

Programmering og udvikling af små systemer samt databaser

EKSAMENSOPGAVE EFTERÅR 2022

Forord

Denne opgavebeskrivelse er udarbejdet af Nicolai Jacobsen, Regitze Sdun og Stefan Reinholdt Jørgensen til brug ved eksamen i faget Programmering og udvikling af små systemer for efteråret 2022.

Eksamensprojektet skal udarbejdes individuelt.

I må gerne diskutere opgaven sammen med andre. Husk dog, at såfremt I benytter kode fra tutorials, eksterne biblioteker eller lignende skal dette angives med en kildehenvisning.

I skal igennem arbejdet med jeres eksamensprojekt vise de kompetencer frem, som I har opnået igennem 1. semester.

Eksamensopgaven tester jeres evner i HTML, CSS, JavaScript, DOM manipulation, API kald og Browser API'er.

I vil kunne finde inspiration og løsninger til store dele af jeres eksamensprojekt igennem de uge- og godkendelsesopgaver, som I allerede har udarbejdet. Det er derfor vigtigt, at alle de omtalte opgaver er lavet, inden du begynder på denne eksamensopgave.

Spørgsmål

Spørgsmål omkring formalia til opgaven bedes lagt på Canvas, så alle kan få glæde af svaret.

Opgaven skal til enhver tid overholde CBS' formalia. Såfremt spørgsmålet omhandler personlige forhold, som for eksempel sygdomsperioder, skal disse sendes til studieadministrationen.

Opgavebeskrivelse

I har opdaget at Regitze, Stefan & Nicolai (RSN) er trætte af kvaliteten af online nyheder samt sidde i timevis og trawle internettet tyndt for at blive opdateret. I har derfor kommet med ideen at starte jeres egen nyheds aggregator hjemmeside, så RSN kan blive glade og opdateret på dagens relevante nyheder.

I skal derfor designe en web-applikation, som understøtter nok funktionalitet, så I kan sælge ideen til RSN. Jeres mentor har informeret jer om, at det er tilstrækkeligt at gemme alt information som flade filer på klientens maskine, så I behøver ikke at bekymre jer om databaser for nu.

Ligesom på andre nyhedssider skal det være muligt at oprette en profil med tilhørende metadata. Endvidere er det et krav at gemme hvilke nyhedsartikler en bruger har læst. I ved også at Stefan elsker at få opdateret information om vejret, og han aldrig vil acceptere en nyhedsside uden en vejr boks, så sådan en skal der også være. Vi foreslår, at I Googler efter et vejr API.

I har fået en designer til at komme med et bud på hvordan jeres app skal se ud. Dette har i vist til jeres investorerne, som på denne baggrund glæder sig til at se jeres prototype, og forventer ikke de store forskelle fra design forslaget.

Til at starte med henter i alle nyhedsartiklerne eksternt:
<https://newsapi.org/docs/get-started>

I skal derfor udvikle app'en som en client-facing front-end applikation. App'en skal have en frontend, som brugerne kan navigerer igennem. Det betyder, at I skal udvikle to dele:

- Klienten (JS, HTML, CSS)
- Storage (En data løsning som kan gemme bruger information og nyhedsartikler som JSON direkte på klient disken)

Alle kravene for de enkelte dele fremgår under tekniske krav til besvarelsen.

Funktionelle krav til besvarelsen

1. Login:

- a. App'en skal tillade brugeren at logge ind
- b. App'en skal gøre det muligt for en bruger at logge ud

2. Profil:

- a. App'en skal tillade en bruger at oprette en profil
- b. App'en skal tillade en bruger at slette sin egen profil
- c. App'en skal tillade en bruger at opdatere sin egen profil
- d. App'en skal tillade at en bruger kan sætte sine yndlings nyhedskategorier.
- e. App'en skal tillade at en bruger kan tilføje en eller flere nyhedsartikler til favoritter.
- f. App'en skal tracke hvilke nyhedsartikler en bruger har læst. Dette gemmes ved hjælp af [localstorage](#).

