

Postmortemrapport

Så här gjorde vi

Med en grupp som bestod av fem medlemmar, och en ständig tillgång till bra lokaler att vistas i, tänkte vi att vi jobbade ihop, med täta möten där programmering och diskussioner byter av varandra. Vi jobbade då efter ett dokument på Google drive som vi kallade för mötesprotokoll. Där planerade vi möten i förhand med de punkter som vi tänkte skulle tas upp under mötet. Dessa punkter kunde vara allt från frågor som skulle diskuteras, till kod som skulle skrivas. Efter varje möte fylldes protokollet också upp med punkter och beskrivningar av vad som faktiskt gjordes, samt vilka beslut som fattades. Mötesprotokollet blev även en tidplan som visade hur mycket tid vi lagt ner på projektet varje dag.

Punkterna som skrevs upp inför kommande möten baserades ofta på den backlog som vi förde i ett separat dokument. I backlogen delade vi in olika funktionaliteter och delar av appen i kategorier efter deras prioritet, och valde därifrån vilka hinder vi skulle tackla närmast. När de viktigaste punkterna var färdiga togs de bort från toppen av backlogen och de som tidigare haft lägre prioritet flyttades upp. När vi stötte på nya oväntade problem skrevs dessa in i lämplig prioriteringskategori i backlogen. När projektet gick mot sitt slut skapade vi nya backlogdokument för specifika ändamål, tex grafisk design.

Vi hade vårt första stora möte under andra veckan där vi gick igenom appens alla beståndsdelar och skapade ett dokument som reflekterade hela gruppens syn på appen. Efter detta mötet kände vi att alla hade en och samma syn på appen och kunde börja ifrån samma utgångspunkt.

Vi gjorde en hel del live-tester under projektets gång. Med detta menar vi att vi testade appen, med telefoner på bussen och busshållplatser. Eftersom flera funktioner i vår app kräver två telefoner för att testas gick det inte att köra enkla unit-test eller motsvarande för att se att just de funktionerna fungerar. Därför var vi tvungna att ägna mycket tid åt dessa live-tester. Då vi höll våra möten på Lindholmen, kunde vi sätta oss i inomhushållplatsen på teknikgatan, vilket förenklade en del. Dessa skedde då i samband med möten.

Den mesta tiden i projektet lades därför ned i dessa möten. Vi hade heldagsmöten, halvdagar och även korta möten där bara ideer diskuterades. Mötena var även tvungna att anpassas efter medlemmarnas diverse andra kurser. Tack vare att flera av oss läste samma kurser gick det dock lättare att planera in möten. Tanken var att vi skulle sträva efter att samlas 2-3 heldagar i veckan men eftersom det framförallt i början var svårt att få till, samlades vi oftare och hade kortare möten. Schemakrockarna minskade allt eftersom och mötena kunde bli längre, detta gjorde dock inte att vi träffades mer sällan, vilket resulterade i att vi oftast gick över den tänkta arbetstiden.

Vi har använt git i tidigare projekt, men det här var första gången vi tog tag i det och försökte lära oss det ordentligt. Under första veckan lade vi ner mycket tid för att alla gruppmedlemmar skulle bli bekanta med git. Det räckte för att git inte skulle kännas som ett hinder, men vi har ändå insett att mer jobb med det krävs. Vi hade en person som agerade git-master, som tog hand om alla merges till master, och hanterade konflikterna.

Nedlagd tid

I början av projektet varierade tiden vi lade ner ganska mycket. De första två veckorna var våra möten bara tillfällen att tänka ut vad vi ville göra, hur vi skulle ta oss dit och vad som behövde göras. Tanken var att öppna upp för mer tid till alla gruppmedlemmar att lära sig Android och git. Detta medförde att vi förmodligen hamnade en bra bit under 20-timmarsgränsen. När vi efter de två veckorna väl kom igång med koden, blev tiderna annorlunda. Våra möten gjordes om till att vara programmeringstillfällen, där vi kunde sitta tillsammans i ett klassrum och hjälpas åt när det behövdes.

