder hvor en evaluering av et mobilt system avviker fra en vanlig brukerorientert evaluering. For det første er mobile systemer oftere drevet fram av ny teknologi, noe som kan føre til at ulik teknologi blir brukt rundt om i en organisasjon. Dette kan

skape forvirring, så det er viktig at ny teknologi velges hovedsakelig for å støtte et behov hos brukerne og ikke for teknologiens skyld. Den raske utviklingen setter også krav til en både rask og effektiv utviklingsprosess. En annen ting som er spesielt for mobile systemer er at nytten i stor grad styres av den kontekst systemet blir brukt i. Bruk i forskjellige miljøer skaper krav til robusthet, mobilitet osv. Systemet vil også ofte være et hjelpemiddel som støtter brukerens hovedaktivitet, noe som det må tas hensyn til både ved utvikling og evaluering. Til slutt settes det fokus på at mobile systemer ofte avhenger av andre systemer/annen infrastruktur, noe som gjør det viktig å identifisere hvor et problem har oppstått.

3.3 Evaluering av prototypen

Boken “Software for use” [Constantine m. fl.] beskriver i hovedsak en brukerorientert software utvikling, men legger også stor vekt på forskjellige måter å evaluere et system på. Vi har valgt å fokusere på tre metoder for å kvalitetssikre og evaluere systemet vårt; ekspertevaluering, metrikk og laboratorie-/felttesting.

En ekspertevaluering av systemet ved å bruke heuristiske prinsipper [Nielsen] og andre brukervennlighets prinsipper [Constantine m. fl. Appendix B] er en enkel og relativt lite tidkrevende prosess som vil kunne sørge for at systemet vårt holder det nivået vi trenger med tanke på brukernes kunnskapsnivå i å bruke tekniske hjelpemidler.

Hovedmålet med å bruke metrikk er å kunne sjekket at den totale prosessen med å samle inn data, digitalisere dem og hente dem ut har blitt kortere ved å fjerne bruken av papir. Vi kan også (hvis det er tid) måle og analysere de ulike delene av prosessen.

Felttesting er en viktig, og ofte presis, metode for å evaluere systemer, men denne typen testing tar mye tid å gjennomføre, samt at den avhenger av hjelp fra både intervjuobjekter og brukere. Bruk av ”ekte” intervjuobjekter skaper også utfordringer med tanke på sensitiv informasjon og hvor ”kunstig” test situasjonen blir. Det mest realistiske for oss vil nok derfor bli å gjennomføre en laboratorietest av systemet hvor gruppemedlemmer besetter de ulike rollene.

3.4 Helsepersonell og pasientforhold

Å evaluere prototypen er viktig – både for å kunne forbedre prototypen i riktig retning, men også for å kunne forbedre og gjøre endringer i eksisterende rutiner og tiltak som organisasjonen allerede har. Ved introduksjon av nytt system skal man helst øke effektivitet, redusere kostnader og øke påliteligheten. Det er dette som er tanken bak introduksjonen av digital innhenting av informasjon.

Hvis man benytter en inkrementell utviklingsmodell (eller en spiralmodell) [Sommerville] kan man ved å introdusere nye eller endrede funksjoner kunne utvikle programvaren sammen med oppdragsgiver, og dermed unngå en del problemer som at pasienten føler seg mer distansert fra helsepersonell [Luff m.fl.].

Hvordan personell håndterer papirversjonen eller den elektroniske versjonen av spørreskjemaene kan ha innvirkning på pasientens oppfatning av situasjonen [Luff m.fl.].

Det kan også være aktuelt å teste dette ved å la en av utviklerne ta pasientens plass og evaluere hvordan hans / hennes oppfatning av situasjonen er når helsepersonellet benytter seg av papirspørreskjemaene kontra PDA. Ved å innhente oppfatningene fra både utvikler og helsepersonellet kan man optimalisere programvaren og samtidig komme med anbefalte arbeidsrutiner slik at man på denne måten ikke distanserer seg fra pasienten ved senere arbeid. Man kan også legge til rett opplæring av helsepersonellet slik at riktige holdninger til pasienten og riktig bruk av PDA blir lært på et tidlig stadium.

Det er heller ikke ønskelig at man ved innføringen av nytt mobilt utstyr at man skaper uheldige arbeidsvaner eller at man ansetter ekstra personell som har som fulltidsbeskjeftigelse å føre inn data i databasen etter at PDA er tatt i bruk. [Luff m.fl.].

