



**NTNU – Trondheim**  
Norwegian University of  
Science and Technology

IT1901 – INFORMATIKK PROSJEKTARBEID 1

INSTITUTT FOR DATATEKNOLOGI OG INFORMATIKK

---

## Rapport gruppe 19

---

*Gruppedlemmer:*

Maciej Przemyslaw Ambrozek  
Magnus Gaustad Eriksson  
Andreas Skifjeld  
John Gullaksen  
Daniel Stupar  
Heidi Brække  
Peder Lexau

*Studentnummer:*

718721  
259503  
476734  
475274  
476700  
476725  
741803

xxxx ord

11. oktober 2017

# Innhold

<b>Introduksjon</b>	<b>1</b>
Faget . . . . .	1
Prosjektet . . . . .	1
Oppgaven . . . . .	1
Gruppa . . . . .	2
<b>Prosjektorganisering</b>	<b>3</b>
Scrum . . . . .	3
Rolle- og Ansvarsfordeling . . . . .	3
Sprinter . . . . .	3
Sprintplanlegging . . . . .	4
Daily Scrum . . . . .	4
Sprint review . . . . .	5
Sprint retrospektiv . . . . .	5
Diskusjoner og avgjørelser . . . . .	5
Sprint 1 . . . . .	5
Verktøy . . . . .	6
<b>Sprinter</b>	<b>7</b>
Sprintlengde . . . . .	7
Organisering . . . . .	7
<b>Referanser</b>	<b>8</b>

# Introduksjon

## Faget

Hentet fra fagets emneside<sup>[1]</sup>:

I dette kurset skal en gruppe på 5 - 9 studenter gjennomføre et mellomstort programmeringsprosjekt. Hovedmålet er å gi studentene forståelse for samspillet mellom prosess og produkt-orienterte utfordringer og aktiviteter i et programmeringsprosjekt. Viktige aspekter er kunnskap om metoder og teorier for organisering av gruppe-programmeringsprosjekter. Studenter må arbeide med og reflektere over integrasjonen av ulike komponenter for å sette sammen et større programvareprodukt.

## Prosjektet

Programmeringsprosjektet til faget er lagt opp til å være mellomstort i forhold til liknende kommersielle prosjekter, men legger vekt på utviklingsprosess og samspill i gruppa. Gjennom prosjektet skal studentene få innsikt i hvordan et ekte utviklingsprosjekt for en kunde fungerer. Selv om det er prosessen og ikke produktet som vektlegges, skal det gjennomføres som om det er for en ekte kunde når det gjelder gjennomførelse og testing. I løpet av prosjektet må gruppa samarbeide, lære seg nye ferdigheter, og til slutt ha et fungerende system som kunden er tilfreds med. Rollen som kunde gjøres av stud.ass. siden studentfestivalen ikke eksisterer.

## Oppgaven

Oppgaven i faget gikk ut på å lage et program for en studentfestival som fiktiv kunde. Programmet skal være inndelt slik at personer med forskjellige roller har forskjellige egenskaper etter hvilke jobber og hva slags ansvar de har. For eksempel skal en arrangør kunne se hvilke konserter som er på hvilke scener til hvilke tider, og hvilke teknikere som hjelper til. Bookingansvarlig skal kunne lage tilbud til band om sted, tid og pris, og sende dette videre til bookingsjef for godkjennelse. Om godkjent kan tilbudet bli sendt til en person med rollen manager. Denne informasjonen ble gitt som forskjellige brukerhistorier, og kan leses i Vedlegg Brukerhistorier.

## Gruppa

Gruppa består av sju studenter som kan omtrent like mye innenfor programmering, med noe ekstra kompetanse på andre områder som bidrar med oppgaven. Seks av gruppens medlemmer kunne Python og Java, mens Peder hadde ferdigheter innen Matlab og C++. Fra før av kunne også Andreas og Magnus en del om Git, John hadde erfaring med MySQL og databaser, og Peder kunne skrive i L<sup>A</sup>T<sub>E</sub>X. Dette gjorde det også lettere for gruppa å kollektivt lære mer om programmene som ble brukt i oppgaven. Sammen har vi gode forutsetninger for å løse oppgaven, og vi har høye ambisjoner.

# Prosjektorganisering

## Scrum

Som en del av prosjektet var det gitt at vi skulle bruke rammeverket Scrum til arbeidsprosessen. Scrum er definert som "A framework within which people can address complex adaptive problems, while productively and creatively delivering products of the highest possible value"[2].

