

Spotify party player eksamens rapport - Tormod Rønning

Ideen

Spotify party player er en app for at flere android enheter på samme nettverk kan kontrollere playeren til en android enhet som spiller musikk. Dette er i utgangspunktet for at det skal være lettere for alle (med android telefon) på fest å queue opp sanger.

Produktet

Appen består av 4 sider:

Hovedside som gir valget om man skal være host for musikk, eller koble seg til en mobil som allerede hoster.

Hvis man velger å hoste, vil man komme til en oppretting av session side. På denne siden er det 5 inputs, og 2 knapper, inputsene er: Navn på festen, ønsket sjanger som skal bli spilt, og 3 passord for forskjellige rettigheter. Det er ikke noe funksjonalitet bak sjanger, eller de 3 passordene for øyeblikket, men de er med for å vise hvordan det er tenkt at det skal fungere. De to knappene er: randomize, opprett ny session. Randomize har ikke funksjonalitet enda, men skal i utgangspunktet bare sette random verdier på inputsene, utifra en liste med strings.

Da man trykker på opprett ny session så vil man bli navigert til en ny side hvor informasjon om sessionen vises, sang som blir spilt, og en liste over sanger som skal bli spilt. Denne siden vil være fylt ut med default verdier, bortsett fra Navn på festen og ønsket sjanger, helt til en sang blir queueet. For å queue en sang kan man trykke på "add" knappen og paste inn en spotify uri. Da en sang blir queueet, vil den spille av sangen hvis det ikke allerede spilles av en valgt sang, hvis det gjør det, vil sangen bli lagt til i ei liste under. Da en sang spilles av kan man trykke på "next" knappen for å spille av neste sang i queue. Hvis "pause" knappen blir trykket på blir sangen pauset, og "pause" teksten vil bli erstattet med "resume", og knappen vil resume' sangen om den blir trykket igjen.

Hvis man velger å koble seg til en mobil som allerede hoster, vil man komme til en available sessions siden. På denne siden er det en liste med sessions som blir hostet over wifi. Disse vil vises duplikat, som er en feil jeg ikke har rukket å fikse. Da man trykker på en session skal man bli koblet til sessionen som hoster og navigert til en session, men det er en bug som gjør at man bare blir koblet til den siste sessionen NSD fant.

Session siden man kommer til via available parties, vil være en side lik siden for den som hoster, men bruker en annen java klasse enn hosten. Funksjonene for knappene skal fungere slik at den sender en string igjennom nsd til hosten. Da hosten motar denne vil analysere stringen for å vite hva den skal gjøre, og sende tilbake informasjonen for å forandre på session siden til den tilkobla enheten. På grunn av dårlig tid etter jeg fikk lagt til NSD, og dårlig testing på emulator, så er denne siden uferdig, og vil føre til at appen crasher på mange forskjellige måter. Se heller på koden i `UserSession.java` og `availableSession.java` for å vurdere dette.

Brukerveiledning

- Åpne appen
- Trykk på Start Session knappen
- Skriv inn noe i de to øverste feltene
- Trykk på start session
- Logg inn på en spotify premium bruker
- Trykk på "add"
- Skriv inn en spotify uri (Dere kan bruke en av disse:
`spotify:track:2w8cJaVndSM5hPTiCzpQ0b`
`spotify:track:6qN6KJHLanSFBoy6E8SqB9`
`spotify:track:3VoYJd1esvExjbp7KgNS5N`
`spotify:track:330gP3Ziu6TAmkvIMjOZ0R`
`spotify:track:2ihCaVdNZmnHZWt0fvAM7B`
)
- Trykk "ok"
- gjenta de 3 forrige stegene noen ganger
- Trykk på "pause"
- Trykk på "next"
- Trykk på et av elementene i listen.

Hvis du går tilbake til start siden og til available session siden, og trykker på elementet i listen, så trykker tilbake til available session siden vil den vise alle hosta sessions på listen. Men funksjonalitet på denne delen av appen er dårlig.