3. Nyheder:

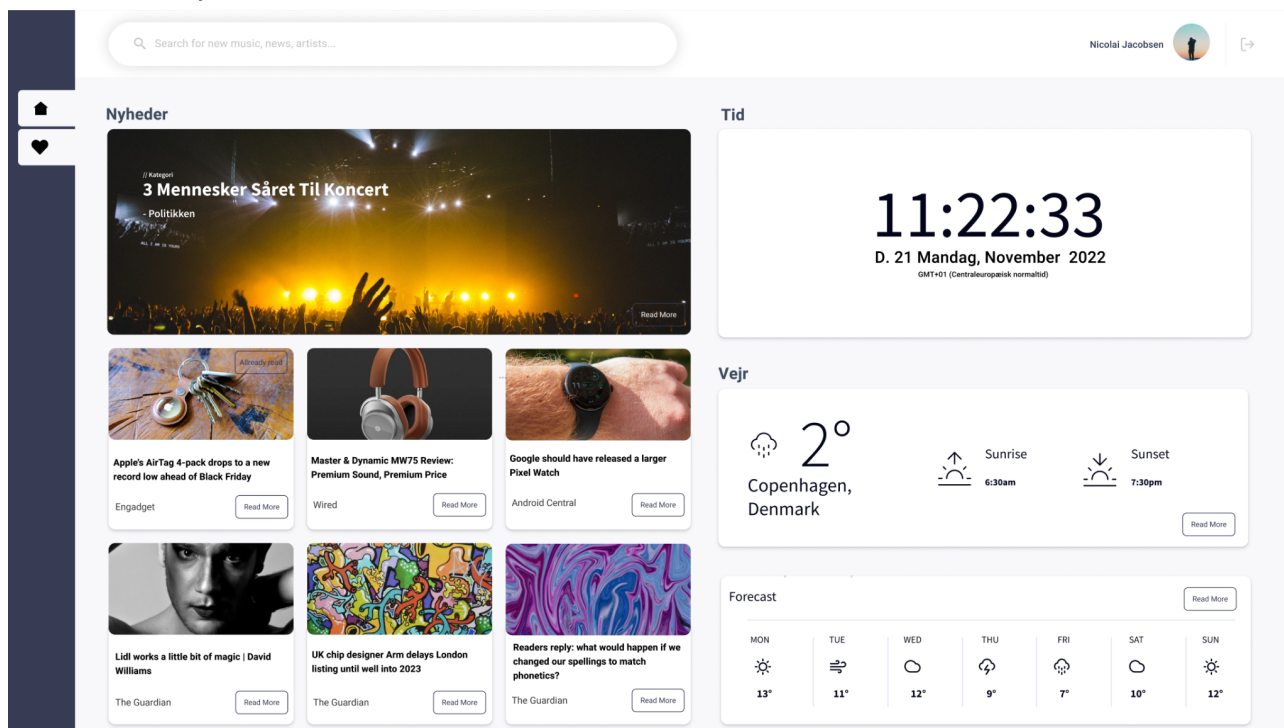
- a. App'en skal vise en liste af de seneste nyheder til brugeren.
- b. App'en skal kunne vise en enkelt nyhed med links og billede.
- c. App'en skal gøre det muligt for en bruger at søge på mindst to forskellige parametre for nyheder. F.eks. Dato, tid, sprog, land, osv... (få inspiration [her](#)).

4. Vejret:

- a. App'en skal vise den nuværende tid og dato.
- b. App'en skal hente og vise vejrudsigten for de næste 7 dage.
- c. App'en skal vise den nuværende temperatur.
- d. App'en skal vise solopgang og solnedgangs tidspunkt.
- e. App'en skal bruge vejrdato for lokationen københavn.

Krav til web applikationens udseende

- Der er til opgaven givet et designforslag ved følgende mockup:



Det forventes at det endelige produkt ligner design forslaget nogenlunde. Da design forslaget kun er lavet på hovedsiden af appen er der ingen krav til de andre sider.

Formelle krav til besvarelsen

1. Rapporten skal afleveres som PDF via Digital Eksamen
2. Alle sider i rapporten – inklusiv forsiden – skal være fortløbende nummereret
3. Antallet af sider skal fremgå af forsiden
4. Rapporten for Programmering og udv. af små systemer samt databaser må højst fylde 30 normalsider. Alt, hvad der ligger ud over disse sider, vil ikke blive taget i betragtning ved bedømmelsen. Se regler for optælling af sider på my.cbs.dk. Er den kortere er det også fint, I bliver bedømt på indhold, ikke længde. En fyldestgørende rapport kan derfor godt skrives på 10 sider.
5. Antallet af normalsider dokumenteres ved f.eks. udskrift af rapport fra tekstbehandlingsprogrammets statistik-funktion
6. Figurer i rapporten skal være tydelige og læselige
7. Rapporten skal være gennemlæst og ikke indeholde stavefejl
8. Jeres kildekode skal vedhæftes afleveringen som en zip-fil, som også indeholder eventuelle andre relevante bilag.

Tekniske krav til besvarelsen

I skal udarbejde to dele til jeres system.

1. Klienten (JS, HTML, CSS)
2. Storage (flade filer på disk (JSON))

1 - Klienten

Jeres klient laves i helt simpelt Javascript, HTML og CSS. Vi forventer at I kommentere koden og beskriver hvordan forskellige stykker af koden løser diverse krav.

2 - Storage

Storage håndtere hvordan data bliver gemt og læst fra disk.

For at opfylde kravene til projektet er det nødvendigt at gemme noget data. Dette data forventes gemt på serveren i et format der er let læseligt (JSON).

De valg I gør jer for storage skal dokumenteres i jeres rapport. Dette omfatter f.eks. hvordan i har valgt at organisere og strukturere jeres data. Hertil kan i vedlægge hvordan en entry i jeres JSON fil ser ud. F.eks.:

User

```
1. {  
2.   "employee": {  
3.     "name":      "Nicolai",  
4.     "salary":    1337,  
5.     "married":   false  
6.   }  
7. }
```

3 - Eksekvering

I skal kun dokumentere det teori, som vi har gennemgået i pensum. Teori uden for pensum belønnes ikke og kan i værste fald trække ned, fordi man ikke får dækket de andre vigtige områder. Man behøver ikke at skrive 30 sider, hvis man kan forklare sig præcist på færre sider.

