

App1

Har gjort spill logikken helt uavhengig av GUI, slik at spillet skal kunne brukes med hvilket som helst GUI. Det er også mulig å velge brettstørrelse og hvor mange på rad man trenger for å vinne. (Dessverre fungerer det ikke å velge antall rader for å vinne siden jeg hadde en bug i denne funksjonaliteten og når jeg skrev den om tenkte jeg meg ikke helt om. Derfor ble den til slutt låst til 3 på rad, selv i 5x5 hvor det er umulig å tape hvis man starter. Koden som tar høyde for hvor mange på rad man vil ha er kommentert ut og ligger i koden. Denne fungerer, med unntak av at den ikke sjekker alle diagonaler i større brett. Rakk ikke fikse traverseringen her.)

Dette gjør at det bare er å velge layout og brettstørrelse når man starter spillet. For å demonstrere dette har jeg laget 2 forskjellige brett. Jeg valgte å lage de i form av layouts og ikke som fragments for å demonstrere at logikken ikke er noe annerledes bare layout. Planen var å lage layout dynamisk programmatisk, men fikk dessverre ikke tid til å regne på størrelsesforhold på brikkene for å få det riktig for en dynamisk løsning. Det ville vært bedre praksis å bruke fragments enn å bytte layout basert på størrelse, men lot det være slik for demonstrasjon ettersom jeg ikke rakk å gjøre det mer dynamisk.

Jeg lagde unit tests når jeg lagde spill logikken og kodet i en variant av «Test first». Dessverre glemte jeg disse når jeg endret på spill logikken nylig fordi Android studio ikke gir noen varsel om at testene feiler. Lagde spill-logikken en stund før jeg fortsatte på resten. Det vil si at testene er for den gamle logikken.

App2

Hotellapplikasjonen er laget for å forenkle sjekk-in på hoteller. Man lager en bruker og skanner QR-koden i resepsjonen så kobles romnummer og nøkkel til brukeren din. Rom nummer kan oppdateres i realtime slik at hotellet kan raskt ordne opp i *romfeil* etc. Dataen lagres også offline så du må ikke ha tilgang til nett for å komme deg inn på rommet ditt når du har registrert brukeren din.

I kartet har jeg valgt å ta data fra firebase i stedet for å bruke Google Maps sitt *places* api. Det er 2 grunner til dette. 1. Demonstrere bruk av firebase som eksternt api, og gjøre at hotellet har kontroll over hvilke restauranter de vil promotere. På den måten kan de ha avtale med de forskjellige restaurantene rundt om gode priser for hotellgjestene for eksempel.

På Login siden har jeg en Activity som går videre til Main Activity i stedet for fragment fordi det fungerer dårlig med brukerautentisering fra firebase i et fragment. Dessuten ville jeg skille Login fra resten av applikasjonen.

I firebase er det duplikasjon av data for bruker og reservasjoner, dette er ikke dårlig praksis ifølge Firebase dokumentasjonen. Det er viktigere å holde treet så flatt som mulig for å hindre å plukke med seg for mye data når man skal hente ut data i et blad på treet.

Json for firebase lagt ved.

App1

Tic Tac Toe

Forsiden på Tic tac toe applikasjonen her kan man velge mellom 3x3 eller 5x5, planen var å kunne velge tall selv før man spiller, men det var alt for vanskelig å få til å skalere dynamisk, så valgte til slutt bare 3 og 5.



Welcome to Tic Tac Toe!

Write the name of the two players and start the game!

