

IDATT1002 Systemutvikling

Prosjekt våren 2023

Grethe Sandstrak
Muhammad Ali Norozi
Surya Kathayat

Tema for dagen...

- Praktisk informasjon om prosjektet
 - Team og arbeidskontrakt
 - Arbeidsmengde
 - Prosjektplan
 - Dokumentasjon av prosjektet
- Prosjektverktøy
- Vurderingskriterier
- Presentasjon av oppgaven

Læringsmål

- Du skal få praktisk erfaring i:
 - Prosjektstyring, dokumentasjon og gjennomføring
 - Praktisk erfaring med å anvende programmerings i samspill med en systemutviklingsmodell for å løse virkelighetsnære problemstillinger
 - Forstå og tolke (kunde) krav
 - Utvikle kundekrav til et programvaresystem
 - Utvikle gode skrive- og framstillingsferdigheter både på norsk og engelsk
 - Jobbe sammen i team

Prosjekt - arbeidsmengde

- Teamstørrelse
 - 4-5 deltakere på hvert team
- Ca. 110 timer per deltaker
 - +/- 10% kan aksepteres
 - Arbeid som er utført skal loggføres
 - Individuell timeliste/statusrapport
 - Total oversikt for teamet
 - Hva har dere brukt tid på i prosjektet
 - Se '[Template Timesheets with status reports](#)'

Roller i prosjektet

- Student
 - Deltar i team som skal utføre et oppdrag gitt av en oppdragsgiver
 - Vurdere eget arbeide/prestasjon i sluttrapport
 - Lede prosjektmøtene – møteleder
- Læringsassistenter
 - Veilede ukentlig under prosjektperioden (sjekk på Blackboard)
 - Gruppe 1 - 5: Daniel Andre Evensen
 - Gruppe 6 - 10: Anna Marie Bøe Tvedt
 - Gruppe 11- 15: Jens Christian Lund
 - Gruppe 16 - 21: William Lie
- Produkteier/ Kunde
 - Hvert team finner sin kunde/ målgruppe på egenhånd
 - Definere krav, delta i brukertester, gi tilbakemelding underveis
- Faglærer
 - Prosjektveileder
 - Vurderer dokumentasjon, prosess og programvaresystem

Veiledningsmøter

- m/ Faglærer
 - Obligatoriske (2 ganger)
 - Annonseres med fast varighet
 - Studentdrevet
 - Møteinnkalling, møtereferat
 - Møteleder, møtereferent
 - Faglærer(e) som veileder
- m/ Læringsassistenter
 - Obligatoriske
 - Ukentlig, avtales i samråd med læringsassistent
 - Minst 2 fra teamet må være tilstede
 - Studentdrevet
 - Vis framdrift fra forrige uke + veiledning på faglige utfordringer evt. spørsmål

Project plan -

Time	Activity/ Milestones
30. January	Project assignment handout
10. February	Arbeidskontrakt /Collaboration agreement ready
3. March	Wireframe, Vision Document and domain-model submission
13. March*	Team meeting 1 – Presentation/discussion/feedback om Vision document/domain model/wireframe
weekly	Wiki- and Gitlab-structure – demonstrate for teaching assistant on weekly meeting
24. March	MVP-submission
27. March*	Team meeting 2 – Presentation of MVP. WIKI requirements review.
28. April	Final delivery report, wiki and code (inspera) Individual reflection notes (blackboard)
Week 18 (1 –5 May)	Presentation

**Dato for team-møter kan avvike – se egen kjøreplan i BB*

Dokumentasjon av prosjektet

- Arbeidsdokumentasjon (se «Project Manual»)
 - Arbeidskontrakt (Collaboration agreement)
 - Gantt-diagram
 - Timelister m/statusrapport
 - Møtereferat, møteinnkallinger
 - Issueboard (gitlab)
- WIKI - med systemkrav i henhold til gitt WIKI-struktur (se lysark på slutten)
- Sluttrapport med teorikapittel
- Individuelt refleksjonsnotat
- Oppgavebeskrivelse og flere detaljer – se Project Assignment i BB.
- Dokumentmaler finner dere også i BB

