

Rapport - Eksamensopgave 2022

Programmering og udvikling af små systemer samt databaser

Af Gustav Manthorpe – 160306



Rapporten indeholder i alt 14 sider, inklusive forside og bilag, svarende til 17601 anslag

Indholdsfortegnelse

Rapport - Eksamensopgave 2022	1
1. Indledning	3
2. Kravspecifikationer	3
3. Kravspecifikationer detaljeret	4
3.1 Login	5
3.1.1 App'en skal tillade brugeren at logge ind	5
3.1.2 App'en skal gøre det muligt for en bruger at logge ud	5
3.2 Profil	5
3.2.1 App'en skal tillade en bruger at oprette en profil	5
3.2.2 App'en skal tillade en bruger at slette sin egen profil	5
3.2.3 App'en skal tillade en bruger at opdatere sin egen profil	6
3.2.4 App'en skal tillade at en bruger kan sætte sine yndlingsnyhedskategorier.	6
3.2.5 App'en skal tillade at en bruger kan tilføje en eller flere nyhedsartikler til favoritter.	6
3.3 Nyheder	6
3.3.1 App'en skal vise en liste af de seneste nyheder til brugeren.	6
3.3.2 App'en skal kunne vise en enkelt nyhed med links og billede.	7
3.3.3 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....	7
3.4 Vejr	7
3.4.1 App'en skal vise den nuværende tid og dato.	7
3.4.2 App'en skal hente og vise vejrudsigten for de næste 7 dage.	8
3.4.3 App'en skal vise den nuværende temperatur.	8
3.4.4 App'en skal vise solopgang og solnedgangs tidspunkt.	8
3.4.5 App'en skal bruge vejrdato for lokationen København.	8
4. Data oversigt	9
5. Løsningsovervejelser	9
5.1 Nyheder, brug af api og indsættelse	10
5.2 updateFavoriteNews	10
5.3 Søgefunktion	11
5.4 Login	11
5.5 Vejr	12
5.6 Tid	12
6. Proces evaluering	12
7. Litteraturliste	13
8. Bilag	14

1. Indledning

Jeg vil i denne rapport komme ind på, hvordan jeg har løst den givne opgave:

At designe og implementere en webapplikation, der kan vise nyheder.

Jeg vil gennemgå, hvordan jeg har løst opgaven, diverse problemstillinger og hvordan mine overvejelser har været under forløbet. Afslutningsvis følger litteraturlisten og bilag.

2. Kravspecifikationer

Funktionelle krav til besvarelsen:

1. Login:	
a. App'en skal tillade brugeren at logge ind	Opfylder
b. App'en skal gøre det muligt for en bruger at logge ud	Opfylder
2. Profil:	
a. App'en skal tillade en bruger at oprette en profil	Opfylder
b. App'en skal tillade en bruger at slette sin egen profil	Opfylder
c. App'en skal tillade en bruger at opdatere sin egen profil	Opfylder
d. App'en skal tillade at en bruger kan sætte sine yndlingsnyhedskategorier.	Opfylder
e. App'en skal tillade at en bruger kan tilføje en eller flere nyhedsartikler til favoritter.	Opfylder

F. App'en skal tracke hvilke nyhedsartikler en bruger har læst. Dette gemmes ved hjælp af localstorage.	Opfylder
3. Nyheder:	
a. App'en skal vise en liste af de seneste nyheder til brugeren.	Opfylder
b. App'en skal kunne vise en enkelt nyhed med links og billede.	Opfylder
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....	Opfylder ikke
4. Vejret:	
a. App'en skal vise den nuværende tid og dato.	Opfylder
b. App'en skal hente og vise vejrudsigten for de næste 7 dage.	Opfylder
c. App'en skal vise den nuværende temperatur.	Opfylder
d. App'en skal vise solopgang og solnedgangs tidspunkt.	Opfylder
e. App'en skal bruge vejrdato for lokationen København.	Opfylder

3. Kravspecifikationer detaljeret

Jeg vil nu redegøre for hvor de forskellige krav bliver opfyldt og kan findes. Inden min forklaring omkring hvordan kravene er opfyldt, kommer der til at være opgivet hvor denne kode er skrevet henne. Hvor koden står, er altid i kursiv.

