

Project reflections

(E.g what tool did you use, how many were involved, did you version control throughout the process, was a text editor a good tool to use....)

1. Difference between individual programming and working in a team

Projektet med att programmera en android-app har varit en lärorik och utmanande uppgift. Vår projektgrupp har bestått av fyra personer som alla studerar på Industriell ekonomi. Vår bakgrund och erfarenhet när det kommer till programmering och liknande uppgifter skiljer sig ganska markant. Det har därför varit en spännande del av projektet att få våra kompetenser att komplettera varandra så att projektets framfart inte bromsats in för mycket. En övergripande reflektion är att det är roligare att programmera i grupp än vad det är att programmera enskilt. Det är inte i alla situationer fördelaktigt men totalt sett har det bidragit till ett bättre resultat.

Att programmera i grupp innebär att en social aspekt adderas till projektet vilket för samtliga gruppmedlemmar är något som är högt uppskattat. Det tillför en dimension till projektet som inte hade varit möjligt vid individuell programmering, helt enkelt att det är trevligare att sitta tillsammans med kamrater runt ett bord jämfört med att sitta ensam framför datorn. Sedan är det roligare av den orsaken att projekt med både större djup och bredd hinns med på samma tid. Fler avancerade funktioner och en bättre app är helt enkelt möjlig att göra vilket bidrar till att det blir mer intressant, större motivation och bättre chans att få den kommersialiserad och konkurrenskraftig bland andra list-appar.

I ett tidigt stadium genomförde gruppen en kompetensinventering för att kartlägga förkunskaper och tidigare erfarenheter. Det visade sig att programmeringskompetenserna spretade ganska mycket. Det utnyttjade vi genom att tilldela ansvar för olika delar av projektet till olika personer. Gruppen är enad om att vi genom detta lyckades erhålla synergieffekter och att den magiska ekvationen att $2 + 2 = 5$ gäller, det vill säga att flera personers input bidrar till en bättre total output, för programmering som genomförs gruppvis. Individuell programmering har, å andra sidan, fördelen att ingen tid behöver läggas på att förklara och det finns inte heller någon risk att det blir "för många kockar och därför sämre soppa".

En svårighet som gruppen delvis upplevt är att utan en erfaren projektledare som genomfört liknande programmeringsprojekt tidigare följer en hel del utmaningar. Det första och kanske tydligaste var svårigheten att veta i vilken ände och med vad vi skulle börja. Med en erfaren projektledare hade denne förhoppningsvis kunnat bidra med mycket stöd och vägledning framförallt i det inledande stadiet. Med individuell programmering å andra sidan är programmeraren sin egen projektledare, vilket gör att nära vägledning av en annan part är ännu svårare. Det borde därför vara så att vid enskild programmering är bekanta uppgifter mer förekommande och som för programmeraren är enklare att utföra. Vid grupprogrammering är sannolikheten säkert större att svårare uppgifter är beställda.

För att vi som grupp skulle vara så effektiva som möjligt lades stort fokus på planering och kommunikation. Det tog en del tid, framförallt i ett tidigt stadiet, men det vann vi tillbaka i slutet. Gruppen bokade också in datum och tider då vi satt tillsammans ganska långt i förväg. Det var en viktig sak att göra eftersom tiden annars ganska snart blir uppbokad av annat. I samband med detta sattes också flertalet deadlines upp vilket var en viktig nyckel för projektets

kontinuerliga framfart. Tydliga deadlines är viktigare vid gruppvis programmering eftersom olika delar av kodavsnitt, klasser etc beror på andras bidrag. Om då inte deadlines följs kan projektet tvingas ta en paus. Av naturliga skäl är vikten av detta inte lika stor vid individuell programmering.

Kommunikationen var också viktig ur ett samarbetsförstånd och för att undvika missförstånd samt att gemensamma mål och krav sattes upp. Det är en nackdel med programmering gruppvis att missförstånd och konflikter kan uppstå. För att det ska undvikas är det viktigt kontinuerligt stämma av och tidigt ta upp saker till ytan som inte fungerar. Låter man det gå för långt är risken att skadan förvärras och att man till stor del får börja om. Kommunikationsarbete är självklart mindre viktigt vid individuell programmering. Dock ska planering inte underskattas även för individuell programmering, den kanske inte behöver vara lika formell och dokumenterad men den är ändå viktig för att kunna genomföra projektet strategiskt effektivt.

Vid gruppvis programmering sätts det vidare höga krav på flexibilitet och en tillmötesgående attityd. Situationer uppstår alltid när grupper av människor samverkar och kanske ännu mer frekvent när komplexa problem ska lösas genom programmering. För att projektet då inte ska fallera och förlora styrfart är dessa egenskaper centrala.

Kodspecifikt blir det viktigare med tydlig, enkel, kommenterad och intenderad kod ju fler personer gruppen består av. Inte minst när man använder sig av versionshantering, annars blir det jättesvårt för andra personer att sätta sig in i och använda koden.

2. Methods used to create the documents

Projektgruppen har i stor utsträckning använt sig av internetbaserade verktyg för att skapa de dokument som rymmer inom ramen för denna kurs. Mest använt har de verktyg varit sm google tillhandahåller, det vill säga google docs och av dem i första hand "Documents" men också "Spreadsheets". Projektgruppen ansåg att möjligheten till att simultant kunna dela och arbeta med samma dokument var en stor fördel. Det blir en gemensam visualiseringsyta som bidrar till överblick och effektivitet men också enklare kommunikation. Inget behov av mail finns för att skicka filer och inte heller något behov av att klippa och klistra i dokumenten uppstår. Dessutom finns det goda och användavänliga alternativ för att kommentera och diskutera text inne i dokumenten. Arbetet med att lägga till spårbarhet mellan krav och testfall med hjälp av google docs var ett enkelt jobb att göra; kraven och testfallen var tydligt presenterade i olika dokument som alla hade tillgång till vilket gjorde att spårbarheten och synkningen med det var väldigt smidigt. Med andra ord har användandet av google docs inneburit stora fördelar för projektgruppen vilket gjort att resultatet påverkats till det bättre.