Brukertest og universell utforming

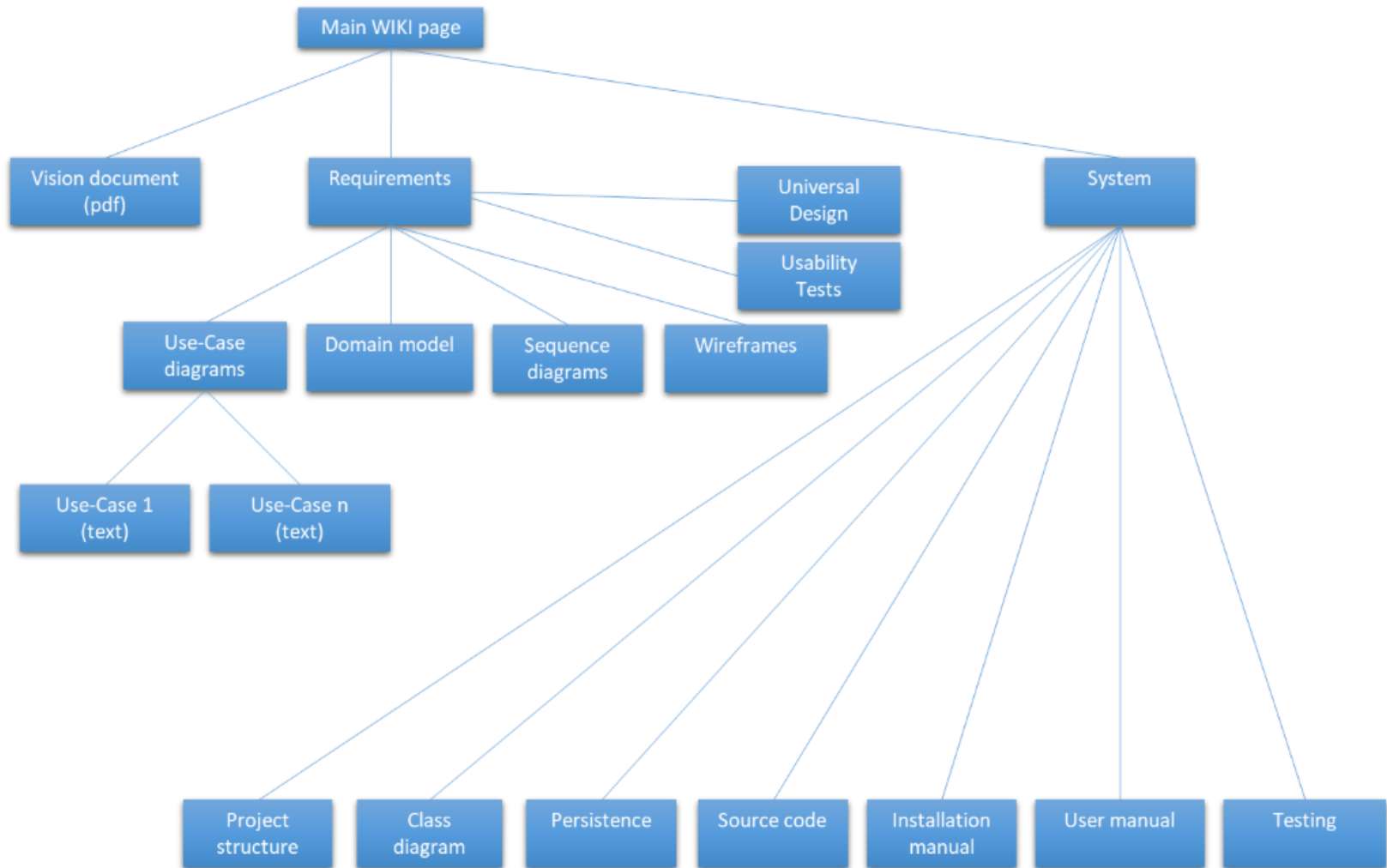
- Utfør brukertest på 1. og 2. iterasjons inkrement:
 - Wireframe
 - MVP
- Følg oppsett fra leksjonene
 - Testplan
 - Rapport om gjennomføringen
 - Konklusjoner
- Følg retningslinjene for WCAG 2.1 – prinsipp 1 Perceivable