Funksjoner å fokusere på

Funksjonene som er i `sessionActivity.java` er det som det har blitt lagt mest tid til. Jeg fikk også til å hente informasjon om `queuelisten`, som spotify SDK ikke gir muligheten til, men ved bruk av informasjon om sangen som spilles av nå, fikk jeg til dette uten å bruke spotify web api. Jeg er også fornøyd med bruk av chat connection over NSD for å sende kommandoer mellom appene, og at jeg omstrukturerte informasjonen jeg trengte fra Session klassen til en tekststreng for å få lov til å sende den som en "chatte melding"

Design

Designet valget er tatt for å få litt følelse av spotify, blandet med litt mørk blå farge som passer godt med grønnfargen.

Add, pause, og next knappene er det brukt et desgin laget i android:drawable, for å få den rund.

Bruk av andres kode

Til NSD har jeg brukt NsdHelper og ChatConnection fra Android developer sin master side. Jeg har også brukt Spotify Authentication, og Spotify player, fra spotify sitt android SDK. hjelp fra nettet til adapter view og lignende funksjoner for å forandre på tekst og lignene i viewet.

Rakk ikke å bli ferdig med:

Jeg rakk desverre ikke å bli ferdig med NSD. Derfor er både available session og user session sidene ikke fungerende slik de skal fungere.

Se i Utvidet projekt beskrivelse for hvilke tanker jeg hadde for funksjonalitet både for appen og videreutvikling av appen.

Arbeids prossessen

