

## Grupp 14

Henrik Andersson, Anton Palmqvist, Tomas Selldén & Marcus Tyrén

# Reflektionsdokument: Projektarbete SE Android

- **Har kursen motsvarat våra förväntningar?**

Vi är medvetna om att kursen ändras lite varje år för att få den så bra som möjligt och det medför en del oklarheter. Vi visste t.ex inte i början om det var en projektkurs eller programmeringskurs, då kurskraven ändrades ofta även om de var utförliga. Det har också varit en del oklara instruktioner, då man kunde få olika svar på en fråga beroende på vem man pratade med. Vi hade gärna önskat genomgångar av Android på föreläsningarna. Kursen har hursomhelst varit både kul och lärorik, då man fått testa på både Scrum och Android som bägge var nytt för samtliga i gruppen.

- **Vilka förkunskaper har vi haft nytta av?**

Alla har några javakurser i ryggen och dels att vi jobbat tillsammans i projekt i våras. Vi har dessutom jobbat mycket med Git och är ganska duktiga på det.

- **Vad har vi lärt oss?**

Innan kursen hade vi ingen koll alls på Android och sättet man jobbar med det. Anton hade visserligen smygstartat lite i somras men var förhållandevis grön vid starten han med. Allt efter kursens gång har vi blivit bättre och bättre på det men måste erkänna att mycket av tiden har gått åt att läsa in sig på dokumentation, googla om androidutveckling och titta på tutorials och läsa på om hur saker fungerar för att kunna få en bild av vad som ska göras och hur det ska göras. Det hade varit önskvärt med föreläsningar i just android och inte bara i Software-engineering. Genomgångar på hur man bäst hanterar activities och byte mellan dessa med intents hade varit en bra föreläsning och sparat oss många timmars googlande som istället hade kunnat läggas på att koda.

Ett exempel är att det enda vi egentligen visste om android innan var att det var javakod och att det fanns xml filer. Dock var det oklart vad en xml fil var och hur den påverkade programmet. Efter någon vecka insåg vi någorlunda hur allt hängde ihop. Detta gjorde det omöjligt att planera kodstruktur innan projektet drog igång. Sedan bestämde vi oss för att använda databaser i vår applikation, något som ingen heller hade arbetat med tidigare. Det tog mycket tid att sätta sig in i SQLite men vi lärde oss mycket och i slutändan blev det bra. Vi har alla blivit bra på att lösa problem på egen hand i en värld man är helt ny i samt lärt oss att det finns inga problem som inte går att googla! Vi har lärt oss otroligt mycket om att hitta i API:er samt att ta till oss information från tutorials och från olika forumtrådar.

- **Vad gick bra?**

Samarbetet i gruppen har varit bra och även om vi stötte på problem har moralen alltid varit hög och vi har hjälpts åt för att få det att fungera. Arbetet i Scrum tog sig också i slutet såväl som våra färdigheter i Android.

- **Vad kunde gått bättre?**

Vi var dåliga i början med att bestämma vad som ska göras, ingen bra plan för strukturen på programmet och på databasen vilket har lett till att det har varit tvunget att ändras radikalt under projektets gång. Vi bestämde aldrig hur vi skulle skicka värden från databasen. Tomas skickade strängar och Marcus trodde att han skulle få geopoints.

Med facit i hand hade det varit mycket enklare att skicka t ex Geopoints från databasen istället för att behöva hålla reda på strängar med lösa koordinater. Har t ex spenderat en hel dags arbete för att få ett Set<Integer> med koordinaterna att fungera. Detta fungerade dock aldrig eftersom koordinaternas interna ordning är av betydelse. När de flyttades från arrayen till Set flyttades ordningen om vilket ledde till att EDIT huset ritades ut någonstans mellan Ryssland och Nigeria.

Vi "råkade" ut för lite dubbla beroenden som inte är så snyggt, men lite mer tid ägnat åt dem hade vi blivit kvitt det.

- **Vad för slags problem stötte vi på?**

Många! Ett problem som gäckade oss var när R.java-filen försvann och koden inte längre gick att kompilera. Vi vet nu att det oftast orsakas av fel i .xml-filer och feldöpta filer i res mappen, men innan vi visste det och innan vi hittade problemet var det svårt och det fanns dålig information om det felet. Det ska sägas att vi blivit duktiga på att felsöka och på så vis ökat känslan och fått ett öga för det.

- **Vad var mest tidskrävande?**

Att sätta sig in i Android, Google API, GPS-API och databaser tog säkert fyra femtedelar av arbetstiden. Sedan under utveckling hela tiden googla för att få svar på frågor när man fastnat, då ingen i gruppen visste särskilt mycket mer än någon annan. Därav hann vi inte med att implementera alla de features vi först tänkte och det blev inte så mycket kod som vi hoppats på.

