

Prosjektrapport IN2000

Case 3: Vær- og farevarsler for yngre brukere



VærSmart Av Team 13

Medlemmer:

- Muntaha Dheeg – muntahad
- Shanza Ehsan – shanzae
- Hanan Mohamud – hananam
- Andreas Klæboe – andrklae
- Christian Thorkildsen – christft
- Viktoriia Olonova – viktorol

Institutt for Informatikk
Det matematisk naturvitenskapelige fakultet

UNIVERSITETET I OSLO

2024

Veiledere: Simon Tidemann Halvorsen & Julia Lundeby

Innholdsfortegnelse

1. Presentasjon	4
1.1 Valgt case	4
1.2 Vår løsning	4
1.3 Hvem er vi?	5
2. Brukerdokumentasjon	5
2.1 Innsiktsarbeid	5
2.2 Applikasjonens funksjonalitet	7
2.3 UI og struktur	8
2.3.1 Appens UI	8
2.3.2 Appens arkitektur	11
2.4 Tilgang til appen	11
3. Produktdokumentasjon	12
3.1 Kvalitetsegenskaper	12
3.1.1 Forskningsmetode – Brukerundersøkelser	12
3.1.2 Evaluering av egenskaper	13
3.1.3 Svakheter i applikasjonen	13
3.2 API-er	14
3.2.1 API-nøkler	14
3.2.2 API-nivå	15
3.2.3 Beskrivelse av API-er	15
3.3 Universell utforming	18
3.4 Mr. Praktisk avatar og animasjon	19
4. Prosessdokumentasjon	20
4.1 Arbeidsmetodikk	20
4.2 Prosjektets gang	21
4.3 Endringer i kravspesifikasjonen	23
4.4 Testing	25

5. Refleksjon	27
6. Avslutning	28
Referanser	29
Vedlegg	31
Vedlegg 1: Spørreundersøkelse og resultater	31
Vedlegg 2: Intervjuguide og intervjunotater for målgruppen 9 - 25 år.	33
Vedlegg 3: Personer	35
Vedlegg 4: Funksjonelle og ikke-funksjonelle krav	36
Vedlegg 5: Low-fidelity og high-fidelity prototype	39
Vedlegg 6: Samtykkeskjema for geriljatesting og brukerundersøkelsen	41
Vedlegg 7: Geriljatesting	53
Vedlegg 8. Brukertesting	55
Vedlegg 9: Felttesting	56
Vedlegg 10: Gantt diagram	57
Vedlegg 11: Kanban board	59
Vedlegg 12: Milepælsplan	59

1. Presentasjon

1.1 Valgt case

I denne oppgaven er case 3- “Vær og farevarsel for yngre brukere” valgt. Gjennom årene har Meteorologisk institutt hatt en betydelig dekningsgrad blant det norske publikum gjennom et bredt spekter av mediekanaler; radio, TV og Yr på web og mobilapper. Imidlertid har MET identifisert en utfordring med å nå ut til en viktig målgruppe - brukere fra 9 til 25 år, som verken ser lineær-TV, hører på DAB-radio eller bruker Yr. Til vår app, *VærSmart*, har vi valgt å øke nedre alder i målgruppen til 13, fordi appen vår benytter KI, og OpenAI er ment for brukere over 13 år (OpenAI, 2024).

Formålet med denne rapporten er å vise fremgangsmåten for å identifisere hvordan MET kan nå ut til denne yngre målgruppen, spesielt når det gjelder farevarsler, og presentere vår løsning og prosess. Vi har kartlagt behovene og ønskene til målgruppen for å forstå hvorfor de bruker alternative værtjenester, og hva som skal til for at en vær app bedre tilfredsstiller deres behov.

1.2 Vår løsning

Basert på vår datainnsamling har vi bestemt oss for at *VærSmart* skal være en enkel vær-app som kun gir den viktigste informasjonen om været og nærliggende farevarsler. Målgruppen er alt fra 13-åringar som ikke har kunnskapen til å kunne forholde seg til komplisert værdata, til 25-åringar som kanskje bryr seg om detaljer. Vi ønsker at appen skal tilfredsstille begge endene av målgruppen. Derfor skal brukeren kunne forstå været gjennom illustrasjoner og korte skriftlige beskrivelser, samtidig som appen skal kunne vise den detaljerte infoen.

Appen har også en KI-basert maskot, *Mr. Praktisk*, som skal gi praktisk informasjon til brukeren med en snakkeboble (for eksempel “Bruk solkrem”, “Ta med en paraply”, “Det kan være glatt ute” osv.). Han skal i tillegg tilby sammendrag av været for dagen og uken. Det er også lagt til funksjonaliteter som lar brukeren tilpasse *VærSmart* etter egne preferanser, som å endre bakgrunn i appen og legge til egne hobbier som Mr. Praktisk tar hensyn til i rådene sine. Vi håper at dette skal gjøre det mer attraktivt for yngre brukere å sjekke været.

1.3 Hvem er vi?

Team 13 består av seks studenter som tar en bachelorgrad i informatikk. Tre av studentene tar studieretningen “design, bruk og interaksjon”, mens de tre andre går “digital økonomi og ledelse”.

CHRISTIAN  Alder: 26 Studieretning: Design, Bruk og Interaksjon Styrke: Overkomme tekniske utfordringer	MUNTAHA  Alder: 20 Studieretning: Digital økonomi og ledelse Styrke: Visuell kommunikasjon
ANDREAS  Alder: 23 Studieretning: Design, Bruk og Interaksjon Styrke: Flink til å programmere	SHANZA  Alder: 21 Studieretning: Digital økonomi og ledelse Styrke: Kommunikasjon
VIKTORIA  Alder: 32 Studieretning: Design, Bruk og Interaksjon Styrke: Visuell design, interaksjonsdesign	HANAN  Alder: 20 Studieretning: Digital økonomi og ledelse Styrke: Løsningsorientert

2. Brukerdokumentasjon

2.1 Innsiktssarbeid

Teamet har gjennomført et grundig innsiktssarbeid for å forstå sine brukere. For yngre brukere under 18 år holdt vi fysiske semistrukturerte intervjuer. Dette ble valgt fordi det er lettere å møte denne målgruppen personlig og innhente samtykke fra deres foresatte. For aldersgruppen 18 - 25 år, sendte vi ut en digital spørreundersøkelse. Vi utførte også fysiske intervjuer med to brukere på 22 og 24 år, som en del av en kvalitativ oppfølgingsmetode for å utdype innsikten og spesifisere brukerkrav. Lenken til spørreskjemaet og intervjuguide legges ved (vedlegg 1-2). Svarene fra brukerundersøkelsen indikerer at farevarsler, værinfo og praktiske tips er viktige for brukerne, mens daglige og langtidsvarsler sendt via push-meldinger var mindre prioritert. Brukerne ønsket praktiske tips for hvordan man skal kle seg etter været, men viste mindre interesse for værrelaterte nyheter (se Tabell 1).

Brukparameter	Resultat
Gjennomsnittsalder	22.8 år
Antall brukerne som (måles i % av totalt antall brukere):	
• sjekker værmeldinger hver dag	58%
• benytter værappen på Android/ iOS	90%
• har aldri brukt Yr-appen	54.8%
• Ønsker å få varsler om mulige katastrofer som flom, orkaner, skred, osv.	93.5%
Hvor viktig anser brukerne det å (måles på en skala fra 1 - 10):	
• få katastrofevarsler	snitt 7.94 / 10
• få daglige varsler om været	snitt 5.9 / 10
• få langtids værvarsler via applikasjonen	snitt 5.2 / 10
Prosent av brukere som ønsker følgende funksjoner (rangert fra mest ønsket til minst ønsket):	
• farevarsler	74.2%
• praktiske tips om hvordan man skal kle seg	61.3%
• daglige notifikasjoner om været	41.9%
• legge til flere sted som favoritter	38.7%
• værkart	22.6%
• værrelaterte nyheter	16.1%

Tabell 1. Resultater av brukerundersøkelsen.

For å presentere brukerne og deres behov har vi laget to personas (vedlegg 3). Personaene viser til en samlet beskrivelse av målgruppen og gir oss en bedre oppfatning av applikasjonens brukere. Basert på innsamlet brukerdata, har vi listet opp funksjonelle og ikke-funksjonelle krav til applikasjonen, se kravspesifikasjonen i vedlegg 4.

2.2 Applikasjonens funksjonalitet

Tabell 2. Oversikt over appens funksjonaliteter.

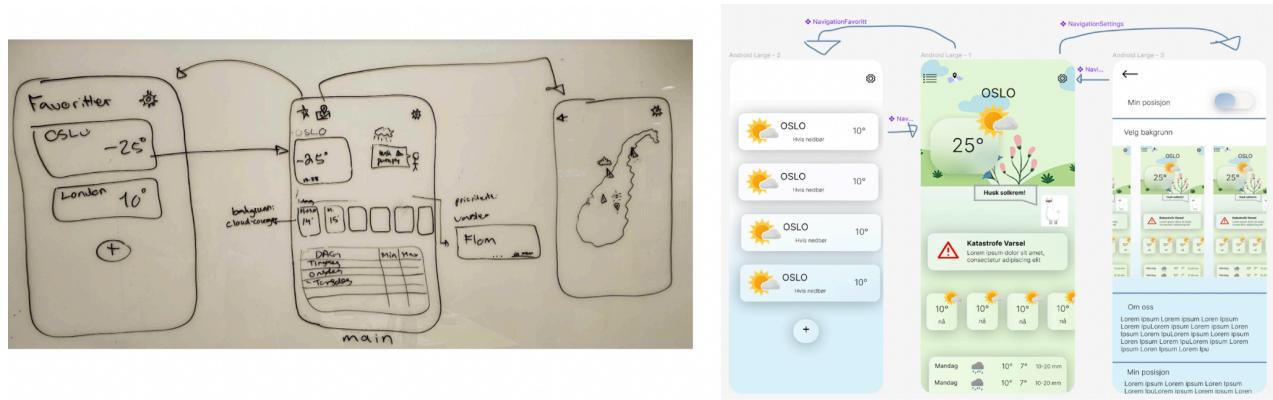
Skjerm	Funksjonaliteter
Hjem-skjerm	Vise været nå <ul style="list-style-type: none"> • Temperatur • Soloppgang / nedgang • Vær-ikon (overskyet, sol, regn etc.) • vind • Nedbør
	Vise været de neste 24 timene <ul style="list-style-type: none"> • Klokkeslett • Temperatur • Vær-ikon
	Vise været neste 7 dagene <ul style="list-style-type: none"> • Ukedag • Temperatur • Vær-ikon
	Mr. Praktisk: KI - Assistent <ul style="list-style-type: none"> • Gir praktiske råd om været akkurat nå, tar hensyn til alder, hobbyer og farevarsler i nærheten • Oppsummerer været de neste 24 timene • Oppsummerer været neste 7 dagene, tar hensyn til alder og foreslår hvilken dag(er) det passer best å drive med hobby
	Farevarsler <ul style="list-style-type: none"> • Vise de nærmeste farevarslene og beskrivelse av dem • Vise avstand til farevarslene • Knapp for å se varselet i et kart
Søke-skjerm	Søke etter steder i Norge
	Velge nåværende posisjon
	Få en oversikt over favoritter
Farevarsel-skjerm	Få en oversikt over alle farevarsler i landet <ul style="list-style-type: none"> • Som en liste eller, • i et kart
Innstillinger-skjerm	Sette alder
	Legge til hobbyer
	Velge bakgrunn

2.3 UI og struktur

2.3.1 Appens UI

Første utkast

Appens brukergrensesnitt og design ble utformet med utgangspunkt i resultatene fra datainnsamlingen beskrevet ovenfor i kapittel 2.1. Alle gruppemedlemmene skisserte hvert sitt forslag til appens utseende, og kom deretter i enighet om et felles design som tilfredsstiller både brukernes og teamets krav (vedlegg 5). Vi lagde deretter en *high-fidelity* skisse i Figma.



Figur 1: Initier skisse av appen vs Figma - skisse

Ferdige produkt

Skjermer og navigasjon

VærSmart har fire skjermer: Hjem, Søk, Farevarsler og Innstillinger. Navigasjonen skjer på bunnen av skermene, der hvert ikon representerer hovedfunksjonaliteten. Skermene ligger i en *horizontal pager*, som betyr at man også kan sveipe horisontalt mellom skermene for å navigere.

Hjem-skjerm

På hjem-skjermen vil man kunne se en rekke funksjonaliteter. Værinformasjonen på toppen av skjermen inkluderer lokasjon, når den sist ble oppdatert og høyeste og laveste temperatur for dagen. Det er også en bokmerkeknapp for å legge til eller fjerne en lokasjon fra favoritter, og en refresh knapp for å laste inn nyeste værdata. Videre kan man se en rad med været de neste 24 timene, og en tabell med være til uka. Man vil også kunne se farevarsler innenfor en

radius på 40 km. Dersom man trykker på kartikonet på farevarslet vil man kunne se et polygon av varslet i et kart. Det vil også være mulig å se nedbør, soloppgang / nedgang og vindhastighet og retning. KI-assistenten Mr. Praktisk dukker opp tre steder:

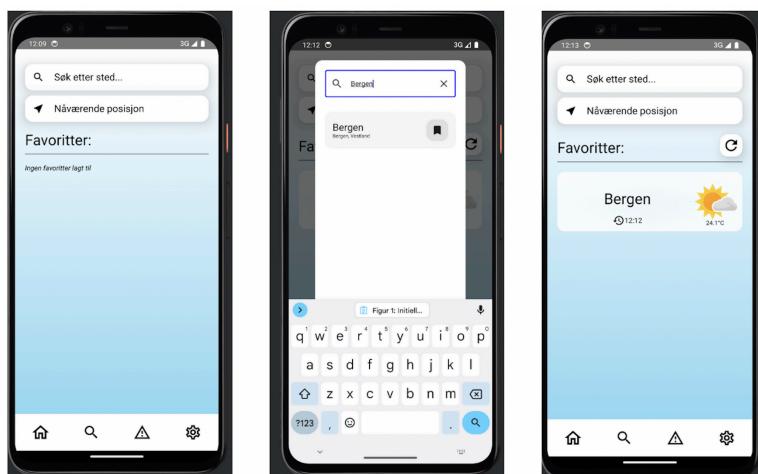
- Øverst for å gi informasjon og tips basert på dagens vær, hobbyer og alder.
- Under raden “været det neste døgnet” og gir en oppsummering for været de neste 24 timene.
- På bunnen av uketabellen for å gi en oversikt over været til uka og anbefale når det passer å drive med hobbyene.



