|  |  |
| --- | --- |
|  | Prosjekt i Databaser og WEB |
|  |  |
| 23.02.2018 | Et prosjekt av Mathias Moxnes, Patrick Pedersen, Kim Næss, Ingeborg Vik Anundsen og Ola Stålberg |
|  | Viser til prosjektoppgaven gitt i emnet 60-65-1 Databaser og WEB våren 2018. Prosjektgruppen jobber med en web-applikasjon som har som hovedmål å gi brukeren innsyn i tilsynsrapporter for norske spisesteder/restauranter. Resultatene er basert på tilsynsrapporter utført av Mattilsynet fra jan. 2016 til d.d.. |

Prosjekt i Databaser og WEB

Et prosjekt av Mathias Moxnes, Patrick Pedersen, Kim Næss, Ingeborg Vik Anundsen og Ola Stålberg

Innhold

[Hva skal applikasjonen kunne utføre 2](#_Toc507181610)

[Hvem skal benytte applikasjonen 3](#_Toc507181611)

[Problemstillinger 3](#_Toc507181612)

[Beskrivelse av databasen 3](#_Toc507181613)

[Restauranter 3](#_Toc507181614)

[Tilsynsrapporter 4](#_Toc507181615)

[Kravpunkter 4](#_Toc507181616)

[Brukerdatabase 4](#_Toc507181617)

[Funksjonsbeskrivelse 5](#_Toc507181618)

[Søkefunksjon 5](#_Toc507181619)

[Smilefjes-resultater 5](#_Toc507181620)

[Hvor restauranten befinner seg 5](#_Toc507181621)

[Søk på restauranter i nærheten 5](#_Toc507181622)

[Kategorier 5](#_Toc507181623)

[Mobilvennlighet 5](#_Toc507181624)

[Alternativt restaurantsøk – Vis de dårligste forslagene 6](#_Toc507181625)

[Tilsynsrapporter og datoer 6](#_Toc507181626)

[Administrasjonsverktøy 6](#_Toc507181627)

[Kort om Programbiblioteker 6](#_Toc507181628)

[Google Maps API: 6](#_Toc507181629)

[USE CASE AV brukerinteraksjonen 7](#_Toc507181630)

[skisser av brukergrensesnittet 8](#_Toc507181631)

[Gjennomføre søk 8](#_Toc507181632)

[Sjekke tilsynsrapport 11](#_Toc507181633)

[Logget inn i administrasjonsverktøy 12](#_Toc507181634)

[Administrasjonsverktøy 13](#_Toc507181635)

[Fyll inn ny tilsynsrapport 14](#_Toc507181636)

# Hva skal applikasjonen kunne utføre

De medvirkende til prosjektet har bestemt seg for at webapplikasjonen skal kunne utføre følgende:

1. Søkefunksjon som gir deg muligheten til å slå opp en gitt restaurant. Resultatet skal vise de siste tilsynsrapportene og deres karakterer. I tillegg håper vi på å kunne vise et egendefinert kvalitetsstempel basert på faktorer beskrevet i del 4 av dette dokumentet.

Søket bør kunne utføres på:

• Navn

• Organisasjonsnummer

• Adresse

• Postnummer

• Og kombinasjoner av overnevnte

2. Klassisk smilefjes-resultater skal vises etter suksessfullt søk.

3. Vi ønsker at resultatet skal vise et kart som viser hvor restauranten befinner seg.

4. (Kanskje) Vi ønsker at du ved hjelp av GEO-lokasjon skal kunne finne de nærmeste restaurantene i forhold til brukerens GEO-lokasjon.

5. Et ønske til applikasjonen er at den skal kunne skille restauranter i forskjellige kategorier basert på %string% som for eksempel:

a. «kinesisk», «Kina», «China», «Asian», «Thai», «Indian», «Tandori» o.l. som Asiatisk

b. «Italia», «Pizza», «Pasta», «Milano», «Italy» o.l som italiensk osv.

6. En mobilvennlig versjon av webapplikasjonen skal fungere.

7. Et morsomt alternativ som vår applikasjon potensielt skal kunne utføre er å vise alle de dårligste restaurantene i et gitt området basert på postnummer og dårlige karakterer i tilsynsrapportene.

8. Alle tilsynsrapportene har datoer tilknyttet. Applikasjonen skal vise hvor lenge siden det var sist t ilsyn ble utført og basert på egne satte regler skal den gi beskjed om det er for lenge siden det ble utført et nytt tilsyn.

9. Vi tenker en administrasjonsside, som krever innlogging, hvor det er mulig å legge til, redigere eller slette tilsynsrapporter, eller tilsynsrapportkravpunkter.

# Hvem skal benytte applikasjonen

Applikasjonen er ment for alle som er interessert i å hente informasjon om restauranter. Det kan være folk som ønsker å spise på et nytt sted, eller mennesker som ønsker å sjekke utestedets tilsynsrapporter. Vi ønsker å innføre et nytt kvalitetsstempel som kan gjøre applikasjonen mer populær enn alternative applikasjoner på nettet.

Vår applikasjon har samlet funksjonalitet som er attraktiv fra flere steder, og gjort den tilgjengelig på et sted. Eksempel er GEO-lokasjon og restaurantsøk fra Google koblet sammen med Matportalen.no sin Smilefjes-oversikt. Dette gjør den attraktiv for mange brukere.

Vi ser at restauranteiere også kan ha ønske om å bruke applikasjonen til å sjekke konkurransen på restauranter i sitt nærområde.

Applikasjonen skal kunne brukes av Mattilsynet for å legge til, redigere eller slette tilsynsrapporter, eller tilsynsrapportkravpunkter.

## Problemstillinger

1. Hvordan skal vi klare å overføre datasettet fra data.norge.no inn i vår egen base uten å skade innholdet? Vi ønsker å bevare integriteten tilsynsrapportene har fra Mattilsynet.
2. Dataen skal være tilgjengelig for alle som ønsker å søke. I tillegg jobber vi mot en løsning som skal tillatte brukere fra Mattilsynet eller andre administratorer å legge til, eller endre på tilsynsrapporter.
3. Kart, GEO-lokasjonsdata og adresser er kjernen i det som skal bli en applikasjon som lar brukeren se hvor restaurantene er, og kanskje oppdage nye restauranter i nærheten. Hvordan går vi frem for å skape en brukervennlig portal til dette formålet? Og hvordan skal vi gjøre den tilgjengelig for mobile brukere?

