



Viktoria



Antonio



Gunnar



Yngve



Hårek



Johanne



Marthe



Jan Henrik



Karthik



Jay



Marius

Software Engineering group

- Located at the 10th floor

+

•

○

Software Engineering Group

- Contact

- Antonio Martini – antonima@ifi.uio.no (group leader)
- Viktoria Stray – stray@ifi.uio.no
- Yngve Lindsjörn – ynglin@ifi.uio.no
- Gunnar Rye Bergersen – gunnab@ifi.uio.no
- Marthe Berntzen – marthenb@ifi.uio.no
- Jan Henrik Gundelsby - janhenrik@gundelsby.com
- Johanne Thunes - johanthu@ifi.uio.no
- Steffen Almås - steffa@ifi.uio.no
- Karthik Shivanskar - karths@ifi.uio.no
- Jayeong Song - jayeongs@ifi.uio.no
- Hårek Haugerud - haarekh@ifi.uio.no
- New postdoc being researcher hired at TRUST

Software Engineering - SE

- SE teaches and conducts research in a variety of topics in software engineering. The research is primary within the following areas:
- Agile methods
- Teamwork
- System development
- Technical debt
- Software Architecture
- Software testing
- Evaluation of IT competences and skills
- Software Process Improvement
- Research methods in software engineering
- AI for software engineering
- Software engineering for AI
- Software Visualization

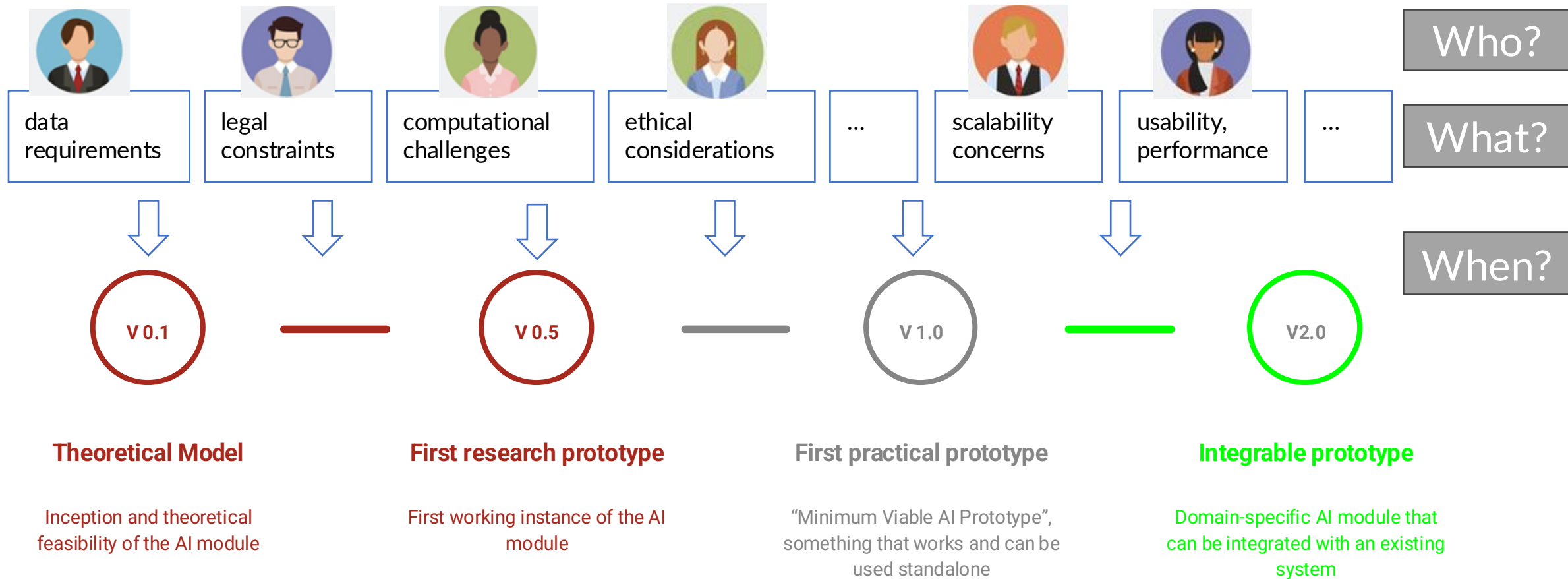
Software Engineering (SE) - Master Courses