Figur 2: Hjem-skjermen

Søke-skjerm

Søke-skjermen gir brukere muligheten til å velge stedet de ønsker å se været for. Man kan enten søke etter et nytt sted, eller velge den nåværende posisjonen til enheten. Søke-resultatene dukker opp som en knapp man kan trykke på for å navigere til hjemskjermen med det nye stedet man har valgt, sammen med en bokmerkeknapp for å lagre stedet i favoritter.



Figur 3: Søke-skjerm

Farevarsels-skjerm

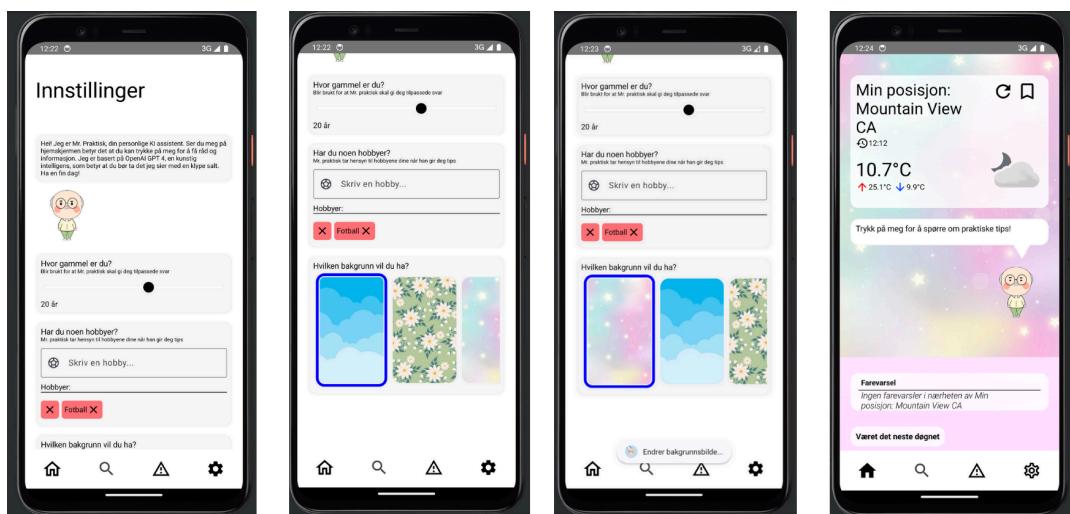
På farevarsels-skjermen får man opp en liste med alle farevarsler i landet, sammen med en knapp for å se dem i et kart.



Figur 4: Farevarsler

Innstillinger-skjerm

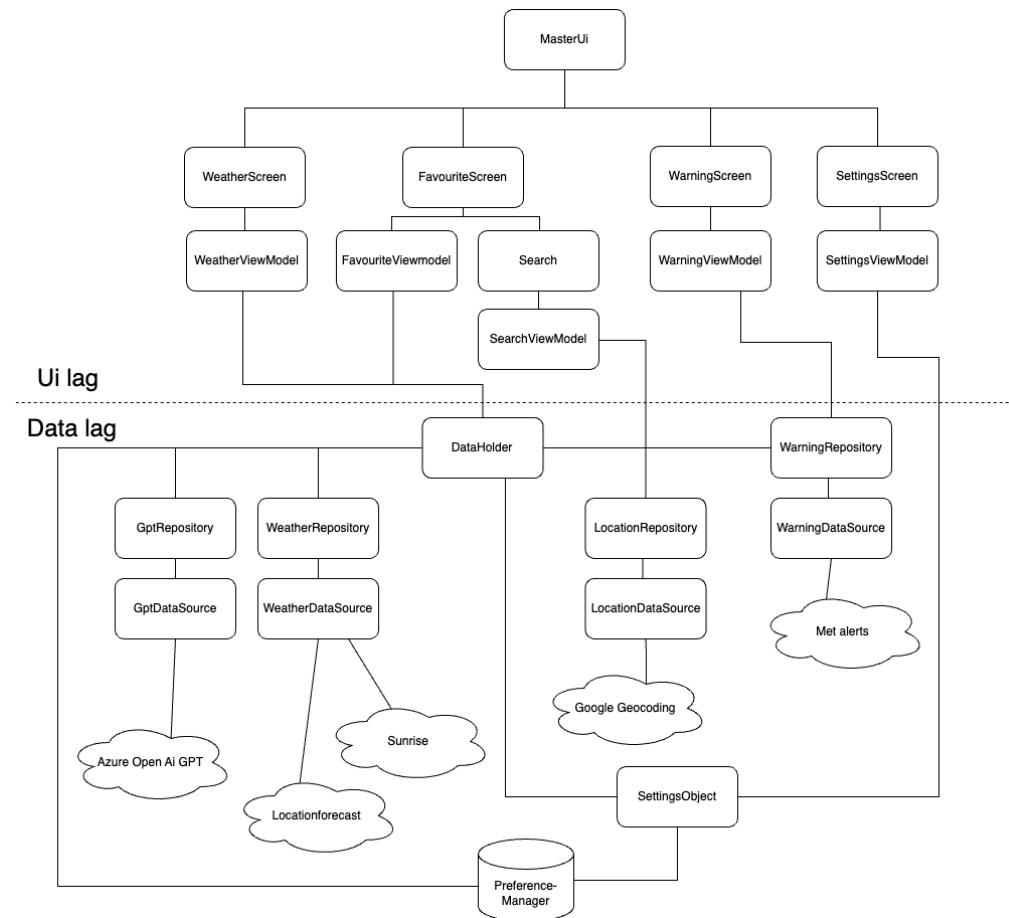
I innstillinger-skjermen kan man sette personlige preferanser for appen. Øverst får man info om Mr. Praktisk, som hvem han er og hva han gjør. Videre kan man sette alder og legge til hobbier for at Mr. Praktisk skal gi bedre og mer tilpassede svar. Man kan også velge ulike bakgrunner for appen.



Figur 5- Innstillinger

2.3.2 Appens arkitektur

Nedenfor er et klassehierarki for VærSmart illustrert.

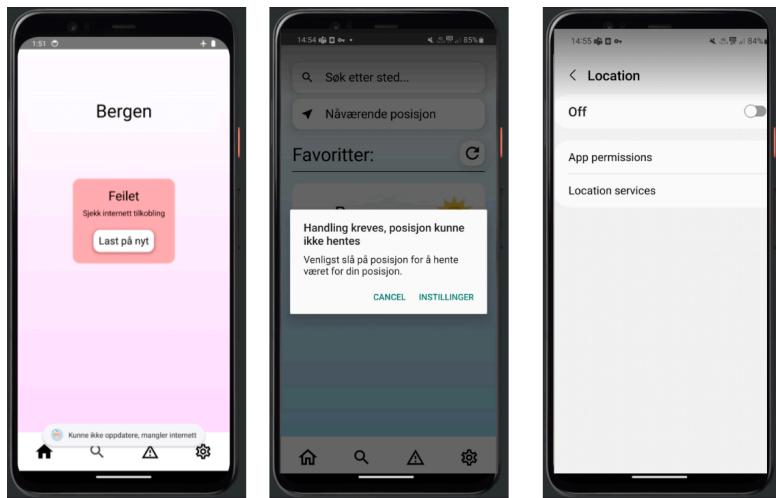


Mer om arkitektur og api-nivå står i ARCHITECTURE.md filen i git.

2.4 Tilgang til appen

VærSmart vil være tilgjengelig for Android enheter med Android 9 (Pie) eller nyere.

Internetttilgang er nødvendig for at appens viktigste funksjoner: hente værdata, farevarsler og lokasjoner, skal fungere. I de tilfellene brukeren ikke har internetttilgang, vil en feilmelding dukke opp, som vist i figur 6. For at appen skal kunne hente nåværende lokasjon, trenger den tilgang til enhetens GPS. Å oppgi denne informasjonen er frivillig og applikasjonen vil vise en standard lokasjon- Ålesund, eller den øverste i favorittlisten, dersom brukeren ikke gir tilgang til sin posisjon. Dette ble gjort fordi vi ikke ønsket at appen skulle være avhengig av å ha tilgang til nåværende posisjon. I det tilfellet hvor brukeren ikke har aktivert stedstjenester på mobilen og forsøker å hente nåværende posisjon, vil en beskjed dukke opp for å gi brukeren muligheten til å navigere til innstillingene og endre dette (se figur 6).



Figur 6- Feilmeldinger

3. Produktdokumentasjon

3.1 Kvalitetsegenskaper

3.1.1 Forskningsmetode – Brukerundersøkelser

For å identifisere kvalitetsegenskapene ved appen har vi tatt i bruk ulike brukerundersøkelser. Når applikasjonen var nærmest ferdigstilt, hadde vi en datainnsamling for å kartlegge manglende funksjonalitet eller ønsker hos målgruppen. Innsamling skjedde gjennom geriljatesting av fem brukere i målgruppen. Et slik representativt utvalg danner et relativt godt perspektiv av ønsker og behov hos målgruppen. Vi ga også en bruker tilgang til appen over en lengre periode på to uker, slik at vi kunne få et mer realistisk inntrykk av hvordan appen er å bruke i hverdagen (se vedlegg 9). Imidlertid er det verdt å bemerke seg at seks undersøkelser ikke er særlig mye, noe som svekker tilliten til resultatene.

Basert på resultatene fra testene, ferdigstilte vi applikasjonen og kjørte en ny datainnsamling, men denne gangen basert på brukertesting. Vi testet appen på syv tilfeldige intervjuobjekter og sendte ut samtykkeskjemaer til dem. I gjennomføringen av testene, ba vi brukerne først utføre noen instrukser og deretter hadde vi et nettskjema for å samle tilbakemeldingene fra brukerne. Av rettslige grunner var disse brukerne over 18 år. Likevel med tanke på oppgavebeskrivelsen, bestemte vi oss for å prøve å komme i kontakt med flere yngre. For å sikre foreldressamtykke var disse brukerne enten familiemedlemmer, eller venner av

teammedlemmene. Tilbakemeldingene fra undersøkelsen kan, som en følge av den nære relasjonen, preges av mangel på objektivitet. Resultatene fra geriljatesting, samtykkeskjemaet og brukertesting kan ses i vedlegg 6-8.

3.1.2 Evaluering av egenskaper

Gjennom de overnevnte brukerundersøkelsene har vi avklart appens kvalitetsegenskaper. Fra resultatene til brukertesting i vedlegg 8, kommer det frem at nærmere 75% har rangert appen som svært intuitiv. I tillegg påstår 85% av deltakerne at det er enkelt å finne frem til nødvendig data, og 70% svarer at appens design og layout er tiltalende. Dessuten oppgir hver kandidat i geriljatestingen at appen er enkel å bruke. Derfor vil brukervennlighet regnes som en av de viktigste kvalitetsegenskapene til appen.

I brukertesting oppgir også majoriteten av deltakerne at appen ikke krasjer eller har treg respons, dette gir en indikasjon på høy ytelse. Vi har gjennom hele prosjektet prøvd å få appen til å krasje ved fuzztesting og korrigert eventuelle feil som oppsto. Det har gjort *VærSmart* til en pålitelig og robust applikasjon, noe vi anser som en viktig kvalitetsegenskap.

En annen viktig kvalitetsegenskap ved *VærSmart* er dens brede spektrum av implementerte funksjonaliteter. En styrke er at appen dekker mange brukerbehov identifisert i innsiktsarbeidet. Sammenligner man kravspesifikasjonen med den ferdige appen, kan man se at alle de funksjonelle- og ikke-funksjonelle kravene er møtt.

3.1.3 Svakheter i applikasjonen

Ved testing har vi oppdaget følgende svakheter i applikasjonen:

- *Laveste og høyeste temperaturer.* Den viser riktig, men api-et brukt for å hente vær finner ikke vær i fortiden. Når klokken er 23.00 finner den kun én temperatur den dagen (siden det kun er én time igjen av dagen) og høyeste og laveste temperatur er da lik. Det kan være noe misvisende/forvirrende.
- *Avstand til farevarsler.* I hjem-skjermen vises kun farevarsler som er nærmere enn 40 km unna. Farevarsler som er lengre unna tenkte vi at ikke er relevant. Forskjellige brukere kan være uenig, så avstanden burde vært en innstilling som brukeren kan endre.

- *Soloppgang / nedgang api.* Dersom man ser været for et sted der det er midnattssol (for eksempel nordkapp), vil sunrise API-et gi “null” ved tidsvariablene for soloppgang og nedgang. UI-et håndterer dette ved å bare ikke vise soloppgang / nedgang. UI-et ser helt naturlig ut, men det ville kanskje vært bedre om det ga en feilmelding i stedet for å ikke vise noe som helst.
- *Personvern for KI.* Brukeren må samtykke til å dele alder og hobby til OpenAI, siden dette blir brukt i promptene. Brukeren kan velge å ikke oppgi noen hobbier, men alderen blir satt med en *slider* med 13 som standard. Det er ikke mulig for brukeren å ikke oppgi alder. Etisk bruk av KI er diskutert senere i rapporten.
- *Tilbakeknapp.* Tilbakeknappen i Android er ikke tatt hensyn til ved utviklingen av denne appen. Dersom man åpner dialoger (som blir brukt til søker og kart) vil tilbakeknappen lukke dialogene, ellers lukker den bare appen. Ved videreutvikling av appen ville vi gitt tilbakeknappen mer relevant funksjonalitet.
- *Mr. Praktisk animasjon.* Mr. Praktisk bruker Lottie Animation for å være dynamisk. Lottie-animasjoner er veldig lange JSON-filer som leses av enheten veldig fort for å oppdatere UI-et. Vi oppdaget senere at dette krever veldig mye GPU og CPU kraft. På Android-enhetene vi testet på, var heldigvis appen fortsatt veldig responsiv og “snappy”, men det vil kanskje oppstå problemer på eldre og billigere telefoner med svakere CPU og GPU.

