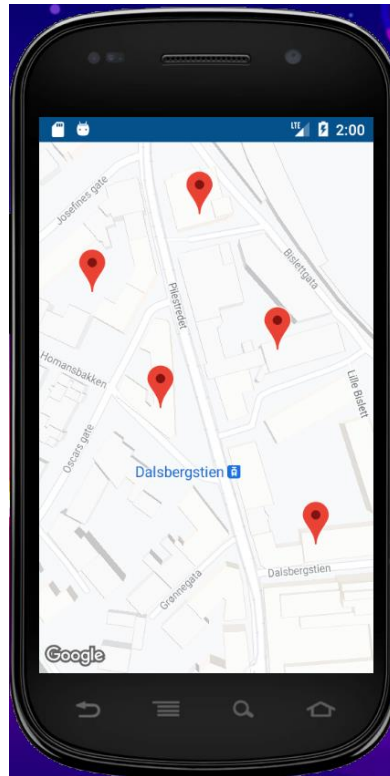


Kart av hus



Apputvikling mappeinnlevering rapport nr. 3 2021

Student nr: s341843

Oslomet

Innhold

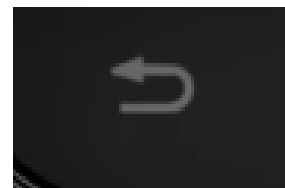
Generelt.....	3
Webservice.....	4
Hovedaktivitet.....	6
Registrering av et hus	8
Redigering av et hus	10
Slette registrerte hus	13
Avslutning.....	14
Kilder	14

Generelt

Denne applikasjonen var designet på en Nexus S med minimum API level 23 (Marshmallow). Der applikasjonen ble kodet mot *Androidx-kompatibilitetsbiblioteket*. Målet for programmet var å kunne registrere, redigere, hente og slette registrerte hus på en *webservice*.

Dette vil jeg gjennomgå mer detaljert senere i rapporten. Systemet burde også kunne vise husene på *Google Maps* i form av markører og vise informasjon om husene. For å holde søkelyset på kart funksjonen til applikasjonen, brukes det ikke aktiviteter bortsett fra hovedaktiviteten. Dermed brukes det f.eks. *DialogFragment*, for å utføre funksjoner som registrering.

Et av *Developer.Android*¹ sine rekommanderinger for design av en *Android* applikasjon, er å bruke telefonens innebygde tilbakeknapp. På grunn av dette, bruker applikasjonen ingen tilbakeknapper i UI-en. Dermed brukes telefonens innebygde tilbakeknapp for å navigere bakover i brukergrensesnittet.



Et annet av *Developer.Android*² anbefalinger er at bruken av aktiviteter og fragmenter, ikke burde lede til en stor *backstack*. Applikasjonen krever da alltid at brukeren maks må trykke tilbakeknappen 3 ganger før programmet avslutter. Der selvfølgelig fra hovedaktiviteten returneres til hjemskjermen.

Systemet bruker fargen blå som ifølge *Sensationalcolor*³ er en profesjonell farge og beroligende. Dette passer til målgruppen for applikasjonen som er eldre, og holder applikasjonen formell.

Programmet sender mange forespørsler mot for eksempel *Google*, *Geocode* eller mot *webservicen* for applikasjonen. Dette vil få telefonen til å henge igjen for å vente på en respons. Derfor bruker systemet *AsyncTask*, som lar telefonen håndtere forespørslene i bakgrunnen, slik at systemet ikke venter på et svar.

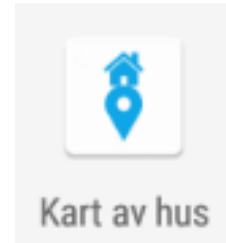
¹ <https://developer.android.com/docs/quality-guidelines/core-app-quality>

² <https://developer.android.com/guide/components/activities/tasks-and-back-stack>

³ <https://www.sensationalcolor.com/meaning-of-blue/>

Applikasjonen inkluderer en `strings.xml` fil, slik at det fremover vil være lettere å kunne støtte ulike språk. Dette er rekommandert av *Developer.Android*⁴ som mener at all tekst som vises i *Android* programmer bør være i filen.

En logo som kommuniserer med brukeren er også en viktig del av applikasjonen, ettersom den gir et overblikk over hva applikasjonen dreier seg om. Logoen kommuniserer dette med et bilde av et hus over en markør, der den primære fargen til applikasjonen er blå. Dette kan kommunisere til brukeren at applikasjonen bruker markører, som peker på lokasjonen til et hus. I tillegg gir det også brukeren innblikk i programmets primære farge.