Player 1

Player 2

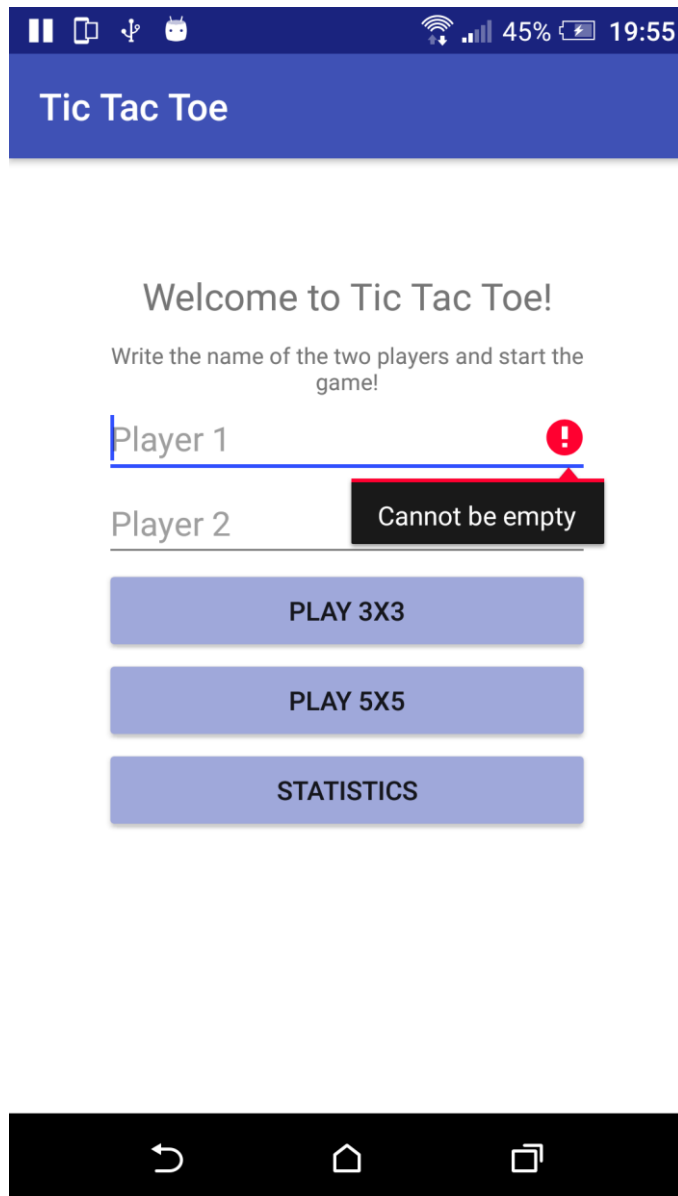
PLAY 3X3

PLAY 5X5

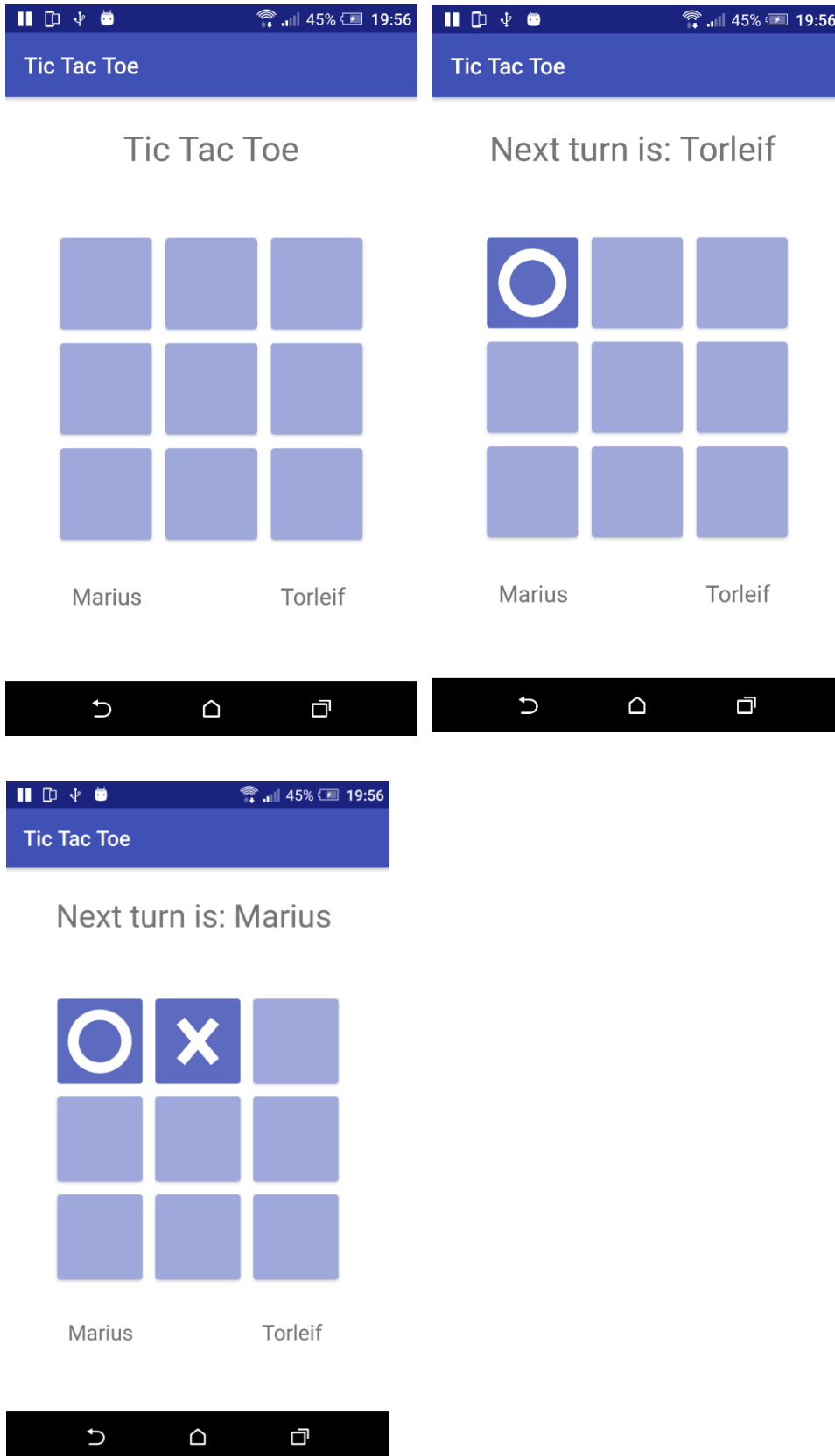
STATISTICS



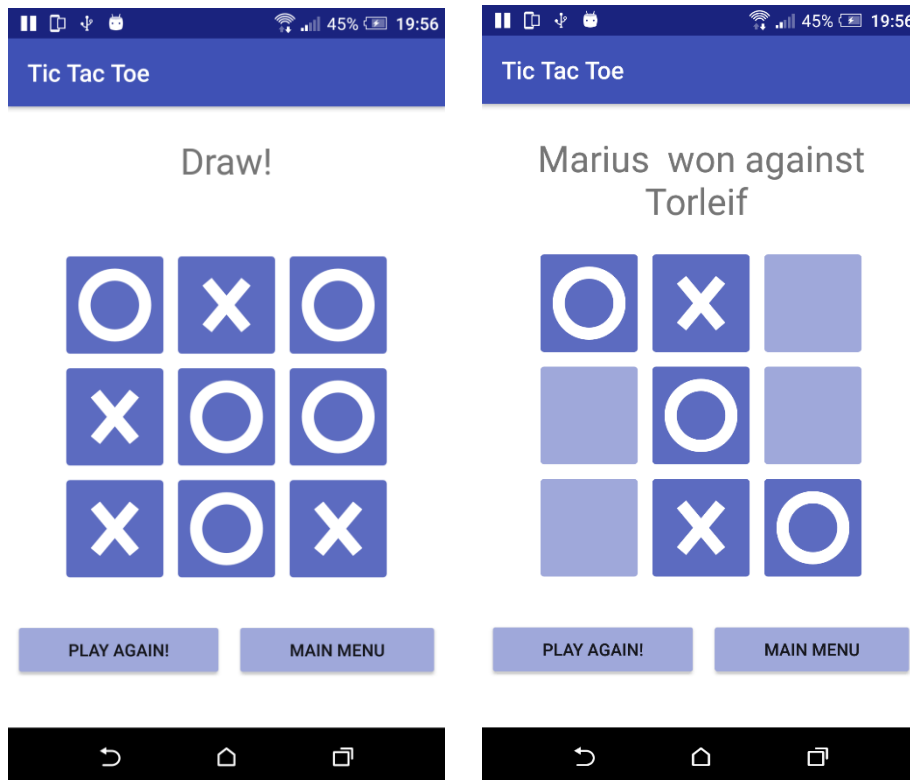
Enkel sjekk for å sikre at spillernavnene ikke er tomme, eller mindre enn 2 karakterer. For at det ikke skal se veldig dumt ut i tabellen. Bruker 2 karaterer slik at de som bare har 2 navn skal kunne sette initialene sine. Det kan blir krasj i statistikken om flere velger samme navn, men ettersom det er