Flere svakheter/overveininger angående ytelse og kravspesifikasjoner er diskutert andre steder i rapporten.

3.2 API-er

3.2.1 API-nøkler

Noen av API-ene vi bruker krever betalte API-nøkler. Ved “best practice” bruk av API-nøkler i Android applikasjoner, burde nøkler ikke bli lagret som klartekst. I det minste burde nøklene ligge i noe som *local.properties* eller i en *secrets.properties* og bli hentet ut ved bruk av en Secrets Gradle plugin, slik at de er filtrert i *.gitignore* fra å bli delt offentlig. I tillegg til at det blir vanskelig å hente ut nøklene ved “key extraction” (“Android Keystore system,” 2024). Sjefsgruppelærer har bekreftet for oss at fordi dette prosjektet er kortvarig, og det er viktigere at vi er sikre på at sensor enkelt kan laste ned og kjøre appen uten problemer, ser vi

det som akseptabelt at nøklene for vårt private prosjekt og repository ligger lagret som klartekst.

3.2.2 API-nivå

Android

Vi har valgt minimum SDK API-nivå 28 med target SDK API-nivå 34. I starten av prosjektet ønsket vi en balanse mellom å være kompatibel med mange enheter og å kunne benytte moderne Android funksjonalitet. Vi vektla mest sistnevnte fordi vi ønsket en nyere versjon av Android, og ikke være begrenset av mangel på funksjoner. Samtidig ønsket vi en stabil versjon, og de aller nyeste kan være litt upålitelige fordi de ikke har blitt testet i like stor grad. API-nivå 28 føltes derfor som et trygt valg. Det kan benyttes av 90,3% av Android-enheter og tilsvarer Android 9 (Pie) som ble lansert i 2018 (“Android API Levels,” 2024). Det møter også Google Play sine minimumskrav til API-nivå for app-oppdateringer. Vi har i etterkant reflektert over om vi kunne ha valgt en eldre Android-versjon som Android 7.0 for å tillate enda flere brukere å bruke appen.

3.2.3 Beskrivelse av API-er

Innhenting av værdata

Sunrise and Sunset

For å hente klokkeslett for soloppgangen og solnedgangen for en gitt posisjon på en gitt dag bruker vi Sunrise 3.0 API-et fra MET Norway, som hentes gjennom proxyen til UIO (Sunrise, 2022). Vi har valgt å inkludere dette API-et fordi det var en type praktisk informasjon vår målgruppe viste interesse for. API-et tar inn to koordinater og en dato og returnerer et *SunriseAndSunset*-objekt som vi kan bruke til å vise klokkeslettene for soloppgangen og solnedgangen for en gitt dag.

MetAlerts

For å vise farevarsler i Norge bruker vi MET Norway sitt MetAlerts API (Metalerts, 2017). API-et gir oss en liste med alle farevarsler, hvor hvert farevarsel inneholder informasjon som koordinater for hvilke områder det gjelder, tidsperiode, alvorlighetsgrad, tekstlige beskrivelser og råd, fargekoder og konsekvenser. Vi bruker informasjonen til å vise brukerne

varslene i nærheten av den valgte posisjonen på hjem-skjermen, og hele listen på farevarsel-skjermen. På begge skjermene kan man interagere med varslene for å enten lese mer tekstlig informasjon, eller få de presentert som polygoner på et kart som brukeren kan interagere med. Tidlig i fasen hadde vi vanskeligheter med å finne hvilken informasjon som burde hentes ut fra API-et som var lesbart for brukeren. Dette var grunnet at API-et ikke alltid kommer i samme format. For eksempel kommer noen ganger “område” med tekstlig beskrivelse og dato, og andre ganger kun stedsnavn.

Location Forecast

Den generelle informasjonen om været for en gitt lokasjon er hentet fra MET Norway sitt Locationforecast API (Locationforecast, 2022). Vi bruker dette API-et til å hente data om været som pågår, kommer de neste 24 timene, og til uken for en gitt lokasjon. En utfordring vi hadde ved implementasjonen av Locationforecast, var at dersom vi ønsket å teste med forhåndsbestemte lokasjoner og tidspunkter, ville API-et ikke returnere data dersom tidspunktet var eldre enn det nåværende tidspunktet.

Kart

Kartet som viser farevarsler er hentet fra et Google Maps SDK API (Google Maps Platform Documentation, 2024). Vi har valgt dette fordi Jetpack Compose tilbyr brukervennlige innebygde metoder og funksjoner for Google Maps-kartet, som kommer som en egen composable. For eksempel tegner vi polygon-objekter på kartet for å representer koordinater med farevarsler fra MetAlerts som fargelagte områder på kartet.

Lokasjons-søk

Vi har kommet fram til at Google sine REST-versjoner av Geocoding- og Reverse Geocoding API-er passer best til våre behov. Vi testet også andre, men synes denne ga best svar basert på søketeksten. Med Google Geocoding API kan vi ta inn en tekst fra brukeren og velge hva slags type resultater API-et skal returnere i JSON-format, som for eksempel kun gatenavn og koordinater fra et spesifikt land. Koordinatene blir videre brukt til API-ene for værdata, slik at vi kan presentere værdata for lokasjonen som blir søkt etter. API-et støtter også automatiske forslag for stedsnavn for hver bokstav brukeren taster, som gjør appen mer brukervennlig. Dersom brukeren gir appen tilgang på deres posisjon, sendes koordinatene til

Reverse Geocoding API-et som returnerer enten navnet på byen eller den overordnede kommunen for hvor brukeren befinner seg.

Kunstig Intelligens

For Mr. Praktisk har vi benyttet OpenAI sin GPT-4 REST-API-modell for *Chat Completions* fra Microsoft Azure OpenAI. Det er kort sagt en prompt som gir et tekstsvar, typ Chat GPT, men som et api kall. Vi valgte denne KI-tjenesten fordi det var det mest ideelle produktet for våre behov. Vi har jobbet mye med prompt engineering og optimalisert bruk av antall tokens, framfor å pøse på med så mye data som mulig og risikere et upresist svar. Når metrikker som værdata, preferanser og alder sendes sammen med vår API-nøkkel, får vi tilbake et tekstsvar i et JSON-format som kan hentes ut og presenteres for brukeren på grensesnittet.

Etisk bruk av KI

Ved bruken av kunstig intelligens oppstår det etiske spørsmål og problemstillinger vi har måttet ta høyde for. Vi har tatt utgangspunkt i retningslinjer utarbeidet av EU-kommisjonen, for å sørge for at vi bruker kunstig intelligens på en måte som er lovlig, sikker og etisk. For det første betyr bruken av skybasert KI at data som vi henter i appen om brukeren, som interesser, alder og posisjoner kan sendes inn og bli knyttet til deres IP-adresse dersom de velger å generere meldinger med Mr. Praktisk. En potensiell løsning her ville ha vært å anonymisere IP-adressen til brukeren ved å gjøre API-kallet gjennom vår egen proxy, noe som vi hadde gjort dersom appen skulle lanseres for offentligheten. Likevel i henhold til prinsipp én i retningslinjene har vi holdt brukerne "in the loop", ved å eksplisitt spørre om tillatelse for posisjon og gjort deling av hobbyer og alder frivillig (Regjeringen, 2020). Dessuten vil OpenAI behandle personopplysningene i henhold til GDPR-loven som sikrer at dataene som lagres, etterlever prinsippene om lovlighet, rettferdighet og gjennomsiktighet ("Security & privacy," 2024).

Til tross for mulige sikkerhetstrusler knyttet til lagring av persondata, har vi konkludert at nyttigheten og fordelene ved bruken av KI veier mer enn ulempene. Implementasjonen er et viktig "proof of concept" for å utforske hvordan KI kan brukes til å nå flere målgrupper med tilpasset informasjon om været. Uten bruk av kunstig intelligens ville det vært nødvendig å bruke flere titalls timer på å skrive mulige meldingskombinasjoner basert på nedbørsmengder, og en omfattende ordbok med alle mulige hobbyer.

3.3 Universell utforming

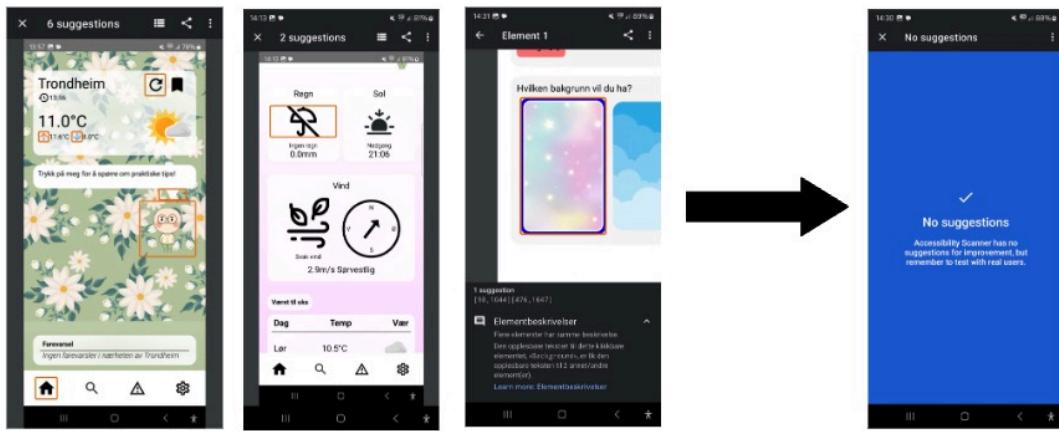
Under apputviklingen fulgte vi anbefalte prinsipper for universell utforming, Web Content Accessibility Guidelines 2.1 (WCAG 2.1, 2023; Universell utforming og tilgjengelighet, 2024). Vi bruker også Jetpack Compose sine tilgjengelighetsverktøy for å gi knapper og bilder beskrivelser (“Accessibility in jetpack compose,” 2024). Appen er designet for å være tilgjengelig og brukbar for barn fra 13 år og oppover uavhengig av deres evnenivå.

For å tilfredsstille UU krav har vi blant annet gjort følgende:

- Bruk av vanskelige ord og lange setninger unngås.
- Informasjon fremstilt i appen er lett synlig og forståelig.
- Farevarsler markeres med enten gult, oransje eller rødt for å indikere alvorsgraden.
- Mr. Praktisk gir skriftlig værbeskrivelse, som vil fungere med TalkBack, noe som øker forståelsen for personer med kognitive utfordringer og synshemninger, og tilfredsstiller suksesskriteriet 1.4.1 i WCAG 2.1.
- Brukeren kan velge mellom ulike bakgrunnstemaer som passer best for brukerens synsbehov og bidrar til personlig tilpasset brukergrensesnitt.
- Klare kontraster passer godt for personer med nedsatt syn og kognitive utfordringer. Kontrastforholdet er på 3:1 for komponenter for brukergrensesnitt, og det tilsvarer suksesskriteriet 1.4.11 i WCAG 2.1.
- Utformingen er forståelig og intuitiv å bruke. Man kan se farevarsler på et kart, som hjelper brukeren med orientering 2.4.8 i WCAG 2.1. Tydelige tilbakemeldinger som toast-meldinger tilfredsstiller *feedback* og *affordance* designprinsipper.
- Appen krever minimal fysisk anstrengelse fra brukere. Vanlig sveiping på berøringsskjerm gjør den komfortabel for langvarig bruk.
- Bilder og knapper har tekstlige beskrivelser.

For å teste appens universelle utforming benyttet vi oss av Google sin Accessibility Scanner (Google LLC, 2023). Den skannet appen og ga tilbakemelding på feil og mangler. Det var litt kontrast-feil og manglende beskrivelser på bilder og knapper, noe som var enkelt å løse. Den ferdige appen hadde noen få anbefalinger fra skanneren, men ingen som ville vært et problem

ved reelt bruk. Ved videre utvikling av appen, ville vi også testet appen med faktiske brukere med ulike funksjonsnedsettelser, ikke bare bruke programvare.

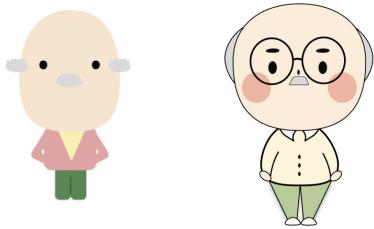


Figur 7: Google LLC accessibility tool

Vi testet også å bruke appen med en skjermleser og Talkback, noe som fungerte greit. Men dersom vi skulle lansert appen, ville vi satt av tid til å teste dette mer nøye med svaksynte eller blinde, ettersom ingen i teamet har en reell forståelse av hvordan det er å være svaksynt og bruke skjermlesere. Applikasjonen er kompatibel med eksterne apper for syns- og hørselshemmede, og har rom for ytterligere oppgraderinger slik at den blir enda mer universelt utformet.

3.4 Mr. Praktisk avatar og animasjon

For å kunne fremstille de praktiske tipsene ønsket vi å ha en avatar. Vi tenkte dessuten at denne avataren skulle ha forskjellige animasjoner når den “snakker” og gir tips, og når den kun er stillestående. I designprosessen av avataren startet vi med å komme med forslag på hvordan avataren kunne se ut. Vi ønsket at den skulle være kjærlig og hyggelig, slik at den kunne appellere til den vide målgruppen, og samtidig fremstå klok siden den gir tips. Valget falt derfor på at Mr.Praktisk skulle være en eldre mann. I Figma skisserte vi ut hvilke elementer vi ville ha med, før vi kom frem til den endelige versjonen av Mr.Praktisk. Figur 8 under viser første skisse av Mr.Praktisk opp mot den endelige versjonen.



Figur 8: Skisser av Mr.Praktisk fra Figma

Da vi hadde en versjon av designet vi var fornøyde med, brukte vi LottieLab for å lage selve animasjonen. Lottie tilbyr direkte integrasjon med Figma og enkle verktøy for å kunne lage ønskede animasjoner. Vi fikk på denne måten blunke-animasjon og snakke-animasjon. For å overføre den til Android Studio brukte vi konverteringsverktøyet Lottie har, for å formatere det til en JSON-fil. Kvaliteten på filen er dårlig, fordi bedre kvalitet krever et betalt Lottie-abonnement.