Under några veckor ökade takten ganska snabbt, och från mitten av projektet fram till slutet snittade vi ca 30 timmar per vecka. Detta är bara tid vi lagt på möten i skolan, och vissa veckor fortsatte folk hemma, ibland även på helgen. Detta skedde inte för att vi kände oss stressade, och det fanns inget gruptryck att lägga ner extra tid hemma. Istället skedde det för att vi tyckte projektet var otroligt roligt att jobba med, och vissa dagar var det svårt att slita sig, speciellt när man var nära ett genombrott.

Fördelar och nackdelar med arbetsättet

Eftersom våra gruppmedlemmar hade olika mycket kunskap om programmering, var det fördelaktigt att jobba nära varandra så att frågor på ett snabbt och enkelt sätt kunde bli besvarade. Dessutom hade vi alltid möjlighet att snabbt byta arbetsuppgifter ifall man blev klar, eller körde fast, och alla gruppmedlemmar fick en god insyn i hur de andra låg till med sina delar. Alla gruppmedlemmar hade en ständigt uppdaterad syn på projektet i sin helhet, vilken riktning och vilket mål det hade, samt hur långt vi hade kommit, och alla motiverades att själva hålla uppe arbetstakten när man märker hur de övriga i gruppen diskuterar.

Tekniken vi använde oss av var framförallt utformad efter våran grupps storlek. Hade vi varit en större grupp hade mötena antagligen blivit mindre produktiva och svårare att organisera. Dessutom hade det blivit mycket svårare att finna tid för så många och långa möten med fler personer att anpassa schemat efter. Men för en liten projektgrupp fungerade denna tekniken utmärkt, och vi skulle förmodligen använda den igen, om än med lite annan struktur på hur vi dokumenterade varje möte. I början var mötesprotokollet ganska avskrapat och användes mest som en lista över ämnen att ta upp, samt vilka dagar vi skulle ses. I takt med projektets fortskridning utvecklade vi det vidare med reflektioner från möten, starttid och sluttid, vilka som var där och vilka tider de var där. Det blev även ett sätt för de som inte var där att följa med i händelseförloppet.

En nackdel med arbetssättet var att vi upplevde att gruppmedlemmar som missade ett möte gick miste om väldigt mycket, oavsett hur detaljerat vi skrev mötesprotokollet. Har man någon i gruppen som föredrar att sitta hemma och arbeta kommer den personen hamna efter. Effektiviteten och de snabba vändingarna i en sammansvetsad grupp blir då en nackdel för den som inte är där. Eftersom beslut togs och problem löstes i en snabb takt gick man lätt miste om mycket av förståelsen och helheten i projektet. Detta blev tydligt då vi just hade en person som inte bara föredrog att sitta hemma och koda, utan också blev sjuk under en dryg vecka.

Vår användning av git skulle definitivt gå att förbättra. Utöver de absoluta basfunktionerna, som olika branches, använde vi inte git så mycket. Under de första tre veckorna arbetade vi utan en

gitignore-fil, eftersom vi inte hade läst på tillräckligt mycket för att veta att den fanns. Att vi bara hade en git-master, och inte använde "pull requests", gjorde oss dessutom sårbara, om till exempel git-mastern skulle blivit sjuk. Många av våra branches var dessutom ofärdiga när de pushades och mergades.

Reflektioner kring arbetssättet

Vi är ganska nöjda med vårt arbetssätt. Om vi skulle påbörja ett nytt projekt hade vi följt ungefär samma stil, men med vissa förändringar. En del av nackdelarna som tas upp ovan skulle gå att lösa med mer struktur och mer krav. På så sätt skulle vi närma oss Scrum lite mer, som är ett beprövat och fungerande sätt att driva utveckling. Framför allt tänker vi att vi hade infört obligatoriska möten, eller åtminstone obligatoriska delar av mötena. Detta till trots vill vi inte ändra på alltför mycket av vårt arbetssätt, då vi alla hade väldigt roligt under projektet. Denna glädje som vi upplevde kunde ibland uttrycka sig i att vi pratade, eller rentav tramsade, för mycket och jobbade för lite. Samtidigt som man gärna vill vara produktiv känns det ändå viktigt, och kanske rentav viktigare, att alla känner att det faktiskt är roligt och givande att vara där. Det är en svår balansgång, mellan regler och struktur å ena sidan, och glädje och gemenskap å andra sidan.

