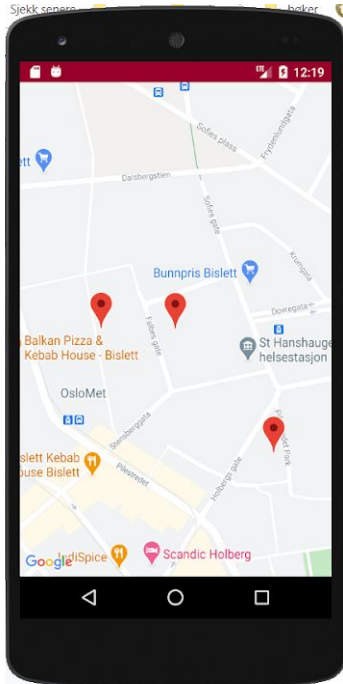


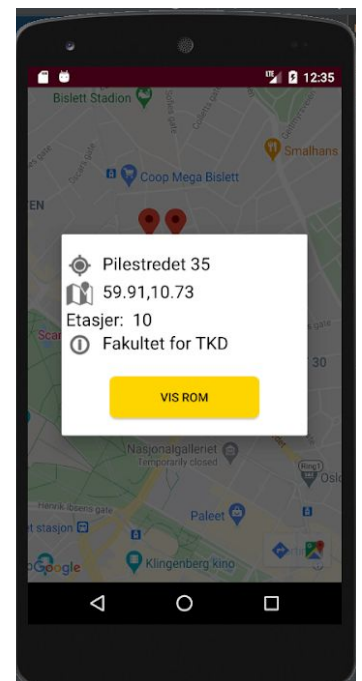
Mappe 3 rapport

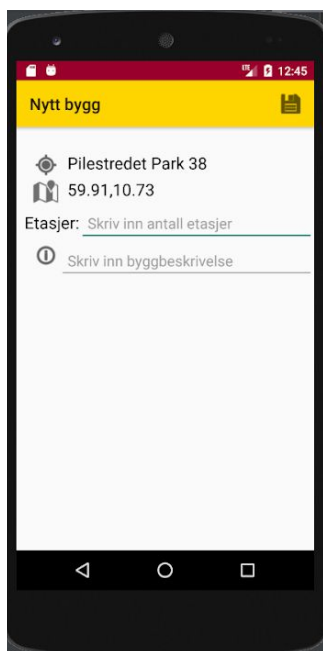
Av Lasse Apalnes Pedersen (s333761) og Fredrik Rikheim (s328084)



Når appen åpnes vises kartet, som zoomes inn på Pilestredet. Markører blir lagt til når brukeren trykker på et punkt på kartet og legger inn korrekte data i et nytt bygg. For å bevare markørerne når man går inn og ut av kartaktiviteten og avslutter appen, legger vi til markørene i SharedPreferences og setter markører på nytt i onMapReady-funksjonen.

Til høyre er et bilde av dialogfragmentet som dukker opp når brukeren trykker på en markør, det vil si et bygg som er lagt til på kartet. Her kommer det opp informasjon om bygget, adresse, koordinater (formatert slik at kun to desimaler vises), antall etasjer og beskrivelse av bygget, samt et knapp hvor brukeren kan gå inn og se på rommene til bygget. For å vise frem informasjonen har vi brukt ikoner på de feltene vi følte det fantes ikoner som var intuitive, som for eksempel et lokasjon-ikon for adresse. På antall etasjer syntes vi ikke det fantes et intuitivt ikon og valgte heller å skrive "Etasjer" med tekst.





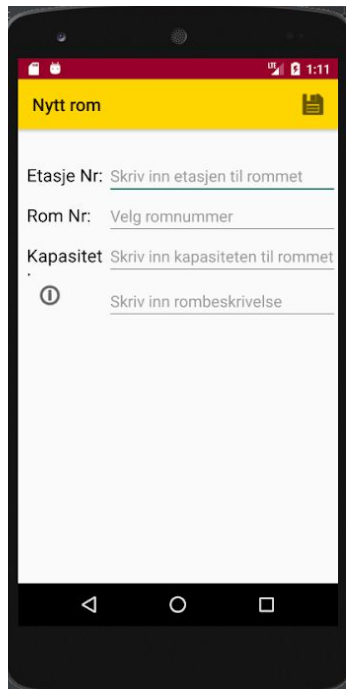
Til venstre er et bilde av aktiviteten brukeren får opp når man trykker på et punkt på kartet hvor man ønsker å legge til et bygg. Ikonbruk er forklart i avsnittet over. Adresse hentes av en Geocoder i kildekoden og vises direkte til brukeren, mens brukeren selv fyller inn antall etasjer og beskrivelse. Antall etasjer brukeren kan legge inn må være mellom 1 og 20, da vi anser dette for å være en rimelig begrensning når det gjelder bygg på OsloMet. Actionbaren i alle aktiviteter har vi valgt å farge gul, lik gulfargen OsloMet bruker. Statusbaren er farget i en burgunderfarge vi har hentet fra OsloMet sine nettsider. Fargene er med andre ord valgt for å gjøre det klart for brukeren at appen omhandler bygg, rom og romreservasjoner på OsloMet.