Webservice

Et krav for oppgaven var at applikasjonen skulle bruke en *webservice*. *Webservicen* inneholder en tabell for alle registrerte hus. Denne tabellen er fundamental for lagring av brukerens registrerte hus. Tabellen inneholder også relevant informasjon etterspurt i oppgaven som adresse, latitude, longitude, beskrivelse og etasjer:

```
mysql> SELECT * FROM Hus;
```

id	adresse	latitude	longitude	beskrivelse	etasjer
1	hpp	0	0	hop	0
2	Bislettgata 6	59.9244	10.7324	En ti etasjes bygning i sentrum?!	10
3	Oscars gate 7B	59.9238	10.7301	Dette er en test	1

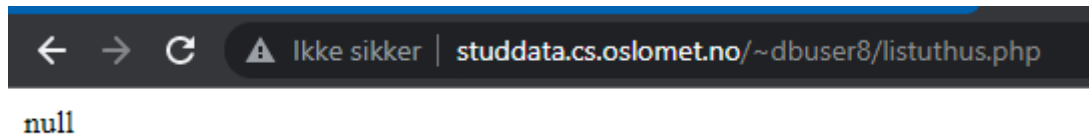
3 rows in set (0.01 sec)

Serveren for applikasjonen inneholder også PHP filer for å kommunisere med databasen utenfor serveren, som et API-grensesnitt. Systemet «Kart av hus» bruker dette for eventuelt hente alle registrerte hus, registrere, redigere og slette hus:

- listuthus.php
- registrerhus.php
- oppdaterhus.php
- deletehus.php

For listuthus.php returneres det et *JSONARRAY* av alle huser lagret i databasen:

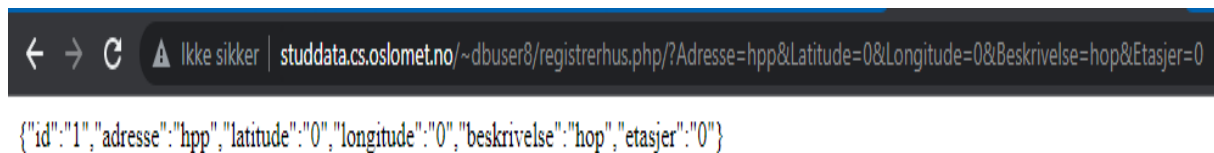
⁴ <https://developer.android.com/guide/topics/resources/localization.html>



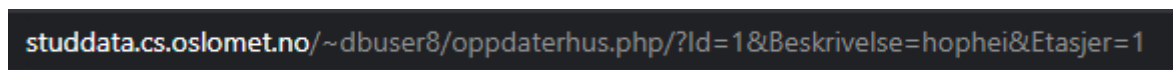
Dersom tabellen ikke er tom

```
[{"id": "1", "adresse": "hpp", "latitude": "0", "longitude": "0", "beskrivelse": "hop", "etasjer": "0"}]
```

Registrerhus.php lar applikasjonen registrere et nytt hus, dette gjøres ved å sende relevant informasjon gjennom lenken. Deretter når skriptet har fullført oppgaven og registrert huset, returneres det et *json object* av huset fra tabellen:



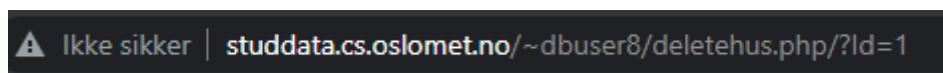
Json objektet kan dermed brukes til å redigere det registrerte huset med oppdaterhus.php. Dette gjennomføres ved at id-en i objektet ekstraheres som i dette tilfelle er 1. En kan med id-en øke etasjen fra 0 til 1 ved å sende inn ny informasjon om huset:



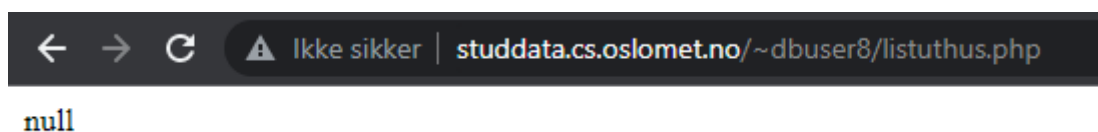
Det listuthus.php gir når

```
[{"id": "1", "adresse": "hpp", "latitude": "0", "longitude": "0", "beskrivelse": "hophei", "etasjer": "1"}]
```

