

Bacheloroppgave i IT og Informasjonssystemer

20.05.2021

Universitetet i Sørøst-Norge

BOP3000

Stian Tonning Flatset

Sigve Eilertsen

Sanan Adnan Maarouf

Erlend Hollund

Forord

Denne rapporten er en del av bacheloroppgaven vår i studiet IT og informasjonssystemer ved Universitetet i Sørøst-Norge (USN), Campus Ringerike. Vi er fire studenter som dette semesteret har laget et sluttprodukt der vi trekker frem kunnskap fra hva vi har lært disse tre årene på studiet. Prosjektet har vært tidkrevende og til tider utfordrende, men ved hjelp av vår oppdragsgiver Caroline Corneliussen har vi også hatt et lærerikt og spennende prosjekt som har engasjert oss. Vi takker for tilliten og tålmodigheten, og håper produktet står til forventningene.

Vi hadde ikke klart å tilrettelegge denne bacheloroppgaven uten råd på veien av veileder Rania El-Gazzar, og vi vil med dette takke for all god støtte og tilbakemelding dette semesteret.

Vi er stolte av det eksterne samarbeidet, det interne gruppesamholdet og sluttproduktet vi leverer.



Stian Tønning Flatset



Sigve Eilertsen



Sanan Adnan Maarouf



Erlend Hollund

Sammendrag

I denne rapporten forklares det hvilket arbeid vi har gjort for å utarbeide en mobilapplikasjon for vår oppdragsgiver Caroline. Hun har gitt oss i oppgave å lage en app som enkelt kan brukes for å dele eget visittkort med andre, samt få andres visittkort i en og samme samling - en app på mobilen kalt Linkus. Vi forklarer hvilke steg vi har tatt for å utarbeide en brukervennlig applikasjon, med en kompleks kodemessig metode, og i tillegg beskrive hvorfor vi har valgt/ikke valgt de funksjonene appen inneholder/ikke inneholder.

I rapporten har vi underveis også beskrevet hva vi kan gjøre for å gi tilfredsstillende oppdragsgivers krav, og hvorfor det vi gjør er tilstrekkelig for appens mål. I rapporten går vi gjennom planleggingsfasen; hvordan vi har valgt det vi har valgt å ta med og begrunnelse for disse beslutningene. Her er det også med estimert tidsbruk for utviklingsprosjektet som vi har gjennomgått dette semesteret. Videre tar vi for oss analysen som går gjennom hvilket kundesegment vi retter oss mot og bruksmønster. Vi beskriver videre designet ved appen og hvordan vi har løst det visuelle aspektet ved en tiltrekkende applikasjon for dagens marked og ikke minst for vår målgruppe. I implementeringen beskriver vi hvordan systemet er bygd opp og hvordan vi har løst utfordringene som har oppstått og oppgavene oppdragsgiver kom med fra start, samt forbedringer fra tilbakemeldinger av veileder og Caroline. Til slutt diskuterer vi prosessen fra start til slutt, der vi forklarer valg av metode av app-utviklingen og interne forhold i gruppen.

Formålet med oppgaven er å lage en applikasjon som står til oppdragsgivers forventninger og veileders krav til utvikling av appen. Vi har brukt Flutter for å lage en så god applikasjon som mulig, og har i denne sammenhengen lært mer om de ulike metodene som kan brukes for å utvikle en velfungerende applikasjon.

Innholdsfortegnelse

Forord	2
Sammendrag	3
1. Introduksjon	6
2. Bakgrunns litteratur	7
2.1 Utviklingsmetode	7
2.2 Prinsipper og uttrykk	8
2.3 Språk	10
2.4 Database	10
2.5 Rammeverk og verktøy	10
3. Prosjektgjennomførelse	12
3.1 Planlegging	13
Utviklingsmetode	19
Arbeidsplan	19
Tidsplan	19
Risikoanalyse	16
Utviklingsspråk og database	19
Verktøy	19
3.2 Analyse	20
Målgruppe	22
Systemkrav	22
Bruksmønster	22
3.3 Design	23
Wireframes	26
High-fidelity prototype	26
Material Design & Universell utforming	25
Databasemodell	26
3.4 Implementasjon	27
Innlogging og registrering	29
Navigasjon	32
Hjem	33
Kontakter	34

Min Profil	37
GPS	38
Innstillinger	39
Avslutning av implementering	39
4. Systemdokumentasjon	40
4.1 Systemarkitektur	40
4.2 Database	42
5. Diskusjon	43
6. Konklusjon	44
7. Referanser	45
8. Vedlegg	48
8.1 Brukerveiledning	48
8.2 Figurliste	52
8.4 Gruppekontrakt	53
8.5 Bruksmønster	54
8.6 Møtereferat	57
8.7 Wireframes	60
8.8 Hi-fi prototype	63

1. Introduksjon

Link Us er en løsning på hvordan man enkelt kan dele visittkort i den nye og raskt voksende digitale verden. Applikasjonens ideen kommer fra Caroline Corneliussen og ble presentert til oss i henhold til skolens presentasjon om bachelor muligheter.

Caroline ønsker en applikasjon som skal fremheve og simplifisere bruk og deling av visittkort på en strukturert og innholdsrik måte. Samtidig vil hun også ha mulighet for deling av innlegg til dine kontakter, vise posisjon av brukere når de er på konferanser og mulighet for både privat og-bedrift brukere. Dette vil gjøre det enkelt å holde alle dine kontakter på et og samlet sted.

Denne applikasjonens mål ved ferdig utviklingsløp, vil være å enkelt kunne tilby kontakter til brukere som trenger forbindelsene til sine arbeids formål.

Hovedpoenget i dette prosjektet for oss er å lage en applikasjon for både android og iOS enheter. En versjon for en nettside er ikke et mål, men det er lagt opp til videreutvikling av dette.

Vi har gjennom semesteret tatt på oss utfordringen med å lære oss et nytt programmeringsspråk for å kunne levere en best mulig applikasjon. I denne rapporten tar vi for oss vårt utviklingsløp gjennom dette semesteret, alt fra planleggingsfasen til rapport fasen og forklaringer om de diverse systemene/programmene vi har brukt.

2. Bakgrunns litteratur

2.1 Utviklingsmetode

Utviklingsmetoden vi har benyttet oss av består av de 4 strukturerte fasene som er definert av Dennis, Wixom og Roth. (Dennis, Wixom, Roth. 2012).

Fasene i denne utviklingsmetoden består av Planlegging, Analyse, Design og Implementering.

Planlegging

I planleggingsfasen skal man ta for seg det meste av tilrettelegging og tanker som er rundt prosjektet, som f.eks. hva vi trenger, hvorfor vi trenger det og om det er mulig å gjøre det. Dette gjør at man kan ha en god og solid bakgrunn å bygge fra, før man begynner å skal gjøre analyser basert på nettopp disse valgene.

Analyse

I analysefasen skal man få en oversikt over målgruppen til systemet, hvilke krav systemet må ha og hvordan brukerne vil bruke systemet. Her gjøres analysen basert på hvilke valg man gjorde i planleggingsfasen for å se om planene vil fungere like bra som originalt tenkt, eller om det må gjøres endringer fra de originale planene.

Design

I designfasen skal man ta for seg hvordan systemet skal se ut, hvordan det skal fungere og alt som omfatter hva som skal være med i applikasjonen. Design av utseendet er en vesentlig del av denne fasen, men andre designvalg som struktur er også en viktig del her.

Implementering

Dette er fasen som skal kombinere og bruke det man har kommet fram til i de 3 første fasene. Dette er en kontinuerlig fase hvor implementeringen av designet kan endres fra dag til dag, og man kan ofte måtte gå tilbake til andre faser etterhvert som man finner ut nye ting. Denne fasen inneholder også støtte av programmet etter at prosjektet er ferdig utviklet.

Agile

Denne utviklingsmetoden er en iterativ arbeidsmåte for å fremstille et fullverdig prosjekt, innenfor agile har vi valgt å bruke Kanban som utviklingsmetode.

Kanban er en visuell presentasjon av et kanban board som gjør det mulig å vise fram ulike arbeidsoppgaver i diverse kategorier, gjerne basert på hvilke type oppgave eller basert på hvor viktig den er. Dette gjør det mulig å aktivt være frampå den iterative framgangs måten som den agile utviklingsmetoden krever.

Kanban har 3 viktige deler:

Visualisering av arbeidsflyt gjør det mulig for å vise hva som må gjøres, hva som gjøres og hva som er ferdig.

Funksjon under utvikling gjør det mulig for alle å hoppe inn for å se hva som gjøres akkurat nå, med mulighet for å komme med hjelp eller tips der det trengs.

Utvikleren velger seg sin arbeidsoppgave basert på hva arbeider selv er mest komfortabel eller mest kompetent med, dette gjør at arbeiderne lettere selv kan velge oppgaver som de selv mener er innenfor det de kan og dermed effektiviserer arbeidet.

2.2 Prinsipper og uttrykk

Universell Utforming

Den internasjonale definisjonen på universell utforming er *“Utforming av produkter, omgivelser, programmer og tjenester på en slik måte at de kan brukes av alle mennesker, i så stor utstrekning som mulig, uten behov for tilpassing og en spesiell utforming. ‘Universell utforming’ skal ikke utelukke hjelpemidler for bestemte grupper av mennesker med nedsatt funksjonsevne når det er behov for det.”* (Lid, 2020)

Frontend

Frontend er den delen av koden man skriver for å forme utseende på en applikasjonen og bestemmer hva som skjer når man interagerer med disse elementene. (Granevang, 2020)

Backend

Backend er den delen av koden man skriver som brukeren aldri ser, det er bakgrunnsprosesser, kalkuleringer og spørringer mot databaser. Man trenger fortsatt frontend for å aktivere backend-prosesser, som oftest må man trykke på en knapp for at noe skal skje i applikasjonen, som for eksempel registrering av en bruker i en database. (Granevang, 2020)

Use case

En use case, eller, bruksmønster er en rekke hendelser utført av en bruker for å oppnå et resultat i applikasjonen. Vi bruker bruksmønster for å få oversikt over alle hendelsene som skjer og som er tilgjengelige for brukeren når en annen hendelse er utført, slik at man enklere kan forstå og utvikle de nødvendige interaksjonene mellom brukeren og systemene. (Dennis, Wixom, & Roth, 2012 s. 149-150)

Navigasjon

Navigasjon i en programmerings-sammenheng handler om en brukers opplevelse når man går fra et sted i applikasjonen til en annen. God navigasjon og god brukeropplevelse går hånd i hånd, det må være tydelig for brukeren hva som kommer til å skje når de trykker på en knapp i applikasjonen. (Tubik, 2017)

API

API er en forkortelse for “Application Programming Interface” og er et hjelpeverktøy i programmering som gir funksjonalitet mellom programvarer uten at man trenger å sette seg inn i kildekoden til det originale programmet. (Rossen, 2020)

State Management

State management referer til oppførselen til en funksjon. Funksjonen endrer brukergrensesnittet basert på tilstanden til applikasjonen. Et eksempel på dette er hvis en bruker er logget inn og lukker applikasjonen, skal brukeren sendes direkte til hovedskjermen når brukeren åpner applikasjonen igjen, og ikke behøve å logge inn på nytt (Flutter, 2021).

2.3 Språk

Flutter

Flutter er et åpent kilde applikasjonsrammeverk laget av Google som bruker programmeringsspråket Dart til å lage applikasjoner for både IOS og Android med bare én kodebase. Flutter benytter seg av “widgets” som er ferdig definerte UI-elementer man kan personalisere. Dette gjør det raskt og effektivt å programmere med Flutter uten at man mister ytelse på sluttproduktet.

Dart

Dart er et objektorientert programmeringsspråk laget av Google med C-style syntax og et hovedfokus på front-end utvikling.

2.4 Database

En database er en organisert samling av data, gjerne i et datasystem.. (Oracle, 2021).

Vi har valgt å gå for en NoSQL database som heter firestore

Firestore er en applikasjonsutvikling platform laget av Google med et fokus på å bygge, forbedre og skalere en applikasjon. Firestore bruker en NoSQL database, som betyr at spørringene ikke skrives i SQL, men bruker egne metoder med egen syntax knyttet til Dart for å fungere. Med Firestore kutter man ut et mellomledd mellom applikasjonen og databasen, alt skjer direkte.

2.5 Rammeverk og verktøy

Discord er en gratis video, tekst og stemme applikasjon som brukes til å kommunisere mellom to eller flere personer. Ved bruk av Discord kunne vi lett ha samtaler/møter når vi trengte det eller hadde spørsmål. (Discord, 2021).

Også verdt å nevne for kommunikasjon er Facebook Messenger, dette er et gratis meldingsprogram fra Facebook som tillater enkle og kjappe samtaler i mellomgruppen uavhengig hvor de er.

GitHub er en applikasjon som gjør det mulig å samle alle sitt arbeid på et sted, samtidig som man lett kan ha god versjonskontroll, deling og administrering av kode. Man oppretter brancher (grener) som man laster opp kode til, og når man ønsker å dele sin branch med andre så merger (sammenslår) man sin branch inn i en annen. Hoved-branchen blir kalt “main” og kun ferdig testet brancher blir merget inn i main (Github, 2020).

Trello er et medium for digital bruk av Kanban med visuelle kort, her kan vi legge til diverse oppgaver som må gjøres, kategorisere dem og sette oss opp på de oppgavene vi holder på med. (Trello, 2021).

Android Studio er et IDE (Integrated Development Environment) for Android applikasjoner. Dette ble brukt til selve programmeringen av applikasjonen, og gjorde tillegginger av såkalte “packages” fra andre lett tilgjengelig. Innebygd støtte av Github gjorde også samarbeid lett. (Android Studio, 2021).

Adobe XD er et verktøy for å kunne visualisere eller modellere modeller av hvordan du vil at applikasjonen skal se ut. Dette er et stort steg opp fra gjerne enkle wireframes hvor man ofte bare har generelle trekk for applikasjonen, i mens Adobe XD gir oss muligheten til å få fram akkurat hvordan man vil ha utseendet. Det er også et stort “bibliotek” å hente genererte ikoner og lignende som vi kan bruke i fremvisningen. (Adobe XD, 2020).

Google Docs er en tekstprogram hvor flere brukere kan samarbeide i samme ark, dette er blitt brukt hovedsakelig til rapportskrivningen.

3. Prosjektgjennomførelse

I dette kapitlet vil vi gjennomgå hvordan vi har utført bacheloroppgaven og vår fremgangsmåte. Vi har fulgt de fire fasene av systemutviklingens livssyklus definert av Dennis, Wixom og Roth. Vi har sørget for jevnlig møter med oppdragsgiver Caroline og veileder Rania. Dette har sørget for at vi har kunnet utvikle en applikasjon som står til forventningene, og enkelt kunne gjøre endringer underveis.

Vi vil først ta for oss en oppsummering av kapitlet, deretter gå dypere i hvert punkt.

