

INF5261

Gadgets

Midtveisrapport

Anders Langberget
Jan Henrik Øverland



14.03.08

Innholdsfortegnelse

| | | |
|-------|--|----|
| 1 | Introduksjon | 3 |
| 1.1 | Bakgrunn..... | 3 |
| 1.2 | Om prosjektet | 3 |
| 1.3 | Brukergruppe | 4 |
| 1.4 | Problemstilling | 4 |
| 2 | Teori..... | 5 |
| 2.1 | Mobile grensesnitt | 5 |
| 2.1.1 | Om mobile grensesnitt..... | 5 |
| 2.1.2 | Problemer med mobile grensesnitt..... | 5 |
| 2.1.3 | Grensesnitt i vårt prosjekt..... | 5 |
| 2.2 | Mobilitet | 6 |
| 3 | Metoder..... | 7 |
| 3.1 | Prototyping | 7 |
| 3.1.1 | Om prototyping | 7 |
| 3.1.3 | Low-fidelity papirprototype av vårt system | 8 |
| 3.1.4 | High-fidelity prototype..... | 8 |
| 3.2 | Evaluering | 8 |
| 3.2.1 | Om evaluering | 8 |
| 3.2.2 | Evaluering i vårt prosjekt | 9 |
| 3.2.3 | Om intervju | 9 |
| 3.2.4 | Intervju i vårt prosjekt..... | 9 |
| 4 | Resultat | 9 |
| 5 | Konklusjon | 9 |
| | Referanser..... | 10 |

1 Introduksjon

1.1 Bakgrunn

Dette er en prosjektoppgave i kurset INF5261 - Utvikling av mobile informasjonssystemer, ved Universitetet i Oslo våren 2008. Vi skal se på bruk av gadgets på mobile plattformer, samt utvikle en high-fidelity prototype.

1.2 Om prosjektet

Vi er en gruppe på to personer som skal lage en gadget som fungerer på en mobil plattform. En gadget er en innretning eller en applikasjon som har en nyttig spesifikk og praktisk hensikt. De blir ofte sett på som en nyvinning når de kommer og har ofte en uvanlig eller intelligent tilnærming til et problem. Innenfor programvarefeltet har det de siste årene kommet flere forskjellige typer små applikasjoner som er tiltenkt skrivebordet på en stasjonær datamaskin eller en mobil enhet. De små applikasjonene har fått mange forskjellige navn, men de mest vanlige er widgets fra Apple og gadgets fra Microsoft og Google. Det som kjennetegner disse små applikasjonene er at de har en bestemt oppgave og gir deg akkurat den informasjonen du er interessert i. I dagligtalen forbindes ofte betegnelsen "gadget" med en liten fysisk gjenstand, mens man med "widget" mener et lite program eller en applikasjon. I så måte kan man gjerne kalle vår applikasjon for en widget, men siden den er nært knyttet til det som i Windows Vista kalles "sidebar gadget" er det naturlig for oss å kalle den gadget.

Vår gadget vil hente informasjonen som skal vises på den mobile enheten fra en web service på en server. Denne web servicen skal forsyne gadgeten med nøkkelinformasjon som for eksempel serverlast, minnebruk og lagringsplass i tillegg til informasjon fra en applikasjon med navn WebX. WebX er utviklet på Microsofts plattform og programmeringsspråk, og er firmaet Webscape sin egenutviklede Content Management System (CMS). WebX gir brukervennlig webpublisering og administrering av innhold og brukere på et nettsted.

Tanken bak gadgets og widgets generelt er at de skal hjelpe deg med å overkomme den enorme mengden informasjon som finnes på internett. I stedet for å åpne en nettleser og gå til en bestemt adresse for å sjekke for eksempel været, kan man i stedet ha en gadget som henter denne informasjonen for deg. På denne måten får du enkelt tilgang til akkurat den informasjonen du ønsker og du kan få den direkte til ditt skrivebord eller din mobile enhet i stedet for at du selv må oppsøke informasjonen. En web service er et samlebegrep for et sett med standarder som bruker XML som representasjonsform. En web service fungerer som et maskin-til-maskin-program som utveksler informasjon over nettverk og er det nyeste innenfor distribuert programmering.