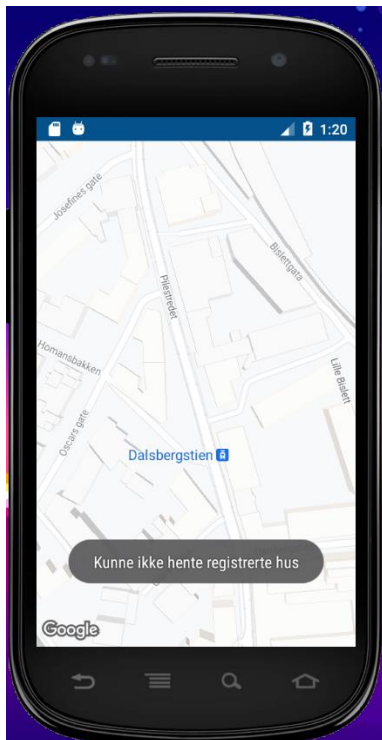
Id-en nevnt over kan også brukes til å slette det registrerte huset med deletehus.php. Dette vil føre til at huset blir fjernet fra tabellen, og dermed ikke bli listet ut av listuthus.php:



Nå eksisterer ikke huset med id 1:

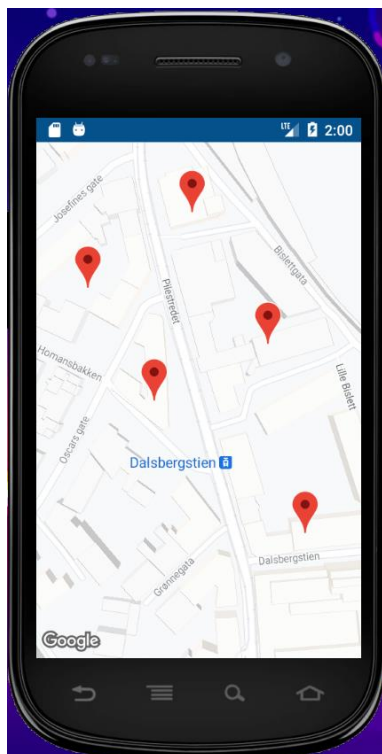


Hovedaktivitet



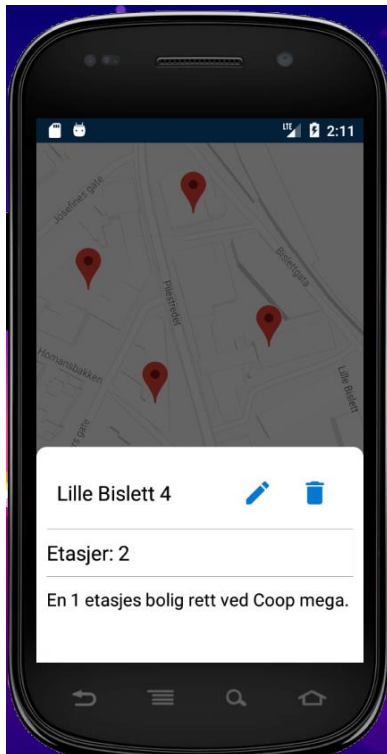
Når brukeren starter applikasjonen får de først opp *google maps*, deretter hentes det alle registrerte hus og markeres på kartet. På grunn av dette krever systemet dermed internetttilgang:

Dersom brukeren har internetttilgang:



Applikasjonen starter zoomet inn på Pilestredet nok til at kartet viser bygningene på kartet, slik at brukeren lettere kan se hvilke bygninger som har en markør. I tillegg til hvor de må trykke for å registrere deres ønskede hus.

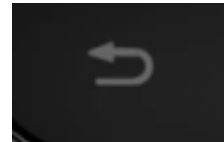
Programmet bruker også *google maps* sine innebygde markører, for å markere adressen til de registrerte husene. Årsaken til dette er for å unngå å forvirre brukeren med ulike markører og gjør det lettere for dem å forstå målet, kommunisert i logoen. I tillegg fjernes det også ekstra ikoner som «Bislet Bad og Trenning» og holder da fokuset på de registrert husene.



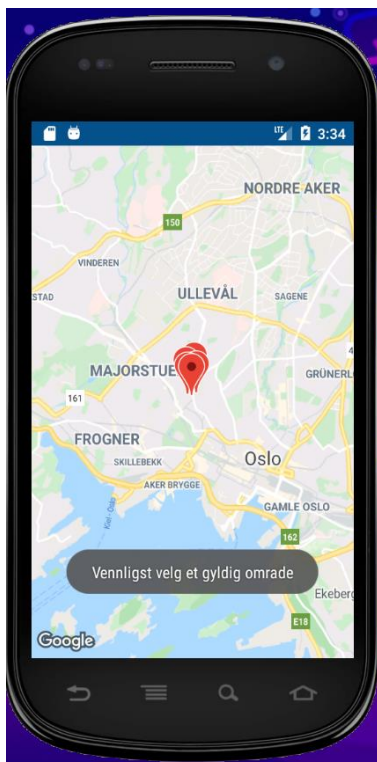
Hvis brukeren trykker på en eksisterende markør, vil de få opp følgende informasjon om huset markøren peker på:

Brukeren får øverst opp i *Fragmentet* adressen til huset, som i eksempelet over er «Lille Bislett 4». Ved siden av adressen eksisterer det to knapper med ikoner, mer om deres funksjon og betydning senere. Under dem vises det informasjon om byggets etasjer som er 2 og en beskrivelse av huset.