Rammeverket består av et eller flere Scrum lag/grupper og dens roller, begivenheter, artefakter og regler som binder det hele sammen. Det er et rammeverk som er iterativt og inkrementell for å håndtere kunder som ofte endrer meninger om hva de ønsker seg og trenger. I dette kapitlet kommer vi innpå hvordan vi har tilpasset rammeverket til vårt prosjekt og utviklingen av prosessen gjennom de forskjellige sprintene.

Vi hadde en periode før sprintene som skulle brukes til å lese pensum og gjøre unna praktiske ting som å velge hvilke verktøy vi skulle bruke og programmeringsspråk vi skulle skrive i. I tillegg brukte vi denne perioden til å lage en produkt backlog og en releaseplan.

## Rolle- og Ansvarsfordeling

### Scrum team:

Scrum master: Magnus Eriksson

Testansvarlig: Daniel Stupar

GUI ansvarlig: Andreas Skifjeld

Databaseansvarlig: John Gullaksen

Rapportansvarlig: Peder Lexau

Backend-ansvarlig: Heidi Brække og Maciej Ambrozek

## Sprinter

Sentralt i Scrum er sprintene/iterasjonene. Etter hver sprint skal man ha en ferdig bit av det endelige programmet som er klart til å utgis. En sprint varer normalt mellom 1-4 uker og inneholder et planleggingsmøte, "daily scrums", retrospektivmøte og et "sprint review" møte. Etersom dette er et deltidsprosjekt, har vi måttet gjort noen tilpasninger, men vi har prøvd så godt det lar seg gjøre å ha med alle disse i prosessen vår.

Vi bestemte oss tidlig for 2 ukers sprinter. En sprint startet på fredag med planleggingsmøte og gikk til onsdag halvannen uke etterpå hvor vi hadde retrospektiv. De ukene hvor det ikke var demo, hadde vi en enkel "sprint review" på torsdager da vi hadde tilgang til kunden denne dagen.

Det ble satt opp fredag 1215-1600 og onsdag 1015-1200 som arbeidsdager i tillegg til torsdager hvor vi enten hadde kunde/stud.ass.møte eller demonstrasjon. I delkapitlene under presenterer vi hvordan vi gjennomførte de forskjellige møtene samt et avsnitt om hver av sprintene vi gjennomførte.

## **Sprintplanlegging**

Vi valgte å følge Knibergs[3] retningslinje om én time planlegging per uke sprint slik at vi hadde to timer til rådighet. Hvis vi ikke var 100 prosent ferdig etter to timer avsluttet vi uansett for å holde tidsboksen vi hadde satt. Agendaen for dette møtet var:

- Sette et mål for sprinten
- Opprette en sprint backlog ut i fra brukerhistoriene i produkt backlog
- Tid og sted for demo til kunden
- Sette en rekkefølge på oppgavene
- Arbeidsfordeling
- Velge vaktmester (se rollebeskrivelse og avsnitt om sprint 1)

## **Daily Scrum**

Ettersom prosjektet er et deltidsprosjekt, bestemte vi oss for å kun ha "daily scrum" de gangene vi møttes. Dette gjennomførte vi stående og ordet gikk på runde hvor alle skulle si:

- hva de hadde gjort siden sist
- om det har vært noen utfordringer
- hva de tenker å gjøre framover
- om det er noe som står i veien for at de klarer å gjennomføre det

Vi hadde god nytte av dette ved at man fikk en rask oppdatering på hvordan folk lå an, at de faktisk hadde gjort noe og de hadde noe å gjøre i den kommende arbeidsøkten.

## **Sprint review**

Denne hendelsen er den vi har hatt størst avvik med opp mot rammeverket. Det har bakgrunn i at det er en fiktiv kunde og at vi hadde begrenset tilgang til han. På torsdagsmøtene vi hadde med kunden etter en sprint gjorde vi følgende:

- Viste fram den nye biten av programmet som var ferdig og fikk inputs fra han.
- Spurte om kunden hadde en ny prioritering av produkt backloggen og diskuterte tenkt fremgang for neste sprint.

## **Sprint retrospektiv**

Retrospektivmøtene startet med innsjekk før scrum master gikk gjennom tidslinjen for sprinten med fokus på de viktigste hendelsene slik at alle var oppdatert på hva som hadde skjedd. Deretter gikk ordet på rundgang hvor vi gikk gjennom hva folk mente gikk bra, hva som kunne gjøres bedre og konkrete forbedringspunkter vi kunne ta med oss til neste sprint. Til slutt stemte vi over hva slags forbedringspunkter vi skulle ha mest fokus på. Vi fikk mye ut av disse møtene som spesielt i starten resulterte i endringer på hvordan vi jobbet sammen. Dette er utdypet mer om i gjennomgang av sprintene i de neste delkapitlene.

