

StudHub

Bjørge Seim Øvstedal: 224910

Mikkel Haraldson Hiorthøy: 224926

Bjørnar Os: 145079

Elias Nesheim: 224924

11.12.2020

Technical

Contribution Claim

	Concept	Programming	Design	Report
Bjørnar Os	25%	25%	25%	25%
Bjørge Øvstedal	25%	25%	25%	25%
Elias Nesheim	25%	25%	25%	25%
Mikkel Hiorthøy	25%	25%	25%	25%

Chosen Programming features

Features	Activity	Status
Brukerprofil – Registrer, logg inn, profil, profilinnstillinger	Registrer.java / LoggInn.java / ProfilActivity.java / ProfilInnstillinger.java	Ferdig
Deling av innhold – Innlegg, kommentarer	PostActivity.java / enkeltInnlegg.java	Ferdig
Lokasjonstjenester – GPS, lydløs	GpsActivity.java / Constans.java / LocationService.java	Ferdig
Notifikasjoner – Kommentar på innlegg		Ikke ferdig
Navigation drawer		Ferdig
Database		Ferdig
Fragments	Bio, It, Jus, Mark, Med, Ped - Fragment	Ferdig

App Running essentials:

For å kjøre appen trenger man ikke være logget inn eller være registrert. Om bruker vil legge inn innlegg eller kommentere på innlegg må han være registrert og innlogget. Da får han også tilgang til profilside.

Innhold

Technical	2
Contribution Claim	2
Chosen Programming features	2
App Running essentials:	2
Introduction	4
Problem space	4
Problem statement	5
Competitors and Innovation	5
Project organization	7
Reasons behind choosing your project	8
Development method	8
Requirements Analysis and design	9
Functional requirements	9
Non-Functional requirements	9
System architecture	11
Information Architecture/ Flow diagram	12
Working Process	13
Task Distribution	13
Technical description	15
Components	18
End product	20
Installation and Running Instructions	20
App functionalities	20
Evaluation Conclusion	23
Future improvements	24
Individual reflection and experience	25
References:	27

Introduction

StudentHub er et forum der studenter kan dele sine studierelaterte problemer med andre studenter som har like fag. Appen er programmert i Android Studio ved hjelp av java og sql.

Problem space

Tanken bak appen er at studenter skal ha et eget område der man kan hjelpe hverandre med krevende oppgaver. Det skal være et hyggelig sted der man ikke trenger å tenke på så mye annet enn å diskutere et tema/oppgave.

Dette vil vi utføre ved å la en student registrere en bruker, der man gir informasjon om hvilket fakultet man studerer ved og hvilket fag man tar. Som en innlogget bruker vil vi gi mulighet for å dele problemer i dedikerte grupper. Gruppene vil for eksempel være inndelt i "Økonomi", "IT" og "Juss". Appen vil være åpen for alle studenter, uavhengig av fakultet. Det gir mulighet for interessante diskusjoner og flere løsninger på et problem.

Informasjon må kunne lagres slik at innleggene og brukerne ikke forsvinner etter bruk. Det første steget vil da være å opprette en server og en database som lagrer alle brukere og deres innlegg, samt andre ting som bilder og innlegg til en kommentar. Når dette er gjort kan man begynne å implementere funksjonene for å registrere og logge inn. Så kan man begynne å implementere hovedfunksjonene, som å lage innlegg og kommentarer. Sluttstegene vil være å legge til profilinnstillinger, der en bruker kan gjøre endringer på sin profil. Så kan man implementere andre alternative requirements.

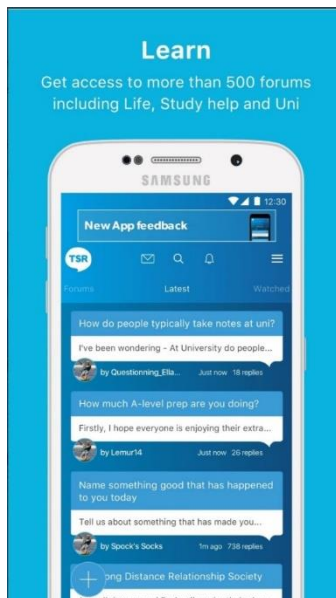
Problem statement

Ideen bak denne appen kom fra våre personlige erfaringer med å trenge hjelp og tips i studieemner vi har vært igjennom i vår studietid. Ved flere anledninger har vi vært usikre på hvordan ting skal gjøres i forskjellige fag, og vi har ofte enten måtte ta kontakt med forelesere eller andre medstudenter og studenter som går på trinn over oss. Dette kan ta tid og kan resultere i at man ikke får noe klart svar på det man lurte på.

Med denne appen har vi som mål å skape et større samarbeidsmiljø blant studenter. Appen skal gi studenter en enkel og brukervennlig metode for å stille spørsmål til andre studenter som igjen kan dele sine erfaringer og tips. Dette kan være alt fra tekniske tips til struktur og utforming av diverse fagrelaterte ting. Studenter skal altså ikke bruke appen til å gi og få direkte svar og løsninger på oppgaver, men heller hjelpe til med elementer rundt emne/faget som kan være uklare.

Competitors and Innovation

The Student Room er en lignende app hvor studenter kan snakke med hverandre og diskutere diverse emner. Denne appen har derimot et større omfang, da den går inn på andre ting som ikke er direkte skolerelatert. I vår app har vi valgt å bare fokusere på skolerelaterte ting, da vi føler at dette passer omfanget og tidsrommet for dette prosjektet.



*The Student Room: forum app for utdanning og andre forhold som kan interessere studenter.
(https://play.google.com/store/apps/details?id=com.thestudentroom.discussions&hl=en_US&gl=US)*

En annen lignende app er Student World Forum for JNTU. Dette er en forum app som skal samle studenter, forelesere og universiteter. Denne appen har også et større omfang enn vår, da den inkluderer funksjoner som f. eks omhandler forelesninger og eksamener.