en bruker som har appen på sin egen telefon er det ikke så sansynlig at like navn på forskjellige personer blir valgt uansett.



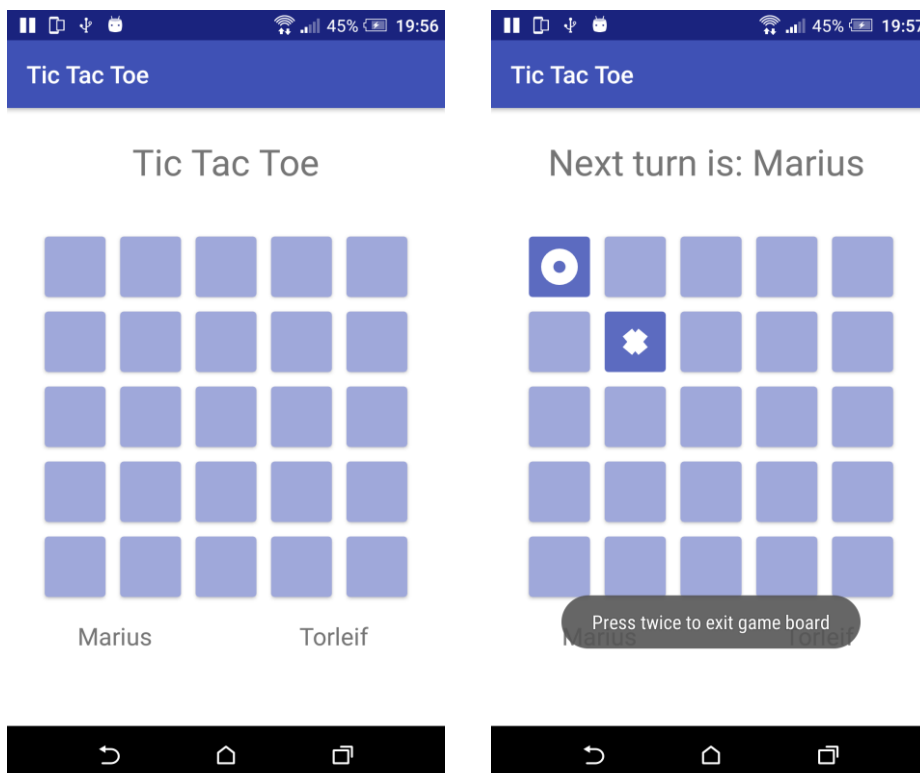
Har valgt et enkel utseende, med farger fra Google sin material design farge palett. Sirkelen og krysset er laget med drawables og er rene og enkle. Kunne kanskje valgt en accent farge på sirklene og kryssene, men følte det var finere å ha en rolig hvit farge. Spillet sier også i fra hvem sin tur det er.