Vi startet med å opprette en gruppekontrakt for alle prosjektets medlemmer. Etter en presentasjon av applikasjonens idé inngikk vi en dialog med oppdragsgiver Caroline. Vi hadde et møte den 16. oktober der vi diskuterte applikasjonens krav og forventninger, og ble så enig om å inngå i et samarbeid.

Etter kontraktinngåelsen begynte vi med planleggingsfasen. Her bestemte gruppen seg for utviklingsmetode, programmeringsspråk, database og hvilke verktøy som skulle benyttes. Vi lagde også en tidsplan og arbeidsplan ved bruk av Trello.

Når disse punktene var i mål gikk vi videre til analysefasen hvor vi definerte applikasjonens målgruppe, krav til systemet og bruksmønster. Med jevnlig møter kom vi sammen med Caroline frem til hvilken brukergruppe applikasjonen skal nå frem til og alle kravene for applikasjonen.

Før vi kunne begynne på selve utviklingen av applikasjonen gikk vi gjennom designfasen. Her lagde vi wireframes og prototyper av applikasjonen. Vi brukte mye tid på hvordan vi ville at applikasjonen skulle se ut. I denne fasen valgte vi fargevalg for applikasjonen og bestemte at vi skulle følge Material Design sitt designspråk. Mye av grunnarbeidet ble lagt ned i designfasen som gjorde utviklingen av applikasjonen enklere.

Til slutt er implementeringsfasen, som er hvor selve utviklingen av applikasjonen skjer. Vi utviklet funksjoner basert på systemkravene og prioriteringslisten oppsatt i Trello, og vi har med et godt samarbeid utviklet en applikasjon med komplekse funksjoner som vi er stolte av.

3.1 Planlegging

Første fasen i systemutviklingens livssyklus er planlegging. Her skal man gjøre rede for struktur, utvikling og hvordan prosjektet skal bli gjennomført. Vi begynte planleggingsfasen etter kontraktsinngåelsen med Caroline.

Utviklingsmetode

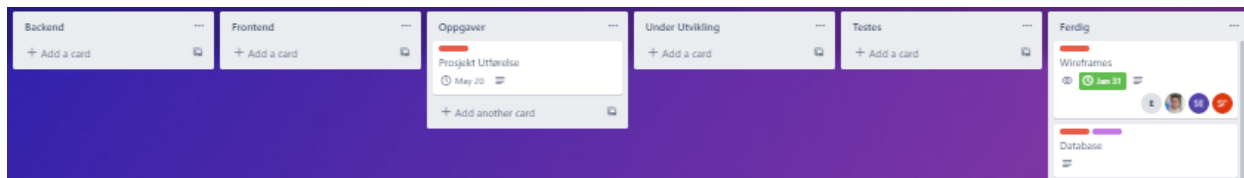
For at gjennomføringen av et prosjekt skal være så effektivt som mulig benytter man seg av utviklingsmetodikk. I systemutviklingens verden finnes det mange typer metodikker, og hvilken metodikk man velger avhenger av prosjektet og utviklingsgruppen. I vårt prosjekt var ikke alle funksjoner forhåndsbestemt og eventuelle endringer kunne forekomme. Agile metode legger til rette for dette, og bygger prosjektet trinnvis. Prosjektet starter da med de mest nødvendige funksjonene og bygger deretter videre med flere funksjoner til prosjektet er ferdig (Stephens, 2015, s. 284).

Kanban er en type av Agile metodikk, og baserer seg på Agile sine retningslinjer. Kanban gir god oversikt over arbeidsoppgavene, hvilken prioritet de har, hva som er ferdig, under utvikling og hva som må gjøres. Vi fant Kanban som et veldig bra valg av utviklingsmetode grunnet prosjektets størrelse og muligheten til å gjøre endringer underveis. Dette i tillegg til at gruppen har tidligere erfaring med Kanban gjorde at valget for utviklingsmetode ble Kanban.

Arbeidsplan

Med Kanban som utviklingsmetode bruker man Kanban Board som gir en oversikt over arbeidsoppgavene i prosjektet. Kanban Board deler utviklingsprosessen inn i kolonner med forskjellig status. I følge Stephens er det vanlig for større utviklingsprosjekter å dele inn i: “Backlog, ready, coding, testing, approval and done” (Stephens, 2015, s. 354). Han nevner videre at Kanban gir frihet til å endre på oppsettet, og anbefaler å gjøre justeringer basert på prosjektet og utviklerne.

Et Kanban Board kan være en tradisjonell tavle, men vi valgte derimot å bruke Trello som er en elektronisk tavle. Vi delte vår Kanban Board inn i: backend, frontend, oppgaver, under utvikling, testes og ferdig. I tillegg hadde vi en generell info kolonne for enkel tilgang til diverse linker.



Arbeidsoppgavene ble tildelt merkelapper for å se viktighetsgrad og om oppgaven tilhørte frontend eller backend. Oppgavene ble sortert etter viktighetsgrad og de med høyest viktighetsgrad ble utført først.

I en Kanban utviklingsmetode har hvert gruppelem selv ansvar for å vise lederskap og velge ut en arbeidsoppgave som må utføres. Her finnes ingen leder som deler ut arbeidsoppgaver, og enhver har selv ansvar for å starte på en ny oppgave når man er ferdig (Stephens, 2015 s. 352).

Med oppgavene sortert etter viktighetsgrad valgte vi fritt om vi ville jobbe på en frontend eller backend arbeidsoppgave. Når man valgte seg en oppgave flyttet vi det notatet over til “Under Utvikling” og man valgte seg selv som ansvarlig for oppgaven. Ettersom oppgaven ble utviklet ble den flyttet til “Testes” og tilslutt “Ferdig”. Tilleggsinformasjon ble lagt til notatet i de tilfellene det var nødvendig.

Tidsplan

Estimering av tidsbruk kan gjøres basert på erfaring, kapasitet og ved hjelp av verktøy. I følge Roth, Dennis og Wixom er det industri standard å dele de fire livssyklusene til systemutvikling inn i prosenter: Planlegging 15%, Analyse 20%, Design 35%, Implementasjon 30%. (Dennis, Wixom, Roth, 2012, s. 62).

Vår tidsplan er basert på disse industri standardene, samtidig som vi har gjort endringer tilpasset vårt prosjekt. Med Kanban som utviklingsmetode har vi friheten til å utvikle funksjoner uten å måtte være ferdig med design fasen for alle funksjonene. For eksempel var ikke detaljene og kravene for “nyhetsveggen” ferdigstilt innen 30. April, men vi kunne fortsatt starte med programmering av funksjonene innlogging og registrering. Dette gjør at de to fasene design og implementering overlapper hverandre.

Våre arbeidstimer er godt sammenlignbart med industri standard, bortsett fra vårt avvik med mere timer på implementasjon enn design. Dette kommer av prosjektets kompleksitet og utfordringene med å programmere de ulike funksjonene. Gruppen trengte derfor mere tid for å få utført implementeringsfasen.

Hendelse	Startdato	Estimert sluttdato	Varighet i uker	Antall arbeidstimer
Signert kontrakter	17.November 2020	1. Desember 2020	2	8

Planlegging	1.Desember 2020	29.Januar 2021	8	45
Analyse	1.Februar 2021	28.Februar 2021	4	70
Design	1.Mars 2021	30. April 2021	8	90
Implementering	22. Mars 2021	30. April 2021	5	100
Ferdigstilling av rapport	1. Mai 2021	20. Mai 2021	2	60
Overlevering	20. Mai 2021	15. Juni 2021	3	5
Innlevering av rapport	20. mai			

Figur 1, Tidsplan

(Antall arbeidstimer er estimert for hvert individuelt gruppe medlem.)

Våre individuelle arbeidstimer har en samlet sum på 378 timer, som møter forventninger fra USN på ca. 375 timer for dette prosjektet.

Risikoanalyse

En risikoanalyse er en proaktiv plan hvor man identifiserer potensielle risikoer, hvordan de kan påvirke et prosjekt og tiltak før det skjer. For hvert steg skal vi kunne avgjøre sannsynligheten for at noe skjer, hvor alvorlig det er, konsekvensene og tiltak.(Stephens, 2015, s. 48)

Alvorlighetsgrad:

1. (Lav) Ikke alvorlig og lett å håndtere.
2. (Moderat) Litt alvorlig men kan håndteres uten at prosjektet påvirkes.
3. (Høy) Alvorlig. Vanskelig å håndtere og vil sette prosjektet i fare.

Sannsynlighetsgrad:

1. (Lav) Liten sannsynlighet for at noe vil skjje
2. (Moderat) Moderat sannsynlighet for at noe vil skjje

3. (Høy) Stor sannsynlighet for at noe vil skje

Problem	Sannsynlighetsgrad	Alvorlighetsgrad	Konsekvens	Tiltak
Sykdom	Lav	Lav	Senket effektivitet	Tiltak for å minske smitte av sykdommer som Covid-19. Vi holder hverandre oppdatert på helsestatus og jobber hjemmefra hvis helsen tillater det.
Nettverksbrudd	Lav	Moderat	Ett nettverksbrudd eller ustabil netting kan føre til at en ikke kan sende eller motta endringer i koden fra Github.	Tiltak som å alltid bruke kablet netting hvis mulig. Hvis kablet/trådløst netting ikke fungerer så kan mobilnetting være et alternativ.
Sitter fast / får ikke til	Moderat	Moderat	Senket effektivitet. Kan føre til brudd i kontrakten om en ikke kan levere ønsket funksjonalitet som lovet i kontrakten.	Si ifra om en sitter fast med en kritisk oppgave så de kan få bistand.
Intern uenighet / konflikter	Lav	Moderat	Interne konflikter kan føre til senket effektivitet og et dårlig arbeidsmiljø.	God kommunikasjon med hverandre for å unngå at konflikter/misforståelser kan oppstå.
Tap av kode	Lav	Høy	Taper vi kode som ikke er lastet opp	Vi bruker Github som versjon

			til Github eller sikkerhetskopiert så må vi starte på nytt. Senket effektivitet.	kontrollsystem så all kode og tidligere versjoner kan hentes fra nettet. Regelmessige sikkerhetskopier blir også tatt.
Oppdragsgiver liker ikke produktet	Lav	Høy	Brudd på kontrakt. Må starte på nytt.	Regelmessige møter med oppdragsgiver for å få feedback og tanker om produktet fra start til slutt.

Utviklingspråk og database

For denne applikasjonen har vi satt oss inn i veldig mye nytt og ukjent innen systemutviklingen for å levere et optimalt produkt til oppdragsgiver. Her går vi i dybden på utviklingspråkene og platformene vi valgte å bruke, hvorfor vi valgte dem og hvilke problemer som har oppstått.

Vi valgte å utvikle denne applikasjonen i Flutter, flutter er et rammeverk innenfor det objektorienterte programmeringsspråket, Dart. Dette var på grunn av at oppdragsgiver ønsket at applikasjonen skulle fungere på både IOS og Android systemer. Flutter har innebygde verktøy for å kompilere koden til både APK-filer for Android-enheter og IPA-filer for IOS-enheter som gjorde det veldig appellerende for oss å velge Flutter over andre rammeverk for andre programmeringsspråk. I rammeverket finner man mange “Widgets” som er ferdig definerte UI-elementer man kan personalisere etter behov, dette var viktig for oss fordi det gjorde det mulig å presentere funksjonelle forslag til applikasjonen for oppdragsgiver både utseendemessig og funksjonsmessig ganske tidlig.

For databasen valgte vi å bruke Firebase. Firebase er utviklet av Google, dette fører til god synergi med Flutter som også er utviklet av Google. Dette førte til at vi kunne spare tid på back-enden av applikasjonen og heller fokusere på brukeropplevelse. Med Firebase trenger vi ikke å programmere

egen autentisering, statistikk og database for applikasjonen, i tillegg blir det en enklere opplevelse for kunden å holde kontroll på data når vi overleverer oppgaven. Firebase er også skalerbart, som betyr at kunden selv kan se på statistikken og velge hvilken pakkeløsning som ønskes ut i fra den. Vi mener derfor at Firebase har vært optimalt for både oss under produksjon og for kunden når applikasjonen overleveres.

Det har vært mye nytt for oss å lære under dette prosjektet. Ingen i gruppen har hatt tidligere erfaring med Dart eller Flutter, men alle har hatt litt erfaring med Firebase fra et tidligere prosjekt som ble tatt med i betraktningen for valget av Googles programmeringsmidler. Dette var bare utfordrende i begynnelsen, alle i gruppen har jobbet hardt og iherdig for å bli godt kjent med både Dart og Flutter slik at vi har kunnet levere et produkt til kunden vi er stolte over selv.

Verktøy

I ethvert prosjekt er det viktig med god planlegging, dette kan gjøres uten hjelp av verktøy, men gjøres mye bedre og effektivt med de rette planleggingsverktøyene. Vi valgte å bruke følgende verktøy: GitHub, Trello, Discord, Google Docs og Adobe XD. Nedenfor forklarer vi hvordan vi har brukt disse verktøyene.

Som forklart i bakgrunns litteratur er ikke Github egentlig et verktøy for planlegging, men vi brukte Github tidlig i prosjektfasen for å teste oss effektivt fram til “ting” som fungerte. Dette kunne vi gjøre effektivt med Github takket deres versjonskontroll og “branches”.

Trello er et verktøy for Kanban vi brukte fra nesten første fase av utviklingen. Vi førte opp felles med oppgaver som måtte gjøres før vi satte oss i arbeidet og satte oss opp på de relevante oppgavene, basert på nødvendighet og kunnskaper. Trello ble brukt gjennom hele prosjektet og oppdatert med nye oppgaver etterhvert som de ble gitt.

For bruk av kommunikasjon når vi ikke kunne være fysisk til stede under arbeid, ble discord brukt i interne gruppemøter. Dette var en essensiell del av vår planlegging grunnet lite

muligheter til å møtes fysisk grunnet korona. Discord ble brukt mye ved planlegging ofte sammen med andre verktøy.

Google Docs var vårt viktigste verktøy for å holde kontroll på møtereferater og ideer/ spørsmål vi ville ta opp til neste møte. Dette ble hyppig brukt under de første møtene vi hadde mellom klient og oss, men ble også brukt under hele prosjektfasen.

Et av våre mest viktigste verktøy for planlegging i var Adobe XD, uten denne hadde det vært vanskelig å få fremvist vår visjon av applikasjonen på en så god måte vi gjorde. Adobe XD gjorde at vi enkelt kunne få tilbakemelding på del applikasjonen og like enkelt gjøre endringer, dette gjorde at når arbeide skulle starte, så hadde vi en solid grunnmur å bygge på.

3.2 Analyse

Analysefasen skal gi en oversikt over målgruppen til systemet, kravene til systemet og bruksmønster. Disse stegene ble utført i samarbeid med oppdragsgiver og veileder gjennom flere møter. Vi vil nå se nærmere på hvert punkt.

