

What to see

Utvikling av interaktive nettsteder 2024 – eksamensoppgave A

Gruppe 8 — Andreas Berg, Mathilde Olesen, Michael Karlsrud, Tuva Marie Andersen

Github-repository

https://github.com/Noespennende/UIN24_eksamensoppgaveA_whattosee

Vanskelighetsgrad

Vi har gått for karakter A.

Github-alias

	Alias	Student
	noespennende	Andreas Berg
	mathildeolesen	Mathilde Olesen
	M1kke95	Michael Karlsrud
	Tuva03	Tuva Marie Andersen

Forutsetninger fra oppgaveteksten

Under 5.6 A-krav 2 står det

“Seksjonen skal vise filmene som matcher kriteriene, men utelukke filmer som ligger i begge brukeres ønskelister eller begge brukeres favorittfilmer (dette dekkes allerede av krav under B-krav og C-krav). Visningen skal markere hvilken bruker og hvilken liste denne filmen tilhører. Det gis ingen bestemte krav til utseende på denne visningen.”

Med tanke på ordlyden her: “men utelukke filmer som ligger i begge brukeres ønskelister eller begge brukeres favorittfilmer”, tolket vi det slik at ingen film skal vises

både i “Catch Up” (5.5.3) / “Go Safe” (5.5.4) og denne nye seksjonen (dobbelt opp).
Altså at hvis to brukere har samme film i favorittfilmene sine eller de to brukerne har samme film i ønskelisten sin, skal disse filmene ikke vises i “Ønskeliste vs Favoritter” seksjonen. Vår implementasjon tar derfor høyde for at en bruker kan ha samme film i sin ønskeliste og favorittliste, og at dette derfor må håndteres.

(Merk: Det er derimot da riktig at en film vises dobbelt opp mellom Catch Up og Go Safe)

```
// User 1 sin kolleksjon av "mitt ønske og andres favoritt"
if (user1Wish.wishlist && user2Fav.favoriteMovies) {
  for(const user1MovieWish of user1Wish.wishlist){
    for(const user2movie of user2Fav.favoriteMovies){
      if(user1MovieWish.imdbid === user2movie.imdbid){
        // Vi har funnet en film som user1 har i sin ønskeliste, og user2 har i sin favorittliste

        // Har user1 filmen i sin egen favorittliste også? hvis ikke...
        if(!user1Fav.favoriteMovies.some(movie => movie.imdbid === user1MovieWish.imdbid)) {
          // Har user2 filmen i sin ønskeliste også? Hvis ikke ...
          if(!user2Wish.wishlist.some(movie => movie.imdbid === user1MovieWish.imdbid)) {

            // Legger til filmen i user1 sin "min ønskeliste som er i andre bruker sin favorittliste"-liste
            user1WishMovieIDs.push(user1MovieWish)
            break
          }
        }
      }
    }
  }
}

// User 2 sin kolleksjon av "mitt ønske og andres favoritt"
if (user2Wish.wishlist && user1Fav.favoriteMovies) {
  for(const user2MovieWish of user2Wish?.wishlist){
    for(const user1MovieFav of user1Fav?.favoriteMovies){
      if(user2MovieWish.imdbid === user1MovieFav.imdbid){
        // Vi har funnet en film som user2 har i sin ønskeliste, og user1 har i sin favorittliste

        // Har user2 filmen i sin egen favorittliste også? hvis ikke...
        if(!user2Fav.favoriteMovies.some(movie => movie.imdbid === user2MovieWish.imdbid)) {
          // Har user1 filmen i sin ønskeliste også? Hvis ikke ...
          if(!user1Wish.wishlist.some(movie => movie.imdbid === user2MovieWish.imdbid)) {
            // Legger til filmen i user2 sin "min ønskeliste som er i andre bruker sin favorittliste"-liste
            user2WishMovieIDs.push(user2MovieWish)
            break
          }
        }
      }
    }
  }
}
```

Først sjekkes bruker 1 sin ønskeliste mot bruker 2 sin favorittliste.

Er filmen felles i disse listene, går sjekken videre til if-løkkene.

Her sjekkes det da om filmen er i bruker 1 sin egen favorittliste, hvis den er det, blir filmen ikke med.

Det sjekkes så videre om filmen er i bruker 2 sin ønskeliste, og hvis den er det, blir filmen heller ikke med.

Det vil si at filmen kun kan finnes i én brukers ønskeliste, og én brukers favorittliste.

Sånn som vi tolker 5.6 A-krav 2 så er det disse to sjekkene som vist i koden ovenfor som kravet spør etter. Vi anser derfor kravet som oppfylt.

Kilder

Massimo, A. (2024, 15. april) *MoviesDatabase API Documentation*. RapidAPI.

Hentet 29. april 2024 fra <https://rapidapi.com/SAdrian/api/moviesdatabase/details>

Sanity (2024, 13. februar). *Query Cheat Sheet – GROQ*.

Hentet 29. april 2024 fra <https://www.sanity.io/docs/query-cheat-sheet>

Sanity (2024, 13. februar). *Schema*.

Hentet 29. april 2024 fra <https://www.sanity.io/docs/schema-types>

Pennell, RD. /Sanity (2022, 14. juli). *Adding reference arrays programmatically in Sanity.io*.

Hentet 29. april 2024 fra <https://www.sanity.io/answers/adding-reference-arrays-programmatically-in-sanity-io>

Mozilla (2023, 27. november): *Array.prototype.some()*.

Hentet 3. mai 2024 fra https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Array/some

Olawanle, J. (2022, 22. februar). *How to Use localStorage with React Hooks to Set and Get Items*. FreeCodeCamp.

Hentet 29. april 2024 fra <https://www.freecodecamp.org/news/how-to-use-localstorage-with-react-hooks-to-set-and-get-items/>

W3schools. (u.å.) *How TO – Responsive Images*.

Hentet 3. mai 2024 fra https://www.w3schools.com/howto/howto_css_image_responsive.asp

Bilder på nettstedet som ikke hentes fra IMDB APllet:

Freepik. (u.å.). *Popcorn free icon* [Ikon]. Flaticon.

Hentet 3. Mai 2024 fra https://www.flaticon.com/free-icon/popcorn_3418886?term=popcorn&page=1&position=3&origin=tag&related_id=3418886

Shaharea. (u.å.). *Photo free icon* [Ikon]. Flaticon.

Hentet 5. Mai 2024 fra https://www.flaticon.com/free-icon/photo_15402170?related_id=15402170

Pavel Danilyuk. (2021, 23 mars). *Friends Watching a Movie* [Bilde]. Flaticon.

Hentet 5. Mai 2024 fra <https://www.pexels.com/photo/friends-watching-a-movie-7234387/>