Prosjektverktøy

- **GitLab Repo**, for kode etc
- **GitLab WIKI**, for dokumentasjon
- **GitLab Issue Board**, en tavle som viser fremdrift
- **GitLab Pages**, for JavaDoc
- **Balsamiq**, for å lage prototyper/ wireframes
- **Mattermost**, chat med kunde (faglærere)
 - Bruk også en eller annen intern chat i gruppa!

WIKI - Structure



Prosjektverktøy

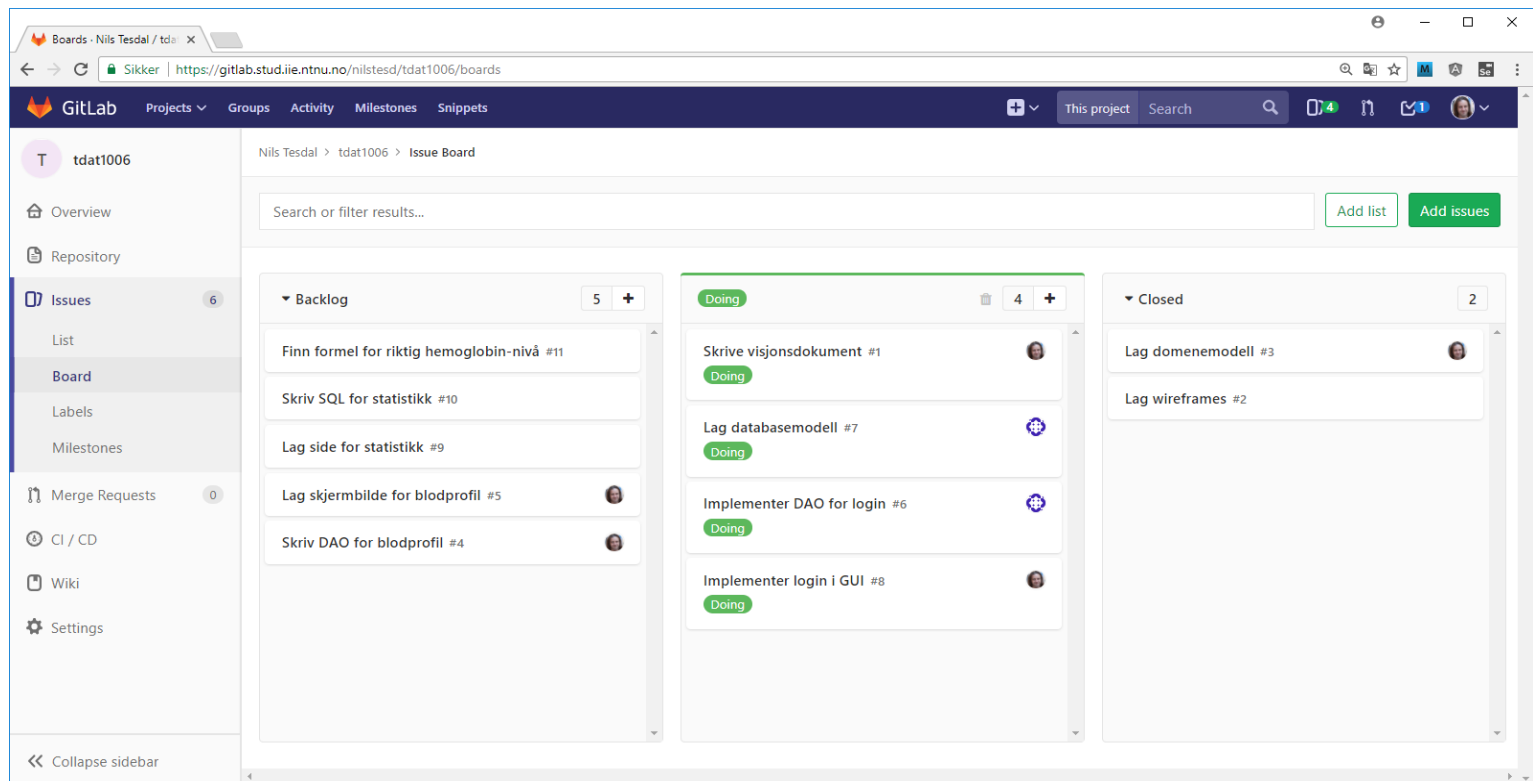
- **GitLab Repo:** det skal opprettes et «repo» for prosjektet for hver gruppe hvor all kode og dokumentasjon skal lagres. Alle team-medlemmene må ha utviklertilgang og vi (Ali, Surya og Grethe) skal ha reporter-tilgang.
<https://gitlab.stud.idi.ntnu.no>
- **GitLab WIKI:** All dokumentasjon skal skrives på WIKI eller linkes til fra WIKI'en (f.eks visjonsdokumentet som er en pdf).
- **Mattermost:** chat med kunde for å avklare krav og eventuelle andre spørsmål
 - https://mattermost.stud.idi.ntnu.no/signup_user_complete/?id=hofs eqwpyjnwmg9xs65fq5tckh