Arbeids prossesen min var planlegging i starten, til alpha levering så startet jeg med selve utviklingen av appen. På beta innlevering så fikset jeg spotify player og authentication. Dette fortsatte jeg med for sluttinnlevering, sammen med å starte på NSD. Jeg slet endel med å få implementert spotify SDK, men det gikk lett etter jeg fikset det sånn at jeg bruker minimalt av funksjoner fra SDK'et og heller har lagt egne funksjoner for det. NSD satt jeg veldig fast på, og mot slutten måtte jeg gi opp på å fikse det ferdig, og heller forkusere på å få resten av appen til å fungere bedre.

Konklusjon

Det har vært et gøy fag å jobbe med, og føler jeg har lært en god del om mange forskjellige ting under prossessen. Jeg hadde håpet at produktet fungerte til den grad at det ikke var noe feil med tilkobling via NSD, men er veldig fornøyd med det jeg fikk til å fungere og måten jeg har løst dette på.

Jeg har lagt til dokumentet jeg laget underveis i projektet, for at dere kan se litt mer om hvordan prossessen for utvikling av denne appen har vært. Dette ligger da under denne siden.

Utvidet Prosjektbeskrivelse av Spotify party player

Beskrivelse

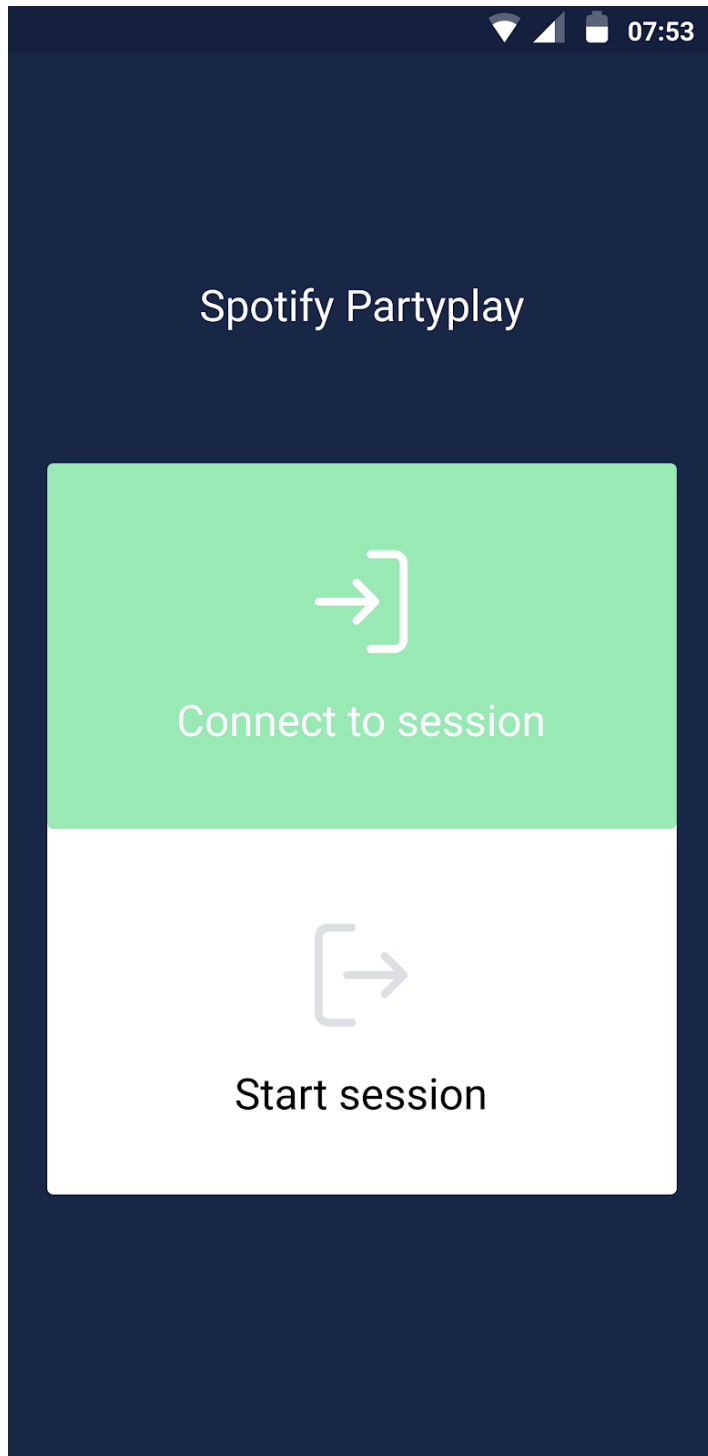
Spotify party player er en app som skal gjøre det enklere for grupper å sette på og queue sanger på spotify, da musikken spilles på høyttalere fra én enhet. Mobilen som spiller musikk, starter en session i appen, og andre kobler seg til den sessionen, for å se sangen som spiller der og da, se hele queue listen, og legge til spotify sang url for å queue opp/ sette på ny sang.