- IN5140 – Smart processes and agile methods in software engineering (Autumn)
 - The course will give a solid fundament for participating in and manage the software projects of an organization and for improving the way such projects are conducted. Focus on Process Improvement and agile methods.
- IN4240 – Software Testing (Spring)
 - This course covers fundamental concepts in software testing and specific areas such as: testing through the life cycle of a software system, static testing techniques, test design techniques, test management, testing tool support, testing user experiences, accessibility testing, exploratory testing, automated testing, test-driven development.
- (planned) AISE – AI Software Engineering (Spring, Master)
 - Learn how to build, deploy and integrate a trustworthy AI-powered application using AI agents. Learn how to choose the right technologies and make important decisions affecting sustainability, security, economy, sovereignty, etc.

Master theses Antonio Martini - 1

- TRUST center: P2P-AI (prototype to product)
 - Research on how to bring a ML prototype to be integrated in a product
 - Multidisciplinary (in collaboration with Law, Philosophy, SINTEF, NR)
- Co-supervised by newly hired postdoc at TRUST
- TRUST: Norwegian center for trustworthy AI <https://www.trust-aicentre.no>

P2P-AI (Prototype 2 Product) project



Master theses Antonio Martini - 2

- Engineering AI (Agents)
 - Harnessing Agents
 - avoiding Cognitive Debt



7 Agent Harness Components



Agent = Model + Harness

Co-supervised by Karthik Shivashankar, experienced AI engineer also working at SINTEF

Johanne Thunes

- Flere mulige problemstillinger, blandt annet:
 - **AI transformasjon:** Hvordan endrer AI software utvikling? F.eks Vibe-koding, AI agenter og low-code/no-code
 - **Citizen's development:** Hvordan åpner AI opp for at folk med ikke-teknisk bakgrunn kan utvikle digitale løsninger?
 - **AI sourcing:** Hvordan påvirker AI strategiske beslutninger knyttet til hvilke digitale løsninger bedrifter kjøper eller velger å bygge selv? Hvilken kunnskap må bedrifter ha internt og hvilken kan/bør hentes inn fra eksterne konsulenter?
- **AI i utdanning:** Teamarbeid i IN2000, og Studenter bruk av AI i utdanning
- Samarbeid med bedrifter og SINTEF i et forskningsprosjekt støttet av forskningsrådet <https://www.knowit.no/forskning/kairos/>
- Hovedsakelig masteroppgaver basert på kvalitativ metode, med bl.a intervju og observasjon i norske bedrifter. Krever skandinavisk språk



Viktorija Stray

- Flere problemstillinger mulig. For eksempel
 - Hvordan sikre en god arbeidsdag i et smidig utviklingsprosjekt?
 - Hvordan skal produktteam organiseres og hvordan jobber de best?
 - Hvordan brukes generativ KI av utviklere? Muligheter og farer.
 - Hvordan sikre autonome effektive smidige team?
 - Hvordan jobbe smart og få til en god digital smidig transformasjon?
 - Teamarbeid i IN2000
 - Studenters bruk av Generativ KI / KI i utdanning
 - AI som konkurransefordel for norske selskaper
 - Samarbeid med bedrifter og SINTEF i et forskningsprosjekt støttet av forskningsrådet <https://www.knowit.no/forskning/kairos/>
 - Oppgaven krever at man snakker skandinavisk da man skal gjøre intervjuer og observasjoner i en norsk bedrift



Viktoria Stray

- Systematic literature review
 - Of a selected topic in Software Engineering. For example,
 - Use of theories in Software Engineering
 - Meetings
 - Coordination
 - Agile coaching
 - Agile Software Development
 - Teamwork
 - Can be short or long thesis

