

**Universitetet i Oslo
Institutt for informatikk**

INF5261 vår 2006

Midtveisrapport

Prosjekt WeGolf

Eivind Berg (eivindab (at) ifi.uio.no)

Anders Gjendem (andegje (at) ifi.uio.no)

Margrethe Store (margrsto (at) ifi.uio.no)

Terje K. L. Thoresen (tkthores (at) ifi.uio.no)

Are Wold (arew (at) ifi.uio.no)

29. februar 2006



Innhold

1 Innledning.....	3
1.1 Bakgrunn.....	3
1.2 Problemstilling.....	3
1.3 Begrensninger.....	3
2 Hovedprosjekt: Opptegning av punktinformasjon på golfbanen.....	3
2.1 Karttype.....	3
2.2 Kartskala.....	4
2.3 Relevante punkter.....	4
2.4 Mulige utvidelser.....	4
2.5 Satellittbilder versus andre fremstillinger.....	4
2.6 Mobiltelefonen som GPS og GIS-enhet.....	5
2.7 Alternativt prosjekt.....	6
3 Golf på mobiltelefonen.....	6
3.1 Sentrale golfbegreper.....	6
3.2 Konkurrenter til WeGolf.....	7
3.3 Intelligolf.....	7
3.4 mScorecard.....	8
3.5 Mobile Golf Scorer.....	8
3.6 iGolfscorer.....	8
3.7 Wireless18.....	9
3.8 Oppsummering.....	9
3.9 Golf og teknologi.....	9
4 Metode.....	10
5 Teknologi.....	10
5.1 J2ME vs J2SE/J2EE.....	10
5.2 Virtual Machines.....	11
5.3 Konfigurasjoner.....	11
5.4 Profiler.....	11
5.5 Hierarki.....	12
5.6 WeGolf og teknologi.....	12
5.7 Utviklingsmiljø.....	12
5.8 Grafikkmuligheter.....	13
6 Samarbeid og kommunikasjon.....	13
6.1 Innad i gruppa.....	13
6.2 Kommunikasjon med WeGolf.....	13
6.3 Tidsplan.....	13
7 Konklusjon og veien videre.....	13
8 Referanser.....	14

1 Innledning

1.1 Bakgrunn

WeGolf er et scorekort for golf som du fører på mobiltelefonen. Løsningen består av to deler, WeGolf Telefon og WeGolf Web som kommuniserer med hverandre over internett. WeGolf Telefon er et program som kan kjøres på alle mobiltelefoner som har Java. Etter at du har lastet ned og installert programvaren kan du føre score for opptil 4 spillere. WeGolf regner ut motatte slag ut fra ditt handicap, slope og rating, beregner score og stableford poeng for deg og de andre spillerne i flighten. I tillegg kan putter og eller straffeslag på hvert hull noteres. WeGolf Web er en webside hvor du kan registrere spillere, baner og runder og holde oversikt over rundene du har spilt.

Regelendringene fra 1 Januar 2006 gjør det lovlig å måle avstander med GPS på golfbanen. Har du en GPS i mobilen eller en GPS med Bluetooth, vil WeGolf gi deg lengden til hullet fra der du måtte finne deg på banen.

WeGolf finnes både i engelsk, norsk, svensk og italiensk versjon.

1.2 Problemstilling

Vi ønsker å generere et kart over golfbanen på brukerens mobiltelefon hvor viktige elementer er tegnet opp. Ved hjelp av informasjon om golfbanen, som fins på WeGolfs server eller som legges inn av brukeren, kan vi plote disse elementene inn på kartet. Golferen vil ha med seg en GPS-kapabel enhet, slik at mobiltelefonen til enhver tid vet akkurat hvor på kartet golferen befinner seg.

Golferen kan da ved hjelp av kartet orientere seg i forhold til hindringer, og spesielt se hva avstanden til en hindring er.

1.3 Begrensninger

WeGolf er dels en mobilapplikasjon, og i mobile enheter medfører spesielle utfordringer for design av brukergrensesnitt [1]. Kontaktflaten mellom bruker og system er mer begrenset, i form av blant annet liten skjermstørrelse og få, små taster, og utvikleren må ta hensyn til at miljøet programvaren benyttes i er mindre forutsigbart enn på en stasjonær datamaskin, som gjerne er det vi i størst grad har erfaring med å utvikle for. Disse begrensningene gjør at for eksempel design av valg og navigasjonskontroller må gjøres med omhu [2].

2 Hovedprosjekt: Opptegning av punktinformasjon på golfbanen

Det er ønskelig å implementere et kart over banen hvor punkter på banen som er relevant for golferen er tatt med. Dette kan for eksempel være bunkere, greenen, treklynger eller vann; objekter på banen som golferen kan orientere seg i forhold til eller bør unngå å treffe med ballen. En GPS-enhet gir data som brukes til å posisjonere golferen på kartet.

2.1 Karttype

De små skjermstørrelsene som er tilgjengelige på mobiltelefoner tatt i betraktning er det viktig å finne balansegangen mellom å vise for mye informasjon på en rotete måte og å ha så lite informasjon med at nytteverdien blir ubetydelig. Både oppløsning, lysstyrke og kontrast varierer mellom mobiltelefoner, så det er svært viktig at kartet på skjermen er utformet med tanke på å fungere på et bredt utvalg skjermer.

