



Androidene

Midtveisrapport

INF5261

Jon Lønne og Øyvind Marthinsen

14. mars 2008



Androidene

Innholdsfortegnelse

Introduksjon	1
Forord	1
Bakgrunn	1
Android	2
Hva er Android?	2
Synopsis	2
Hvem står bak	2
Lisensiering	2
Teknisk introduksjon	3
Innledning	3
Arkitektur	3
Utviklingsverktøy	4
Applikasjonenes livssykel	4
Oppsummering	5
Prototypen vår	6
Idé	6
Brukergrensesnitt	6
Litt teknisk	7
Konklusjon	8



Androidene

Introduksjon

Forord

I vår prosjektoppgave ønsker vi å utforske Googles mobilsatsing Android. Android er en mobilplattform som har som mål å gjøre det enklere for utviklere å lage applikasjoner til mobile enheter. Vi ønsker å lære mer om Android gjennom å utvikle en prototype på plattformen: en handleliste for mobiltelefoner. Vi tror en handleliste-applikasjon vil være en nyttig applikasjon for mange familier, samtidig ligger det store utfordringer i å designe den.

Bakgrunn

Gruppen består av to masterstudenter i informatikk. Vi skal skrive en masteroppgave sammen, innenfor mobile systemer. Vårt utgangspunkt for oppgaven er å se nærmere på mobileplattformer. Dette er en av hovedgrunnen til at vi i dette prosjektet har valgt å titte nærmere på Android.



Androidene

Android

Hva er Android?

Synopsis

Android er en mobilplattform utviklet av Google og Open Handset Alliance. Den er basert på Linux, og vil tilgjengeliggjøres under en åpen kildekode-lisens. Android er et operativsystem, mellomvare og tilbyr nøkkelapplikasjoner som telefonfunksjonalitet, kontaktliste og kart.

Telefonens kjernefunksjonalitet er stilt på lik linje med applikasjoner utviklet av tredjeparter, og er med andre ord ment som å være en åpen plattform. Plattformen skal også inkludere alt håndsett-utviklere og operatører trenger for å kunne tilby en telefon på markedet.

Hvem står bak

Open Handset Alliance består av selskaper fra forskjellige markeder. Foreløpige medlemmer som produserer håndsett er HTC, LG, Motorola og Samsung, og maskinvareleverandører som Broadcom, Intel, nVidia og Qualcomm. På operatørsiden er det store navn som NTT DoCoMo, Sprint Nextel og T-Mobile med i alliansen. Siden plattformen skal være åpen, har alle muligheter for å utvikle applikasjoner på plattformen, men det er også flere software-selskaper med i alliansen, som eBay og naturligvis Google.

Lisensiering

Android skal utgis under Apache 2.0-lisensen, som er en åpen kildekode-lisens vennlig for kommersielle formål. I praksis betyr det at tilbydere av håndsett kan utvikle unike applikasjoner og nyvinninger som de ikke trenger å gi tilbake til prosjekte, som igjen betyr at hver tilbyder kan by på sitt unike innhold - samtidig som man har full tilgang til tredjepartsapplikasjoner som er tilgjengelig for alle håndsett basert på denne plattformen. På denne måten kan håndsett-tilbydere og nettverksoperatører tilby skreddersydde løsninger til sine kunder, uten at dette går utover tilgangen på de åpne applikasjonene.



Teknisk introduksjon

Innledning

Android-utviklingen er ledet av Google, med bistand fra Open Handset Alliance.

Utviklingsplattformen er for øyeblikket utgitt for å gi utviklere et tidlig innblikk i hva den har å by på. Når plattformen har en mer ferdig form, har Google gitt uttrykk for at de planlegger å gi ut de fleste komponentene under Apache 2.0-lisensen.

Når man først leser om Android er det kanskje lett å tro at det bare er et operativ system, men Android inneholder også mellomvare og nøkkelapplikasjoner. Det er gitt ut et sett med utviklingsverktøy for å hjelpe utviklingen av Android applikasjoner. Alle Android applikasjoner blir skrevet i programmeringsspråket Java. I tillegg er det lagt opp til bruk av XML, blant annet til konfigurasjon og for å forme layouten.

Vi har brukt Googles offisielle Android-sider som basis for våre undersøkelser av plattformen. Sidene er tilgjengelig på <http://code.google.com/android/>.