Når spillet er over får man vite hvem som har vunnet, eller om det ble uavgjort. I tillegg dukker det opp alternativer for å gå tilbake til menyen eller spille igjen.

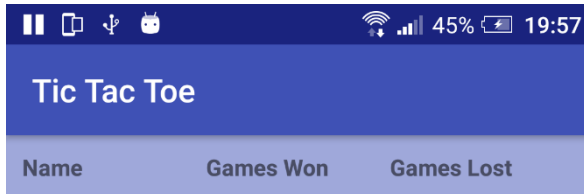


Slik ser spillet ut med 5x5. På begge layoutene må man klikke 2 ganger innen 2 sekunder for å gå ut av spillet slik at man ikke ved uhell skal nullstille brettet.



På statistikk-siden har jeg valgt å ta med antall spill man har tapt også, men den er sortert etter antall vunnet slik som oppgaven sier. Her har jeg brukt tabell layout istedenfor ListView, har egentlig ikke noen god grunn for det, tenkte å endre til Listview etter hvert siden det hadde vært mer fornuftig her, men glemte det rett og slett. List view med custom adapter er brukt i app2 derimot.

På forsiden er det også en toast som forteller at man må klikke 2 ganger for å gå ut av appen. Det er alltid fryktelig irriterende å lukke en app ved uhell ved å komme borti tilbakeknappen.



Name	Games Won	Games Lost
Marius	1	0
Torleif	0	1

Marius	1	0
Torleif	0	1



Welcome to Tic Tac Toe!

Write the name of the two players and start the game!

Marius

Torleif

PLAY 3X3

PLAY 5X5

STATISTICS

Press twice to exit application



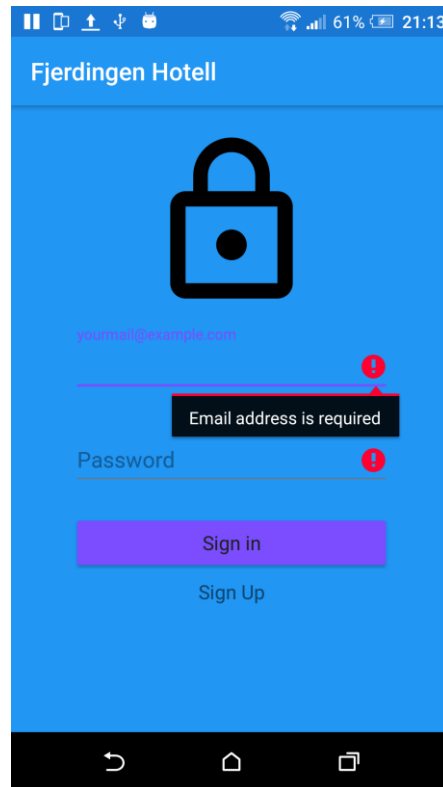
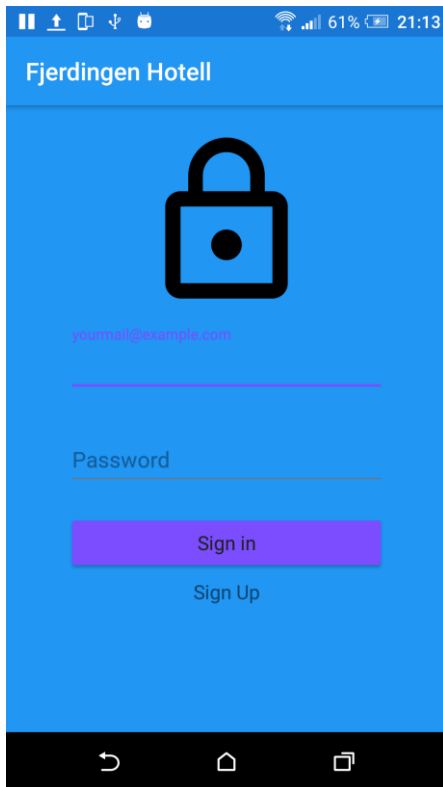
App2