Marthe Berntzen – Marthenb@ifi.uio.no

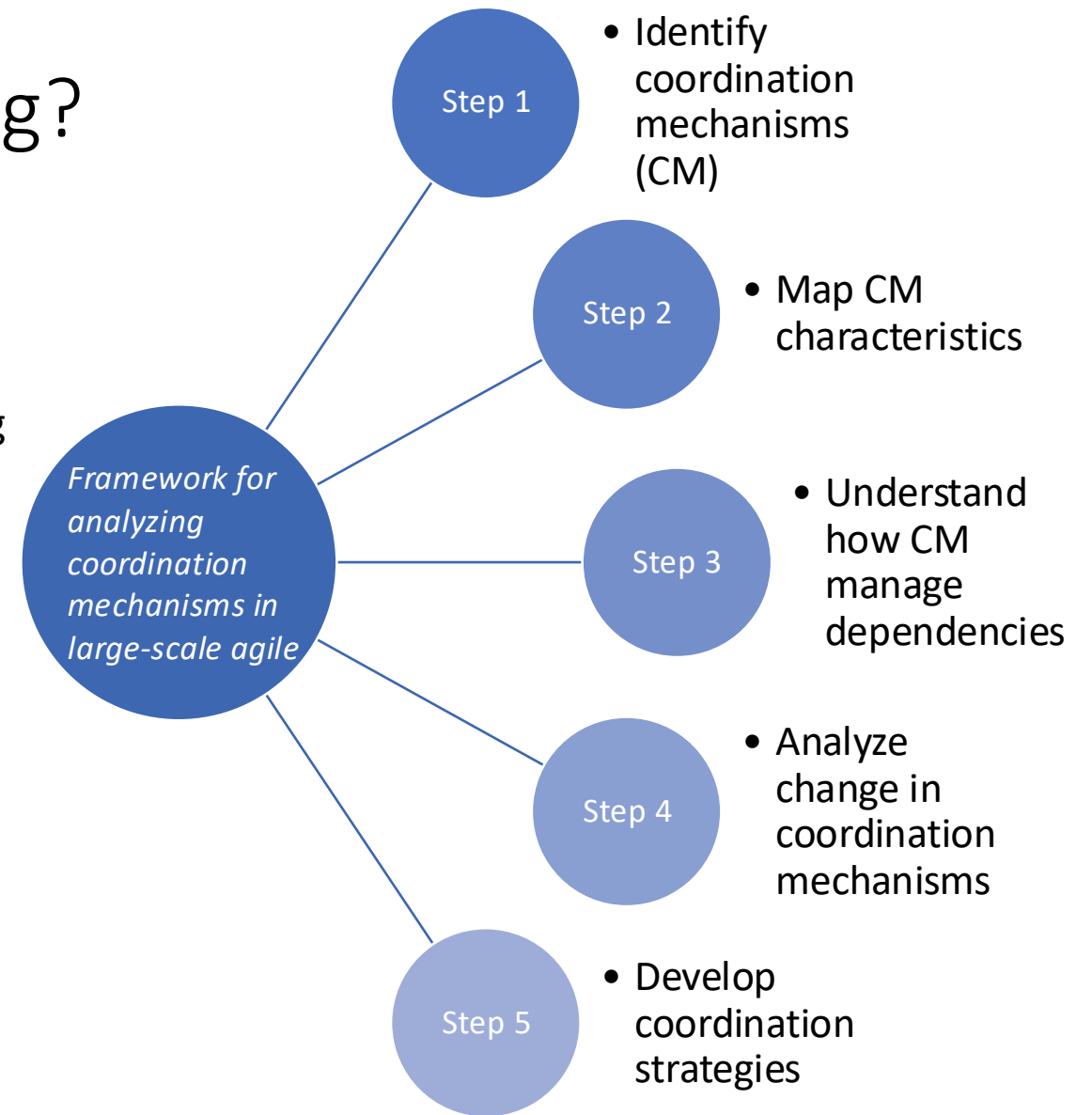
- Currently
 - Postdoctoral researcher in Software Engineering
 - Researcher and project manager at Knowit
 - Leading a 4-year research project on AI in software engineering
- Research areas
 - Large-scale agile
 - Team- and inter-team coordination
 - Product management and product organization
 - The role of AI in software engineering teamwork and processes



1. Hvordan optimalisere koordinering i AI-drevet utvikling?

- Koordinering innad i og mellom team forblir en av de største utfordringene i organisering av softwareutvikling.
- Det er uklart hvordan AI-agenter påvirker koordinering og samarbeid i softwareutvikling.
 - Hvilke koordineringsmekanismer trenger vi nå? Hvordan skal vi bruke dem i dagens AI-landskap?
- Oppgaven vil anvende et eksisterende rammeverk for analyse av koordinering (Berntzen, 2023)
- Andre innfallsvinkler kan også diskuteres 😊