Arkitektur

Som tidligere nevnt består Android av et operativsystem, mellomvare og noen nøkkelapplikasjoner. Android sin arkitektur kan deles inn i fem deler: applikasjoner, applikasjonrammeverk, bibliotek, Android kjøretid og Linux-kjernen.

Android kommer med en rekke essensielle applikasjoner for en mobiltelefon: e-postklient, kalender, kontaktliste, Google-kart og en nettleser. Tredjepartsapplikasjonene vil ligge under den samme delen av arkitekturen, og er altså likestilt med applikasjonene Google tilbyr.

Android-arkitekturen inneholder et rammeverk for applikasjoner. De sentrale applikasjonene fra Google, nevnt ovenfor, benytter seg også av dette rammeverket. Målet med dette laget i arkitekturen er å gjøre det enklere å lage applikasjoner. Rammeverket tilbyr alle applikasjoner å dele sin funksjonalitet med andre programmer. I tillegg til dette ligger en rekke tjenester og systemer under alle applikasjoner. Et eksempel er hendelsehandleren som gir alle applikasjoner mulighet til å vise advarsler om hendelser i et statusfelt.



Android inkluderer også flere biblioteker, som er tilgjengelig gjennom applikasjonrammeverket. Et par eksempler her er SQLite, en lettvekt databasemotor og Media-bibliotek som støtter avspilling flere populære lyd- og videoformater.

Android kjøretid inkluderer et kjernebibliotek som inneholder det meste av funksjonaliteten du har tilgjengelig i programmeringspråket Java. Hver Android-applikasjon blir kjørt i sin egen prosess og som en egen instans i en Dalvik virtuell maskin. Denne virtuelle maskinen er optimisert for minnebegrensningene Android-telefoner har, og er ikke kompatibel med Suns virtuelle maskin.

Android er avhengig av Linux-kjerne versjon 2.6 for kjernefunksjonalitet som sikkerhet, minnehåndtering, proseshåndtering, nettverkslag og drivere.

Utviklingsverktøy

Android sitt sett med utviklingsverktøy inneholder flere verktøy som hjelper deg å utvikle mobile applikasjoner på Android-plattformen. De to viktigste er Android-emulatoren som lar deg teste applikasjonen på en virtuell mobil og den andre er en plugin til utviklingsverktøyet Eclipse. I tillegg til disse to finnes det en rekke programmer for å debuge, pakke og installere applikasjonen på emulatoren.

Applikasjonenes livssykel

En applikasjons livssykel er en sentral og viktig del av Android-plattformen. Et viktig element å få fram her er at hver applikasjon i Android kjøres i en egen prosess. Dog dukker det opp en uvanlig egenskap her: applikasjonen har ikke selv direkte kontroll over dens livstid. Dette blir bestemt av systemet på bakgrunn av hvilke deler av applikasjonen som kjøres, hvor viktig disse delene er for brukeren og hvor mye minne som er tilgjengelig i systemet.

Siden systemet kan drepe applikasjoner når det begynner å gå tomt for minne, er det svært viktig at utviklere vet hvordan de skal bruke applikasjons komponenter som virker inn på prosessens levetid. For å avgjøre hvilke prosesser som skal drepes blir de lagt i et hierarki basert på hvor viktig prosessen er. Hierarkiet delt inn i fem forskjellige nivåer:

1. Forgrunnprosess – nødvendig for det brukeren gjør nå.
2. Synlig prosess – inneholder en aktivitet som er synlig for brukeren på skjermen, men ikke i forgrunnen.



Androidene

3. Tjensteprosess – inneholder en tjeneste. Tjenesteprosseser er vanligvis ikke synlige, men de gjør viktige ting for brukeren.
4. Bakgrunnprosess – inneholder en aktivitet som ikke er synlig for brukeren for øyeblikket.
5. En tom prosess – inneholder ingen aktive applikasjonskomponenter. Eneste grunnen for å ta vare på en slik prosess, er for å korte ned starttiden neste gang en komponent i applikasjon skal kjøres.

Oppsummering

Android har et tradisjonelt oppsett som utviklere er kjent med fra før og raskt vil sette seg inn i. Java er også et kjent språk for mange, og likheten med J2ME-stacken (som brukes i ordinære Java-baserte mobilapplikasjoner) er mange. En utfordringen utviklere kommer til å støte på er hvordan livssyklusen til en Android-applikasjon skal implementeres.



