

Norges Teknisk-Naturvitenskapelige Universitet

IT1901

Individuell Rapport
00 – Arbeidsvaner

507632

Gruppe 02

Antall ord: 1409

Individuell rapport

Når man skal innlede et teamarbeid og et prosjekt er to viktige aspekter å være enige om arbeidsvaner og hva man skal lage. Ved å ha en enighet om disse faktorene, kan være med på å spare mye frustrasjon og misforståelser senere. Heldigvis er det utviklet flere standarder for hvordan man burde strukturere prosjekt- og teamarbeidet. Hvis alle som jobber på et prosjekt har en viss kjennskap til en eller flere arbeidsmetoder er det lettere å finne en som passer for gruppen og som alle kan føle seg komfortable med å bruke.

I vår gruppe definerte vi aldri hva slags arbeidsmetode vi skulle bruke, men vi hadde alle Scrum i bakhodet. Vi lagde en teamkontrakt der vi ble enige om faste møter, straff for å komme sent og arbeidsmengde på 5-12 timer i uka for hver enkelt. Vi bestemte oss for å ikke fordele noen roller eller ansvarsområder, selv om dette er en vanlig ting å gjøre i Scrum. (Schwaber, 2020, Scrum Definition)

Scrum er en av disse standardene man kan bruke som et rammeverk. (Schwaber, 2020, Scrum Definition) Den tar for seg alt fra start til slutt og er med på å strukturere møter, gruppen, oppgaver og hvor ofte man snakker sammen. (Schwaber, 2020, Scrum Definition) Hvis man følger Scrum Guiden så starter man et prosjekt ved å definere en produkteier, en Scrum master, og sprinter. (Schwaber, 2020, Scrum Definition) Som tidligere nevnt, valgte vi aldri roller og lot være å følge Scrums retningslinjer. (Schwaber, 2020, Scrum Definition) Vi hadde en viss forståelse for at hver innlevering ble som en sprint, men vi snakket heller aldri om dette. (Schwaber, 2020, Scrum Definition)

Noen av problemene opplevde er treig fremgang, ustrukturerte møter og dårlig bruk av utviklingsoppgaver på GitLab. Ved å benytte oss av Scrum mer aktivt og definere en Scrum master, kunne disse problemene vært løst. (Schwaber, 2020, Scrum Definition) Dette er ettersom at Scrum masteren kunne vært en fast møteleder og passet på at alle kom på møtene. (Schwaber, 2020, Scrum Definition) Scrum inkluderer også ofte en daglig innsjekk med hva du har gjort, hva du skal og hvilke utfordringer har oppstått. (Schwaber, 2020, Scrum Definition) Med dette som utgangspunkt, ville det kunne strukturert møtene våre. Det ville da også vært naturlig å bruke utviklingsoppgavene som utgangspunkt i hva som er gjort/skal/utfordringer, som er med på å gi struktur. Med tanke på treig fremgang kan man anta at utviklingsoppgavene våre var for store eller utydelige, og at dette også handler om ustrukturerte møter. (Schwaber, 2020, Scrum Definition) Det hele koker ned til at med en Scrum master ville noen hatt direkte ansvar for dette, og problemene ville vært unngått.

(Schwaber, 2020, Scrum Definition) Altså, det å definere utviklingsoppgavene bedre og gi oss strukturen vi sårt manglet. Basert på denne erfaringen ville jeg absolutt antatt at ved å ha en tydeligere struktur og brukt Scrum mer aktivt ville gitt oss et bedre sluttprodukt. (Schwaber, 2020, Scrum Definition)

Dårlig bruk av grener på GitLab og dårlig bruk av Git generelt har også vært et problem for oss. Dette kommer igjen ut fra samme problemet med at utviklingsoppgavene kanskje ble for store og ikke så tydelige som de burde vært, og dermed ble grenene større og varte lenger enn ønsket. (Chacon, 2014, Pro Git) I Scrum så ønsker man å produsere et produkt i løpet av en kort periode og dermed få selve produktet kjapt ut til kunde. (Schwaber, 2020, Scrum Definition) Ved å ha selve utviklingen liggende ute i en gren, eller lokalt på maskinen sin vil dermed være uproduktivt, ettersom at master er der alt skal ligge slik at kunden får brukt det. (Schwaber, 2020, Scrum Definition) For dette problemet kan vi se på et prinsipp kalt trunk based development, eller stammebasert utvikling, for forgrening og bruk av master.