World Wide Web Consortium (W3C) definerer en Web service slik: "A Web service is a software system designed to support interoperable machine-to-machine interaction over a network. It has an interface described in a machine-processable format (specifically WSDL). Other systems interact with the Web service in a manner prescribed by its description using SOAP-messages, typically conveyed using HTTP with an XML serialization in conjunction with other Web-related standards."

Det er flere ting som motiverer oss ved denne oppgaven. Utgangspunktet er at Anders skriver masteroppgave for Webscape. Dette er et norsk selskap som står bak CMS (Content Management System) WebX og som leverer utviklings- og konsulenttjenester knyttet til internett og web services. Masteroppgaven går ut på å utvikle fire forskjellige Microsoft Vista Sidebar gadgets for Webscape, og å se på nytteverdien av disse små applikasjonene. I dette prosjektet ønsker vi altså å gjøre én av disse gadgetene tilgjengelig på en mobil plattform. Hovedgrunnen til

dette er at vi begge har lyst til å programmere noe som er beregnet på mobiltelefon. Dette er helt nytt for oss begge, og vi synes dette virker spennende å lære noe om.

Rent funksjonelt kan sannsynligvis en nettside gjøre samme nytte som en applikasjon i vårt tilfelle. Med Super-3G og raske nettlesere vil det nok heller ikke være en tregere løsning. Grunnen til at vi likevel velger å lage en applikasjon, er at vi har lagd nettsider som er tilpasset mobiltelefon før og at vi som nevnt over har lyst til å prøve noe nytt. Så lenge vår løsning ikke har klare begrensninger i forhold til nettside-løsningen, så vil aspektet ved at vi lager noe som er nytt for oss føre til økt motivasjon. Dessuten er det visse fordeler ved å velge en web service-løsning generelt sett. De er klient- og språkuavhengige, samt at de er skalerbare og bruker standard protokoller. XML-filer kan parses hos klienten, og da spiller det ingen rolle om klienten er for eksempel swing-gui, webapplikasjon, desktopapplikasjon eller mobilapplikasjon. Innholdet kan dessuten representeres i javascript ved hjelp av "document object model" (DOM).

I tillegg til at dette sannsynligvis blir gøy, så er det nyttig kunnskap og erfaring vi tilegner oss. Vi tror Java ME, Ajax og web services også er aktuelt langt fram i tid.

1.3 Brukergruppe

Som nevnt ovenfor skal en av gruppens medlemmer skrive en masteroppgave for en ekstern bedrift. Denne bedriften ønsket å få utviklet fire forskjellige gadgets til Microsoft sitt siste operativsystem, Windows Vista. Hver enkelt av disse gadgetene skal ha forskjellig brukergruppe og presentere informasjon som er relatert til den brukergruppen den representerer. Vi har bestemt oss for å konsentrere oss om gadgeten som er rettet mot systemansvarlig i bedriften. Denne gadgeten vil typisk fortelle systemansvarlig om prosessorlast, minnebruk og lagringsplass på bedriftens forskjellige servere, samt gi beskjed når en server "går ned". Grunnen til at vi har valgt å konsentrere oss om én gadget, er at det er enklere å konsentrere seg om én bestemt brukergruppe. I tillegg ville det bli for omfattende å utvikle fire gadgets for en mobilplattform i dette kurset.

1.4 Problemstilling

Vi ønsker å undersøke om en mobilapplikasjon, lik den som er beskrevet over, vil ha en hensikt for en systemansvarlig i en bedrift. Vil nedetid på servere kunne reduseres ved bruk av denne applikasjonen? Vi vil finne ut om en mini-applikasjon på en mobiltelefon som henter informasjon fra en web service har noen hensikt. Kan denne applikasjonen også gjøre hverdagen lettere for brukeren?

