



CoClean

The Way to Keep it Clean

Denne rapporten vil kort presentere apppen
CoClean og dens bruksområder.

Berzi Nawzad Wasfy
berzinw@hiof.no

Introduksjon

CoClean er en app for å forenkle fordeling av arbeidsoppgaver i et kollektiv.

Appen er veldig dynamisk da den skalerer opp og ned basert på antall beboere av kollektivet, dette gjør at appen kan implementeres på flere områder enn Remmen Studentboliger, som appen ble laget for i første omgang.

Når jeg begynte planleggingsprosessen hadde jeg mange planer og ideer, dessverre har tiden vist seg knapp og jeg har måtte sette ned mine forventninger. Jeg valgte derfor å gå bort i fra «Messages» og «My Cleaning Weeks».

Gjennom rapporten vil jeg presentere appen min, dens hovedfunksjoner, styrker og veien videre.

Bakgrunn

Alle som har bodd i en leilighet eller kollektiv med andre, vet hvor vanskelig det er å holde det rent og ryddig, når det er studentboliger er det enda vanskeligere. Jeg valgte derfor å utvikle en app som skulle forenkle rengjøringsprosessen.

Renhold påvirker trivsel, derfor er det viktig at det er rent og ryddig, men hvem skal gjøre hva? Hvordan holde styr på alt? CoClean er en plattform for samarbeid, du og de andre i kollektivet kan registrere leiligheten og rommen deres og arbeidsoppgaver, derfra vil CoClean ta over arbeidet for deg.

Hurtigstart

Appen er veldig enkel i bruk, når du starter appen vil den sjekke om du er logget inn også sende deg til hovedaktiviteten, om du ikke er det vil den gi deg mulighet til å logge inn. Her har du valget mellom å logge inn med email og passord, Google, Facebook eller Twitter. Hvis du prøver å logge inn uten å ha en brukerkonto vil den automatisk lage en for deg.

Appen sjekker om den trenger mer informasjon om deg og sender deg så til riktig aktivitet.

Jeg anbefaler å flytte inn i B3 da jeg allerede har populært denne leiligheten med «Dummy Data» slik at funksjonene kan testes, om du ønsker kan du også opprette nye leiligheter, seksjoner og rom.

Appen har 4 «hovedaktiviteter»

This Week: Oversikt over uken og arbeidsoppgavene for kollektivet.

Apartment: Oversikt over alle som bor i kollektivet.

Tasks: Her kan du lage nye TaskList og tasks

Settings: Her kan du bytte navn, kollektiv, seksjon og rom.

Applikasjonen

Teknisk

Appens fundament er bygget rundt Firebase, alt av informasjon er sendt til Firebase for så å bli hentet ned igjen og behandlet på client-side, ved å sette opp en slik struktur har jeg sikret at data du ser alltid er nyeste tilgjengelige informasjon. Dette gjør det lettere for flere brukere å bruke appen samtidig.

Teknisk sett er appen ser veldig annerledes enn min detaljerte systemdesign, dette skyldes at jeg hadde lite erfaring med Androidutvikling når jeg først utarbeidet den, dog skal kjerne aspektene være det samme.

Registreringsdelen var kanskje det som tok mest tid, da jeg måtte sette opp «paths» for databasekallene mine, dette førte til mange kreative løsninger. Her vil du kunne se fragment inni fragment, inni fragment. Fragmentene sender data mellom seg selv og aktiviteten. Til slutt vil du få registreringsknappen opp, ved å trykke på denne oppretter du din brukerprofil i «/users/» og en referanse til din UID.

I første omgang hadde jeg planer om å implementere objekter i databasen, dessverre viste dette seg å være vanskelig med Firebase, jeg gikk derfor over til HashMaps, når jeg tilslutt forsto hvordan de fungerer forenklet det arbeidet mitt betraktelig.

Design

Appen har en veldig enkel design, jeg tok beslutningen om å holde det rent og pent da jeg innså at ganske tidlig i planleggingsprosessen når jeg fant ut at jeg ikke trengte å gjøre så mye mer ut av det for nå.

Til videreutvikling tenker jeg at det er vitalt med mer grafiske elementer.

Jeg har brukt mye «Constraint Layout» gjennom appen da appen da denne er veldig enkel å ha med å gjøre, men jeg har også brukt Scroll View, Horizontal og Vertical Layout samt andre view elementer for å gjøre appen mer kompatibel med de tusenvis av telefoner som kjører Android.

Design- og tekniske valg

Under produksjonsprosessen har jeg måtte mange valg som gikk bort i fra den detaljerte systemdesignet jeg laget før jeg begynte med koding av selve appen.

