

# Arbeids struktur

## What to See

### Vårt mål er karakter A

Vi har grundig gjennomgått oppgaven og laget en liste med punkter for det som skal gjøres. Arbeidet startet med å opprette en React-app, inkludert Sanity og SCSS for styling.

#### 1. Hovedstruktur

- Vi begynte med branchen **main**, hvor vi bygde hovedstrukturen til appen vår.

#### 2. Deling av Arbeid

- Etter å ha etablert grunnlaget i **main**, opprettet vi to nye brancher:
  - **Dashbord**
  - **Data-From-API-To-Sanity**

#### 3. Branchene og Oppgavene

- I branchene **Dashbord** og **Data-From-API-To-Sanity** delte vi opp Sanity-relaterte oppgaver og fortsatte utviklingen. Dette sikret at vi kunne jobbe parallelt og effektivt med ulike aspekter av prosjektet.

## **Arbeidsmetode**

### **1. Oppdeling av Oppgaver**

- Vi delte opp oppgavene mellom oss to, slik at hver person kunne fokusere på spesifikke deler av prosjektet. Denne strategien sørget for effektiv arbeidsflyt og tydelig ansvarfordeling.

### **2. Individuelt Arbeid og Samarbeid**

- Hver av oss jobbet selvstendig med våre tildelte oppgaver. Ved vanskeligheter arrangerte vi møter hvor vi samarbeidet om å finne løsninger. Disse møtene var essensielle for å overkomme utfordringer og sikre fremgang i prosjektet.

### **3. Commit Strategi**

- Commitene for løsningene funnet under samarbeidsmøter er merket med («Live Coding») i GitHub. Dette hjelper med å spore hvilke løsninger som ble utviklet gjennom samarbeid. Generelt sett, commiter merket med («Live Coding») indikerer at vi kom frem til en løsning etter en felles innsats. Dette sikrer at arbeidet blir anerkjent for begge teammedlemmene.

## Hjelpemidler

### Sanity Dokumentasjon

Vi brukte Sanity-dokumentasjonen flittig for å forstå og implementere funksjonaliteten til Sanity. Spesielt nyttig var "Query Cheat Sheet - GROQ", hvor vi hovedsakelig brukte (object projection) for å hente data fra Sanity, som for eksempel filmer fra favorittlisten og sjangre vi har til felles.

### Dokumentasjon fra Forelesninger og Kursets GitHub

Dokumentasjonen fra forelesningene og materialet på kursets GitHub-side var uvurderlige referanser. Vi brukte dette materialet til å finjustere og forstå viktige løsninger og implementeringsmetoder.

### ChatGPT

Vi brukte ChatGPT som et verktøy for å sjekke syntaksen på forskjellige JavaScript-koder og funksjoner, samt Sanity-spøringer når vi møtte utfordringer. ChatGPT var nyttig for å få raske svar og eksempler på hvordan vi kunne løse spesifikke problemer.

### Tidligere Erfaring med `.some`-funksjonen

Når det gjelder bruken av `.some`-funksjonen, hadde en av oss tidligere erfaring med denne funksjonen fra oblig 4. Dette gjorde det lettere å implementere funksjonen effektivt i prosjektet vårt.

Fra chatgpt

```
{
  "Khalil": *[_type == "users" && username == "Khali"]
  "favoredGenres": favoredGenres


},{
  "Odai": *[_type == "users" && username == "Odai"]
  "favoredGenres": favoredGenres
}
```

er denne santiy queryen riktig?



Nei, det ser ut til at det er noen syntaksfeil i din Sanity query. Her er hvordan du kan korrigere den:

javascript

 Kopier kode

```
{
  "Khalil": *[_type == "users" && username == "Khalil"]{
    "favoredGenres": favoredGenres
  },
  "Odai": *[_type == "users" && username == "Odai"]{
    "favoredGenres": favoredGenres
  }
}
```