



Utvikleroppgave

Oppgavebeskrivelse

Innledning

Formålet med oppgaven er å prøve å lage noe ved hjelp av relevante teknologier – mest for å vise hvordan man tenker og løser problemer mer enn å bli ferdig eller hvordan et evt. ferdig resultat ser ut.

Oppgave

Oppgaven består i å lage et system bestående av en front-end og en back-end og som kan brukes til å vise og redigere en liste med navn og alder for et sett personer. Det skal være mulig å redigere både navn og alder for hver person samt slette og legge til personer. Data skal kunne hentes fra og lagres til back-end fra front-end.

Datamodell

Datamodellen består av en liste med personer, en personliste, hvor modellen for hver person består av en tekststreng som inneholder hele navnet til personen og et tall som inneholder alderen til personen.

Back-end

Back-end skal lages ved å bruke Java-rammeverket Spring Boot^{[1][2]} og bruke byggesystemet Gradle^[3]. Kommunikasjon med front-end skal skje via REST-endepunkter som back-end API og data skal overføres i JSON format. Data trenger ikke å bli lagret til disk/database, men må kunne lagres i minne mens applikasjonen kjører. Som språk kan man bruke enten Java eller Kotlin^[4] – man velger selv det man synes passer best for seg. Koden må kunne kjøres med Java 8 eller nyere versjoner av Java. Det skal være mulig å bygge en selvkjørende JAR-fil vha. byggesystemet.

Front-end

Front-end skal lages ved bruk av JavaScript-rammeverket Vue.js^[5]. Vue.js kan «installeres» på forskjellige måter^[6] og man velger selv hvilken måte som man synes passer best for seg. Man kan, om man synes det er bedre, bruke UI komponenter fra Vuetify.js^[7] istedenfor å lage mark-up/komponenter selv. Endelig resultat kan se ut som forslaget nedenfor, men man kan endre på dette om man har andre idéer om hvordan dette kan løses.

Navn	Alder	X
Navn	Alder	X
Navn	Alder	X
<div>+ Add</div>		
<div>Reload Save</div>		

Forslag til stegvis progresjon

Da oppgaven kan være omfattende når teknologien og rammeverkene er ukjente kan det være en god idé å sette opp en stegvis progresjon slikt at man kan gjøre noen steg ferdige og heller kutte andre steg slik at man har en mulighet for å komme i mål med noen punkter heller enn å være halvferdig med alle.

En måte å gå frem på kan være:

1. Lage en Spring Boot back-end-applikasjon som bruker Gradle som byggesystem. Applikasjonen trenger ikke gjøre mer enn å skrive ut «Hello World!» på et eller annet vis.
2. Utvide back-end med et REST-enderpunkt som skriver ut «Hello World!»
3. Utvide back-end til å kunne sende ut en forhåndsdefinert personliste via REST-enderpunktet i JSON format.
4. Lage en front-end-applikasjon som viser teksten «Hello World!» i nettleseren.
5. Utvide front-end til å vise en forhåndsdefinert (i front-end) personliste.
6. Utvide front-end til å hente personlisten fra back-end og vise denne.
7. Utvide front-end til å tillate redigering av person-navn og -alder.
8. Utvide back-end til å ta imot en personliste.
9. Utvide back-end slik at den, ved neste forespørsel, sender ut den sist mottatte personlisten tilbake til front-end.
10. Utvide front-end slik at man kan legge til flere personer.
11. Utvide front-end slik at man kan slette personer.

Merk: Dette er bare et *forslag* – man kan gjøre dette som man selv vil (f.eks. starte med front-end istedet for back-end eller dele opp oppgaven på en helt annen måte).

«Innlevering»

Når man er ferdig med oppgaven (så langt man rekker) gjøres koden for både back-end- og front-end-prosjektene tilgjengelig via et GitHub-repo^[8].

Referanser

1. <https://spring.io/projects/spring-boot>
2. <https://spring.io/guides/gs/spring-boot/>
3. <https://gradle.org/>
4. <https://kotlinlang.org/>
5. <https://vuejs.org/> (versjon 2 eller 3)
6. <https://vuejs.org/v2/guide/installation.html> for v2, eller <https://v3.vuejs.org/guide/installation.html> for v3.
7. <https://vuetifyjs.com/en/>
8. <https://github.com/>