Reflektioner kring hur vi gjorde och varför

Vi valde att lägga de första två veckorna av projektet på att skissa på vår vision, bekanta oss med Android studio och git, samt planera arbetet. Vissa i gruppen hade redan lite vana med de verktygen medan det var helt nytt för andra. Vi fattade beslutet att ha flera möten i veckan, samt skapa ett mötesprotokoll, tidigt i projektet, någonting som vi sedan använde oss av under hela utvecklingen. Att det skulle fungera med dessa täta, långa möten var någonting vi dels baserade på att vi under tidigare projekt arbetat på liknande sätt med framgång men också på att det påminner en del om Scrum i att vi har möten och en backlog.

Någonting annat som vi gjorde tidigt under projektet var att försöka komma på olika tekniska eller konceptuella problem med vår idé. Att hitta tekniska problem med en idé eller ett koncept, när man aldrig tidigare har gjort något liknande, var svårt. Försöket var inte meningslöst; vi lyckades identifiera några problem, men vi kanske inte hade behövt lägga så mycket tid på det. Vi fattade ganska tidigt beslutet att vi skulle lägga mycket tid på planeringen. Detta underlättade när vi väl började skriva kod, då vi redan hade en uttänkt struktur och till och med ett minimalt UML-diagram att utgå ifrån. Vi lärde oss att oavsett hur mycket vi planerade, kunde vi inte förutse vilka delar som skulle ta mest tid eller vara svårast att slutföra. Dessutom lärde vi oss att också inse när man har kört fast och bör leta efter alternativa lösningar.

Då vi hade en del bekymmer med nätverket (se dokumentet "Utvecklingsproblem.pdf") och det enda sättet att debugga och att försöka lösa dessa var att debugga på busshållplatsen så lades mycket tid ned där. Detta gjorde ju att gruppen splittrades upp. Några fortsatte med appens funktionalitet och de andra debuggade nätverksdelen på hållplatsen. Att sitta och debugga på en busshållplats visade sig inte vara optimalt, då bussar och människor kom och gick, som inte bara störde arbetet utan också gjorde debuggningen svårare. Detta på grund av ständigt ändrade accesspunkter vid varje ny buss. Att först få appen att fungera på egna nätverk, för att sedan upptäcka att det inte fungerade alls på ElectriCitys var en kämpig process.

Gruppen och hur det fungerade

Arbetet i gruppen fungerade bra genom hela projektet, och då vi skrev den mesta koden och gjorde i stort sett allt på gruppmöten, var detta någonting som var extra viktigt. Hade gruppkemin inte funnits där hade projektet blivit mycket svårare att genomföra, åtminstone med det arbetssättet vi valde. Men vi tror att en av anledningarna till att vi fungerade så bra som grupp var just att vi jobbade väldigt mycket tillsammans. Man lär sig snabbt var gruppmedlemmarna står i respektive kunskapsnivå, hur de arbetar, och kan därefter dela ut uppgifter som passar respektive person.

En så välfungerande gruppdynamik visade sig dock vara ett tvåeggat svärd, då många diskussioner hade en tendens att bli längre än nödvändigt, och gå in på andra områden än de som diskuterades. Många diskussioner fortsatte även efter vi nått en lösning och tog därför onödigt mycket tid.

Hade man haft någon som stod lite utanför som kunde hålla diskussionen från att spåra ur, hade vi kunnat spara en del värdefull tid. Istället var alla i gruppen lika inne i diskussionerna, och ibland var det svårt att tyda om det som diskuterades var vettigt och relevant för projektet som det såg ut då, eller om vi diskuterade problem och lösningar som vi ändå aldrig skulle få tid att implementera.

Hur vi ställde oss till de olika workshops som organiserades

Alla workshops gick vi igenom i god tid, vi tog reda på vad de skulle handla om, och därefter fick alla själva bestämma om man ville gå av eget intresse. Då ingen av dessa workshops tog upp någonting som vi tyckte var intressant för vårt projekt fick dessa ingen prioritet, utan sågs som något man kunde gå på om man kände för det. Vissa workshops var också lagda på kvällen, vilket inte heller passade alla i gruppen. Vi försökte också hålla på tiderna, så att vi inte behövde jobba kvällar och helger.

