

Eksamen 2024 – Utvikling av interaktive nettsteder

Link til GitHub-repository: https://github.com/fitorevs/EKSAMEN_group12.git

Vanskelighetsgrad: A

Studenter:

- Fitore Victoria Seferaj: fitorevs
- Kinga Maria Jakubowska: Hatelyn
- Nikolai Lysebraate: ResistancelsCharacterForming

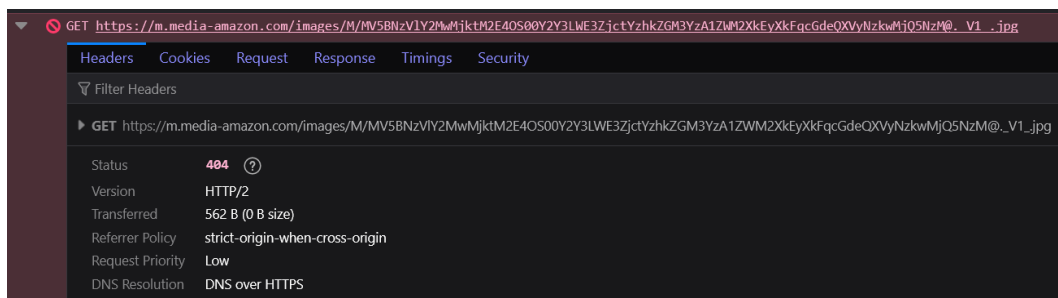
Brukere for testesystemet – Innlogging i «What to see»:

1. Brukernavn: J. Wayne
Passord: Whiskey
2. Brukernavn: ChatNoir
Passord: marinette123
3. Brukernavn: Askepott
Passord: 3nøtter

Utfordringer:

Gjennom eksamen i emnet "Utvikling av interaktive nettsteder" støtte vi på flere utfordringer som krevde omvurdering og nye løsninger. Vi opplevde gjentatte konflikter mellom branches, da denne arbeidsmetoden fortsatt er nokså ny for oss. For å sikre et ryddig og strukturert repository, besluttet vi å opprette et nytt og pushe våre individuelle og felles bidrag til det. Dette gjør det både enklere for sensor å lese og gir oss bedre oversikt over arbeidet vi har utført gjennom prosjektperioden.

I vår React-applikasjon opplevde vi også problemer med GET-meldinger i konsollen. Problemet vi står overfor er at hver gang vi prøver å fetche bildet fra URL gitt i API-et, kaster det følgende feil, fordi selve URL-en gammel og fungerer ikke lenger.



Vi har forsøkt flere løsninger for å håndtere denne feilen, men det ser ut til at problemet stammer fra API-et selv, og det er derfor ikke noe vi kan gjøre på vår side for å løse dette.

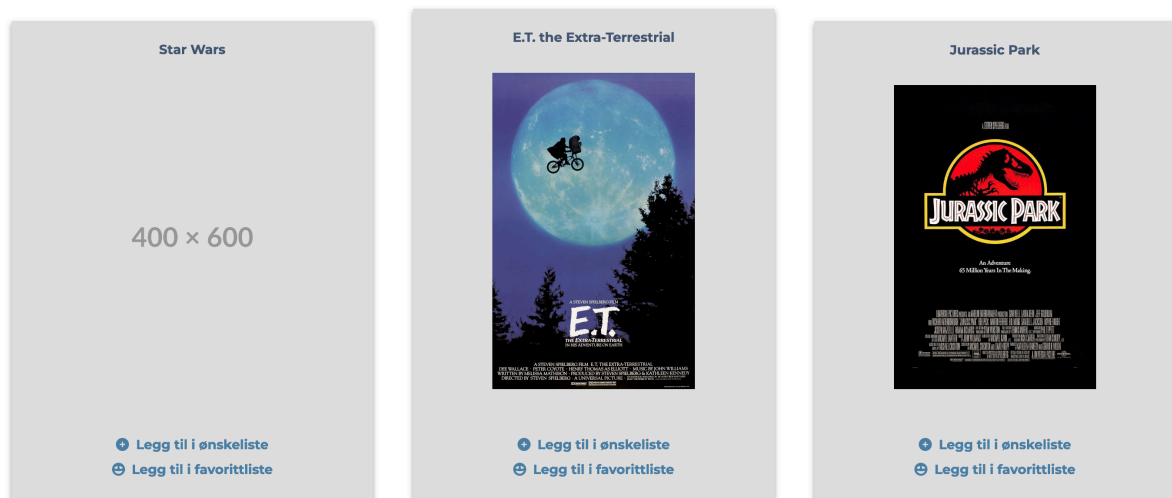
Vi har i tillegg lagt til en ny funksjonalitet i form av en ekstra side kalt "Bla gjennom filmer" i vår applikasjon (se nav). Formålet med denne siden er å gi brukerne muligheten til å utforske og bla gjennom et bredt utvalg av filmer gjennom vårt API. Vi ønsker å tilby brukerne en mer omfattende opplevelse ved å la dem utforske og oppdage filmer som ikke nødvendigvis er lagret i vår Sanity-database.