3.1 Login

3.1.1 App'en skal tillade brugeren at logge ind

Koden for login kan findes i register.js

Denne JavaScript-kode opretter et objekt med navnet "users" og gemmer det i "localStorage". Det definerer også to funktioner: "store" og "check".

Funktionen "store" tager værdierne fra en række input-felter og gemmer dem i "users"-objektet. Dette gøres ved at hente værdierne fra de relevante input-felter ved hjælp af deres "id"-attributter og derefter gemme dem i "users"-objektet ved hjælp af brugerens navn som nøgle.^{1,2}

Funktionen "check" tager værdierne fra to andre input-felter og sammenligner dem med de gemte værdier i "users"-objektet for at kontrollere, om de matcher. Hvis de matcher, vil brugeren blive logget ind. Hvis de ikke matcher, vil en fejlmeddelelse blive vist.

3.1.2 App'en skal gøre det muligt for en bruger at logge ud

Frontpage.html linje 37-40

For at logge brugeren ud gøres brug af en simpel HREF knap, hvor at brugeren bliver sendt til register.html.

3.2 Profil

3.2.1 App'en skal tillade en bruger at oprette en profil

register.js

Besvaret i 3.1.1

3.2.2 App'en skal tillade en bruger at slette sin egen profil

editAccount.js linje 5-11

Funktionen deleteAccount() bruger removeItem() til at fjerne de gemte oplysninger om brugerens navn, kodeord og favoritindhold fra localStorage. Derefter vises en meddelelse, der bekræfter, at kontoen er slettet, og brugeren sendes derefter tilbage til forsiden af nyhedssiden.³

1 W3Schools, (2022) Retrieved from:
https://www.w3schools.com/howto/howto_css_login_form.asp

2 W3Schools, (2022) Retrieved from:
https://www.w3schools.com/jsref/prop_win_localstorage.asp

3 W3Schools, (2022) Retrieved from:
https://www.w3schools.com/jsref/met_storage_clear.asp

3.2.3 App'en skal tillade en bruger at opdatere sin egen profil

editAccount.js linje 13-17

Funktionen `updateFavoriteNews()` henter værdien af den nyeste indtastede favoritnyhed og gemmer det herefter i `localStorage`. Derefter vises en meddelelse, der bekræfter at favoritnyheden er blevet tilføjet.⁴

3.2.4 App'en skal tillade at en bruger kan sætte sine yndlingsnyhedskategorier.

register.js linje 4-14

Dette opfyldes igennem funktionen `store()`, hvor brugeren allerede fra oprettelsen af sin profil kan vælge en yndlingsnyhedskategori.

3.2.5 App'en skal tillade at en bruger kan tilføje en eller flere nyhedsartikler til favoritter.

favorites.js

Funktionen `addFavoritex()` gør det muligt for brugeren at markere en nyhed som deres favorit ved at trykke på knappen "favorite", der vises ved siden af nyheden. Når brugeren trykker på knappen, opdateres den til at blive rød og vise teksten "favorited" for at bekræfte, at nyheden er blevet føjet til brugerens favoritter.⁵

3.3 Nyheder

3.3.1 App'en skal vise en liste af de seneste nyheder til brugeren.

newsPictures.js

Når denne kode køres, vil den først oprette en URL, der indeholder en API-nøgle, der giver adgang til WSJ.com's nyhedsfeed. Derefter bruges `fetch()` til at sende en forespørgsel til denne URL for at hente nyhederne. Når der er modtaget et svar fra serveren, vil `.json()`-metoden blive bedt om at konvertere det til et JavaScript-objekt, der kan sættes ind i koden.