For designet av *BottomDialogFragment*, brukes det linjer over og under etasjer *TextViewen* for å separere feltene. Dette gir elementene separering i tillegg til mer plass. For knappene brukes det applikasjonens primære farge for å skille dem ut mer enn teksten, dette kan indikere at de er annerledes fra teksten



og er knapper som kan trykkes. Brukeren kan eventuelt navigere seg tilbake ved å trykke på de mørkere områdene av skjermen, eller med tilbakeknappen.



Når brukeren trykker på et område på kartet, sjekkes det med *Geocode* om latituden og longituden gitt, leder til en eksisterende adresse. Dersom det er en ugyldig adresse, får brukeren opp følgende feilmelding:

Registrering av et hus

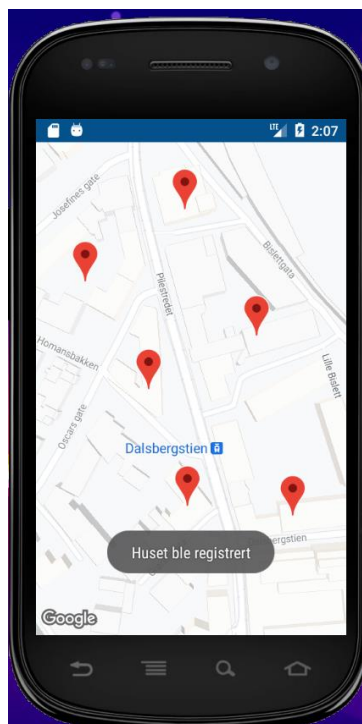
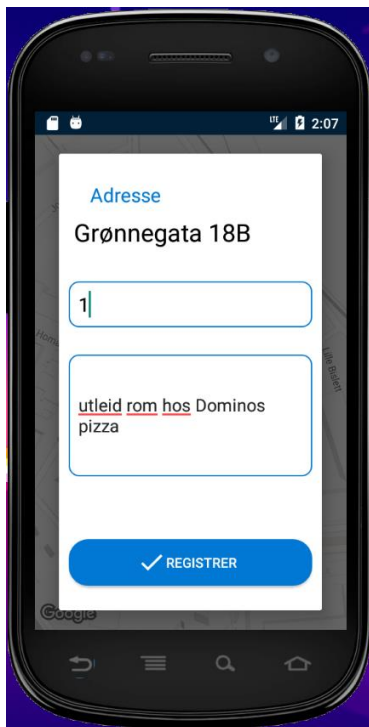


Dersom brukeren trykker på et gyldig område, burde *Geocode* returnere en adresse og brukeren få opp følgende

DialogFragment:

