

Webudvikler Eksamen

Juni 2023

Leospa Beauty Spa



Situationsbeskrivelse

Du er ansat i et webbureau, som udvikler dynamiske webløsninger. Du indgår i et udviklingsteam, der består af en designafdeling og en kodeafdeling. Bureauet har fået en ordre på et website fra virksomheden Leospa Beauty Spa. Layoutet og sitets generelle struktur er allerede fastlagt og godkendt af kunden. Din overordnede opgave bliver at kode sitets funktionalitet.

Opgavebeskrivelse

Leospa Beauty Spa har et ønske om at få en mere synlig profil på Internettet. Derfor ønsker firmaet at få bygget et website, der kan indgå i deres samlede markedsføringsstrategi.

Formålet med sitet er at præsentere firmaets mærkesager på en overskuelig måde, samt give brugere mulighed for let og overskueligt at finde informationer. De ønsker derudover også en ny side til at præsentere deres “services” på en professionel måde og en side til “features” som skal indeholde eventuelle fremtidige events som Leospa afholder.

Aflevering

Sitet skal afleveres som en zip-fil - samt evt. på en lokal eller webbaseret serverløsning efter eget valg. Husk, du må IKKE rette i din originale version af projektet efter aflevering – vil du forberede/øve noget til den mundtlige eksamen, så lav en kopi af sitet, så du kan gå til eksamen med det oprindelige uredigerede projekt.

Desuden skal du ifølge GDPR-lovgivningen give accept til, at eksaminator og censor må gennemgå opgaveløsningen på din computer i forbindelse med den mundtlige eksamen.

Du får yderligere info om aflevering, når afleveringstidspunktet nærmer sig.

Tekniske krav

- Generelle **designfiler og skrifttyper** udleveres sammen med opgaven.

- Som udgangspunkt skal du **følge designet i opsætningen**, men kunden er modtagelig for ændringer i designet, hvis disse kan argumenteres til kundens fordel.
- Sitet skal som minimum opbygges med **HTML, CSS og JavaScript** – fx React og gerne med brug af Sass/Less (eller tilsvarende), TypeScript mv.
- Sitet skal opfylde de gængse standarder inden for **semantisk**, korrekt kode, **accessibility**/tilgængelighed og søgemaskineoptimering/**SEO**. Husk også favicons/manifest.
- Hjemmesiden skal være **responsivt** tilpasset relevante skærmstørrelser – herunder almindelige computerskærme (desktop/laptop), tablet, mobiltelefoner osv.
- Sitet bør indeholde **fejlhåndtering** på alle relevante fejl, så brugeren får en fejlmeddelelse der tager hensyn til sitets indhold.
- Der skal være **validering** på alle formularer/inputelementer.

Web API

Sitets data/indhold kan hentes via det udleverede API/backend-projekt.

Opsætning og opstart af backend/API fremgår af beskrivelser i mappen ”_INSTALLATION” i roden af det udleverede backend-projekt.

Hvor det er nødvendigt, skal du selv **tilpasse billedernes størrelse** til sitets layout.

Det er vigtigt, at du **henter data (tekster, billeder mv.) fra det udleverede API**, hvor disse er tilgængelige i API’et.

I førnævnte mappe er der json-filer til import i **Mongo**-databasen + til din lokale Postman.

Der er også en ekstern **Postman**-dokumentation her:

<https://documenter.getpostman.com/view/12464673/TzCQb77e>

Billeder

Du har fået udleveret grafik og fotos sammen med opgaven.

Billeder, der skal hentes som en del af API-data, findes med følgende endpoints:

- <http://localhost:5029/images/recommendation/paavej.jpg>
- <http://localhost:5029/images/treatment/paavej.jpg>
- <http://localhost:5029/images/team/paavej.jpg>

Port-nummer 5029 forudsætter at du ikke laver om i opsætningen i backend.

Rapport

Du skal udarbejde en projektrapport sammen med opgaveløsningen.

Projektrapporten skal som minimum indeholde følgende:

- Vurdering af din **egen indsats** og gennemførelse af opgaveforløbet, kommentarer til din tidsplan/estimering af tid mv.
- **Argumentation for valg** du har truffet under løsningen af opgaven
- Hvis du har **ændret/tilføjet noget til "backenden"**/API-projektet, så beskriv hvad og hvorfor i rapporten.
- Redegørelse for **oprindelsen af de forskellige kodeelementer** i opgaveløsningen, som du evt. ikke selv har udviklet (link til oprindelsen af koder til fx slider, galleri, "smoothscroll" mv.)
- Angivelse af url-adresser (fx til GitHub), brugernavne og passwords der er nødvendige for at lærer og censor kan se opgaven
- En beskrivelse af eventuelle særlige punkter til bedømmelse

Bilag til rapporten:

- Din **tidsplan**
- Evt. **fotos, screenprints** mv. som dokumenterer din tidsplan, din arbejdsproces mv.