Derefter gør koden brug af et loop til at gå gennem hver nyhed i objektet. For hver nyhed, der loopes over, bliver der udprintet noget HTML, der indeholder information

⁴ W3Schools, (2022) Retrieved from:
https://www.w3schools.com/jsref/prop_win_localstorage.asp

⁵ W3Schools, (2022) Retrieved from:
https://www.w3schools.com/js/js_htmlDOM_html.asp

om nyheden, såsom et billede, en titel, hvilken avis der skrev det og en "read more"-knap.⁶⁷

Det er værd at bemærke, at denne kode kun udskriver en nyhed ad gangen og at jeg derfor er endt med kode, der gentager sig selv meget. Dette skyldes, at løkken kun køres en gang med et loop, der starter ved 0 og stopper, når den er større end eller lig med 1. Hvis man vil udskrive flere nyheder på én gang, burde et loop der nogenlunde ligner nedenstående kunne hjælpe mig.⁸

```
for (let i = 0; i < 7; i++) { // Udskriver 7 nyheder // ... }
```

Dog blev dette aldrig løst.

3.3.2 App'en skal kunne vise en enkelt nyhed med links og billede.

newsPictures.js

Opfyldt gennem 3.3.1

3.3.3 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....

newsPictures.js linje 229-237

Dette blev desværre aldrig opfyldt. Jeg har valgt at inkludere mit forsøg i kommentarer som kan ses på linje 229-237

3.4 Vejret

3.4.1 App'en skal vise den nuværende tid og dato.

clockBox.js

Funktionen viser tiden ved at oprette et Date-objekt, der indeholder den aktuelle dato og tid, og derefter hente timerens timer, minutter og sekunder fra dette objekt. Derefter formateres tiden som en streng, og p tagget med id'et "timer" opdateres til den aktuelle tid.⁹

Derefter bruges setInterval() til at opdatere siden en gang i sekundet.¹⁰

⁶ The Coding Train, (28/04-2018) *Youtube.com*, Retrieved From:

https://www.youtube.com/watch?v=QO4NXhWo_NM

⁷ Ha IT, (4/12-2022), *hait.cbss.dk* Retrieved From:

<https://hait.cbss.dk/weeks/uge-46.html>

⁸ Chat Open AI, (9/12-2022) Retrieved from:

<https://chat.openai.com/chat>

⁹ W3Schools, (2022) Retrieved from:

https://www.w3schools.com/php/func_date_getdate.asp

¹⁰ GitHub Copilot (2022) Retrieved from:

<https://github.com/features/copilot>

Datoen oprettes ved brug af Date-objektet, og derefter laves det til en streng ved hjælp af toLocaleDateString()-metoden. Derefter sættes indholdet til p tagget "theDate".^{11,12}

3.4.2 App'en skal hente og vise vejrudsigten for de næste 7 dage.

weather.js

Gennem brug af kode og API lavet af visualcrossing.com har jeg oprettet en vejrudsigt for de næste 7 dage.¹³

3.4.3 App'en skal vise den nuværende temperatur.

weather.js

Se 3.4.2

3.4.4 App'en skal vise solopgang og solnedgangs tidspunkt.

sun.js

Gennem brug af API lavet af visualcrossing.com kan man se solopgang og solnedgangs tidspunkt.^{14,15}

3.4.5 App'en skal bruge vejrdata for lokationen København.

Weather.js og sun.js

Se 3.4.2 og 3.4.4

¹¹ W3Schools, (2022) Retrieved from:

https://www.w3schools.com/js/js_timing.asp

¹² Chat Open AI, (9/12-2022) Retrieved from:

<https://chat.openai.com/chat>

¹³ Visual Crossing, (2022) Visualcrossing.com Retrieved from:

<https://www.visualcrossing.com/resources/documentation/weather-api/how-do-i-add-weather-forecast-to-my-webpage/>