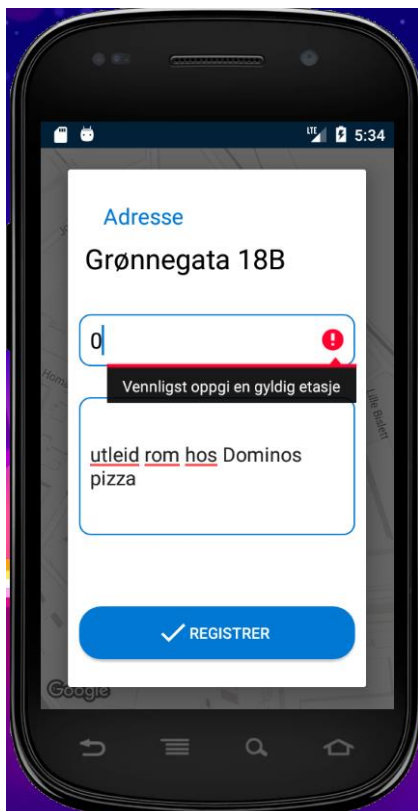
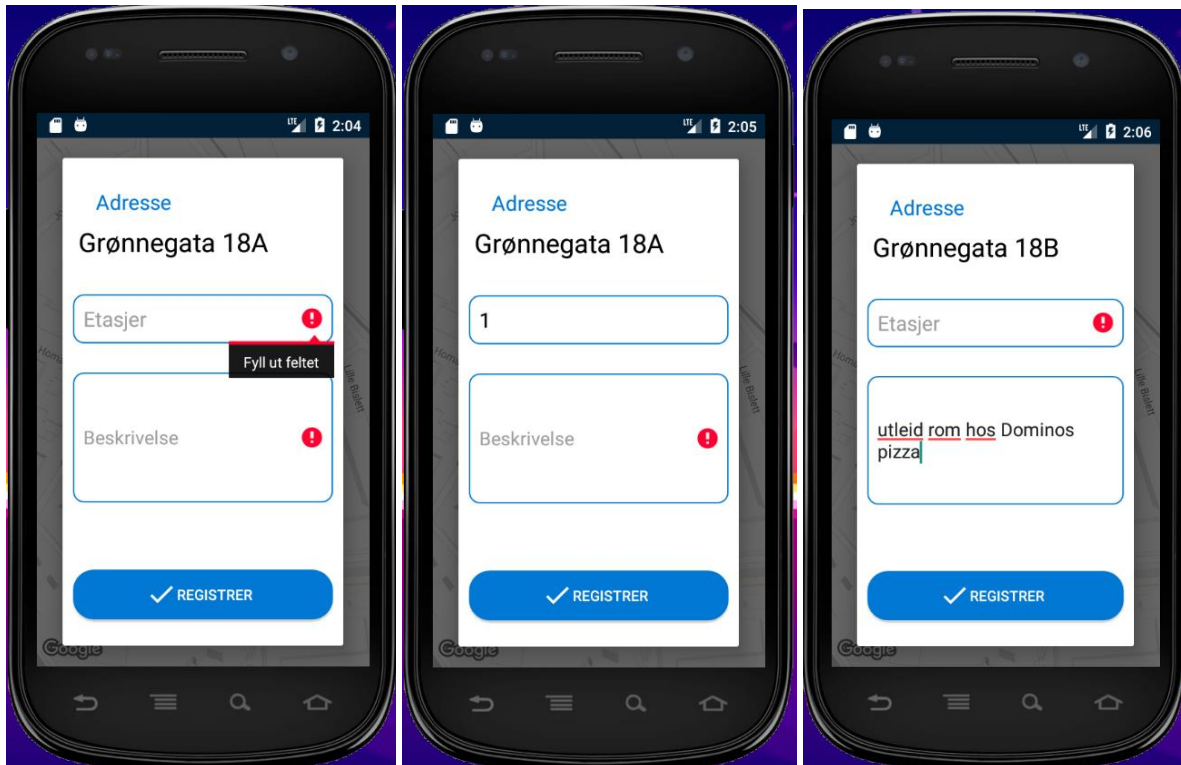
For registrerings dialog boksen, brukes det igjen den primære blå fargen rundt *EditTextViewene* for å vise brukeren området til elementet og hvor de eventuelt kan trykke. For adresse feltet brukes det ikke en firkant rundt elementet øverst i dialogboksen. Grunnen til dette, er for å kommunisere med brukeren at denne delen er fast og kan ikke endres. Feltet har også en *TextView* over den med teksten «Adresse». Dermed er brukeren er klar over at f.eks. «Grønnegata 18A» er adressen, og kan sikre seg at de har valgt deres korrekte adresse. En *EditTextViewene* bruker derimot *hint* for å vise brukeren hva

slags informasjon de krever.



Dersom brukeren fyller inn gyldig informasjon om boligen, altså fyller inn antall etasjer og en beskrivelse, kan brukeren trykke på registrerings knappen med hakk ikonet. Registreringen indikeres med at det sendes en toast og at dialogboksen forsvinner. Dette leder også til at en markør blir plassert som peker til det registrerte huset:

Dersom brukeren ikke har fylt ut feltene etasjer og beskrivelser, gis det en feilmelding gjennom *setError*:



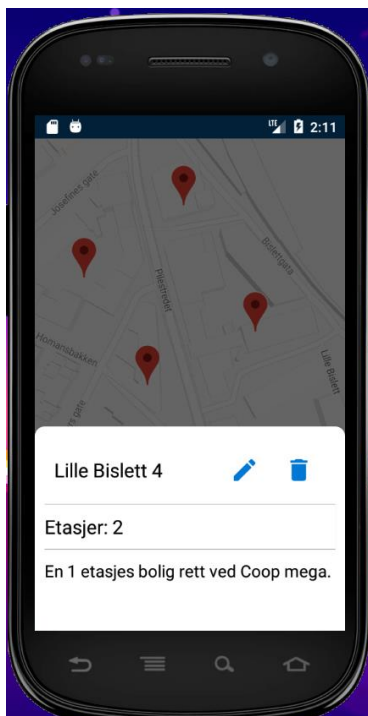
Dersom brukeren også oppgir en ugyldig etasje, som tallet «0», får de opp følgende feilmelding med *setError* metoden:



I tillegg brukes den samme blå fargen for registrerings knappen med hvit tekst og et ikon, slik at dersom brukeren ikke skjønner ikonet kan teksten gi kontekst til mening. Årsaken bak den hvite teksten på knappen, er for å ha en skarp nok kontrast mellom bakgrunnen til knappen, og teksten i tillegg til ikonet som er rekommandert av *Develop.Android*⁵.

Årsaken bak bruken *setError* er fordi det viser brukeren hvor det mangles informasjon og fargen rød brukes. Ifølge *Bourncreative*⁶ er rød en farge som tar brukeres oppmerksomhet og kan signalisere en advarsel til brukeren.

Redigering av et hus



For redigering av et registrert hus må brukeren på kartet, trykke på deres foretrukne markør for det ønskede hus:

Derfra kan brukeren trykke på rediger knappen fra *Fragmentet* i form av en blyant ikon for å endre informasjonen om huset. Brukeren får da opp følgende dialogboks:



Her kan brukeren se adressen, etasjene og beskrivelsen registrert om huset. Brukeren får redigere antall etasjene registrert om huset beskrivelse. Derimot er adressen og lokasjonen fast og kan ikke endres.

⁵ <https://developer.android.com/docs/quality-guidelines/core-app-quality>

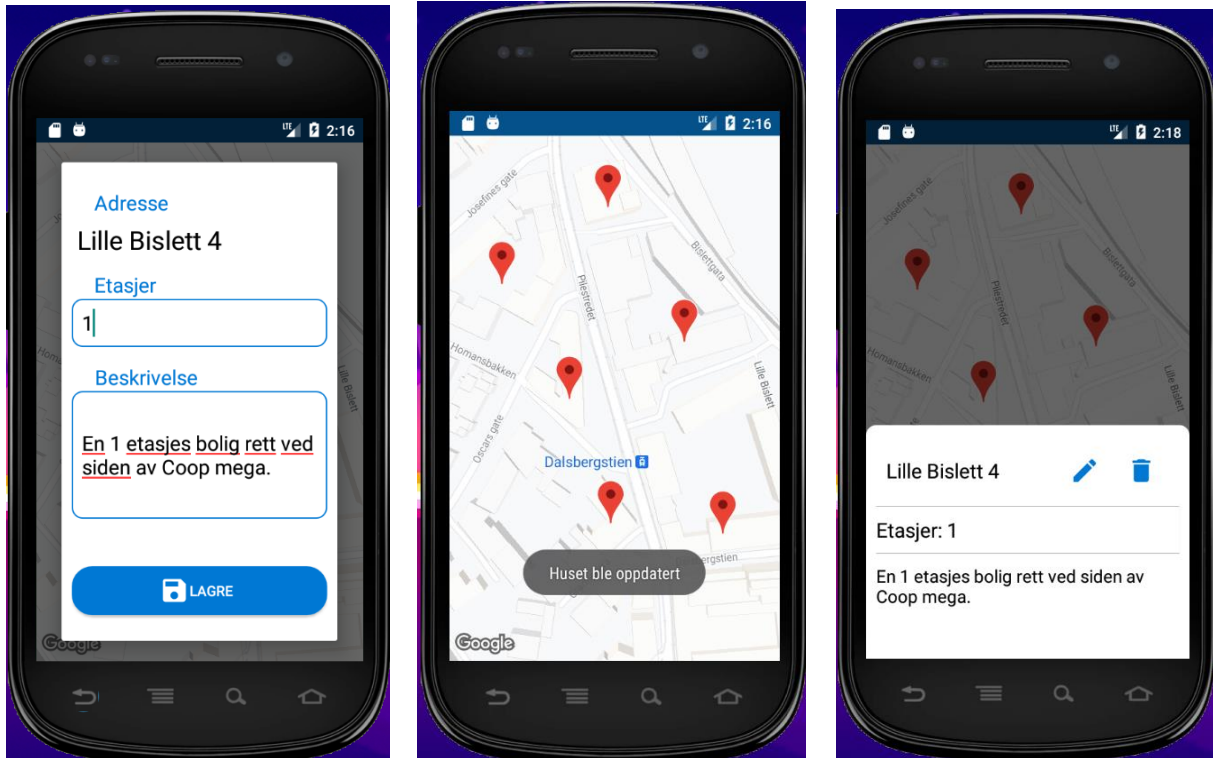
⁶ <https://www.bourncreative.com/meaning-of-the-color-red/>