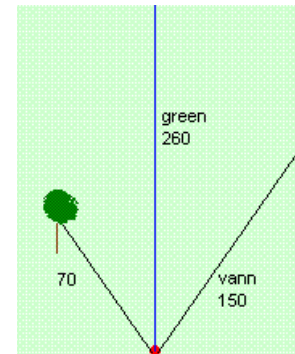
2.2 Kartskala

Utarbeiding av algoritmer for å velge ut et passende kartutsnitt å vise frem på skjermen er en sentral problemstilling. Golferen må vite om hindringene som er relevant for det neste slaget; samtidig er det ønskelig å få et overblikk over hele hullet. [3] demonstrerer en algoritme for å produsere kart velegnet for fremstilling på begrenset skjermareal. [4] viser en metode for å synliggjøre punkter som befinner seg utenfor kartutsnittet som vises. Begge artikler viser interessante metoder for å gjøre informasjon mer lettfattelig på små skjermer, og vi vil vurdere om vi kan benytte oss av en lignende metodikk.

2.3 Relevante punkter

Koordinater på golfbanen som vanligvis vil være registrert er greenen, vann, trær og bunkere. Områder, som green og vann, er oftest registrert med et frontpunkt, et midtpunkt og et bakpunkt. Vi må vurdere hva vi skal vise i slike tilfeller; det er nærliggende å tro at frontpunktet vil være mest interessant ("hvor langt unna er den vanddammen?"). Hvilke punkter som til enhver tid skal tegnes med på skjermen må vurderes nøye. Golferen må lett kunne orientere seg på hullet og forstå hvor hindringene skjermen peker ut er i forhold til ham selv. Her må vi gjøre vurderinger omkring rotasjon av kartet og hvordan vi skal posisjonere dette.

Illustrasjon 1 er et svært primitivt eksempel på hvordan det tegnede kartet kan se ut. Golferen har sin egen posisjon, retning og avstand mot greenen og en vanddam, og har en nær, synlig hindring (treet) markert for å lette orientering i forhold til de andre punktene.



Illustrasjon 1: Et eksempel på hvordan kartet kan se ut; golferens posisjon nederst

2.4 Satelittbilder versus andre fremstillinger

En løsning, som vi planla å jobbe med i første omgang, var om det var mulig å importere satelittbilder til telefonen, for så å kunne legge ekstra informasjon som et "overlay". Vi vurderte forskjellige løsninger for å oppnå dette:

- **Import av satelittbilder over golfbaner direkte**

Dette viste seg å ikke være realistisk, da det sannsynligvis ikke finnes noen organisasjoner som tilbyr dette gratis for kommersiell bruk. Kjøp av satelittbilder vil sannsynligvis ikke være spesielt kostnadseffektivt for WeGolf i forhold til nytteverdien.

- **Bruk av Google Earths biblioteker**

Det eksisterer et åpent språk, KML, for å kunne legge egne data som overlay i Google Earth. Det ser ikke til at det er mulig å ekstrahere selve satelittbildene til egen mobilapplikasjoner. En mulig anvendelse av KML og Google Earth er å lagre golferens slag som koordinater og så ved hjelp av KML få dette ut på Googles satelittkart. Dette er imidlertid noe som kun lar seg gjøre på web, på grunn av tilgjengeligheten av Google Earths bilder.

Det er med bakgrunn i dette at vi kan tenke oss en løsning der man tegner opp et enkelt kart over golfbanen. En mulig utvidelsesmulighet her er å sy dette sammen med fritt tilgjengelige oversiktskart fra for eksempel golfbanenes egne hjemmesider.

En annen mulig fremtidig utvidelse av hovedprosjektet kan være å hente inn en golfers tidligere runder på et hull og inkludere de på kartet over banen.

- **Bruk av tegnede kart**

En langt rimeligere løsning er å benytte tegnede kart. Enten kan vi tegne opp enkle representasjoner av golfbanene selv eller benytte oversiktsbilder av golfbanene som ofte er tilgjengelige på golfklubbenes hjemmesider. Det er denne løsningen vi ser for oss på nåværende tidspunkt.

- **Tekstlig representasjon**

En annen løsning er en mer primitiv og tekstlig fremstillingsmåte der brukeren får omtrentlig retning og avstand til forskjellige feature på banen (bunker, freeway, rough osv...). Det vil være hensiktsmessig å starte med en slik representasjon, da vi på denne måten får kontrollert at klassifiseringen av de ulike områdene fungerer. Vi betrakter dette som en nødløsning.

2.5 Mobiltelefonen som GPS og GIS-enhet

Etterhvert som mobiltelefonene har blitt kraftigere og fått større skjermer, har et annet applikasjonsområde som tidligere var forbeholdt

applikasjonsespesifikke enheter blitt tilgjengelig for mobiltelefonen, nærmere bestemt bruk av mobile terminaler som GPS. Ifølge [8], har såkalte Gis systemer (Geographical information systems) stort sett vært relevant for pc.

Man kan tenke seg mange forskjellige bruksområder for en «mobil GIS-utgave», ikke minst WeGolf der et *utopia* vil være et system som på mange måter er like funksjonelle som et vanlig GISsystem. Blant annet kan man

tenke seg en rik applikasjon der man har tilgang til ikke bare oversikt over golfbanene men også flere «layers» som inneholder informasjon om elevasjon, bakkeforhold og annen informasjon som er relevant for en golfspiller. På den annen side er det trolig bare proffe golfspillere som virkelig ville hatt nytte av slike data, og i tillegg ville dette kreve dyre og multispektrale satellittbilder. Vi betrakter derfor slik funksjonalitet som et endelig utopia. For vår del er imidlertid enkel GIS-funksjonalitet som klassifisering av punkter eller områder relevant for oss, for å kunne gruppere de forskjellige elementene man finner på en golfbane, og vil derfor konsentrere oss om dette.