4 Kildevurdering

Når man vurderer kilder er det to ting man må vurdere, om kilden er pålitelig og om den er gyldig. Dette har vært spesielt viktig for oss siden vi i stor grad har brukt informasjon vi har funnet på internett.

Vi har brukt en del artikler, noen av disse er pensum i faget og en del andre har vi funnet selv. De artiklene som er pensum og vi fant aktuelle for vår oppgave kan vi si er både pålitelige og gyldige. De vi fant på egenhånd er funnet gjennom referansedatabaser som ACM Digital Library og Inspec og det gjør at man kan stole på dem. Men det sikrer ikke nødvendigvis at de er gyldige for oppgaven vår. I IT-verden hvor utviklingen går fort kan det være viktig å ta hensyn til når en artikkel er skrevet. Vi mener at de artiklene vi bruker har både god pålitelighet og gyldighet.

Vi har også brukt Wikipedia som oppslagsverk. Dette oppslagsverket er av en slik natur at alle kan gå inn å redigere informasjonen som ligger tilgjengelig. Så selv om vi finner informasjon som er svært relevant kan vi aldri stole 100% på informasjonen som ligger her. Den kan ha vært lagt inn informasjon som ikke stemmer helt og informasjonen forandrer seg ofte etter hvert det blir skrevet mer. Washington Post har skrevet en artikkel om Wikipedia[WikkiWash] og her kommer det fram at Wikipedia her god kvalitet men ikke på høyde med de mer kommersielle oppslagsverkene. På en annen side har Wikipedia ofte mer oppdatert og mer fullstendig informasjon.

Vi har også søkt på internett for å finne informasjon om teknologier og da på hjemmesidene til de kommersielle leverandører og vi går ut i fra at informasjonen om deres produkter og teknologi stemmer.

Alt i alt synes vi at vi kildene vi har brukt har god pålitelighet og gyldighet. Vi kunne kanskje ha brukt oppslagsverk som er noe bedre kvalitetssikret, men da ville det også blitt vanskeligere å finne fram til den informasjonen vi trengte.

5 Oppsummering

5.1 Konklusjon

[hva vi har kommet fram til]

5.2 Videre Arbeid

[hvor går veien videre]

6 Referanser

[Aldex] Aldex Software Ltd,
”SQL vs Access”,
http://www.cypressinland.com/access_vs_sql2.htm

[Constantine m. fl.] Constantine, L. L. & L. A. D. Lockwood,
”Software for use”,
Addison Wesley / ACM Press, 1999.

[ForskriftPerson] Lov Data
”FOR 2000-12-15 nr 1265: Forskrift om behandling av personopplysninger (personopplysningsforskriften)”,
http://www.lovdata.no/cgi-wift/ldles?doc=/sf/sf/sf-20001215-1265.html

[Glass] R. Glass,
“Pilot Studies: What, Why and How”,
Journal of Systems and Software, 1997. 36(1): p. 85-97.

[Han m. fl.] Han, S. & H. Oh & J. Park,
”The improved data encryption standard (DES) Algorithm 1996”.

[Handango]
“History of the Personal Digital Assistant”,
http://www.handango.com/PDAHistory.jsp?siteId=1

[Hernandez m.fl] Hernandez, J. C. & P. Isasi,
“New New Results on the Generic Cryptanalysis of TEA and Reduces-Round Versions of XTEA”

[Ko m. fl] Ko, Y. & S. Hong & W. Lee & S. Lee & J. Lim,
“Related key differential attacks on 26 rounds of XTEA and full rounds of GOST”
In Proceedings of FSE '04, Lecture Notes in Computer Science, 2004. Springer-Verlag.

[LovPerson] Lov Data
”Lov om behandling av personopplysninger (personopplysningsloven)”,
http://www.lovdata.no/cgi-wift/wiftldles?doc=/usr/www/lovdata/all/nl-20000414-031.html&dep=alle&titt=personopplysning&

[Luff m.fl.] Luff, P. & C. Health,
”Mobility in Collaboration”,
In Proceedings of CSCW’98. November 14 – 18, Seattle. Pp. 305-314,
ACM Pre 1998.

[Mathiassen m. fl.] Mathiassen, L. & Munk-Madsen, A. & Nielsen, P. A. & J. Stage,
“Object Oriented Analysis & Design”,
Aalborg: Marko Publishers, 2000.

