

# Oblig 2 info134

Grupped medlemmer: Joakim Moss Grutle, Øyvind Skeie Liland

<http://wildboy.uib.no/~jgr036/Semesteroppgave/Sem2/> eller  
<http://wildboy.uib.no/~oli050/INFO134/Sem2/>

## Arbeidsprosess:

Vi jobbet sammen med oppgaven i gruppetimene og på lesesal. Jobbet også litt individuelt hjemme, men det meste av arbeidet skjedde på campus sammen. Det var ikke mye fordeling av oppgaver, vi diskuterte hva som måtte gjøres og så valgte man selv hvilken av oppgavene man ville gjøre, men vi sørget for at vi ikke gjorde det samme.

Arbeidsmengden var ca. 50/50, begge har arbeidet med ulike deler av oppgaven og ble enige om design og struktur av nettside sammen. Øyvind har hatt et større fokus på frontend og JSON og Joakim har fokusert mest på Javascript, men begge to har bidratt på begge områdene.

Dette har fungert veldig greit for oss, vi får til å være effektive og jobber bra sammen.

## Programmeringsoppgaven:

Hvordan utføre advancedSearch: Trykk på "avansert søk" og kryss av i boksene eller fyll inn verdier (klokkeslett må ha punktum eks. 16.00)

Hvordan utføre freeSearch: Her kan man skrive inn plassering, adresse eller bare bokstaver som man vil lete etter, det trengs ingen stoppetegn for plass-søket (f.eks. bare søke etter "bergen" funker fint).

Om man vil legge til flere kriterier vil syntaxen være "kriterie:on" eller "kriterie:gittVerdi" for de ulike kriteriene

Kriterier som matcher med "kriterie:on" (case sensitive):

- rullestol:on
- stellerom:on
- gratis:on
- åpenNå:on

Kriterier som matcher med "kriterie:gittVerdi"

- kjønn:verdi (herre eller kvinne)
- åpen:verdi (klokkeslett med punktum eks. 16.00)
- maksPris:verdi (valgfritt tall)

I freeSearch kan du ha både plassering og kriterier samtidig med mellomrom mellom, men plassering må alltid være først (eks. "nonneseter kjønn:herre rullestol:on"). Man kan også bare ha plassering eller bare kriterier (eks. "nonneseter" eks. "åpen:22.00"). Det er fullt mulig å utføre både freeSearch og advancedSearch samtidig. Om en utfyller samme kriterie i både freeSearch og advancedSearch vil advancedSearch overkjøre freeSearch.

### **Lesing av JSON-filen:**

JSON filen er lastet inn som en variabel i JSON.js

Denne variabelen brukes i funksjonen readJSON(). I denne funksjonen parser vi JSON-filen og itererer gjennom alle toalettene med en løkke. For hvert toalett itererer vi gjennom alle Entries: Hver key og value (entry) blir lagret i et Key-Value objekt som pushes til dette toalettet sitt array. Når alle entries for dette toalettet er iterert gjennom og er lagt til i arrayet pushes arrayet til toiletArr (Et to-dim-array. hvor hver indeks er et toalets array med objekter)

### **Hvordan fungerer algoritmen:**

På refresh/submit:

- Match URL med Regex-uttrykk i variablene "advancedSearchRegex" og "freeSearchRegex"
- Evt. split uttrykkene opp og legg til i arrays (freeSearchArray og advancedSearchArray)

1. Onload: kjør advancedFreeSearch() - funksjonen
2. advancedFreeSearch() kaller readJSON() som oppretter toiletArr - array med toaletter og deres entries
3. Oppretter SearchCriteria-objektet, med alle verdier lik false gjennom searchCrit() funksjon
4. Se om det er blitt gitt noe input i freeSearch eller advancedSearch (hvis det ikke er noen input ignorer pkt. 5-7)
5. Bruk searchMatching() funksjonen til å sette verdiene i searchCriteria-objektet til verdiene fra gitt søk (freeSearchArray og advancedSearchArray)
6. matchToiletWithCriteria()-funksjonen matcher searchCriteria-verdiene med toalettene som har blitt lest inn gjennom readJSON() (toiletArr)
7. Oppdater toalettene i toiletArr basert på de som har matchet alle søkekriterier
8. Oppdater numerert liste (med oppdatert toiletArr)
9. Oppdater kartet (med oppdatert toiletArr)