

Project Reflections

Difference between individual programming and working in team

För vår grupp, som bestod av tre studenter från Industriell Ekonomi och en student från Automation och Mekatronik, var det en omställning att programmera med en grupp på fyra stycken. I tidigare kurser har programmeringsuppgifter lösts genom att man programmerat antingen helt individuellt eller i grupper om två studenter. En av skillnaden mellan att programmera i grupp jämfört med individuellt är att mer tid måste läggas på att strukturera upp och fördela arbetet, så att inte samma saker görs av olika personer samt att alla uppgifter blir gjorda. En viktig del blir därmed att tidigt utse vilka roller de olika personerna ska ha och att även utse en projektledare som har en god blick över hela projektet. Projektledaren kan då fördela ut delar av arbetet på andra i gruppen och själv ha ansvar för att alla delar görs och att de görs i tid. Redan innan vi startade med projektet så bestämde vi vilken roll var och en i gruppen skulle ha. Dessa roller hölls inte till hundra procent genom projektets gång utan vi gjorde saker som överlappade och gick in på en annan persons arbetsuppgift. Det viktigaste var dock att den roll som en person fick hade ansvar för att det som skulle göras inom det området gjordes.

Ett problem som kan uppstå när man programmerar i en grupp är att studenterna har olika bakgrund och därmed varierande förkunskaper. Detta kan innebära att viss frustration från de studenter med bättre förkunskaper samtidigt som de studenter med sämre förkunskaper känner sig förbisedda och att de inte bidrar med så mycket. Detta var dock inget som blev ett problem i vår grupp. Vi hade lite skilda bakgrunder och förkunskaper men då detta projekt innehöll flera olika delmoment kunde arbetet fördelas ut på ett lämpligt sätt. Att vi hade olika bakgrunder var istället något vi kunde dra nytta av. Vi fick en ökad dynamik i gruppen där vi inte blev inlåsta i samma tankesätt utan kunde lösa problem som uppstod genom att personer med olika sorters input kunde hjälpa till.

Kodmässigt är det också en viss skillnad mellan att arbeta individuellt och i grupp. Framförallt var det nytt för oss alla att använda github. Detta skapade vissa problem och viss frustration i början men efterhand som vi använde det mer och mer så blev det också enklare att använda och blev till slut ett viktigt verktyg för oss. Viktigt är också att kommentera mycket utav koden, vilket är något som är ännu viktigare när man arbetar flera stycken tillsammans. Om det mesta av koden är kommenterad så blir det enklare för en annan person att sätta sig in i vad det stycket av kod gör för något.

Methods used to create the document

Dokumentet skapades inledningsvis i google:s webb-baserade ordbehandlingsprogram googledocs för att det skulle vara möjligt för alla i gruppen att se och redigera i texten. Detta verktyg var en stor fördel för oss och sparade mycket tid. När texterna var helt klara skapade vi en pdf-fil av dokumentet och lade sedan upp dem på github.

Fördelarna med googledocs är att det är mycket enkelt att använda och att det är hela tiden finns tillgängligt oavsett vilken dator man än använder. Dessutom finns det ingen risk att dokument går förlorade då dokumentet sparas automatiskt hela tiden. Googledocs har även en bra versionshantering där man kan gå tillbaka i historiken och se vad man skrivit innan. Detta innebär att man på ett smidigt sätt kan gå tillbaka i historiken utifall man råkar radera ett stycke eller gjort ändringar som de andra i gruppen inte samtycker med.

Även dropbox har använts i viss utsträckning för att dela med sig av dokument och annat material. Detta har främst använts för att dela dokument som skapat en ökad förståelse kring bland annat github och android programmering. Dropbox har således enbart använts för att dela och läsa dokument och inte för att redigera dokument.

Testfallen skrevs till en början i Wiki på github, men då detta visade sig vara ett väldigt osmidigt sätt att dokumentera testfallen på övergav vi detta och började istället använda googledocs och omvandlade sedan dokumentet till pdf-fil.

Appropriateness of the adopted SE techniques

I detta projekt har en blandning av olika "software engineering" - tekniker använts. De två teknikerna vattenfallsmetoden och scrum har använts i viss utsträckning men inte fullt ut då vi inte ansett att dessa passat perfekt för vårt projekt.

Scrum är en utvecklingsteknik som kan användas vid projektbaserad programmering och innebär att man följer vissa riktlinjer. I ett Scrum-projekt finns följande huvudroller:

- En "Scrum Master" som ser till att processen följs, tar bort hinder och ser till att projektgruppen skyddas från avbrott.
- En "Product Owner" som representerar intressenten av slutprodukten
- Ett "Development Team" som är ett tvärfunktionellt team som gör själva kodningen; testningen, designen och implementationen.

Vi har inte strikt följt de riktlinjer som Scrum föreslår utan har använt "Scrum but" som kan ses som en variant på Scrum. Egentligen handlar detta om att man använder Scrum till en viss del men inte fullt ut. En utav de riktlinjer som Scrum fastställt är att man varje dag ska ha ett dagligt möte "daily scrum" där gruppmedlemmarna berättar vad de har gjort sedan föregående dag, vad de

ska göra denna dag och om det finns några problem eller hinder. Detta har vi inte haft då vi har haft fler kurser utöver denna kurs och dessutom är detta något som vi inte känt att vi behövt med tanke på projektets storlek. Vid större projekt kan dagliga möten däremot vara något som bidrar med värde.

Vattenfallsmetoden innebär att projektets utförs i olika steg där varje fas ska vara helt klart innan nästa påbörjas. Faserna eller stegen är; förstudie, krav, design, implementation, test, integration/leverans och drift/underhåll.

Vi har använt vattenfallsmetoden i viss utsträckning i vårt projekt. De olika faserna har gjorts mer eller mindre i den ordningen som metoden föreslår men vi har inte gjort en fas och sedan avslutat den, påbörjat nästa och inte gått tillbaka till föregående fas. Istället har processen varit mer iterativ, en fas har utvecklats och sedan har ändringar gjorts inom denna fas trots att nästa fas har påbörjats.

General comments about the project as a whole

Projektet har inneburit ett annorlunda sätt att jobba på och har skilt sig mycket från när man arbetar individuellt. Detta har varit mycket lärorikt och spännande. Dessa fördelar tillsammans med att projektet varit väldigt roligt att utveckla har övervägt de nackdelar som det innebär att arbeta i grupp. Projektet har varit extra intressant eftersom man utöver det man lärt sig i kursen och arbetet i grupp också fått fram en produkt/app som man kan lägga upp på android market och sedan visa upp för sina kompisar. Detta ger ett mervärde till projektet och leder till att man lägger ner lite mer tid än man annars kanske skulle ha gjort.

Att använda sig utav ett repository på github innebar till en början vissa problem men efter ett litet tag insåg vi vilka stora fördelar detta verktyg hade. Arbetet har varit mycket smidigare att genomföra med hjälp av github.

Att arbetet har varit så fritt som det varit har mestadels varit positivt då det inneburit att vi kunnat planera och lägga upp arbetet som vi själva har velat. Nackdelen med detta upplägg är dock att det blir svårt att veta hur lång tid de olika delmomenten i projektet kommer att ta. Eftersom alla i gruppen har liten erfarenhet från att arbeta med ett IT-projekt i grupp, så har det varit svårt att veta hur lång tid man ska lägga på varje del och hur lång tid varje del "normalt sätt" ska ta. Detta har delvis lösts genom diskussion med vår handledare vid handledningstillfällena men har ändå varit lite problematiskt.