Målgruppe

En målgruppe er den folkemengden applikasjonen skal appellere til. Sammen med vår oppdragsgiver Caroline kom vi frem til at vi har en bred målgruppe. Vi har sammen med Caroline definert målgruppen på best mulig måte for at applikasjonen skal ha en god løsning som tilrettelegger alle brukerne.

Linkus har som mål å ta over for tradisjonelle (fysiske) visittkort, og ønsker alle som har visittkort til å erstatte disse og gå over til applikasjonens digitale visittkort. I Norge var det i 2019 i overkant av 2,9 millioner jobber (SSB, 2020). Dette tilsier et potensielt brukerantall på 2,9 millioner for Linkus, basert på at alle lønnstakere vil skaffe seg visittkort. En annen type brukere som også er en del av målgruppen er privatpersoner som ønsker seg et personlig visittkort, f.eks. studenter eller jobbsøkere.

Applikasjonen skal ta hensyn til målgruppen, og her finner vi både ansatte i statlig og private firma og privatpersoner. Vi måtte derfor utvikle applikasjonen med dette i tankene, og at informasjon brukerne ønsker å oppgi vil variere. En student ønsker muligens å oppgi sitt studiested og mobilnummer, men en statlig/kommunalt ansatt må være mer anonym. Vi ble derfor enig om at det skal være mulig å opprette to typer brukere. En for privat sammenheng og en for jobbsammenheng. Hvordan de to bruker typene fungerer kommer vi tilbake til senere i rapporten i punktet implementasjon.

Systemkrav

I kontrakten som vi inngikk med Caroline definerte vi en overordnet liste med krav til applikasjonen. Vi delte disse inn i tre kategorier; må krav, kan inneholde og skal legge til rette for. Figuren under viser oversikten over kravene.

Krav til applikasjonen	
1	Fungerende applikasjon for Android operativsystem
2	System for innlogging og registrering.
3	Mulighet for å opprette nytt visittkort og redigere via applikasjonen
4	Visittkort skal deles ved bruk av NFC og QR-kode
5	Dele visittkort via ønsket sosiale medier og via telefonsamtale/meldinger
6	Mulighet for oppkobling til brukerens egne sosiale medier. Fritt valg for deling av lenker til ønsket nettsted.
7	Gjøre anrop og sende mail direkte via visittkortet.
8	Universell utforming av applikasjonen
9	Brukerveiledning/informasjon om bruk av applikasjonen.
10	Søkefunksjon for søking av andre brukere (firmanavn, tittel, navn osv.)
Applikasjonen kan inneholde (Hvis vi har tilstrekkelig tid)	
11	Utvikle for iOS operativsystem
12	System for innlogging via LinkedIn (Eventuelt andre sosiale medier).

13	Opprette kataloger med dine kontakter.
14	Oppkobling til Telenorkatalogen.
Applikasjonen skal legge til rette for videre utvikling av	
15	Skanne inn ordinære visittkort for lagring
16	Chattefunksjon
17	Nettside av applikasjonen
18	Mulighet til å endre informasjon på PC
19	Integrasjon av Google Maps, deling av posisjon til andre brukere

Figur 2, Systemkrav

Bruksmønster

På bakgrunn av systemkravene laget vi bruksmønstrene. Vi lagde først en mal bruksmønstrene skulle følge, og bestemte i fellesskap hvilke bruksmønstre som var nødvendige å opprette. Vi følte det ikke var nødvendig å ha med alle bruksmønstrene i rapporten, og viser derfor til figur 3 nedenfor. Figuren viser stegene og interaksjonen brukeren tar for å legge til en kontakt i applikasjonen. Oversikt over alle bruksmønstrene finnes under vedlegg (punkt 8.4 Bruksmønster).

Navn	Legge til en kontakt
Prioritet	Høy
Forutsetning	1. Brukeren er innlogget i applikasjonen
Hendelse	<ol style="list-style-type: none"> 1. Brukeren navigerer til siden “Kontakter”. 2. Brukeren trykker på legg til kontakt ikonet, som åpner siden for å legge til en kontakt 3. Brukeren taster inn navnet på kontakten som ønskes, og legger den til. 4. Kontaktlisten til brukeren oppdateres i databasen og brukeren har nå tilgang til kontaktens informasjon
Alternativ hendelse	<ol style="list-style-type: none"> 1. Legge til kontakt ved bruk av QR-kode eller NFC <ol style="list-style-type: none"> a. Brukeren er i nærheten av kontakten som ønskes å legges til b. Brukeren navigerer til “Legg til en kontakt” siden. c. Her velges enten “Legg til med QR-kode” eller “Legg til med NFC”. d. QR-koden skannes evt. NFC mellom brukerne. e. Kontaktlisten til brukeren oppdateres i databasen, og

	brukeren har nå tilgang til kontaktens informasjon
Unntak	<ol style="list-style-type: none"> 1. Skrivefeil som gjør at kontakten ikke dukker opp. 2. Brukeren mister tilgang til internett

Figur 3, Bruksmønster for å legge til kontakt

3.3 Design

Wireframes

Wireframes er en teknisk skisseringsmetode som gir en grov oversikt over hvordan en side vil se ut og fokuserer på hvordan plass blir brukt, hva slags innhold som vil bli vist, funksjoner som vil være tilgjengelig og hvordan man navigerer seg gjennom sidene. (Lemay, L. Colburn, R. Kyrin, J. 2016. s.34) For å holde wireframes enkle å forstå mellom utvikler, designer og klient så styles elementene lite for å minimere sjansene for at misforståelser kan oppstå.

Wireframes kommer typisk 3 varianter:

1. “Low-fidelity”
Lavkvalitets skisser lagd for hånd på et stykke ark eller post-it lapp. Praktisk når en først får en ide.
2. “Mid-fidelity”
Digitale skisser som er mer detaljert med dummytekst, flere elementer og litt farger. Standard nivå å starte på.
3. “High-fidelity”
Høykvalitets digitale skisser med flere detaljer, farger, bilder, elementer og relevant tekst. (Hannah, 2021)

Vi har gått for en “Mid-fidelity” wireframe som du kan se i Figur



4.

Figur 4, wireframe

Flere eksempler av wireframes fra prosjektet finner du i vedlegg (punkt 8.7, Wireframes)

High-fidelity prototype

En prototype kan beskrives som en simulasjon av det ferdige produktet for å teste hvordan en potensiell sluttbruker vil samhandle med grensesnittet.

Prototyper kommer som regel i 2 varianter:

1. “Low-fidelity” prototype. Det er en grunnleggende representasjon av et konsept laget av materiale som papir eller papp.

2. “High-fidelity” prototype. Det er en høykvalitets digital prototype med tilnærmet lik design, farger, animasjoner og funksjoner som sluttproduktet. (Babich, 2017)

Vi valgte å gå for en “High-fidelity” prototype ettersom vi hadde tidligere erfaringer med verktøyet AdobeXD for å lage wireframes og prototyper. Med AdobeXD kan vi jobbe sammen og lett dele prototypen med oppdragsgiver, få tilbakemelding og gjøre endringer umiddelbart.



Figur 5
Hi-Fi prototype

Flere eksempler av High Fidelity-prototype finner du i vedlegg (punkt 8.8, Hi-Fi prototype)

Material Design & Universell utforming

For applikasjonens komponenter og designprinsipper så følger vi Material Design sine retningslinjer. Material Design er et design system utviklet av Google for å skape vakre grensesnitt som også følger krav for god universell utforming. (Material Design, 2021).

Fargepaletten utformet vi i samarbeid med oppdragsgiver gjennom flere møter og ble enige om fargene i figur 6 som har blitt brukt på forskjellige sider i applikasjonen. Men vi har også fått en del kreativ frihet til å bruke andre farger der vi mener det passer for å opprettholde lesbarhet og synlighet i applikasjonen.

I tillegg til å følge Material Design sitt designspråk så må applikasjonen også må den også følge lovene for universell utforming.

Universell utforming handler om å bygge tjenester som kan brukes av hvem som helst uavhengig av alder, funksjonsevne og utdanningsnivå. (Uutilsynet, 2021)

Når vi utformet applikasjonen sitt grensesnitt så har vi alltid hatt universell utforming i baktankene våres.

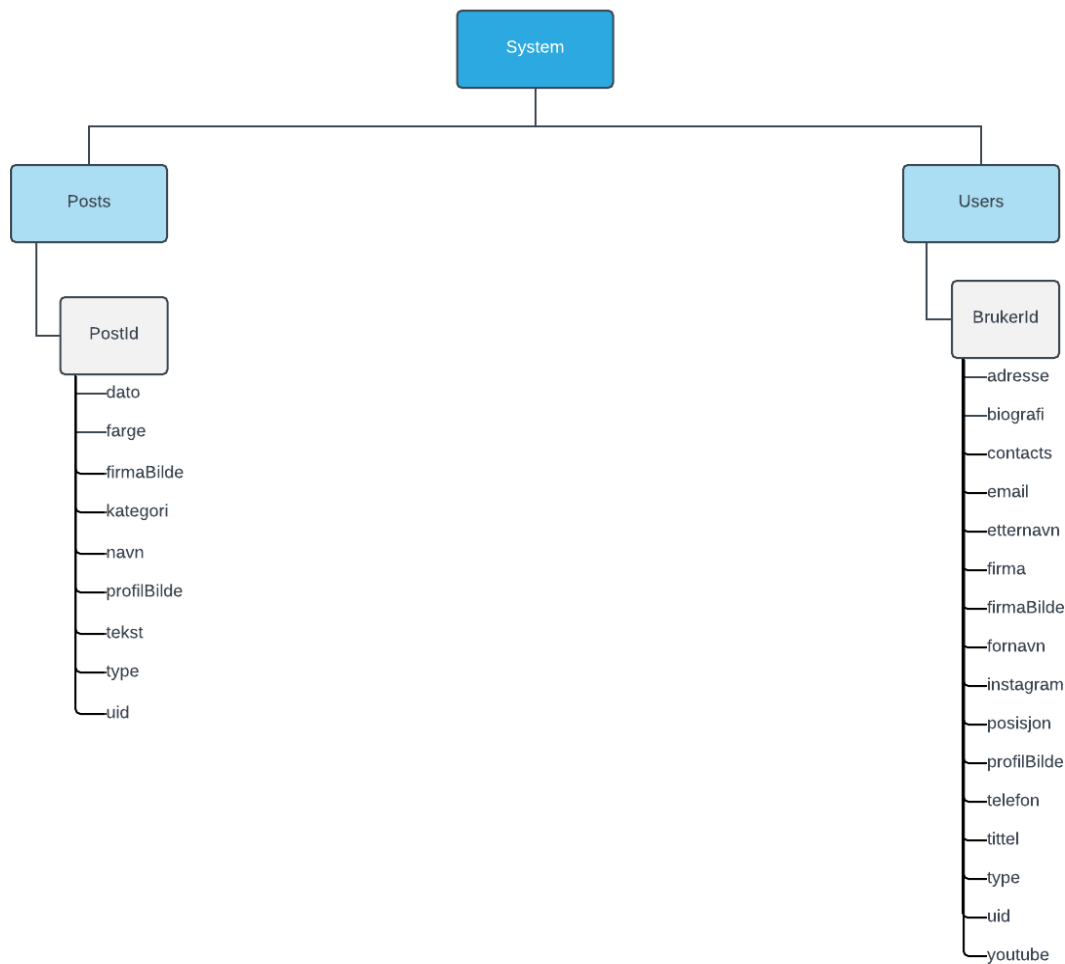
Vi har gått gjennom mange fargekombinasjoner, tilpasset layouts og input felt for at teksten skal være lesbar og synlig, synlig fokus på elementer som er valgt, god og simpel navigasjonsmetode og animasjoner som gir brukeren en indikasjon på noe skjer eller at de blir sendt til en annen side.



Figur 6, Fargepalett

Databasemodell

En databasemodell er en visuell representasjon av hvordan en database er bygd opp, hvor dataen flyter og dataens relasjon til hverandre. Databasemodellering blir brukt i de tidlige fasene når man skal utvikle en database eller en applikasjon som benytter seg av en database. I ikke-relasjonsdatabaser som Firebase, så lagres dataen i en semi-strukturert dokumentstruktur med dokumenter som kan ha forskjellige formater som JSON og XML. (Kristoffersen, 2016, s.433) Dette gjør at vi har en simpel og oversiktlig database med mapper som har dokumenter med hver sin unike id. Bilde av våres databasemodell kan du se i figur 7.



Figur 7, NoSQL database

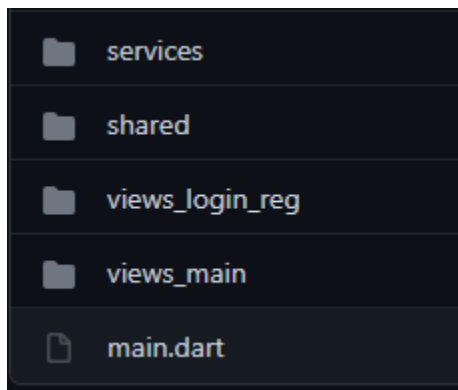
3.4 Implementasjon

Implementasjon er siste fasen av systemutviklingens livssyklus. Det er i denne fasen at systemet blir utviklet (Wixom, Dennis og Roth, 2012, s. 15).

Utviklingen av applikasjonen er gjennomført i henhold til Kanban og prioriteringslisten oppsatt i Trello. De funksjonene som var av høyest prioritet ble utviklet, testet og ferdigstilt først. I dette punktet vil vi først gjennomgå hvordan vi startet med implementeringen og deretter se nærmere på hver funksjon.

Vi begynte implementeringsfasen med å opprette et Github prosjekt og sørget for at gruppe medlemmene hadde tilgang til å hente og skrive til prosjektet. Vi delte prosjektet inn i følgende brancher; master, test-master, erlend, sigve, stian og sanan. Vi jobbet individuelt på hver vår branch, og når en funksjon var ferdig utviklet og testet ble den så merget inn i test-master. Ved hver merge ble test-master gjennomgått og testet for eventuelle feil før den så ble merget inn i main. På denne måten sørget vi for at main branch var jevnlig oppdatert med de nyeste funksjonene, samtidig som vi utelukket problemer og feil som kunne oppstå.

En viktig del av alle systemutvikling prosjekter er mappestruktur. Når man oppretter et Flutter prosjekt er mange av filene allerede inndelt i en mappestruktur, filer som er nødvendige for at prosjektet skal kunne kjøres på Android og iOS. I mappen “lib” (library) oppretter man de mappene og filene som prosjektet skal kjøre, altså de vi skal utvikle. Før vi begynte med arbeidsoppgavene opprettet vi derfor mapper i lib.



Dette bildet illustrerer vår mappestruktur.

I services ligger filer som applikasjonen betjener seg av, dette er hovedsakelig backend tjenester.