Kravspesifikasjon

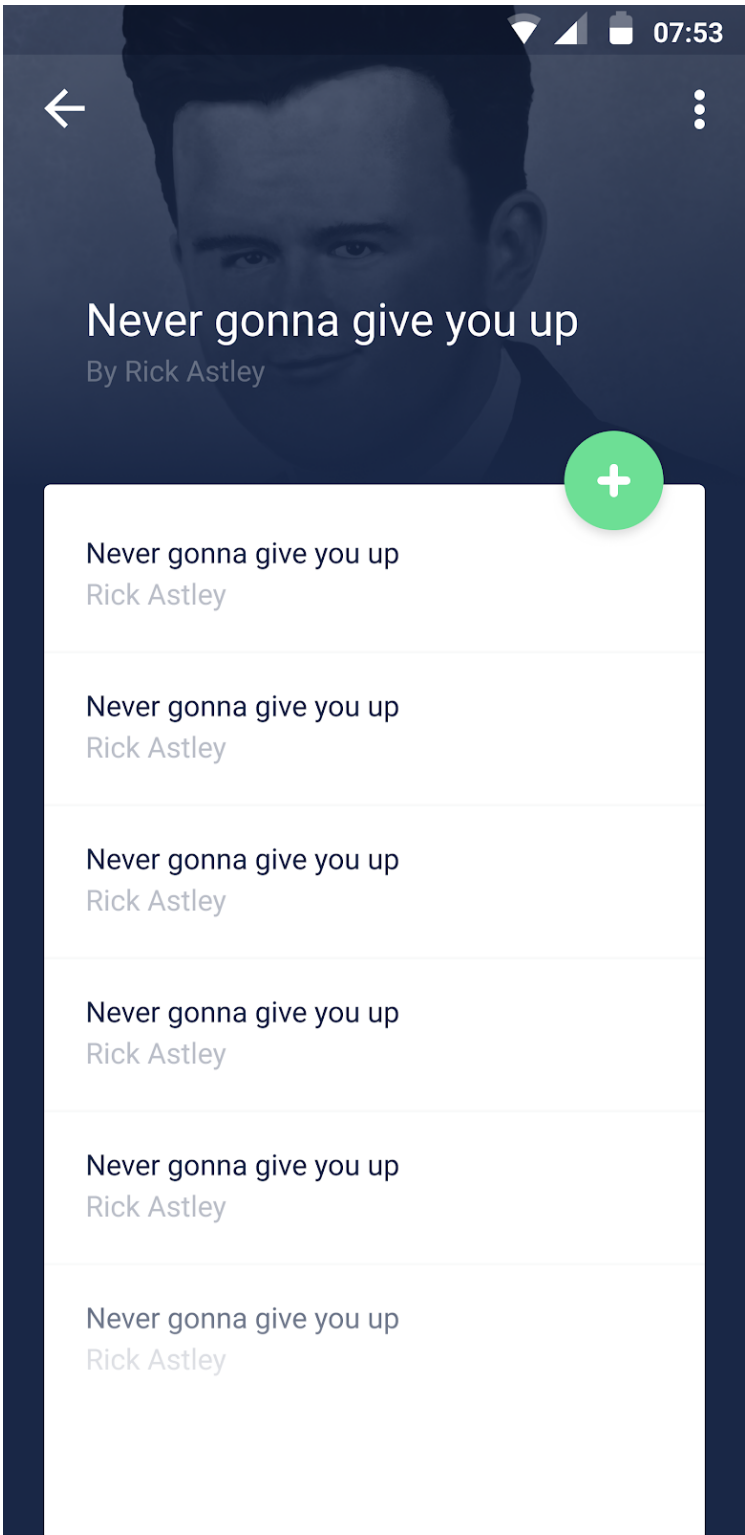
- Da man åpner appen skal start siden åpnes. Den skal inneholde to knapper, en for å opprette session, og en for å søke etter sessions som er i gang.
- På alle sider skal det være en knapp som navigerer til start siden.
- Sessions skal broadcaste gjennom Network Service Discovery (NSD).
- For å opprette en session i appen, må en mobil være koblet opp til spotify så opprette en session. Dette blir da party player enheten.
- Da en session skal bli opprettet, skal det settes en navn på session, velge om det skal være passord eller ikke for å bli med i session, og velge om det skal være passord eller ikke for å queue' / sette på ny sang.
- Da man har fylt inn det man ønsker for sessionen, skal man kunne trykke på en "opprett"- knapp, eller en "avbryt"-knapp.
- Da man trykker på "avbryt"-knappen skal appen navigere tilbake til start siden.
- Da man trykker på opprett, skal appen opprette en session, koble den opp til NSD, så navigere til session siden.
- Session siden skal bestå av 3 deler:
 - 1:Sangen som spilles av nå, som skal vises på øvre del av skjermen.
 - 2:Resten av skjermen skal vise hvilke sanger som er de neste i queue lista.
 - 3:En knapp som forandrer resten queue lista, om til et input felt som man skal legge til spotify sang url i, og en knapp som legger til sangen i queue i party player enheten.
- Da en sang blir lagt til i queue i session side del 3, skal siden forandre tilbake til session del 2 automatisk.
- Da en party player enhet kobler seg av sessionen, skal alle andre tilkoblede enheter bli satt på vent, med mulighet til å navigere tilbake til start siden.
- Da en party player enhet kobler seg av sessionen, skal party player enheten bli navigert til startsidene, men med en ekstra knapp for å re-opprette den tapte sessionen.
- Om party player enheten re-oppretter en gammel session, skal alle enheter som står på vent, bli automatisk koblet til sessionen igjen.