## **Diskusjoner og avgjørelser**

### **Sprint 1**

Den første perioden var preget av at gruppen ønsket å komme tidlig og godt i gang, samt at vi hadde lest oss opp for lite på scrum. Vi hoppet fort inn i programmeringen, og hadde lite fokus på prosessen. Dette viste seg i releaseplanen og sprint backloggen, og resulterte i en dårlig planlagt sprint. Etter møte med studentassistent ble vi enige om å ta et skritt tilbake å gjennomføre et ordentlig planleggingsmøte, hvor vi faktisk brukte rammeverket. Vi fikk med en gang et bedre utgangspunkt for videre arbeid, og ble ferdig med sprinten i tide.

Under retrospektivmøtet fikk vi diskutert forskjellene før og etter restart av sprinten. Alle var enige om at rammeverket funket så lenge vi brukte det. Vi hadde fortsatt en del å gå på angående kommunikasjon og dokumentasjon, og det ble dermed opprettet en rullerende vaktmesterrolle som skulle passe på at dette ble gjort.

## Verktøy

I arbeidet vårt har vi kun brukt gratis verktøy. Vi har valgt å bruke Java som programmeringsspråk, og skrive koden i IntelliJ. JavaFX var plattformen vi brukte til GUI, og den ble skrevet ved å bruke FXML og CSS. Koden var integrert i Git. For å ha oversikt over oppgaver og fremgang av arbeidet med dem, brukte vi Trello. I vår interne kommunikasjon brukte vi Facebook. Slike dokumenter møtereferater, diagrammer, eller beskrivelse av user stories o.l. gjorde vi i Google Drive. Databaser ble laget i phpMyAdmin og vi brukte SQL server fra NTNU. Selve rapporten ble skrevet ved å bruke L<sup>A</sup>T<sub>E</sub>X.



# Sprinter

## Sprintlengde

Sprintlengden kan variere i lengde, og den kan bli alt fra 1 til 4 uker, hvorav Kniberg[3] anbefaler korte sprinter. Vi har valgt å sette av 2 uker på en sprint og prøvde å forholde oss til det hele tiden. Denne lengden viste seg å fungere fint for oss.

## Organisering

Vi bestemte oss for å begynne sprinten på fredag og avslutte på onsdagen halvannen uke senere. I løpe av uken hadde vi to felles obligatoriske arbeidsøkter, hvor vi traff hverandre, hadde stand-up meetings for å få innblikk i hvor langt vi hadde kommet, koordinerte jobben vår, og diskuterte utfordringer og løsninger sammen. Disse møtene tok plass på onsdager fra klokken 10.15 til klokken 12.00, og fredager fra klokken 12.15 til klokken 16.00, så det utgjør seks timer per uke som gikk til felles arbeidsøkter og møtevirksomhet. Vårt første møte tok plass fredag 01.09.2017, og siste onsdag 15.11.2017. Utenom planlagte faste møter, møttes vi hvis det var behov for det, men vi jobbet også alene. Bortsett fra ovennevnte møter, hadde vi også møter med product ownersom vi brukte til å hente nye krav angående produktet og møter med stud.ass., som vi brukte til å spørre om ymse, som for eksempel tekniske krav, frister eller generelt om råd og veiledning osv. På begynnelsen av en sprint tok vi for oss organisering av arbeidet for den. Det gikk ut på å fortolke kravene gitt fra kunden til brukerhistorier, og så fra brukerhistorier til konkrete løsninger, som vi fordelte mellom oss i felles diskusjon. Disse oppgavene ble så lagret i Trello og der fulgte vi også fremgangen av arbeidet. Vi pleide også å estimere hvor mye tid vi tenkte å bruke på dem. Vi klarte ikke alltid å estimere tiden riktig i starten, men det var noe vi ble flinkere senere i prosjektet.

## Referanser

- [1] Hentet fra: <https://www.ntnu.no/studier/emner/IT1901#tab=omEmnet>  
06.10.2017
- [2] Hentet fra: <http://www.scrumguides.org/scrum-guide.html> 11.10.2017
- [3] **Scrum and XP from the trenches**, side 20. *Henrik Kniberg, 2015*