I mappen shared ligger filer med kode som benyttes flere steder i applikasjonen.

Views_login_reg inneholder filer for innlogging og registrering.

Views_main inneholder alle filene som har et utseende, altså noe som blir presentert til brukeren.

Vi inndelte vår mappestruktur i gruppefelleskap. Vår struktur skiller frontend og backend og vi mener dette gir en god oversikt for utviklerne å finne riktige filer. Dette sørger også for en god struktur når det gjelder videre utvikling av applikasjonen hvis det skal endres eller legges til filer.

Flutter er et deklarativt rammeverk som gir oss muligheten til å bruke state management. Den første funksjonen vi implementerte var nettopp dette. Når applikasjonen starter kjøres main.dart. Her sjekkes det om brukeren er logget inn eller ikke. Er brukeren logget inn kjøres “navigation_view.dart” og brukeren har full tilgang til applikasjonen. Hvis brukeren ikke er logget

inn kjøres “login_reg_menu.dart” og brukeren må logge inn/registrere seg for å få tilgang til applikasjonen.

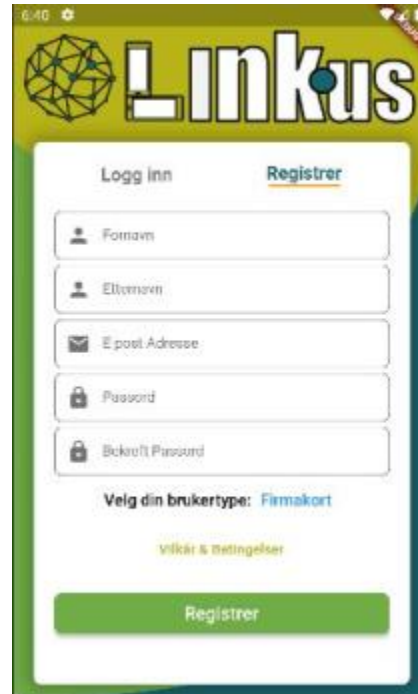
Med alt dette på plass kunne vi individuelt begynne på hver vår arbeidsoppgave. Vi vil videre i dette punktet gjennomgå funksjonene vi har implementert og forklare hvordan de fungerer.

Innlogging og registrering

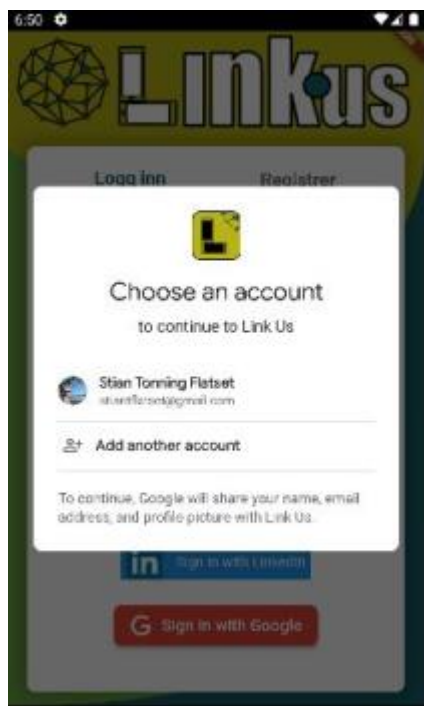
Et av systemkravene til applikasjonen var et fungerende registrering og innlogging system. Gjennom Firebase sin autorisering lagde vi innlogging/registrering via email og Google. Når en bruker skal registrere seg sjekkes email-adressen og passordet. Kravet for at en bruker skal bli registrert er (Firebase, 2021):

- Email-adressen kan ikke allerede være i bruk.
- Email-adressen må være en gyldig adresse
- Passordet må inneholde minimum seks karakterer

Vi har også implementert innlogging via Google, som gjør at man logger seg på med sin google bruker og må tillate Linkus til å få tilgang til brukerdata.

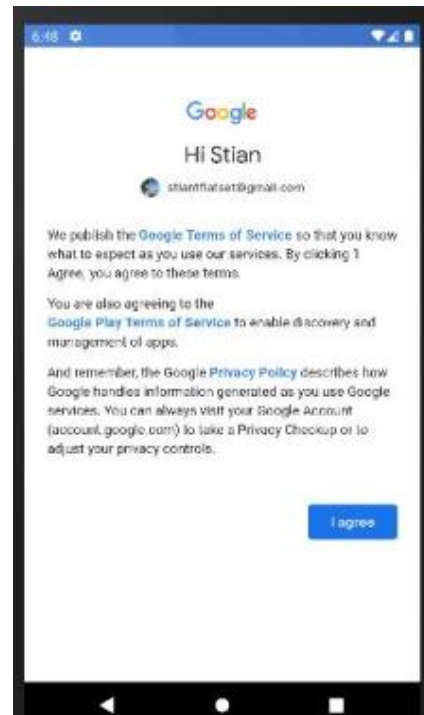
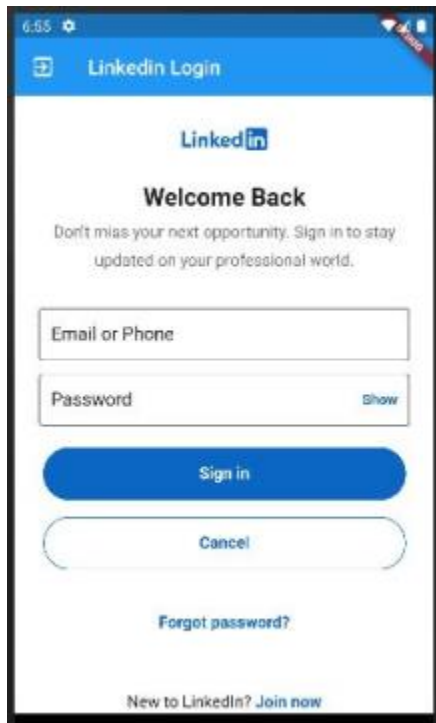


Vi mener vår implementering av innlogging og registrering gir en god brukeropplevelse samtidig som det er moderne. Man kan enkelt sveipe mellom de to sidene, eller trykke på knappene på



toppen. Når man er på en side vil knappen få farge og en understrek, dette for å vise tydelig til brukeren hvilken side man er på.

Hvis man registrerer gjennom en Google bruker vil Google ta brukeren gjennom noen få steg for at Linkus skal få tilgang til brukerens informasjon.



Vi har også begynt implementasjonen av innlogging gjennom en LinkedIn bruker. LinkedIn bruker sin egen autoriserings tjeneste som gjorde det mer krevende å implementere.

Vi har fått opprettet tilkobling til LinkedIn sin API som åpner et vindu der man kan logge seg på sin LinkedIn bruker. Autoriseringen fungerer med APIen, men vi har ikke fått den til å fungere sammen med Firebase.

Siden vi allerede har vårt eget innloggingsystem og Google sin, ble vi enig med Caroline å sette LinkedIn på vent. Funksjonen er nært ferdig, men ble lagt på vent på grunn av tidsbruk.

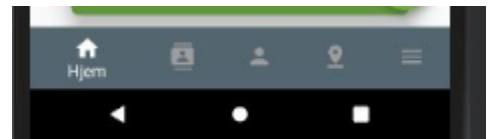


Som tidligere nevnt har vi implementert to typer brukere. Hvis man ønsker å bruke Linkus i jobbsammenheng velger man typen firmakort, hvis man ønsker å bruke Linkus i privatsammenheng velger man personkort. Ved bruk av firmakort kan man oppgi sin stilling og firma, og ved bruk av personkort kan man oppgi en ønsket tekst vi kaller biografi.

Typen brukeren velger blir lagret i databasen. Applikasjonen gjør endringer basert på hvilken brukertype som er lagret. Dette illustreres i punktene 1.4.4.Kontakter og 1.4.5.Min profil.

Navigasjon

Vi har implementert en nedre navigasjonsbar som sørger for enkel navigasjon i applikasjonen. For å komme seg til de ulike sidene kan man trykke på navigasjonsbaren eller sveipe til venstre/høyre på skjermen.



Hjem



Vi har laget en oversikt øverst på hjem og kontakter siden som viser innlogget bruker. Her er også en knapp for enkelt deling av kontakt. Dette forklarer vi dypere i neste punkt.

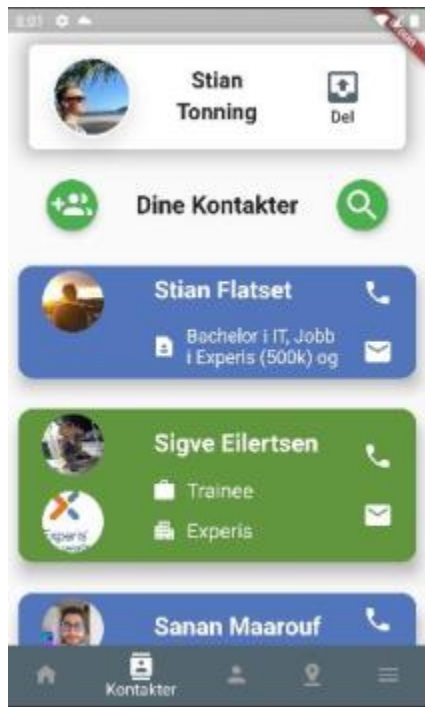
Hjem-siden er en nyhetsvegg som viser innlegg du og dine kontakter har publisert. Innleggene hentes fra Firestore og oppdateres i sanntid. Hvilken farge innlegget får er basert på hvilken kategori som publiseres. I samarbeid med Caroline opprettet vi muligheten for å publisere som; privat, firma, arrangement og reklame. Vi utviklet også en side som viser alle dine publiseringer, med muligheten til å fjerne innlegget.

Kontakter



Et ønske fra oppdragsgiver var at brukerne skulle enkelt kunne legge til og dele sin kontaktinformasjon. Når man trykker på “del knappen” så dukker det opp en QR-kode som er brukerspesifikk og muligheten til å skanne en QR-kode.

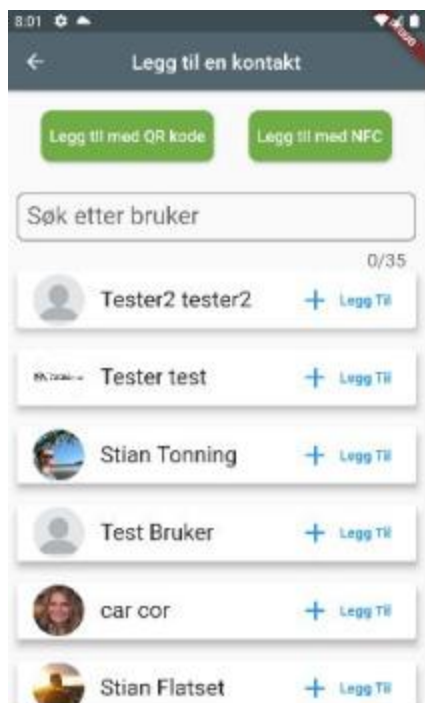
Når man skanner en QR-kode blir personens UID (brukerens egen identifikasjon) lagt til i brukerens kontaktliste, og man får tilgang til kontaktens informasjon.



På siden kontakter hentes kontaktene brukeren har lagret i databasen, og presenterer de i en liste.

I tillegg til navn og bilde vises også informasjon brukeren har lagret. For personkort vises en biografi, for firmakort vises arbeidssted og stilling.

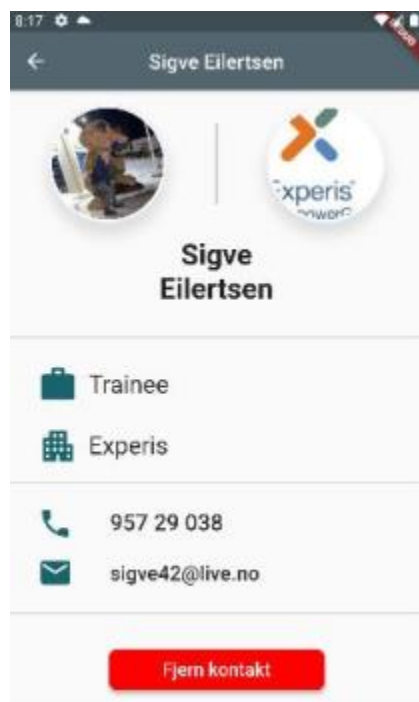
Systemkrav 7 spesifiserer at man skal kunne ringe og sende mail direkte via visittkortet og vi har derfor implementert dette inn i kontaktlisten.



Vi utviklet en egen side for å legge til en ny kontakt. Her kan man legge til med QR-kode, NFC eller søke manuelt. Listen viser en oversikt over alle brukere, og sorteres basert på inntastingsfeltet. Denne funksjonen har en sanntidsoppdatering med databasen og henter kun de brukerne som passer opp mot det som står i inntastingsfeltet.

Vi opplevde problemer med implementering av NFC og har ikke fått denne til å fungere skikkelig. Vi har opprettet en egen NFC fil, men fikk kun feilmeldinger. NFC er et avansert system, men vi mener med mer tid kunne vi fått det til å fungere. Vi vet teoretisk hvordan det skal implementeres, men

på grunn av tid fikk vi ikke NFC til å fungere som ønsket.

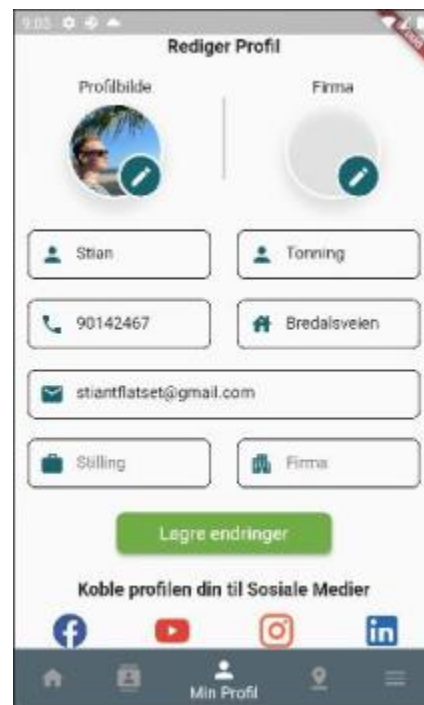
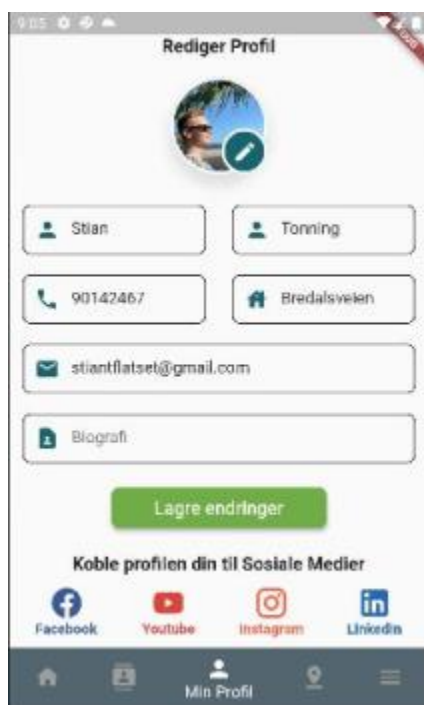


Hvis man trykker på en av sine kontakter vil en side med kontaktens informasjon dukke opp. På bildene over ser man Sanan som har registrert seg som personkort og Sigve som har registrert seg som firmakort. Her brukes state management igjen og hvordan siden bygges opp på er basert på brukertypen brukeren er registrert som i databasen.