[Medicom] Jørgensen, M & B. Dalhaug,
”Patient Response Module”,
Hovedprosjekt ved Høgskolen i Oslo 2003,
http://student.iu.hio.no/hovedprosjekter/2003/data/11/

[Nielsen] J. Nielsen,
“Ten Usability Heuristics”,
http://www.useit.com/papers/heuristic/heuristic_list.html.

[Perlow] J. Perlow,
“Comparing Windows CE with Palm OS”,
Palmpower Magazine, issue 3/1999, ZATZ Publishing.

[Rahlff m. fl.] Rahlff, O. & A. Følstad,
“A Low Cost Design For User-Centered Pilot Studies – Evaluation Mobile Work Support Systems”,
NordicCHI ‘04, SINTEF, 2004.

[Robshaw] M. J. B. Robshaw,
“Security estimates for 512-bit RSA“,
WESCON/'95. Conference record. 'Microelectronics Communications Technology Producing Quality Products Mobile and Portable Power Emerging Technologies' , 7-9 Nov. 1995.

[Sellen m. fl.] Sellen, A. & R. Harper,
“Paper as an Analytic Resource for the Design of New Technologies”,
CHI 97, ACM Inc, 1997.

[Sommerville] I. Sommerville,
“Software Engineering Seventh Edition 2004”,
ISBN: 0-321-21026-3, Side 71-74

[WikiPDA]
http://en.wikipedia.org/wiki/Personal_digital_assistant

[WikiDes] [WikiDes]
http://en.wikipedia.org/wiki/DES

[WebopediaDES]
http://www.webopedia.com/TERM/D/DES.html

[WikiPubKey]
http://en.wikipedia.org/wiki/Public_key

[WikiRSA]
http://en.wikipedia.org/wiki/RSA

[Wiki3DES]
http://en.wikipedia.org/wiki/Triple_DES

[WikiMITMA]
http://en.wikipedia.org/wiki/Man-in-the-middle_attack

[WikiWash] Walker, L.
Spreading Knowledge, The Wiki Way.
09.09.2004: Washington Post.
http://www.washingtonpost.com/ac2/wp-dyn/A5430-2004Sep8?language=printer
Besøkt 03.04.05

[WikiXTEA]
http://en.wikipedia.org/wiki/XTEA

[WebopediaPKC]
http://www.webopedia.com/TERM/p/public_key_cryptography.html

[WebopediaRSA]
http://www.webopedia.com/TERM/R/RSA.html

[WebopediaRSA]
http://en.wikipedia.org/wiki/Diffie-Hellman

Vedlegg

Vedlegg 1 - Lov / forskrift
Vedlegg 2 - Ten Usability Heuristics [Nielsen]
Vedlegg 3 – Use Case
Vedlegg 4 – Framdriftsplan

Vedlegg 1: Lov / forskrift

§ 2-10. Fysisk sikring

Det skal treffes tiltak mot uautorisert adgang til utstyr som brukes for å behandle personopplysninger etter forskriften her.

Sikkerhetstiltakene skal også hindre uautorisert adgang til annet utstyr av betydning for informasjonssikkerheten.

Utstyr skal installeres slik at ikke påvirkning fra driftsmiljøet får betydning for behandlingen av personopplysninger.

§ 2-11. Sikring av konfidensialitet

Det skal treffes tiltak mot uautorisert innsyn i personopplysninger hvor konfidensialitet er nødvendig.

Sikkerhetstiltakene skal også hindre uautorisert innsyn i annen informasjon med betydning for informasjonssikkerheten.

Personopplysninger som overføres elektronisk ved hjelp av overføringsmedium utenfor den behandlingsansvarliges fysiske kontroll, skal krypteres eller sikres på annen måte når konfidensialitet er nødvendig.

For lagringsmedium som inneholder personopplysninger hvor konfidensialitet er nødvendig, skal behovet for sikring av konfidensialitet fremgå ved hjelp av merking eller på annen måte.

Dersom lagringsmediet ikke lenger benyttes for behandling av slike opplysninger, skal opplysningene slettes fra lagringsmediet.

§ 2-12. Sikring av tilgjengelighet

Det skal treffes tiltak for å sikre tilgang til personopplysninger hvor tilgjengelighet er nødvendig.

Sikkerhetstiltakene skal også sikre tilgang til annen informasjon med betydning for informasjonssikkerheten.

Alternativ behandling skal forberedes for de tilfeller informasjonssystemet er utilgjengelig for normal bruk.