Mobiltelefonene har enten innebygd GPS-enhet, eller man kommuniserer via Bluetooth med ekstern GPS-enhet, som vist på bildet.



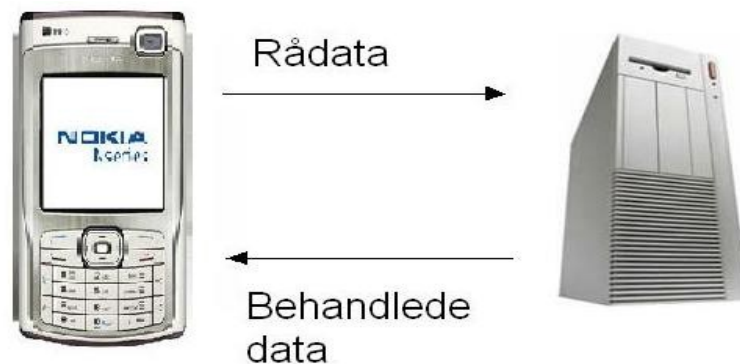
Videre må vi ta hensyn til de forskjeller mellom «vanlige» GIS systemer og mobile GIS-systemer, relatert til *Distribuerbarhet*, *perspektiv* og *mobilitet*. Disse forskjellene skiller det som omtales som vanlige GIS-systemer fra såkalte LBS (Location-Based Services).

- **Mobilitet**

Der vanlige GIS-systemer er basert på et overordnet bilde av ofte store landflater er LBS systemer basert på brukerens perspektiv i forhold til de *nærmeste* omgivelsene. Det er for oss uinteressant å ta med elementer utenfor golfbanen. I tillegg må representasjonene av golfbanene tilpasses de vesentlige mindre mobilskjermene.

- **Distribuerbarhet**

Mobiltelefoner har vesentlig mindre kapasitet enn en vanlig PC med tanke på lagringsplass, minne og prosessorhastighet. For å kunne tilby selv grunnleggende GIS-funksjonalitet kan det fort bli nødvendig å ty til en distribuert arkitektur der mobiltelefonen sender data til en server over mobilnettet for å utføre mer krevende beregninger. Dette bør ikke bli et problem for WeGolf, da noe av eksisterende funksjonalitet er basert på en slik arkitektur.



- Perspektiv

Henger sammen med Mobilitet. Siden golfbaner er relativt store, bør man trekke mobilitetsprinsippet enda lengre ved å tilby visning av den delen av golfbanen brukeren befinner seg i. Fra brukerens perspektiv vil dette oppleves som å «zooome» inn på den relevante delen av banen.

2.6 Alternativt prosjekt

Et alternativ til hovedprosjektet er å designe en IM-klient for mobiltelefonen som er i stand til å kommunisere med et tenkt diskusjonsforum under en golfers sider på WeGolf sin server. Da kan man se for seg at venner og bekjente følger runden på nett og sender beskjeder til golferen gjennom WeGolf-applikasjonen. Golferen kan svare tilbake gjennom applikasjonen ved anledning.

Da dette prosjektet involverer en anseelig bit arbeid på serversiden, har vi vurdert det som mer interessant å prioritere arbeid med opptegning av kart på mobilterminalen.

3 Golf på mobiltelefonen

3.1 Sentrale golfbegreper

Siden det blir mye snakk om golf tenkte vi det kunne være greit å gå igjennom noen sentrale begreper.

Birdie: Ett slag bedre enn hullets par

Bogey: Ett slag mer enn par på ett hull.

Bunker: Sandgrop, gjerne plassert i nærheten av green.

Drive: Et slag fra utslagsstedet.

Eagle: To slag bedre enn hullets par.

Fairway: Kortklipt område mellom utslagsstedet og greenen.

Green: Området på banen hvor hullet og flagget er.

Handicap: System for å utligne forskjeller mellom spillere, slik at en profesjonell og en amatør kan konkurrere mot hverandre.

Mottatte slag: Antall slag på en bane for en spiller basert på handicap. Korrigert for banens vanskelighetsgrad.

Par: Det antall slag man har satt opp for å nå greenen pluss to putter. For eksempel på et par 3 hull bruker man normalt ett slag inn på greenen og to putter.

Rough: Tykt og langt gress som er utenfor fairwayen.

Stableford: Poengsystem for å bedømme en runde. To poeng får man for å spille et hull på sin par. Sin par er korrigert etter mottatte slag og handicap. Tre poeng for ett slag bedre enn sitt par. Har man ett mottatt slag på et par 4 hull kan man bruke fem slag for å få to poeng. Får man over 36 poeng på en runde kan man skrive seg ned i handicap.

Tee: Utslagssted.

3.2 Konkurrenter til WeGolf

Vi har sett nærmere på fem konkurrenter til WeGolf. Alle fem ser ut til å ha tilfredstillende funksjonalitet for å fylle ut scorecards og beregne poeng og lignende, så forskjellene på programmene er mer hvilken ekstrafunksjonalitet man har, pris og plattform de kjører på.

3.3 Intelligolf

Intelligolf er bestefar i familien. Det er de som har vært på markedet desidert lengst, og de har en enorm markedsandel. Werner i WeGolf nevnte noe om 90%, uten at vi vet noe annet enn at de har hundretusener av kunder i 125 land. De startet allerede i 1997 og har med det mange års forsprang på konkurrentene.