*Student World Forum: app for studenter og forelesere m.m.
(<https://play.google.com/store/apps/details?id=com.jntuworldforum.jwfandroid.appoverview>)*

Project organization

Rollefordelingen ble diskutert innad i gruppen og ble fordelt i følgende tabell. Rollene viser hva den enkelte hadde som hovedansvar, men oppgavene som ble fordelt gjaldt både front-end og back-end.

Rolle	Hovedansvarlig	Medlemmer
Prosjektleder	Bjørge	Bjørnar, Mikkel, Elias
Design	Mikkel	Bjørnar
Front-end	Bjørnar	Mikkel, Elias
Back-end	Elias, Bjørge	Mikkel

Prosjektleder

Prosjektlederens ansvar er å ha oppfølging av oppgavene som skal utføres. Det er her viktig at medlemmene oppfyller kravene som ble satt i planleggingsfasen og designer fasen. Videre har prosjektlederen ansvar for problemer i gruppen som gjelder både tekniske, sosiale og prosjekt problemer. I tillegg har prosjektleder et ansvar for å få gruppen motivert og at oppgavene blir ferdig til fastsatte tidspunkt.

Design

Hovedoppgaven til den ansvarlige for design, er å passe på at designet som ble diskutert i designfasen blir riktig utført. I tillegg har ansvarlige ansvaret med å implementere designet.

Front-end

Front-end ansvarlige har ansvar for brukergrensesnittet. Her er det snakk om oppgaver med tanke på hvordan brukerne opplever sluttproduktet og hvor enkelt eller vanskelig den er å bruke. I dette prosjektet vil de fleste av oppgavene omhandle navigasjon, knapper, veksling mellom aktiviteter- og fragments, og visuelle presentasjoner.

Back-end

Som back-end ansvarlig, har medlemmer som oppgave å snakke med databasen via appen. Prosjektet omfatter en app som fungerer som et forum, og det er da mye data som skal lagres og vises for brukerne. Oppgavene som utfører lagring, er for eksempel registrering og endring av informasjon av en bruker. Henting av data utføres hvor det skal vises informasjon som for eksempel innleggene til brukere og selve informasjonen om dem.

Reasons behind choosing your project

Prosjektideen ble til ved at vi diskuterte i gruppen hvilke funksjoner som var interessante for hvert enkelt medlem. Det var også ønske om at ideen skulle være noe som kunne brukes i den virkelige verden, ikke bare for gruppen sin del, og at den var nyttig. Ideen ble også til ved at det var fokus på at den skulle være nyttig i form av skolegang og læring.

Development method

For dette prosjektet valgte vi å bruke Sashimi waterfall metoden, ettersom denne lot oss være mer fleksible i forhold til arbeidstid og mengde. Ettersom prosjektet ble påbegynt i midten av 2020, var det klart at store deler av arbeidet ville bli gjort individuelt. På grunn av dette var det viktig for oss å ha en utviklingsmetode som lot oss, i så stor grad som mulig, jobbe individuelt og i vår egen fart. Vi har funnet at Sashimi waterfall gir individer, i relativt stor grad, muligheten til å kunne jobbe på egne vilkår og initiativ, og med relativt lite "oversight".

Problemer kan oppstå hvis enkelte medlemmer ikke er sikre på hva de gjør eller skal gjøre, og det kan være vanskelig å holde oversikten over hvordan prosjektet egentlig ligger an. Generelt har dette gått bra, men det har vært tilfeller av ineffektivt arbeid på grunn av manglende informasjon. Til tross for dette har Sashimi waterfall fungert bra for oss, og har hjulpet oss å jobbe effektivt selv om vi har jobbet individuelt.

Requirements Analysis and design

Functional requirements

Opprette en bruker
Lagre brukeres informasjon
Redigere profil
Opprette innlegg
Kommentere innlegg
GPS funksjon som sporer brukers lokasjon og setter telefonen på lydløs i valgt sone
Varsler som opprettes når en bruker kommenterer et innlegg