2 Teori

2.1 Mobile grensesnitt

2.1.1 Om mobile grensesnitt

I følge Buranatrived og Vickers (2004) har mobile enheter en mengde problemer sammenlignet med stasjonære datamaskiner, blant annet; mindre minne, lavere prosesseringskraft; mobile grensesnitt er ofte mindre sofistikert og relativt små. De nevner også en del fordeler ved mobile enheter; umiddelbar tilkobling, lokalisering, og muligheten til å motta informasjon og utføre transaksjoner hvor som helst, til hvilken som helst tid. For å utnytte dette må man utvikle effektive og nyttige applikasjoner med brukervennlige grensesnitt. Designe for mobilitet, en spredtliggende og heterogen populasjon, begrensede muligheter for inndata og utdata, og støtte multioppgavekjøring med en mengde forstyrrelser rundt er ingen enkel oppgave. Formelen for suksess er å utvikle en applikasjon som passer til begrensningen på skjermen, minnet og prosesseringskraften. Et annet viktig poeng Buranatrived og Vickers (2004) nevner er at et godt designet mobilt grensesnitt betyr mer enn og bare å presse informasjonen inn i en liten skjerm. Buranatrived og Vickers (2004) viser i en tabell til en undersøkelse gjort av Forester Research i 2004 som viser til at 37 % av brukerne har valgt dårlige brukergrensesnitt som grunn for at de ikke bruker mobilen til å handle med. Den eneste grunnen som scoret høyere var redd for å bli lurt.

En teknologi som Java 2 Micro Edition (J2ME) lar utviklerne jobbe effektivt ved at de kan skrive en applikasjon én gang og ta denne i bruk på mange forskjellige mobile plattformer. Alt som kreves er at det lages en profil til hver av plattformene du skal bruke den på. Dette har klare fordeler når det gjelder applikasjonskonsistens og effektivitet under utviklingen. Buranatrived og Vickers (2004) skriver videre at dette gjør at alle enheter viser grensesnittet nesten likt. Det betyr også at ved å ignorere enkelte karakteristikk ved individuelle enheter, så kan vi gå glipp av store forbedringer i brukervennligheten og tilfredsheten blant brukeren.

2.1.2 Problemer med mobile grensesnitt

Davies M. definerer fem vanskeligheter ved å lage grensesnitt til mobiltelefoner. Den største og viktigste er begrensningene til hver enhet man skal lage en applikasjon til. Mobile enheter har veldig forskjellig måte å manøvrere på og det er mer knotete å skrive inn informasjon. Ofte er det viktig å være først på markedet for få kundemassen, og derfor er det viktig at hele designprosessen går så smidig som mulig. Til nå har den mobile utviklingsindustrien vært for mye fokusert på hva som er teknologisk mulig og ikke vært nok fokusert på brukeren og hva han/hun trenger og vil ha (Davies, Case study).

2.1.3 Grensesnitt i vårt prosjekt

Noe vi må ta hensyn til er at vi har få piksler å implementere grensesnittet på. Det kreves derfor at utformingen av applikasjonens grensesnitt er svært nøye gjennomtenkt, for å være sikker på at informasjonen blir presentert på en lettfattelig måte. Derfor har vi valgt å kjøre papirprototyping på dette for å slippe å måtte programmere om grensesnittet senere. Dessuten finnes det fortsatt mange mobiltelefondisplay med dårlig fargegjengivelse, så det kan være fornuftig å styre unna nyanser i fargevalgene og heller basere seg på klare kontraster.

2.2 Mobilitet

<Tekst i kursiv er ikke ferdig skrevet.>

Her kommer det teori rundt mobilitet - spatial mobilitet, temporal mobilitet og kontekstuell mobilitet. Se artikkel: Expanding the 'Mobility' Concept av Masao Kakiyama & Carsten Sørensen.

However, in spite of the upsurge of concern with mobility in our social lives, current research perspectives define the notion of mobility quite narrowly, exclusively in terms of humans' independency from geographical constraints.

Mobilitet i vårt prosjekt vil være at vi gjør det mulig å overvåke bedriftens servere uavhengig av hvilken geografisk plassering man har, så lenge det er dekning på mobilen.