4. Prosessdokumentasjon

4.1 Arbeidsmetodikk

Vi benyttet den smidige metodikken kalt Scrumban, en hybrid av Scrum og Kanban (Atlassian, 2024). Vi brukte elementer fra Scrum som sprinter, retrospektiv, stand-ups og sprintmøter. Fra Kanban brukte vi elementer som Kanban board, definerte retningslinjer, kontinuerlig forbedring og fleksibilitet i arbeidsfordeling og tidsbruk. Dette optimaliserte arbeidsflyten og hjalp med å administrerte arbeidsoppgaver, samt sikre kontinuerlig fremgang og fleksibilitet under applikasjonsutviklingen og unngå flaskehalsar.

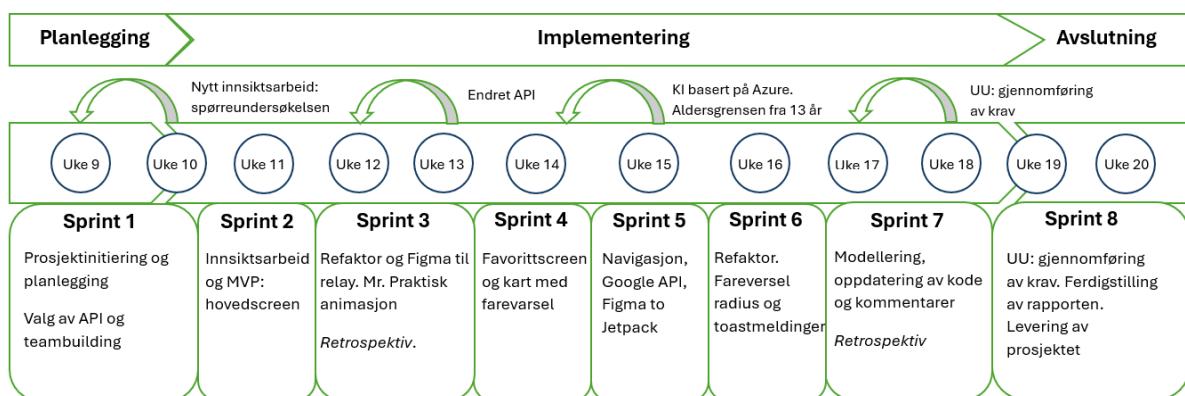
Teamet jobbet i sprinter på 1-2 uker med daglige sprintmøter, hvor hver sprint begynte med sprint planlegging, og avsluttet med sprint review. Vi hadde to sprint retrospektiver for å evaluere oppnådde resultater, finne ut problemer og identifisere forbedringsområder (se Prosesslogg, ss. 10-12; 16-17). Arbeidsoppgaver ble satt i en backlog, og ble i hvert planleggingsmøte flyttet over i *to-do*. Dette er også kjent som Kanban board, hvor vi illustrerer arbeidsoppgaver i en backlog framfor “ready to start”. Vårt board inkluderer en "til godkjenning" kolonne som representerer pull requests eller andre arbeidsoppgaver som må godkjennes av andre teammedlemmer før det anses som ferdig (se vedlegg 11). Vi benyttet oss av en “trunk based” branchingsstrategi. Vi lagde nye brancher for hver ny funksjonalitet,

og merget dem til main så fort det fungerte bra nok til å bli implementert i appen, og et annet teammedlem hadde sett over og godkjent.

Vi programmerte stort sett individuelt og parallelt med hverandre, der hvert medlem fokuserte på hver sin funksjonalitet. For de mer krevende oppgavene, drev vi ofte med par-programmering. Vi prøvde også mobb programmering noen ganger, men synes ikke dette var særlig effektivt. Vi hadde ikke fast Scrum Master og byttet på rollene slik at hvert teammedlem fikk tverrfaglig erfaring med software development og bruk av smidige metodikker. Vi arbeidet smidig og med lite byråkratiske prosedyrer for å effektivisere arbeidet.

4.2 Prosjektets gang

Prosjektet tok totalt åtte sprinter. Som regel møttes teamet fire timer, tre ganger i uken. Prosjektets implementering er presentert som et Gantt diagram (se vedlegg 10). I grove trekk kan prosjektets gang bli delt inn i tre faser: oppstart / planlegging, implementering og avslutning (se figur 9).



Figur 9: Prosjektets gang fordelt i sprinter.

Fase 1: Oppstartsfasen (Uke 9-10)

Gjennom oppstartsfasen bestemte vi lengden på sprinter, fordele roller for nåværende sprint og lagde et Kanban board for backlog-oppgaver. Vi lagde også en overordnet milepælsplan for å få et overblikk over ønsket fremgang og implementeringsprosess (se vedlegg 12).

Oppstartsfasen kan bli karakterisert som prosjektplanlegging og fordeling av produktbacklog. Teammedlemmene hadde blitt enige om at prosjektplanen skulle oppdateres basert på resultater fra innsiktarbeid og brukertesting, siden brukersamarbeid og svar på endringer er

viktigere og smidigere enn å følge en bestemt plan. Vi hadde også bestemt at prosjektutviklingen blir inkrementell og iterativ, noe som ble tatt i bruk med en gang i løpet av den første sprinten. Det første forsøket på et innsiktsintervju mislyktes, og teamet måtte revidere og starte over med et nytt spørreskjema for brukerundersøkelsen. For å forstå behovene til brukerne enda bedre og kartlegge spesifikke behov hos brukeren, gjennomførte vi flere dybdeintervjuer. Den første sprinten initialiserte oppstartsfasen, og teamet begynte å implementere løsningen samtidig med planleggingen allerede i uke 10. Derfor regnes denne uken som en del av både oppstarts- og implementeringsfasen.

Fase 2: Implementering (Uke 10-18)

Prosjektets implementering består av en serie med sprinter på én til to uker med noen enkelte oppgaver som varte i flere sprinter, for eksempel animasjon av Mr. Praktisk og Maps-API-et. I denne fasen siktet vi mot en teknisk avansert og moderne løsning, som er brukervennlig og følger UU prinsipper. Produktutviklingen var iterativ og vi drev ofte med brukerdrevet produktutvikling. Vi gjorde dette ved å revidere kravspesifikasjoner flere ganger basert på brukernes tilbakemeldinger. Når vi oppdaget feil og mangler gikk vi tilbake og endret, og gjorde endringer basert på tester vi gjorde selv.

Vi oppdaget blant annet at lokasjons api-et vi brukte ga til tider dårlige svar, så vi endret til Google sitt geocoding api. Vi prøvde i sprint to å bruke relay for å overføre figma filer til Jetpack Compose. Dette fungerte dårlig, så vi begynte på nytt og skrev Jetpack Compose koden fra scratch med figma filene som en guide. Etter evalueringer har vi flere ganger i løpet av implementeringsfasen gjort endringer til UI-et, så det ferdige produktet er ikke identisk med de første figma skissene.

Vi arbeidet smidig med implementering av ny funksjonalitet. Dersom noen hadde en god ide var det ingenting i veien for å lage en ny git branch og jobbe autonomt. Det gjorde arbeidet mer tidseffektivt og motiverende, ettersom at man ikke trengte å vente på neste møte før man satte igang med noe nytt. For å unngå forvirring holdt vi hverandre oppdaterte i gruppechat når nye ideer dukket opp og ble utforsket. Alle gruppemedlemmene opplevde denne smidigheten som svært positiv, fordi ideer kunne prøves mens de var ferske i hodet, og det gjorde det lettere å nå tidsfrister. Et eksempel var i sprint fire, da vi kom på ideen om å bruke KI. Ideen ble presentert i gruppechatten, og en prototype ble utviklet frem til neste møte. På

møtet ble det vurdert om det skulle implementeres. Samtidig var det til tider vanskelig å implementere nye ting på denne måten, som ved endringer i UI.

I sprint seks endret vi målgruppen fra 9-25 år til 13-25 år grunnet bruk av KI. Teamet styrte også prosjektets økonomiske kostnader og søkte om støtte fra UiO til KI. Vi fikk en Azure brukerkonto med tilgang til OpenAI. API-kall gjennom Azure gjør API-et noe tregere og mer upålitelig. Vi valgte alikevel å benytte oss av Azure grunnet kostnader og tilgang til GPT-4.

Fase 3: Avslutning og levering (Uke 19-20)

Applikasjonen var nesten ferdig allerede i uke 17, og vi begynte å avrunde utviklingen med modellering, små endringer og kommentering av kode. Teamet holdt et retrospektivt møte, hvor det var bestemt å sjekke om applikasjonen fullførte UU-krav, teste den for accessibility og ferdigstille rapporten. Etter små justeringer i koden ble appen ferdig for levering. Teamet jobbet videre med prosessdokumentasjon og produktrapport, og leverte både produktet og følgende dokumentasjon på Devilry på planlagt dato.

4.3 Endringer i kravspesifikasjonen

Kravspesifikasjonen er delt inn i funksjonelle- og ikke-funksjonelle krav. Den fullstendige oversikten vil ligge i vedlegg 4. Disse kravene er formulert med tanke på oppgavebeskrivelsen, tilbakemelding og ønskede funksjonaliteter. Tabellen viser til alle funksjonene som er tilgjengelig i applikasjonen.

Navigering og layout

I de første skissene (se figur 1) hadde vi en top app bar for å navigere mellom skjermene. Etter å ha laget funksjonelle prototyper og testet med brukere dukket det opp utfordringer og nye ideer som ikke var like åpenbare fra de opprinnelige skissene. Vi fant blant annet ut av at appens navigering og layout trengte noen endringer. Vi byttet til å ha navigeringen på bunnen av skjermen, i tillegg til mulighet for å sveipe mellom skjermene. Dette gjorde det mye mindre anstrengende å navigere, noe som gjorde appen mer brukervennlig. Samtidig tilfredsstiller det i høyere grad det tredje prinsippet for universell utforming om enkel og intuitiv bruk, og det sjette prinsippet om lav fysisk anstrengelse (University at Buffalo, 2024). Vi endret også favoritt-skjerm til å bli en søker-skjerm. Den fikk ikke noe ekstra funksjonalitet, men både vi og brukere synes det ga mer mening å kalle dette for en

søke-skjerm siden den primære funksjonaliteten var å finne nye steder. Kart-skjerm valgte vi å ikke inkludere, men heller gi mulighet for å se farevarslene på et kart som en dialog i stedet for en hel skjerm. Siden hjemskjermen filtrerer ut farevarslene som er lengre enn 40 km unna den gitte posisjon, valgte vi å gi farevarslene en egen skjerm slik at brukeren også kan få en oversikt over farevarslene i hele landet.

Kunstig intelligens

Etter de første skissene la vi merke til at appen ikke skilte seg noe særlig fra andre vær apper. Et av målene var å nå ut til yngre brukere, så vi tenkte å implementere en avatar / maskot - Mr. Praktisk, som skulle gi praktiske råd basert på været, noe som for yngre brukere er lettere å forstå enn komplisert værdata. I starten var planen å kode disse praktiske tipsene på egen hånd, men før vi rakk å begynne på koden kom ideen om å bruke KI i stedet. Dette synes vi virket veldig spennende, og noe vi ønsket å implementere i appen. Da vi begynte å teste KI API brukte vi en privat OpenAI brukerkonto til det ene medlemmet for å skaffe en API-nøkkel. GPT-4 var ganske kostbart, så vi brukte GPT-3.5 Turbo. Vi søkte om støtte fra UiO, og fikk en Azure brukerkonto vi kunne bruke til OpenAI GPT. UiO sponsorer da bruk av API-et, og vi oppgraderte derfor til GPT-4. I den ferdigstilte appen anser vi KI som en av de viktigste funksjonalitetene.

Ønsker fra brukere

I brukerundersøkelsene av en hi-fi prototype ble intervjuobjektene spurta om de savnet noen funksjonaliteter i appen (se vedlegg 7 og 8). Følgende ønsker ble avdekket:

- *Nordlysvarslet*: Det synes vi hadde vært kult å implementere, men ikke noe vi valgte å prioritere ved utvikling av appen. Funksjonen kan bli implementert ved videreutvikling.
- *Vindretning*: Dette ønsket kom fra en bruker som var interessert i golf. I prototypen som ble testet i brukerundersøkelsen viste appen kun vindstyrke, og ikke retning. Å få data om vindretning krevde ikke et nytt API, så dette var ganske enkelt å greit å implementere. I den ferdige appen vises vindretningen i et kompass. Den samme brukeren ønsket også at kompasset skulle roteres relativt til retningen mobilen peker. Dette var heller ikke noe vi valgte å prioritere, men noe vi hadde implementert ved videre utvikling av appen.
- *Soloppgang og nedgang*: Dette krevde et nytt API, men det var ikke komplisert å implementere. I den ferdige appen vises klokkeslett for soloppgang / nedgang sammen med et ikon.

- *Mr. Praktisk:* Tidligere var Mr. Praktisk kun tilstede ved nåværende vær og de neste 24 timene. Flere brukere ønsket også at han skulle gi tips til uka og hvilken dag det passer best å drive med hobbyene. Denne funksjonaliteten er implementert i den ferdige appen. Ved brukertesting ga også Mr. Praktisk litt rare svar (for eksempel å spille golf innendørs eller ta en skitur i Oslo i April), så vi satte av tid til å drive med prompt engineering.

4.4 Testing

Metodikk

Allerede fra første runde med implementasjoner av funksjoner og klasser, har testing gjennom versjonskontroll vært helt essensielt. Med utgangspunkt i de syv prinsippene for testing, forsøkte vi å unngå teknisk gjeld (Soleng & Vihovde, 2022, s. 13-17). Selv om vi ikke kan fange opp alle feil og problemer i koden ved testing, betyddet likevel testing at vi totalt sett kunne spare tid ved å finne typiske feil som samler seg i klynger. Vi gjorde testene kontinuerlig etterhvert som funksjonaliteter ble implementert i sprintene. Vi forsøkte for hver sprint å følge V-modellen (Soleng & Vihovde, 2022, s. 30). Hver gang ny funksjonalitet ble utviklet startet vi med å samle krav og bevegde oss nedover V-modellen til modellering. Mens vi kodet bevegde vi oss oppover igjen. Etterhvert som koden ble mer implemeterbar, testet vi på høyere nivå i V-modellen.