Prosjektverktøy

- ***GitLab Issue-Board:***
dere skal opprette en *backlog* med «GitLab Issues» og bruke et «Issue Board» for å visualisere progresjon i prosjektet.
- ***GitLab Pages:***
her dere skal publisere *JavaDoc* og eventuell annen statisk html som er en del av dokumentasjonen. Hvis dere legger til en fil, **.gitlab-ci.yml** på rota i repoet, med innhold som under, vil all html som ligger i mappa *javadoc* bli publisert til:
<http://brukernavn.pages.stud.idi.ntnu.no/prosjektnavn/>
(se eksempelprosjektet!)

Backlog / Issue Board (for each sprint)

- Create backlog items on the issue board
- Assign items
- Issue board must be dynamic (move items)



Wirefram tool: Balsamiq

Dere skal lage klikkbare wireframes for de viktigste skjermbildene i applikasjonen.

Balsamiq Classroom License – finner dere i
Blackboard

GitLab - Eksempelapplikasjon

- Lenke i GitLab
 - <https://gitlab.stud.idi.ntnu.no/surya/idatt1002>
- Kan klonenes eller brukes som inspirasjon
- Inneholder følgende
 - Enkel WIKI-forside
 - Issue-board
 - Kodestruktur

Eksempelapplikasjon

- Lenke finner dere i Blackboard
- Kan klonas eller brukes som inspirasjon
- Lager automatisk JavaDoc til GitLab Pages (`.gitlab-ci.yml`)
- Har Java-klasser med JavaDoc
- Viser forslag til enkel kodestruktur (MVC)
 - V - GUI-klasser: `myapp.view`
 - M - Data-klasser: `myapp.data`
 - C - DAO-klasser: `myapp.repo`
 - Alt mulig annet: `myapp.util`
- Enkel WIKI-forside
- Issue-board
- En test-klasse

JavaDoc

- Kommentarer i koden med spesiell syntaks:

```
/**
 * Repository for the MyEntity-entity
 *
 * @author nilstes
 */
public class MyEntityRepo {
    /**
     * Get object with given id
     *
     * @param id the entity id
     * @return an instance of MyEntity
     */
    public MyEntity getMyEntity(String id) {
```

- Blir til:
<https://surya.pages.stud.idi.ntnu.no/idatt1002/>

Biblioteker/teknologier

- Det forventes at dere bruker det dere har lært i Programmeringseminnene (idatt1001 og idatt2001)
- **Hele** gruppa må være enige om valg av teknologier utover det som dere bruker i programmeringsfagene!
- Dobbeltsjekk også med faglærer hvis dere gjør spesielle teknologivalg.