- Andet som du finder relevant at vedlægge

På forsiden af rapporten skal fremgå

- Opgavens navn, dit fulde navn og holdnummer/klasse (webith120)
- URL til sitet, hvis det ligger online
- Eventuelle relevante brugernavne og adgangskoder

Omfanget af rapporten skal være på **maksimum 5 A4-sider** eksklusive bilag.

Projektrapporten afleveres som pdf-fil, og **afleveres i roden af det færdige projekt/website** – og navngives **Rapport.pdf**

Kravspecifikation

Her følger et overblik over de krav der stilles til sitets indhold.

Med opgaven er der udleveret en jpg-fil med layout, som understøtter disse krav.

Elementer i layoutet, som ikke er nævnt i teksten, er som udgangspunkt også en del af opgavens krav (spørg hvis du er i tvivl). Hvor opgaver/elementer ikke er beskrevet eller vist i detaljer, har du frie hænder til selv at vælge, hvordan du vil løse disse elementer – men du skal følge design-stilen og tænke kundens/brugernes behov ind i dine løsninger. **Du skal kunne argumentere for dine frie valg.**

Generelle krav

- Sitet bygges i standardmål efter layoutet
- Sitet skal være **responsivt**

Læs også afsnittet **"Tekniske krav"** herover.

Faste elementer på alle sider

Følgende elementer skal indgå på alle sider (Forside, Services, Features mv.):

- **Logo**
- **Navigation** Menu
- **Footer** med kontaktinformationer (hentes fra API)

På admin-siderne tilretter du navigation mv., så det matcher admin-områdets formål.

Opgaver – sitets indhold

"Forsiden"

- **Header**, navbar mv.
- **Hero** (med link til fancybox/popup/modal el.lign. med visning af Hero's videoklip)
 - Vis kun den Hero, som har `show=true` i API'et
- **About** (med link til "Features"-siden)
- **Treatments** (procedures/services/behandlinger)
 - Først **4 tilfældige treatments/procedures** (kun foto) (lav selv et script som udvælger 4 tilfældige blandt "alle treatments") – med link til "Services"-siden
 - Herefter **3 treatments/procedures** (fx de 3 første i API'et), med både foto og tekst, hvor **teksten er afkortet (...)**, så boksen holder en rimelig størrelse (se layoutet) - med link til "Services"-siden.
- **Kundefeedback/recommendation** (slider)
 - Vis **3 nyeste recommendations** (brug det relevante API-kald)
 - Hver recommendation **linker til en side/oversigt** med alle recommendations (ikke vist på layoutet – følg design-stilen)
- **Team** (vis alle behandlerne)

- **Book en tid** (skal poste ind i API'et)
- **Footer – med kontaktoplysning-data** (ikke vist på layoutet) fra API'et (footer)

“Services” (treatments/procedures)

- Skal **linkes til fra** navbaren/”Service” OG når man trykker på en af de 7 treatments/procedures (4 tilfældige + 3 udvalgte) på forsiden.
- Denne side skal **du selv opsætte** - husk at følge designprofilen.
- Skal vise **ALLE treatments/procedures** (de behandlinger stedet tilbyder)
- Vis gerne noget af teksten med mulighed for ”Læs mere” som så viser en pop-up/modal, folde-ud-tekst el.lign., hvor hele teksten fremgår.

“Features”

- Skal **linkes til fra** navbaren/”Feature” OG når man trykker på knappen i afsnittet ”About...” på forsiden.
- Denne side skal **du selv opsætte og designe**, og du skal vise, at du kan følge designprofilen.
- Opsæt **et eller flere fiktive SPA-/beauty-events** på denne side.
Indholdet af events'ene er ikke vigtigt (kunne fx være aftener med foredrag, præsentation af skønhedsprodukter, løbekonkurrencer med præmier ... bare find på noget) – fokuser på designet/stilen.
- Husk det skal repræsentere **designprofilen**.
- Skal **ikke** indeholde udtræk fra API'et – brug udleverede/ billeder og noget lorem- ipsum/dummy-tekst

Admin-del

Et admin-område (evt. beskyttet bag login – se tilvalgsopgaverne), hvorfra det er muligt at administrere (**oprette/rette/slette**) data til min. følgende:

- **Treatment** (opret, ret slet)
 - **formatering af "content"-teksten** (fed, linjeskift osv.) *fx ved brug af en **editor** (CKEditor el.lign.)*
- **About** (ret)
 - **formatering af "content"-teksten** (fed, linjeskift osv.) *fx ved brug af en **editor** (CKEditor el.lign.)*
- **Hero** (opret, ret, slet)
 - + giv administratoren mulighed for at **vælge, hvilken Hero som skal have **show=true**** (= den Hero som vises på forsiden) – se API'ets PATCH-request
- **Footer** (ret)

API'et har de nødvendige request-metoder (POST, PUT, DELETE osv.) – mangler der noget, eller er der fejl i API'et, så gør opmærksom på det/spørg.