3 Metoder

3.1 Prototyping

3.1.1 Om prototyping

Prototyping og papirprototyping gir en mulighet til å gjennomføre raske design som adresserer problemer og lar designeren jobbe gjennom alle detaljene av brukererfaringen tidlig i prosessen (Davies, Case study).

Prototyping på papir er en rask og iterativ prosess, og på den måten kan problemer utforskes fra mange ulike vinkler raskt. Med prototyper kan man lage et kart over hele brukergrensesnittet for å vurdere flyten i programmet, men like fullt også designe detaljer som menyer, undermenyer, hovedfunksjoner og lignende. For en bedrift, er tiden designerne og utviklerne bruker en stor kostnad. Ved å benytte seg av papir-prototyping vil man spare tid i starten av et prosjekt. Dette fører til at iterasjoner og diskusjoner kan fullføres før designerne eller utviklerne starter på sitt tidskrevende arbeid med å lage visuell design eller skrive kode. Da spares det masse tid og penger siden det "riktige" produktet utvikles fra starten av. Utvikleren kan da basere sitt arbeid på det som allerede er laget i papir-prototypen. Som en del av arbeidet med slik prototyping, vil designerne av det grafiske grensesnittet arbeide tett med prototyping-teamet, for å identifisere potensielle brukererfaringer så tidlig som mulig i et prosjekt. Dette sparer også tid og penger siden brukervennlighetsproblemer blir fanget opp før det tidkrevende arbeidet med design settes i gang. (Davies, Case study)

3.1.2 Prototyping i vårt prosjekt

I begynnelsen av dette dokumentet skrev vi at det skulle utvikles en high fidelity prototype som en del av dette prosjektet. En high-fidelity prototype er ifølge Sharp et al (2007) en prototype som er laget av materialer som er forventet å bli brukt i den endelige versjonen av produktet. Materialer i vårt tilfelle vil da være en prototype laget i Java ME, kjørende på en mobiltelefon. Den ferdige versjonen skal også utvikles med Java ME, men den vil vært utstyrt med fullstendig funksjonalitet.

Før vi starter på high-fidelity prototypen, vil vi skissere en low-fidelity papirprototype for å raskt komme opp med et design vi ønsker å gå videre med. Slik unngår vi å skyte i blinde og med stor sannsynlighet bli nødt til å vrake kode, som tar lang tid å produsere sammenliknet med illustrasjoner på papir.

3.1.3 Low-fidelity papirprototype av vårt system

SKJERMBILDE 1

| | |
|------------------------|---|
| WebX - Webfarm gadget. | |
| Velg server: | |
| SERVER 1 | ▼ |
| Oppdateringsfrekvens: | |
| CPU last: | <input type="text" value="0,1"/> (Minutter) |
| Ledig harddiskplass: | <input type="text" value="5"/> (Minutter) |
| Ledig minne: | <input type="text" value="95"/> (Minutter) |
| Auskult | |

SKJERMBILDE 2

| | |
|-----------------------|--------------|
| WebX - Webfarm gadget | |
| Servernavn: | SERVER 1 |
| CPU last: | 37% |
| Ledig harddiskplass: | 62,5GB |
| Ledig minne: | 1024/2048 MB |
| Tilbake Auskult | |

3.1.4 High-fidelity prototype

Her kommer bilder av high-fidelity prototypen.

3.2 Evaluering

3.2.1 Om evaluering

Evaluering er en viktig del av designprosessen. Evaluering samler informasjon om brukeren eller potensielle brukere sin opplevelse av interaksjon med prototypen, datasystemet, en komponent av et datasystem eller en design-artifact for å kunne forbedre designet. Evalueringen fokuserer på både brukervennligheten og brukerens opplevelse av interaksjon med systemet for å kunne fastslå hvor tilfredstillende, morsomt eller motiverende interaksjonen er (Sharp et al, 2007, s.584).