Hvis en bruker har linket til sin Instagram eller Youtube profil vil man ved å trykke på kontakten kunne se dette. I bildet over blir man videresendt til Sanan sin Instagram-profil. Hvis Instagram er installert på enheten åpnet applikasjonen, hvis ikke åpnet nettleseren. Det samme gjelder Youtube.

Min Profil

Siden “Min profil” gir brukeren muligheten til å endre informasjon om seg selv og koble seg til sosiale medier. Siden endrer seg basert på om brukeren er registrert som firmakort eller personkort, som vist på bildene under.

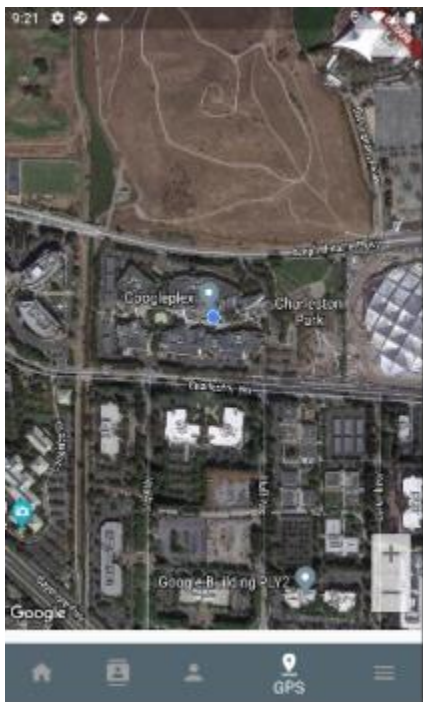




Oppkobling mot Youtube og Instagram fungerer ved at brukeren taster inn sitt profilnavn. Hvis navnet står tomt er verdifeltet i databasen tom og andre brukere vil da ikke se Youtube/Instagram knapp i kontaktinformasjon.

Facebook og LinkedIn har en noe annerledes måte å linke til profilen på og på grunn av tidsforbruk ble ikke disse ferdig implementert. Funksjonaliteten for viderekobling er allerede oppsatt i databasen og i prosjektet, det kreves derfor bare litt mer tid for å få disse ferdig implementert.

GPS



I systemkravet punkt 19 er det beskrevet at vi skal legge til rette for videre utvikling av integrasjon av Google Maps. Måten vi har utført dette på er å opprette en egen side for GPS, samtidig som vi har implementert selve kartet. I databasen blir det lagret geolokasjon til brukeren. Når man åpner GPS-siden finner den frem til brukerens lokasjon.

Dette sørger for god tilrettelegging for videreutvikling av funksjonen, da mye av funksjonaliteten allerede er implementert.

Brukeren må tillate deling av posisjon i applikasjonen for at funksjonen skal fungere.

Innstillinger



I enden av navigasjonsbaren finner man Innstillinger-siden. Hvilke funksjoner denne siden skulle inneholde var lenge usikkert, men vi avtalte sammen med Caroline å sørge for god videreutvikling av denne siden for fremtiden. Vi har kommet med noen forslag for funksjoner som kan implementeres på siden, og lagt til rette for dette. Funksjonene vi har implementert er; bytting av passord, endring av brukertype og lenke til personvern.

Avslutning av implementering

Målet vårt basert på tidsplanen var å være ferdig med implementasjonsfasen 30.April, som vi hovedsakelig klarte. Etter møte med Caroline 14.April gjorde vi noen finpuss og små endringer som var ferdigstilt 14.Mai. Vi hadde da fullført de aller fleste av systemkravene, samtidig som vi hadde implementert funksjoner som går under: “kan” og “legge til rette for”. Vi er svært fornøyde med resultatet og funksjonene vi har implementert. Vi har jobbet effektivt sammen som gruppe, og Kanban som utviklingsmetode har fungert utmerket for oss.

Vi har gjennom implementeringsfasen vist frem funksjonene til Caroline og Rania, og testet de grundig på ulike emulatorer og egne enheter. Ved slutten av implementeringsfasen opprettet vi APK-filer sånn at de selv fikk teste applikasjonen. Vi har anbefalt Caroline å ha en egnet testperiode der brukere i ulike aldre med ulike enheter tester ut Linkus i en bestemt tidsperiode.

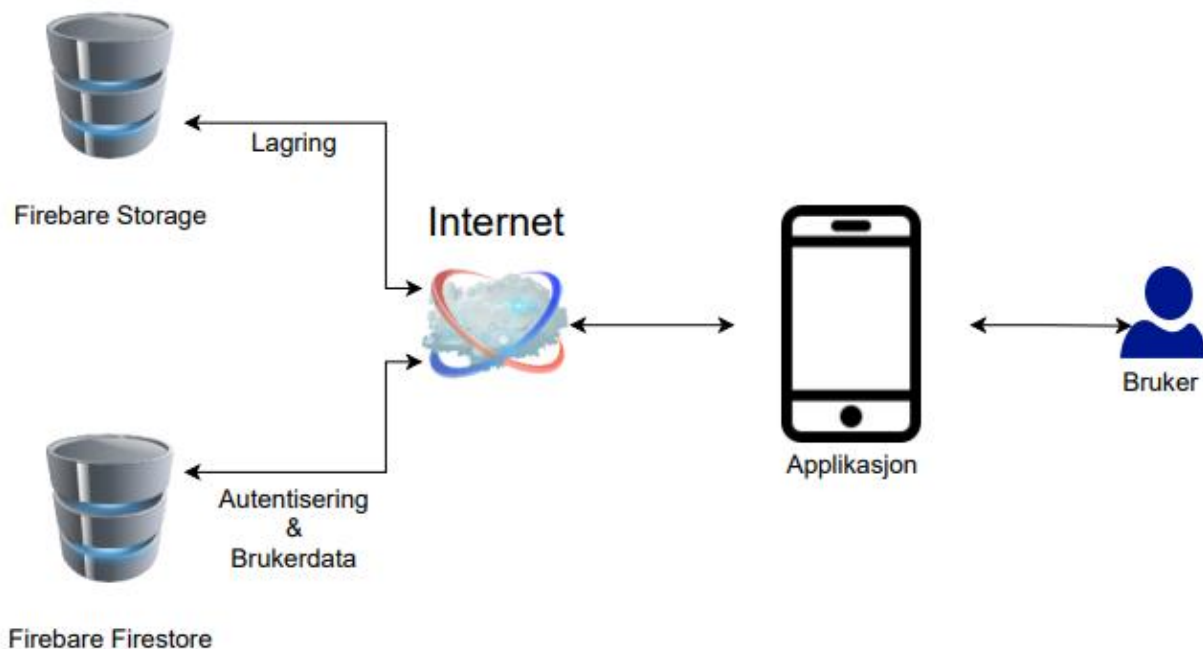
Dette for å luke bort eventuelle feil eller finne forbedringer som kan gjøres. Etter testperioden kan applikasjonen lastes opp til Appstore og Google Play.

4. Systemdokumentasjon

4.1 Systemarkitektur

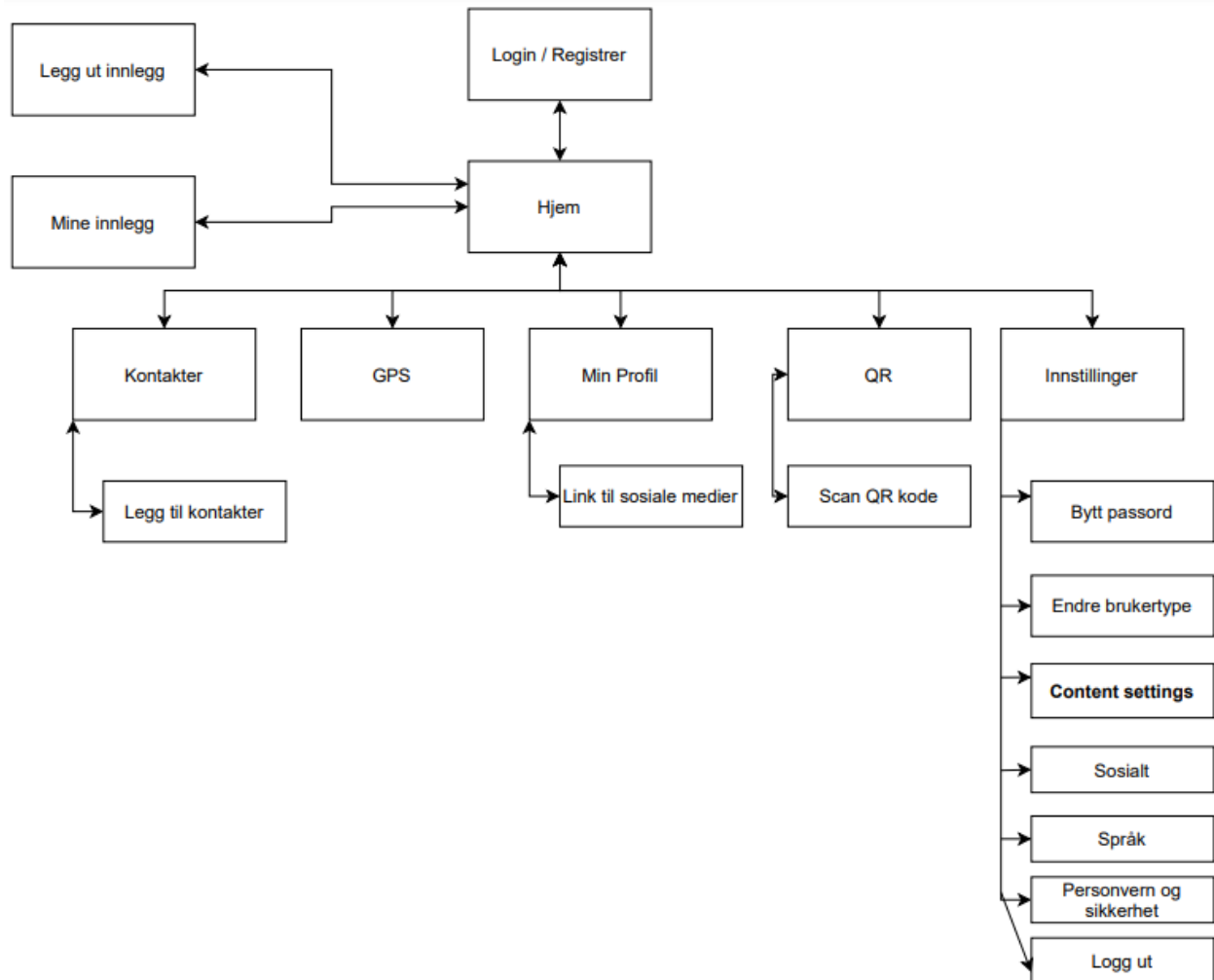
Systemarkituren er en konseptuell modell som beskriver strukturen og oppførselen til et system.(Sawakinome, 2021)

Ettersom vi har alt av lagring av data hos Firebase så skjer alt over skyen. Vi bruker Firestore til autentisering av bruker når de skal logge inn/registreres og for lagring av bilder så bruker vi Firebase storage og linker til bildet i Firestore slik at brukeren kun får opp bilde de har lastet opp og ikke noen andre sitt.



Figur 8, Systemarkitektur

Sitemap



Figur 9, Sitemap

4.2 Database

Databasen vi har valgt å bruke er Firestore, dette er en såkalt “NoSQL” database. Dette vil si at det ikke er relasjoner mellom entitetene, men heller en mappestruktur.

Vi har to hoved mapper i vårt system. “posts” og “users”. Posts inneholder all informasjon om innleggene alle brukerne har gjort, i mens Users inneholder all persondata om brukerne. Nedenfor ligger en dypere forklaring om hva de ulike mappene inneholder og deres datainnhold.

Posts:

I denne mappen ligger alle innleggene identifisert med sin egen unike identifikator(id), hvert innlegg har igjen sine egne “undermapper”. De er som følger:

Dato: Datoen innlegget ble opprettet.

Farge: Fargen som blir fremvist på kortet, fargen er basert på om kortet er et privat kort eller firmakort.

Firmabilde: Hvis brukeren er tilkoblet et firma vil dette feltet være fylt med posisjonen til bildet på serveren vår.

Kategori: Viser hvilke kategori kortet hører hjemme i, f.eks. Arrangement eller Privat.

Navn: Navnet på brukeren som la ut innlegget.

Profilbildet: Det private bilde til brukeren som la ut bilde.

Tekst: Her ligger teksten som blir presentert i innlegget.

Type: Inneholder enten “personkort” hvis innlegget er bare fra en privat bruker eller “firmakort” hvis innlegget er fra et firma.

Uid: Her ligger identifikatoren til brukeren som opprettet innlegget.

Users:

I denne mappen ligger alle brukerne, de er også lagret med sine egne identifikatorer. Feltene underlagt hver bruker er:

Adresse: Adressen til bruker.

Biografi: Kort fortalt om bruker.

Contacts: Dette er en liste med alle andre brukere på plattformen som bruker har lagt til. Denne blir oppdatert hver gang bruker legger til nye eller sletter kontakter.

Email: Mailen til bruker.

Etternavn: Etternavnet til bruker.

Firma: Hvis bruker tilhører et firma kan det lagres her.

FirmaBilde: Hvis bruker tilhører firma kan deres bilde lagres her.

Fornavn: Fornavnet til bruker.

Instagram: Tilkobling til Instagram.

Posisjon: Sist oppdaterte posisjon til bruker.

Profilbilde: Personlige bilde til bruker.

Telefon: Telefonnummeret til bruker.

Tittel: Stillingstittel.

Type: Inneholder enten “Privatkort” eller “Firmakort” basert på om bruker er privat eller firma.

UID: Her ligger identifikatoren til bruker.

Youtube: Hvis bruker er tilkoblet en youtube konto vil den ligge her.

5. Diskusjon

I denne delen av rapporten vil vi gå igjennom valgene som ble tatt underveis i prosjektet og hvordan vi kom frem til de.