De har en lang rekke med priser, blant annet ”Pocket PC magazine Best Software Award 2005”, så det er tydelig at de har en høy status i markedet. Programmet i seg selv ser ut til å ha mye og relevant statistikk både for pda'en og pc-applikasjonen. I databasen har de også 21.500 baner, noe som er vesentlig mer enn noen av konkurrentene.

Jim				
Par	3	4	5	All
Ave. Score	3.5	4.4	5.6	4.5
Ave. Putts	2.0	1.7	2.0	1.8
Ave. Putts, GIR	2.0	1.6	2.0	1.8
Ave. Drive	262	290	276	
Long Drive	280	325	325	
% Fairways	28	33	30	
% Greens	50	42	33	41
% Scrambling	0	25	50	28
% Up & down		100		100
% Sand saves		50		50

Hole 18 GPS	
324	100 214
314	150 164
304	200 114
118 Distance from shot Start	
142 Stream	Shot Track
247 Bunker Right	Start
273 Oak Tree Left	Stop
47°36.0063N 122°20.3799W ↗ 10	

Intelligolf er også en av de som tilbyr støtte for GPS. Den gir deg avstanden til ni punkter (200 meter, 150 meter og 100 meter fra hullet, front, bak, og midt på green, og tre valgfrie punkter) på hullet fra hvilken som helst posisjon på hullet.

Intelligolf er for øyeblikket kun kompatibel med PDA og PC (for de som ikke har en PDA, men likevel vil føre statistikk over resultater). De skal visstnok være i gang med å konvertere til mobiltelefoner, men vi har ikke sett noe til dette enda.

Det kommer i fem versjoner avhengig av hva man trenger og hvor mye man vil betale. Den billigste versjonen koster \$29.95 (har kun PDA-funksjonene), mens den dyreste koster \$49.95 og inkluderer GPS.

3.4 mScorecard

Dette er et relativt nytt produkt. Det ble utviklet først i 2003.



Vi har inntrykk av at mScorecard lider av en noe mangelfull internettside, men det har høstet mye ros for meget raskt og bra brukergrensesnitt, og var blant annet finalist hos Handango's beste nye applikasjon-konkurranse i 2005 (Handango er verdens ledende distributør av løsninger for mobile enheter).

mScorecard har heller ingen støtte for GPS, og prisen er kanskje deretter. Det koster kun \$19.95, noe som kan sies å være billig hvis de lover det de gjør på nettsidene, med gode statistikkverktøy og at det er et svært raskt og lettbetjenelig program. mScorecard er også kompatibelt med både mobiltelefoner, Blackberry og PDA'er.

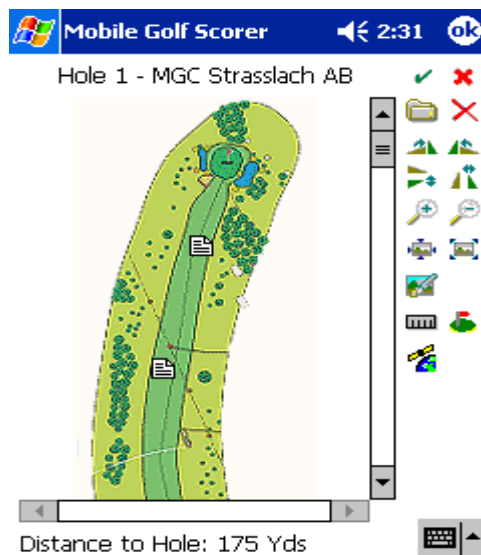
3.5 Mobile Golf Scorer

Også utviklet fra 2003 av en IT-konsulent som ikke klarte å finne noen gode programmer på markedet. Dette ser ut til å være blant de mest avanserte på markedet, med både integrasjon mot Google Earth, GPS og en stor statistikk modul.

Om man har en skikkelig PDA med 640*480 oppløsning får man tilgang til en avstandsbok, som gir bilder over hull og man kan definere punkter.

De tilbyr også GPS, som gir deg avstand fra ballen til alle koordinater du selv har lagret. I tillegg kan man eksportere alle koordinatene for å gi en meget pen oversikt over runden når man kommer hjem på sin egen PC.

Det er kun kompatibelt for PDA og PC, og kommer i tre versjoner. Den billigste koster \$35, mens den dyreste som inkluderer GPS koster \$107.



3.6 iGolfscorer

De sier de er verdens første på GPS på mobiltelefon. iGolfscorer gir distansen til hullet fra seks punkt på hullet, altså noe færre enn de andre applikasjonene med GPS-funksjonalitet.

Det er som sagt kompatibelt med mobiltelefoner, for øyeblikket relativt nye utgaver av kun Nokia og Sony Ericsson. iGolfscorer ser ikke ut til å noe eksternt programvare, den eneste eksterne kommunikasjonen ser ut til å være å sende scorecards via e-mail. Funksjonaliteten og statistikken (over maks 50 runder) som gies på mobiltelefonen ser dog ut til å være god, selv om 50 runder ikke vil holde for en golfer som virkelig vil holde oversikten.