Vurdering

Vanligvis gruppekarakter,
men gis også individuell
karakter

- Obligatorisk Arbeidskrav (Uke 10)
 - Deltakelse på DevOps-workshop
- Vurderingskriterier knyttes til
 - Prosjekt (45/100)
 - Oppmøte veiledningsmøter med
 - Faglærer (alle på teamet må møte)
 - Læringsassistent (minimum 2 stk fra teamet må møte)
 - Deltakelse/ bidrag i prosjekt dokumenteres m timelister – (**individ-basert**)
 - Dokumentasjon (**team-basert**)
 - Implementasjon - kode/ produkt (**team-basert**)
 - Refleksjonsnotat (**individ-basert**)

Vurdering av prosjektet skjer på bakgrunn av 3 deler der alle tre deler må være bestått:

Del 1: Dokumentasjon 45%

- Visjonsdokument
- Krav (ihht til oppgavebeskrivelsen)
 - Use Case modell og tekstlig beskrivelse
 - Domenemodell
 - Wireframes
 - Usability
 - Universell utforming
 - Brukertester
 - Testing
- Sluttrapport
 - teori

Konsistens mellom utviklet system og dokumentasjon

Del 3: System/ Produkt 35%

- Presentasjon av ferdig applikasjon
- Funksjonalitet
- Struktur
- Konsistens mellom utviklet system og dokumentasjon
- Klassediagram
- Sekvensdiagram
- Db-modell (hvis relevant)
- Egnethet
- Installasjonsveiledning
- Brukermanual
- Kodekvalitet (kohesjon/coupling/ kommentarer/ lesbarhet)

Del 2: Prosess 20 %

- Issue-board
- Ryddighet når det gjelder planer og statusrapporter
- Møteinnkallinger og møtereferater
- Evne til selvstendighet, initiativ
- Teamsamarbeid
- Arbeidsinnsats (dokumentert)
- ***Evne til å ta feedback*** (ie tilbakemelding fra veileder og brukertester)

Prosjektoppgaven - 2023

- Verktøy for økonomistyring for
 - en person eller
 - personer med felles økonomi (familie) eller
 - en liten virksomhet
- Funksjonalitet
 - skal gi oversikt over økonomien
 - hjelpe med beslutninger
 - budsjett
 - regnskap