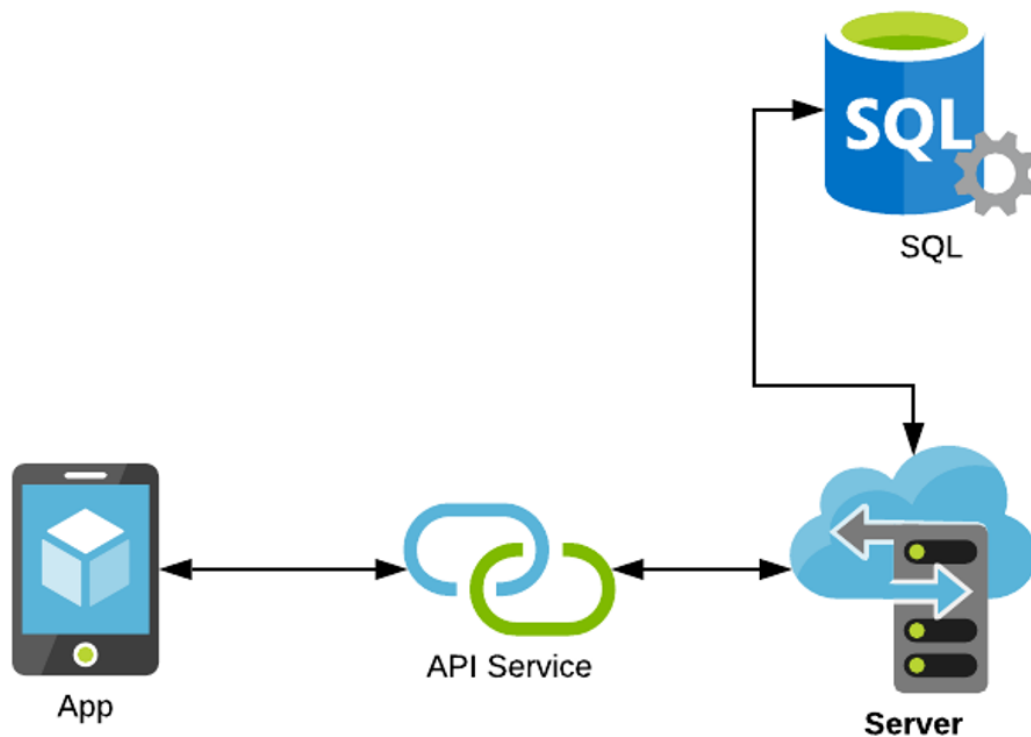
Non-Functional requirements

Appen skal fungere på 95-97% av android enheter (APler fra 19 til 30)
Appen skal være mulig å navigere i både portrait og landscape
Appen skal være skalerbar så den passer til flere telefoner
Appen skal gi tilbakemelding for utførte aktiviteter og problemer som oppstår
Appen skal være enkel og intuitiv å navigere seg gjennom

MoSCoW

Requirements	Must	Should	Could	Won't
Brukerprofil – opprette bruker, lagre informasjon	X			
Deling av innhold – opprette innlegg og kommentarer	X			
Stedstjenester – GPS som sporer lokasjon og setter på lydløs	X			
Notifikasjoner – Varsel når man får svar på forumtråd	X			
Legge til profilbilde		X		
Anonym posting og kommentering			X	
Mørkt tema (gjør om lys bakgrunn til mørk bakgrunn)			X	
Upvotes på kommentarer			X	
Startside er lik faget man har registrert (IT studenter har IT som startside)			X	
Se andre brukeres profiler		X		
Søke på brukere				X

System architecture



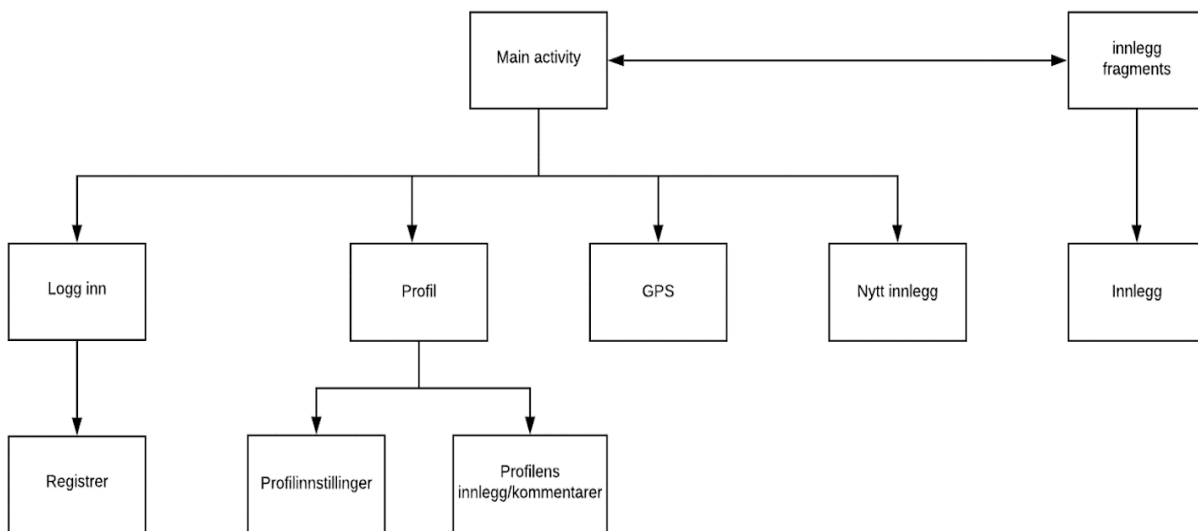
I denne figuren kan vi se hvordan appen får informasjon fra databasen. “Volly” biblioteket blir brukt for å sende https requests gjennom java, og spør serveren (13.49.175.230) om å kjøre en php skript. For eksempel, i itFragment, sender appen en forespørsel til <http://13.49.175.230/getPosts.php?linje=1>

En kan klikke på denne linken for å se informasjonen som blir printet. `getPosts.php` kan en finne i vedleggene til dokumentet, som også inneholder alle php skriptene som er brukt i appen. Når php blir kjørt bruker den `mysqli` til å spør `mySQL` databasen, som er på den samme serveren.

```
$sql = mysqli_query($conn, "SELECT inn_id, innBruker, tittel, inn_tekst, inn_tid, brukernavn, fag_id FROM innlegg, bruker WHERE innlegg.fag_id = '$linje' AND innlegg.innBruker = bruker_id;");
```

Serveren er en AWS ec2 instance som kjører ubuntu-server (noGUI) og er i zone “eu-north-1” som ligger i stockholm. Serveren kjører Apache som backend og skriptene ligger i mappen `/var/www/html`

Information Architecture/ Flow diagram



Her kan en se strukturen for programmet og hvordan den er koblet sammen. Alt leder tilbake til mainActivity, og informasjonsflyten følger for det meste den same strukturen, unntaket er fragments som får informasjon ved hjelp av SharedPreferences.

Working Process

Fra tidligere erfaringer med å jobbe i grupper, har vi funnet ut at det er best å ikke ta i bruk for mange verktøy for prosjektledelse og utvikling. Dette er fordi det er noen verktøy som ender opp med å ikke bli brukt.