Skisser

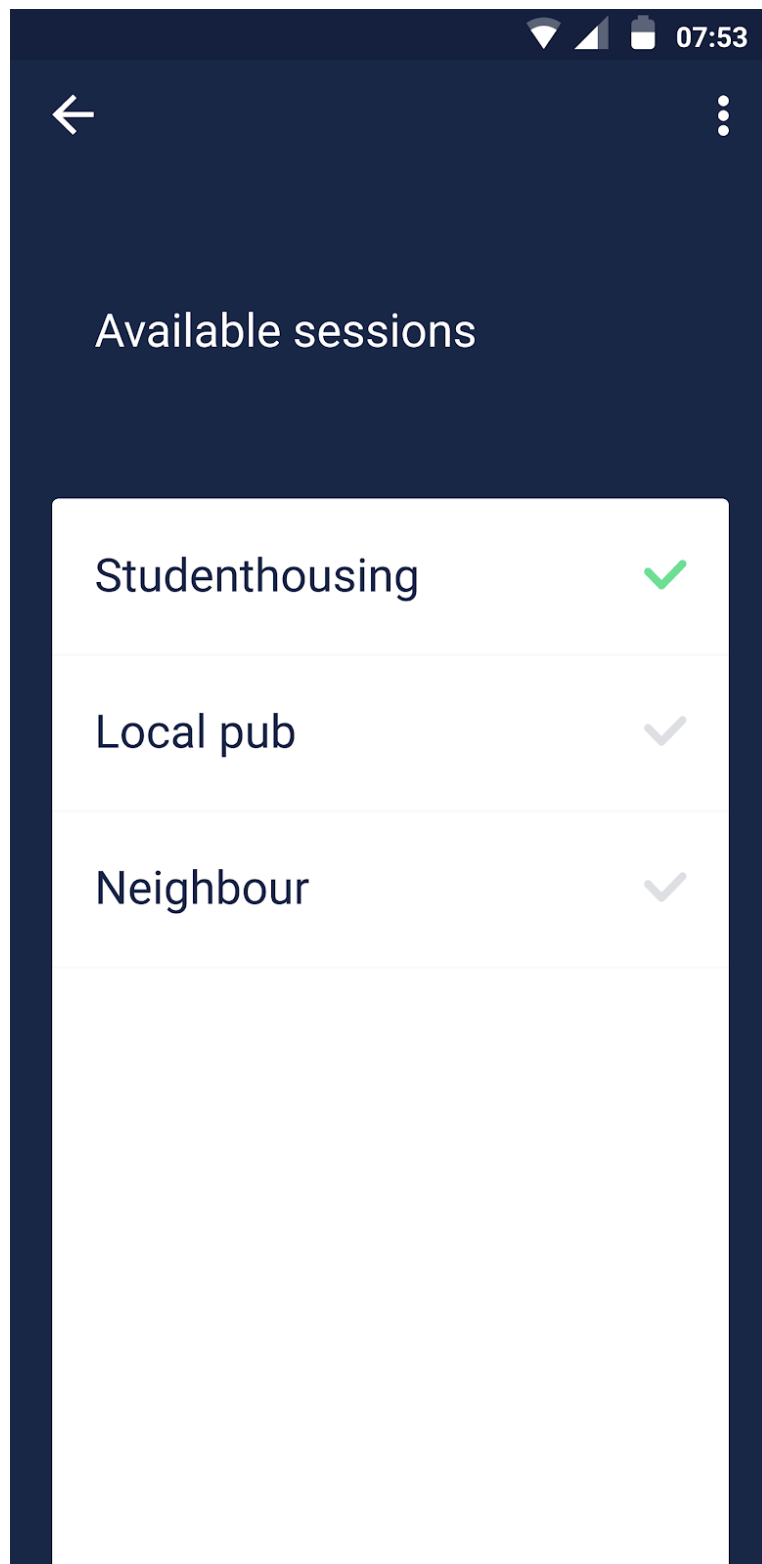
Start side



Session side



Tilgjengelige sessions side



Unikhet

Det finnes forskjellige apper som ligner denne. Denne skiller seg ut på den måten at den bruker bare tilkobling enkelt via lokalt nettverk. Poenget er å gjøre det så enkelt for bruker som mulig, og gi brukerne all makt over sangene som spilles.

Eventuelle utvidelser

- Knapp for å queue link direkte fra clipboard.
- Neste sang knapp.
- Tilkobling via bluetooth.
-

Detaljert systemdesign av Spotify party player

Oppdeling

Appen skal bestå av fire sider i appen, Start side, Opprett session side, Tilgjengelige sessions side, og Session side. Den skal også bruke NSD og Spotify SDK.

Start side

Innhold

Denne siden vil være den første mann kommer til da appen starter. Her vil det være to knapper: en som navigerer til Opprett session side; en som navigerer til Tilgjengelige sessions side; Over knappene skal det stå navnet på appen "Spotify party player".

Kontroller

Hvis "Back"-knappen blir trykka på, på denne siden, skal appen avsluttes.