Problemstillingen forutsetter at man snakker skandinavisk fordi datainnsamling (intervjuer, observasjon) vil gjennomføres i bedrifter der norsk er arbeidsspråk



2. Systematic literature review on Thematic Analysis in software engineering

- *Thematic analysis* is an analytical framework that is widely used in qualitative software engineering (SE) research. In this thesis, the student will conduct a systematic literature review (SLR) of the software engineering literature
- Possible research questions:
 - How much many studies have used thematic analysis in SE research?
 - How is it used?
 - What type of research questions is it used for?
 - What type of publication outlets publish TA research?

It is also possible to explore how AI tools can assist in the SLR process 😊

Steps in a systematic review





3. Et kontinuerlig observatorium for AI-bruk i utviklingsteam

Vår kunde, Tet digital, bygger et observatorium: et system som samler metadata om hvordan utviklingsteam jobber med AI. Studenten skal undersøke hva metadataene faktisk kan si om teamets arbeid. Noen mulige problemstillinger:

- * Hvilke metadata fra systemer som for eksempel Slack/Teams, issue-tracker, Git, CD/CI-pipeline og AI-bruk sier noe om teamets effektivitet og kvalitet, og hvilke måler bare aktivitet?
- * Hvilke samarbeidsmønstre i og mellom team kan avleses fra metadata?
- * Hva kan dataene si om team arbeidsprosesser, sammensetning og samarbeid?
- * Hva kan metadata fortelle om sikkerhet og arkitektur?

Veiledere: Marthe Berntzen, postdoktor ved IFI og Jan Henrik Gundelsby, leder R&D i Knowit.

Oppgaven forutsetter at man snakker skandinavisk fordi datainnsamling vil gjennomføres i en bedrift der norsk er arbeidsspråk.

Gunnar Rye Bergersen

Associate Professor - Software Engineering

Unge som tar *informerte utdanningsvalg* vil bidra til å tette kompetansegapet i Norge.

Vi skal bygge en plattform som gir unge de nødvendige verktøyene for å gjøre de riktige valgene for egen utdanning og karriere

Unge som tar *informerte utdanningsvalg* vil bidra til å tette kompetansegapet i Norge.

Vi skal bygge en plattform som gir unge de nødvendige verktøyene for å gjøre de riktige valgene for egen utdanning og karriere



App development for kids' career choices (React, Next.js, Payload CMS)

✓ 6 IFI MSc students (from 2025) have build infrastructure for first version of app

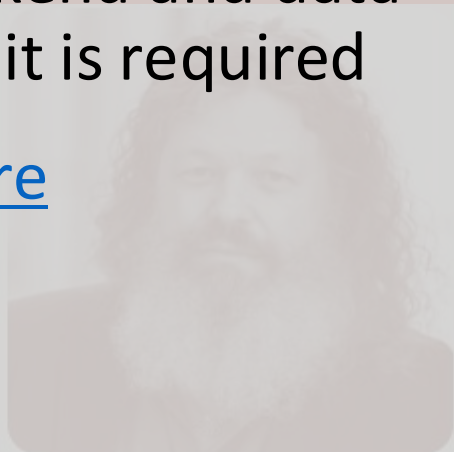
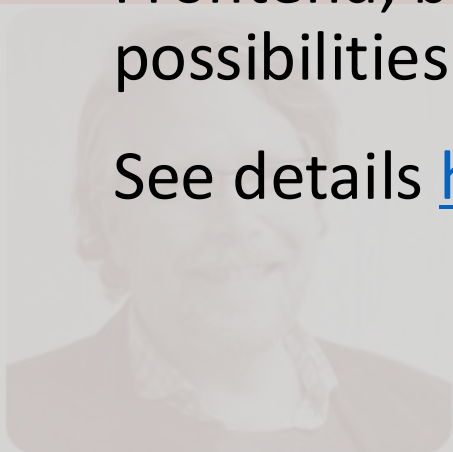
➔ **TODO: build, test, refine and evaluate new or changed functionality**

