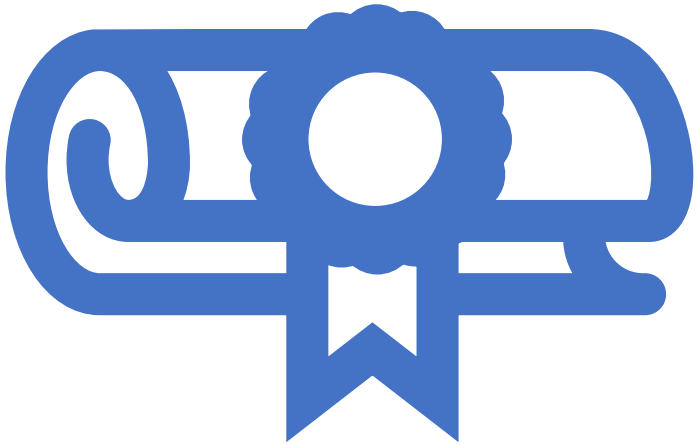


Software Engineering & Testing Eksamen

Prosjektinnlevering

Fredag 22/11 kl. **14:00!**

Hva skal leveres?

Zip-mappe med git-repoet deres	Prosjektdokumentasjon	Timelister, fullverdige	Refleksjonsnotat (Frivillig)

Prosjektinnlevering – git-repo

- Innleveringen SKAL inkludere en lokal mappe av git-repoet deres (zip-mappe)
  - .git-mappen MÅ være inneholdt
  - Kan gjerne linke til GitHub-repoet i tillegg, men den lokale mappen **skal** være med
  - Pass på å merge alle endringer inn i main/hovedbranchen før dere leverer
  - Kan gjerne inkludere en kort README-fil som beskriver overordnet hvor man finner hva
    - Prosjektfiler
    - Prosjektdokumentasjon
    - osv.

Prosjektinnlevering – Prosjektrapport (pdf)

Trenger IKKE være skrevet i rekkefølgen presentert her

- Dokumentasjon av hvordan dere har jobbet (Prosess)
  - Sprints, Scrum, bruk av git osv.
  - **Diskusjon rundt erfaringer**
- Dokumentasjon av prosjektet
  - Kravspesifikasjon / Mål
    - Personas, Brukerhistorier osv.
  - Arkitektur – F.eks. med modeller
  - Avhengigheter med begrunnelse for hvorfor
  - Eventuelle kodeprinsipper dere har fulgt
  - Hva og hvordan dere har testet
  - Hvordan kjøre prosjektet og tester
    - Hva er nødvendig av miljø-oppsett?
    - Kan være fornuftig å presentere resultat av kjøring (i prosjektrapport, video, e.l.)
  - **Diskusjon rundt valg gjort underveis**
  - **Diskusjon rundt levert produkt og tiltenkt videre arbeid**
- «Diskusjon» → Hvorfor vi gjorde... Hva som funket bra. Hva vi burde gjort annerledes og hvordan.

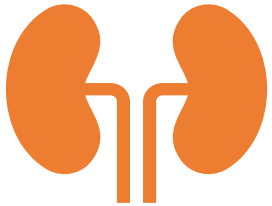
Prosjektinnlevering - Timelister

- Dere bør ha ført timer per gruppemedlem underveis i prosjektet
- ... Men dersom dere ikke har gjort dette. Beskriv i det minste
  - Hvem som har vært ansvarlig for hva
  - Ca. hvor mye tid dere har benyttet på forskjellige deler i prosjektet
    - F.eks. Per sprint, overordnede komponenter / biter med funksjonalitet
  - Osv.

Prosjektinnlevering - Refleksjonsnotat

- Refleksjonsnotat
  - **Frivillig!**
  - Kan være skrevet som en gruppe eller per medlem
    - GruppereflekSjon vil egentlig ofte inngå i diskusjoner i prosjektrapport
    - Egen individuell innlevering er tilgjengelig for individuelle medlemmer
  - Kan være nyttig hvis det er ting som er litt på siden av alt annet
    - Gruppeproblemer / ubalansert innsats
      - Bør kunne støttes av timelister
    - Personlige erfaringer og følelser rundt prosjektet
      - Hva JEG lærte. Hva JEG kunne gjort bedre. Osv.

