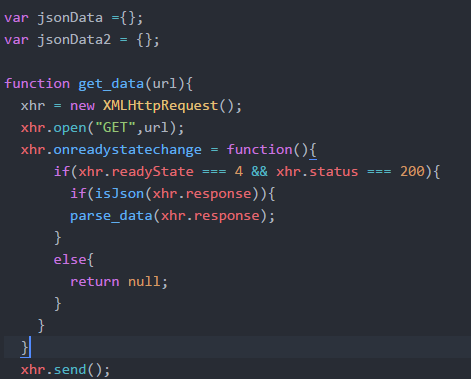
Semesteroppgave info134/Klientprogrammering

Gruppemedlemmer: 134, 197

Denne rapporten vil ta for seg hver del av semesteroppgaven i info134. Rapporten vil inneholde informasjon om hvert html dokument, js og css. I tillegg vil vi vise vår løsning av oppgavene. Mappe hierarkiet vårt er som det står i oppgaven hvor html dokumentene ligger i ytterste nivå, deretter er css filene i en css mappe, js filen i en js mappe og bildene er i media mappen.

Object.js

Object.js inneholder alt vårt javascript for denne semesteroppgaven. Vi har valgt å ha all kode her fordi nesten all kode blir brukt på alle html dokumentene og funksjonene bygger på hverandre, derfor er det enklere for oss å ha alt i en fil.

Figuren nedenfor viser hvordan dataen hentes, funksjonen har et parameter url som i vårt tilfelle er de forskjellige JSON dokumentene. Funksjonen benytter seg av et XMLHttpRequest-objekt som foretar seg forespørselen for å få hentet dokumentene. Hvis den ikke klarer å hente dokumentet så returneres det null. 

Figur 1: Screenshot av Object.js screenshot tatt i Atom.

Funksjonen henter url’en når body lastes i html dokumentene. jsonData variablene blir brukt til å fylle ut listene og kartet. Videre i filen har vi de ulike typene søk og funksjonene som sammenligner distansen mellom toalettene og lekeplassene.

Index.html og Style.css

Index.html er vår forside, vi har valgt å ha et bilde av Bergen som et design brennpunkt og et paragraf under som sier hva siden inneholder. På toppen av siden er det navigasjon til hvert av de andre html dokumentene. Navigasjonen inneholder også vårt hurtigsøk som er plassert til høyre som vil bli gått igjennom senere i teksten. På bunnen av siden er det en footer som inneholder vårt gruppe nummer. Hele body’en er i en flexbox som vi har valgt å kalle flexkasse som er spesifisert i style.css filen. Flexkassen vises i en column og alt er sentrert igjennom flexkassen. Style.css har også responsivitet med media queries som trår inn når skjermen er mindre enn 700px hvor bildet i index.html sin bredde blir mindre og navigasjonen endres med at paddingen og a-elementenes marginer minimeres. Kartet sin bredde blir også mindre. Navigasjon, hurtigsøk og flexbox er likt på hvert av html dokumentene. Vi har valgt å holde css’en samlet slik at designet på siden er ryddig og likt på hver av sidene.

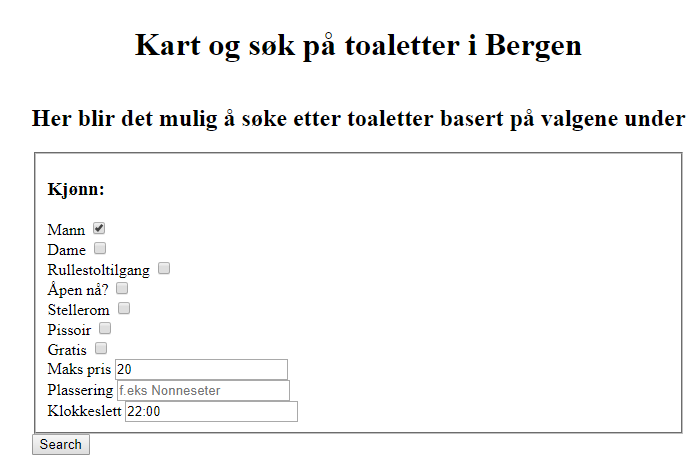


Figur 2: Screenshot av index.html, screenshot tatt i google chrome Versjon 66.0.3359.117 (Offisiell delversjon) (64-bit) Oppløsning 375x812 ( iPhone X)

Figuren over viser at navigasjonen er endret og at siden er responsiv.