Opprett session side

Innhold

Denne siden skal inneholde input fields(stjerne merket input fields må være fylt ut) for:

- *Session navn (Default verdi: "Best party evvah")
- *Ønsket sjanger (Default verdi: "Whatever")
- Passord for å komme inn(Default verdi: "")
- Passord for queue rettigheter(Default verdi: "")
- Passord for om rangering av queue liste, og neste sang rettigheter(Default verdi: "bestPasswordEvvah")

Om et stjerne fylt input field ikke er fylt ut, skal det komme en feilmelding om at det må fylles ut, etter at opprett knappen har blitt trykka på. Hvis enheten allerede er koblet til en session, og prøver å opprette en ny session, skal det komme opp en popup som sier at for å opprette en ny session, må man koble av nåværende session. Passord for å komme inn, og for queue rettigheter skal satt til default "" fordi det å se på lista og queue skal være den vanlige måten å opprette session. Passord for om rangering av queue liste, og neste sang rettigheter, skal være "bestPasswordEvvah" for å vise at det kan være smart å sette passord på den.

Den skal også inneholde to knapper: en som tar informasjonen skrevet i input, og oppretter en ny session class og navigerer til session siden; en som navigerer til tilgjengelige sessions siden;

Kontroller

Om det blir swipet til venstre på denne siden, skal det navigeres til Tilgjengelige sessions siden.

Om "Back"-knappen blir trykka på, skal det navigeres til Start side.

Om "Back"-knappen blir dobbel tappet så skal appen avsluttes.

Tilgjengelige sessions

Innhold

Denne siden skal inneholde liste med alle tilgjengelige sessions appen finner med bruk av NSD. Hvert listelement skal inneholde navn på session, ønsket sjanger, navn på sang som nå blir spilt, og tre ikoner som representerer om det er passord eller ikke på de forskjellige rettighetene. Hvis et listelement blir trykka på skal devicen bli koblet til sessionen, og appen skal navigere til den session siden.

Over denne listen skal det en overskrift som det står "Available sessions".

Siden skal også inneholde en knapp som navigerer til opprett session siden.

Kontroller

Om det blir swipet til høyre på denne siden, skal appen navigere til Opprett session siden.

Om det blir swipet til venste på dennes siden, skal appen navigere til Session siden, hvis enheten allerede er koblet til en session, hvis ikke skal den ikke gjøre noe.

Om "Back"-knappen blir trykka på, skal det navigeres til Start side.

Om "Back"-knappen blir dobbel tappet så skal appen avsluttes.

Session

Innhold

Sessions siden skal inneholde en liste over hvilke sanger som er i queue på party player enhetens spotify. Her skal den neste sangen i queue være øverst, neste etter den under, og sånn fortsetter det.

På toppen av den listen skal det være en knapp. Da den knappen blir trykka på skal det poppe opp et vindu, som har en input for spotify url, og to knapper: en for å legge til link som er skrevet inn i input; en for å legge til link som allerede er "copied to clipboard";

Da knappen for å legge til link fra input blir trykka på skal appen finne sangen igjennom spotify api, og legge den til i queue på session hvis den finnes, hvis den ikke finnes skal det komme en feilmelding som sier at linken ikke er gyldig. Da knappen for å legge til link via clipboard blir trykka på, skal appen hente link fra clipboard og finne sangen igjennom spotify api, og legge den til i queue i session hvis den finnes, hvis den ikke finnes skal det komme en feilmelding som sier at linken ikke finnes.

Om det blir trykket et annet sted på skjermen enn på vinduet for å legge til sang i queue, skal vinduet bli borte og vanlig session side vil vises.

Over queue listen skal det stå hvilken sang som spilles i nåtid, og hvor lenge det er igjen før neste sang, tiden igjen skal vises ned til sekundet. Bakgrunnen på session siden skal være samme som bildet til sangen som er på spotify, så appen skal også hente bildet igjennom spotify apiet.

Da en sang er ferdig, skal neste sang i queue komme over listen, og den sangen skal fjernes fra øverste plass i queue, for så å dytte alle sanger et hakk oppover.

Kontroller

Om "Back"-knappen blir trykket på da queue ny sang vinduet er åpent, skal vinduet bli lagt ned.

Om "Back"-knappen blir trykket på da session siden er oppe, skal appen lukkes, men ikke avsluttes.

Om "Back"-knappen blir dobbel tappet da session siden er oppe, skal appen avsluttes.

Om det blir swipet til høyre da session siden er oppe, skal det navigeres til tilgjengelige sessions siden.

