

Timebox 6 – APP struktur

Oversigt

OpgaveNavn	Generel struktur		
Implementering af krav	N/A		
Udført af	Begge	Dato	28-10-2021
Timebox	6	Område	Mobilapp

Contents

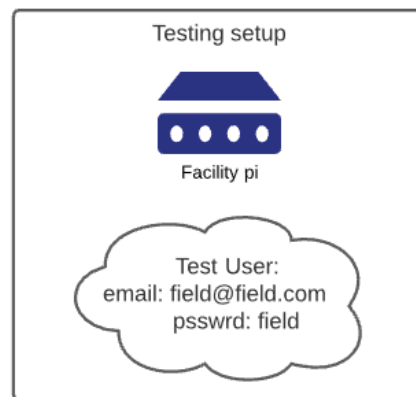
INTRODUKTION.....	1
ANALYSE OG DESIGN.....	2
KONKLUSION	7

Introduktion

Dette dokument omhandler struktur for mobilappen, hvordan og hvilke elementer den skal indeholde, hvilke interfaces der er, fra mobilens side.

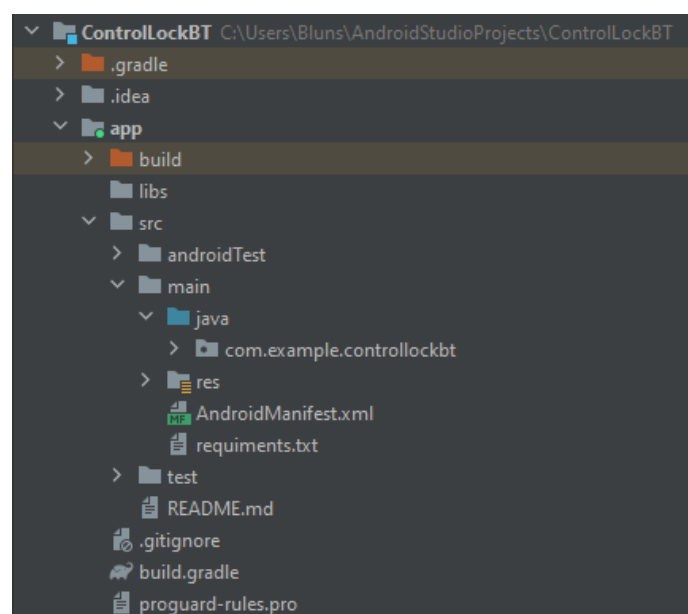
Analyse og Design

Som beskrevet i "Launch Phase" rapporten anvendes der en Raspberry Pi i testsetup'et



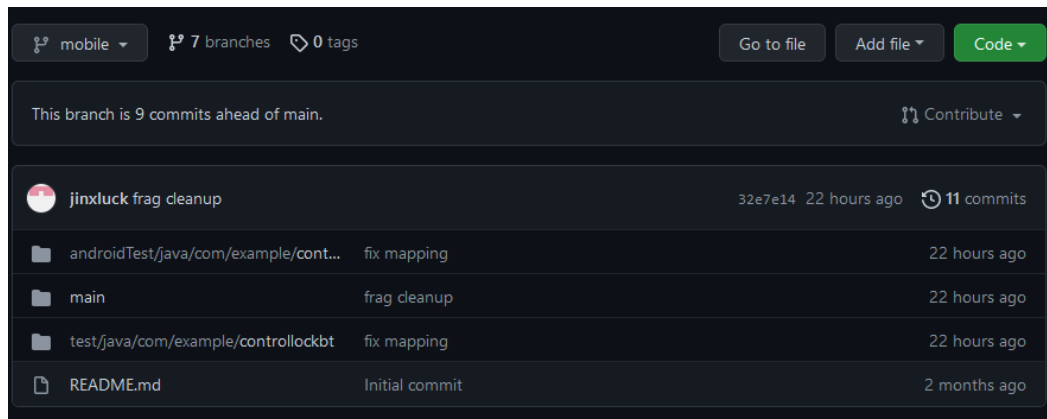
Figur 1 Testsetup

Opsætning af appen er som på Figur 2, hvor der er tilføjet en fil "requirements.txt", som indeholder alle de dependencies, som er tilføjet til projektet. Den er blevet tilføjet for at gøre det nemmere, at dele projektet på GitHub, uden at ødelægge ens IDE-opsætning, ellers er strukturen som standard ved oprettelse af projektet.



