

Kult! Her er litt info om teknologiene vi bruker idag som forberedelse til møtet.

Har dere noen preferanser for teknologier eller rammeverk på backend, frontend og databaser?

Ja, vi har allerede en kjørende energioppfølgingsapplikasjon (EOS) og en mikro-tjeneste backend infrastruktur.

All kommunikasjon mellom klient og backend går gjennom et [GraphQL API](#).

Klientsiden er bygget med microfrontends, i React med [Shadcn](#) for ui, plot med [HighCharts](#), og diverse [TanStack](#) biblioteker som tanstack query, tanstack table og tanstack router. Vi vurderer sterkt å bruke [TanStack AI](#) som chat interface når det blir litt mer modent.

Backend er bygget opp av mikro-tjenester i [Kubernetes](#), med [NATS](#) som intern meldingsbuss.

Vi har en felles [Falkor](#) Graf-database, hvor all data er semantisk beskrevet med hjelp av ontologier som [Brick](#), REC og Energy (som vi skal publisere snart).

På frontend vil dere sannsynligvis jobbe inn mot eksisterende Repo og følger oppsettet som er der.

På backend vil dere få et eget repo for mikro-tjenesten(e), og står dermed litt mer fritt til å velge teknologi som trengs for å løse oppgaven. Primært er backend skrevet i Go lang og Python, alt etter hva som passer best for oppgaven.

Finnes det eksisterende interne API-er, datasett eller systemer vi bør bygge videre på?

Ja, vi har et felles GraphQL API som alle Klienter bruker. Vi håndterer utvidelser om det trengs, og federation fra GraphQL til riktig mikro-tjeneste.

Dere bør sette dere inn i hvordan man bruker GraphQL for å hente data, men trenger ikke tenke på å implementere utvidelser hvis det trengs. Vi bruker Apollo client som frontend på API'et ved testing.

Her er en kort og god forklaring på [semantikk](#) og [ontologi](#)

Og Brick ontologi docs/schema: <https://ontology.brickschema.org/>

Dere trenger ikke gå i dybden på dette, men det er fint med en prinsipiell forståelse av konseptet. Vi har som sagt strukturert all data, som gjør at man kan gjøre GraphQL kall som: "Gi meg en liste over alle 'brick:Building' og returner 'id' og 'name'".

Eller "For hver 'brick:Building', hent 'brick:Building_Electrical_Meter' og hent tidsserie på forbruk for periode '2025' med oppløsning '1w'.

Har dere bestemt dere for hvordan fokus dere ønsker?

Backend/ML-fokus | Frontend-fokus | Helhetlig/Mix-fokus