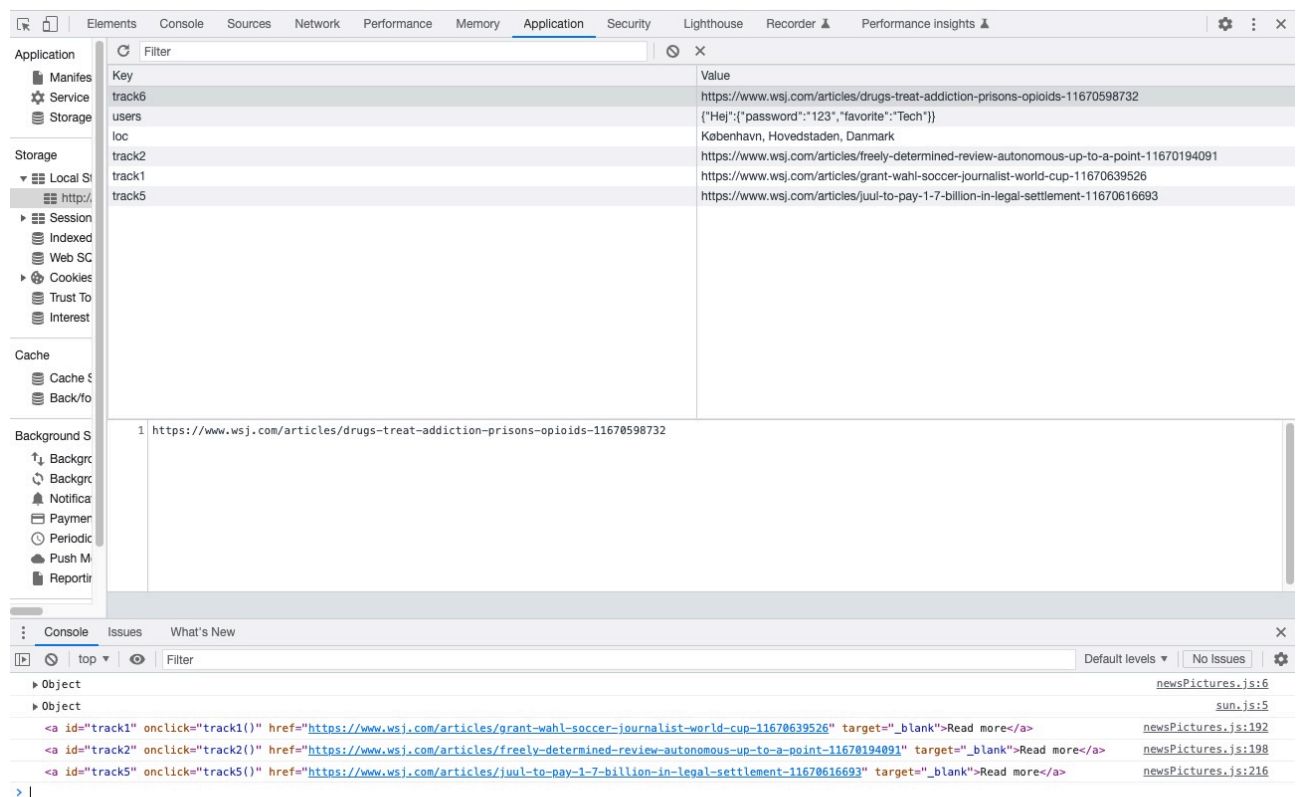
¹⁴ Ha IT, (4/12-2022), hait.cbss.dk Retrieved From:

<https://hait.cbss.dk/weeks/uge-46.html>

¹⁵ GitHub Copilot (2022) Retrieved from:

<https://github.com/features/copilot>

4. Data oversigt



Ovenover ses et screenshot af indholdet på localStorage. Det ses her, hvordan den gemmer brugerens username, password og favorite news. Derudover kan det ses at de forskellige track funktioner fortæller at brugeren har læst 4 forskellige artikler, alle fra Wall Street Journal.

5. Løsningsovervejelser

Under processen prøvede jeg mange forskellige metoder, som alle var nyttige for slutproduktet af applikationen. Jeg fandt meget vejledning og dokumentation på nettet for forskellige funktioner, frameworks, osv. som jeg prøvede at navigere imellem for at finde den bedste løsning. Udover inspiration fra nettet, kunne jeg også trække mange paralleller til pensummet, hvis materialer jeg brugte som yderligere inspiration til at løse kravspecifikationerne.

5.1 Nyheder, brug af api og indsættelse

Min indsættelse af selve nyheds API er på ingen måde optimal og der er mange forhold der kunne forbedres, hvis jeg havde haft tid til det. For det første bruges der en fast størrelse på billederne, hvilket kan gøre det visuelt mindre pænt, hvis de ikke passer ind i den forudbestemte størrelse. Dette kan forbedres ved at bruge CSS til at style billedstørrelsen dynamisk i stedet for at definere en fast størrelse i HTML.