Hotell applikasjon

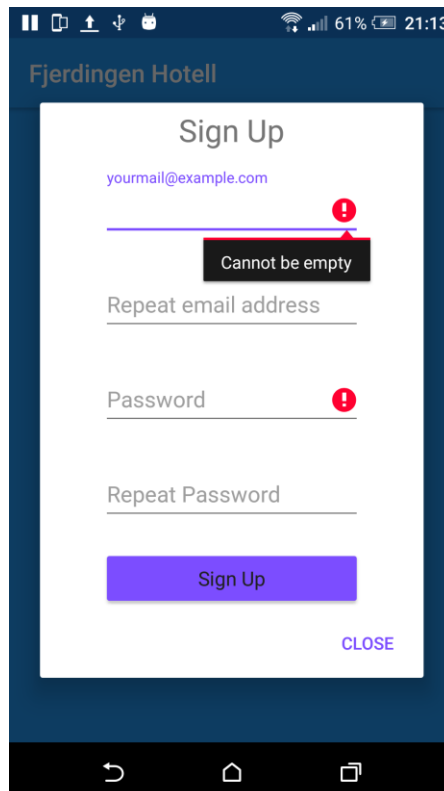
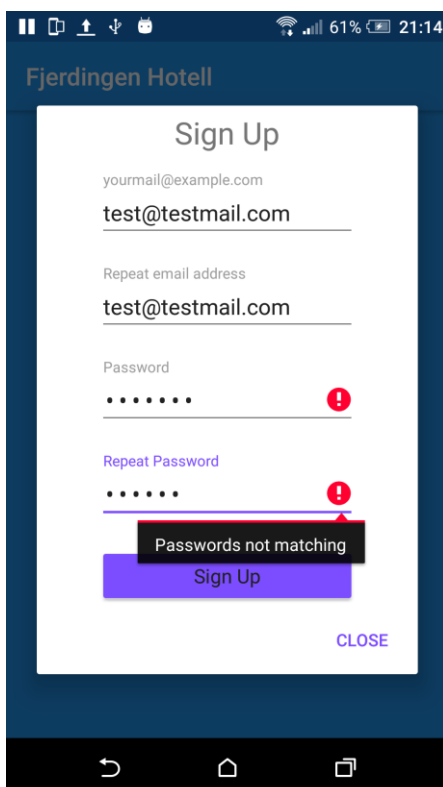
Her har vi login siden til hotellapplikasjonen. På denne appen har jeg prøvd å følge material design med tanke på farger ved å bruke 500 som hovedfarge, og en annen farge for accent. Jeg fikk aldri dette til å bli så fint som jeg håpet, men rullet med denne kombinasjonen til slutt. Den er ikke så fin på login skjermen hvor det ble for mye farge, men fungerer godt i UI i resten av applikasjonen

Deep Purple	Blue
500 #673AB7	500 #2196F3
50 #EDE7F6	50 #E3F2FD
100 #D1C4E9	100 #BBDEFB
200 #B39DDB	200 #90CAF9
300 #9575CD	300 #64B5F6
400 #7E57C2	400 #42A5F5
500 #673AB7	500 #2196F3
600 #5E35B1	600 #1E88E5
700 #512DA8	700 #1976D2
800 #4527A0	800 #1565C0
900 #311B92	900 #0D47A1
A100 #B388FF	
A200 #7C4DFF	
A400 #651FFF	
A700 #6200EA	

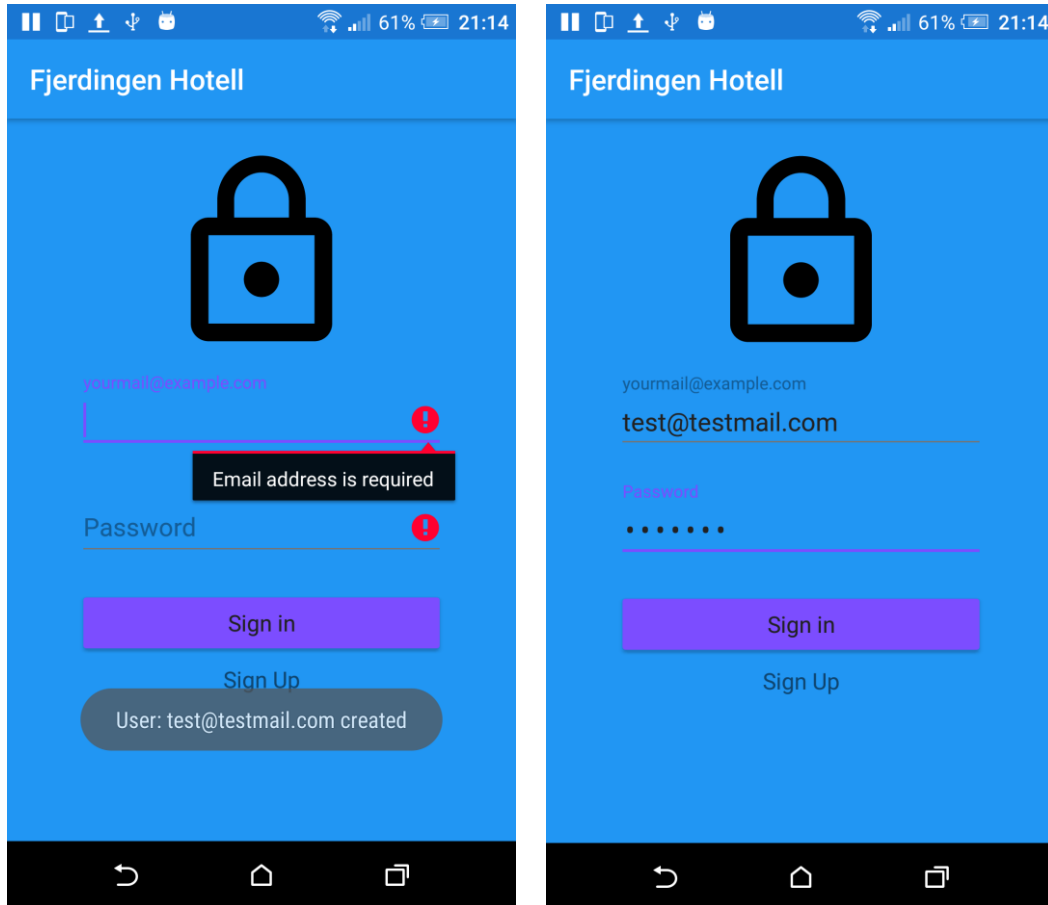
Validering av input på samme måte som tic tac toe.



