

# FURPS – Buster

Krav som ikke nødvendigvis fremgår af opgavebeskrivelsen:

- **Functionality:**
  - Appen skal kunne håndtere at sortere og filtrere på én gang.
  - Der skal være nogle checks undervejs, som helst umuliggør duplikering af musikere i databasen
  - Der skal være en bekræftelse, før en kunstner bliver slettet
- **Usability:**
  - Det skal ikke kræve nærlæsning af hele siden fra brugerens side for at finde ud af, hvor de skal klikke
  - Knapperne skal have en vis størrelse og farve, så de falder i øjnene.
  - Hjemmesiden skal desuden have en vis mængde styling og de vigtigste informationer vises først.
  - Derudover skal det være tydeligt, hvilke kunstnere der er sat som favoritter.
  - Det skal ligeledes være let både at fjerne og tilføje kunstnere til favoritlisten.
  - Tingene skal stå "normalt", det vil sige ingen "undefined" eller "true"/"false", hvis en property ikke har nogen værdi/har en boolean-værdi.
  - Tingene skal i det hele taget stå "kort" – selvom det også skal være "intuitivt", hvor på skærmen der skal klikkes, for at udføre bestemte operationer, skal tekst generelt også skrives på så få ord som muligt. Dette hjælper med overblikket
- **Reliability:**
  - Der skal være testet for oplagte bugs, fx skal man kunne oprette en bruger og opdatere den samme bruger lige bagefter.
  - Man skal derudover kunne forvente at de samme knapper opfører sig ens på tværs af hele siden – der skal ikke være to "update"-knapper, som gør noget forskelligt.
  - Ligeledes skal alle filtreringsmulighederne filtrerer "den samme vej", dvs. der skal ikke være nogle, som kun viser kunstnere, som rent faktisk opfylder den valgte filtreringsmulighed, og andre som viser kunstnere, der *IKKE* opfylder det valgte filtreringskrav.
- **Performance:**
  - Der skal ikke udføres flere fetch-kald end allerhøjst nødvendigt – det er svært at gøre meget ved hastigheden ved hver enkelt kald, men ved at minimere antallet af kald, bl.a. ved at sørge for at gemme den hentede data i variabler, kan man mindske den samlede ventetid/belastning på serveren
  - Derudover skal der ikke hentes mere data end nødvendigt per fetch-kald
  - ... og backend'en skal ikke udføre en masse unødige beregninger, som sænker hastigheden for tilbageleveringen af den relevante data
- **Supportability:**
  - Der skal skrives kommentarer, hvor nødvendigt

- Koden skal følge principperne for separation of concerns – den skal inddeles i små funktioner og flere forskellige moduler