Spotify API'et

Denne appen skal bruke spotify api for å kommunisere med spotify appen på telefonen. Den skal kunne bruke følgende funksjoner:

- Lese hvilken sang som blir spilt
- Lese hvor lenge det er igjen av sangen som blir spilt
- Lese hvilke sanger som er i queue lista på spotify
- Legge til sanger i queue lista på spotify
- Ha tilgang til neste sang knappen
- Ha tilgang til pause knappen

Network service discovery

Denne appen skal broadcaste en host gjennom NSD, andre enheter med appen skal kunne se disse hostene og koble seg opp i en session gjennom NSD. Om forbindelsen blir brutt skal appen prøve å gjenopprette forbindelsen. Hvis bruker velger å koble seg av session, skal forbindelsen mellom enhetene bli brutt, og sessionen legges ned.

Klasser

Mange av feltene i klassene jeg har lagd, er avhenge av å ha en default null verdi, for så å bli oppdatert ettersom appen blir brukt. Hvordan å få en unik ID har jeg ikke tenkt ut helt enda, men regner med at jeg bruker en ID som allerede er satt på enheten (eventuelt at jeg oppretter user klassen ved oppkobling på session, istedenfor ved start av app, om jeg gjør dette, kommer det til å bli forandring i klasse oppsettet).

Class User

ID (Private int)

Nickname (Private string)

ConnectedSession (Private Session)

ConnectionToNSD (Private Boolean)

Dette blir en klasse som er en per enhet. Da appen startes skal den generere en User klasse som vil være klassen som kobles til en Session etterhvert. Denne klassen er for å skille de forskjellige enhetene som er koblet på en Session.

Class Session

ID (Private int)
 Name (Private string)
 Genre (Private string)
 ViewRightsPassword (Private string)
 QueueRightsPassword (Private string)
 FullRightsPassword (Private string)
 CurrentSong (Private Song)
 Queuelist (List<Songs>)
 Host (Private User)
 ConnectedUsers (List<Users>)
 ConnectionToSpotify (Private Boolean)
 ConnectionToNSD (Private Boolean)

Denne klassen vil være klassen for sessions. Da en Session blir opprettet blir det lagd en ny instance av denne klassen. Denne klassen er avhenge av en host som oppretter sessionen. Denne hosten vil være den mobilen som spiller musikk fra spotify, og er derfor avhenge av at spotify er åpent på mobilen. Andre users som kobler seg på sessionen vil bli lagt til i ConnectedUsers lista. Det skal være en listener som oppdaterer queuelist til å være lik queuelist på spotify fra hosten sin spotify.

Underutvikling regner jeg med at denne klassen kommer til å forandre seg litt, det kommer ann på hvilken løsning jeg velger for passord/ inngangs koder, hvordan NSD funker, hvilke problemer jeg støter på med spotify apiet, og eventuelle forandringer/ utvidelser jeg velger.

Class Song

URI (Private String)
 Name (Private String)
 Picture (Kommer ann på spotify)
 Length (Private float)
 Start (Private long)

Denne klassen vil holde på informasjonen som trengs per sang. Da det blir lagt til en ny instance av denne klassen, vil den legge den direkte inn i Session.queuelist

Ressurser

Lyder

Det skal spilles av en enkel lyd på enheten ved oppretting av session, tilkobling til session, og frakobling fra session. Jeg har ordnet avtale med hobby artist Sander Brage Berg, for at han skal lage disse lydene for meg.

Bilder

Alle bilder brukt til denne appen skal bli tatt direkte fra spotify sangen som blir spilt, om sangen ikke får til å velge bilde, så vil det bli et default tema som bakgrunn istedenfor.

Usikker

Bakgrunn listener

Jeg har lyst til å legge til en listener på appen min, som fortsetter å høre etter da appen har blitt lagt ned (men ikke avsluttet), og den skal merke da en bruker er inne på spotify og trykker på queue knappen innenfor en sang. Da skal den istedenfor å queue opp på spotify, heller sende queue funksjonen til Spotify party player appen, så skal appen sende queue kommandoen til spotifyen til host mobilen.

NSD

Er litt usikker på hvordan dette faktisk fungerer, fordi jeg har ikke lest meg fullt opp på dette enda. Ut ifra det jeg har lest fikk jeg inntrykk at det ikke var så vanskelig å få broadcaste med en host. Hvis det oppstår mye problemer med dette så er nok dette noe jeg kommer til å trenge mest hjelp med.