Frontend, backend and data possibilities. Git is required

See details [here](#)

Method: Design Science or mixed-methods (quantitative and qualitative)

Contact: Gunnar Bergersen
gunnab@uio.no



Building LLM Feedback Systems for Open-Ended Software Engineering Tasks

Supporting students while they work on a task

Jayeong Song, Gunnar Bergersen

Software Engineering Group

Department of Informatics, University of Oslo

17 Sep 2026



**UNIVERSITY
OF OSLO**

Building LLM Feedback Systems for Open-Ended Software Engineering Tasks

Motivation & Problem

- Feedback supports learning
- Students need feedback while working on the task
- Open-ended SE tasks allow multiple valid solutions, making providing feedback challenging
- LLMs offer new possibilities for scalable, real-time personalized feedback

Relevant Skills & Experience

- Build & deploy GenAI solutions
- Work with & evaluate LLMs
- Python & data analysis
- Web development
- API & database integration

Your master thesis topic

- Choose an open-ended SE task (coding or non-coding)
- Build a new prototype/MVP or extend an existing one
- Define your own RQs
- Build on insights from four current master's students

Supervisors



gunnab@ifi.uio.no



jayeongs@ifi.uio.no

Examples of Open-Ended Tasks in SE

Programming

Implement a feature or solve a problem in code.

```
fun calculateTotal(
    items: List<Item>
): Double {
    return items.sumOf {
        it.price * it.qty
    }
}
```

Refactoring

Improve existing code while keeping the same behaviour.

```
class UserService {
    fun getUser(id: Int): User {
        if (id != null) {
            return repo.find(id)
        } else {
            return User()
        }
    }
}
```



```
class UserService(
    private val repo: UserRepository
) {
    fun getUser(id: Int): User? {
        return repo.find(id)
    }
}
```

Testing

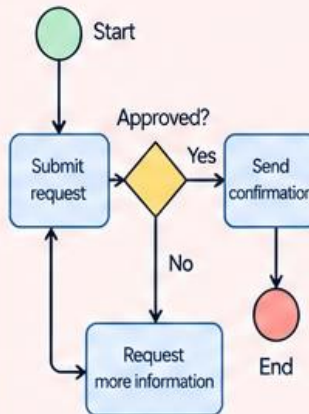
Design and write tests to verify software behaviour.

```
@Test
fun add_returnsSum() {
    val result = calculator.add(2, 3)
    assertEquals(5, result)
}
```

✓ testAdd() Passed
✓ testNegative() Passed
✓ testZero() Passed

BPMN Process Modeling

Create a process model to describe a business process.



Requirements Engineering

Elicit, analyse and specify requirements for a system.

Requirements for Study Planner App

- 1 Users can create courses
- 2 Users can set deadlines
- 3 The system sends reminders
- 4 Users can track progress
- 5 ...

Prompting (Asking LLMs)

Write effective prompts to solve SE tasks with LLMs.



Refactor this code to improve readability and explain the changes.



Here is the refactored code with explanations:

```
class UserService {
    fun getUser(id: Int): User {
        if (id != null) {
            return repo.find(id)
        } else {
            return User()
        }
    }
}
```

Key changes:

-
-
-

Short-Answer Questions

Answer conceptual questions about SE principles, methods and practices.

Q

Explain the key differences between Lean and Agile.

A

Lean focuses on minimizing waste and maximizing efficiency, while Agile emphasizes frequent releases and collaboration. Lean uses a pull system, while Agile uses a push system.

Code based

Non-Code based

Possible Research Directions & Example RQs

Possible Research Directions

- **What** should we give feedback on?
 - Artifact or process
- **How** should we generate the feedback using LLM?
 - Prompting, RAG, single/multi-agent
- **When** should we give the feedback?
 - After submission, during the task, when needed
- Can we **evaluate** the quality and impact of the feedback?
 - Feedback quality, impact on students such as learning outcomes, preferences, motivation

Example RQs

- To what extent can LLMs provide feedback on students' prompts during AI-assisted refactoring?
- To what extent does multi-agent feedback improve accuracy and usefulness compared with a single LLM?
- How does interactive feedback during coding influence students' problem-solving skills?