

# Bunden forudsætning - Node.js og Express

## FURPS:

- **Functionality:** Applikationen skal omhandle kunstnere og skal være todelt. En front-end i form af en webside og en back-end i form af en nodejs database og REST API. Applikationen skal kunne oprette, redigere, læse og slette data fra en JSON fil i back-enden. Det skal være muligt for brugeren at tilgå og lave CRUD på dataen i JSON filen fra front-end websiden. Brugeren skal vises data'en læseligt og overskueligt på websiden. Det skal være muligt for brugeren at sortere og filtrere i den viste data efter forskellige parametre, de selv vælger gennem brugerfladen. Brugeren skal også kunne tilføje og fjerne kunstnere fra deres favoritter.
- **Usability:** Brugergrænsefladen skal være brugervenlig og nem at navigere. Alle CRUD-operationer skal være let tilgængelige, og filtrerings/sorteringsfunktionerne skal være intuitive. Derfor placeres søgs-, filtrerings- og sorteringsfunktionalitet i toppen af siden hvor brugeren altid nemt kan tilgå dem og hvor de ofte findes på andre websider og applikationer. Dataen vil blive vist i form af et css grid hvori hver kunstner vil blive vist med relevante data og knapper til at slette, redigere og favorisere. Brugeren skal gives feedback på hvilke handlinger de udfører fx når der klikkes slet åbnes et dialog vindue hvori brugeren skal bekræfte at de vil slette denne kunstner. Ligeledes vil det tydeligt vises, hvilke kunstnere der er blandt deres favoritter. Brugeren bliver automatisk scrollet til den relevante del af siden, når de starter en handling. Fx når de trykker opdater, vil de blive taget til formularen for opdatering af dataen, med den nuværende version af dataen allerede udfyldt, i formularen, så brugeren kun behøver tænke på at rette den data de har brug for, uden at skulle udfylde hele formularen hver gang.
- **Reliability:** Applikationen skal være stabil og pålidelig. Dataintegritet og korrekt håndtering af CRUD-operationer er afgørende. Derfor
- **Performance:** Applikationen skal have en acceptabel ydeevne, herunder hurtig datahentning og responsivitet i brugergrænsefladen.
- **Supportability:** Koden skal være nem at sætte sig ind i med tilstrækkelige kommentarer. Koden skal kunne bygges videre på og udvides. Der skal være instrukser for hvordan programmet installeres og køres i form af en readme.md fil.
- **+**: Der skal også tages hensyn til andre relevante krav og overvejelser for at sikre en vellykket udvikling og implementering af applikationen. Ud over FURPS-kravene skal udviklingen også overholde generelle principper for kodestruktur, modularitet og afhængighedsstyring, herunder Separation of Concerns, Loose Coupling og High Cohesion.