

Furps+

Functionality:

- Backend er implementere et REST API ved hjælp af Node.js og Express.js.
- Backend har ruter med endpoints for HTTP-metoderne GET, POST, PUT/PATCH, og DELETE for at muliggøre CRUD-operationer.
- Backend kan udføre CRUD-operationer ved at læse og skrive data til en JSON-fil.
- Backend kan hente alle objekter fra JSON-filen.
- Backend kan hente et objekt baseret på et specificeret id fra JSON-filen.
- Frontends UI (User Interface) er implementeret med HTML, CSS og JavaScript.
- Brugere skal kunne oprette, læse, opdatere og slette data (CRUD-operationer) via frontend.
- Brugere har mulighed for at filtrere og sortere kunstnere baseret på udvalgte properties.
- Brugere har mulighed for at markere en kunstner som favorit.
- Brugere kan se en liste over alle favoritkunstnere.
- Alle favoritter gemmes i en JSON fil, med et sessions ID, som bliver gemt på brugerens PC vha. cookies.

Usability:

- Brugergrænsefladen er nem at navigere rundt i, grundet dets simple design.
- CSS Grid og relaterede HTML-elementer er anvendt for at sikre en visuel appel og en god brugeroplevelse.

Reliability:

- Backend skal være pålidelig og kunne håndtere CRUD-operationer korrekt.
- Dataintegriteten i JSON-filen, bliver opretholdt, vha. at sikre korrekt input fra forms mm.

Performance:

- Backend og frontend er responsiv og har en acceptabel ydeevne.
- Frontend skal kunne håndtere filtrering og sortering effektivt
- Frontenden understøtter filtrering og sortering effektivt

Supportability:

- Koden er opdelt i moduler for at lette vedligeholdelse og udvidelse.
- Det er nemt data i JSON-filen eller tilføje nye kunstnere.
- Der er tilstrækkelig dokumentation for både backend og frontend, hvor det er nødvendigt.

+:

- Listen af favoritkunstnere bliver gemt i backenden i en separat JSON-fil. Cookies anvendes til at identificere brugeren i JSON-filen og finde de specifikke kunstnere, som den aktuelle bruger har markeret som favoritter.