# Beskrivelse av databasen

For å unngå redundans har vi valg å dele Mattilsynets tilsynsrapporttabell i to tabeller. Tabellenb ‘Restauranter’ som inneholder en liste av alle spisesteder i Norge, og tabellen ‘Tilsynsrapporter’ som holder på alle tilsynsrapporter.

Dette gjør at vår database inneholder totalt 4 tabeller som til sammen skal supportere vår applikasjon med data:

## Restauranter

Denne tabellen inneholder informasjon om restauranten og består av:

* TilsynsobjektID
* Navn
* Organisasjonsnummer
* Adresselinje 1 og 2
* Postnummer

TilsynsobjektID er primærnøkkelen i denne tabellen

Postnummer er fremmednøkkel til tabellen(kolonnen) Poststed(postnummer).

## Poststed

* Postnummer
* Poststed

Postnummer er primærnøkkel i Poststed.

## Tilsynsrapporter

Denne tabellen inneholder informasjon om tilsynsrapporter utført på norske restauranter, og består av:

* TilsynsID
* Sakref
* Status
* TilsynsobjektID
* Dato
* Totalkarakter/Smilefjeskarakter
* Tilsynsobjekttype
* Karakter\_Rutiner\_Ledelse
* Karakter\_Lokaler\_Utstyr
* Karakter\_Mathåntering\_Tilberedning
* Karakter\_Merking\_Sporbarheit

TilsynsID er primærnøkkel til denne tabellen. TilsynsobjektID er fremmednøkkel til Restauranter-tabellen.

Alle datatyper til kolonnene er valgt med optimalisering av lagringsplass i tankene.

## Kravpunkter

Kravpunkter er en tabell som inneholder alle kravene som en tilsynsrapport består av. Den er koblet sammen med tilsynsrapporten ved hjelp av fremmednøkkel og skal ha auto\_increment på KravpunktsID. Kravpunkter har følgende kolonner:

* TilsynID
* Dato
* Ordningsverdi
* Kravpunktnavn
* Karakter
* Kommentar

TilsynsID sammen med Ordningsverdi er primærnøkkel til tabellen. TilsynsID er også fremmednøkkel til Tilsynsrapportertabellen sin kolonne TilsynsID.

## Brukerdatabase

Vi skal ha tilgang til en Brukerdatabase som inneholder brukere som har tilgang til å legge til, redigere eller slette tilsynsrapporter, eller tilsynsrapportkravpunkter.  
Det krever innlogging med passord for å gjøre endringer.

Databasen skal inneholde følgende kolonner:

* BrukerID
* Passord
* Fornavn
* Etternavn
* Telefonnummer
* Adminrettighet
* BrukerID er primærnøkkel i Brukerdatabasen.

Adminrettighet er en Boolean tenkt til å være True om du har rettighet til å legge til, redigere eller slette tilsynsrapporter, eller tilsynsrapportkravpunkter.

# Funksjonsbeskrivelse

## Søkefunksjon

Søkefunksjonen vår skal være 1 søkefelt hvor bruker bestemmer hva de søker på. Vår applikasjon søker opp kundens «input» mot restaurantnavn, organisasjonsnummer, adresse, postnummer og kombinasjoner av overnevnte. Når du skal søke på nettsiden, så får du valg om hva du ønsker å søke på. Dette gjør det enklere for Applikasjonen å forstå hva brukeren ønsker å søke på. Vi håper at 2-3 søkefelt vil være nok for å tilfredsstille brukeren.

I tillegg til vanlig søk fra brukeren, så vil det være en knapp tilgjengelig som søker opp de nærmeste restaurantene i forhold til brukerens GEO-lokasjon. Vi ønsker å gjøre alle søkene trygge mot SQL-planting og tversoverskripting.

Søkene vil kobles mot en spørring som gjøres i databasen. Resultatet av spørringen skal vises i nettleseren til brukeren. Ved GEO-lokasjon har vi ikke lært oss helt ennå hvordan vi skal koble det til vår database, men vi håper at API’et til Google har muligheten til å returnere adresser på restauranter i nærheten, som vi igjen kan presentere til brukeren som resultat. Denne delen er noe usikker foreløpig.

## Smilefjes-resultater

Når brukeren har utført et søk i vår applikasjon og valgt en restaurant, så gjennomfører vi en spørring for å sjekke totalkarakteren til kunden. Basert på resultatet i denne spørringen så viser applikasjonen et smilefjes som reflekterer om restauranten har fått godkjent karakter på tilsynsrapporten. Vi vil også vise frem et eget kvalitetsstempel basert på restaurantens score over flere tilsyn. Stempelet skal fremheves som et alternativ til Mattilsynets smilefjesrangering.

## Hvor restauranten befinner seg

Til dette bruker vi Google Maps Embedded API. Dette krever at vi har en Google API-key, som fanger opp hvem det er som bruker deres API i tillegg til adressen til en restaurant. Dette sendes inn som parametere til Google Maps og vi får tilbake et kart med en markør som viser lokasjonen til restauranten. iFrame brukes for å implementere dette kartet inn på vår nettside.

## Søk på restauranter i nærheten

Vi bruker innebygd html5-funksjonalitet for å hente ut brukerens GEO-lokasjon i longitude og latitude. Vi skal forsøke å overføre databasens restaurantadresser til GEO-lokasjoner, og det gjøre at vi kan gjøre et kalkulert søk på GEO-lokasjoner innenfor en radius på 5km i forhold til brukeren.

## Kategorier

Applikasjonen skal ha flere kategorier å velge mellom. Dette er forhåndsopprettede tabeller som inneholder restauranter innenfor forskjellige kategorier. Eksempel på kategori kan være ‘Indiske restauranter’. Velger du ‘Indiske restauranter’ setter vi databasen søket blir utført i til en database som typisk kun inneholder restauranter med indisk tilknytting. Da restaurantene er opprettet i forkant vil det være mulig for brukeren å søke på et stedsnavn og kun få opp restaurantene som er inkludert i den kategorien. Som eksempel vil søket: ‘Bø’ med valgt kategori ‘Kinesisk’ ekskludere for eksempel Pizza-fjoset.

## Mobilvennlighet

Vi følger Google sin anbefaling om responsiv web design. Vårt oppsett skal tillatte at webserveren sender den samme html-siden til alle klientene og CSS blir brukt til å gi siden egenskaper basert på klienten webleser og oppløsing. Alle sidene skal godkjennes av ved å bli sendt gjennom W3 sin HTML/CSS-validering. I tillegg til dette vil vi følge grunnleggende SEO.