Värt att nämna är även kommunikationsprogrammet Skype som haft en central roll i detta projekt. Projektgruppen har ömsom suttit tillsammans och ömsom arbetat på enskilt håll. I de fall vi suttit åtskilda har Skype fungerat som en bra plattform att kommunicera ifrån. Skillnaden och nackdelen mot att sitta tillsammans kunde på detta sätt minimeras vilket möjliggjorde färre misstag och ett roligare projekt.

De dokument du skriver inom ramen för projektet, t ex testplan, design/arkitektur.

Fanns det något bra eller dåligt med de verktyg (google docs, libre office, word ..). Utgjorde dessa verktyg ngt hinder? Var det lätta att lägga till spårbarhet mellan krav och testfall?

3. Appropriateness of the adopted SE techniques

Detta projekt är genomfört utan att strikt följa en viss mjukvarutvecklingsteknik. Projektgruppen

har tagit inspiration från båda vattenfallsmetoden och scrum men undvikit att fullt ut låsa sig till en av dem. Projektgruppen vill i detta avseende vara aningen självkritiska eftersom den här typen av tekniker ofta når sin fulla potential om de används fullt ut; används de endast delvis riskerar de att brista i effektivitet. Trots detta menar projektgruppen att det sätt som vi arbetat efter varit lämpligt.

Vare sig man vill eller inte innebär användandet av nya utvecklingstekniker en inkörsträcka eftersom det kräver inläsning och ett ganska strikt förhållande till ramar och regler. Projektgruppen upplevde därför att det skulle inskränka på effektiviteten att helt förhålla sig till en specifik teknik. Eftersom personerna i projektgruppen sedan tidigare kände varandra väl, både till person och varandras kompetenser, anågs det med lämpligt att anpassa upplägg och uppgifter för att på så sätt tillvarata gruppens totala resurser.

Hursomhelst, projektgruppen inspirerades av vattenfallsmetoden på några punkter. Att arbeta med tydliga faser liknar det sett som vattenfallsmetoden bygger på, det vill säga att sekventiellt färdigställa faser i tur och ordning. Projektet är ju uppbyggt i ett antal faser som hänger ihop med ett antal releaser. Varje fas innehåller en viss funktionalitet och som i fas efter fas utvecklas och förfinas. Många av de faser som traditionellt sett också ingår i vattenfallsmetoden har också ingått i detta projekt. Kravspecifikation, design, kodning och testning har varit centrala processer i detta projekt och de har också mer eller mindre genomförts sekventiellt. Däremot har inte underhåll ingått som en planerad del i projektet. Vattenfallsmetoden har med andra ord delvis varit lämplig att utgå ifrån och får därför anses som viktig för resultatet. Projektgruppen är dock ense om att den inte lämpade sig för att fullt ut utnyttja, eftersom projektet delvis haft en iterativ process bortsett från faserna och dess releaser.

När det kommer till scrum finns det vissa aspekter som vi anammat. Först och främst liknar scrum vår projektgrupp i det avseendet att gruppen består av ett tvärfunktionellt team med individer som alla besitter olika kompetenser. Vissa har till exempel mer erfarenhet av programmering och därför tagit ett större ansvar på det området medan andra har erfarenheter från att leda projekt och därför tagit ett större ansvar för det, ungefär på samma sätt som scrum skulle beskriva en scrum master.

Vidare bygger scrum på inkrementell och iterativ utveckling vilket projektgruppen också varit noga med att också använda. Appen har byggts ut i små steg, ett efter ett och på så sätt har projektet kontinuerligt tagits framåt i rätt riktning i en ganska konstant hastighet, utan större avbrott. Varje fredag har projektgruppen haft en intern sprint, det vill säga att vi har planerat för nästa steg och identifierat de uppgifter som ska göras närmast. Likaså har vi haft en form av daily scrum med många men korta återkommande möten där vi uppdaterat varandra samt definierat uppgifter för var och en. Detta har varit positivt för kommunikationen och effektiviteten i projektet.

På detta sätt tycker vi att tekniken vi använt fungerat väl, vi har plockat de delar ur både vattenfallsmetoden och scrum som vi ansåg lämpade sig väl för vårt projekt. Man skulle kunna säga att vi skäddarsytt en teknik som som fungerat väldigt väl och som hämtat inspiration från flera olika tekniker.

4. Coverage

Patrik får skriva ngt här..

5. General comments about the project as a whole

Kursen har varit givande och rolig. Skapandet av en app har varit lärorikt och bidragit till ökat intresse för mjukvaruutveckling samtidigt som kursen gett erfarenhet i att genomföra ett projekt i en mindre grupp med alla de fördelar och nackdelar som det innebär. Git och användandet av versionshantering har varit effektivt i och med att det möjliggjort för projektgruppen att allokera resurser och kompetenser utan att för den skull inskränka på projektets resultat. Det har dessutom möjliggjort en större flexibilitet med avseende på tid och rum. Appen och projektets övriga delar har kunnat utvecklas parallellt tack vare versionshanteringen.

Angående byggproceduren så kompileras koden automatiskt i eclipse

Skriv i reflektionerna att den låga komplexiteten (konceptet för appen) har gjort att det var enklare att ta till sig andra saker i kursen.

Anteckningar:

Byggprocedur: hur kompilerar vi koden? Eclipse gör det automatiskt. eclipse, android development toolkit, vad mer?

Komma på tester som kan härledas till krav.