Enhetstesting

Noen funksjonaliteter måtte passere enhetstester før vi kunne gå videre på integrasjonstesting, for å unngå teknisk gjeld. Det ble laget 10 tester med JUnit versjon 4.13.2. Testene består av; fem tester for å sammenligne datoer og klokkeslett, som er helt nødvendig at fungerer som forventet, og fem tester for å teste koordinater, polygoner og avstander. Dette er funksjoner som må fungere for at været skal gis på riktig tidspunkt, og at avstand til farevarsler skal være riktig.

Integrasjonstesting

Etterhvert som kode ble mer implemeterbar, bevegde vi oss videre til integrasjonstesting, for å se om de ulike klassene og metodene av en funksjonalitet fungerte sammen som én komponent. Vi lagde klasser for alle funksjonalitetene som API-kallene skulle kunne gjennomføre, som for eksempel å hente været fra en gitt posisjon og tidspunkt, eller hente

adresser og farevarsler. Dette var hvit boks-testing som en strukturbasert testteknikk, fordi disse klassene skal kunne kjøre uavhengig av hverandre, samtidig som at de skal kunne vise oss hvordan data blir overført mellom ulike klasser og interface-repositories (Soleng & Vihovde, 2022, s. 101). Siden det andre prinsippet for testing tilsier at fullstendig testing er umulig, ville vi i hvert fall være sikre på at fundamentet for appen fungerte som forventet.

Systemtesting

Etterhvert som ny funksjonalitet kunne bli implementert i appen, gikk vi videre til systemtesting. Vi testet om den nye funksjonaliteten fungerte sammen med resten av applikasjonen, og at grensesnittet tillater bruk av funksjonaliteten. Appen har blitt testet grundig av oss som utviklere, og i samarbeid med brukere for å se om systemet fungerer ende til ende. Som utviklere har vi gjort utforskende testing hvor vi har forsøkt å “knekke” systemet ved å endre på tidssoner og andre innstillinger på Android-emulatorene. På dette stadiet testet og evaluerte vi også om prinsippet om lav kobling og høy kohesjon ble tilfredsstilt. Det var fra systemtestingen at vi lærte den største andelen kunnskaper og perspektiver over hvilke svakheter vi hadde ved appen, som for eksempel at vi måtte endre på forespørsels-begrensningen for Google API-kall, da vi hadde mange brukere samtidig som testet appen. Denne begrensningen var satt for å beskytte oss selv fra unødvendige kostnader, men maksgrensen var satt for lav til at så mange kunne bruke appen samtidig

Akseptansetesting

Det er i akseptansetestingen sitt nivå at vi tester i hvilken grad appen dekker behovene til brukeren, gitt at appen skal være klar for eventuell lansering. Vi gjennomførte ikke akseptansetester for hver funksjonalitet, kun de viktigste. Målet med akseptansetesting var blant annet å teste nytteverdien og brukervennligheten av appen. Vi så ikke et behov for dette hver eneste sprint, spesielt ikke før appen nærmet seg et ferdig produkt. Ved evaluering med brukere som en del av den utforskende testingen, observerte vi hvilke problemer og ulemper vi hadde ved appen. For eksempel oppdaget brukerne at appen ikke kjørte like bra på noen mobiltelefoner. Vi analyserte disse resultatene og fant det nødvendig å gjøre videre testing, hvor vi til slutt fant ut at Lottie-animasjonene ble generert på en veldig lite effektiv måte, slik nevnt tidligere i rapporten.

Vi lot brukerne prøve å fremprovosere feil i meldingene generert av Mr Praktisk ved å gi *korrupte* inputverdier. En slik fuzztesting av meldingene gjør at vi kan teste at den genererte teksten ikke inneholder temaer knyttet til vold, selvskade, hat og seksuelt innhold. Ved å teste meldingene fra OpenAI flere ganger kunne vi teste om informasjonen brukeren fikk var pålitelig og forståelig. Ved videreutvikling av appen ville vi sett videre på optimalisering av ytelse og kode-effektivitet

5. Refleksjon

Teamet har jobbet tett ved alle aspekter av prosjektet. Alle viktige beslutninger har blitt tatt i fellesskap og alle har bidratt i utviklingen av appen. Det har ført til at alle har hatt kontroll på hva som har blitt gjort og hvordan vi har ligget an i prosjektet. Det gjør også at alle kan føle eierskap til produktet. Tverrfaglig erfaring av teammedlemmene førte til økt kreativitet og bedre ytelse i både apputviklingen og utforming av relatert dokumentasjon. Teamet vårt har hatt et godt samarbeid med et godt miljø og har derfor oppnådd en synergi. Vi prøvde å følge “Agile Manifesto” og “12 Agile Principles” i arbeidet (Bell et al., 2017, ss. 28-29). Vi har hatt en kultur der alle kan si sin mening fritt. Dette har vært viktig ved den iterative utviklingen der det har kommet forslag i alle stadier av prosjektet. Gruppens mål ble prioritert over individuelle mål, og alle har bidratt til å nå målene.

Selv om samarbeidet fungerte smertefritt, møtte vi likevel på noen utfordringer. Noen av API-ene, spesielt Azure sitt for KI, var veldig vanskelig å få til å fungere. Git konflikter møtte vi også på (spesielt i starten da vi fortsatt lærte oss Git), noe som kunne være frustrerende å løse. Applikasjonen vi har laget har mye funksjonalitet, og å få alt til å fungere sammen og samtidig oppnå lav kobling og høy kohesjon har vært både tidkrevende og utfordrende. Det kunne også til tider være utfordrende å koordinere arbeidet og holde alle medlemmene oppdatert, men dette var noe vi prioriterte høyt og fikk til å fungere. Det har også vært litt uenigheter om beslutninger (for eksempel om utseende av appen) som har ført til litt diskusjoner. Heldigvis kom vi alltid til enighet til slutt, og vi har vært flinke på å inngå kompromisser og unngå krangler.

6. Avslutning

Erfaringer og lærdommer fra hvert medlem:

- *Andreas*: "Jeg har blitt veldig mye dyktigere på programmering, spesielt funksjonell programmering som var helt nytt for meg før dette prosjektet. Også Git var helt nytt for meg, men jeg føler jeg har fått god kontroll på det nå. Jeg har også fått mer praktisk erfaring i smidige metodikker og å samarbeide i et team. Teammedlemmene var veldig greie å jobbe med, så jeg møtte ikke på mange utfordringer knyttet til det. Erfaringene angående samarbeid var derfor utelukkende positive. Jeg synes det var spesielt spennende og moro (selv om det til tider var frustrerende) å jobbe med KI, og å få det til å fungere i en app"
- *Christian*: "Jeg har lært mye om hvordan det er å jobbe med et komplekst og stort program samtidig som flere andre gjør endringer på det samme prosjektet. For å få dette til å gå uten konflikter har jeg blitt flinkere til å kommunisere med andre gruppemedlemmer slik at de vet når jeg ønsker å gjøre endringer, og vice versa."
- *Viktoria*: "Dette studentprosjektet føles som et meget verdifullt utviklingsprosjekt. Gjennom det lærte jeg mye om smidige metodikker og fikk solid erfaring med alle steg av applikasjonsutvikling, samt UX, UI og praktisk bruk av KI til brukervennlige løsninger."
- *Muntaha*: "Det har vært utrolig spennende å kunne delta i dette prosjektet. Jeg har fått muligheten til å virkelig teste kunnskapen jeg har tilegnet meg, samt lære noe nytt. Jeg synes det var spesielt gøy å kunne lære å lage animasjoner."
- *Shanza*: "Jeg synes prosjektet har vært svært lærerikt, og mener jeg har tilegnet meg masse ny kunnskap. Jeg har lært å bruke Figma, bruke Git for versjonshåndtering, utvikle smidig, og ikke minst samarbeide i et team effektivt. Det har vært god kommunikasjon mellom alle medlemmene, og jeg har ikke støtt på noen nevneverdige utfordringer tilknyttet samarbeidet. Alt i alt et veldig lærerikt prosjekt og emne."
- *Hanan*: "Dette var et spennende og lærerikt prosjekt. Jeg fikk prøvd meg på Figma og KI som jeg ellers aldri ville ha kommet bort i. Videre var det hyggelig å bli kjent med alle teammedlemmene da vi hadde et trygt læringsmiljø"

Ett tips til neste års studenter:

Sett av tid til teambuilding og bli trygge på hverandre. Et godt team gjør prosjektet mye lettere og mer motiverende.

Referanser

Accessibility Scanner - Apps on Google Play. (2023, 21 November). *Google Play*. Hentet 13. mai, 2024, fra <https://play.google.com/store/apps/details?id=com.google.android.apps.accessibility.auditor&pli=1>

Android API levels. (2024). API levels | Android versions, SDK/API levels, version codes, codenames, and cumulative usage. *API Levels*. Hentet 26. april, 2024, fra <https://apilevels.com/>

Android Keystore system | App quality. (2024, 3 Januar). *Android Developers*. Hentet 30. april, 2024, fra <https://developer.android.com/privacy-and-security/keystore>

Atlassian. (2024). Scrumban: Mastering Two agile methodologies. *Atlassian*. Hentet 26. april, 2024, fra <https://www.atlassian.com/agile/project-management/scrumban#:~:text=Agile%20project%20management.-,What%20is%20Scrumban%3F,work%2Din%2Dprogress%20limitations.>

Bell, L., Brunton-Spall, M., Smith, R., & Bird, J. (2017). Agile Application Security: Enabling Security in a Continuous Delivery Pipeline. O'Reilly Media.

Geocoding API overview. (2024, 25. april). *Google for Developers*. Hentet 26. april, 2024, fra <https://developers.google.com/maps/documentation/geocoding/overview>

Google Maps Platform Documentation | Maps SDK for Android. (2024, 29. januar). *Google for Developers*. Hentet 26. april, 2024, fra <https://developers.google.com/maps/documentation/android-sdk>

Locationforecast. (2022). *MET*. Hentet 26. april, 2024, fra <https://api.met.no/weatherapi/locationforecast/2.0/documentation>

Metalerts. (2017, 9. november). *MET*. Hentet 26. april, 2024, fra <https://api.met.no/weatherapi/metalerts/2.0/documentation>

Sunrise. (2022). *MET*. Hentet 26. april, 2024, fra <https://api.met.no/weatherapi/sunrise/3.0/documentation>

Model-View-ViewModel - .NET | Microsoft Learn. (2022, November 3). *Learn Microsoft*. Hentet 26. april, 2024, fra <https://learn.microsoft.com/en-us/dotnet/architecture/maui/mvvm>

OpenAI. (2024). Is ChatGPT safe for all ages? *OpenAI Help Center*. Hentet 26. april, 2024, fra <https://help.openai.com/en/articles/8313401-is-chatgpt-safe-for-all-ages>

Security & privacy (2024). *OpenAI*. Hentet 26. april, 2024, fra <https://openai.com/security/>

OpenAI. (2020, 18. september). *OpenAI blog*. Hentet 26. april, 2024, fra <https://openai.com/blog/openai-api>

Pager in Compose | Jetpack Compose. (2024). *Android Developers*. Hentet 26. april, 2024, fra <https://developer.android.com/develop/ui/compose/layouts/pager>

Regjeringen. (2020, 14. januar). Nasjonal strategi for kunstig intelligens - regjeringen.no. *Regjeringen*. Hentet 30. april, 2024, fra <https://www.regjeringen.no/no/dokumenter/nasjonal-strategi-for-kunstig-intelligens/id2685594/>

University at Buffalo. (2024). Universal Design Principles - Accessibility at UB. *University at Buffalo*. Hentet 13. mai, 2024, fra <https://www.buffalo.edu/access/help-and-support/topic3/universaldesignprinciples.html>

Universell utforming og tilgjengelighet. (2024). *Bufdir*. Hentet 26. april, 2024, fra <https://www.bufdir.no/likestilling/universell-utforming/>

Web Content Accessibility Guidelines (WCAG) 2.1. (2023, September 21). W3C. Hentet 7. mai, 2024, fra <https://www.w3.org/TR/WCAG21/>

Soleng, H. H., & Vihovde, E. H. (2022). Plettfrí kode. Universitetsforlaget.

Accessibility in jetpack compose (2024). *Android developers*. Hentet 7. mai, 2024, fra <https://developer.android.com/develop/ui/compose/accessibility>

Google LLC. (2023). Accessibility Scanner. *Google play store*. Hentet 7. mai, 2024, fra <https://play.google.com/store/apps/details?id=com.google.android.apps.accessibility.auditor&hl=en&gl=US>

Vedlegg

Vedlegg 1: Spørreundersøkelse og resultater

Spørreskjema for spørreundersøkelsen: [Værapp - Nettskjema](#)

Formål: Å samle inn informasjon om brukernes vaner og preferanser angående værvarslinger og apper.

Aldersgruppe: 9 - 25 år

Nettskjemaet sendes til brukere i valgt aldersgruppe (9-25 år) via e-post og sosiale medier som Messenger, Instagram. Nettskjemaet er åpent for alle med lenke. Spørreundersøkelsen unngår innsamling av sensitive data som persondata, helseinformasjon eller noe som kan forbinde svarene med konkrete fysiske personer. Hentet informasjon brukes kun til undervisningsformål. Svarene lagres i ett år og slettes etterpå automatisk fra UiOs servere.

Resultatene er lagret som pdf-fil: [!\[\]\(3f1d9e9f6bcc0837db71d34c7a09f75d_img.jpg\) resultater_spørreundersøkelsen.pdf](#)

Hvilke av disse funksjonene ville du foretrukket å ha i din værapp?

Antall svar: 31

Svar	Antall	% av svar	
Daglige notifikasjoner om været (push-varsler)	13	41.9%	 41.9%
Praktiske tips hvordan man skal kle seg (skjerf, lue, solbriller)	19	61.3%	 61.3%
Katastrofevarsler	23	74.2%	 74.2%
Legge til flere sted som favoritter	12	38.7%	 38.7%
Værkart	7	22.6%	 22.6%
Værrelaterte nyheter	5	16.1%	 16.1%

Vedlegg 2: Intervjuguide og intervjunotater for målgruppen 9 - 25 år.

