

Eksamensoppgave for Backendprogrammering (PGR209)

- **Emnekode:** PGR209
- **Emnenavn:** Backendprogrammering
- **Vurderingskombinasjon:** Funksjonalitet og kodekvalitet i fungerende program deployet på Azure
- **Innleveringsfrist:** Mandag 21. november 09:00
- **Filformat:** Github classroom-lenke og ZIP-fil fra Github med Java-kildekode og README.md-fil med link til deployment på Azure

Innlevering

- Oppgaven **skal** løses i Github vha Github Classroom med link fra Canvas. Repository på github **skal** være private
- Oppgaven **skal** leveres i *Wiseflow* som en ZIP-fil og link til Github Classroom. Bruk Github > Code > Download ZIP for å opprette ZIP-fil
- README.md på Github **skal** linke til Github Actions som skal kjøre enhetstester uten feil
- README.md på Github **skal** linke til deployet website på Azure
- Oppgaven **skal** løses parvis. Ønsker du å levere alene eller i gruppe på tre må dette avklares med foreleser innen siste forelesning
- Innleveringen kan deles med 1-2 andre grupper for gjensidig tilbakemelding. Tilbakemelding bør gis i form av Github issues
- Koden **skal** lese database settings fra en fil som heter `application.properties` og ser ut som følger:

```
dataSource.url=...  
dataSource.username=...  
dataSource.password=...
```

Oppgave

Oppgaven for PGR209 er å lage en applikasjon for å utveksle meldinger mellom brukere på nett.

Applikasjonen skal komme med et forhåndslagret sett med brukere som kan sende og motta meldinger. Brukeren av applikasjonen skal kunne velge hvilken bruker de skal representere og opprette, se på og besvare meldinger for denne brukeren.

Applikasjonen skal inneholde en database med Azure SQL Server, en backend implementer i Java med Jetty på Azure App Services og *en enkel* frontend i React som kommuniserer med backend med JSON. Programmet skal utvikles på en måte som demonstrerer programmeringsferdigheter slik det vises i undervisningen. Spesielt skal all funksjonalitet ha automatiske tester og være fri for grunnleggende sikkerhetssvakheter. Den innleverte koden skal demonstrere at kandidatene mestrer funksjonalitet for HTTP, JSON og databaser i Java.

Funksjonelle krav

Programmet skal støtte følgende funksjonalitet. Dere oppfordres til å implementere funksjonaliteten med egendefinerte felter og arbeidsflyt så lenge applikasjonen kan demonstrere at brukere kan opprette, se og besvare meldinger

1. Når man kommer til applikasjonen kan man velge hvilken bruker man skal representere fra et sett med brukere som dere har opprettet på forhånd i databasen
 - For full uttelling skal man kunne endre informasjon om en eksisterende bruker

- For full uttelling må brukeren ha felter utover navn og emailadresse
- 2. Når man har valgt en bruker skal man kunne se alle meldingstråder der denne brukeren er sender eller mottaker (krav til E)
 - For full uttelling skal *oversikten* over meldingertråder inneholde vise per tråd navnet på alle mottakere for meldinger i tråden
 - For full uttelling bør dere forhåndspopulere med noen meldingstråder
- 3. Når man har valgt en bruker kan man opprette en ny meldingstråd med mottaker. Meldingen skal kunne inneholde tittel og meldingstekst (krav til E)
 - For full uttelling må det være mulig å opprette en meldingstråd til med flere mottakere
 - For full uttelling må det meldingen inneholde flere felter enn tittel og meldingstekst
- 4. Når en bruker velger en meldingstråd skal de se alle meldinger i tråden
 - For full uttelling må backend gjøre en join mellom melding- og brukertabellen for å vise avsenders navn
- 5. Når en bruker velger en meldingstråd skal det være mulig å svare på meldingstråden
 - For full uttelling må svaret inneholde flere felter enn meldingstekst
 - For ekstra poeng: Registrer når en bruker har lest en melding og vis dette tidspunktet til andre brukere som kan se meldingen

For å få A må bortimot alle punkter angitt med "for full uttelling" være løst. For B må en god del av disse være løst. For C må alle hovedpunkter være løst

Ikke-funksjonelle krav

- Applikasjonen skal kjøre på Azure (krav til D)
- Applikasjonen skal lagre data i Azure SQL Server (krav til C)
- Database skal ikke duplisere data mellom flere tabeller (krav til C). Spesielt skal navnet på avsender til en melding ligge i brukertabellen og ikke i meldingstabellen
- README.md skal inneholde et diagram over databasemodellen
 - Dere bør forvente å ha 4 eller flere tabeller med relasjoner mellom
- Koden skal følge standard Java-konvensjoner med hensyn til store og små bokstaver og indentering
- Koden skal utvikles på Git med Maven og kjøre en Jetty-basert webserver (krav til E)
- Bevarelsen tester på Github Actions (krav til D)
- Koden ha god testdekning (krav til B)

Utseende på frontend *er ikke* et vurderingskriterie for karakteren.

Vedlegg: Sjekkliste for innlevering

- ☐ Dere har lastet opp en ZIP-fil med navn basert på navnet på deres Github repository
- ☐ Koden er sjekket inn på github.com/kristiania-repository
- ☐ Dere har committed kode med begge prosjektdeltagernes GitHub konto (alternativt: README beskriver arbeidsform)

README.md

- ☐ README.md inneholder en korrekt link til Azure website
- ☐ README.md inneholder en korrekt link til Github Actions
- ☐ README.md beskriver eventuell ekstra leveranse utover minimum
- ☐ README.md inneholder et diagram som viser datamodellen

Koden

- ☐ `mvn package` bygger en executable jar-fil
- ☐ Koden inneholder et godt sett med tester
- ☐ `java -jar target/...jar` (etter `mvn package`) lar bruker legge til og liste ut data fra databasen via webgrensesnitt
- ☐ Programmet leser `dataSource.url`, `dataSource.username` og `dataSource.password` fra `pgr203.properties` for å connecte til databasen
- ☐ Programmet bruker Flywaydb for å sette opp databaseskjema
- ☐ Server skriver nyttige loggmeldinger, inkludert informasjon om hvilken URL den kjører på ved oppstart

Funksjonalitet

- ☐ Websiden inneholder et forhåndsdefinert sett med meldingsbrukere
- ☐ Bruker kan velge hvilke meldingsbrukere de skal representere
- ☐ Bruker kan opprette en ny melding til en annen bruker
- ☐ Bruker kan se mottatte meldinger
- ☐ Bruker kan besvare meldingen
- ☐ Meldinger vises i en tråd med svar