Toaletter.html og avansert søk

Dette dokumentet inneholder vårt avanserte søk, her kan man søke på kjønn, rullestoltilgang, pissoir, hvorvidt toalettene er åpne, åpent på et klokkeslett, stellerom, maks pris, om det er gratis og plassering. Søket lager et objekt og matcher med regex på uri’en. Med avansert søk sjekker man for eksempel at Herre=On i urien som blir genert fra checkboksene. Under avansert søk så skrives først ut hele listen av toalettene og kartet med markørene. Både listen og kartet er nummerert og oppdateres med resultatet når det søkes på toaletter. Listen er en ordered list.



Figur 3: Screenshot av toaletter.html, screenshot tatt i google chrome Versjon 66.0.3359.117 (Offisiell delversjon) (64-bit)

Her matcher søket med 3 inputs, mann, maks pris 20 og et klokkeslett og resultatet blir 4 toaletter. For mer enn et alternativ så itereres det igjennom tabellene og alternativene som er matchet puttes inn i en ny tabell via funksjonen find\_common.

På plassering er det både adresse og plassering søk.

Lekeplasser.html

I dette dokumentet skrives ut en nummerert liste av alle lekeplasser som er i JSON-dokumentet og et nummerert kart som inneholder plassering av lekeplassene, her har vi gjenbrukt kode fra toalett listen og kartet.

Favtoalett.html

Her velger man sitt favoritt toalett ved hjelp av drop down menyen som inneholder listen av alle tilgjengelige toaletter. Drop down menyen blir lagt i funksjonen velgFavorittToalett, man må trykke på velg her og deretter velge toalett. Her blir det brukt select elementer. Valget blir lagret og sendt videre til funksjonen lagListeMedLekeplasser. Funksjonen kalkulerDistanse tar inn koordinatene som parameter og regner ut distansen mellom toalettet og lekeplassen. Funksjonen bruker Haversine sin formel som gjør om distansen mellom to punkt på sfæren til en kule fra grader til km. Deretter vil funksjonen sorter\_lekeplasser sortere lekeplassene på distanse. Når ditt favoritt toalett er valgt så vises de 5 nærmeste lekeplassene i en sortert liste med distansen mellom det valgte toalettet og lekeplassene. Hvis distansen er mer enn 1000 meter så vil funksjonen lengdeTekst gjøre det om til km i stedet for meter. Før man velger favoritt toalett så vises alle toalettene på kartet.



Figur 4: Screenshot av favtoalett.html, screenshot tatt i google chrome Versjon 66.0.3359.117 (Offisiell delversjon) (64-bit)

Figuren over viser de 5 nærmeste lekeplassene til toalettet som er på Skyss kundesenter. Etter det er valgt så vises de 5 nærmeste lekeplassene også på kartet. Lekeplassene er nummerert så man kan se på kartet hvor de er posisjonert.

Utveksling.html

På dette dokumentet har vi importert et json dokument som inneholder informasjon om alle utvekslingsavtalene til universitetet i Bergen. Her er det mulig å søke etter land for å se om UiB har utvekslingsavtale og i så fall hvor mange avtaler som finnes med ulike universiteter.

Land -> dropdown til fakultet/institutt?

Figur 5 åpne data ting

Hurtigsøk

Hurtigsøk funksjonen tar inn uri’en til dokumentet, kjører regex på uri’en og lagrer resultatet i en tabell. Resultatet blir så søkt igjennom og matchet med json settet, for eksempel søkes det igjennom hele lengden til resultat tabellen, hvis resultat matcher plassering så søkes det igjennom alt som er av index plassering og matcher en string etter kolon. Alle hurtigsøk resultatene blir sendt til toalett.html siden, hvor listen og kartet blir oppdatert med resultatet hvis det er match. Her er noen eksempler på hvordan man søker ved hjelp av hurtigsøk på siden:

Plassering:nonneseter

Herre:1

Plassering:nonneseter herre:1 pris:20

Strømgaten åpen:nå

Adresse og plassering blir tolket som fritekst med en gang og vil søkes for å finne match fra JSON dokumentet. De kan ikke søkes på samtidig da det ikke er noen måte å skille de på. I så fall må man skrive plassering som et kriterie som vist over.

Strømgaten herre:1 pris:10 åpen:nå

Det er også mulig å søke fritekst på engelsk på plasseringer.

Figur liste

Figur 1: Screenshot av Objekt.js

Figur 2: Screenshot av Index.html

Figur 3: Screenshot av Toaletter.html

Figur 4: Screenshot av favtoalett.html

Figur 5: åpnedta