Vi har tatt i bruk Trello for å distribuere oppgaver og dele enkle filer relatert til oppgavene. Som f.eks. oppgaven wireframes, inneholder også bildene som hører til oppgaven. Med Trello har det blitt oversiktlig hvilke oppgaver som er utført, og hvilke oppgaver som gjenstår. Trello er også brukt til å sette opp møtetider og oppsummere innholdet av møtene.

For å kommunisere med gruppen har vi opprettet gruppe på Facebook Messenger og discord. Dette er de kommunikasjonsverktøyene vi har erfaring med å bruke. Disse er blitt brukt til å avtale møter og diskutere problemer. Etter hvert ut i semesteret har vi hatt alle møtene våre over discord, der vi deler skjermer for å hjelpe hverandre med problemer i koden.

GitHub er blitt brukt for å samle sammen prosjektet og lagre flere versjoner av prosjektet. Skulle det da skje at en nyere versjon ikke fungerer som den skal, så kan man rollback til en eldre versjon. For samarbeid med rapporten har vi brukt Google Docs, da det er lett tilgjengelig og enkelt å bruke. Siden flere kan gjøre endringer i sanntid er det effektivt og man slipper å samle sammen dokumenter.

Task Distribution

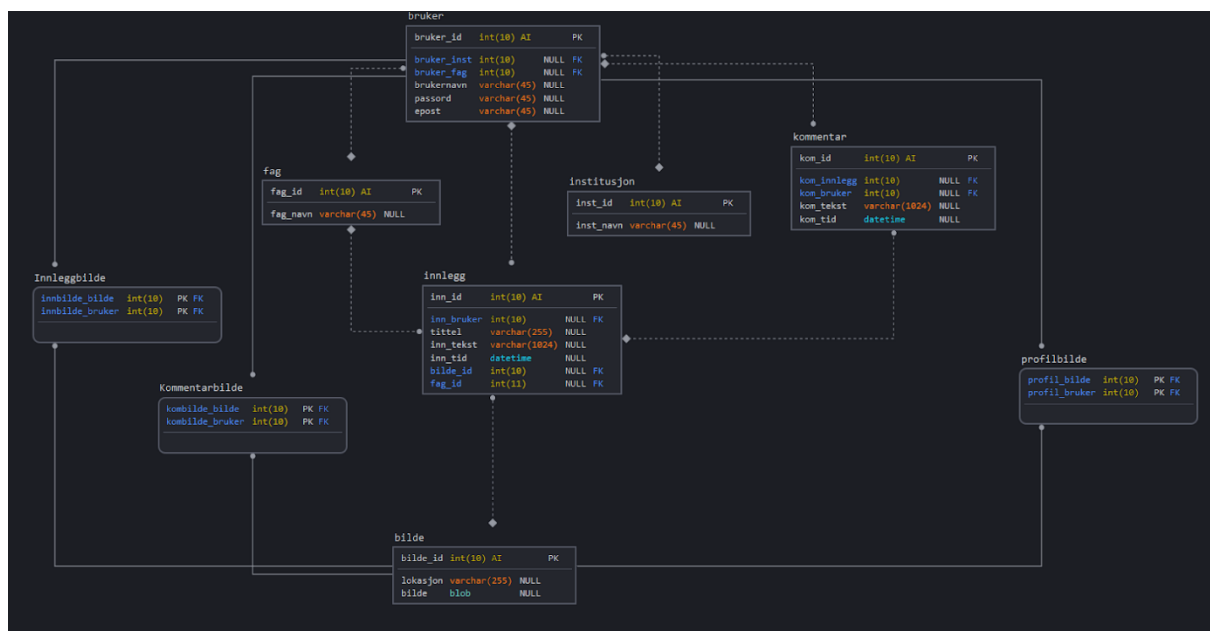
Oppgave	Utført av	Hjelper
Wireframes	Alle	
Databasestruktur	Alle	
Tilkobling til server	Elias	
PHP scripts	Elias	alle

Aktivitet for innlogging	Bjørnar	Bjørge
Aktivitet for registrering	Bjørnar	Bjørge
Framside med fragments for hver linje	Bjørge	
Navigation drawer	Bjørge	Bjørnar
Aktivitet for profil	Mikkel	
Aktivitet for profilinnstillinger	Mikkel	
Aktivitet for å opprette innlegg	Elias	
Aktivitet for å skrive kommentarer	Elias	
Aktivitet for å sette GPS	Bjørge	Bjørnar
Landscape og portrait mode	Mikkel	
Debugging	Alle	
Introduksjon	Bjørnar	Mikkel
Problem space/statement	Bjørnar	Mikkel
Competitors and innovation	Mikkel	
Project organization	Bjørge	
Requirements and Innovation	Bjørnar	
Working Process	Bjørnar	
Technical Description	Alle	
Components	Mikkel	
End product	Bjørge	
Evaluation Conclusion	Alle	

Technical description

Database

Appen bruker MySQL som RDBMS. For å kommunisere informasjon fra og til databasen har vi laget en egen API basert på PHP, som appen kan nå ved å bruke HTTP biblioteket volley. Slik kan vi hente og legge inn data via PHP skript. Alle aktivitetene som som trenger kontakt med databasen sender slike forespørsler.



Her kan en se databasestrukturen, som stort sett har vært uendret siden den første versjonen.

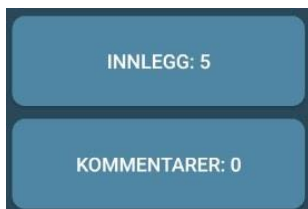
Navigasjon og brukerinteraksjon

Brukeren kan navigere seg rundt om i appen via toolbaren, knapper og lenker. I toolbaren ligger det på de fleste sider en navigation drawer. På enkelte sider ligger det bare en tilbakeknapp her da vi ikke følte det var nødvendig å ha navigation drawer på alle sider. Et eksempel på dette er hvis brukeren går via profilsiden til aktiviteten over innleggene han har lagt ut. På denne aktiviteten ligger det en tilbakeknapp i toolbaren, da dette gjør det enkelt for brukeren og komme tilbake til profilsiden. Vi har derfor med to layout filer for toolbar, en med navigation drawer og en enkel med tilbakeknapp. På hovedsiden/forsiden har vi også med et TabLayout. Tabene her navigerer gjennom forskjellige fragments. Fragmentene inneholder de forskjellige fagenes innlegg.