Materialer og gode råd

Som baggrund for at gennemføre opgaven anvendes denne opgavebeskrivelse samt fagets pensum. I kan derudover søge oplysninger i andre offentligt tilgængelige kilder (husk at angive kilde med APA-modellen).

Der opfordres til, at I benytter forløbets forskellige muligheder for at diskutere projektet og de problemstillinger I møder undervejs. Dertil opfordres I til at sparre med hinanden og tilbyde reviews af hinandens løsninger. Ofte ender man med at se sig blind i det man selv laver, hvorfor det kan være nyttigt med et sæt friske øjne.

I er velkommen til at poste tekniske problemer på Canvas, hvor vi gerne vil hjælpe. Når I spørger om hjælp er det krav at I som minimum beskriver hvad I allerede har forsøgt for at løse problemet. Vi forventer at I også hjælper hinanden og byder ind i forhold til hinandens problemer.

Det er vigtigt, at I holder jer for øje, hvad dette fag fokuserer på og dermed undlader elementer, som for dette fag ikke er relevante.

I skal lægge vægt på refleksion og diskussion, hvilket bør vægte højere end beskrivelse. Såfremt I benytter teori er det derfor vigtigt, at I ikke blot beskriver teorien, men reflekterer over brug af teorien og dermed konkretiserer teorien til lige netop jeres projekt.

Rapporten

Nedenstående er minimumskravene listet til de enkelte del-leverancer:

Kravspecifikationen er det endelige produkt. I afsnittet om tekniske krav, skal I lave en tabeloversigt over hvilke krav, der er opfyldt og hvilke der ikke er. Herefter skal I gå hvert krav systematisk igennem, én for én og beskrive hvordan de er opfyldt brug gerne visualisationer fra frontenden. Hertil opfordres i til at angive løsningsovervejelser for de tekniske krav I ikke har opfyldt.

Som bilag til rapporten skal I vedlægge en kort video af jeres applikation, som viser at alle krav I skriver er opfyldt rent faktisk er opfyldt. Dette kan gøres ved at klikke igennem jeres applikation og vise funktionaliteten. Brug f.eks. [Loom](#). Sørg for ikke at uploade videoen, men bare linke til den.

Data oversigt viser hvordan i har valgt at organisere og strukturere jeres data. Det skal give et hurtigt overblik over, hvordan I har valgt at gemme jeres data. Det forventes derfor, at I vedlægger et eksempel på hvordan jeres JSON objekter der holder de forskellige informationer er gemt.

Løsningsovervejelser skal være en tungtvejende del af jeres opgave. Vi forventer at høre omkring de overvejelser I har gjort jer i forhold til strukturen for jeres respektive applikationer, samt hvilke begrænsninger som eventuelt måtte være.

Procesevalueringen har til formål, at I forklarer jeres læring i forløbet. I dette afsnit kan I for eksempel beskrive 3-5 hændelser, der er opstået undervejs i jeres projektforsløb og hvordan I valgte at løse disse hændelser metodisk. Dette behøves ikke være ting som var succesfulde, der er typisk lettere at fokusere på de fejl man har lavet, og perspektivere omkring dette.

*Rapporten skal først og fremmest beskrive løsningen, dvs. produktet og de dertilhørende designvalg, men ikke processen af at nå frem til produktet **bortset fra i procesevalueringens afsnittet**, som netop omhandler dette.*

Eksamen i Programmering og udv. af små systemer samt databaser

Eksamen udføres af fagets underviser med intern censur, og bedømmes efter 7-trins skalaen. Karakteren gives efter en samlet bedømmelse af rapporten og jeres kodebase

Bedømmelsen af rapporten vil lægge vægt på følgende:

- Disposition og rapport-systematik
- Jeres kodebase og konstruktionen heraf
- Jeres forståelse for tekniske og forretningsmæssige begrænsninger af jeres implementering, samt refleksion over de valg I har truffet heraf
- Beskrivelse af jeres forudsætninger, valg af perspektiv(er) og redegørelse for jeres valg
- Begreber og definitioner (deres præcision, konsistens og relevans)
- Teori- og metodeanvendelse
- Tabeller og figurer (deres informationsværdi, konsistens og relevans)
- Anvendt litteratur (omfang og dybde)
- Kildekritik (både teori og data)
- Forklaringer: redegørelse for årsag og virkning
- Procesevalueringens kobling til pensum
- Bilag (deres kvalitet og relevans)
- Rapportens omfang (den skal holde de formelle retningslinjer)
- Sprog, kildehenvisninger og struktur
- Evt. afskrift / direkte reproduktion uden kildehenvisninger

Der gives karakter efter fagets læringsmål, som kan ses i [kursuskataloget](#).