## Alternativt restaurantsøk – Vis de dårligste forslagene

Som en artig sideløsning, vil vi også bruke GEO lokasjon til å lage en oversikt over de dårligste restaurantene innenfor et område på fem kilometer. Dette vil fungere som søk etter beste restauranter i nærheten, men resultere de med dårligst tilsynsrapportkarakter.

En god funksjon for å vite hvor man ikke skal spise når man er på en ny plass og er usikker på hvor man skal spise.

## Tilsynsrapporter og datoer

Alle rapporter har en tilknyttet dato. Når brukeren utfører et søk, og velger en restaurant, så vil databasen utføre en spørring som sjekker om det er for lenge siden det er blitt utført et tilsyn av bedriften. Et alternativ her kan være å få opp en liste når man logger seg inn som administrator. Dette kan være et nyttig verktøy for Mattilsynet for å passe på at restauranter ikke går for lenge uten å bli sjekket.

## Administrasjonsverktøy

Brukeren kan logge inn og få tilgang til administrasjonsverktøy direkte fra web-applikasjonen. Etter innlogging blir brukeren presentert med muligheten for å legge til, endre eller slette tilsynsrapporter. Velger brukeren å legge til en ny rapport må de fylle inn et skjema som inneholder alle felter som må fylles ut (se tabellen «Tilsynsrapporter»).

Når brukeren velger "lag ny rapport" vill formet bli konvertert til en db-spørring med php og sendt til db ved hjelp av msqli-verktøy og funksjoner.

Ønsker brukeren å endre på kravpunkter, så har brukeren mulighet til dette.

Brukeren må fyller inn et skjema på lignende måte som når det skal opprettes et nytt tilsyn.

Hvert kravpunkt er representert på en enkel rad, og består av informasjon hentet fra Kravpunkts-tabellen, og hele skjemaet blir sendt ved hjelp av php som spørring til databasen.

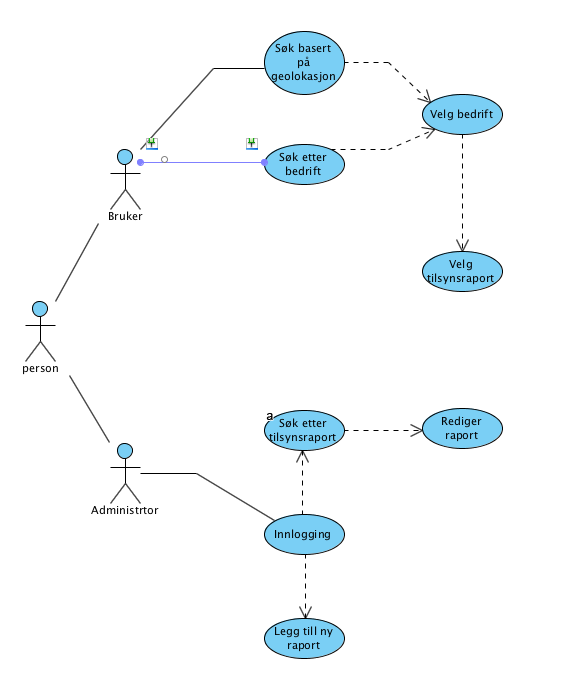
# Kort om Programbiblioteker

## Google Maps API:

(<https://developers.google.com/maps/documentation/embed/>)

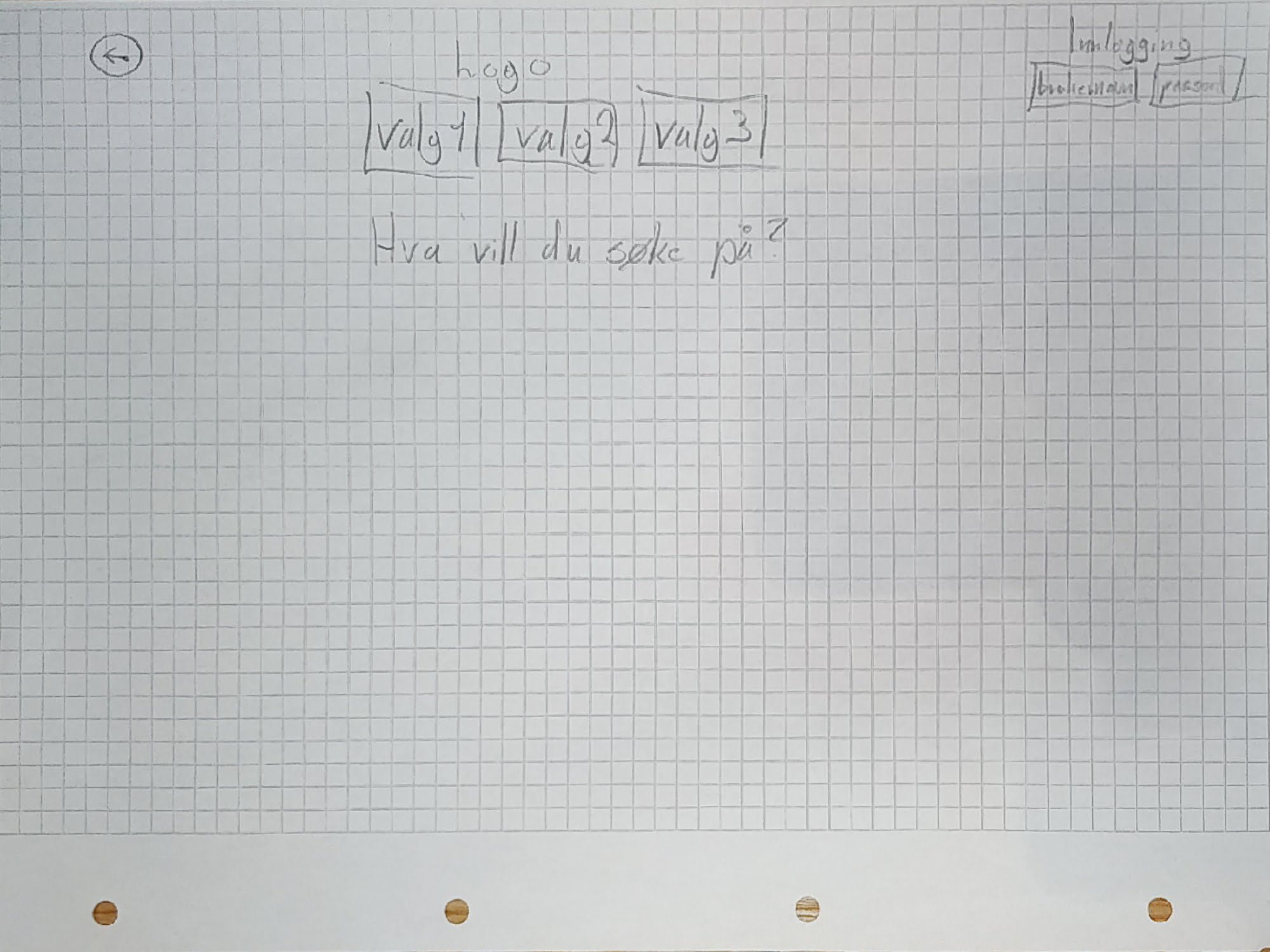
Vi har vært i kontakt med fagansvarlig Bjørn og fått godkjent bruken av Google Maps API for å integrere kart i webapplikasjonen. Det er mulig at løsningen som er innebygget i HTML5 er nok i seg selv. Vi vil undersøke nærmere hvilke API som prosjektet kommer til å ta nytte av, men GEO-lokasjon fra listen under er det stor sjanse for at inkluderes:  
<https://developers.google.com/maps/get-started/>

