

Post Mortem Report

Anders Nordin(anordin@student.chalmers.se)

Maj 31, 2013

1 Bakgrund

Projektgruppen bestod utav fem personer.

Fredrik Einarsson Datateknik

Anders Nordin Datateknik

René Niendorf Datateknik

Sofie Peters Industriell Ekonomi

Niklas Johansson Industriell Ekonomi

Vår ursprungliga vision var att skapa "The Complete Chalmers Experience" för studenter och besökande. Detta gjordes genom att skapa en Android-applikation med **Google Maps v2 API** i grunden. Vi byggde sedan på med funktioner som vi tyckte var nyttiga så som navigering, sök och en slags kalorimätare för pubkvällar m.m. Till vår app skapade vi en begränsad databas med information över olika platser på campus Johanneberg: bankomat, grupprum, föreläsningssalar och mycket mer.

2 Utvecklingsmetoder och verktyg

Eftersom vi var fem personer i ett relativt litet projekt var vi tidigt överens om att dela in arbetet för att inte trampa varandra på tårna. Då gruppen bestod av medlemmar med varierande bakgrund så var det även viktigt att tilldela medlemmarna uppgifter som de kände sig bekväma med. Som många andra projekt är det viktigt att ha en god kommunikation med varandra, till vår hjälp använde vi flera verktyg för att kunna ha det, utöver fysiska möten.

2.1 Arbetsmetoder

I vårt projekt har vi följt en agil utvecklingstrategi genom att blanda delar från *SCRUM* och *eXtreme Programming*. Mer exakt om hur vi gått tillväga kommer i nedanstående avsnitt.

2.1.1 SCRUM

- 7 dagars sprint
- 3 *SCRUM*-möten per vecka
- Måndagar extra viktiga med *sprint review*, *sprint retrospective* och *sprint planning*
- Ingen beställare(*product owner*).
- Sprint Backlog
- Ingen *SCRUM Master*

Under våra korta *SCRUM*-möten fick medlemmarna besvara tre frågor:

1. Vad har jag gjort sedan senaste mötet?
2. Vad ska jag göra tills nästa möte?
3. Finns det något som hindrar mig?

I början agerade jag *SCRUM Master* men efter projektet gång tonades den rollen ned eller snarare behövdes inte. Alla var väldigt drivna och målmedvetna vilket gjorde att det här projektet löpte på ovanligt bra. Jag tror dock inte det skulle fungera med en annan grupp vid ett annat tillfälle, det är verkligen något ovanligt. För att komma tillbaka till berättelsen om grisen och hönan som skulle öppna restaurang så kan det ha att göra med att ingen av oss medlemmar utgjorde grisen i historien.

2.1.2 eXtreme Programming

- *User stories*
- Acceptance tests för att bedömma om en user story implementerats rätt
- Kollektivt ägande av koden

- Nästan alltid har en fungerande version funnits tillgänglig
- Vi har regelbundet refaktorert koden
- Strävat efter flera releaser, vi släppte en pre-release och en slutgiltig.
- Arbetade utifrån en produktvision, "The Complete Chalmers Experience"

I våra user stories använde vi oss av ett poängsystem: 1p, 2p, 3p, 5p och 10p. Där 1p kunde vara att lägga till bilder medan 10p kunde vara hela navigeringssystemet. Poängen motsvarar inte riktigt antalet timmar utan snarare hur omfattande *User stories* var. Allt eftersom projektet fortgick så valde vi vilka *user stories* vi skulle implementera. Men hade ständigt i åtanke vår produktvision.

2.1.3 Arbetet i gruppen

Jag la ungefär 20 timmar i genomsnitt på kursen, bortsett från de två första veckorna. I detta inkluderar jag: veckans kursmaterial, läsa på om funktionen jag ska implementera och slutligen projekttid. Största delen av arbetet utförde jag i koden där jag bl.a. skapade navigeringen för vår karta och refaktorert koden.

Jag upplevde att vi hade en jämn fördelning av arbetet i gruppen. Vi hade tydliga ansvarsområden där Fredrik mestadels jobbade med databasen, Sofie med dokumentationen, Niklas med GUI och Rene med själva datan(platsinformation) vi skulle använda i appen. Det ska dock nämnas att man ibland även gjorde delar utanför sitt huvudområde.

Som jag tidigare nämnde under 2.1 så fungerade vår grupp väldigt bra. En bidragande orsak till det tror jag kan ha varit att vi har väldigt olika bakgrund, med två från industriell ekonomi och tre med datainriktning. Det gjorde att vi verkligen har kunnat fokusera på det man är bra.

