

PG6300 Webutvikling og API-design: Mappeelement 1

Det er en selvfølge at en organisasjon har en egen dynamisk nettside i 2016. Du skal lage en applikasjon bestående av server og webklient for selskapet PP Records BRL ASA Inc., som lister opp alle album de har tilgjengelig. En liste over album finnes i fila `albums.txt`. I tillegg skal du kort besvare noen spørsmål.

Spørsmål

Besvarelsen på denne oppgaven bør totalt være på 250–300 ord.

1. Hva skiller et view-bibliotek som React fra tradisjonell webutvikling der serveren genererer HTML, og hvilke fordeler og ulemper har disse tilnærmingene?
2. Hva er en *Component* i React-sammenheng?
3. Forklar kort oppgavene til følgende komponenter med dine egne ord:
 1. React
 2. Node.js
 3. npm
 4. Babel
 5. Webpack

Features

Fra webklienten skal det være mulig å:

1. Se informasjon om alle tilgjengelige album (artist, tittel, og årstall) i en liste;
2. Sortere lista på enten artistnavn eller årstall;
3. Filtrere ("søke i") lista ved hjelp av et input-felt. Lista skal oppdateres ved hver endring i input-feltet. Det trengs *ikke* å gjøres spørringer til serveren for å oppnå dette.

Tekniske krav

1. Webklienten skal benytte React for å generere all HTML mellom `<body>` og `</body>`, unntatt
 - én div: `<div id="container"></div>`, og

- referanse til bundle-en som bygges av Webpack: `<script src="..."></script>`.
- 2. Albumene skal hentes fra en Node-server ved hjelp av Fetch-API-et. Serveren skal benytte modulen Express og servere dataen som JSON. Tips: husk CORS (<http://npmjs.com/package/cors>). Grunnleggende REST-arkitektur skal følges.
- 3. Webklienten skal bygges med Webpack, og støtte "hot reloading" av komponenter (tips: få dette på plass tidlig).
- 4. Alle tredjepartsbiblioteker (for eksempel Express og Webpack) skal deklarerer på passende vis i fila `package.json`. Det skal være mulig å klargjøre og starte applikasjonen med følgende kommandoer i prosjektets rotmappe:
 1. `npm install`
 2. `npm run build`
 3. `npm start`

Bonusoppgaver

- Støtt eldre nettlesere som ikke støtter Fetch (<https://github.com/github/fetch>) eller Promise.
- Last inn `albums.txt` ved hjelp av `fs`-API-et til Node når applikasjonen startes.
- Lever kun kildekode (ikke tredjepartsbiblioteker eller kode generert av Webpack). Forutsetter at relevante *Tekniske krav* fremdeles møtes.
- Last inn eventuell CSS ved hjelp av Webpack.

Øvrig informasjon

- Besvarelsen skal leveres som ett pakket arkiv (`.zip` eller `.tar.gz`). Denne skal inneholde all kildekode som trengs for å klargjøre og kjøre applikasjonen, samt besvarelse på spørsmålene.
- Du skal lage applikasjonen på egen hånd. Det er lov til å ta inspirasjon fra internett og andre ressurser, men ikke kopier kode eller tekst uten å referere. Den største delen av arbeidet skal være ditt eget (unntatt tredjepartsbiblioteker).
- Første versjon av besvarelsen leveres på It's Learning innen 21. februar 2016 klokka 23:59. Frist for eventuell oppdatering annonseres senere.
- Eventuelle spørsmål rettes per epost til martin@westerdals.no.