Man lagrer bygget når man trykker på lagre-ikonet i actionbaren, et ikon vi mener er intuitivt for brukere for lagring.

Til høyre er et bilde av aktiviteten brukeren kommer til når hen trykker på "Vis rom"-knappen i dialogfragmentet som dukker opp når man trykker på en markør på kartet. Her vises øverst informasjon om bygget og under listes det opp rom registrert i bygget. Vi har valgt å vise frem romnummer, etasjenummer og kapasitet, men ikke beskrivelse om rommene i bygget. Man kan legge til rom når man trykker på pluss-ikonet i actionbaren, et ikon vi mener er intuitivt for brukere for å legge til. Vi har valgt å legge lagre- og pluss-ikon i actionbaren, slik at eventuelle knapper nederst på skjermbildet ikke blir borte, avhengig av hvilken enhet man bruker appen på. Når det kommer til informasjonsvisning av rom har vi valgt ren tekst, og ingen ikoner. Vi tror ikke det finnes intuitive ikoner for romnummer, etasjenummer og kapasitet, og har derfor valgt å vise disse med bare tekst.

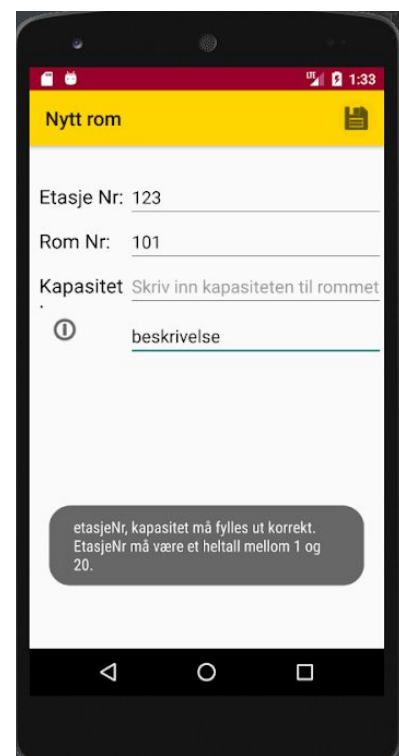


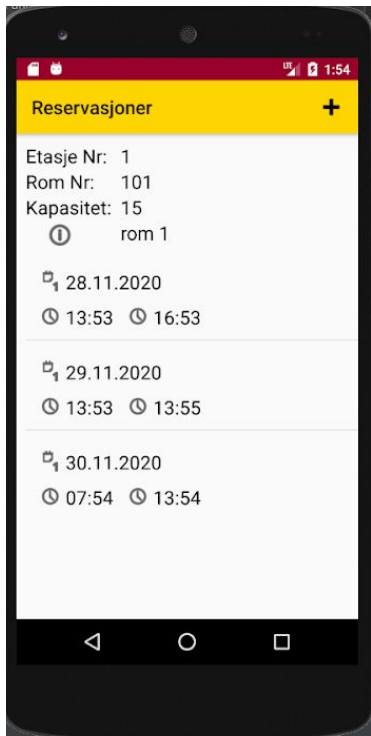
Rommene i listen er sortert på etasjenummer.



Til venstre er et bilde av aktiviteten brukeren kommer til dersom hen trykker på pluss-ikonet i aktiviteten der rommene i et bygg vises frem. Her sjekker vi om etasjenummer er et tall mellom 0 og 20 og at det ikke er større enn antall etasjer i bygget, og kapasitet er et tall over 0. Om ikke så sendes det en toast med tilbakemelding til brukeren om at disse feltene må fylles inn korrekt. På romnummer sjekkes det kun om den er fylt ut, fordi vi går ut ifra at romnummer kan ha bokstaver eller andre tegn i seg, som for eksempel "A101". Ikonbruk er beskrevet i avsnittet over, og i denne aktiviteten har vi valgt å ha med informasjon-ikonet om beskrivelse av rommet, slik at dette ikonet gjenbrukes gjennom hele appen.