Spotify SDK

Jeg regner med at mesteparten av dette kommer til å gå fint, jeg har litt erfaring med bruk av soundcloud api, som jeg har fått inntrykk av at er ganske så likt. Det jeg tror blir mest problematisk med dette er å implementere dette i selve appen, men da jeg har fått til det, tror jeg at jeg klarer å bruke apiet til det jeg trenger.

Alpha

Til alphaen har jeg lagd layout for de fire forskjellige sidene, oppsettet er tilnærmet lik hvordan det skal se ut, men mangler noen små justeringer på farger, størrelser av elementer og tekst. Det er navigasjon mellom aktivitetene ved trykk av knapper, ikke ved swipe. Det er ikke koblet opp mot spotify eller NSD enda, så funksjonalitet er begrenset. I resources har jeg lagt til de fleste strings appen trenger, og lagt til noen farger, har ikke prøvd å lage tema.

På forsiden kan man navigere til available session siden eller start session siden.

På available session siden, er det en overtekst og et ListView som blir fylt med antall sessions som er tilgjengelig. (Antall sessions tilgjengelig er definert av hvor mange ganger mann oppretter en session på start session siden. Dette er fordi oppkobling til NSD ikke er tilgjengelig i denne versjonen), Det er satt en funksjon som legger til onclick på liste elementene, men den onlicken gjør ikke noe på denne versjonen.

På start session siden er det 5 inputs, med hint tekst om hva de skal inneholde, hvis name eller genre inputen ikke er fylt ut da create session knappen blir trykket, vil det komme en feilmelding i form av at password3 (femte input) får tekst om at mann må fylle ut de to første. Password inputene trenger ikke å være fylt ut for å lage sessionen. Det er en ekstra knapp som det står "Randomize" på, denne har ikke noe funksjonalitet akkurat nå, men skal i senere versjon fylle ut feltene med random verdier tatt ut ifra string-arrays fra resources. Hvis de to første er fylt ut, og create session knappen blir trykket, vil det navigeres til session siden, og da vil den nye sessionen også bli tilgjengelig på available sessions siden.

På sessions siden er det lagt til tre tekster helt øverst, som inneholder 1:session navn, 2:genre, 3:brukernavn. Brukernavn går ikke an å forandre enda. Under disse tekstene er det sangtittel og artistnavn, disse er satt til en default verdi fordi det ikke er koblet opp mot spotify enda.

Beta

Til beta har jeg lagt til Spotify SDK, og ryddet opp i klasser, samt flyttet på instansiering av dem. Nå er det mulig å spille av sanger i appen, og queue sanger i Session siden. Da man oppretter en Session vil det spilles en sang, og tittel og artist av sangen som spiller vil vises. Hvis man trykker på den grønne knappen som det står "PLUS" på, kan man linke inn en spotify URI, og den vil bli lagt til i listen under, med sanger som er queueet. Første sangen som spilles starter 200000 millisekund inn i sangen for å slippe å høre igjennom hele sanger før neste spilles. Spotify SDK har ikke mulighet til å vise metadata om andre sanger enn den som spilles, forrige og neste sang, for å vise tittelen til andre sanger enn førstkommande i queuelist, må jeg koble den opp til spotify web api i tillegg. Oppkobling med flere mobiler er fortsatt ikke mulig pga jeg har ikke koblet den opp til NSD enda, dette vil komme i neste versjon. Det er heller ikke lagt til sjekker for "uforventede hendelser" så ved feil input eller mulige andre hendelser er det mulig at appen crasher.

Kilder

NSD:

<https://developer.android.com/training/connect-devices-wirelessly/nsd.html>

Lignende apper:

<https://festify.rocks/>

<https://jukestar.mobi/>

Design hjelp fra Troy Bergland:

<https://www.faen.co/>

Lyd hjelp fra Sander Brage Berg:

<https://www.facebook.com/SanderBrageBerg>

Android:

<https://developer.android.com/design/patterns/navigation.html>

<https://developer.android.com/design/material/index.html>

<https://developer.android.com/design/get-started/principles.html>

Spotify:

<https://spotify.github.io/android-sdk/player/>