DialogFragmentet har også validering, der endringene ikke blir lagret. Dersom brukeren etterlater informasjon som husets etasjer eller beskrivelse. Programmet bruker *setError* for å vise hvilke felt som ikke er gyldige og krever informasjon:



Valideringen inkluderer også inputen for etasje feltet for å hindre verdier som 0 å bli registrert. Dersom brukeren oppgir en slik ugyldig etasje, får det opp følgende feilmelding:

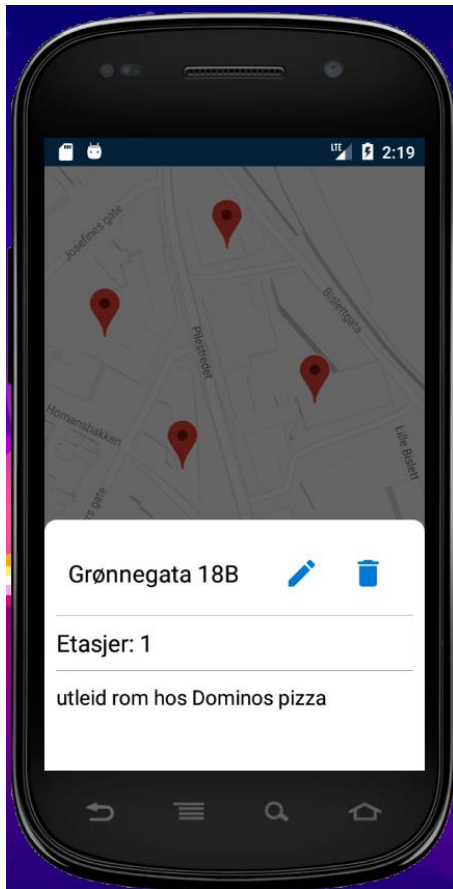
Hvis brukeren fyller inn alle feltene med lovlig informasjon vil endringen av huset bli registrert. Applikasjonen viser dette ved at *DialogFragmentet* fjernes og brukeren får en toast. For eksempel endrer vi etasjen til 1 slik at det stemmer med beskrivelsen:



For redigerings dialogboksen, brukes det igjen den primære fargen blå rundt *EditTextViewene* for å vise brukeren området til elementet og hvor det kan trykkes. Som i registrerings siden har adresse feltet ikke noen grense rundt objektet i tillegg har teksten «adresse» over, slik at brukeren blir informert om hvilke felt som er interaktiv. Derimot brukes det ikke *hint* for *EditTextViewene*, men som adressen *TextViews* slik at brukeren enkelt kan vite hva slags informasjon hvert felt krever.

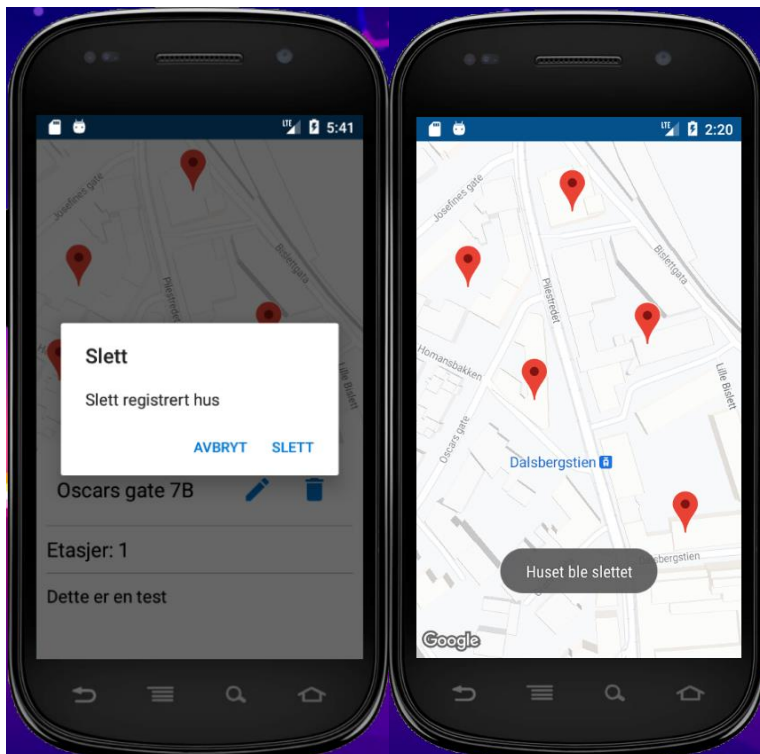
Det brukes igjen også den samme blå fargen for redigerings knappen med hvit tekst og et ikon, slik at brukeren ikke trenger å skjønne ikonet i sammenheng med knappens funksjonalitet. Der tekst fargen i tillegg til fargen på ikonet igjen er hvit for å skape kontrast. I redigerings dialogboksen brukes det *setError* metoden, som tidligere nevnt informerer brukeren hva de mangler å fylle ut og hvordan det visuelt kommuniseres.