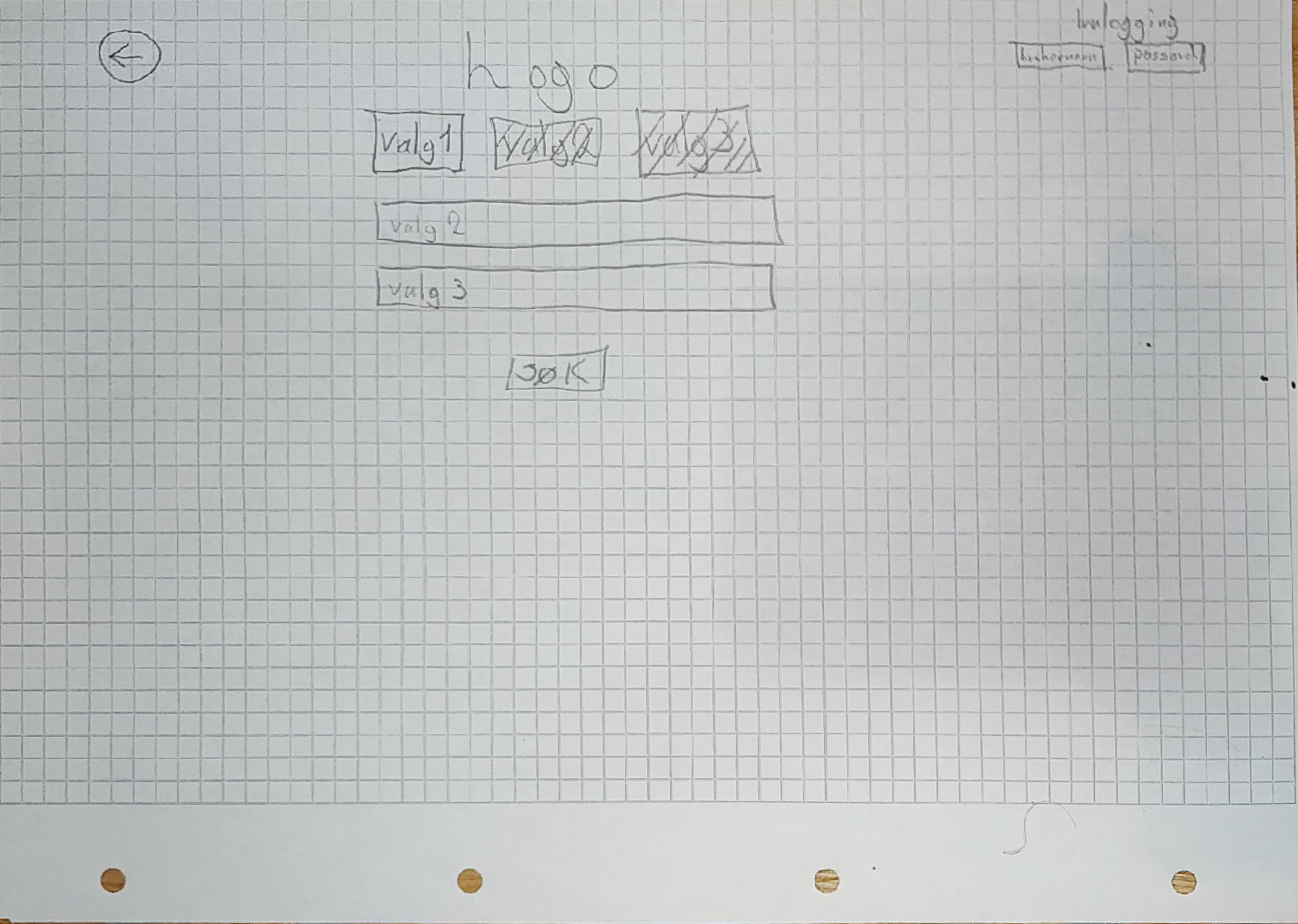
# USE CASE AV brukerinteraksjonen

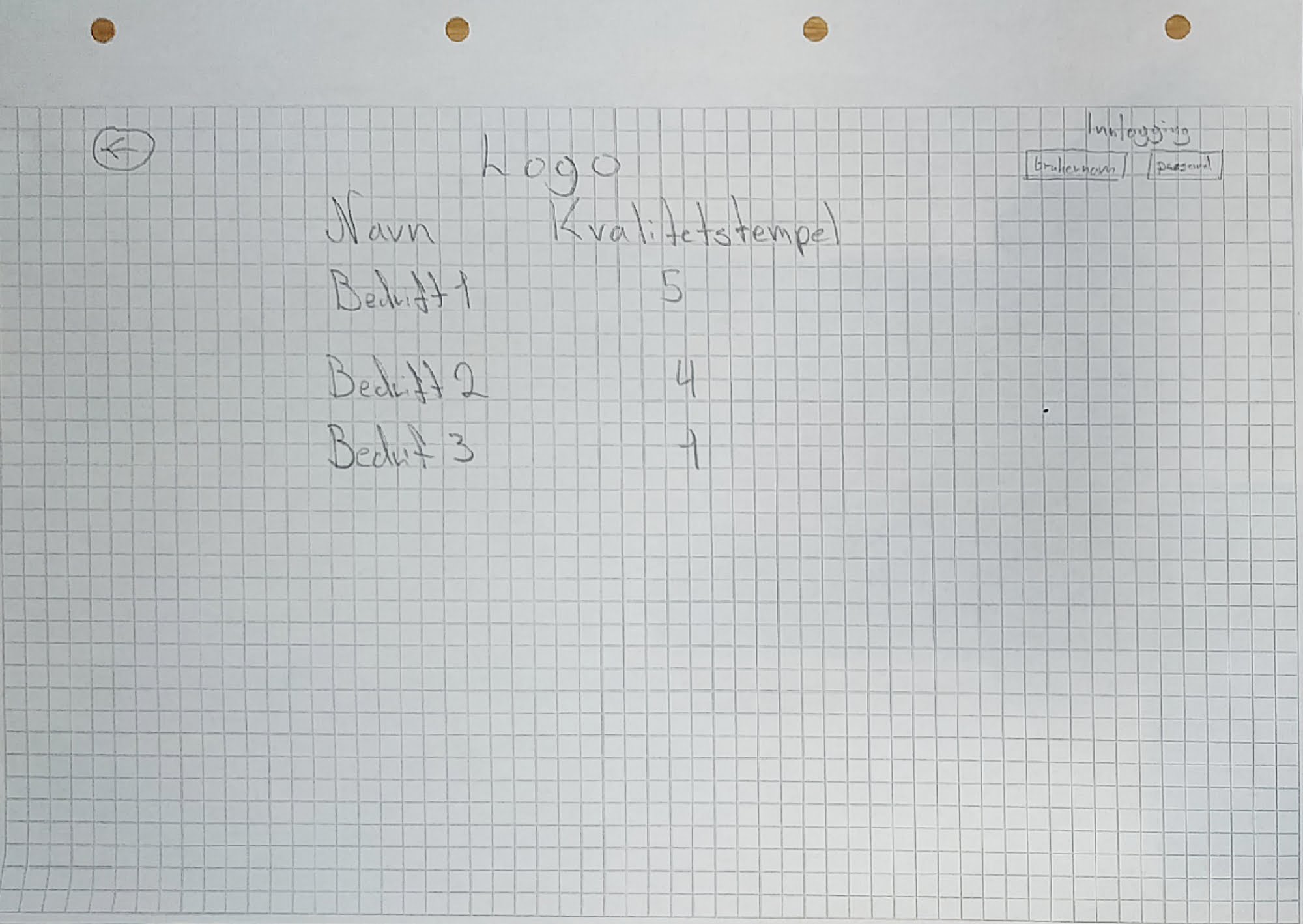


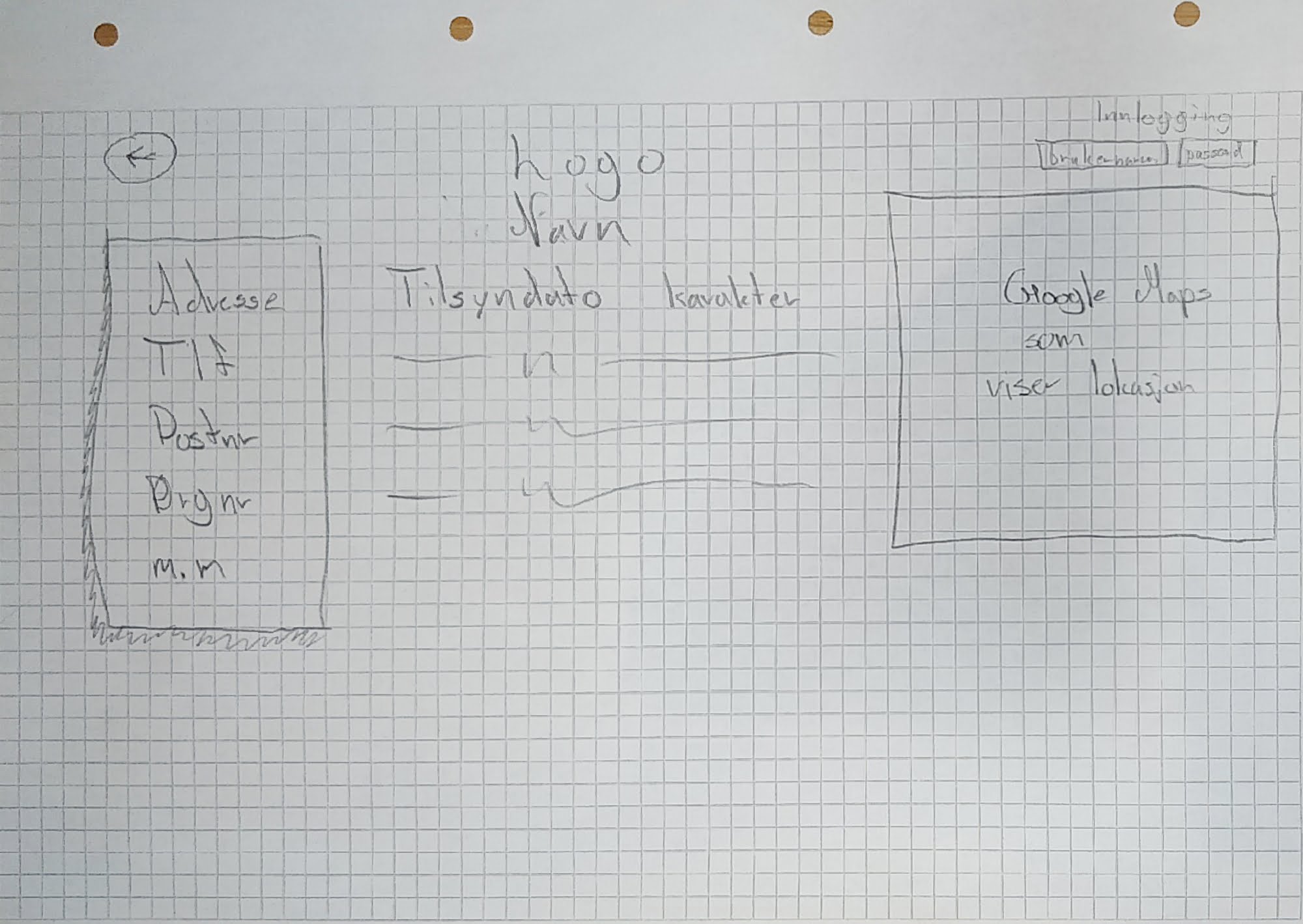
# skisser av brukergrensesnittet

## Gjennomføre søk

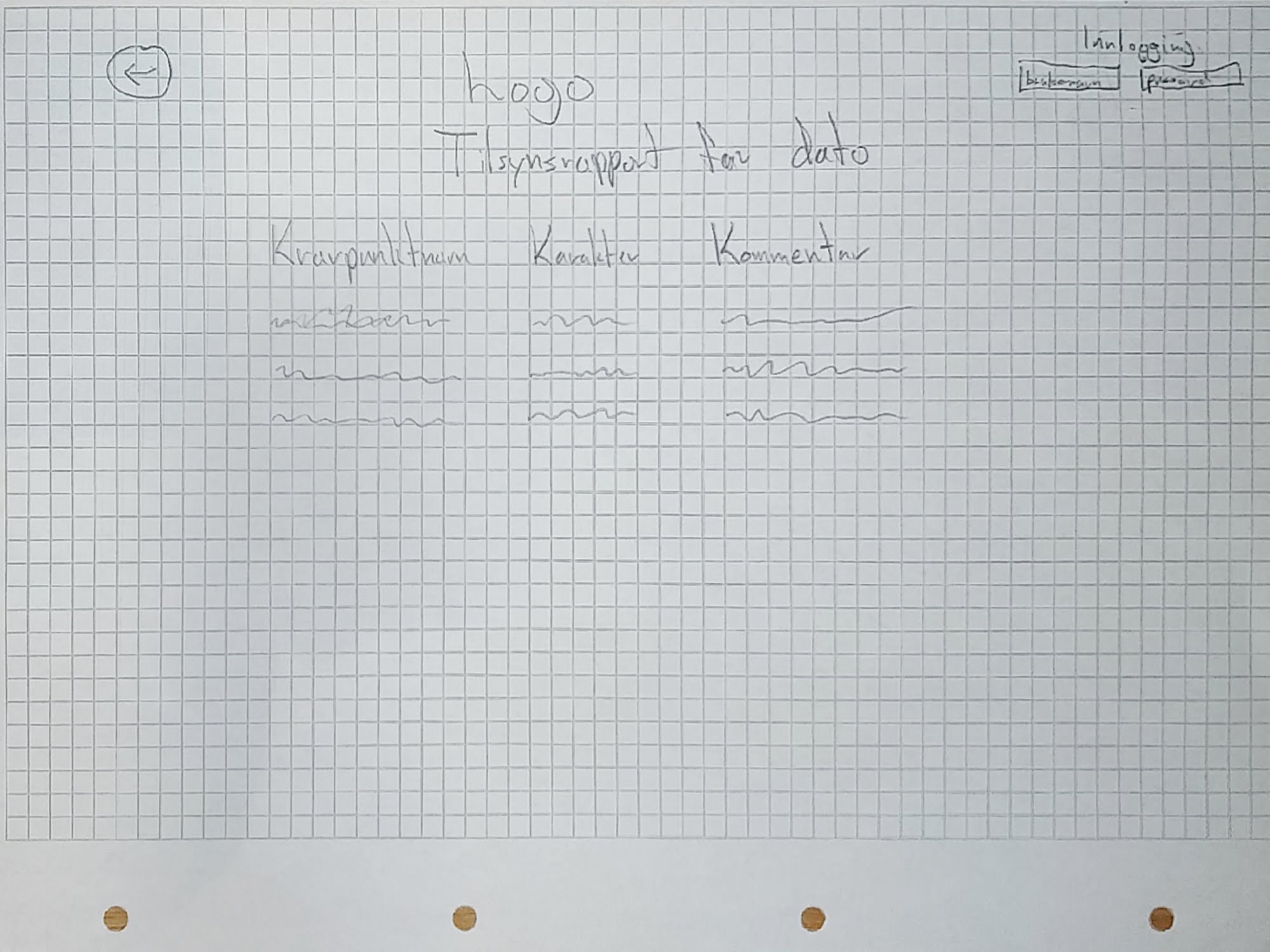