Det er et enormt mangfold av interaktive produkter med funksjoner som behøver evaluering. Man kan enten velge å gjøre en evaluering i et laboratorium eller i produktets naturlige miljø. Hva som lønner seg avhenger av oppgaven. Er for eksempel oppgaven å få en bruker til å finne noe på en webside, kan dette godt gjøres i en lab. Dette kommer av at evaluatoren vet hva han skal se etter og har kontroll på hva han vil undersøke. Andre ting, som om barn liker en ny leke som skal lanseres blir evaluert bedre i naturlige omgivelser, som barna kjenner seg igjen i. Det er også viktig å ta i betraktning at forskjellige brukere har ulike preferanser for å utføre en oppgave. Ta for eksempel en 25-åring med gode datakunnskaper, denne personen vil naturlig nok løse en datarelatert oppgave enklere enn en 85-åring med minimal eller ingen kjennskap til data.

3.2.2 Evaluering i vårt prosjekt

Siden applikasjonen vår kun skal presentere informasjon og ha noen få skjermbilder, så er det ikke behov for store evalueringsprosesser med stor grad av brukerinteraksjon. Det vi føler vi kan få mest ut av med tanke på tilbakemelding fra brukere, er å kontakte en person som jobber som systemansvarlig, feks en fra drift på IFI, jevnlig underveis i prosessen. En annen mulighet er å foreta et intervju med den systemansvarlige i bedriften som gadgetene blir utviklet for. Resultatet av dette intervjuet kan hjelpe oss med å bestemme hva som burde vises av informasjon i applikasjonen. Vi kan i tillegg kartlegge hvor mye systemansvarlig bruker applikasjonen vår, hva han liker med den og eventuelt hva han ikke liker.

3.2.3 Om intervju

Intervjuer kan ses på som konversasjon med en bestemt hensikt (Kahn and Cannell, 1957). Det finnes fire forskjellige typer intervju: ustrukturert, strukturert, semi-strukturert og gruppeintervju (Fontana and Frey, 1994). Forskjellen på de tre første variantene er graden av kontroll intervjueren har på samtalen. Et eksempel på strukturert intervju er når intervjueren har med et forhåndslagt sett med spørsmål. Det siste alternativet involverer en liten gruppe som blir guidet av en tilrettelegger.

Det mest passende alternativet velges ut i fra hensikten med intervjuet, hvilke spørsmål som skal adresseres og hvor langt man er kommet i prosessen. Hvis hensikten er å observere brukeres førsteinntrykk av en ny designidé, for eksempel på et interaktivt kart, så er sannsynligvis et uformelt og åpent intervju uten begrensninger på tid eller på å få komme med egne innspill det mest hensiktsmessige alternativet. Hvis du derimot skal få feedback på feks layouten i et program, så kan det være fornuftig med et strukturert intervju. Dette er fordi målene og spørsmålene er mye tydeligere og mer spesifikke i dette tilfellet. Det er også en mulighet å benytte både strukturerte og ustrukturerte teknikker (altså semi-strukturert) der dette er hensiktsmessig.

3.2.4 Intervju i vårt prosjekt

Vi kommer til å benytte oss av semi-strukturert intervju. For oss er det ønskelig å få svar på forhåndslagde spørsmål, som hovedsaklig handler om å få brukeren til å prøvekjøre applikasjonen, samtidig som vi gjerne tar i mot innspill og forslag til forbedringer som brukeren selv kan komme på.

4 Resultat

5 Konklusjon

Referanser

Sharp H., Rogers Y., Preece J.; Interaction Design - Beyond human-computer interaction; 2nd edition; John Wiley & Sons Inc. (2007)

Buranatrived J. og Vickers P.; A Study of Application and Device Effects Between a WAP Phone and a Palm PDA; 6th International Symposium on Mobile Human-Computer Interaction, Mobile HCI (2004)

Davies M.; Paper Prototyping as a core tool in the design of mobile phone user experience, http://www.id-book.com/downloads/Casestudy_11point2.pdf

Kahn, R. and Cannell, C. (1957) The Dynamics of Interviewing. John Wiley & Sons, New York.

Fontana, A. and Frey, J.H. (1994) Interviewing: The Art of Science. In N. Denzin and Y. Lincoln (eds), Handbook of Qualitative Research. Sage, London, pp. 361-376.