Slette registrerte hus

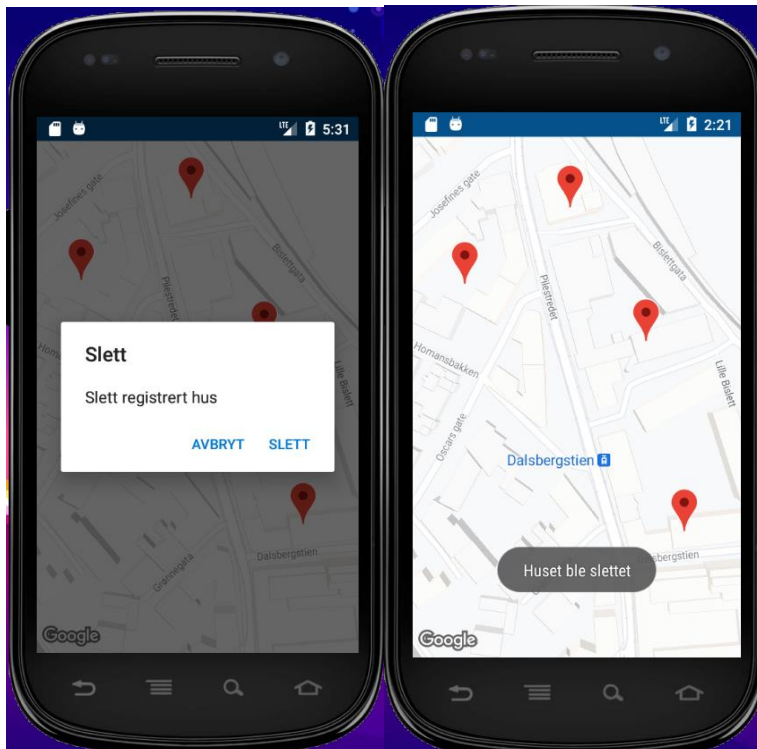


For å slette et registrert hus må brukeren igjen trykke på en markør knyttet til det ønskede huset:

Fra der kan de trykke på søppelbøtte ikonet som henviser til en slette knapp. Dersom brukeren trykker på knappen, får de opp følgende *AlertDialog*:



Deretter får brukeren et valg for å bekrefte slettingen, der «AVBRYT» avbryter slettingen av huset, mens «SLETT» sletter det valgte huset. Dersom brukeren velger «SLETT» får de opp en toast om at huset ble fjernet, dette leder også til at markøren til huset fjernes fra kartet:



Brukeren kan også fjerne et av de registrerte husene ved å holde ned på markøren til det ønskede huset. Dette vil igjen vise brukeren en *AlertDialog*, som igjen spør dem om bekreftelsen til slettingen av huset og markøren:

Avslutning

For å konkludere dette prosjektet, har applikasjonen klart å oppfylle kravspesifikasjonene satt av oppgaven. Hvor målet var å lage et program som lagrer registrerte hus. De registrerte husene blir da visuelt visst frem på *Google maps*, i form av markører. Disse kan da redigeres, slettes og i tillegg kan det registreres flere gjennom et brukergrensesnitt.

Et av utfordringene som kom frem under utviklingen, var implementering av funksjonaliteten for sletting av et hus med *longclick*. Problemet oppsto ved at *Google maps* modulen ikke inneholder en innebygd metode for *longclick*. På grunn av dette måtte applikasjonen bruke en form av dette gjennom markørens drag metoder. Et annet problem som oppstod under utviklingen, var *Geocode*'s stabilitet. Der av og til *Geocode* ikke returnere gateadressen for et gyldig område.

Kilder

- <https://developer.android.com/docs/quality-guidelines/core-app-quality>
- <https://developer.android.com/guide/components/activities/tasks-and-back-stack>
- <https://www.sensationalcolor.com/meaning-of-blue/>

- <https://www.bourncreative.com/meaning-of-the-color-red/>
- <https://developer.android.com/guide/topics/resources/localization.html>