Prototypen vår

Idé

Ideen vår går ut på å lage en kollaborativ handleliste, en applikasjon hele familien har tilgang på fra sine håndholdte enheter. Man kan opprette en ny handleliste, og deretter kan alle i familien legge til varer på denne listen. Den personen som skal handle den dagen, kan bruke handlelisten i butikken og "huke av" ting etterhvert som man legger det i den fysiske handlekurven.

Hovedmålet med handlelisteapplikasjon er at den er kollaborativ og enkel å bruke. Den åpner for mange spennende muligheter og kan benytte mange av Androids funksjoner.

Kjernefunksjonaliteten har mange muligheter for kjekke tillegg. For eksempel kan man legge til støtte for maler. En handlelistemalen går ut på at brukeren spesifiserer en hverdagsmal som inneholder brød, melk og andre hverdagslige produkter man vanligvis kjøper. Helgemalen kan på en annen side inneholde vin og lørdagsgodt.

Brukergrensesnitt

Etter å ha drodlet rundt handlelisteapplikasjonen, kom vi frem til at det vi aller mest ønsket var å fremstille handlelistene ved hjelp av ikoner for varene. Vi har prøvd å vise dette i den vedlagte mockup'en av prototypen.

Tanken er at man skal dra en og en vare ned i handlekurven nederst for å legge til ting i handlelisten. På samme måte ser vi for oss at applikasjonen kan fungere i butikken; varene tas ned i handlekurven etterhvert som de fysisk legges i kurven, og varen blir borte fra listen. På en enkel måte kan man da se hva man har glemt å kjøpe. Vi har lyst til å utforske denne muligheten, da den benytter seg av Androids touch-funksjonalitet.



En av utfordringene med å bruke ikoner til hver vare, er at alle varer må da ha et bilde og være lagt inn i applikasjonen. En måte vi tenkte å løse dette problemet på, er å la brukeren



Androidene

legge inn flere varer ved hjelp av mobilens kamerafunksjonalitet. Dette kan gjøre ting vel avansert for vår prototype, men det er i hvertfall muligheter om vi senere skulle bygge ut funksjonaliteten videre ved et senere tidspunkt. At brukeren legger inn varer kun som tekst kan være en enklere løsning.

En annen mulighet er å bruke lister. Med dette tenker vi et skjermbilde der tingene du skal handle ligger i en liste. Man kan så sette et merke ved de tingene man putter i handlekurven. Dette er en enklere og mer tradisjonell form enn ikoner. Folk er allerede vant til å håndtere lister når de er ute å handler, derav navnet handleliste. Om vi velger å bruke lister i vår applikasjon, ville det sannsynligvis være enklere å lage applikasjon. Dog ville den lide av at den ble mindre visuelt attraktiv og muligens ikke like intuitiv. Et annet argument for bruk av lister er at det også ville fungere på eldre mobiltelefoner.



Vi synes bruk av ikoner er et mer interessant og tidsriktig utgangspunkt. Det gir oss også mulighet til å utforske hvordan støtten for et slikt brukergrensesnitt er i Android.

Litt teknisk

En av de største utfordringene er å gjøre handlelisteapplikasjonen kollaborativ. Vi ser for oss at dette kan løses ved å bruke en web service som lagring. Selv ved bruk av web services er



Androidene

det nødvendig å ha en datastruktur i mobilapplikasjonen for mellomlagring (caching), og vi har derfor valgt å begynne med å lage en simpel ikke-kollaborativ prototype i Android. Hvis vi får tid etter vi har utviklet prototypen for én bruker/enhet, har vi lyst til å implementere en web-basert backend for lagring og utveksling av data mellom forskjellige håndsett.

Konklusjon

Vi synes Android virker som en spennende mobilplattform og gleder oss til å utforske den mer i detalj. Android bruker godt kjente teknologier, så vi håper dette hjelper oss med komme raskt i gang med vår prototype. Handlelisteapplikasjonen vår er også noe vi ser nytten av og har lyst til å utvikle. Det er mulig ikke all funksjonalitet vi ønsker kommer med i denne omgang. Dog byr den på mange problemstillinger både i form av tekniske utfordringer og designutfordringer som er relatert til vårt prosjektet.