For det andet bruger koden en fast værdi for antallet af nyheder, der skal vises. Dette begrænser fleksibiliteten og muligheden for at vise flere eller færre nyheder, hvis dette skulle ønskes fremover. Det skal derfor understreges at jeg er fuldstændig klar over, at det burde være blevet løst, ved brug af et for-loop, men jeg kunne desværre ikke få dette til at fungere.

Ved at opdele koden i 7 forskellige funktioner, der gør nogenlunde det samme, skaber jeg et problem med fleksibilitet og læsbarhed. Hvis jeg vil tilføje en ny funktion, skal jeg gå ind i hver enkelt funktion og tilføje dette, hvilket kan resultere i fejl eller forvirring. Derfor kan det være en bedre idé at samle lignende funktioner i én større funktion og lade denne håndtere de forskellige opgaver.

Endelig er koden måske ikke så læsbar og nem at forstå for andre udviklere, der måske skal vedligeholde eller ændre den i fremtiden. Dette kan forbedres ved at bruge klare navne på variabler og funktioner, og organisere den på en måde, der gør den nemmere at forstå.

5.2 updateFavoriteNews

addFavorite-funktionen er et godt eksempel på en funktion, der led under den dårlige indsættelse af nyheds-API 'en. Jeg prøvede også at få et for-loop til at fungere, men det lykkedes desværre ikke. Dette skyldes sandsynligvis, at jeg ikke havde brugt et for-loop fra starten af. Det bør bemærkes, at hvis jeg havde brugt mere af min tid på dette, ville jeg have været i stand til at få det til at fungere, men da der ikke var tid til alt, endte jeg med denne løsning.

```
Godkendelsesopgaver > AfsluttendeOpgave > JS favorites.js > ...
1  function addFavorite1 () {
2      let favorite = document.getElementById("favoriteBtn1");
3      favorite.style.backgroundColor = "red";
4      favorite.style.color = "white";
5      favorite.innerHTML = "Favorited";
6  } // Styler knappen til at blive rød og skrive
7  // "Favorited" når den bliver trykket på
8
9  function addFavorite2 () {
10     let favorite = document.getElementById("favoriteBtn2");
11     favorite.style.backgroundColor = "red";
12     favorite.style.color = "white";
13     favorite.innerHTML = "Favorited";
14 }
15
16 function addFavorite3 () {
17     let favorite = document.getElementById("favoriteBtn3");
18     favorite.style.backgroundColor = "red";
19     favorite.style.color = "white";
20     favorite.innerHTML = "Favorited";
21 }
```

5.3 Søgefunktion

Søgefunktionen må siges at være akilleshælen i denne opgave. Jeg har brugt mange timer på at få den til at fungere uden held. Mit bedste gæt på, hvorfor søgefeltet ikke fungerede, er igen indsættelsen af nyhederne. Dette kan jeg sige på baggrund af at have set flere eksempler på, hvordan det skulle fungere, men ikke kunne få min egen til at virke.

5.4 Login

Loginfunktionen havde jeg også en del overvejelser omkring. Den opfylder kravene, men kun også lige præcis. Dette skyldes at den kun kan gemme én bruger ad gangen i localStorage. Jeg overvejede at bruge længere tid på at se hvad det skyldes, men eftersom at den opfylder kravet omkring at: "App'en skal tillade brugeren at logge ind" valgte jeg at bruge min relativt sparsomme tid andetsteds.

5.5 Vejr

Efter at have diskuteret fejl og mangler ved de andre dele af opgaven, må jeg erkende, at det er svært at finde problemer med vejrfunktionen. Jeg overvejede forskellige vejr APIs, men endte til sidst med at bruge Visual Crossing Weather, som er meget intuitiv og brugervenlig

5.6 Tid

Der var flere overvejelser omkring håndteringen af tiden i denne opgave. Jeg overvejede at bruge en API eller en form for funktion, og jeg forsøgte i et stykke tid at bruge Get Date-funktionen. Dette var dog ikke tilfredsstillende, da tiden kun blev opdateret når hjemmesiden genindlæste. Efter et stykke tid, kom jeg så i tanke om setInterval funktionen, som derefter løste hele problemet.