Lagde en popup for signup. Litt av grunnen til dette er fordi jeg ikke ville lage fragments på login siden da jeg ikke fikk dette ordentlig til med Firebase. Men syntes uansett denne løsningen var fin.

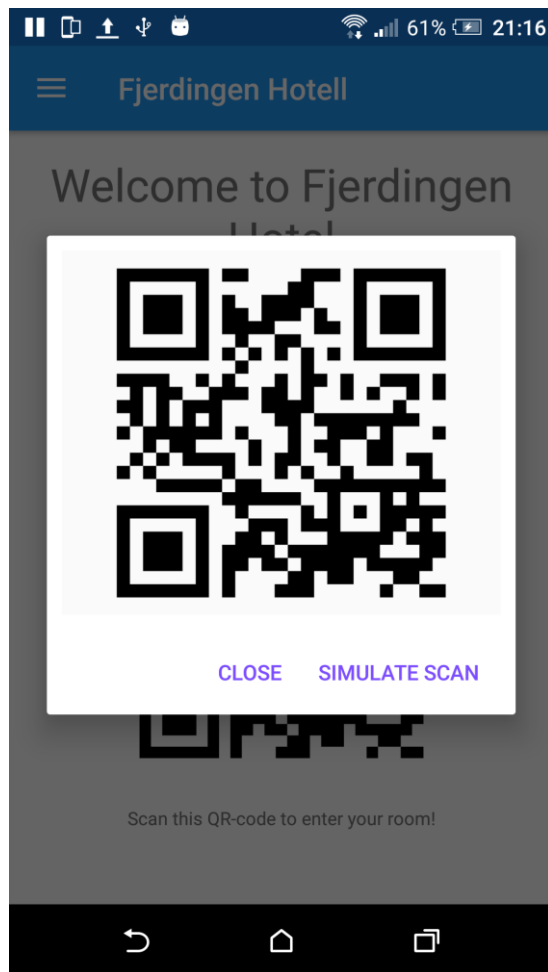
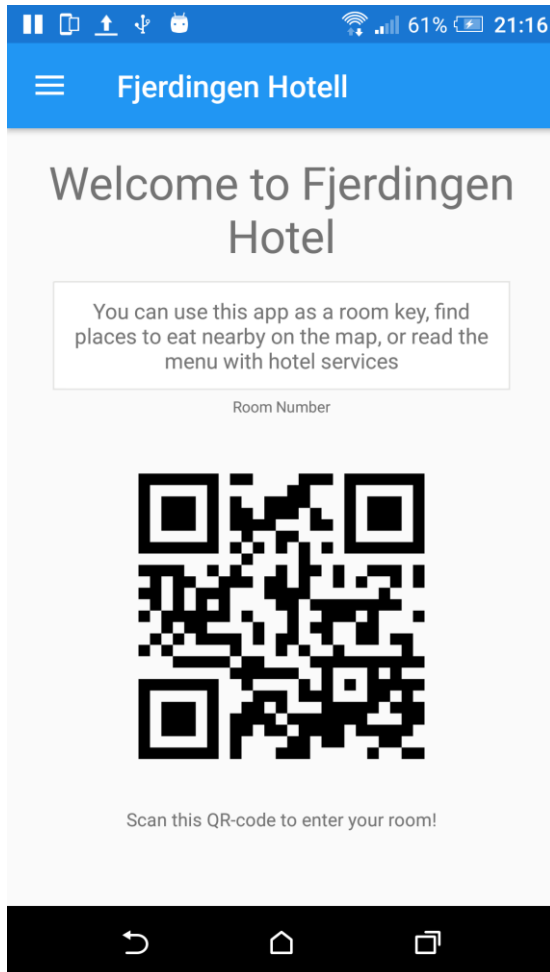


Får en liten toast om at bruker er opprettet. Login tar deg til hoved aktiviten som har fragments til undersider. Hvis man allerede er logget inn vil den gå rett til MainActivity slik at man slipper bruke login skjermen mer enn en gang hvis man ikke logger ut. Det er kjedelig å måtte taste passord hver gang du skal inn på rommet ditt, passord har man allerede på telefonen og vanlig romnøkler krever ikke passord.