Brukerprofil

Brukerprofilen opprettes i aktiviteten Registrer.java. Infoen brukeren skriver inn her blir lagt inn databasen. Når brukeren er registrert får han tilgang til et valg for profilsiden i navigation drawer. Dette valget åpner en ny aktivitet, ProfilActivity.java. Brukernavnet til brukeren som er logget inn blir lagt med når aktiviteten starter, og aktiviteten inneholder følgende:

- Info om bruker som hentes fra databasen via PHP og forespørsler til serveren.
- Profilbilde som lastes ned og vises fra en mappe på serveren. Dette gjøres ved hjelp av Picasso som er et bibliotek for nedlasting og caching av bilder. Om det ikke er lagt til noe profilbilde vises et midlertidig png fra drawable mappen.
- Hvor mange innlegg brukeren har lagt til. Dette vises fra databasen, med tekst og tall på en knapp. Tallet som vises her er en count gjort via PHP. Knappen går til en ny aktivitet, Brukerinnlegg.java. Denne aktiviteten viser alle innleggene til brukeren som ligger i databasen, og her kan man også gå inn på enkeltinnlegg.



- Hvor mange kommentarer brukeren har lagt til, med samme oppsett/struktur som innlegg.
- Knapp for profilinnstillinger. Denne knappen går til en ny aktivitet, Profilinnstillinger.java. Her kan man endre på all infoen som ligger i databasen, med unntak av brukernavn. Man kan også legge til profilbilde. Dette gjøres ved å trykke på en knapp som har et kamera ikon. Profilbilde blir lagret i en mappe på serveren. Her kommer det også opp tillatelser som spør om brukeren vil la appen få tilgang til media mappen.



All dataen som hentes fra databasen vises ved hjelp av SQL spørringer i PHP script som ligger på serveren.

GPS services

Tjenesten kjører i bakgrunnen og vil alltid følge med på hvor enheten befinner seg i forhold til sirkelen. Dette gjøres ved hjelp av `FOREGROUND_SERVICE` som benyttes i manifestet. Brukeren kan når som helst gå inn på aktivitet igjen får å endre størrelse eller flytte på sirkelen. Tillatelse som blir brukt ligger i `AndroidManifest.xml` under taggen `uses-permission`. GPS aktiviteten bruker `ACCESS_FINE_LOCATION`, `FOREGROUND_SERVICE` og `ACCESS_NOTIFICATION_POLICY`. I manifestet er det også en tagg for service hvor informasjonen om tjenesten er lagret. Taggen meta-data inneholder api-key for google maps som blir brukt i prosjektet. Implementation som blir brukt er `gms.play.services`, både map og location. Disse befinner seg under dependencies i `build.gradle` filen.

Denne aktiviteten heter `GpsActivity.java` og bruker `Constans.java` klassen til noen variabler, mens `LocationService.java` er klassen som lokaliserer og sender data om enheten til aktiviteten. Brukeren må ha installert Google Play services for å benytte både kart og lokasjon tjenesten, og om alt av tillatelser og Google Play services er oppdatert vil kartet og funksjonalitet fungere.

Noen av metodene, og koden for de, er merket som rød eller feil på grunn av at det ikke er lagt inn tillatelse for de. Dette gjøres i starten av aktivitet før brukeren kan bruke funksjonen, og det er ikke nødvendig å legge inn tillatelsen på nytt.

GPS tjenesten fungerer på alle apier, men man må manuelt inn og slå på gps lokasjon i innstillinger på enheten for api 19 og under. Brukeren vil ikke få opp automatisk tillatelse om brukeren ikke har skrudd på gps/lokasjon på enheten sin.

GPS tjenesten ble først utviklet med `setOnMyLocationChangeListener`, men siden dette interfacet er deprecated, ble det byttet ut med `FusedLocationProviderApi`. Dette gav både raskere og mer presis lokalisering av enheten samtidig som den brukte mindre strøm.

Navigation Drawer

Appen har med en navigation drawer for lettere navigering. Denne inneholder en switch metode/funksjon som starter forskjellige aktiviteter/metoder. Om bruker ikke er logget inn kommer det opp en logg inn knapp som går til en ny aktivitet,

LoggInn.java. Om brukeren er logget inn kommer det opp følgende valg:

- Profil: dette valget starter en ny aktivitet for profilsiden.
- Innlegg: dette valget starter en ny aktivitet for å legge til et nytt innlegg.
- GPS lydløs: dette valget starter en ny aktivitet for lokasjons tjeneste og lydløs funksjon.
- Logg Ut: dette valget starter en funksjon som logger brukeren ut.

Fragments

På fremsiden av appen navigeres det mellom fragments via et TabLayout. Her vises det et fragment for hvert fag i de forskjellige TabItemene. Hvert fragment inneholder alle innlegg fra et enkelt fag.

Components

Verktøy vi har brukt i denne appen er:

- MySQL: for å kjøre spørringer mot databasen. Vi valgte å bruke en ekstern løsning da vi føler dette representerer et mer realistisk bilde av hvordan en forum app skal fungere. Med en lokal database ville ikke brukere kunne se hverandres innlegg. Det er derimot nyttig for backup om internett/server er nede.
- PHP: ble brukt for å kunne koble sammen med server. Alle spørringer ble kjørt gjennom PHP skript.
- Xampp: for å teste database tilkobling og spørringer lokalt.
- phpmyadmin: ble brukt for å administrere databasen.
- FileZilla: Lastet opp filer til server

Bibliotek vi har brukt i denne appen er:

- Volley: et HTTP-bibliotek som gjør nettverksforespørsler for appen enklere og raskere.