Figur 2 mappe strukturen

Opsætningen af GitHub er som på Figur 3, hvor der arbejdes i ud fra branchen "mobile" og det der sendes op på GitHub, er "src" mappen fra projektet.

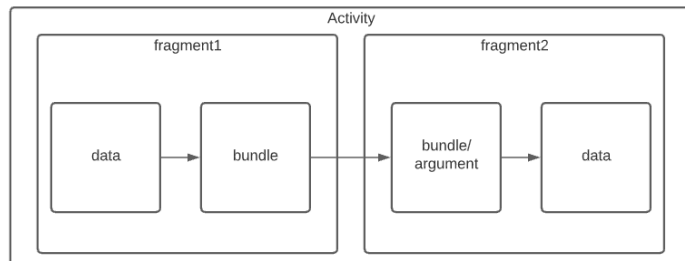


Figur 3 Git-opsætningen

Når man så laver en "clone" skal man selv tilføje til ens "build.gradle" fil, de dependencies som står i "requiments.txt" filen.

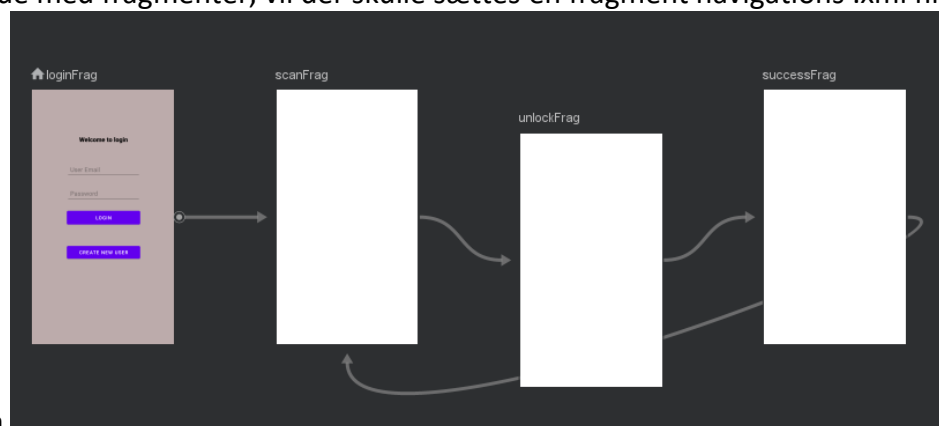
Fragments og Bundles:

Ide'en bag fragments er så det er nemmere at dele data, imellem to forskellige sider på en app. Måden der deles mellem fragments, kan ses på Figur 4, hvor der er 2 fragmenter som kører i same activity ("process"), hvor data'en der skal passers mellem de 2 fragmenter kan pakkes sammen til et bundle, der så kan sendes som et argument til et nyt fragment, som derefter kan "udpakke" data'en og gemme dem i fx. variabler.