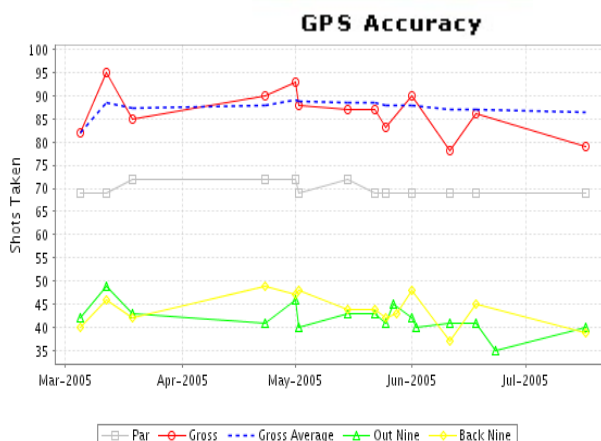
Prisen er ikke avskrekkende, for €19.99 gir den alle funksjoner bortsett fra GPS som du må betale €10 ekstra for.



3.7 Wireless18

Wireless18 har gått fra å bli en tanke hos en IT-konsulent som kjedet seg i 2003 til å bli et program med tusener av brukere i 130 land. Det har også vunnet Handango's People's Choice Award som vitner om et populært program. For de ser ut til å ha funnet sin lille nisje.

Det som skiller Wireless18 fra andre er at de ser ut til å bry seg mer om selve



community'et. Et eksempel på dette er at de har rankinglister der alle brukerne kan sammenligne seg selv med andre. Dette er naturligvis frivillig å være med på. Programmet er også gratis, men alle rundene vil bli slettet fra web'en etter sju dager om man ikke har betalt. Det har dog ingen støtte for GPS. Det er tilgjengelig på alle mobiltelefoner som støtter Java, i tillegg til PDA og Blackberry.

3.8 Oppsummering

Det er nok en fordel for WeGolf at de to mest omfattende programmene ikke har støtte for mobiltelefon, og i tillegg at det ene programmet som har støtte for GPS på mobiltelefon ikke ser ut til å være spesielt imponerende (siden iGolfScorer er det eneste programmet som ikke tilbyr ekstern lagring av informasjon).

Prismessig er det stor variasjon, og WeGolf er nok blant de dyreste i klassen, men det kommer jo alltid an på hvilken funksjonalitet brukeren krever. Det er ihvertfall mange programmer som holder på å bli etablerte, og det kan være vanskelig å slå igjennom på det jeg vil tro kan være et noe begrenset marked (ihvertfall i Norge).

3.9 Golf og teknologi

Golf er en veldig gammel og tradisjonsrik sport. Den har eksistert i over 500 år fra den oppstod i Skottland. Generelt har golf blitt sett på som en finmannssport som nå på det nærmeste betegnes som en folkesport. Utviklingen av regelverket har på en måte gått samme vei med en liberalisering av strenge kodekser, alt fra kleskodekser til turneringsregler.

For nå har Royal and Ancient Golf Club (golfens svar på FIFA) bestemt at det er tillatt å bruke hjelpemidler for å måle avstander også i turneringer. Det vil si at vi kanskje får se Suzann Pettersen gå rundt med en kikkert som måler avstander under turneringer snart. At GPS begynner å bli akseptert på denne måten har jo stor innvirkning på firmaer som satser på denne teknologien, som f.eks WeGolf.

Det er stadig kjente golfansikter ute og kritiserer de teknologiske fremskrittene i golfverdenen. Allerede på 70-tallet begynte man å bekymre seg, og fremskrittene til i dag har vært enorme. Det er ikke så lenge siden man spilte med køller som var laget av tre. Så kom køllene laget av jern, og nå senest bruker alle graffitkøller (som gjør skaftet lettere, det er bøyelig, så man får lengre baksving og høyere fart på køllehodet som igjen gir lengre slag). De siste 20 årene har også en gjennomsnittlig golfkølle blitt ca. 4cm lengre, noe som også gir lengre baksving og lengre slag. Før var også golfballene laget av lær som var stappet av gåsefjær. I dag lages det golfballer som passer for alle slags spillestiler. Om man vil ha høy spinn på ballen, ekstra høyde, eller ekstra lange slag. Det er også forskjell på baller for damer og herrer, samt at man har gjort ballene stadig mer aerodynamiske.

Så i dag slår golfproffene mye lengre enn de brukte å gjøre, som har ført til vesentlige endringer på golfbaner. En av verdens mest kjente golfbane, Augusta i Georgia USA, som holder en av fire Major-turneringer i året har måttet gjennomgå store endringer de siste 20 årene for å holde tritt med teknologien. Hullene har blitt lengre, fairwayen blir smalere, roughen blir tjukkere og lignende.

Så man kan si at golfsporten endres teknologisk hele tiden. Det blir stadig flere og bedre hjelpemidler for en jevnlig golfer, og hadde det ikke vært for en golfers stadig ønske om å ha det beste utstyret for å få de beste resultatene ville vi kanskje ikke sett samme utviklingen. Med mobiltelefon og PDA'er lager man nå programmer som gjør livet til en golfer enda lettere, spesielt med tanke på all statistikk man kan ha tilgjengelig slik at man får se hvilke steder det trengs forbedringer. Det vil alltid være noe motstand mot slikt utstyr siden mobiltelefon eller lignende ikke har noe å gjøre på en golfbane historisk sett, mens de andre teknologiske endringene vi ser er på utstyret som man bruker for å utføre sporten. Men, spørsmålet er om det fortsatt er en mobiltelefon om man bruker det til noe helt annet. Selv tror jeg at det vil alltid bli sett på som en uting, men det mest ideelle hadde kanskje vært om man kunne slå på telefonen og velge om man bare kunne kjøre

golfapplikasjonen (som MP3-spilleren på Sony Ericssons nyeste telefoner fungerer for å spare batteri) uten å ha telefonfunksjoner. Dette kan vel riktignok bli vanskelig om det ikke er produsenten av selve telefonen som står bak applikasjonen.