Personopplysninger og annen informasjon som er nødvendig for gjenoppretting av normal bruk, skal kopieres.

§ 2-13. Sikring av integritet

Det skal treffes tiltak mot uautorisert endring av personopplysninger der integritet er nødvendig.

Sikkerhetstiltakene skal også hindre uautorisert endring av annen informasjon med betydning for informasjonssikkerheten.

Det skal treffes tiltak mot ødeleggende programvare.

§ 2-14. Sikkerhetstiltak

Sikkerhetstiltak skal hindre uautorisert bruk av informasjonssystemet og gjøre det mulig å oppdage forsøk på slik bruk.

Forsøk på uautorisert bruk av informasjonssystemet skal registreres.

Sikkerhetstiltak skal omfatte tiltak som ikke kan påvirkes eller omgås av medarbeiderne, og ikke være begrenset til handlinger som den enkelte forutsettes å utføre.

Sikkerhetstiltak skal dokumenteres.

Vedlegg 2: Ten Usability Heuristics [Nielsen]

These are ten general principles for user interface design. They are called "heuristics" because they are more in the nature of rules of thumb than specific usability guidelines.

Visibility of system status : The system should always keep users informed about what is going on, through appropriate feedback within reasonable time.

Match between system and the real world: The system should speak the users' language, with words, phrases and concepts familiar to the user, rather than system-oriented terms. Follow real-world conventions, making information appear in a natural and logical order.

User control and freedom: Users often choose system functions by mistake and will need a clearly marked "emergency exit" to leave the unwanted state without having to go through an extended dialogue. Support undo and redo.

Consistency and standards: Users should not have to wonder whether different words, situations, or actions mean the same thing. Follow platform conventions.

Error prevention: Even better than good error messages is a careful design which prevents a problem from occurring in the first place. Either eliminate error-prone conditions or check for them and present users with a confirmation option before they commit to the action.

Recognition rather than recall: Minimize the user's memory load by making objects, actions, and options visible. The user should not have to remember information from one part of the dialogue to another. Instructions for use of the system should be visible or easily retrievable whenever appropriate.

Flexibility and efficiency of use: Accelerators -- unseen by the novice user -- may often speed up the interaction for the expert user such that the system can cater to both inexperienced and experienced users. Allow users to tailor frequent actions.

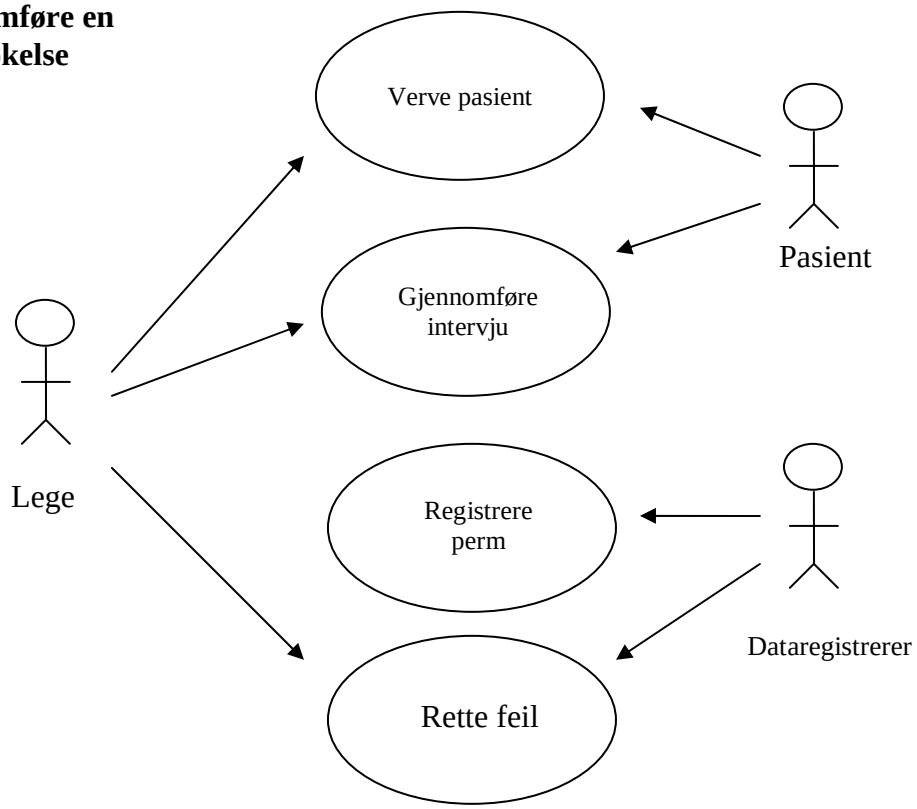
Aesthetic and minimalist design: Dialogues should not contain information which is irrelevant or rarely needed. Every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility.