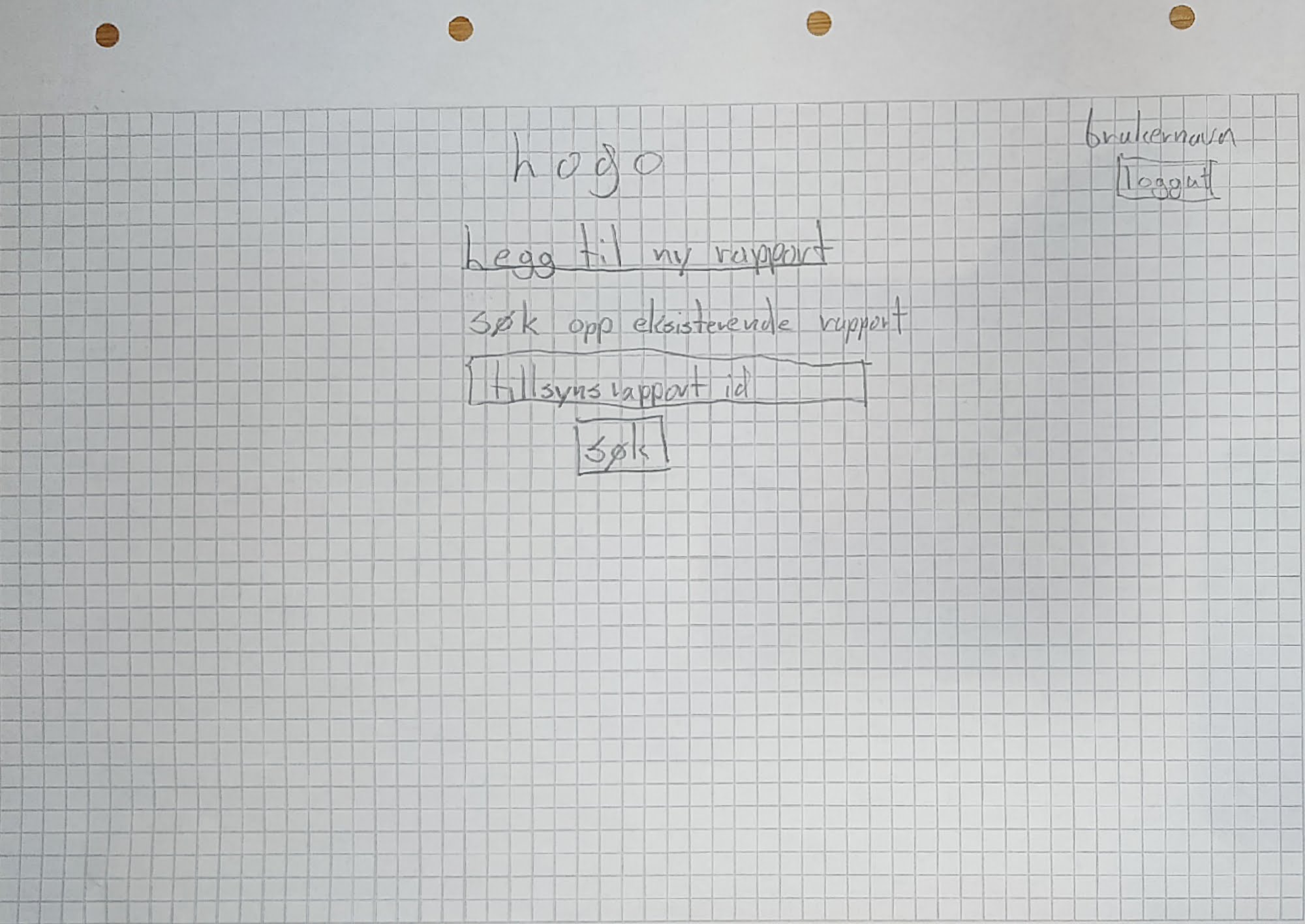




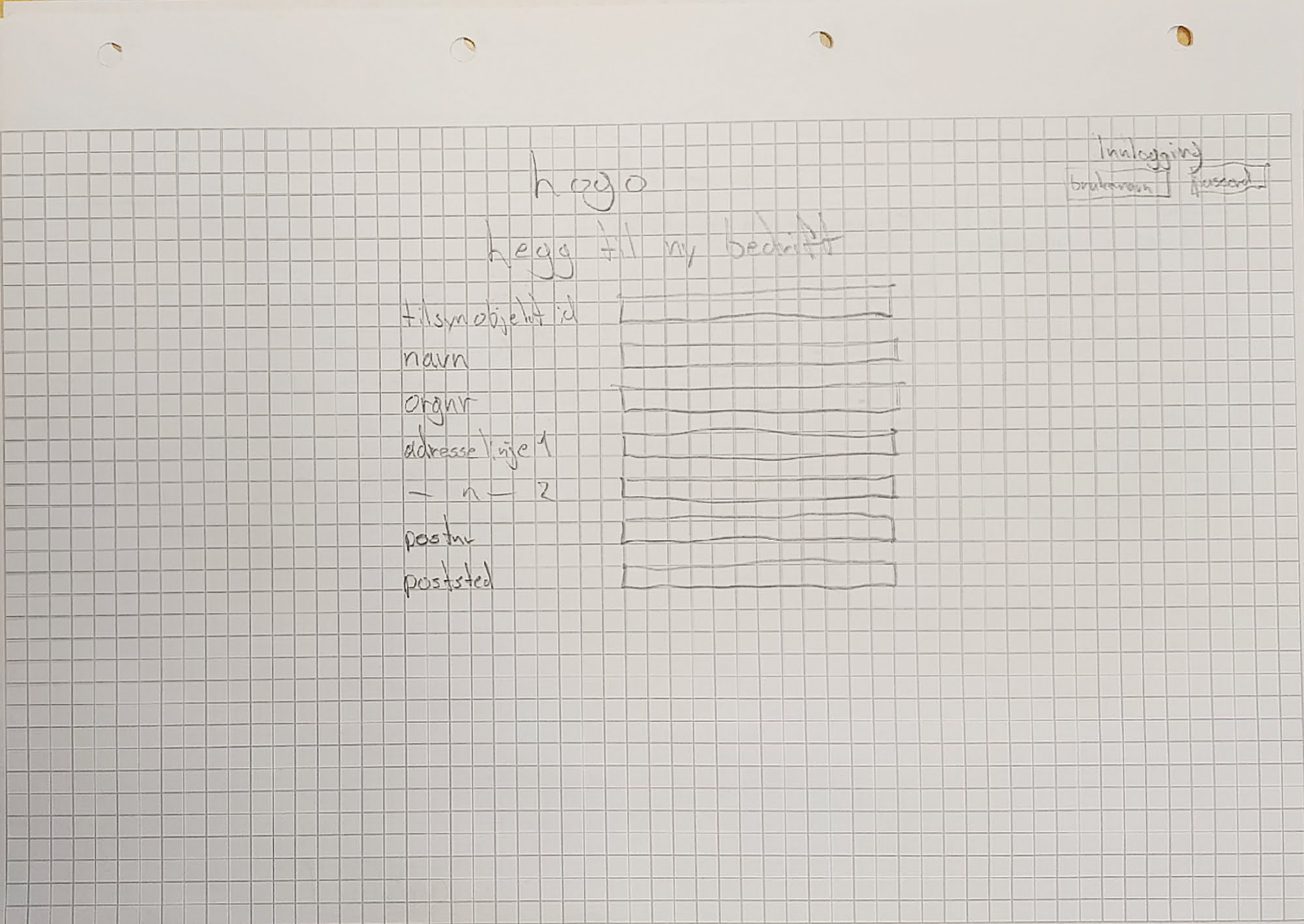
## Sjekke tilsynsrapport

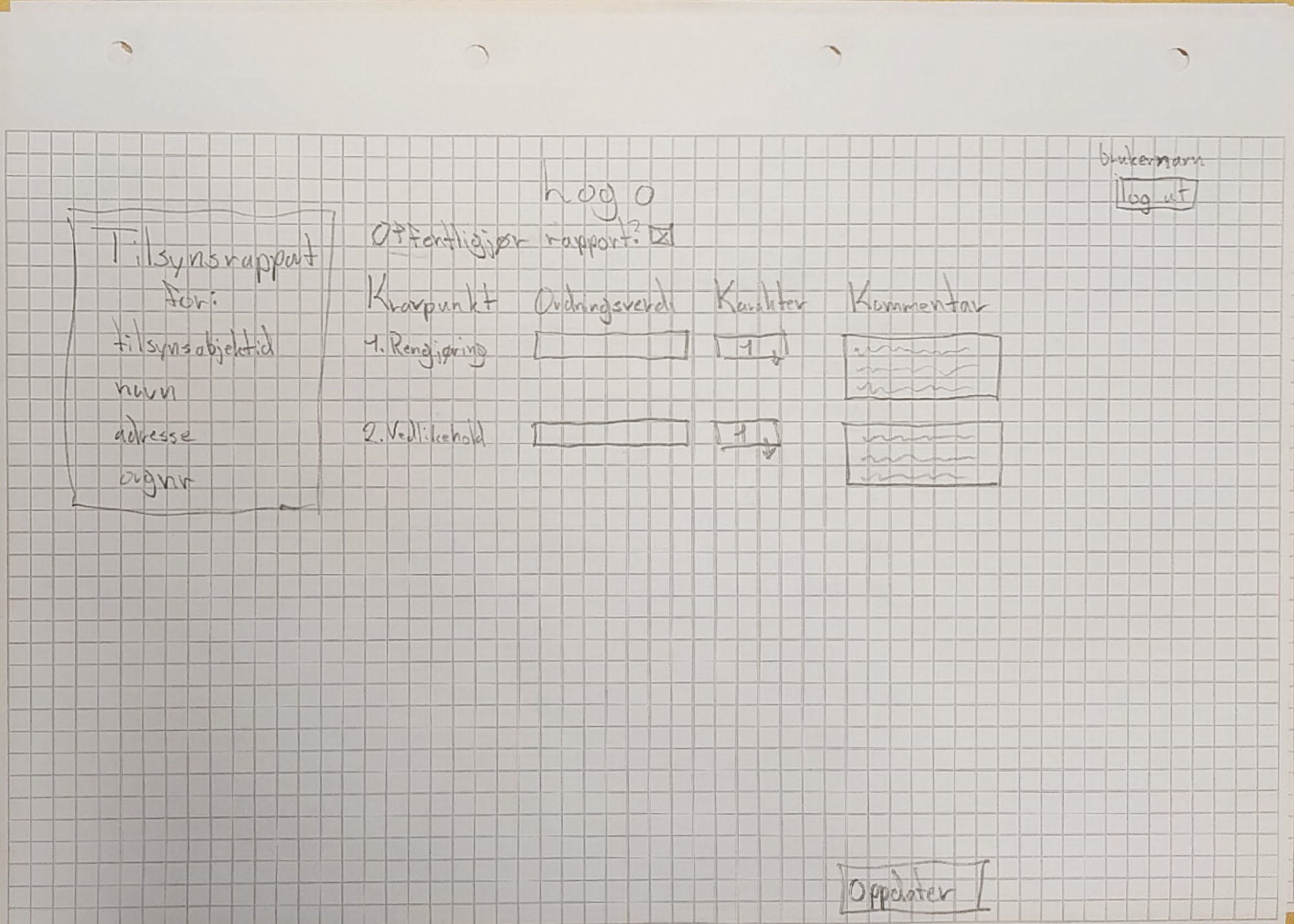


## Logget inn i administrasjonsverktøy

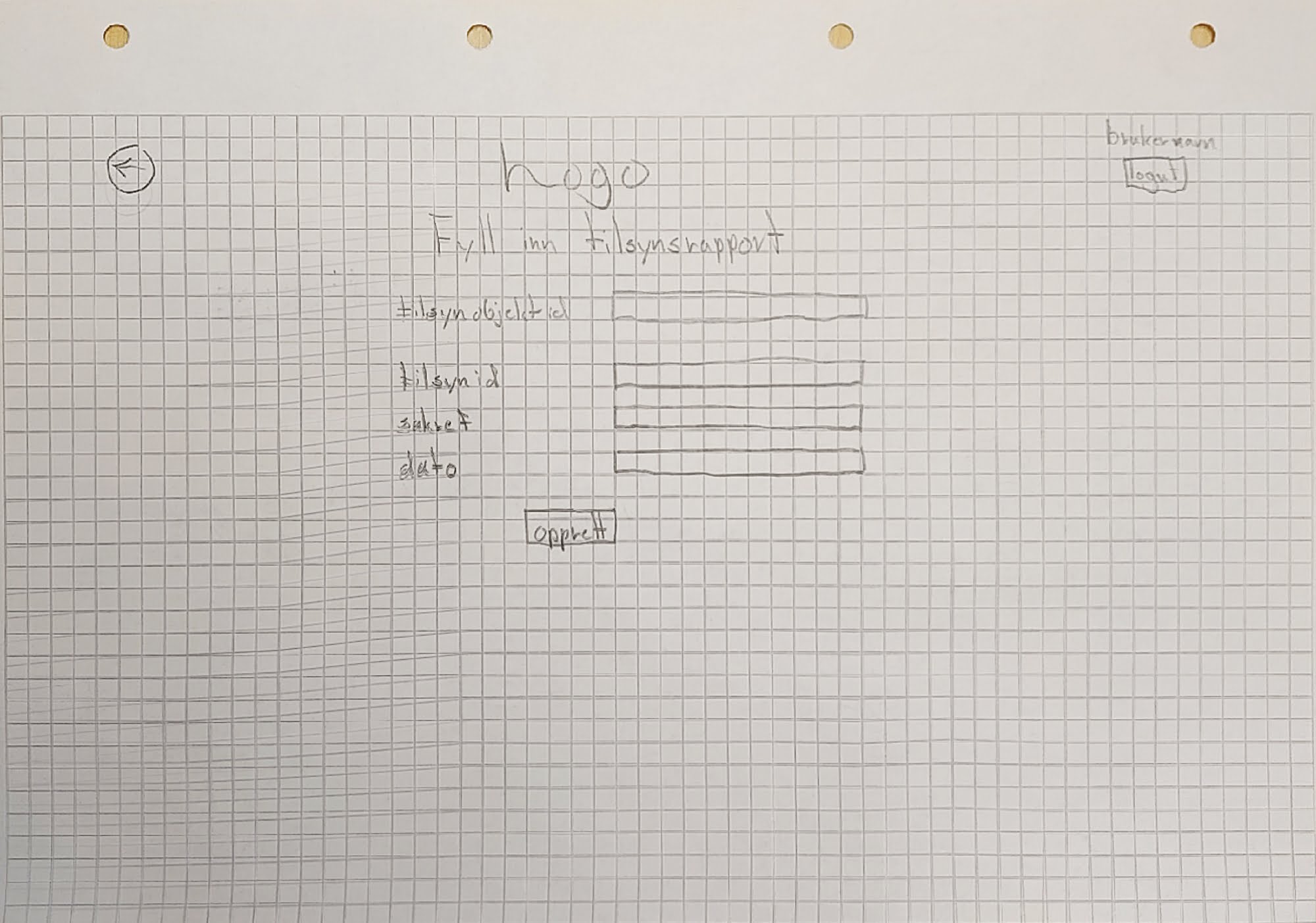


Administrasjonsverktøy





## Fyll inn ny tilsynsrapport



## Nye Ideer, tanker, annet:

### Triggere/rutiner

++ for å gjøre noe lagrede rutiner som ikke er lik det han gjør på tavla i leksjon 7.

++ for å lage triggere som ikke er de samme som er i leksjon 8. Tanker:

- Trigger som legger inn poststed først om man gjør insert på Restaurant-tabellen

- Legge alle navnene i poststed som store bokstaver, ha en Trigger som kjører UPPER() på siste innsatt rad, eller på raden som skal settes inn.

- Kan sette inn datostempling på nye insert i tilsynsID, eller vi kan lage "endret" kolonne på alle og legge til dato der, dersom det blir utført endring på den raden.