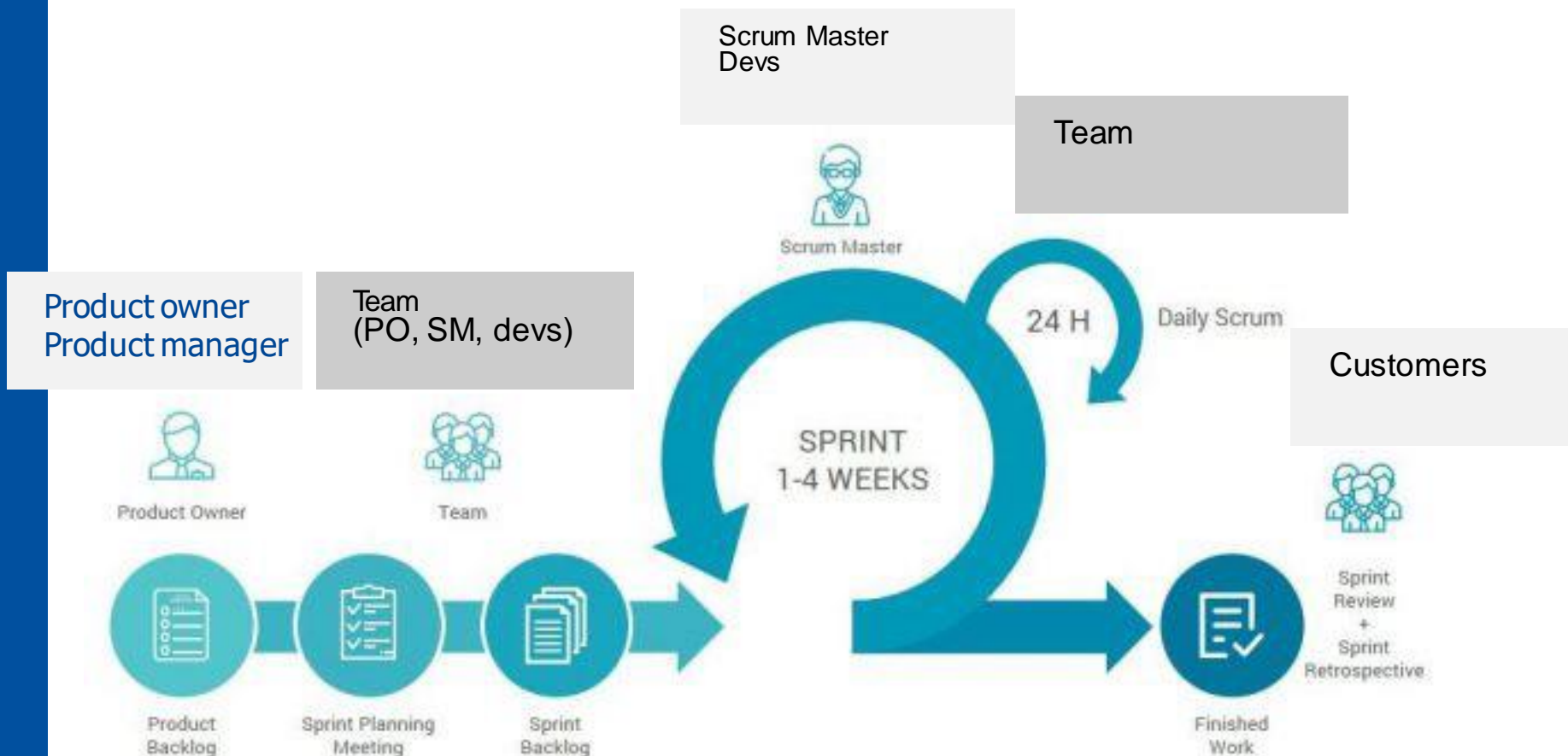
Prosjekt 2023 oppdragsgiver(e)

- Hver gruppe har ansvaret
 - å finne noen som kan være (spille rollen) som oppdragsgiver og bruker (ikke dataingeniør, utvikler eller student på data)
 - gjøre en avtale med disse om å bidra til prosjektet (innledende samtaler og diskusjoner, tilbakemelding på utviklernes forslag, minst to runder med brukertesting)
- Erfaringer fra tidligere år
 - interessert(e) og aktive(e) oppdragsgivere vil bidra til et interessant prosjekt og et godt resultat
 - det kan ta tid å finne og få et forpliktende samarbeid med oppdragsgiver

Prosjekt 2023 innledende arbeid

- Anbefalinger
 - idemyldring i gruppen om hva slags applikasjon og hvem som kan/vil bidra
 - tenke på brukernes behov
 - diskutere egne erfaringer fra området
 - lese om økonomi og lære terminologien
 - arbeide videre med flere alternativer og velge det mest lovende

Scrum – smidig utviklingsmetode



<https://www.mshowto.org/agile-ve-scrum-nedir.html>