Denne tilleggsfunksjonen gir brukerne friheten til å utforske og oppdage et bredere spekter av filmer, slik at de kan finne noe som passer deres personlige preferanser og interesser. Vi tror dette vil forbedre brukeropplevelsen og gjøre applikasjonen enda mer engasjerende og underholdende for våre brukere.

Vi ville sørge for at det ikke kunne finnes duplikater av brukere i systemet. Dermed tenkte vi å bruke e-post og passord for innlogging. Vi fant en løsning av å bruke brukeren sin e-post uten spesialtegn som Id, fordi Sanity behøvde en egen Id, hvis vi skulle undersøke at brukeren var unik. Men på grunn av forskjellige komplikasjoner gikk vi tilbake til å bruke brukernavnet i stedet for e-posten for å ha et solid system som virket.

Det har dessuten vært en del utfordringer med API-et. Det har vært vanskelig å hente data med det. Søket vårt er hentet gjennom API-et, og dersom det har oppstått problemer med brukersøket så reflekterer det API-et og ikke nødvendigvis vårt system.

SASS – Utfordringer:



Figur 1: Utfordring med å få alle filmkortene like, på grunn av E.T. sin størrelse. (Desktop)



Figur 2: På grunn av knappene som ble lagt til i filmkortet, ble også placeholder + tittel mindre. (Mobil)

Kilder:

https://www.flaticon.com/free-icon/movie_2503508

<https://www.sanity.io/docs/query-cheat-sheet>

<https://developer.mozilla.org/en-US/docs/Web/JavaScript>

<https://stackoverflow.com/>

<https://react-icons.github.io/react-icons/>

<https://www.npmjs.com/>

<https://rapidapi.com/SAdrian/api/moviesdatabase/>

<https://coolors.co/>

<https://stackoverflow.com/a/52807512>

<https://css-tricks.com/quick-css-trick-how-to-center-an-object-exactly-in-the-center/>

<https://getbootstrap.com/docs/4.0/components/modal/>

<https://placeholder.co/>

[https://stackoverflow.com/questions/71670582/how-to-delete-a-document-attribute-](https://stackoverflow.com/questions/71670582/how-to-delete-a-document-attribute-in-sanity-io)

[in-sanity-io](https://stackoverflow.com/questions/71670582/how-to-delete-a-document-attribute-in-sanity-io) <https://www.sanity.io/docs/http-patches>

<https://www.tutorialspoint.com/filter-array-with-filter-and-includes-in-javascript>

[https://www.sanity.io/docs/js-client#client create -is-removed-replace-with-one-of-clientcreate-clientcreateifnotexists-or-clientcreateorreplace](https://www.sanity.io/docs/js-client#client_create_is_removed-replace-with-one-of-clientcreate-clientcreateifnotexists-or-clientcreateorreplace)

<https://www.sanity.io/answers/filtering-by-array-references-in-sanity-io>

[https://developer.mozilla.org/en-](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Array/findIndex)

[US/docs/Web/JavaScript/Reference/Global_Objects/Array/findIndex](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Array/findIndex)

<https://game-icons.net/1x1/delapouite/3d-glasses.html>

Kilder – Chat GPT:

"How to insert into an array of reference in sanity"

- Brukt til å prøve finne setIfMissing. Dersom det var en tom array for user wish/fav gikk det ikke an å pushe noe nytt, kun når de hadde et element eller fler. SetIfMissing hjelper å pushe om det ikke finnes et element fra før av.

Example: Adding a Reference Programmatically

```
javascript Skopiu kjode

import sanityClient from '@sanity/client';

const client = sanityClient({
  projectId: 'your-project-id', // replace with your project ID
  dataset: 'your-dataset', // replace with your dataset name
  useCdn: true, // use CDN for faster response, set to false for latest data
  apiVersion: '2023-05-19', // use a UTC date string
  token: 'your-auth-token', // use an authentication token with write access
});

async function addAuthorToArticle(articleId, authorId) {
  try {
    const patch = await client.patch(articleId).setIfMissing({ authors: [] }).append('authors', authorId);
    console.log('Added author to article:', patch);
  } catch (err) {
    console.error('Error adding author to article:', err);
  }
}

// Replace with actual document IDs
const articleId = 'article-document-id';
const authorId = 'author-document-id';

addAuthorToArticle(articleId, authorId);
```