Help users recognize, diagnose, and recover from errors: Error messages should be expressed in plain language (no codes), precisely indicate the problem, and constructively suggest a solution.

Help and documentation: Even though it is better if the system can be used without documentation, it may be necessary to provide help and documentation. Any such information should be easy to search, focused on the user's task, list concrete steps to be carried out, and not be too large.

Vedlegg 3: Use case beskrivelse

Gjennomføre en undersøkelse



En samlet use case beskrivelse for overstående figur

Use Case	Gjennomføre undersøkelse
Aktør	Lege, pasient, dataregistrere
Trigger	Lege ønsker å intervjue pasient
Pre-betingelser	Pasienten ønsker å delta på intervju. Pasienten møter opp til intervju.
Post-betingelser	

Normal hendelsesflyt	1. Lege ute i felt spør pasienter om de ønsker å delta i undersøkelsen 2. Pasienten avtaler intervjutid med lege 3. Pasienten møter intervju 4. Lege henter intervjuperm i arkivet på Ullevål 5. Lege møter pasient på avtalt sted og tid 6. Intervju blir gjennomført 7. Lege går gjennom intervjuperm og arkiverer denne 8. Dataregistrerer henter permen i arkivet 9. Dataregistrerer registrer intervjuperm i database og kontrollerer for feil 10. Permen blir markert som registrert i arkivet. 11. Feil blir meldt til lege som prøver å rette opp disse 12. Oppdateringer blir registrert i både perm og database
Variasjoner	1a) Pasienten ønsker ikke å delta 3a) Pasienten møter ikke opp 3a1) Lege må purre på pasient 6a) Pasient klarer ikke gjennomføre intervjuet 6a1) Dersom pasient ønsker det lages ny avtale 11a) Lege oppdager feil på egenhånd og retter disse. Dersom perm er markert som registrert blir oppdateringer meldt til dataregistrerer.

Vedlegg 4: Framdriftsplan

	MILEPÆLER Ferdig med	
M1	Undringsdokument	09.feb
M2	Utkast til programstruktur	01.apr
M3	Midtrapport	06.apr
M4	Prototyp ferdig	19.apr
M5	Evaluering	04.mai
M6	Sluttrapport	13.mai
M7	Eksamen	24.mai

	LEVERANSER	
L1	Undringsdokument	09.02.2005
L2	Midtrapport	06.04.2005
L3	Sluttrapport	13.05.2005

Figur 4 Milepælsplan

I tiden fram til den endelige rapporten skal leveres har vi valgt å legge vekt på koding (ferdigstille prototyp), gjennomføre en evaluering og trekke flere artikler inn i drøftingene våre.

GANTT-diagram (X angir hovedtidsrommet for aktiviteten, x angir "slakk")

		M1										M2	M3		M4		M5	M6			
	Aktivitet	**U3**	**U4**	**U5**	**U6**	**U7**	**U8**	**U9**	**U10**	**U11**	**U12**	**U13**	**U14**	**U15**	**U16**	**U17**	**U18**	**U19**	**U20**	**U21**	**Leveranse**
T1	**Prosjektstyring**																				
T1.1	Prosjektetablering	x	x	X																	
T1.2	Prosjektplanlegging			X	x	x	x														
T1.3	Gruppemøte		X	X	X	X			X			X	X	x	x	x	X	X			
T1.4	Planlegging av neste iterasjon		x	X				x	X						x	X					
T2	**Forretningsmodellering**																				
T2.1	Definere roller og ansvar			x	x	X															
T3	**Krav**																				
T3.1	Analysere problem			x	x	x	x	x	x	x	x	x	x	x	x						
T3.2	Forstå "stakeholderenes" behov			x	x																
T3.3	Definere systemet							x	x	x	x										
T4	**Analyse og design**																				
T4.1	Finne egnet system-arkitektur																				
T5	**Test**																				
T5.1	Definere test-prosedyrer												x	x	X						
T5.2	Utføre test-prosedyer														x	X					
T5.3	Utbedre															X					
T6	**Presentasjon**																			X	L6

Figur 5 Detaljert framdriftsplan