

Eksamensdokument - Gruppe 15

Karaktergrad

Jeg har valgt å levere på karakter D-kraven. Jeg skulle egentlig først prøve ihvertfall å få opp til C-kravet men var i en situasjon der hvor jeg måtte prioritere andre eksamener/bacheloroppgaven som gjorde at jeg ikke fikk nok tid.

Github bruker: Kingzayadi = Sajad Adel Al-Zayadi

Githublenke: <https://github.com/Kingzayadi/billettlystgruppe15>

Hva jeg har gjort

Jeg startet prosjektet med å registrere meg på Ticketmaster Developer-portalen og skaffe API-nøkkel, som jeg brukte til å hente arrangementer. Deretter satte jeg opp prosjektet med create react app og installerte react router dom for å lage flere sider med visninger i applikasjonen.

Jeg begynte med å lage en home.jsx komponent hvor jeg hentet inn festivaler fra Ticketmaster sitt API. Jeg valgte ut noen festivalnavn og søkte dem opp med API-kall. Det kom opp en del utfordringer, spesielt med CORS-feil (ligger mer informasjon i utfordringer) For å omgå dette brukte jeg en CORS-proxy under utvikling, og reduserte antall kall. Til slutt valgte jeg å hente fire spesifikke festivaler (Roskilde, Heartland, Alive Festival og Midgardsblot) som returnerte data fra API-et. Nevnte festivaler som Findings og Neon fungerte ikke som gjorde at jeg ikke brukte de. For de som ikke ble funnet, blir det da vist et varsel med navnene.

Jeg implementerte navigasjon med en navbar som er synlig på alle sider. Jeg laget egne sider for dashboard, categorypage og eventpage. På eventpage brukes useParams til å hente ut ID fra URL, slik at man får detaljert info om det valgte arrangementet. Der vises bilde, navn, sted, dato og en lenke til å kjøpe billetter. I tillegg vises tilhørende artister med en artistcard komponent.

Categorypage viser arrangementer filtrert etter type som for eksempel musikk, sport eller teater. URL-stien bestemmer søketype, og jeg koblet slug opp mot riktig nøkkelord i API-kallet.

Jeg brukte også eventcard komponenten til å vise arrangementer i kortformat på forsiden, dashboardet og kategorisidene. Jeg har også jobbet med litt styling for at layouten skal være oversiktlig og responsiv.

På dashboard laget jeg en enkel innloggingsside som viser en velkomstmelding og dummy-arrangementer dersom brukeren fyller ut e-post og passord. Dette er kun frontend-funksjonalitet uten backend/autentisering.

GitHub ble satt opp tidlig, men brukte ikke branches eller commit på starten av prosjekt fordi jeg hadde jo jobbet med dem alene og tenkte da det var lettere hvis jeg hadde bare fortsettet med oppgaven. Jeg så da på oppgaven at det skulle bli vurdert på og brukte da commits og branches i midten av oppgaven. Noen av branches jeg brukte er for eksempel feature/artistcard, footer og kommentarer og jeg har skrevet forklarende commit-meldinger. Jeg har til slutt lagt inn kilder og kommentarer i kodebasen, og gjort opprydning i CSS og ubrukte filer før innlevering.

Utfordringer og det jeg ikke rakk

Jeg prøvde først å bruke søkeord som "Findings" og "Neon", men disse ga ikke treff i Ticketmaster sitt API. Det var en del av tingene jeg slet veldig mye med og fikk ikke gjort så mye. Jeg brukte også mye tid på å prøve å søke på nettet men selvom jeg refererte API id til de festivalene så kom det opp null resultater, så jeg startet med å bruke festivalnavn som "Roskilde" og "Heartland" og det fungerte istedenfor. Jeg vet ikke hvorfor det skjedde men jeg brukte de andre festivalene for å hente ut spesifikke arrangementer for å da bevise at jeg klarer å gjøre det men at det da blir forhindret med det som skjedde.

Jeg hadde en del problem med filene jeg lastet opp når jeg ikke klarte å få opp de 4 festivalene og det skjedde et feil i filen som førte at jeg måtte laste ned hele dokumentet fra github som jeg heldigvis ikke brukte commits som gjorde at jeg fikk lastet ned det jeg hadde som et backup, men jeg fikk da fikset det som trengs og fortsettet da med oppgaven.

Noen komponenter som categorypage viser flere teaterkort fra api som jeg da tenkte på om å kanskje filtere de bedre eller ikke men jeg tok valget da til å la dem være siden dataene var direkte fra api-et og det kan du se da at de er unike med å gå inn på vis mer detaljer på kortene.

Jeg har jo også jobbet med dette alene med begrenset tid, men følte at hvis det var et krav jeg kunne få gjort til slutt så var jo det minste jeg kunne oppnå en D-krav. Jeg har også sett på resten av eksamensoppgaven og da kanskje har gått litt langt og prøvd å gjøre noen ting bedre enn det jeg trengte (dashboard for eksempel, en fungerende logg inn skjema), men jeg følte da at dette var godt nok til å få minst en D.

Kilder brukt i oppgaven

Kildene under er det jeg har brukt for å forbedre oppgaven og for å da forstå hvordan jeg får oppfylt kravene. Jeg har referert dem i koden men jeg tenkte også det er lurt å bare lime inn alt her også. Jeg har gjort det da enkelt å gjøre det med å skrive da hvilket kilder som tilhører hva for å ha mer kontroll over de:

KILDER PÅ HOME.JSX:

Brukes for å hente festivaler en etter en i en løkke med await for hver api-forespørsel:

<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Statements/for-await...of>

Bruker corsproxy.io for å omgå CORS-feil i utviklingen

Kilde: <https://corsproxy.io/>

brukt for å sikre at programmet ikke krasjer når vi henter data som kanskje ikke eksisterer i API responsen.

Kilde: https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/Optional_chaining

KILDER PÅ ARTISTCARD.JSX:

Brukt for å sjekke at bilde og klassifisering eksisterer før det vises:

https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/Optional_chaining

KILDER PÅ DASHBOARD.JSX:

Brukt til å behandle input-felt for e-post og passord og for å endre visning av innhold etter innlogging (useState)

<https://react.dev/reference/react/useState>

KILDER PÅ EVENTPAGE.JSX:

Brukt for å hente ID fra URL-en slik at jeg kan hente riktig event fra API-et:

https://api.reactrouter.com/v7/functions/react_router.useParams.html

Brukt for å hente data fra Ticketmaster sitt API:

https://developer.mozilla.org/en-US/docs/Web/API/Fetch_API/Using_Fetch

brukes til å forstå hvilket felt som kan bli hentet ut som for eksempel: attractions, _embedded, osv...

<https://developer.ticketmaster.com/products-and-docs/apis/discovery-api/v2/#search-events-v2>

https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/Optional_chaining

KILDE PÅ EVENTCARD.JSX:

Brukt for å lage lenker til detaljerte sider (event/:id) for hvert event .

https://api.reactrouter.com/v7/functions/react_router.Link.html

Igjen, koden er kommentert der jeg har brukt ekstern inspirasjon, og jeg har lagt inn linker til dokumentasjon der det er relevant i koden. Som du kan se på kildene har jeg brukt mye fra developer mozilla siden det var en sikker nettside jeg kunne bruke. Jeg brukte også webtricks lms for å gjøre grunnleggende koding.