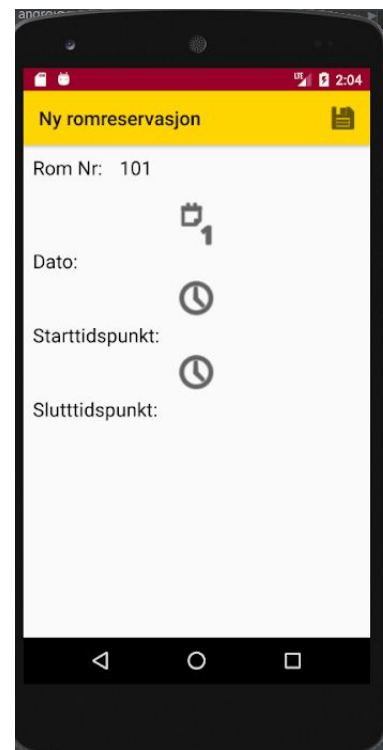
Til høyre er et bilde av et eksempel på en toast hvor brukeren får tilbakemelding om at feltene må fylles inn korrekt. Brukeren har skrevet inn et etasjenummer som er over 20, og har latt vær å fylle inn kapasiteten. Brukeren får da beskjed om at disse feltene må fylles ut korrekt, samt mer spesifikk informasjon om hvordan feltet som er fylt inn feil skal fylles inn korrekt.





Til venstre er et bilde av aktiviteten brukeren kommer til når hen trykker på et rom i aktiviteten hvor alle rommene i et bygg vises frem. Her vises øverst informasjonen om rommet, så listes alle reservasjonene som er registrert for dette rommet. Informasjonen om reservasjonene inkluderer datoen på den øverste linjen, for så å vise starttidspunkt og sluttidspunkt for reservasjonen på neste linje. Her har vi valgt å bruke ikoner, da vi føler kalender-ikon og klokke-ikon er intuitive ikoner for dato og klokkeslett. Vi bruker samme ikon for start- og sluttidspunkt, men vi føler likevel det er intuitivt at starttidspunktet kommer først, for så sluttidspunktet, og at dette ikke er til forvirring for brukeren. I listen har vi valgt å sortere reservasjonene på dato, for så starttidspunkt.

Til høyre et et bilde av aktiviteten brukeren kommer til om hen trykker på pluss-ikonet i aktiviteten hvor det vises frem reservasjoner på et rom. Øverst viser vi hvilket rom hvor brukeren har valgt å legge til en reservasjon. Her har vi valgt å bruke bildeknapper med ikoner som er beskrevet i avsnittet over. Når brukeren trykker på kalender-knappen åpnes et datepickerfragment hvor vi kun gjør det mulig å velge datoer fra og med dagens dato. Når brukeren trykker på en av klokke-knappene åpnes et timepickerfragment. Under knappene har vi tekstbokser som fylles ut når brukeren velger dato og tid, slik at bruker kan se dette i klartekst før reservasjonen lagres ved å trykke på lagre-ikonet i actionbaren. Dersom brukeren fyller inn en



dato og tid hvor rommet allerede er booket eller velger et starttidspunkt samtidig eller etter sluttidspunkt får brukeren tilbakemelding om dette ved en toast.

Vi har konsekvent valgt å ikke legge til egne tilbake-knapper, og går ut ifra at brukeren brukeren benytter seg av mobilen sin egen tilbake-knapp. I aktiviteten hvor man legger til et nytt bygg har vi overskrevet funksjonaliteten til denne tilbake-knappen slik at en markør ikke legges til på kartet dersom brukeren trykker på tilbake-knappen.

Når det kommer til innlegging av json fra webserver har valgt å lage tre php-script på webserveren for innleggelse av data, et for bygg, ett for rom og ett for romreservasjon. I koden har vi da en egen SendJSON-klasse som utfører selve overføringen av json-objekter til webserveren, mens i aktivitetsklassene hvor vi legger til enten bygg, rom eller reservasjon har vi en sendJson-funksjon hvor vi tar inn variablene som skal sendes til webserver og formaterer url-en og sender denne inn i SendJSON-objektet sin execute-funksjon.

Når det kommer til uthenting av json fra webserver har vi valgt å lage fire php-script på webserveren for uthenting av data, ett for å hente ut et spesifikt bygg, ett for å hente ut et spesifikt rom, ett for å hente ut alle rom, og ett for å hente ut alle reservasjoner. I koden har vi valgt å lage getJson-klasser som vi legger inn i de klassene der vi trenger de. Her utføres selve overføringen av data fra webserveren, og i postExecute-funksjonen formaterer vi json-objektene basert på hva vi ønsker å gjøre med de. For å begrense antallet asynkrone kall på json-funksjoner har vi valgt å benytte oss i stor grad av putExtras- og getExtras-funksjonene til intenter for overføring av variabler mellom aktiviteter.

Appikonet vårt er en mobil som viser et kart med en markør på, og fargen er i hovedsak gul. Dette føler vi representerer appens funksjon godt, da vi benytter oss av en kartaktivitet for å legge til bygg (markører) på OsloMet (som benytter seg av den tidligere nevnte gulfargen).