Det første valget vi måtte bli enige om innad i gruppen var hvilken utviklingsmetode vi skulle velge når vi skulle starte. Vi visste allerede fra tidligere prosjekter at det lureste var å velge en Agile arbeidsprosess for å gi litt mer frihet blant gruppemedlemmene. Det sto i all hovedsak mellom Kanban og Scrum som potensielle utviklingsmetoder. Vi så på fordelene og ulempene med begge og konkluderte med at vi ikke ønsket at et gruppemedlem skulle være “sjef” over de andre i form av Scrum leader, men heller ha et sunt og likestilt arbeidsmiljø gjennom Kanban. I tillegg var friheten til Kanban mye mer tiltrekkende enn de full-fokuserte sprintene til Scrum.

Etter at vi valgte utviklingsmetode måtte vi bli enige om hvilke utviklingsverktøy vi skulle bruke for utvikling av applikasjonen. Oppdragsgiveren ønsket at denne applikasjonen vi skulle utvikle skulle fungere på både IOS og Android, det var en ny utfordring for alle i gruppa å utvikle applikasjoner for IOS-enheter. Dette betydde at vi måtte sette oss inn i noe nytt, vi undersøkte en del på hvilke muligheter vi hadde og konkluderte med at vi enten måtte lære oss React Native eller Flutter. Igjen bestemte vi oss for å se på fordelene og ulempene med hver av dem og undersøkte på nett for å hjelpe og finne ut hvilket utviklingsverktøy som passet best for oss og denne applikasjonen. Konklusjonen var altså at Flutter passet mye bedre for oss som ønsket at appen skulle fungere helt likt for både IOS og Android. I tillegg er Flutter lansert av Google som betyr

Side 43 av 68

at det enkelt kan knyttes mot Firebase som vi har hatt tidligere erfaring med. Så på grunn av kravene fra oppdragsgiver, enkelhet å lære seg og sammenkobling med Firebase konkluderte vi at Flutter var utviklingsverktøyet vi ønsket å bruke for denne oppgaven.

Videre i utviklingsprosessen bestemte vi oss også for å bruke Github for å holde versjonskontroll og oversikt over nye oppdateringer som skjer under utviklingen. Alle på gruppen har hatt erfaring med Github før så det var en ensidig stemme for å bruke det, men det var fortsatt et valg vi aktivt tok. Med Github har vi også kunne jobbet både samtidig og hver for oss igjennom grener. Vi har hatt en “mester-gren” som inneholder den nyeste versjonen av produktet med 0 error-meldinger. I tillegg har vi hatt flere egne grener slik at vi kan jobbe med vårt eget arbeid uten å komme i veien for andre, så kobler man alt sammen mot mester-grenen for å få inn alt arbeidet gruppemedlemmene har bidratt med.

6. Konklusjon

Vi fikk i oppgave å lage en plattform for bedrifter og enkeltpersoner å holde styr på visittkort, samt gjøre det enklere å dele sine egne gjennom en applikasjon som skal fungere på både IOS og Android-enheter. Vi føler at vi har klart å utvikle et bra og funksjonelt program som holder opp mot dagens standard og forventningene til oppdragsgiveren.

Det har vært en hard, men utrolig lærerik prosess å jobbe med denne oppgaven. Vi har hatt konstant fremgang med et effektivt samarbeid og fullt engasjement fra alle parter. Vi har hatt tett oppfølging med både rådgiver og oppdragsgiver som har ført til et godt samvær mellom alle partene. Dette har vært viktig for oss slik at vi er sikker på at alt vi gjør er etter oppdragsgivers ønsker og vi har fått masse innspill for hvordan det ønskes at vi skal implementere funksjoner og fikse på utseende på applikasjonen.

Vi er veldig fornøyd med utviklingsmetoden vi valgte under oppstart av prosjektet, Kanban har gitt oss den friheten vi trenger samtidig som at vi har hatt et klart syn på målene fremfor oss og datoer på når funksjoner skal være ferdig. Dette har også vært med på å bidra til et godt samvær i gruppa, noe som har vært veldig viktig for oss.

Vi er stolte over produktet som vi leverer og fremgangsmåtene vi har brukt for å utvikle det. I tillegg er vi veldig fornøyd med oss selv som gruppe, vi har vært engasjerte, støttende og virkelig forstått det vi holder på med. Vi er også fornøyde med både rådgiver og oppdragsgiver, de har vært flinke til å komme med tilbakemeldinger og gjennom sin egen entusiasme engasjert oss til å jobbe enda hardere.

7. Referanser

Dennis, Alan. Wixom, Barbara H. Roth, Roberta M.(2012) *System Analysis And Design (5th ed)*. USA: John Wiley & Sons, Inc.

Oracle (2021), *What is database*, Hentet 09.05.21, fra:

<https://www.oracle.com/database/what-is-database/>

Lemay, Laura. Colburn, Rafe. Kyrnin, Jennifer.(2016) *Sams Teach Yourself: HTML,CSS & Javascript Web publishing (7th edition)*. USA. Sams publishing.

Stephens, Rod. (2015). *Beginning Software Engineering (1st ed.)*. Indianapolis: John Wiley & Sons, Inc.

Kristoffersen, B. (2016). *Databasesystemer (4.utgave)*. Norge: Universitetsforlaget

Flutter. (2021, 18/05/21). Flutter - Beautiful native apps in record time. Hentet fra <https://flutter.dev/>

Material Design (2021, 19/05/21). Material Design - Introduction Hentet fra <https://material.io/design/introduction>

Flutter. (2021). State Management. Hentet fra: <https://flutter.dev/docs/development/data-and-backend/state-mgmt/intro>

Babich, N. (2017, 29.November). *Prototyping 101: The difference between Low-Fidelity and High-Fidelity Prototypes and When to Use Each*. Hentet fra:

<https://blog.adobe.com/en/publish/2017/11/29/prototyping-difference-low-fidelity-high-fidelity-prototypes-use.html#gs.17xqq0>

Sawakinome. (2021, 19/05/21). *Hva er forskjellen mellom systemarkitektur og programvarearkitektur*. Hentet fra:

<https://no.sawakinome.com/articles/technology/what-is-the-difference-between-system-architecture-and-software-architecture.html>

Thomas, G. (2019, 18/05/21). What is Flutter and Why You Should Learn it in 2020. Hentet fra <https://www.freecodecamp.org/news/what-is-flutter-and-why-you-should-learn-it-in-2020/>

Stevenson, D. (2018, 18/05/21). What is Firebase? The complete story, abridged. Hentet fra <https://medium.com/firebase-developers/what-is-firebase-the-complete-story-abridged-bcc730c5f2c0>

Gravevang, M. (2020, 18/05/21). Frontend. Hentet fra <https://snl.no/frontend>

Granevang, M. (2020, 18/05/21). Backend. Hentet fra <https://snl.no/backend>

Lid, I. (2020, 18/05/21). Universell Utforming. Hentet fra https://snl.no/universell_utforming

Rossen, E. (2020, 18/05/21). API. Hentet fra <https://snl.no/API>

Tubik. (2017, 18/05/21). UI/UX Design Glossary. Navigation Elements. Hentet fra <https://uxplanet.org/ui-ux-design-glossary-navigation-elements-b552130711c8>

Statistisk sentralbyrå. (2020, 25. mars). Hvor mange jobber er det i Norge?. Hentet fra <https://www.ssb.no/arbeid-og-lonn/artikler-og-publikasjoner/hvor-mange-jobber-er-det-i-norge#:~:text=Statistikken%20viser%20at%20det%20i,det%201%20421%20000%20jobber.>

Uutilsynet. (2021, 20/05/21). Universell utforming av apper. Hentet fra: <https://www.uutilsynet.no/regelverk/universell-utforming-av-apper/230>

Hannah, Jaye. (2021, 18. April). What Exactly is Wireframing? A Comprehensive Guide. Hentet fra: <https://careerfoundry.com/en/blog/ux-design/what-is-a-wireframe-guide/>

Discord (2021), What is Discord, Hentet 10.05.2021, fra:
<https://discord.com/safety/360044149331-What-is-Discord>

Trello (2021), What is trello, Hentet 10.05.2021, fra:
<https://help.trello.com/article/708-what-is-trello>

Android Studio (2021), Intro, Hentet 11.05.2021, fra
<https://developer.android.com/studio/intro>

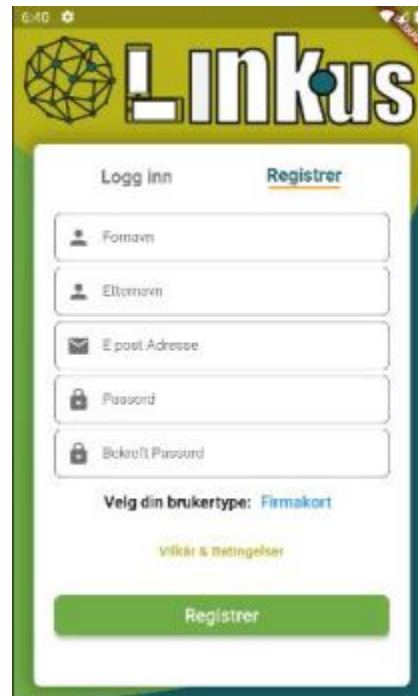
Adobe xD (2020) What is adobe xd used for, Hentet 10.05.2021, fra
<https://developer.android.com/studio/intro>

Github. (2020). What is Github, Hentet fra: <https://guides.github.com/activities/hello-world/#branch>

Firebase. (2021). Documentation, Hentet fra
<https://firebase.google.com/docs/reference/js/firebase.auth.Auth#createuserwithemailandpassword>

8. Vedlegg

8.1 Brukerveiledning



Man registrerer seg ved å trykke “registrer” til høyre, eller om man allerede har en bruker velger man “logg inn” til venstre. Man kan også velge å logge inn med linkedIn eller en google konto, slik at man får knyttet en ny konto opp mot det man allerede har.

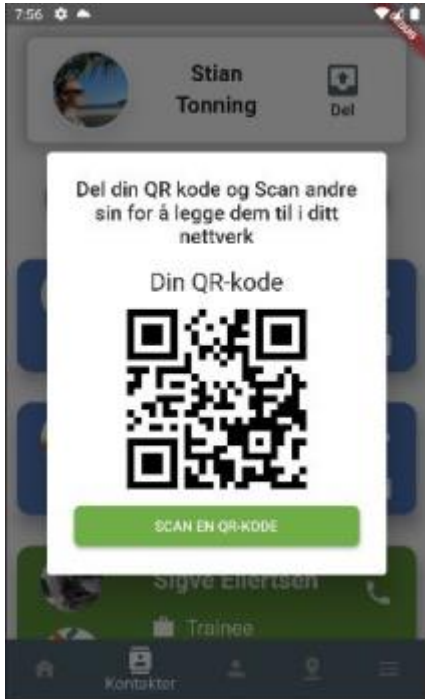


Det første som kommer opp når man har logget seg inn er forsiden, der man får opp de nyeste oppdateringene i appen. Ønsker man å legge ut en oppdatering/innlegg trykker man på pluss-tegnet (+) der det vil komme opp en rekke valg for å kategorisere sitt innlegg. Valgene er arrangement, privat, firma og ... Hver kategori har sin egen farge. På bildet til venstre ser man grønn; som er et privat innlegg, og gul; som er et arrangement.

Nedenfor pluss-tegnet, finner vi tegnet for "oversikt". Der kan man se tidligere innlegg, og annen historikk.

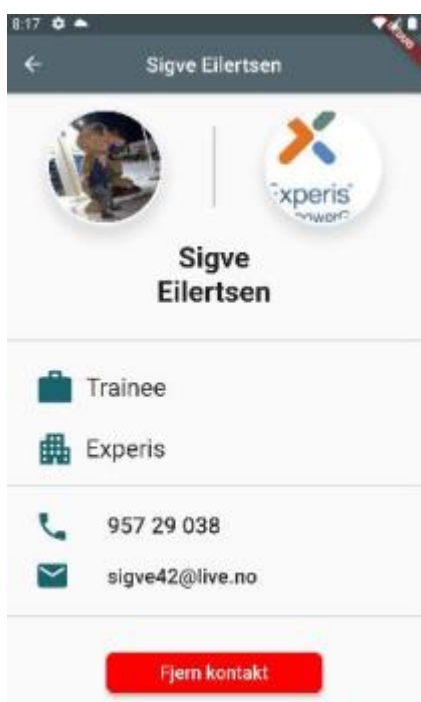
Man scroller opp og ned for å se de ulike innleggene som er blitt lagt ut av de man har delt og fått delt visittkortet med. Dette bidrar til at man enkelt kan finne det som fanger interesse for den enkelte

applikasjons-brukeren.



Øverst ser man sitt eget navn og kan lett trykke til høyre på ikonet “del” for å dele QR kode til andre, eller man kan scanne QR koden til en annen bruker for å legge dem til i sitt LinkUs nettverk.

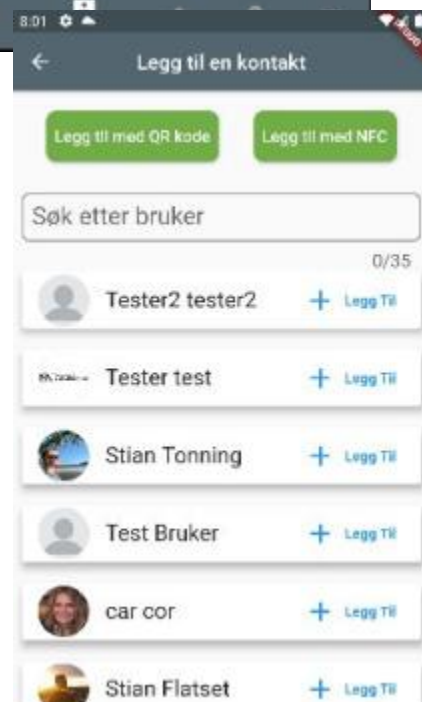
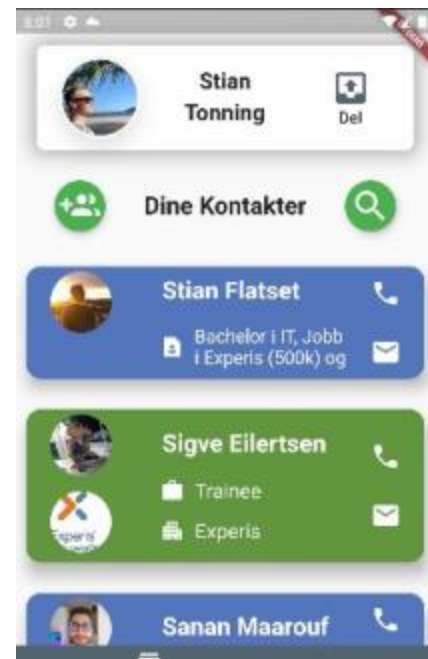
På kontakter finner man de brukerne man har lagt til eller som har lagt til deg i en oversikt med tydelige navn for å enkelt finne de man har interesse av å kontakte. Man kan ringe eller sende mail for å ta direkte kontakt, noe man gjør etter man har funnet det navnet man har interesse av. Nedenfor navnet er det punktvis noe info om brukeren, blant annet arbeidsplass og bosted.