Vi tog tävlingen seriöst, på det sätt att vi hade en idé som vi verkligen trodde på, och vi tänkte köra ända in i kaklet för att realisera den. En sak vi hade kunnat göra för att ha en bättre chans att vinna tävlingen vore att lägga mer tid på dessa olika events, gå på workshops och där göra reklam och få feedback på idén vi hade. Där missade vi förmodligen några chanser. Men anledningen varför vi inte gjorde detta var just på grund av att dessa workshops hade strikta scheman med vad som såg ut att vara föreläsningar som behandlade ämnen vi inte kunde koppla till vårt projekt. Den enda workshopen, som vi kände att det var synd att vi missade, var den sista, där fokus låg på tävlingsdeltagarna och deras idéers förverkligande.

Något annat vi kunde lärt oss från workshops var att förklara vår idé. Detta var något vi insåg först dagarna innan finalen; att vår idé inte var den lättaste att förklara för någon som inte var insatt i projektet. Hade vi varit på workshops, där vi hade behövt förklara och beskriva vår idé så hade vi kanske tidigare tagit tag i hur vi skulle pitcha och presentera idén och prototypen. På grund av detta har vi verkligen tagit till oss att oavsett hur självklart något verkar för dig själv och gruppen, behöver så inte vara fallet för andra, och vi har förstått vikten av att faktiskt visa det för utomstående för att veta vad som inte framgår.

Vad vi hade gjort annorlunda

Till nästa projekt tänker vi oss att skapa ett utförligt mötesprotokoll tidigare, där vi i god tid bestämmer när möten skulle äga rum och vad som skulle göras. Det är viktigt att dokumentera vad man har gjort eller ska göra, vi tycker att det kunde gjorts bättre. Till exempel om en

medlem sitter hemma och arbetar, blir det svårt för denne att ha koll på vad de andra medlemmarna har gjort, och vice versa. Med bättre dokumentation minskar man dessa problem något.

Vi känner även att vi hade kunnat gå på de workshops som man fick tillgång till under kursens gång, eller lagt tid på att lära oss förklara vår idé för andra. Vi hade även kunnat göra lite mer research över vad potentiella framtida användare känner om vårt koncept och vision. Till exempel hade man kunnat åka med elbussarna och fråga resenärer om de hade använt en app med våra egenskaper. Man hade då kunnat ta åt sig feedbacken och tänka över vissa designval och egenskaper.

I början av projektet bestämde vi oss för att var och en skulle sitta själv och bekanta sig med Android studio och hur man programmerar Android överlag. Det var en bra idé, men vi drog ut på det lite för länge. Istället för att detta skulle ske under en hel vecka, hade det räckt med ett par dagar. I nästa projekt borde vi vara snabbare på att be om hjälp när vi stöter på problem, framför allt med utomstående produkter. Till exempel hade vi problem med Icomeras routrar, som vi försökte lösa på egen hand, istället för att helt enkelt fråga hur vi kunde lösa problemen.

Till nästa gång har vi lärt oss att lägga mer energi på att få till en bättre presentation och en bättre samt mer informativ projektfilm. I efterhand kan man se tillbaka på vår projektfilm och inse att den inte gav all information som den hade behövt, speciellt inte som stöd för vår presentation inför juryn. Inför presentationen förberedde vi oss på alla frågor vi kunde komma på, men vi borde också ha försökt presentera projektet inför familj och vänner, så vi kunde komma på svar även till utomståendes frågor.

Användandet av git i nästa projekt kommer automatiskt bli annorlunda. Vi har lärt oss mycket om hur man använder git och vilka funktioner som finns. Tack vare detta kommer vi vid liknande projekt få mycket lättare att redan från början använda oss av versionshanteringsprogram på ett mer korrekt sätt. Vid nästa projekt kommer vi med en gång sätta upp en gitignore-fil, samt använda branches och rebase-kommandot mer. Vi skulle även kunna ha högre krav på kvalitén på branches innan de mergas in, samt merge oftare och med mindre delar på varje branch.