Intervjuguide

Formål: Å få dypere innsikt i brukernes individuelle oppfatninger og preferanser samt tilfredsstillhet ved bruk av væranner.

Introduksjon:

- Kort introduksjon av prosjektet og formålet med intervjuet.
- Rettigheter og forpliktelser. Samtykkeskjema.
- Forklare at alle svarene blir brukt til undervisningsmål, navnene blir anonymisert.

Oppvarming:

- Kan du fortelle litt om deg selv? Hvor gammel er du?
- Hvordan får du vite om været: fra foreldrene, TV, app, osv?

Hoveddel:

- Bruker du noen væranner i dag?
- Hvilke funksjoner liker du mest av alt?
- Er det noe som disse appene mangler? Noe du ville ta bort (irriterende)?
- Ville du foretrukket å få værmeldinger daglig?
- Tror du at det er en god ide å vise anbefalinger på hvordan man kan kle på seg, f.eks. lue, skjerf, osv.

Avrundning:

- Hvor ofte ville du bruke værappen hvis du er helt fornøyd med den?
- Foretrekker du fargerike eller nøytrale farger i appen?

Avslutning:

- Takker for deltagelse og gi kontaktinformasjon for videre spørsmål.

Intervjunotater

Intervjuobjekt 1:

Anna (navnet er anonymisert), 24 år. Anna jobber deltid som barnehageassistent og studerer fulltid ved UiO, pedagogikk, årsenhet. Anna får vite om vær gjennom værapp integrert i sin mobil (Android), og har vært notifikasjon synlig på hovedskjermen. Hun verken ser på værprogrammer på TV eller hører på radio. Anna liker ganske godt at appen viser henne små animasjoner og det gir henne hints om hva hun skal kle på seg. Hun foretrekker å få værmeldinger daglig og liker nøytrale farger i appene sine.

Intervjuobjekt 2:

Henrik (navnet er anonymisert), 22 år. Henrik jobber som selger i sportsbutikken. Han tenker ikke så mye om været, men sjekker av og til været på sin mobil (Apple). Han er stort sett interessert om det skal regne for mye og om han skal hente paraplyet med seg. Henrik kan også sjekke været for å bestemme om han skal sykle eller gå til fots til jobben sin. Hvis han sykler, prøver han også å sjekke “regnfrie partene av dagen”. Henrik bryr seg ikke om fargene brukt i appen så vidt de ikke er irriterende. Han ville foretrekke å få push-meldinger med “farlige notifikasjoner” om ekstremt vind/ regn, osv som for eksempel kan bli “fare med sykling fra 7-9”, “ta med brodrene”, osv.

Intervjuobjekt 3:

Ole (navnet er anonymisert), 12 år. Ole går på 7.trinn og bruker for det meste Snapchat for å se været. Han bryr seg ikke om hvordan været er, og vil bare vite om han trenger å ha på en jakke eller ikke. Den viktigste funksjonen for han er å kunne se hvor mange grader det er ute, og han har ikke merket et behov for noe mer. Hvis han blir fornøyd kan han se for seg å bruke værappen daglig.

Intervjuobjekt 4:

Sverre (navnet er anonymisert), 15 år. Sverre går i tiende klasse og bruker vanligvis apple værapp på mobilen, og yr når han bruker pc-en for å sjekke været. I en perfekt værapp vil han ha daglige notifikasjoner, farevarsler og praktiske tips. Han synes værappen til Yr er forvirrende og skulle ønske det var lettere å få med seg været for hele dagen.

Intervjuobjekt 5

Helene (navnet er anonymisert), 18 år. Helene går siste året på videregående og jobber kun når hun har ferie, slik som i sommerferien gjennom sommerskolen. Hun sjekker været hver dag gjennom apples værapp. Hun liker når apper er oversiktlige, og hun synes yr-værapp er forvirrende og inneholder for mange funksjonaliteter. Det er kult med de ulike funksjonene, men det gjør appen vanskelig å bruke.

Vedlegg 3: Personer



Preben

Alder: 21

Utdanning: studerer pedagogikk

Sivil status: bor i kollektiv

Yrke: barista på Starbucks

Prosentstilling: 20%

"Det er kjedelig å sjekke Yr, og så vidt jeg vet mangler det værvarsler. Jeg kan kanskje planlegge dagen min mer fornuftig hvis jeg vet om farer i nærheten"

Mål

- Ikke sjekke værelaterte nettsteder som Yr
- Få værvarsler på mobilen uten å gjøre noe
- Bruke vær- og katastrofevarsler for bedre planlegging

Frustrasjoner

- Det er kjedelig og irriterende å sjekke været hver dag
- Vi har utetermometer. Men jeg har ikke tid å se på den om morgenen
- Væroptimist – men blir ofte våt eller kald

Preben er 21-årig fulltidsstudent ved UiO. Han tar en bachelorgrad innen pedagogikk og jobber deltid som barista på Starbucks. Preben sykler mye både til/fra universitetet og jobben sin. Han er ganske praktisk og vil ha vær- og katastrofevarsler på mobilen for å planlegge sine ruter og klær på forhånd. Han er glad i teknologier som kan gjøre hans hverdag lettere og mindre stressfull.



Asma

Alder: 15

Utdanning: Går på ungdomskole

Sivil status: bor med foreldre

Yrke: judo-instruktør

Hobbyer: fotball, judo

"Jeg sjekker været hver dag før skolen. Jeg vil vite hvordan været er ut dagen og hvordan jeg bør kle meg"

Mål

- Vite hvilke klær som passer til været
- Få vite været ut dagen
- Vite hvordan været påvirker hobbyene mine
- Ha oversikt over eventuelle vær- og katastrofevarsler
- Ha en app med pent design

Frustrasjoner

- Værutviklingen gjennom dagen er ikke alltid åpenbart for meg.
- Jeg forstår ikke hva antall mm med nedbør betyr
- Utseendet på min nåværende app er ganske kjedelig

Asma går på ungdomsskole og har fritidsaktiviteter etter skoletid. Det er viktig for henne å kjenne til værutviklingen gjennom dagen, fordi hun ikke ønsker å ta med unødvendige ting som f.eks. paraply hvis det holder å ha på seg en god jakke. Asma er også opptatt av designet til appene og ville foretrukket å bruke noe som ser pent ut.

Vedlegg 4: Funksjonelle og ikke-funksjonelle krav

Funksjonelle krav

Type	Beskrivelse
Systemkrav	Applikasjonen skal kunne vise aktuell temperatur og værforhold for brukerens nåværende lokasjon eller en forhåndsinnstilt lokasjon.
	Applikasjonen skal ha Hjem-skjerm, Søke-skjerm, Favoritt-skjerm og Innstillinger-skjerm.
	Applikasjonen skal kunne vise farevarsel for de neste timene og dagene, inkludert passende vær-ikoner.
	Brukeren skal kunne legge til ett eller flere steder som favoritter.
	Applikasjonen skal kunne gi varsler om ekstremvær og farlige værforhold som er relevante for brukerens forhåndsinnstilte eller valgte lokasjon.
	Applikasjonen skal vise vindretning.
	Applikasjonen skal vise tidspunktet på når det er solnedgang og soloppgang.
	Farevarsler skal være lett synlige og forståelige, med bruk av ikoner eller fargekoder for å representer alvorsgraden av situasjonen.
	Daglig værmelding skal bli gitt som push varsling og være basert på brukernes lokasjon (spør og henter posisjon).
	Applikasjonen skal tillate brukere å legge til personlige hobbyer eller interesser, og basert på disse skal appen kunne gi værrelaterte anbefalinger eller varsler.
Funksjonelle krav	Applikasjonen skal inneholde en AI-maskot, Mr. Praktisk, som gir skriftlig informasjon om dagens vær og praktiske tips.
	Applikasjonen skal kunne identifisere brukerens nåværende lokasjon, dersom brukeren ønsker å oppgi det, for å gi nøyaktig værinformasjon og varsler.
Funksjonelle krav	Applikasjonen skal tilby en onboarding aller første gang appen tas i bruk, der det blir bedt om samtykke og valg av innstillinger.

	<p>Systemet skal huske data fra forrige økt/app-besøk.</p>
	<p>Applikasjonen skal ikke kræsje dersom systemet mangler internett, men heller gi feilmeldinger.</p>
	<p>Systemet skal gi brukeren tidspunkt for siste oppdatering.</p>
	<p>Systemet skal kunne vise brukeren områder for farevarsler på et kart.</p>
	<p>Systemet skal gi brukeren Toast-meldinger for endring av favoritter og bakgrunn.</p>
	<p>Systemet skal ha tilgang til stedsnavn i Norge (byer, kommuner, bydeler, etc.) som brukeren skal kunne søke etter og se vær for.</p>
	<p>Systemet skal fungere på Android-enheter som har Android 9 eller nyere.</p>
	<p>Applikasjonen skal ha et intuitivt design der kontrollene og deres virkninger er tydelig og direkte knyttet sammen.</p>
	<p>Systemet skal avklare til brukeren når informasjonen på skjermen oppdateres.</p>
	<p>Applikasjonen skal ikke kunne roteres til horisontal visning.</p>
Brukerkarav	<p>Man skal ha tilgang til appen uten behov for registrering eller innlogging.</p>
	<p>Brukeren skal kunne administrere sine favorittsteder (legg til, slette).</p>
	<p>Brukeren skal kunne tilpasse appens bakgrunn for å skape et mer personlig grensesnitt.</p>
	<p>Brukere skal kunne benytte funksjonalitet som ikke krever internett, selv når systemet mangler internett tilkobling.</p>
	<p>Brukeren skal oppdatere data manuelt.</p>

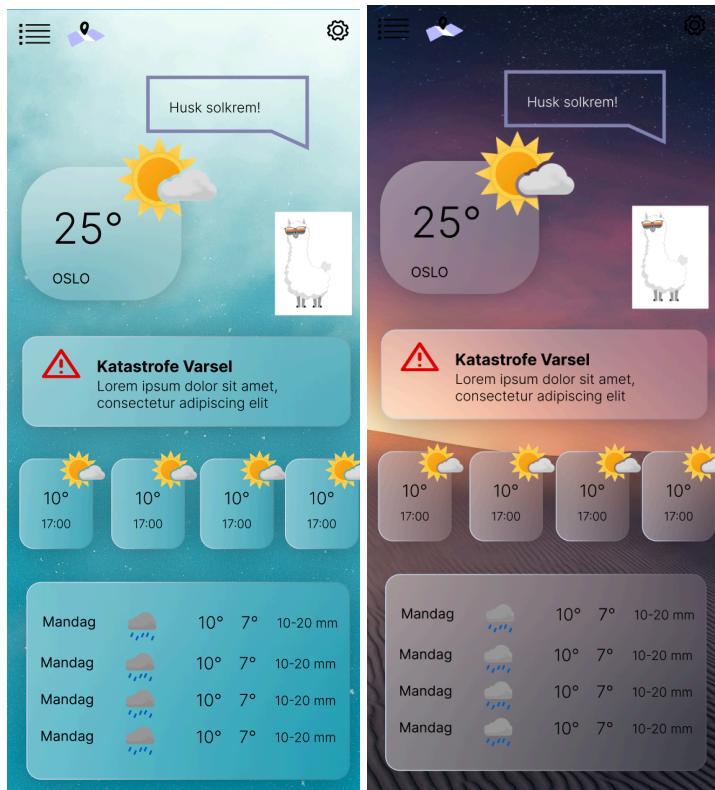
Ikke funksjonelle krav

Type	Beskrivelse
Produktkrav	<p>Applikasjonen skal være tilgjengelig på mobil (mobilapplikasjon).</p> <p>Grensesnittet skal være brukervennlig og intuitivt å bruke og navigere.</p> <p>Lyse farger skal brukes for å ha klare kontraster mellom tekst, bilde og bakgrunn.</p> <p>Brukergrensesnittet skal være intuitivt og enkelt nok for barn så unge som 13 år.</p> <p>Applikasjonen skal bruke barnevennlige ikoner og bilder.</p> <p>Teksten skal være kompatibel med enheters forhåndsbestemte tekststørrelser.</p> <p>Navigering og scrolling skal skje raskt, sømløst og “snappy”.</p> <p>Applikasjonen skal bruke ikoner for å vise når skjermene laster.</p> <p>Applikasjonen skal tilrettelegge for vertikal scrolling for å gi brukerne mulighet til å navigere nedover på skjermen for å vise mer innhold.</p> <p>Applikasjonen skal gjøre det mulig å navigere mellom skjermene horisontalt.</p> <p>Koden skal være skrevet effektivt slik at appen ikke krever mer kraft fra enheten enn nødvendig.</p> <p>Kall fra Met sine api skal skje gjennom en proxy for datasikkerhet og å unngå overbelastning av kall.</p> <p>Applikasjonen skal ha rask responsivitet.</p> <p>Navigasjonen mellom skjermer skal skje på bunnen av skjermen med ikoner.</p>
Organisasjonskrav	<p>Utvikling av applikasjonen skal skje med Github for versjonshåndtering og vedlikehold.</p> <p>Utviklingsmetoder og prosesser skal være smidige.</p>
Eksterne krav	Lagring og sletting av brukerdata skal være i henhold til vilkår gitt i applikasjonens samtykkeskjema.

Applikasjonen skal unngå henting og lagring av sensitive data.

Vedlegg 5: Low-fidelity og high-fidelity prototype





Vedlegg 6: Samtykkeskjema for geriljatesting og brukerundersøkelsen

Vil du delta i vårt studentprosjekt?

Vi er studenter i emnet *IN2000 – Software Engineering med prosjektarbeid* ved Institutt for informatikk ved Universitetet i Oslo. Med dette skrivet ønsker vi å informere hva prosjektet vårt har som formål, spørre deg om du vil delta i prosjektet, samt berette hva deltagelse vil innebære for deg.

Formål

I kurset *IN2000* utvikler Bachelorstudenter i informatikk en mobilapplikasjon for vær- og farevarsler for yngre brukere (9 - 25 år). Formålet er å få innsikt i hvordan målgruppen bruker værapplikasjoner, samt hvilke funksjoner brukerne er glade i og hvilke bør unngås i utvikling av applikasjonen.