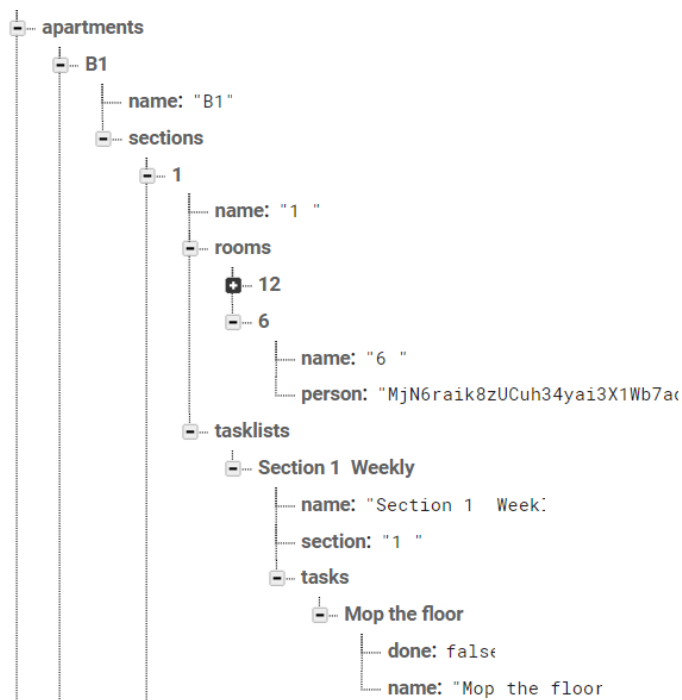
Teknisk

Jeg hadde i første omgang tenkt at all data om leiligheten og brukeren skulle hentes ved oppstart av appen, jeg gikk bort fra dette og henter heller relevant data i hver Activity eller fragment separat. Dette gjør at appen kanskje blir litt tregere, men da blir dataen du ser alltid korrekt.

Firebase

Implementeringen av Firebase viste seg å være å være mer komplisert enn jeg antok, men ved bruk av dens API'er og brukerveiledninger har jeg klart å produsere løsninger for de fleste problemstillinger jeg møtte på under produksjon.

Ved å bruke lyttere på de forskjellige nodene i databasen henter den data hver gang det blir gjort



endringer i denne noden eller dens barn kontra å hente dataen kun en gang. Bruken av lytterne er essensielt gjennom appen min, du kan finne disse lytterne gjennom hele appen.

Et eksempel på bruken av disse lytterne er i Apartment aktiviteten:

Først henter den alle seksjoner i kollektivet, sender seksjonen som paramenter for å hente alle rom i den seksjonen, deretter sender den UID'en til personen som bor i det rommet som en parameter for å hente personen som er koblet opp mot den UID'en.

Design

Jeg har valgt å bruke en blanding av aktiviteter og fragments da dette var mest hensiktsmessig for meg. Alle elementene i hovedmenyen min består av aktiviteter, mens fragment har blitt brukt for å forenkle registreringsprosessen, vise arbeidsoppgaver og personene som bor i kollektivet.



Veien videre

Jeg hadde flere planer for appen, blant annet å kunne be om bytte av vaskeuker, legge ut beskjeder til kollektivet og å kunne se dine fremtidige arbeidsoppgaver, disse fikk jeg dessverre ikke tid til å lage i denne iterasjonen, men de kommer i neste.

Jeg ble dessuten ikke fornøyd med hvordan jeg håndterer arbeidslistene, her skulle jeg gjerne ha hatt muligheten til å kunne krysse av for når man er ferdig, dette skulle så bli sendt til databasen slik at alle kunne se at arbeidsoppgaven er ferdig. Helt optimalt hadde jeg kunne se hvilken seksjon personen skal vaske også.

Jeg ble heller ikke helt ferdig med TaskList og Tasks, her ville jeg lage en variabel som definerte hvor ofte oppgaven skal fornye seg, for øyeblikket støtter appen kun ukentlige oppgaver

Jeg satte meg som mål å forbedre trivselen på studentboliger ved å gjøre det renere, og jeg tror at denne appen vil klare å gjøre det.

Gjennom utviklingsprosessen har jeg klart implementert teknikker og metoder som har blitt presentert gjennom emnet, i tillegg til disse har jeg også implementert andre metoder som har jeg selv tilegnet kunnskap om

Dette prosjektet har mye potensiale; Den kan lanseres som et enkeltstående app, eller så kan løsningen selges inn til f.eks Samskipnadens apper.

Appen skal nå rulles ut til noen kollektiv for testing, deretter vil en ny iterasjon av appen bli produsert. Det er først denne iterasjonen jeg tenker å vise fram til aktører som kan ha interesse av en slik app.

Eksterne ressurser og kode

Firestore

Ved utvikling av denne appen har jeg basert mye av koden min på Firestore sitt API veiledning, mye av koden er standardkode som har blitt modifisert til å passe til appen min. Eksempler på koder som er hentet fra disse dokumentene er lytterne og autentisering, men selv disse har blitt modifisert mye slik at de fungerer slik jeg ønsker at de skal gjøre.

Login Flow

For innlogging har jeg valgt å bruke Firestore Auth UI, dette gjør at jeg får en helhetlig innloggingsprosess, koden for dette er hentet fra bibliotek dokumentasjonen.

Lyttere

Jeg har tatt i bruk Firestore ValueEventListener gjennom hele appen min, basiskoden for de har blitt hentet fra Firestore dokumentasjonen og modifisert til å passe til mitt prosjekt.

RecyclerView

Jeg har brukt RecyclerView for å vise fram personer i kollektivet og for å vise arbeidsoppgaver, for å kunne få til dette har jeg fulgt guiden til CodePath. Dette kan du se gjennom adapterne og modellene som tilhører de spesifikke «RecyclerView»-ene.

<https://guides.codepath.com/android/using-the-recyclerview>

Nettsjekk

Etter mye tid med å programmere min egen nettsjekk valgte jeg å gå for noe som fantes allerede da han gjorde det på tre linjer mens jeg trengte 15+ linjer.

<https://stackoverflow.com/a/4009133/5414015>