For spillerene i dag er generelt svært negative til mobiltelefoner siden det er sett på som et forstyrrende moment. Det er et enormt irritasjonsmoment om en telefon ringer når man slår. Ernie Els, en av verdens beste spillere, foreslår at man burde bruke metalledetektorer for å luke vekk mobiltelefonen fra tilskuerene. Det vil neppe skje, men det viser at man har en fiendtlig holdning til selve objektet.

Teknologi som GPS og Wi-Fi har også fått innpass på andre områder i golfverdenen, ikke bare hos spillere. Flere klubber, spesielt i USA benytter seg nå av GPS (som blant annet firmaet GPS Industries spesialiserte seg på) som hjelpemiddel. De har blant annet GPS på selve golfbilene (amerikanere går jo ikke på golfbanen) for å både se avstand til hull, men også for klubbene og monitorere farten spillet går på. (I golf er det ofte kø for å spille, og med 10-15 minutter mellom hver gruppe som går ut så trenger det bare en gruppe i løpet av en dag for å ødelegge hele tidstabellen). Bruksområdene er mange, f.eks. vise resultater under turneringer, studere overvåkningssystemer, eller la spillerene bruke mobilen til å få se menyer fra restauranten og bruke den til å bestille mat mens man fremdeles er på banen.

4 Metode

Vi bruker ikke et spesielt metode-rammeverk i dette prosjektet. Vi forsøker imidlertid å ha jevnlig møter med oppdragsgiver, og følger i grunn en standard systemutviklingsprosess, som vi har delt opp i følgende punkter.

- Kartlegging av krav
- Modellering
- Implementering og testing

I tillegg lærer vi oss også J2ME parallelt med disse aktivitetene, og vi vil også ha flere møter med oppdragsgiver underveis.

Gruppen befinner seg nå i modelleringsfasen.

5 Teknologi

5.1 J2ME vs J2SE/J2EE

Sun har valgt å lage forskjellige utgaver av Java tilpasset forskjellige typer omgivelser. På mobiler er det vanlig å bruke J2ME – Java 2 Micro Edition, en utgave spesielt tilpasset enheter med begrensede ressurser. I praksis betyr dette for utvikleren at man har et relativt begrenset API å forholde seg til for å skrive koden, og at mange av de klassene og metodene de fleste Javautviklere er kjent med fra den vanligste versjonen, J2SE (Java 2 Standard Edition) rett og slett ikke eksisterer. Kombinasjonen av begrensede ressurser og manglende APIer gjør utvikling for enheter som bruker J2ME til en utfordring i mange situasjoner. Et eksempel på dette er at flyttall mangler på mange enheter, og man kan risikere å måtte lage metoder for å konvertere heltall til flyttall for visning på skjerm selv ved hjelp av tekstmanipulasjon.

5.2 Virtual Machines

Enheter som bruker J2ME har på grunn av sine begrensede ressurser også en annen Virtual Machine enn J2SE/J2EE (Disse bruker normalt HotSpot VM (Virtual Machine) i dag – “fullversjonen”), mens J2ME bruker KVM – Kilobyte VM. En del “mellomstore” enheter bruker en mellomting –

nemlig CVM – Compact VM, eksempel på dette er Set-Top-bokser og enkelte PDA'er. Foreløpig har vi altså 3 forskjellige versjoner. Problemet Sun da stod ovenfor var at de ikke ville begrense J2ME helt ned til laveste felles nevner. Løsningen ble å innføre forskjellige konfigurasjoner og profiler under J2ME igjen.

5.3 Konfigurasjoner

En konfigurasjon definerer hvilke muligheter som skal være tilgjengelig. En konfigurasjon brukes på en hel serie av produkter med tilsvarende funksjonalitet og behov. Konfigurasjonen definerer for eksempel minimum maskinvarespesifikasjon, som minne, skjerm osv., samt hvilke bibliotek som skal være med.

CDC: Connected Device Configuration

- Set-top-bokser og PDA'er, bruker CVM.

CLDC: Connected Limited Device Configuration

- Stort sett mobiler og noen PDA'er. Denne er også kjent som “You-wanna-run-Java-on-what?!”

For å gjøre det enda litt mer detaljert er disse igjen oppdelt i versjoner. CLDC finnes for eksempel i utgave 1.0 og 1.1 – sistnevnte konfigurasjon støtter for eksempel flyttall som nevnt over. CLDC 1.1 er vanlig på de fleste nyere mobiler i dag, men det er svært mange ute i markedet fremdeles som ikke støtter mer enn versjon 1.0.

5.4 Profiler

En profil er igjen en slags utvidbar underkategori av en konfigurasjon. CLDC sier ingenting om GUI, det gjør profilen i vårt tilfelle; MIDP – Micro Information Devices Profile. Profilen er det laget som legger til kategori-spesifikke komponenter på toppen av en konfigurasjon; med andre ord er MIDP mer direkte rettet mot mulighetene mobiltelefoner har. MIDP definerer igjen en minste felles nevner for hva disse enhetene skal tilby av hardware og kommunikasjonsmuligheter samt minimumsspesifikasjoner for disse. (Tastatur, nettverk, minne (Temporært minne, ikke-volatilt for f.eks. konfigurasjon av programvaren/settings og for selve programvaren som kjøres), skjerm)

I tillegg er det MID-profilen som faktisk tilbyr bibliotekene man programmerer mot.