Trykker man på navnet får man opp all informasjonen om brukeren. Ønsker man å fjerne kontakten er det et rødt synlig ikon “Fjern kontakt” som ligger inne på hvert enkelt navn.

Her kan man også finne andre brukere som man ikke allerede har. Ikonet for å legge til flere brukere er til venstre, der man enten kan søke etter navn, eller legge til med QR-kode eller NFC. Har man lagt til mange kontakter, kan man enkelt velge å søke på et navn ved hjelp av søkeikonet opp til høyre,

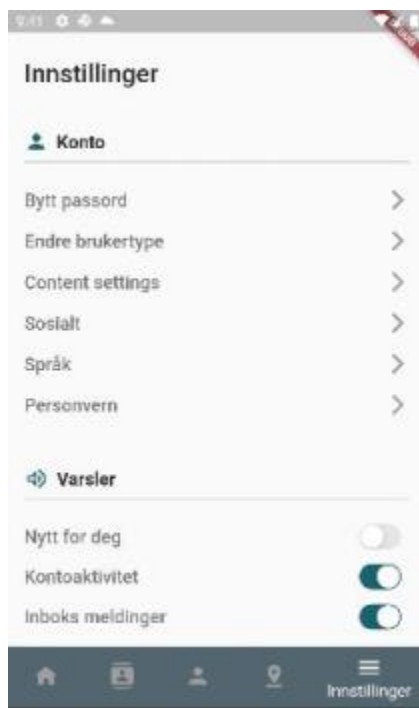
istedet for å scrolle gjennom hele kontaktlisten.





På “Min profil” kan man endre og redigere navn, etternavn, mobilnummer, adresse/bosted, email biografi og profilbilde. Man velger den grønne knappen “Lagre endringer” når man har endret noe av dette. Også på denne siden kan man velge å koble til sin LinkUs profil/bruker til andre sosiale medier, dersom man ønsker det. Her er valgene Facebook, Youtube, Instagram og LinkedIn.

GPS-en gjør at man enkelt kan finne brukere i nærheten, og kan her zoome enten med manuelt på kartet eller ved å trykke på pluss eller minus nederst til høyre. Øverst til høyre er knappen for egen lokasjon, dersom man vil vite hvor på kartet man befinner seg.



På innstillinger finner man kontooversikt og varsel-oversikt. Under konto kan man bytte bruker, endre brukertype, Content settings, sosialt, språk og personvern & sikkerhet. Under varsler huker man av for om man ønsker å få opp nytt for deg, kontoaktivitet og innboksmeldinger. Det er også her man logger ut ved å trykke på ikonet nederst i midten “Logg ut”.

8.2 Figurliste

Figur 1, Tidsplan - 15

Figur 2, Systemkrav - 20

Figur 3, Bruksmønster for å legge til kontakt - 21

Figur 4, Wireframe - 22

Figur 5, Hi-Fi prototype - 23

Figur 6, Fargepalett - 24

Figur 7, NoSQL database - 26

Figur 8, Systemarkitektur - 37

Figur 9, Sitemap - 38

Figur 10, Mal til bruksmønster - 54

Figur 11, Bruksmønster for å opprette bruker - 54

Figur 12, Bruksmønster for å redigere brukerinfo - 55

Figur 13, Bruksmønster for anrop/mail direkte via visittkort - 55

Figur 14, Bruksmønster for deling av lenke - 56

Figur 15, Bruksmønster for å legge til kontakt - 56

8.3 Gruppekonsakt

Gruppekonsakt

Kontaktinfo innad i gruppen:

Navn	Mobil / Telefon	Epost	Studentnr
Sanan Maarouf	47264992	Sanan.adnan97@gmail.com	216374
Erlend Hollund	46805976	Erlend.hollund29@gmail.com	224914
Stian Tonning Flatset	90142467	Stian.tonning@hotmail.no	224935
Sigve Eilertsen	95729038	Sigve42@live.no	224882

Generelt:

- Fravær og oppgaver som ikke er utført loggføres i Trello for at alle skal ha god oversikt over prosjektet.
- Stian er ansvarlig for planlegging av møter, avgjørelser gjøres i felleskap i gruppen.
- Alle gruppemedlemmer møter på gruppemøter til avtalt tid, eventuelle forsinkelser meldes på Facebook Messenger.
- Gruppemøter avholdes primært mellom 8 og 16 med mindre annet avtales og bekreftes av alle i gruppen.
- Vi gjennomgår avtalt tema på hvert gruppemøte.
- Vi har et bevisst forhold til at jobb og andre aktiviteter ikke kommer foran prosjektet.
- Hvis et gruppemedlem ikke møter opp uten forvarsel gjentatte ganger blir dette tatt opp i felleskap. Hvis det ikke forbedres vil veileder kontaktes.
- Gruppen ønsker å oppnå karakteren A.
- Møte presis til avtalt tidspunkt. Respekt for andres tid.
- Taushetsløfte innad i gruppa, dette gjelder både arbeidsmessig og privat informasjon som måtte komme frem under samarbeidet i gruppa.
- Vi respekterer hverandre og tar vare på hverandre.

8.4 Bruksmønstre

For å opprette mal for bruksmønstre brukte vi Roth, Dennis og Wixom sin forklaring av bruksmønstre (Dennis, Wixom, Roth, 2012, s. 149).

Mal av bruksmønstre:

Navn	Enkel beskrivelse av bruksmønsteret
Prioritet	Prioriteten av bruksmønsteret i systemet. Rangeres fra lav, middels til høy.
Forutsetning	Forutsetning før bruksmønsteret starter
Hendelse	Hva som utføres i bruksmønsteret / målet med bruksmønsteret
Alternativ hendelse	Alternative hendelser som kan oppstå i prosessen (Hvis det er ingen alternativer skal feltet stå tomt).
Unntak	Unntak som kan oppstå i prosessen.

Figur 10, mal til bruksmønstre

Oversikt over våre bruksmønstre:

Navn	Opprette en bruker
Prioritet	Høy
Forutsetning	<ol style="list-style-type: none">1. Linkus applikasjonen må være nedlastet på enheten.2. Brukeren må være tilkoblet internett.
Hendelse	<ol style="list-style-type: none">1. Brukeren navigerer til siden "Registrer".2. Brukeren må fylle inn de nødvendige inntastingsfeltene; navn, epost og passord.3. Brukeren velger sin brukertype; firmakort eller personkort.4. Brukeren trykker på "Registrer" og får opprettet sin bruker.
Alternativ hendelse	<ol style="list-style-type: none">1. Brukeren ønsker å logge inn med Google eller LinkedIn<ol style="list-style-type: none">a. Gjøres fra "Logg inn" sidenb. Autorisering gjøres gjennom Google/LinkedIn
Unntak	<ol style="list-style-type: none">1. Brukeren skriver inn en ugyldig epost-adresse2. Passordfeltene må samsvare med hverandre for å opprette bruker3. Brukeren aksepterer ikke våre vilkår og betingelser

Figur 11, Bruksmønstre for å opprette bruker

Navn	Redigering av brukerinfo
Prioritet	Middels
Forutsetning	<ol style="list-style-type: none"> 1. Applikasjonen må være nedlastet på enheten. 2. Brukeren må ha tilgang til internett og være logget inn.
Hendelse	<ol style="list-style-type: none"> 1. Brukeren navigerer til siden "Min Profil". 2. Brukeren endrer informasjonen i tekstfeltene. 3. Brukeren trykker på "Lagre endringer" som oppdaterer databasen.
Alternativ hendelse	
Unntak	<ol style="list-style-type: none"> 1. Brukeren mister tilgang til internett. 2. Feil i oppkobling til database. <p>I begge tilfeller blir ikke brukerens data oppdatert i databasen.</p>

Figur 12, Bruksmønster for å redigere brukerinfo

Navn	Gjøre anrop og sende mail direkte via visittkortet
Prioritet	Høy
Forutsetning	<ol style="list-style-type: none"> 1. Brukeren må være innlogget i applikasjonen. 2. Brukeren må ha kontakter registrert. 3. Brukeren som skal kontaktes må ha lagt inn sitt tlfnr/email-adresse.
Hendelse	<ol style="list-style-type: none"> 1. Brukeren navigerer til siden "Kontakter". 2. Brukeren finner ønsket kontakt og kan direkte trykke på kontaktens telefonnummer eller epost-adresse. 3. Brukeren blir videresendt til enhetens anrop applikasjon eller email applikasjon
Alternativ hendelse	<ol style="list-style-type: none"> 1. Brukeren har ingen kontakter, og må derfor legge kontakten til.
Unntak	<ol style="list-style-type: none"> 2. Kontakten som skal kontaktes har ikke delt mobilnr eller email adresse.

Figur 13, Bruksmønster for anrop/mail direkte via visittkort

Navn	Oppkobling/Deling av ønsket sosiale medier
Prioritet	Middels
Forutsetning	1. Brukeren må være innlogget i applikasjonen.
Hendelse	<ol style="list-style-type: none"> 1. Brukeren navigerer til “Min Profil”. 2. Brukeren velger hvilket sosiale medie det skal oppkobles til. 3. Brukeren skriver inn sitt profilnavn på valgt sosiale medie. 4. Brukeren trykker “Oppdater” for at oppkoblingen skal oppdateres i databasen 5. Brukerens sosiale medie vil nå være tilgjengelig for dens kontakter.
Alternativ hendelse	
Unntak	1. Brukeren skriver inn feil profilnavn, og oppkoblingen kommer til feil side evt. Feilmelding

Figur 14, Bruksmønster for deling av lenke

Navn	Legge til en kontakt
Prioritet	Høy
Forutsetning	2. Brukeren er innlogget i applikasjonen
Hendelse	<ol style="list-style-type: none"> 5. Brukeren navigerer til siden “Kontakter”. 6. Brukeren trykker på legg til kontakt ikonet, som åpner siden for å legge til en kontakt 7. Brukeren taster inn navnet på kontakten som ønskes, og legger den til. 8. Kontaktlisten til brukeren oppdateres i databasen og brukeren har nå tilgang til kontaktens informasjon
Alternativ hendelse	<ol style="list-style-type: none"> 2. Legge til kontakt ved bruk av QR-kode eller NFC <ol style="list-style-type: none"> a. Brukeren er i nærheten av kontakten som ønskes å legges til b. Brukeren navigerer til “Legg til en kontakt” siden. c. Her velges enten “Legg til med QR-kode” eller “Legg til med NFC”. d. QR-koden skannes evt. NFC mellom brukerne. e. Kontaktlisten til brukeren oppdateres i databasen, og

	brukeren har nå tilgang til kontaktens informasjon
Unntak	3. Skrivefeil som gjør at kontakten ikke dukker opp. 4. Brukeren mister tilgang til internett

Figur 15, Bruksmønster for å legge til kontakt

8.5 Møtereferat

Vi har holdt jevnlige møter med oppdragsgiver og veileder. Vi har laget en mal som oppsummerer møtet.

Mal for møtereferat

Tidspunkt	
Tilstede	
Hva som ble drøftet	

Liste over møter:

- 16. Oktober 2020
- 29. Januar 2021
- 11. Februar 2021
- 22. Februar 2021
- 11. Mars 2021
- 14. April 2021
- 14. Mai 2021
- 18. Mai 2021

Tidspunkt	16. Oktober 2021
-----------	------------------

Tilstede	Stian Tonning Flatset, Sigve Eilertsen, Sanan Adnan Maarouf, Erlend Hollund og Caroline Corneliussen
Hva som ble drøftet	Krav og forventinger til applikasjonen. Inngåelse av samarbeid og kontrakt.

Tidspunkt	29. Januar 2021
Tilstede	Stian Tonning Flatset, Sigve Eilertsen, Sanan Adnan Maarouf, Erlend Hollund, Caroline Corneliussen og Rania El-Gazzar
Hva som ble drøftet	Nyhetsvegg for innlegg, liste over kontakter, wireframes og at applikasjonen skal i første omgang være på norsk.

Tidspunkt	11. Februar 2021
Tilstede	Stian Tonning Flatset, Sigve Eilertsen, Sanan Adnan Maarouf, Erlend Hollund, Caroline Corneliussen og Rania El-Gazzar
Hva som ble drøftet	Personvern, legge til en kontakt ved å søke, oppretting av privat og bedriftskonto

Tidspunkt	22. Februar
Tilstede	Stian Tonning Flatset, Sigve Eilertsen, Sanan Adnan Maarouf, Erlend Hollund, Caroline Corneliussen og Rania El-Gazzar
Hva som ble drøftet	Fargevalg - fargepalette, ikon til applikasjonen, logo for bruk i applikasjonen

Tidspunkt	11. Mars
-----------	----------

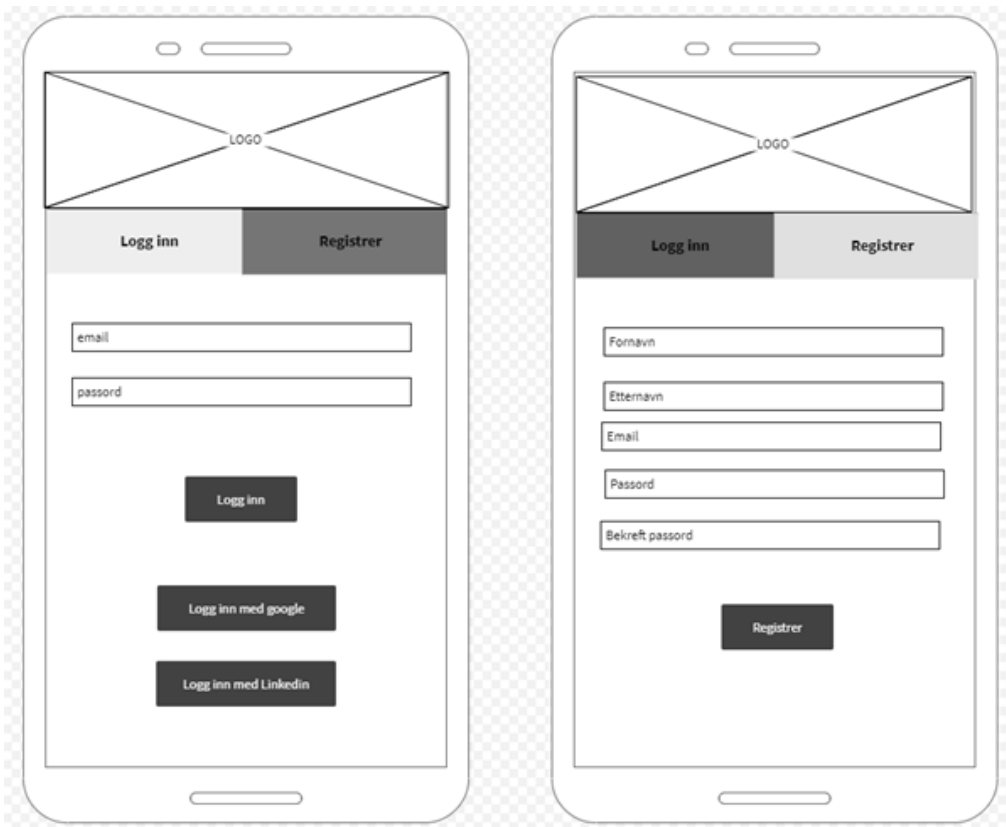
Tilstede	Stian Tonning Flatset, Sigve Eilertsen, Sanan Adnan Maarouf, Erlend Hollund og Caroline Corneliussen
Hva som ble drøftet	Link til Trello, gi Caroline tilgang til Firebase