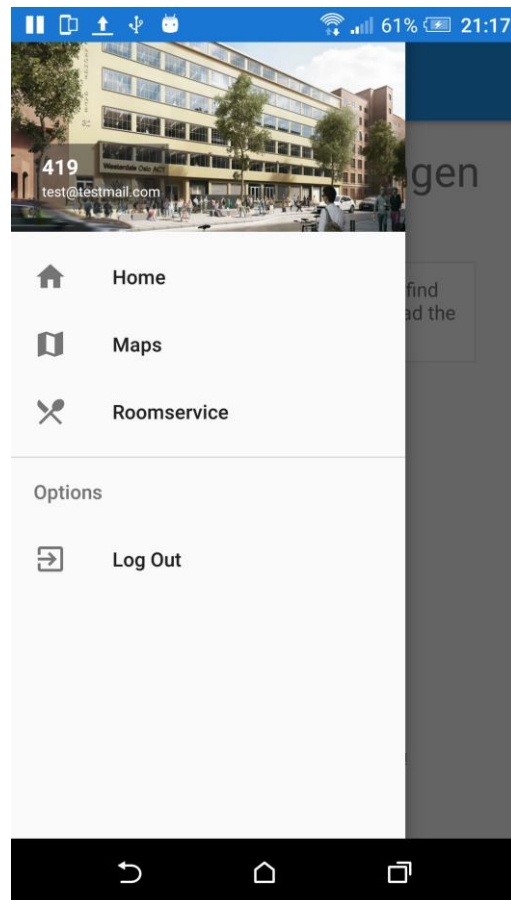
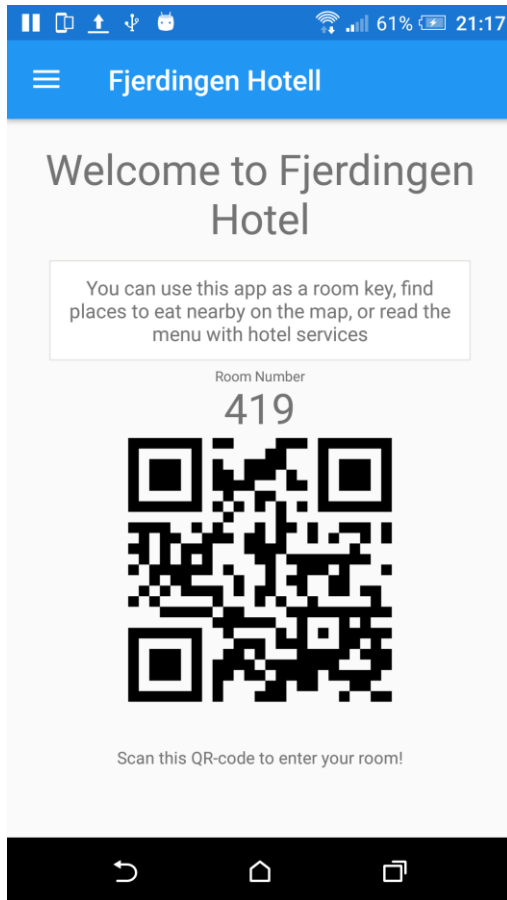
Eksternt API

Det første man ser på hovedsiden er en QR kode laget fra din bruker ID, som resepsjonen kan skanne slik at den kobles sammen med en reservasjon. Jeg har valgt å simulere denne skanningen ved å opprette en reservasjon og knytte denne til brukeren ved å trykke på qr koden og trykke «Simulate scan». QR koden blir generert av biblioteket QRGGen¹ basert på ZXING. Og åpnes i en AlertDialog med custom layout.



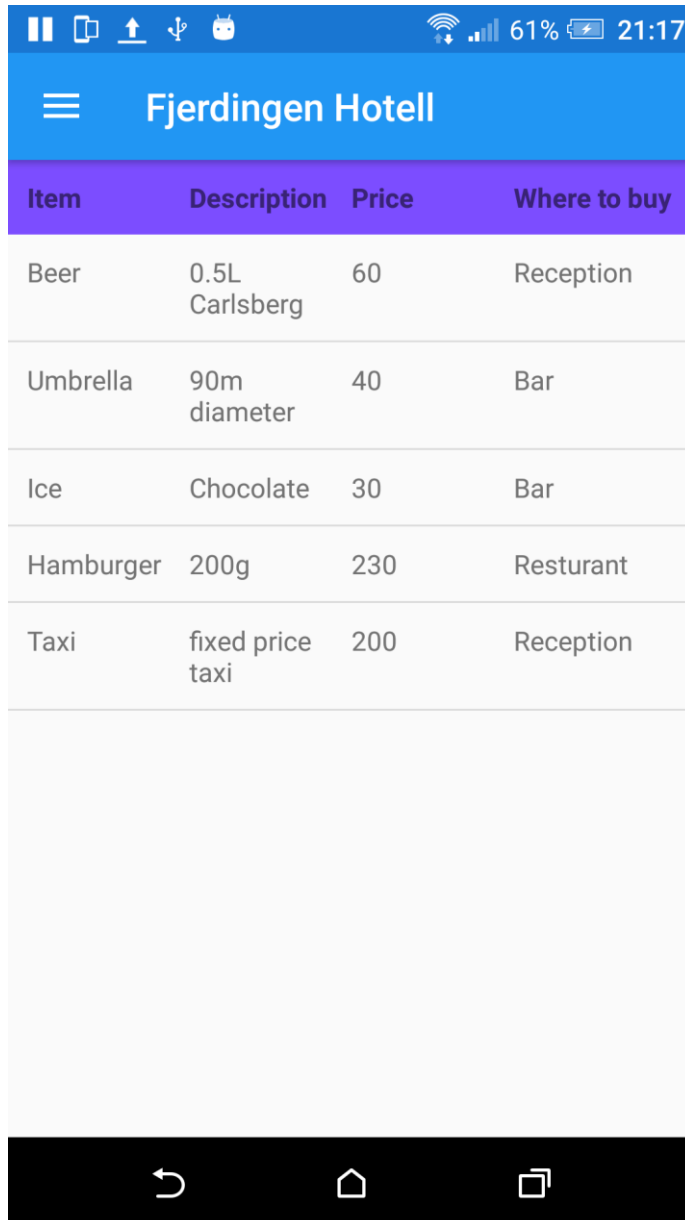
¹ <https://github.com/kenglxn/QRGen>

Så fort man har trykket på «Simulate scan» får man et romnummer på brukeren sin og dette oppdateres i realtime på applikasjonen. Da dukker det opp et romnummer både på forsiden og i draweren.