Deltakelse

Du blir bedt om å delta på en brukertesting av applikasjonen utviklet i løpet av prosjektet, som skal vare mellom 5 og 15 minutter. For deltakere som er yngre enn 18 år skal samtykkeskjemaet underskrives av en foresatt. Flere pauser kan tas ved behov. Dersom du velger å delta ønsker vi å ta lydopptak fra intervjuet for vår datainnsamling.

Frivillig deltagelse

Det er frivillig å delta. Du kan når som helst avslutte intervjuet eller trekke tilbake informasjon som er gitt. Du kan når som helst velge å trekke samtykket uten å måtte oppgi grunn. Dersom samtykket trekkes vil eventuelle personopplysninger som er innsamlet om deg slettes og det vil ikke innebære noen negative konsekvenser for deg at du velger å trekke ditt samtykke.

Personvern: innsamling, oppbevaring, behandling og bruk av dine opplysninger

Ingen sensitive personopplysninger (jf. Personvernforordningens artikkel 9 og 10) vil bli innsamlet. Personlige opplysninger om deg vil kun benyttes til formålene beskrevet i dette informasjonsskrivet. Vi behandler opplysningene konfidensielt og i samsvar med personvernregelverket.

Kursledelsen (Yngve Lindsjørn, Steffen Almås og Viktoria Stray) vil få tilgang til de fullstendige prosjektrapportene, og vi vil presentere resultater fra våre studier med medstudenter og kursledelsen i gruppseminarer (Simon Tidemann Halvorsen og Julia Lundeby). Rapportene vil ikke bli delt med andre enn kursdeltakerne. Deltakerne vil bli anonymisert i rapporten.

Navn og kontaktinformasjon som eventuelt oppstår under undersøkelsen vil bli sensurert. Dataen kan ettersendes til deg ved ønske. Dataen som oppbevares, inkludert anonymisert data, vil ikke bli publisert og vil heller ikke kunne tilbakeføres til deg.

Hva skjer med innsamlet data når studentprosjektet avsluttes?

Alle opptak blir slettet senest 3.juni.2024. Dette gjelder også anonymiserte og avidentifiserte opplysninger om deg.

Rettigheter

Vi behandler opplysninger om deg basert på ditt samtykke. Så lenge du kan identifiseres i datamaterialet, har du rett til:

- innsyn i hvilke personopplysninger som er registrert om deg, og å få utlevert en kopi av opplysingene,
- å få rettet personopplysninger om deg,
- å få slettet personopplysninger om deg, og
- å sende klage til Datatilsynet om behandlingen av dine personopplysninger.

Hvis du har spørsmål til undersøkelsen, eller ønsker å benytte deg av dine rettigheter, ta kontakt med en av oss ved e-post eller telefon.

Før intervjuet begynner, ber vi deg om å samtykke i deltagelsen ved å undertegne på at du har lest og forstått informasjonen på dette samtykkeskjemaet, og ønsker å stille opp til intervju.

Med vennlig hilsen,

Andreas Klæboe
e-post: andrklae@ifi.uio.no
tlf: 967 98 001

Christian Fjeld Thorkildsen
e-post: christft@ifi.uio.no
tlf: 478 13 835

Viktoriaa Olonova
e-post: viktorol@ifi.uio.no
tlf: 485 00 584

Muntaha Dheeg
epost: muntahad@ifi.uio.no
tlf: 477 31 239

Hanan Mohamud
epost: hananam@ifi.uio.no
tlf: 907 888 11

Shanza Ehsan
epost: shanzae@ifi.uio.no
tlf: 968 26 318

Samtykkeerklæring

Jeg har mottatt og forstått informasjon om undersøkelsen, og har fått anledning til å stille spørsmål. Jeg samtykker til:

- lydopptak under undersøkelsen

Jeg samtykker til at mine opplysninger behandles frem til semesteret er avsluttet.

Sted og dato

Fullt navn

Signatur

Vedlegg 7: Geriljatesting

Instrukser til brukeren

1. åpne appen med wifi av og løs problemer som oppstår
2. Finn nærmeste værvarsel i bergen, og se hvor det er i kartet
3. Finn vær for nåværende posisjon
4. Finn temperaturen for fredag i Oslo
5. Legg til et valgfritt sted i norge i favoritter, og se været for det stedet
6. Naviger til innstillinger og sett det etter dine preferanser
7. Få en oppsummering av været det neste døgnet av Mr. Praktisk
8. Utforsk appen på egenhånd

Spørsmål etterpå

- Var appen enkel å bruke?
- Er det noen funksjonalitet du savner?
- Var det noe du misforsto?
- Har du noen forslag til forbedringer?
- Hva synes du om tipsene til Mr. Praktisk?

Mal for rapport:

Kandidat #[nummer]

Alder:

Kjønn:

Utdanningsnivå:

Yrke:

Oppgaver:

1. åpne appen med wifi av og løs problemer som oppstår
 - a. suksess [ja / nei]
 - b. kommentarer

2. Finn nærmeste værvarsle i bergen, og se hvor det er i kartet
 - a. suksess [ja / nei]
 - b. kommentarer
3. Finn vær for nåværende posisjon
 - a. suksess [ja / nei]
 - b. kommentarer (hvor i appen søkte h*n etter posisjonen?)
4. Finn temperaturen for fredag i Oslo
 - a. suksess [ja / nei]
 - b. kommentarer (hvor i appen søkte h*n etter posisjonen?)
5. Legg til et valgfritt sted i norge i favoritter, og se været for det stedet
 - a. suksess [ja / nei]
 - b. kommentarer (hvordan fant h*n posisjonen og hvordan ble den lagt til?)
6. Naviger til innstillinger og sett det etter dine preferanser
 - a. suksess [ja / nei]
 - b. kommentarer
7. Få en oppsummering av været det neste døgnet av Mr. Praktisk
 - a. suksess [ja / nei]
 - b. kommentarer
8. Utforsk appen på egenhånd
 - a. kommentarer

Spørsmål:

- Var appen enkel å bruke?
- Er det noen funksjonalitet du savner?
- Var det noe du misforsto?
- Har du noen forslag til forbedringer?
- Hva synes du om tipsene til Mr. Praktisk?

Oppsummering

Skriv en skriv en oppsummering

Kandidater

Kandidat #1

Alder: 23

Kjønn: Mann

Utdanningsnivå: Student

Yrke: Kundeservice

Oppgaver:

1. åpne appen med wifi av og løs problemer som oppstår
 - a. ja
 - b. kommentarer
2. Finn nærmeste værvarsel i bergen, og se hvor det er i kartet
 - a. delvis
 - b. tok tid å skjonne hvordan åpne varselet i kartet
3. Finn vær for nåværende posisjon
 - a. ja
 - b. trykket på ikonet på toppen av hjemskjermen. kompassnål hadde vært bedre enn lokasjon ikon
4. Finn temperaturen for fredag i Oslo
 - a. ja
 - b. Fant det enkelt og greit
5. Legg til et valgfritt sted i norge i favoritter, og se været for det stedet
 - a. ja
 - b. søkte fra hjemskjerm. bokmerket ga mening, men “stjerne hadde fungert like bra om ikke bedre”.
6. Naviger til innstillinger og sett det etter dine preferanser
 - a. ja
 - b. enkelt å greit
7. Få tips og råd av mr. praktisk
 - a. ja
 - b. “rart den foreslo innendørs golfbane” ellers veldig fornøyd og fascinert.
Perfekt lengde på svar
8. Få en oppsummering av været det neste døgnet av Mr. Praktisk

- a. ja
 - b. God oppsummering, dårlig layout.
9. Utforsk appen på egenhånd
- a. mye testing med gpt og ulike hobbyer: mye latter, morsomme og brukbare svar

Spørsmål:

- Hvordan var brukervennligheten
 - Veldig grei og enkel
 - posisjon burde vært et kompassnål
 - kan være litt forvirrende i starten, men raskt å lære
- Er det noen funksjonalitet du savner?
 - Vindretning (kompassretning, kan være dynamisk med hvilken retning telefonen peker)
- Var det noe du misforsto?
 - “ikke egentlig, bare et helt nytt layout som man må bli kjent og vant med”
- Har du noen forslag til forbedringer?
 - Layout på døgn oppsummering av mr. praktisk
 - døgn burde ha mulighet for å se som tabell i tillegg (ikke i stedet for, men i tillegg) til skroll (brukte yr sin som et bra eksempel)
 - sol oppgang og nedgang
- Hva synes du om tipsene til Mr. Praktisk? har han nytteverdi
 - kult og moro
 - Stor nytteverdi
 - lett og grei oppsummering av været
 - en som gir ikke svare i stedet for “se ut av vinduet” fra foreldre
 - burde vite dato så han ikke foreslår f.eks. en skitur i april
 - burde også oppsummere uken. har man en hobby bør han si hvilken dag det passer best (f.eks. golf den dagen det er finest vær)
- Annet
 - Valg om flere design for mr. praktisk, og en frøken praktisk.

Oppsummering

Ganske koselig app. Kandidat likte KI assistent og ville bruke appen når den blir ferdig.

Kandidat #2

Alder: 25

Kjønn: Kvinne

Utdanningsnivå: vgs

Yrke: fulltid

Oppgaver:

1. åpne appen med wifi av og løs problemer som oppstår
 - a. ja
 - b. klarte det greit
2. Finn nærmeste værvarsel i bergen, og se hvor det er i kartet
 - a. suksess [ja / nei]
 - b. kommentarer
3. Finn vær for nåværende posisjon
 - a. nei
 - b. fant tilslutt, men trengte hint om hva vær varsel er og hvordan åpne kartet
4. Finn temperaturen for fredag i Oslo
 - a. ja
 - b. fant i hjem skjermen, brukte litt tid, men fant frem enkelt
5. Legg til et valgfritt sted i norge i favoritter, og se været for det stedet
 - a. ja
 - b. Fant enkelt og greit, og søkte fra hjemskjermen.
6. Naviger til innstillinger og sett det etter dine preferanser
 - a. ja
 - b. fikk det til enkelt
7. Få en oppsummering av været det neste døgnet av Mr. Praktisk
 - a. ja
 - b. Gikk enkelt og greit
8. Utforsk appen på egenhånd
 - a. Velger man min posisjon bør det stå hvor i landet det er.

Spørsmål:

- Hvordan er appen i bruk

- Enkelt å forstå
 - oversiktlig
- Er det noen funksjonalitet du savner?
 - Sol oppgang og nedgang
 - Nordlys varsel
 - kort tekst som beskriver været typ mr. praktisk, men kortere (1 - 2 setninger)
- Var det noe du misforsto?
 - hva “værvarsel” betyr
 - At “min posisjon” betyr at man ser været for min posisjon. Legg til stedsnavn for min posisjon over. Bør være så nøyaktig som bydel.
- Har du noen forslag til forbedringer?
 - Burde se tidspunkt for værvarsel og om det er pågående eller på vei
 - pilen på værvarsel bør være større og nedenfor linja
 - Farevarsler i varsel skjermen bør være sortert etter hvor nærmeste de er fra der man er nå.
 - på hjem skjerm bør det stå stedet varslene er x km unna.
- Hva synes du om tipsene til Mr. Praktisk?
 - Noen ganger ga han uforståelige svar
 - koselig og hyggelig
 - gir relevante tips for været
 - God melding, god lengde
- annet
 - rart at det kom opp så mange plasser langt unna på værvarsler for sted
 - “værvarsel” var litt forvirrende. kunne heller hete “farevarsel”
 - burde kunne filtrere ut farevarsler dersom de er langt unna.

Oppsummering

Koselig og hyggelig app som gir relevante tips for været. Enkelt å forstå, oversiktlig, intuitivt å bruke. Det er kult og moro å ha Mr. Praktisk som gir grei oppsummering av været.

Kandidat #3

Alder: 13

Kjønn: Mann

Utdanningsnivå: Grunnskole

Yrke: Ingen

Oppgaver:

1. åpne appen med wifi av og løs problemer som oppstår
 - a. *Ja*
 - b. *Vanskelig å gjøre dette gjennom skjermen fra datamaskinen. Jeg er ikke vant til det.*
2. Finn nærmeste værvarsel i bergen, og se hvor det er i kartet
 - a. *Ja*
 - b. *Gikk gjennom favorittskjermen for å søke etter Bergen, men tok litt tid å skjonne at ikonet på værvarslet skulle vise et kart.*
3. Finn vær for nåværende posisjon
 - a. *Ja*
 - b. *Gikk gjennom favorittskjermen*
4. Finn temperaturen for fredag i Oslo
 - a. *Ja*
 - b. *Ganske enkelt å finne, siden det er sånn vær-apper vanligvis ser ut.*
5. Legg til et valgfritt sted i norge i favoritter, og se været for det stedet
 - a. *Ja*
 - b. *Gikk gjennom favorittskjermen*
6. Naviger til innstillinger og sett det etter dine preferanser
 - a. *Ja*
 - b. *Kult at man kan bytte bilde, jeg skulle ønske jeg kunne velge noe selv også.*
7. Få en oppsummering av været det neste døgnet av Mr. Praktisk
 - a. *Ja*
 - b. *Det er morsomt at han er skalla.*
8. Utforsk appen på egenhånd

- a. *Veldig enkel å bruke, gøy at man kan tilpasse etter hva man liker også. Den gamle mannen ser morsom ut.*

Spørsmål:

- Var appen enkel å bruke?
 - Ja
- Er det noen funksjonalitet du savner?
 - Jeg skulle ønske jeg kunne velge noen bilder selv.
- Var det noe du misforsto?
 - Jeg skjønte ikke helt at ikonet for værvæselet var for kart.
- Har du noen forslag til forbedringer?
 - Jeg skulle ønske at jeg kunne velge bakgrunn selv.
- Hva synes du om tipsene til Mr. Praktisk?
 - Han svarte på det jeg lurte på mest, om jeg kunne spille fotball eller ikke, så jeg er fornøyd.