Dette ble brukt for å kunne hente og legge inn data i databasen på serveren. Vi prøvde flere forskjellige løsninger for å få til en ekstern databaseløsning, men endte på denne da den viste seg å være den raskeste og mest effektive løsningen. Styrker bak dette biblioteket var bl. a automatisk scheduling av nettverksforespørsler og flere samtidige nettverkstilkoblinger. I tillegg har det gode funksjonaliteter for debugging.

problem som har oppstått: Volley har i noen tilfeller skrevet fire mellomrom foran respons "stringen", dette har vi løst ved å bruke linjen: if (

- Picasso: Et bibliotek for å laste ned og cache/vise bilder. Dette brukte vi for å vise bilder som ligger i mappe på serveren. Fordeler med dette biblioteket er at det kan transformere bilder til å passe bedre i layout/ImageViews og redusere minnestørrelsen.
- Dexter: Et bibliotek som forenkler prosessen med å be om tillatelser ved kjøretid. Dette ble brukt når vi skulle be om tillatelser i forbindelse med opplasting av profilbilde. Biblioteket gjør det lettere å kode hvor tillatelser skal komme.
- Google play services: En bakgrunnstjeneste og API pakke for Android-enheter. Ble brukt for å kunne programmere GPS funksjonen. Fordeler med denne er at man får tilgang til et verdenskart og plassering av bruker.

End product

Installation and Running Instructions

Appen er som en helt vanlig app der du må registrere deg for å få tilgang til å poste innlegg eller å kommentere. Brukere som ikke er registrert, vil fortsatt ha tilgang til GPS funksjonen og til å se andre sine innlegg. Først når brukeren har registrert seg får han nye valg i hamburgermenyen opp til venstre.

Appen er generelt enkel å bruke, men krever en del tillatelser til å bruke noen av funksjonene. Disse tillatelsene gjelder notifikasjons- og lokasjon tillatelser for GPS, og ekstern lagring tillatelser for profil siden.

App functionalities

Brukerprofil

For å lage en brukerprofil må brukeren registrere seg. For å registrere seg må brukeren skrive inn følgende:

- Brukernavn
- E-post
- Passord
- Hvilket universitet man går på
- Hvilket fag man tar

Når brukeren er registrert får han tilgang til et valg for profilsiden i navigation drawer.

Dette valget går til profilsiden som viser:

- Profilbilde
- Brukerinfo
- Hvor mange innlegg brukeren har lagt til, med knapp som går til en side som viser alle innleggene til brukeren. På denne siden kan brukeren også gå inn på et enkelt innlegg.

- Hvor mange kommentarer brukeren har lagt til, med knapp som går til en side som viser alle kommentarene til brukeren, og hvilket innlegg brukeren har kommentert på.
- Knapp for profilinnstillinger, hvor brukeren kan endre infoen sin og laste opp profilbilde.

For å laste opp profilbilde må brukeren trykke på knappen med et kameraikon, og godkjenne at appen har tillatelse til å gå inn på ekstern lagring. Når brukeren endrer info eller profilbilde kommer det opp meldinger om hva som har blitt endre, eventuelt error meldinger.

Innlegg

Da appen vår er et forum, er innlegg en viktig del av den. Brukere kan legge inn innlegg og kommentere på innlegg om de er registrert i appen. Om de ikke er registrert kan de bare lese innlegg. For å legge inn nytt innlegg må brukeren trykke på innlegg valget med et pluss symbol i navigation drawer. Da kommer man inn på en side hvor man må skrive inn tittel og tekst, og trykke på knappen for å laste opp innlegget. Innlegg blir lagt ut på forsiden og bruker kan navigere mellom hvilket fag de vil se innlegg om via tabmenyen. Her kan man også gå inn på enkelt innlegg. Denne er ikke helt optimalisert for landscape mode.

Kommentarer

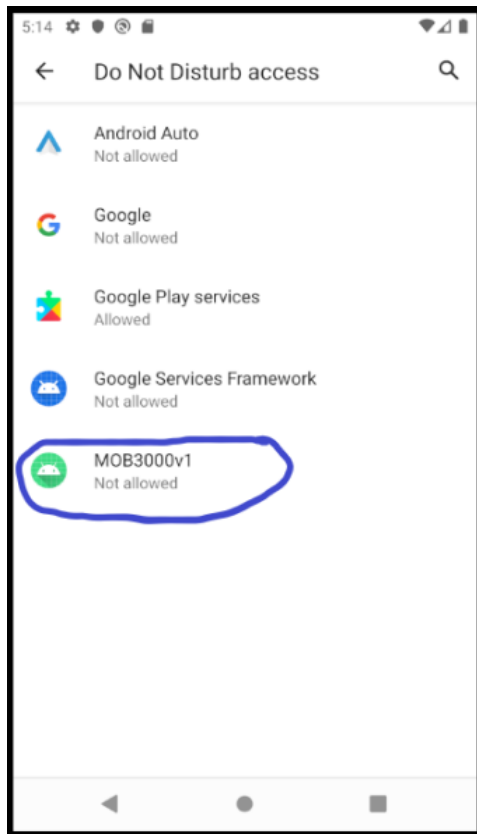
Etter man har oppretter innlegg så har alle brukere mulighet for å kommentere disse. Kommentering skjer ved å trykke på et innlegg, så blir man sendt til en ny side med innlegget man har trykt på og man får muligheten til å skrive kommentar og publisere. Man kan så se kommentarene under innleggene.