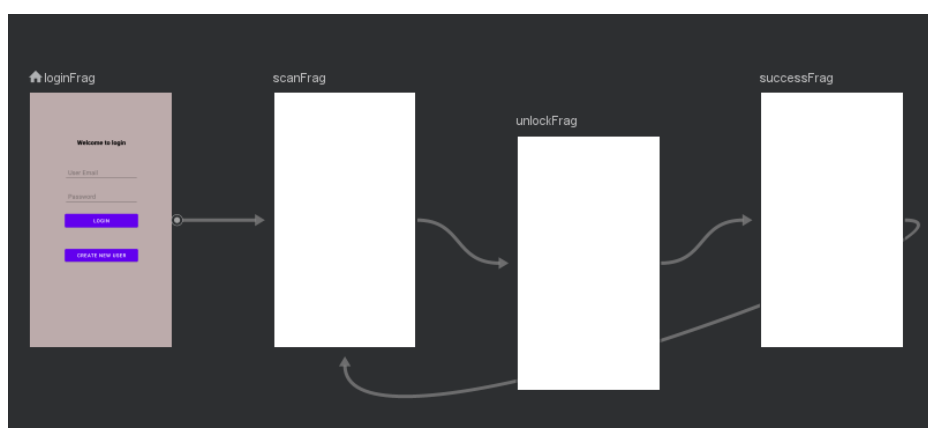
Figur 4 fragment diagram

For at kunne arbejde med fragmenter, vil der skulle sættes en fragment navigations .xml fil



op, som kan ses på

Figur 5. Navigationsfilen sætter hvilke fragmenter der er, hvilket fragment der er start fragmentet, og hvordan der kan passeres rundt i fragmenterne.



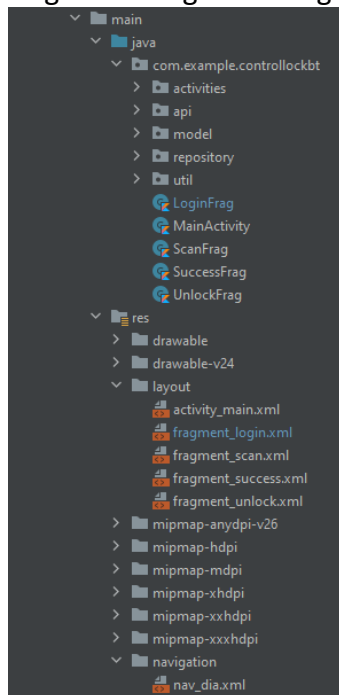
Figur 5 nav dia

Efter fragmentnavigationsdiagrammet er kreeret, vil der i MainActivity .xml filen skulle sættes en fragment view widget ind, som er den widget der vil vise fragment xml filerne.

Til hvert fragment, vil der blive oprettet en Kotlin fil (.kt), til at håndtere hvad der sker i fragmentet.

Kotlin filen til MainActivity vil ved brug af fragmenter, kan forblive urørt, hvis alt blive kørt igennem fragmenterne.

Den endelige struktur, vil være som vist på Figur 6, hvor Kotlin filerne er i Java mappen, xml filerne til layoutet er i res(resources)->layout mappen, og til sidst er fragmentnavigationsdiagrammet i res->navigation mappen.



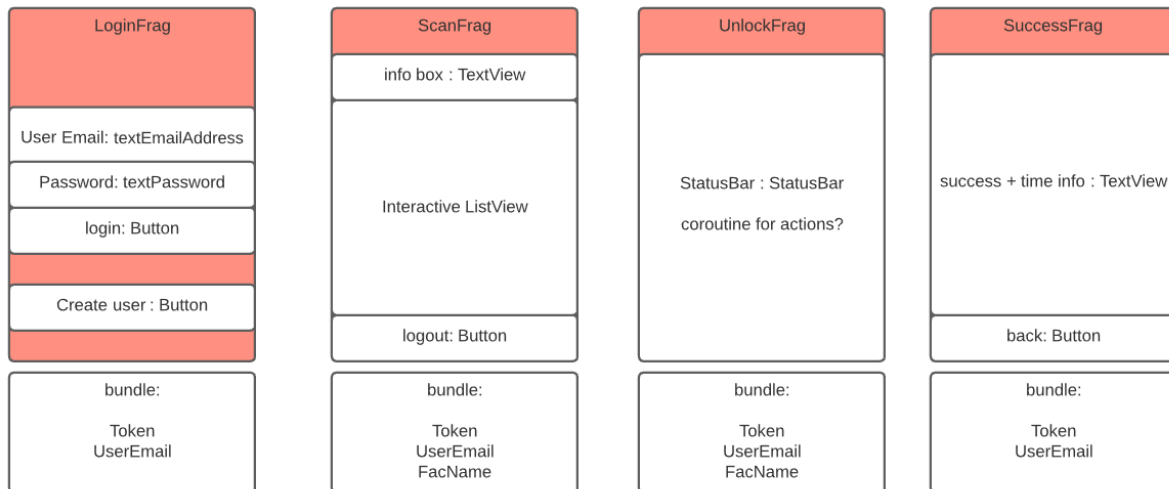
Figur 6 mappe struktur

I dette projekt, vil der være 4 fragmenter, hvor der skal passeres data imellem de forskellige fragmenter.