Oppsummering

Enkel å forstå og bruke. Gøy med egne tilpasninger og tips fra Mr.Praktisk

Kandidat # 4

Alder: 16

Kjønn:Mann

Utdanningsnivå: Videregående

Yrke: Ingen

Oppgaver:

1. åpne appen med wifi av og løs problemer som oppstår
 - a. *Ja*
 - b. *Enkelt og greit*
2. Finn nærmeste værvarsel i bergen, og se hvor det er i kartet
 - a. *Ja*
 - b. *Gikk ganske fort å sjekke*
3. Finn vær for nåværende posisjon
 - a. *Ja*

- b. Gikk gjennom favorittskjermen
- 4. Finn temperaturen for fredag i Oslo
 - a. Ja
 - b. *Gir mening å skrolle ned for å se*
- 5. Legg til et valgfritt sted i norge i favoritter, og se været for det stedet
 - a. Ja
 - b. Enkelt å forstå
- 6. Naviger til innstillinger og sett det etter dine preferanser
 - a. Ja
 - b. *Kult å kunne bytte bakgrunner*
- 7. Få en oppsummering av været det neste døgnet av Mr. Praktisk
 - a. Ja
 - b. *Morsom figur*
- 8. Utforsk appen på egenhånd
 - a. *Enkel navigering og oversiktlig informasjon*

Spørsmål:

- Var appen enkel å bruke?
 - Ja det gav mening
- Er det noen funksjonalitet du savner?
 - Nei, ikke noe jeg klarer å tenke på nå
- Var det noe du misforsto?
 - Nei alt var ganske enkelt
- Har du noen forslag til forbedringer?
 - Nei
- Hva synes du om tipsene til Mr. Praktisk?
 - Ganske nyttig og han er morsom å se på.

Oppsummering

Jeg pleier ikke å bry meg om væranner, men det var gøy å se noe unikt. Første gang jeg har sett en værapp som er tilpasningsdyktig.

Kandidat #5

Alder: 18

Kjønn: Mann

Utdanningsnivå: Videregående

Yrke: Ingen

Oppgaver:

1. åpne appen med wifi av og løs problemer som oppstår
 - a. *Ja*
 - b. *Enkelt å få til.*
2. Finn nærmeste værvarsel i bergen, og se hvor det er i kartet
 - a. *Delvis*
 - b. *Skjønte ikke hva "værvarsel" betydde, men klarte å finne frem til Bergen.*
3. Finn vær for nåværende posisjon
 - a. *Ja*
 - b. *Søkte seg frem gjennom favorittskjermen*
4. Finn temperaturen for fredag i Oslo
 - a. *Ja*
 - b. *Fant fram veldig enkelt*
5. Legg til et valgfritt sted i norge i favoritter, og se været for det stedet
 - a. *Ja*
 - b. *Søkte og la til et sted fra favorittskjermen*
6. Naviger til innstillinger og sett det etter dine preferanser
 - a. *Ja*
 - b. *Gøy at man kan velge mellom bakgrunner*
7. Få en oppsummering av været det neste døgnet av Mr. Praktisk
 - a. *Ja*
 - b. *Morsom kommentar*
8. Utforsk appen på egenhånd
 - a. *Ingen kommentarer*

Spørsmål:

- Var appen enkel å bruke?
 - Ja ganske enkel og grei.
- Er det noen funksjonalitet du savner?
 - Jeg vil ha lysere bakgrunnsbilder, det var litt mye med astronaut valget.
- Var det noe du misforsto?
 - Jeg skjønte ikke hva “værvarsle” betydde
- Har du noen forslag til forbedringer?
 - Kanskje bedre bilder for å sette bakgrund
- Hva synes du om tipsene til Mr. Praktisk?
 - Veldig nyttig

Oppsummering

Tipsene fra Mr.Praktisk var ganske nyttig og applikasjonen var ganske enkel og grei å bruke.

Forslag til forbedring:

- Mer intuitiv ikon for kart
- Erstatt navnet “værvasler” med “farevarsler”
- Legge til soloppgang og solnedgang
- Forkorte tekst som beskriver Mr. Praktisk
- Sette opp delay slik at Mr. Praktisk ikke skriver noe rart når man trykker på den flere ganger på rad
- Slå sammen boksen med nåværende posisjon og temperatur, dvs. Oslo, værikon og temperatur vises en boks
- Søkefeltet skal byttes til å kun være i favourite screen
- bokmerkeikon skal byttes med søkerikon i navigasjon
- Velg lysere bakgrunnsbilder

Vedlegg 8. Brukertesting

Instrukser til brukeren

1. åpne appen med wifi av og løs problemer som oppstår
2. Finn nærmeste farevarsel i bergen, og se hvor det er i kartet
3. Finn vær for nåværende posisjon
4. Finn temperaturen for fredag i Oslo
5. Legg til et valgfritt sted i norge i favoritter, og se været for det stedet
6. Naviger til innstillinger og sett det etter dine preferanser
7. Få en oppsummering av været det neste døgnet av Mr. Praktisk
8. Utforsk appen på egenhånd

Bruk 1: "Jeg synes appen er veldig söt". Jente, 25 år

1. Problemer med internett tilgang. Måtte bruke mob internett istedenfor eduroam
2. Tok omrent 20 sec, vi måtte forklare oppgaven
3. Tok 3-5 sec
4. Tolk 5 sec
5. Lett, tok flere sec. Men hun trykket først på "Favoritter" istedenfor søkerikon.
6. Enkelt å tilpasse
7. Hun likte Mr. Praktisk. "Han er veldig söt".
8. Opplevde litt dårlig internettforbindelse igjen. Likte bakgrunnsskjema

Brukeren svarte også på nettskjemaet.

Bruk 2: "" Gutt, 23 år

1. lett
2. Jeg fant det. Det tok ikke lang tid
3. Det var lett, men tok litt lengre i
4. kjempefrott
5. Jeg kan legge til favoritter hvor jeg bor eller hvor jeg er fra. Det tok 2-3 sec for å forstå at favorittene ligger under bokmerke-ikon
6. La "tennis" som hobby
7. Er det chat GPT? Han leste alt og la, siden Mr. Praktisk ga anbefalinger ikke å spille tennis
8. Sky i kveld. Kult å lage egne bilder.

Bruk 3: "" Gutt, 20 år og Bruker 4: "Det er gode tips" jente, 21 år (testet samtidig)

1. Lett
2. Nice

3. Kjapt, tok 3 sec
4. Tolk litt tid.
5. Lagt til sted enkelt
6. Hobby “sosialantropologi”. Sjekker tips etterpå. “Det er gode tips”. Mr. Praktisk skrev på engelsk og startet med “Hey bro”

Bruk 5: jente, 20 år “Å så söt. Det var gøy” - angående tips fra Mr. Praktisk

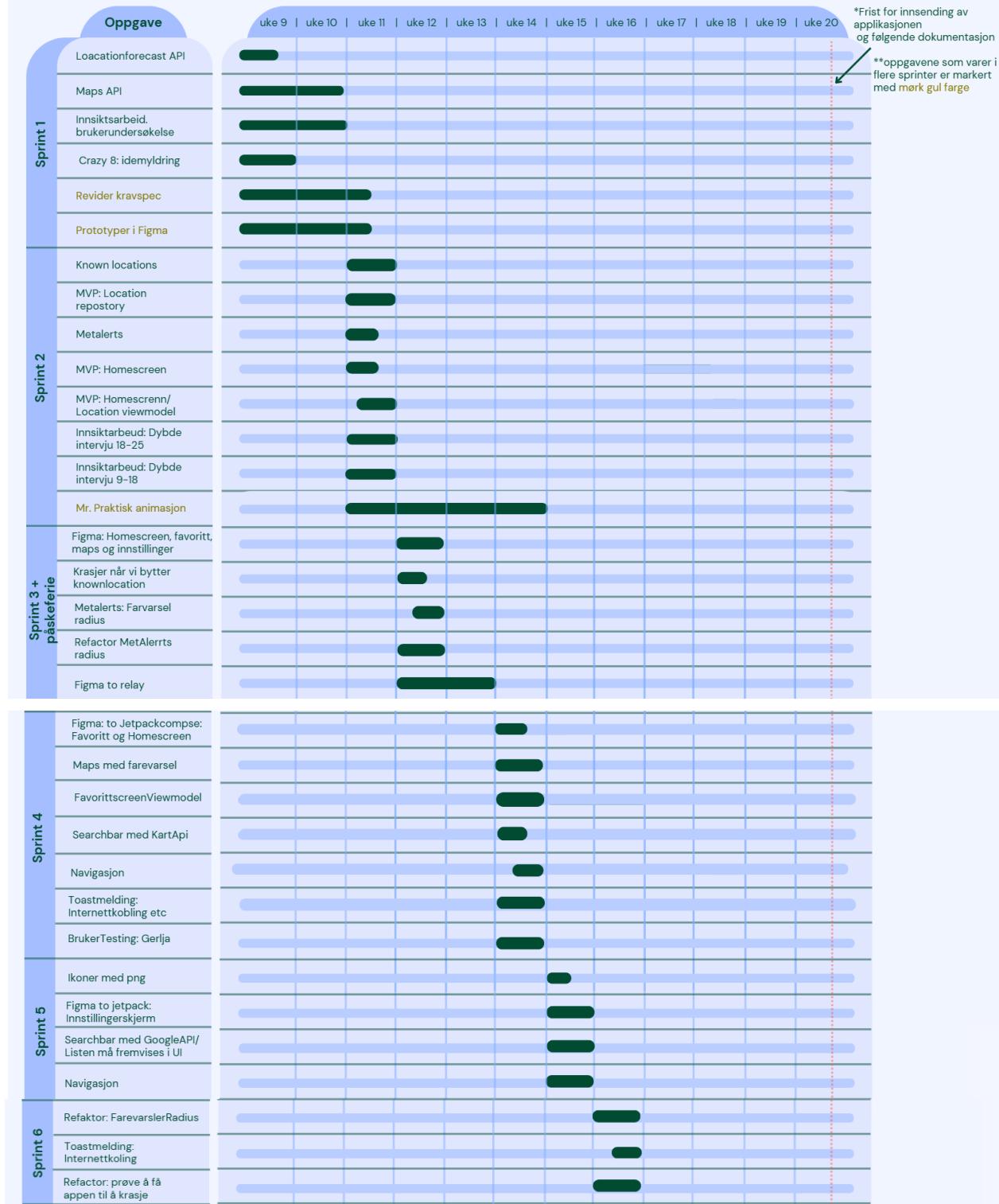
1. lett
2. Lett å åpne kartet
3. Oppdaterte nåværende posisjon enkelt
4. Scrollet for været på fredag
5. Fant søkefeltet og tapte. Det var enkelt å legge en posisjon til
6. Skrev flere hobbyer. Spurte om tips. “Oh, så söt”
7. Den gir ordentlig informasjon. Hun likte væroppsummeringen.
8. Testet ulike hobbyer.

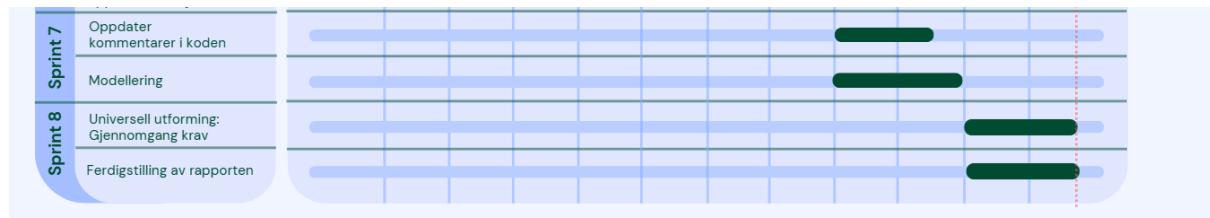
Vedlegg 9: Felttesting

Notater fra bruker	Endringer gjort
Søke skjermen er noe forvirrende, hvor trykker man for å søke?	Knapper på søke skjermen er større, ser mer ut som en knapp og har skriftlig beskrivelse
Prikkene når AI-en “tenker” er litt små	Prikkene når Mr. Praktisk “tenker” er endret fra “.” til “•”
Når Ai-en “skriver” går det litt treigt	Hastigheten på hvor fort Mr. Praktisk “skriver” er økt
Overskrifter burde være større	Overskrifter er forstørret
Når man velger min posisjon bør det stå hvor man er (for eksempel “Min posisjon: Oslo” i stedet for bare “Min posisjon”)	Reverse geocoding api er implementert for å legge til stedsnavn basert på koordinater til min posisjon
Det bør stå retningen vinden blåser fra	Vindretning er lagt til i et kompass
Ai-en genererer tekst ganske tregt og er noe upålitelig	Skyldes dårlig og tregt api fra Azure. Ingen endringer gjort.

Vedlegg 10: Gantt diagram

Prosjektets implementasjon: Gantt Chart





Vedlegg 11: Kanban board

IN2000 prosjektarbeid, team 13						
BACKLOG		TO-DO	IN-PROGRESS	TIL GODKJENNING	DONE	
Utforsk ikoner	Lage aktivitetsdiagram	Favorites Screen	Implementere farevarsel-apier Hvem: parprogrammering	Starte på rapport: "Presentasjon /intro"	Fylle ut milepælsplan: frister	Stinkyfish Barometeret Lage skjelett for WeatherRepository
Enriching the backlog	Utforske push varslere	Sekvensdiagram for api kall	Finne din posisjon i gradie Hvem: Andreas?	Innsiktarbeid: Intervju/ spørreskjema		Sette opp github og API nivå Gjennomføre workshop Hvem: Andreas
Bli kjent med Figma	Sekvensdiagram på mermaid for README.md	Internettforbindelse snackbar	Lage homescreen for MVP (viser været der du er) Hvem: Parprogrammering	Maps API		Mobb programmering Utfores og velge API-er Hvem: Alle
Use case på mermaid for README.md	Maps ViewModel	Maps Screen	Retrospektiv- for påske, tirsdag 19.mars	Skrive om datastrukturer i rapporten. Hvem: Shanza		Arkitekturskiss e og datatflyt Teste vær api-er Hvem: Andreas Fiks gradle herk Hvem: Andreas
			Test ut TDD på tre funksjonalitet- r-alerts	Skriver om spørreundersøkelsen Hvem: Viktoria		Koble opp til ifi-proxy Sette opp filstruktur og arkitektur til app
						Fiks Location api -- finn by/adresse riktig. Uvide known locations Hvem: Andreas
						Teste kart API-er Hvem: Christian

Vedlegg 12: Milepælsplan