GPS lydløs

GPS location er en av funksjonen som tilbys i appen. Brukeren kan ved hjelp av denne funksjonen markere et område på kartet som blir en sone hvor mobilen blir slått på lydløs. Brukeren trykker på kartet der midten av sirkelen skal være, og kan ved hjelp av scrollbar justere radiusen på sirkelen. Sirkelen har et overlegg over kartet hvor man ser området tydelig. Den er fortsatt litt gjennomsiktig så det er mulig å se hva som ligger bak. Når enheten er innenfor sirkelen, vil enheten automatisk slå

seg på lydløs. Går brukeren ut fra sirkelen, vil mobilen slå seg tilbake til den modusen den var på før enheten befant seg i sirkelen.

Når brukeren først åpner GpsActivity.java vil den automatisk bli sendt til Google Play Store om ikke enheten har oppdatert Google Play services til versjon 13400000 eller høyere. Når brukeren har versjonen som trengs, vil brukeren få spørsmål om å tillate do not disturb access for appen. Dette er vist i bildet under.



Etter at do not disturb access har blitt godkjent, vil brukeren kunne se kartet. Her kan den bla rundt, men lokasjonen er ikke på før brukeren trykker start. Når brukeren trykker start, vil en ny tillatelse komme frem. Her må brukeren tillate at appen kan bruke lokasjonen til brukeren. Først når dette er godkjent, vil lokasjonen starte og brukeren kan se hvor han er på kartet. En liten knapp øverst til høyre blir synlig, og brukeren får en varsel om at LocationService har startet. Knappen oppe til høyre fører kamera direkte på brukerens lokasjon. Videre kan brukeren trykke hvor som helst på kartet, hvor han ønsker å ha sonen for lydløs. Baren til høyre på skjermen tillater brukeren å justere på størrelsen på sirkelen.

Tjenesten kjører i bakgrunnen og vil alltid følge med på hvor enheten befinner seg i forhold til sirkelen. Brukeren kan når som helst gå inn på aktivitet igjen får å endre størrelse eller flytte på sirkelen.

Evaluation Conclusion

Det som er ferdig og det som ikke er ferdig:

Features	Status
Brukerprofil – Registrer, logg inn, profil, profilinnstillinger	Ferdig
Deling av innhold – Innlegg, kommentarer	Ferdig
Lokasjonstjenester – GPS, lydløs	Ferdig
Notifikasjoner – Kommentar på innlegg	Ikke ferdig
Navigation drawer	Ferdig
Database	Ferdig
Fragments	Ferdig

Brukerprofil er ferdig, da stegene er å registrere, logge inn, vise profil og gjøre endringer på profil. Lokasjonstjenester har vi løst med GPS sporing og muligheten å kunne velge et område der telefonen blir satt på lydløs.

Deling av innhold har vi misforstått litt fra start av. Vi har tenkt at det gjelder når brukerne kan dele innlegg som andre brukere kan se, men ser nå i etterkant at det gjelder over plattformer. Vi tenker dette skal gå greit da vi har med flere andre features i appen. Deling av innhold kommer fortsatt med som en feature som gjelder for innlegg og kommentarer.

Notifikasjoner var noe vi ønsket å utvikle, men de andre oppgavene har vært for tidkrevende til å kunne implementere denne featuren. XMLen for notifikasjon er laget og ligger i navigasjonen som en bjelle. Tanken var at når noen kommenterer ditt innlegg så skulle det generere en notifikasjon på telefonen og i appen.

Future improvements

Det er en del funksjoner som vi ønsket skulle være med i appen. En av disse var dark mode, eller dark theme. Dette ville gjøre temaet på appen mørkere for brukere som ønsket et mørkere alternativ av designet. En annen funksjon vi ønsker skulle vært med er at brukeren skal kunne poste og kommentere anonymt. Dette er for å gjøre brukere mer komfortabel med å skrive innlegg og kommentere. I tillegg ønsket vi at det skulle gå an å trykke "liker" eller upvote andre sine kommentarer. Dette vil legge de mest relevante kommentarene øverst og det vil gjøre at flere kan bidra ved å velge den "rette" løsningen. I innlegg ville vi at brukeren skulle kunne legge til bilde i et innlegg, som gjør det lettere å dele problemer. GPS aktivitet sin XML er foreløpig simpel og minimalistisk. Designet kunne blitt bedre hvor navigasjon eller tilbakeknapp blir lagt til. I tillegg var det ønsket å legge til en switch på menu item'et til GPS. Her kunne brukeren enkelt skru av eller på gps lokalisering uten å gå inn på selve aktiviteten. Det er også en måte for å se om lokaliseringen er på.

Databasen har tatt mye av fokuset i dette prosjektet, og det har sinket flere oppgaver. Det er litt synd, siden det er en stor del av årsaken til at vi ikke har kunnet utføre det vi har ønsket. Samtidig har vi lært mye om dette og det er noe man kan ta med videre.

Det ble forsøkt å lage egne klasser for kode som gjentar seg. Det er foreløpig en del redundant kode som toolbar og navigation drawer. Disse burde bli lagt til i egne klasser som inkluderes der navigasjonen trengs.

Bugs:

kommenter uten å være innlogget. Det har oppstått en bugg nå på slutten hvor hvis en bruker har logget in tidligere, men så logget ut, er det fortsatt mulig å kommentere på innlegg. Planen var at det skulle vær mulig å kommentere som anonym hvis en ikke var innlogget. Men når vi nå prøver å legge til denne featuren ender den opp med å ødelegge helle kommenterings funksjonen, om en er logget in eller ikke. Hadde vi hatt mer tid ville vi fikset dette.