SQLite

Når det gjelder databasen lagde jeg bare en liste over tjenester man kan få på hotellet ettersom jeg ikke fant noe fornuftig bruk av SQLite når jeg hadde brukte firabase APllet som håndterer offline data. Databasen er derfor bare for å fylle oppgavekravet og er veldig enkel og blir vist i en ListView med custom adapter og layout.

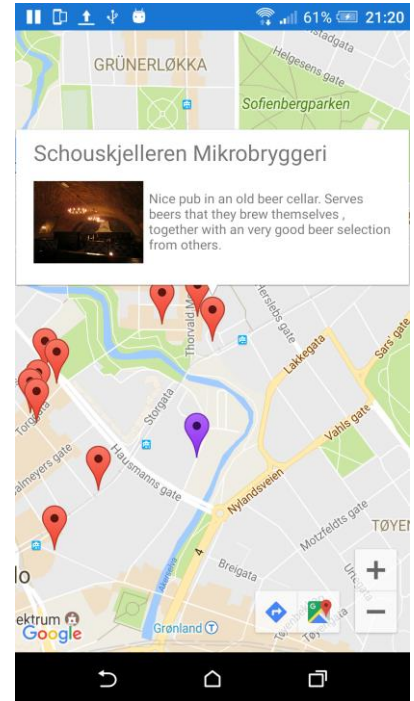
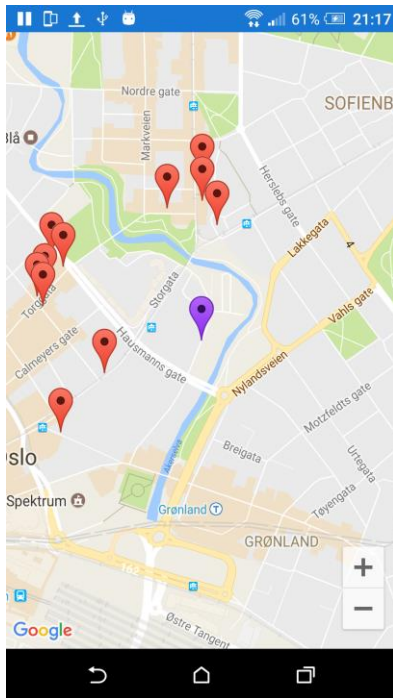


Item	Description	Price	Where to buy
Beer	0.5L Carlsberg	60	Reception
Umbrella	90m diameter	40	Bar
Ice	Chocolate	30	Bar
Hamburger	200g	230	Resturant
Taxi	fixed price taxi	200	Reception

Kart

I kart delen lagde jeg et kart for å vise spisesteder (eller eventuelt andre aktiviteter) som hotellet har lyst til å promotere eller har en avtale med. Her har jeg brukt firebase for å hente data og koordinater for spisestedene i stedet for Google Maps sitt places api for å demonstrere bruk av firebase, og fordi da har hotellet fullstendig kontroll over hvilke steder de vil anbefale. Hotellet er markert med en annen farge enn de andre punktene så det skal være lett å skille dem.

Informasjonen er lagt inn i markeren sitt info layout.



Kilder:

Bilder brukt på restauranter:

- https://c1.staticflickr.com/9/8079/8335062470_62b4e7c602_b.jpg
- <http://maxpixel.freegreatpicture.com/Dish-Food-Grilled-Chicken-Plate-Meal-Barbecue-1836053>
- https://pixabay.com/p-1018962/?no_redirect
- https://pixabay.com/p-1561970/?no_redirect
- https://upload.wikimedia.org/wikipedia/commons/5/5d/Beef_pho_2.jpg
- https://c1.staticflickr.com/8/7175/6587956229_fac0da6b4b_b.jpg
- <http://maxpixel.freegreatpicture.com/Seafood-Japanese-Sushi-Japanesefood-Fish-Food-1769758>
- https://upload.wikimedia.org/wikipedia/commons/thumb/e/e5/Tapas_%28165938155%29.jpg/1280px-Tapas_%28165938155%29.jpg
- https://c1.staticflickr.com/5/4111/5053316701_cd44ec2102_b.jpg
- https://c2.staticflickr.com/4/3284/2885879361_7b2c0e64a8_z.jpg?zz=1
- <https://upload.wikimedia.org/wikipedia/commons/c/c1/Indian-Food-wikicont.jpg>

Beskrivelsen av utestedene hentet fra «reviews» fra Google.