6. Proces evaluering

Formålet med eksamen var at lave en hjemmeside, der viser nyheder. Selvom søgefunktionen ikke fungerede som forventet, lykkedes det at oprette en hjemmeside der opfylder langt størstedelen af kravene. Det er jeg derfor tilfreds med.

Der er flere ting jeg ville have gjort anderledes i dag med den viden og den proces, jeg har været igennem efter dette forløb.

Layoutet på hjemmesiden er ikke det pæneste, men ligner nogenlunde skabelonen.

Når jeg tænker tilbage på forløbet ville jeg have fokuseret mere på at sætte logikken omkring de forskellige div'er ordentligt op fra starten af.

På den måde ville jeg have kunnet spare meget tid med at gå frem og tilbage for at rette småfejl. Dette var langt hen ad vejen det, der frustrerede mig mest.

En anden proces, der ikke var optimal, var måden hvorpå nyhederne blev indsat. Jeg har tidligere nævnt dette, men baggrunden for at processen endte sådan var, at jeg havde brugt meget tid på at få arrayet fra API'en til at fungere. Derfor endte jeg med at bruge denne løsning for at få noget konkret ind på hjemmesiden.

Til at starte med havde brugeren mulighed for at slette alle profiler og clear LocalStorage fra loginsiden. Det gik dog op for mig at dette ikke var helt den bedste løsning og jeg endte derfor

med at flytte Delete Account-knappen til Edit profil-siden. Dette var meget lærerigt og interessant, da det ville være en stor sikkerhedsrisiko at gøre det muligt for alle brugere at slette alle profiler.

Afslutningsvis må det siges at en af de ting, jeg har lært af denne eksamen, er at man ikke skal hoppe over hvor gærdet er lavest, men at man skal tage sig tid til at gøre tingene ordentligt fra starten af, så man undgår problemer senere. Dette gælder især når man har en større opgave og pludselig skal rette mange funktioner individuelt, når man skal ændre en taste- eller logisk fejl.

7. Litteraturliste

- 1 W3Schools, (2022) Retrieved from:
https://www.w3schools.com/howto/howto_css_login_form.asp
- 2 W3Schools, (2022) Retrieved from:
https://www.w3schools.com/jsref/prop_win_localstorage.asp
- 3 W3Schools, (2022) Retrieved from:
https://www.w3schools.com/jsref/met_storage_clear.asp
- 4 W3Schools, (2022) Retrieved from:
https://www.w3schools.com/jsref/prop_win_localstorage.asp
- 5 W3Schools, (2022) Retrieved from:
https://www.w3schools.com/js/js_htmlDOM_html.asp
- 6 The Coding Train, (28/04-2018) Youtube.com, Retrieved From:
https://www.youtube.com/watch?v=QO4NXhWo_NM
- 7 Ha IT, (4/12-2022), hait.cbss.dk Retrieved From:
<https://hait.cbss.dk/weeks/uge-46.html>
- 8 Chat Open AI, (9/12-2022) Retrieved from:
<https://chat.openai.com/chat>
- 9 W3Schools, (2022) Retrieved from:
https://www.w3schools.com/php/func_date_getdate.asp
- 10 GitHub Copilot (2022) Retrieved from:
<https://github.com/features/copilot>

Gustav Manthorpe

Studie nr.: 160305

CBS – HA IT

11 W3Schools, (2022) Retrieved from:

https://www.w3schools.com/js/js_timing.asp

12 Chat Open AI, (9/12-2022) Retrieved from:

<https://chat.openai.com/chat>

13 Visual Crossing, (2022) Visualcrossing.com Retrieved form:

<https://www.visualcrossing.com/resources/documentation/weather-api/how-do-i-add-weather-forecast-to-my-webpage/>

14 Ha IT, (4/12-2022), hait.cbss.dk Retrieved From:

<https://hait.cbss.dk/weeks/uge-46.html>

15 GitHub Copilot (2022) Retrieved from:

<https://github.com/features/copilot>

8. Bilag

Link til video hvor kravene bliver vist:

<https://www.loom.com/share/8485827f0dd2411b8c6fe2eb83f765da>