Profilbug: Når man bruker mobilen sin tilbakeknapp, blir bruker sendt til den aktiviteten de var på forrige gang. Dette gjør at man ikke alltid kommer tilbake til

hovedsiden ved å trykke på denne. Vi har derfor lagt til en tilbakeknapp på disse sidenen. I tillegg kan hjem knappen i NavDrawer brukes til å komme tilbake til hovedsiden. Om vi hadde hatt bedre tid hadde livssyklusen til appen vært forbedret her.

Individual reflection and experience

Bjørge:

Tidligere erfaring har hjulpet med både prosjektplanlegging og dokumentering for prosjektet. Vi ønsket å ha en database som alle brukerne var koblet til, i stedet for firebase som var lokal database på enheten. Dette førte til en del arbeid der en av de største utfordringene var å lage vår egen API. Denne ble laget ved hjelp av PHP som snakket med databasen. Sidene ble hostet ved hjelp av apache hvor da android enheten fikk og sendte informasjonen fra. Dette var veldig lærerikt hvor jeg lærte mye hvordan en android app kan kommunisere med en ekstern database. Metodene var svært annerledes i forhold til tidligere fag hvor jeg har lært hvordan en java applikasjon kommuniserer direkte med en database uten API.

Selv om både jeg og resten av gruppen har erfaring med språket java, var det fortsatt en del utfordringer med å lage en app. Android Studio, og det å lage appen i seg selv, var svært krevende i form av at mye av koden var basert på metoder og biblioteker. Til vanlig har jeg skrevet kode uten noen form for biblioteker og dens metoder og API, så utfordringen var da å forstå hva slags metoder som trengst og hvordan de fungerte. Det var også nytt for meg å skrive funksjonell kode der man jobbet både med funksjonalitet og utseende. XML ble her brukt som styling og var nytt for oss alle. XML var ganske så greit å forstå da det lignet veldig på css og html fra tidligere fag. Android Studio var her ganske enkelt å bruke siden du kunne søke på widgets du ønsket å ha med. I tillegg kunne du se endringen direkte, noe som gjorde enkelt å greit å lage layouten til de forskjellige aktivitetene.

Prosjektet har lært meg mye om hvordan det å lage en app. Ideen ga nok utfordringer til at det ble jobbet jevnt utover semesteret.

Bjørnar:

Dette har vore et interessant prosjekt, med en del små utfordringer som har tatt mye tid. Å få Android Studio til å fungere og emulatoren til å kjøre er noe av det, men også å få en kobling til databasen. I dette prosjektet har jeg lært mye om hvordan man bruker Android studio og java knyttet til det. Har tidligere ikke jobbet med noen biblioteker, så det har vore veldig lærerikt. Som gruppe har vi jobbet mye sammen og hjulpet hverandre når det trengs. Hvis det skulle vært noe forbedringer så er det å planlegge og bruke mer prosjektverktøyene underveis til å lage mindre oppgaver som kan deles. Selv om vi ikke ble helt ferdig med alt vi hadde tenkt, så er jeg godt fornøyd med hvordan prosjektet ble til slutt.

Mikkel:

Gruppearbeidet under prosjektet har vært bra. Vi har hatt god kommunikasjon og samarbeidet mye. Fordelingen av oppgavene har vært jevn, og alle har bidratt med sitt. Under dette prosjektet har jeg lært mye nytt, da jeg aldri har laget en app av et slikt omfang før. Jeg har jobbet en del med java tidligere, og det var interessant å lære nye måter å bruke det på i forbindelse med denne appen. Jeg har også lært mye nytt i forhold til design og xml. Vår eksterne databaseløsning har vært lærerik og interessant å jobbe med, men også krevende til tider. Da jeg har jobbet mye med brukerprofil har jeg måttet jobbe mye med databasen og det har krevd en del testing og prøving for å få alt til å stemme. Jeg testet mye opp mot lokal server (localhost), og skulle nok testet mer mot den eksterne databasen da alt ikke fungerte på samme måte der. Det kunne

Elias:

MOB3000 har vært en veldig lærerik erfaring, og har gitt meg mye trening i programmering, samt å lære raskt. Å gjør MOB3000 samtidig som jeg tok OBJ2000 førte til noen vanskeligheter og mangler i kunnskapen, men jeg tror jeg klarte å gjøre min del av oppgaven allikevel, takket god støtte fra gruppen. XML har vært meget interessant og lære seg, og var kjekt å jobbe med. I starten var det veldig vanskelig å få satt opp database tilkoblingen, men når den var på plass var det stort sett smooth sailing etterpå. Totalt sett er jeg meget fornøyd med besvarelsen, og er litt kry.

References:

Ahlawat, A. A. (2020). *Using multiple Layouts and Views to design a GUI*. Studytonight.
<https://www.studytonight.com/android/hierarchical-arrangement-in-android>

Fragment Navigation Drawer | *CodePath Android Cliffnotes*. (2019). Codepath.
<https://guides.codepath.com/android/fragment-navigation-drawer>

Google APIs for Android. (2019, 6. februar). GoogleMap.OnMyLocationChangeListener. Hentet fra
<https://developers.google.com/android/reference/com/google/android/gms/maps/GoogleMap.OnMyLocationChangeListener>

Google APIs for Android. (2019, 23. september). FusedLocationProviderClient. Hentet fra
<https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient>

Google APIs for Android. (2019, 30. september). Service. Hentet fra
<https://developer.android.com/reference/android/app/Service>

Developers. (2019, 12. november). App permissions best practices. Hentet fra
<https://developer.android.com/training/permissions/usage-notes>

Bibliotek:

Volley overview |. (2020, June 5). Android Developers.
<https://developer.android.com/training/volley>

Picasso. (n.d.). Picasso. Retrieved November 1, 2020, from <https://square.github.io/picasso/>

K.D. (n.d.). *Dexter*. GitHub. Retrieved October 20, 2020, from <https://github.com/Karumi/Dexter>