MIDP-program har begrenset tilgang til ressursene på enheten. Alle ressurser som skal brukes må være tilgjengelig via CLDC og/eller MIDP-bibliotekene, i JAR-filen som programmet selv kom i, eller via programmets beskrivelsesfil – JAD-filen. JAR-filen er en utvidet ZIP-fil på samme måte som i vanlig J2SE, den inneholder de kompilerte klassene, og eventuelle andre ressurser programmet trenger som bilder og tekst-data. JAD-filen er en ekstern fil, hovedgrunnen til at den eksisterer er at man skal slippe å laste ned hele JAR-filen for å finne ut hva denne inneholder. JAD-filen er i hovedsak en nedskalert utgave av manifestfilen som befinner seg inne i JAR-filen. Disse beskriver innholdet i JAR-filen; det være seg navnet på programmet i JAR-filen, ikon, hvilken klasse det er som er MIDleten – altså den klassen som tilsvarer main-klassen i vanlig J2SE/J2EE. Utover dette kan den også inneholde andre innstillinger som programmet kan lese underveis – dette kan for eksempel være nyttig ved kompatibilitetsproblemer mellom enheter ved at man kan sette spesielle innstillinger for en gitt enhet i en egen JAD-fil spesielt for den/de enhetene og la programmet lese denne, for så å tilpasse seg underveis i kjøringen av programmet. En JAD og JAR-fil kan inneholde flere forskjellige programmer, og de kalles da en MIDlet-suite.

MIDleten er som nevnt den klassen som tilsvarer main-metoden i J2SE – men den har ingen main-metode. Derimot har den tre forskjellige andre metoder, som blir kalt når programmet starter, når programmet blir stoppet (enten eksternt eller ved at programmet terminerer seg selv) – og en metode som blir kalt når kjøremiljøet vil pause programmet – for eksempel fordi det er en

innkommende samtale. Forøvrig fungerer pakker og klasser stort sett som vanlig.

Det er to forskjellige versjoner av MID-profilen på markedet nå; MIDP 1.0 og MIDP 2.0.

MIDP 1.0 var først ute, og inneholder en svært nedstrippet Java-versjon med støtte for timere, enkel GUI-støtte (2d-opptegning og menyer, tekstbokser osv), støtte for persistent lagring (RMS) og nettverk som HttpConnection. Dette tillater man å lage enkle programmer med menyer og utseende som ligner på enhetens eget – slik tilfellet med WeGolf er.

MIDP 2.0 er en solid utvidelse av den gamle spesifikasjonen. Ny funksjonalitet er for eksempel støtte for HTTPS-protokollen, støtte for lyd, bedre støtte for GUI-programmering, både på den gamle måten og funksjonalitet spesielt rettet mot spill. (Layers, transparente bilder) og sikkerhet.

5.5 Hierarki

Operativsystem -> JVM -> Konfigurasjon -> Profil -> Brukerprogram

5.6 WeGolf og teknologi

WeGolf fokuserer på et kjapt og enkelt program som ikke skal forstyrre brukeren eller være i veien. MIDP 1.0 sine enkle menyer og GUI-funksjonalitet er derfor et godt valg – dette er etterhvert godt innarbeidet, inneholder lite feil og er relativt konsistent fra mobil til mobil, samtidig som det ikke bryter dramatisk med mobiltelefonens eget utseende. I tillegg når WeGolf ut til et mye større marked ved å holde fast på MIDP 1.0 – det er fremdeles svært mange enheter på markedet som ikke støtter den nyere varianten. Ny funksjonalitet som oversiktskart osv. vil dog antakelig være enklere å implementere på en god måte ved bruk av MIDP 2.0 – og det kan derfor være greit å vurdere å ha to versjoner på markedet – der den nyeste utnytter større deler av mulighetene som de nye mobilene tilbyr. På den måten vil også brukerne få mer igjen for sin investering i dyrt utstyr. Det er mulig at et bedre alternativ dog vil være å ta utgangspunkt i MIDP 1.0 hele veien, men la programmet selv teste for nyere funksjonalitet og utnytte dette om det finnes tilgjengelig. Slik slipper brukeren å vite noe om hvilken profil mobiltelefonen har – svært få brukere vet slike tekniske detaljer.

5.7 Utviklingsmiljø

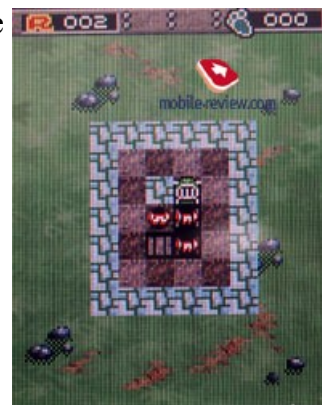
WeGolf har til nå blitt utviklet i det verktøyet som er regnet som det beste for mobilutvikling; nemlig Netbeans. Netbeans har over lengre tid fokusert sterkt på støtten for mobilutvikling - dette er et område hvor det lenge har vært ganske dårlig verktøystøtte. Netbeans Mobility Pack støtter ved hjelp av GUI-valg hele prosessen fra prosjektopprettelse til utprøving på den fysiske enheten. Den tette integrasjonen mot både mobilen og emulatorer fra de forskjellige telefonleverandørene gjør en ellers ganske kronglete prosess enkel.