(Hammant, 2020, Introduction) Dette prinsippet bygger på at man har en master, eller stamme, der utviklingen skjer og er hovedgrenen for commits og utvikling. (Hammant, 2020, Introduction) Deretter kan man ha mindre grener ut fra denne stammen som kun varer i noen få dager og dermed aldri har store mangler fra master. (Hammant, 2020, Introduction) Ved å jobbe på denne måten vil man kunne vite at prosjektet er samla og at når man jobber så har man mest sannsynlig siste versjon, og eventuelt mangler kun småting. (Chacon, 2014, Pro Git) Ved å følge stammebasert vil man også ha flere sammenslåinger som vil legge opp til at man oftere går gjennom hverandres kode. (Hammant, 2020, Introduction)

Det andre problemet vi hadde med Git var lite commits. (Chacon, 2014, Pro Git) Det skjedde flere ganger at en gren hadde kun en til to commits i løpet av flere uker, der den første gjerne inneholdt lite og den andre fullførte resten av utviklingsoppgaven. (Chacon, 2014, Pro Git) Det vil da si at alt av utvikling som skjedde, vil ikke eksistere i repoet. I tillegg vil all jobbing på master, eller nye grener, ikke henge sammen og kan gi sammenslåingskonflikter. (Chacon, 2014, Pro Git)

En av fordelene ved å bruke Git, er at man kan etter hvert som man gjør noe, commite det, men også kunne gå tilbake til tidligere versjoner. (Chacon, 2014, Pro Git) Dette er kjekt hvis ting fungerte bedre før siste commits, for eksempel. (Chacon, 2014, Pro Git) I tillegg vil det også være lettere å sette seg inn, hjelpe til, parprogrammere og samarbeide hvis man commiter ofte med gode commitmeldinger. (Chacon, 2014, Pro Git) Dette er et problem som kunne vært fikset med bedre forståelse av hvordan Git skal brukes, og om vi hadde snakket

om prinsipper som stammebasert utvikling for å hindre at dette ble et problem. (Chacon, 2014, Pro Git)

Alle problemene vi fikk kan skyldes mangel på kommunikasjon og at vi endte opp med skippertak. Dette manifesterte seg ved dårligere løsninger, så kalte 'quick fixes' og stress, som også ga dårlige commit-meldinger slik at det var mye å nøste opp i til neste innlevering. (Chacon, 2014, Pro Git) Dette er kanskje de tingene som påvirket koden vår mest negativt. I perioder slet jeg med at jeg ikke fikk svar på spørsmål jeg stilte, og møter ble glemt eller avlyst hyppig. Dette førte igjen til at det gikk bort mye tid til å lete i koden etter svar på hva de andre hadde gjort og ikke gjort, før en selv kunne begynne å jobbe.

Noen av de mer positive sidene med at vi aldri bestemte oss for en fast måte å jobbe på er at en kunne jobbe når det passet for en selv. Vi har i gruppen også vært preget av mye sykdom i løpet av perioden, i tillegg til at halvparten av gruppen har hatt mye obligatorisk i form av praksis og lignede. Ved å ikke ha for faste rammer har det vært lettere å ta pauser når man trenger, samt at det har vært mer naturlig at vi har hatt forskjellige ansvarsområder.

For å sikre suksess på dette området vil jeg i fremtiden følge Scrum mer. Det å definere rollene tidlig, ha faste strukturerte standup møter og det å følge stammebasert utvikling vil jeg absolutt se på som nyttige for å unngå flere av problemene vi har hatt under dette prosjektet. Jeg vil også i fremtiden være mer nøye på å definere hvilke utviklingsmetoder, av fossefall, inkrementell og evolusjonær, vi skal bruke. (Stocia, 2021) Jeg innså ikke at dette var en mismatch før sent, og jeg tror vi kunne unngått noe stress og misoppfatninger. Det viktigste jeg tar med meg fra denne erfaringen er å starte med en lenger forventningsavklaring, der vi snakker om flere aspekter rundt programmeringen, hvordan vi forventer å bruke hjelpemidler og hva som er realistisk for oss å oppnå i løpet av prosjektperioden. Jeg ønsker også at alle er på samme minimums kunnskapsnivå innenfor Git, Scrum og smidige utviklingsmetoder for å ta i bruk disse på en god måte – og dermed unngå misforståelser, ha en god kommunikasjon og få bedre resultat. (Chacon, 2014, Pro Git) (Hammant, 2020, Introduction) (Schwaber, 2020, Scrum Definition)

Kilder:

Chacon, S. & Straub, B. (2014). *Git Basic and Getting Started*. Pro Git (2. Edith). Apress.

Dybå, T., Dingsøy, T., & Moe, N. B. (2002). *Praktisk prosessforbedring - en håndbok for IT-bedrifter*. Fagbokforlaget.

Hammant, P. (2017-2020). *Introduciton*. Trunk Based Development:
<https://trunkbaseddevelopment.com/> (2020)

Schwaber, K. & J. Sutherland, 2020. *Scrum Definition*. Scrum Guides:
<https://scrumguides.org/scrum-guide.html>

Stocia, G.A. (2021). *Utviklingsmetode: Scrum* [Lysarkpresentasjon]. Blackboard.
<https://ntnu.blackboard.com/>