Fragmenterne vil være som på Figur 7, og er sat op således:

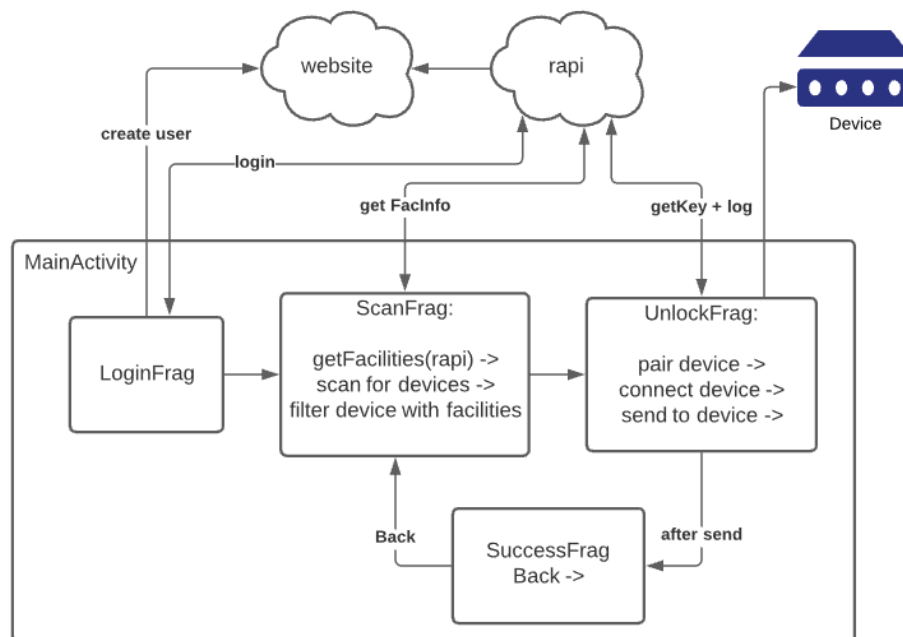
- LoginFrag:
 - Håndterer login af bruger, ved at tjekke op mod restAPI'en.
 - Kan dirigere hen til hjemmesidens "create new user" side.
- ScanFrag:
 - Håndterer scanning af devices i nærheden, og filterer dem ud, hvis de ikke har specielle keywords
- UnlockFrag:
 - Håndtering af Pair device
 - Håndtering af Connect til et device
 - Hente nøgle fra restAPI til device, og sende log meddelelse til website gennem restAPI
 - Sende besked til device
 - Håndtering af Disconnect fra device
 - Håndtering af Unpair device
- SuccessFrag
 - Står for at give status til bruger, omkring device.
 - Fx. Dør til Y facilitet er åben i X tid.

På Figur 7, vil der også kunne ses hvad der skal puttes i et bundle fra et fragment og sendes med over til det næste fragment.



Figur 7 Fragments, deres opsætning og hvilke data de skal sende med

For at give et bedre overblik er der udviklet et interface design over PoC med fokus på mobilappens side, se Figur 8, som vil kunne give en idé over flowet i appen, og hvor i appen forskellige eksterne elementer bliver kontaktet.



Figur 8 Mobil interface analyse

Konklusion

Teamet har analyseret de elementer, der skal til for at lave en mobilapp struktur og hvad elementerne skal indeholde for at fylde deres rolle for funktionaliteten. Der er udviklet en interfaceanalyse med fokus på mobilappen.