Mobiltelefonleverandørene leverer sine egne emulatorer for sine modeller - disse emulatorene kjører ofte den samme programvaren som de faktiske telefonene slik at testing blir realistisk på et stort antall enheter, uten at man faktisk må investere relativt store penger for å få testet mot alle de forskjellige modellene. Det har dog vært et problem at ikke alle emulatorene faktisk gjør alt 100% likt på den fysiske telefonen, for eksempel pga. forskjellig programvareversjon, eller rett og slett at emulatoren har kjørt en helt annen kodebase. På litt eldre enheter er dog slike feil stort sett fjernet siden emulatorene relativt lett kan oppgraderes og er en ekstern plugin til Netbeans Mobility Pack.

De forskjellige emulatorene gir ofte også mulighet for at man kan velge ytterligere støttede (valgfrie) API - slik at programvaren blir spesialtilpasset for mobiltelefoner med støtte for eksempelvis Messaging, 3D-grafikk og lignende.

5.8 Grafikkmuligheter

MIDP 2.0 fikk som nevnt utvidet støtten for spillprogrammering. En av de viktigste nye funksjonene som ble lagt til var støtte for flere lag med grafikk som opererer uavhengig av hverandre, og som mobilen selv i større grad håndterer gjennom API. Illustrasjon 2 viser et eksempel på et tile-basert spill. Her ser vi helt tydelig at det antakelig er brukt såkalte tiles i bakgrunnen, dvs. at grafikken gjerne blir laget i ett stort bilde, og at dette igjen blir delt opp i små biter som settes sammen av de innebygde API'ene i den rekkefølgen programmereren ønsker. Slik sparer man både antall filer, tiden det tar å tegne opp grafikken, og ikke minst slipper utvikleren mye ekstraarbeid med å få programmet til å tegne opp enkeltpunkter og sette disse sammen til større bilder. Nyere mobiler støtter nå også 3D via egne API; illustrasjon 3 er et eksempel på dette. MIDP 2.0 lar selvfølgelig også utvikleren kombinere disse tile-baserte lagene med andre lag, og ren punkt/strek-tegning over disse igjen. Dette er antakelig noe vi kan dra nytte av i en senere implementasjon av WeGolf, ved at vi tegner opp en mer komplett bane enn det vi i første omgang tar sikte på. I utgangspunktet regner vi med å kun bruke MIDP 1.0 støtte tegning som tillater punkter og streker med tilhørende tekst, slik at vi får opp avstand til de nærmeste farene på banen og retningen spilleren skal.



Illustrasjon 2: Eksempel på tile-basert spill i MIDP 2.0



Illustrasjon 3: 3D-grafikk i MIDP 2.0

6 Samarbeid og kommunikasjon

6.1 Innad i gruppa

Vi har valgt å ha fast møtetid hver torsdag, med mulighet til å utvide ved behov. Til kommunikasjon baserer vi oss på e-post og en wiki.

6.2 Kommunikasjon med WeGolf

Kommunikasjonen med WeGolf har stort sett gått over mail, men med noen møter for å informere og diskutere videre arbeid (møte 2. mars og 22. mars). Neste møte har vi avtalt å gjennomføre over nettet, med IP-telefon og webkamera.

Det er lagt opp til at vi skal få en innføring i golf på WeGolf sin regning på et senere tidspunkt.

6.3 Tidsplan

Innlevering undringsdokumenter:	08.02.2006
Tilbakemelding undringsdokumenter:	01.03.2006
Presentasjon av prosjektet i kurset:	22.03.2006
Presentasjon av artikkel ("Just say Nokia"):	22.03.2006
Innlevering midtveisrapport:	29.03.2006
Tilbakemelding midtveisrapport:	19.04.2006
Innlevering sluttrapport:	10.05.2006

7 Konklusjon og veien videre

Vi har fått en introduksjon til golfsporten, til WeGolf-applikasjonen som den er i dag, og teknologien vi vil bruke til å utvikle kartvisning i WeGolf-applikasjonen. Videre vil vi fokusere på å finne algoritmer for å velge ut relevant kartdata og tilegning av J2ME-ferdighetene som trengs for å realisere dem på mobiltelefonen.

8 Referanser

- 1: “The Challenge of Mobile Devices for Human Computer Interaction”, Dunlop, Brewster, Personal and Ubiquitous Computing (2002), 6:235-236
- 2: “Evaluating Interface Design Choices on WAP Phones: Navigation and Selection”, Chittaro, Dal Cin, Personal and Ubiquitous Computing (2002), 6:237-244
- 3: “Rendering effective route maps”, M. Agrawala, C. Stolte, ACM SIGGRAPH (2001)
- 4: “Halo: A technique for visualizing off-screen locations”, P. Baudisch, R. Rosenholtz, CHI 2003 (2003)
- 5: «J2ME Game Programming» - Martin J. Wells, 2004 (Thomson Course Technology, ISBN: 1592001181)
- 6: Golf Club Technology, Phillip Eide, John Nelson 2001 - http://www.students.dsu.edu/eidep/technology_paper.htm
- 7: GPS Industries: <http://www.sportandtechnology.com/features/0240.html>
- 8: «From GIS to LBS»: An Intelligent Mobile GIS – Christopher Frank, David Caduff, Markus Wuersch, Intelligent Spatial Technologies & Dep. Of Geography, University of Zürich