Tilvalgsopgaver

Du skal vælge **minimum 2 af følgende tilvalgsopgaver**. Du må dog gerne vælge flere, hvis øvrige/vigtigere krav er opfyldt eller opgivet (tjek prioritetslisten!):

1. Login

Lav **login-beskyttelse af administrations-delen**, så kun brugere/users oprettet med e-mail og password kan få adgang. Følgende bruger er oprettet i systemet:

hn@leospabeauty.dk med password: **admin123**

Du kan selv oprette flere brugere med Postman eller login.rest (i backenden)

2. Tilmeld nyhedsbrev

Brugere skal kunne **tilmelde og afmelde sig Leospas nyhedsbrev** - du har frie hænder til at designe hvor og hvordan. Der er API-endpoints til både tilmelding og afmelding.

I administrationsdelen skal administrator kunne se en liste/side med alle tilmeldinger – fx med mulighed for at slette tilmeldinger.

4. Administration af “appointments”

En side i admin-delen over **alle kunders kommende appointments/aftaler**, som klinikkens personale kan bruge som en oversigt over de kommende dages bestillinger (evt. til print)

Oversigten skal være opdelt i dato-orden/dato-overskrifter, så det er tydeligt, hvad der er af aftaler/appointments **dag for dag**.

Vis kun appointments fra ”i dag” og frem = frasorter gamle aftaler/appointments.

Og for hver aftale/appointment, **skal det vises tydeligt**, hvilken behandling, der er bestilt (fx fremhævet med forskellige farvekoder), kundens kontaktoplysninger, noter osv.

Udvid evt. med ekstra funktionalitet, så klinikken fx kan indtaste to datoer og se alle aftaler i den pågældende periode – eller en enkelt dates aftaler el.lign.

5. Administration af “users”

Kun users/brugere med **admin=true** (se API’ets user-endpoints) skal have adgang til en side i admin-delen, hvor han/hun kan oprette, rette og slette brugere/users.

Brugere/users KAN fx være behandlerne/teamet – men behøver ikke være det. Bare tænk i funktionalitet her.

6. Ekstra validering af ”Make an appointment”

Udvid validering af dato og tidspunkt, så man **kun kan bestille appointments fra datoen ”i morgen” og frem**. Og højst **6 måneder frem** i tiden. Og kun **inden for åbningstiden**

(mandag til fredag kl. 8 til 16). Lad det enten være umuligt at vælge andet og/eller giv en fejlbesked.

7. Kunder kan se egne appointments/aftaler

Brugere/kunder skal kunne **se egne appointments** ved at medsende/poste telefonnummer (**phone**) og **email**. Du har frie hænder til at designe hvor og hvordan.

Der er API-endpoint til at håndtere et sådan request.

*Nix – det overholder selvfølgelig ikke GDPR-reglerne, men godt at du er opmærksom her :-)
Men ignorer det og fokuser kun på at vise, at du kan opsætte siden og funktionen.*

Ekstra

Har du løst minimum 2 af tilvalgsopgaverne og opfyldt opgavens øvrige krav, har du frie hænder til at udvide med yderligere funktioner – **men de skal være relevante, og du skal kunne argumentere for dine valg.**

HUSK at notere i rapporten hvilke tilvalgsopgaver, du har løst.

Prioritetsliste

1. Generel opsætning af **responsivt design** (husk SEO og accessibility)
2. Sitet skal fremstå med en høj **færdighedsgrad**, som matcher designoplægget - opbygget med html og css mv.
3. Integration af **tekst og billeder udtrukket fra API**
4. Book en tid/**appointment**
5. Siden **Services/treatments**
6. Siden **Features**

7. Admin-del med opret/ret/slet af tekst/billeder fra API'et - **se afsnittet om "Admin-del"** herover, hvor kravene er beskrevet.

8. Tilvalgsopgaver

Aflevering

Fristen for aflevering er lørdag den 17. juni 2023 kl. 14:45 + tid til opsætning af API.

Inden fristen skal du aflevere en zip-fil med følgende på fællesdrevet:

- Mappe med dit færdige webprojekt (UDEN node_modules!!!)
- Rapport.pdf – lagt i roden af webprojektets mappe
- "Backend"/API hvis du har ændret i dette (hvis du ikke har ændret noget, skal du IKKE aflevere backenden).

Afleveres i en mappe på VIDonline efter aftale, når afleveringen nærmer sig.

Held og lykke med opgaven 😊 ...