- **Hur gick Scrum?**

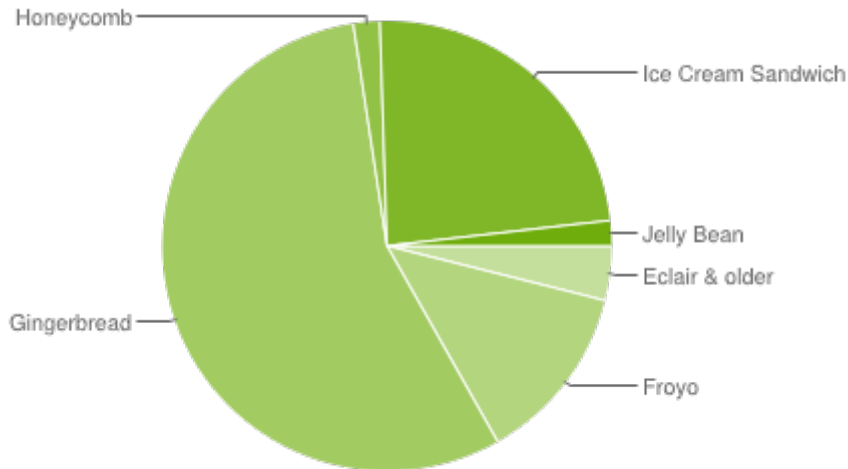
I början var det ovant då vid det enda tillfälle som vi jobbat med projekt tidigare hade innefattat dokumentation med vattenfallmetoden. Vi gillade arbetssättet med korta sprintar och uppdelade arbeten med estimerad tidsåtgång. Man jobbade i små intervaller och det kändes som man bit för bit kom allt närmare målet, den slutliga applikationen. Skillnaden där man i vattenfall kan känna att sträckan från start till slut är lång känner man i Scrum att man lyckats komma framåt stegvis varje vecka. I början gjorde vi dock alldeles för övergripande User Stories, men ändrade dem senare till lite mindre. Vår ScrumPlanning gick bra fast att vi tyckte att det var väldigt svårt att uppskatta tidsåtgång för något man nästan aldrig eller aldrig sysslat med tidigare. Det var tråkigt när en uppgift sträckte sig över flera sprintar men har fått lära oss att det är ganska vanligt när man jobbar enligt Scrum.

- **Blev appen som vi tänkte från början?**

På sätt och vis. Vi hade önskat att vi lyckats anamma Android lite bättre så att vi kunde hinna implementera fler av våra tänka features. Eftersom vi inte visste hur lång tid allt skulle ta var det svårt att se hur mycket man skulle hinna göra. Den skulle vara enkel att navigera i och vi tycker att vi lyckats med

det. Detaljer och utseende kan alltid slipas på men vi ville inte prioritera det för mycket då det inte skulle väga in i betyget såvida inte det var något spektakulärt.

I applikationens specifikationer satte vi minimum Android API från 8 (Froyo) upp till 16 (Jelly Bean) eftersom de flesta som har Android då kan använda appen.



**Diagram över användare av olika de versionerna av Android**

- **Prioriteringar**

Vad gäller testning valde vi att prioritera bort testning av maps och GPS. Detta var dels för att då dessa är färdiga bibliotek så utgår vi från att dessa uppfyller krav och uppfyller tester med råge. Vi hade gärna testat detta ändå tillsammans med den logik som används i våran applikation men på grund av tidsbrist blev det tvungen att lägga tid på annat.

Vidare har vi valt att lägga extra prioritet på dokumentation. Detta är eftersom vi insåg att vi inte skulle kunna komma upp i de antal rader kodrader vi hoppats på på grund av den långa inlärningskurvan i början av projektet. Därav lades extra fokus på att verkligen göra dokumentationen ordentlig, strukturerad och lättbegriplig. Särskild guldkant lades på User-guiden där alla steg är noggrant ordnade steg för steg med pedagogiska screenshots med orangea markeringar på vad som skall tryckas på i de olika stegen och för att markera var förändringar i programmet sker vid utförande av manövrarna i de olika stegen.

- **Exempel på features som man kan bygga ut med?**

Att använda accelerometern för att bestämma på vilket våningsplan man befinner sig på i en byggnad och på så vis få rätt inomhusvy visad.

Options menyn kan byggas ut med lite olika valfria inställningar. Saker som kan tänkas lägga där är att välja satellitvy som default, välja att visa toast message, olika språk.

Man skulle kunna lägga till fler klassrum på skolan och skulle då också kunna använda ett filter för att bestämma vilka man vill visa.

I browsevyn ska sökfältet filtrera klassrummen i listan dynamiskt med avseende på den ordning bokstäver skrivs in.

Vi började utveckla en klass som skulle känna av om man var i en byggnad, alltså om en location befann sig inom en byggnads koordinater. Men denna skrotades eftersom det gick dåligt och tog för mycket tid för en feature som i slutändan inte är speciellt värdefull för appen som helhet.

En annan önskvärd funktion vi vill bygga ut med är att få rutten för vägbeskrivning dragen längs gator och inte fågelvägen som nu är fallet.