2.1.4 För- och nackdelar

Jag har tidigare i mitt kandidatarbetat arbetat enligt hur denna kurs tidigare såg ut. Vi följde en agil metod men framförallt så hade vi mycket mer dokumentation. Vi hade dokument för bl.a. users cases, software design document, Requirements and design document och m.m. Jag tycker det är väldigt positivt att vi inte har arbetat med detta eftersom det tar mycket tid och energi som jag tycker borde läggas på att skriva bra kod. Här tycker jag våra user stories har utgjort väl spenderad tid.

Att alla hade ett gemensamt ansvar för koden hade både för och nackdelar, dels kunde vi ha en fungerande version uppe nästan hela tiden då det alltid var något som tog ansvar för eventuella fel. Den negativa sidan var att medlemmarna ofta la sin kod på fel ställe eller skapade en metod som redan fanns, det eftersom man inte hade tillräcklig kunskap av koden som helhet. Vidare så gjorde det att kodens struktur fick lida.

En klar nackdel med tillvägagångssättet vi använde var att vi inte hade något beställare. Det gjorde att vi kunde välja och vraka lite bland funktionaliteten. Tyvärr är det inte så när man väl har en kund, då krävs en väl motiverad förklaring till varför man inte kunde skapa en viss funktion.

Jag gav tidigare en del beröm till vår grupp och fördelarna med att vi kom från olika program. Det negativa med det har varit att vi ibland haft svårt att arbeta tillsammans, det är många scheman och kurser som ska stämma. När man arbetar i mjukvaruprojekt är det mycket bättre att kunna sitta tillsammans på samma kontor så man snabbt kan ställa eller diskutera en fråga.

2.2 Kommunikation

Utöver de fysiska möten som var en del av *SCRUM*, beskrivet i 2.1, så har vi använt flera andra verktyg. En Facebook-grupp skapades för att kunna användas som ett forum. För ärenden som gällde själva koden använde vi **GitHubs Issues**. Tack vare att vi hade delat in vårt projekt i tydliga delar så har det varit förhållandevis enkelt för medlemmarna att hitta rätt person att vända sig till.

2.3 Verktyg

För att utföra vårt projekt hade vi flera verktyg till vår hjälp, vissa var obligatoriska i kursen och andra inte.

Eftersom vi utvecklade en Android-applikation så var det ganska givet att använda **Eclipse** med **Android Developer Tools**. Istället för emulatorn använde vi våra egna Android-telefoner för att testköra applikationen.

GitHub användes som versionhanteringsprogram. Eftersom de flesta av oss var ganska ovana med verktyget till en början så har det tagit oss en del tid hantera det.

Pivotal Tracker gick dock snabbare att komma igång med och har varit vårt projektledningsverktyg. Här har vi hanterat våra **User Stories** och genomfört våra **SCRUM**-sprints. Verktyget har även använts flitigt för att tilldela uppgifter till medlemmarna.

3 I ett framtida projekt

I ett liknande mjukvaruprojekt skulle jag fortfarande använda mig av de verktyg vi har använt; eclipse, gitHub, pivotal tracker. Det jag dock skulle tycka var intressant är att verkligen testa *SCRUM* hela vägen. Vad gäller pivotal tracker så skulle jag, nu när jag kan det bättre, verkligen lägga ytterligare vikt vid att få till bra *user stories*.

3.1 Arbetsmetoder

Jag förstår varför agil utveckling används på så pass många företag runt om i världen. T.ex. *SCRUM* är enkelt att förstå men har ändå tydliga riktlinjer på vad som gäller. Det gör att man som utvecklare lätt kan arbeta i olika projekt.

Jag tror *SCRUM* framförallt lämpar sig i projekt med en beställare. Beställaren får god insikt i projektet och vidare så kan utvecklarna verkligen säkerhetsställa att den efterfrågade funktionen blev utformade så som kunden ville.

Jag tror däremot att s k in-house projekt, där företaget själva är beställare, inte passar för en agil utvecklingsmetod. Man har större makt och kontroll över projektet då och behöver inte kunna anpassa sig hela tiden.

4 Återkoppling på kursen

Jag tycker kursen har lärt ut ett effektivt sätt att hantera projekt på. Många mjukvaruprojekt så ägnar man mer tid åt dokumentation än att faktiskt skapa produkten vilket jag alltid tyckt varit konstigt.

Till nästa år bör betygskritierna vara klara i tid tillsammans med vad som verkligen förväntas i kursen. Jag har självklart förståelse för att detta var första gången kursen hölls av dig.

Vår kommunikation med handledare har fungerat bra. Vi fick bra feedback och hjälp om vi behövt, ska kanske tilläggas att vi hade Max Witt som handledare.

Våran grupp var ganska mixad, folk med tre olika bakgrunder. Till nästa år tror jag att man ska försöka få gruppen mixad, det har fungerat väldigt bra för oss. Dels får man arbeta med människor från andra program samtidigt som att det är mer likt ett verkligt projekt.