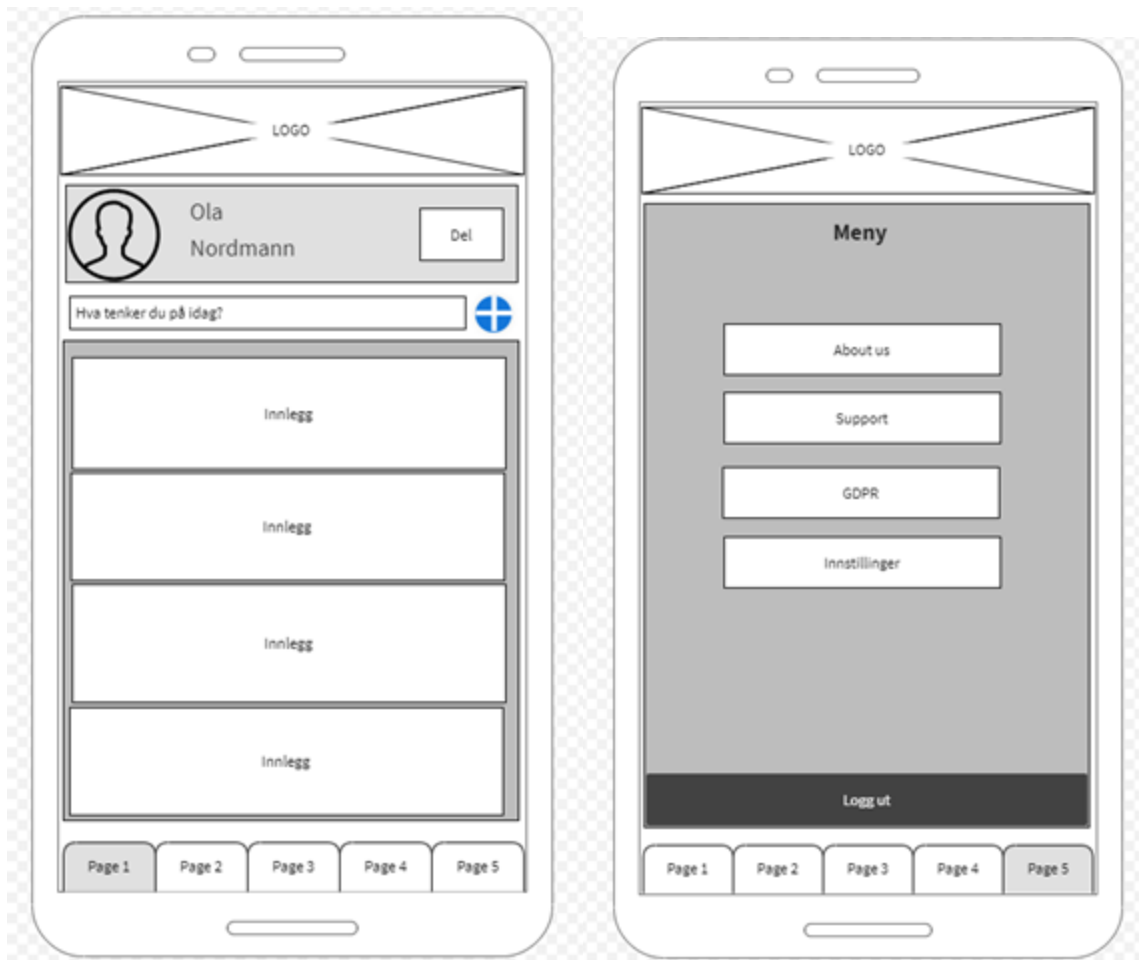
Tidspunkt	14. April
Tilstede	Stian Tonning Flatset, Sigve Eilertsen, Sanan Adnan Maarouf, Caroline Corneliussen og Rania El-Gazzar
Hva som ble drøftet	Generell gjennomgang av applikasjonen, drøfting om utseende av nyhetsvegg og kontaktliste, installering av APK-fil til Caroline og Rania

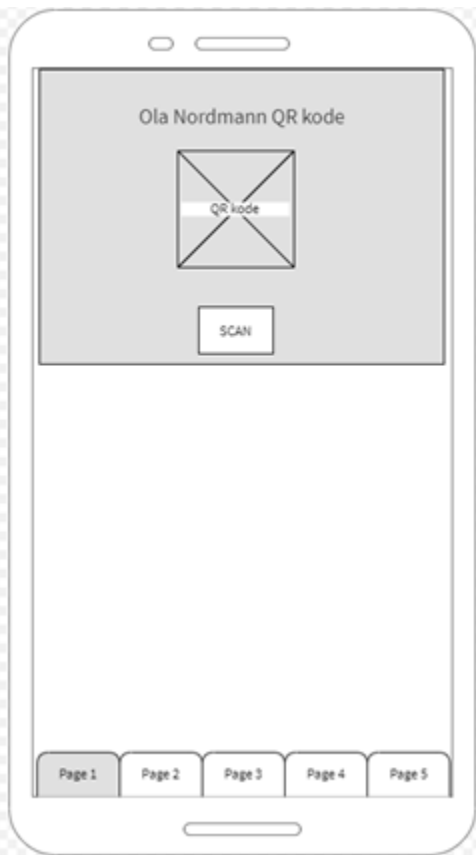
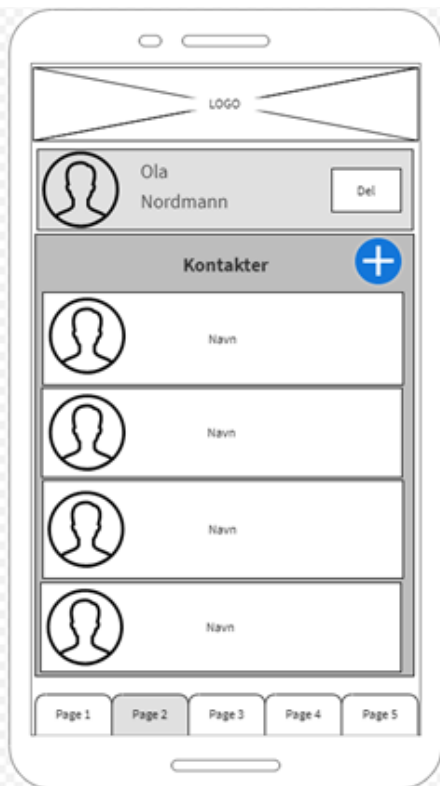
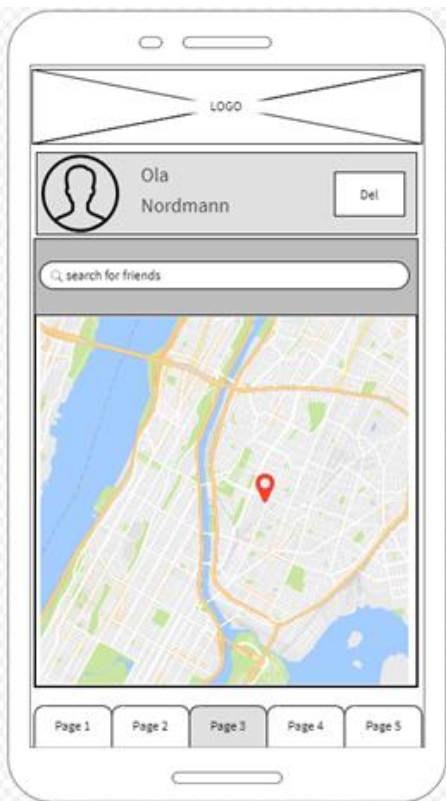
Tidspunkt	14. Mai
Tilstede	Stian Tonning Flatset, Sigve Eilertsen, Sanan Adnan Maarouf, Erlend Hollund og Rania El-Gazzar
Hva som ble drøftet	Gjennomgang av applikasjonen med veileder, samt generelle spørsmål om rapporten og presentasjon

Tidspunkt	18. Mai
Tilstede	Stian Tonning Flatset, Sigve Eilertsen, Sanan Adnan Maarouf, Erlend Hollund og Caroline Corneliussen
Hva som ble drøftet	Gjennomgang av applikasjonen og overlevering av applikasjonen.

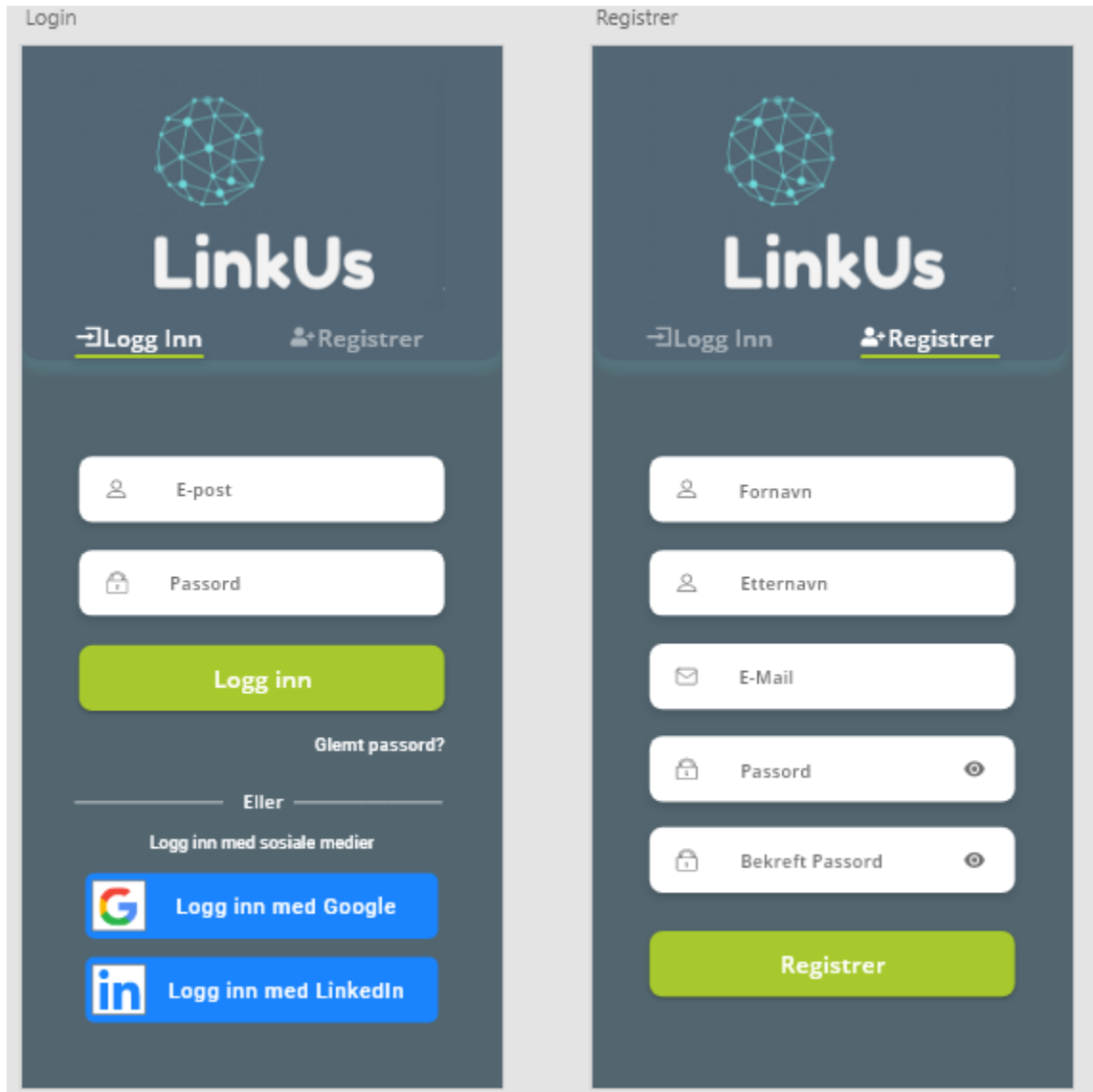
8.6 Wireframes







8.7 Hi-fi prototype





Kari Nordmann



Hva har du lyst til å snakke om?



Politiet
Ola Nordmann



Jeg holder konferanse i Abrahamshallen onsdag 25.02. Les mer om arrangementet her:
www.politiet.no



USN
Erik Eriksen



USN tilbyr flere masterprogram med lønn i studietiden! Sjekk ut hvordan du kan ta master med lønn i studietiden! <http://usn.no>



Kiwi
Emma Nordmann



Vi åpner ny butikk i Norgesgata den 06.03 Ekstra gode tilbud på åpningsdagen! Kom innom våre nye lokaler, vi sees!



Elkjøp
Nora Nordmann



Hjem



Kontakter




GPS






Min profil









Meny










 **Kari Nordmann** 






Kontakter 

 **Ola Nordmann**
 B-Klikken
 test@mail.com

 **Ola Nordmann**
 B-Klikken
 test@mail.com

 **Ola Nordmann**
 B-Klikken
 test@mail.com

 **Ola Nordmann**
 B-Klikken
 test@mail.com 

 **Hjem**  **Kontakter**  GPS  Min profil  Meny



 **Kari Nordmann** 

Ola Nordmann
CEO

 McLaren Lørenskog  test@mail.com

 J E Abrahamsens veg 41  924 63 847

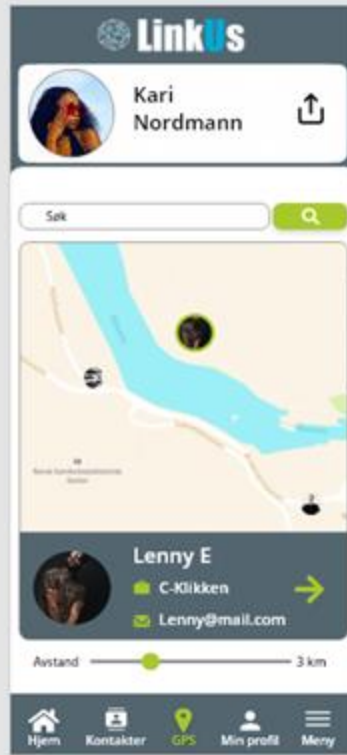
Fjern kontakt 

 **Hjem**  **Kontakter**  GPS  Min profil  Meny

GPS



GPS valgt kontakt



GPS valgt kontakt full profil