Skriftlig eksamen



3 timer skriftlig - 11/12

Hjelpemidler i form av kursmateriale gjøres tilgjengelig på eksamensdagen

Går mye på refleksjon / forståelse av gjennomgåtte konsepter

Tekstlige svar – Ikke flervalg

Lite fokus på begrepsdefinisjoner, mer hvordan å benytte konsepter / forståelse

- Typisk basert på et case

Ikke fullstendige kodeoppgaver, men kan be om å definere tester

- **Pseudokode** eller valgfritt programmeringsspråk
- Syntaks er IKKE viktig

Pseudokode

- Betyr å beskrive kodelogikk med vanlig språk
  - Altså uavhengig av syntax / programmeringsspråk
- Holder kodelogikk mer overordnet
  - Typisk på nivå av konseptuelle kodelinjer / kodesekvenser, f.eks. i en metode / funksjon
  - Hva som skal utføres / oppnås
    - Ikke like viktig å beskrive nøyaktig hvordan
    - Bør likevel skrives i tiltenkt rekkefølge

Peudokode

Ren kode

```
public ArrayList<Student> getStudentsInListOfPersons(
                ArrayList<Person> persons) {

   ArrayList<Student> students = new ArrayList<>();

   for (Person person : persons) {
      if (person instanceof Student) {
         students.add((Student) person);
      }
   }

   return students;
}
```

Pseudokode

Metodedefinisjon for getStudentsInListOfPersons. Tar liste med personer som parameter:

Opprett tom liste for å holde på studenter.

Gjennomgå alle inneholde personer i person-listen, og legg til de som er studenter i student-listen.

Returner listen med studenter.

Alternativ:

Metodedefinisjon for getStudentsInListOfPersons. Tar liste med personer som parameter:

Opprett tom liste for å holde på studenter.

For hver person i listen med personer,
hvis personen er en student,
legg den til i listen med studenter.

Returner listen med studenter.

Psudokode

Pseudokode er også nyttig for å overordnet beskrive gjentagende (kjedelige) handlinger.

- Meget nyttig på en eksamen hvor fokuset ligger på forståelse ...

Ren kode

```
Student student1 = new Student(/*....*/);
Student student2 = new Student(/*....*/);
Student student3 = new Student(/*....*/);
Lecturer lecturer1 = new Lecturer(/*....*/);
Lecturer lecturer2 = new Lecturer(/*....*/);
```

Pseudokode

Oppretter 5 objekter hvorav 3 av de er studenter og 2 av de er forelesere. Alle defineres med unik informasjon gjennom parametere.

Pseudokode – Tips til forberedelse

- Øv på å skrive pseudokode av ...
  - Tidligere gjennomgåtte eksempler av enhetstester
  - Egne tester skrevet i forbindelse med prosjektet
- Sørg for at du beskriver nok til å vise at du forstår enhetstest-konsepter
  - Arrange, Act, Assert
  - Mocking / Test Doubles
  - Parameteriserte tester
  - osv.
- Du kan også kombinere vanlig kode og pseudokode
  - Skrive vanlig kode som standard, men skrive pseudokode der vanlig kode hadde krevd unødvendig mye detaljer
    - Objektparametere, gjentagende handlinger som skilles med nyanser osv.

Relevante Temaer for Skriftlig Eksamen

- Det vi vært gjennom i forelesninger ...
  - Fokus på overordnet og helhetlig forståelse, ikke implementasjon

- Krav: Personas, Scenarier, Brukerhistorier, Estimering
- Agile utvikling
- Versjonskontroll og overordnet forståelse av git-operasjoner/-funksjonalitet
- Testing og Enhetstesting (krever litt forståelse av koding)
  - Arrange, Act, Assert
  - Test Doubles
  - Test Driven Development (TDD)
- Avhengigheter
- Arkitektur og løst koblet kode
- Diskusjon rundt modellering
- Continuous Integration og Continuous Deployment
- Containerization og Orchestration