



Høgskulen
på Vestlandet

BACHELOROPPGAVE

PreGO! – En prototype for et kunnskapsbasert verktøy til støtte for klinikere i møte med gravide innvandrerkvinner

PreGO! – A prototype of a knowledge-based tool to support clinicians in dealing with pregnant immigrant women

Tommy Tran

William Pedersen

Dataingeniør og informasjonsteknologi

Fakultet for ingeniør- og naturvitenskap

Veileder: Svein-Ivar Lillehaug

22.05.2023

Jeg bekrefter at arbeidet er selvstendig utarbeidet, og at referanser/kildehenvisninger til alle kilder som er brukt i arbeidet er oppgitt, jf. Forskrift om studium og eksamen ved Høgskulen på Vestlandet, § 10.

TITTELSIDE FOR HOVEDPROSJEKT

<i>Rapportens tittel:</i> PreGO! - En prototype av et kunnskapsbasert verktøy for klinikere i møte med gravide innvandrerkvinner	<i>Dato:</i> 22.05.2023
<i>Forfatter(e):</i> William Pedersen og Tommy Tran	<i>Antall sider u/vedlegg:</i> 46
	<i>Antall sider vedlegg:</i> 65
<i>Studieretning:</i>	<i>Antall disketter/CD-er:</i> 0
<i>Kontaktperson ved studieretning:</i> Carsten Helgesen	<i>Gradering:</i> Ingen
<i>Merknader:</i>	

<i>Oppdragsgiver:</i> HVL	<i>Oppdragsgivers referanse:</i>
<i>Oppdragsgivers kontaktperson:</i> Eline Skirnisdottir Vik og Nilsen Miodini Nilsen	<i>Telefon:</i> 93601611(Eline)

Sammendrag:

Prosjektet handler om å lage en applikasjon som skal gjøre det enklere for klinikere å innhente informasjon om gravide innvandrerkvinner.

Det å tilby tilpasset behandling for en gravid innvandrerkvinn er en stor utfordring i Norge. Denne gruppen er spesielt sårbar innenfor norsk fødselsomsorg med økt risiko for uheldige svangerskapsutfall. Det å redusere de uheldige svangerskapsutfallene blant gravideinnvandrekvinner ville utgjøre en betydelig samfunnstjeneste, særlig med tanken på at denne gruppen utgjør 28% av gravide kvinner i Norge. En av årsakene til at denne gruppen er spesielt sårbar er at klinikere i Norge ikke har nok informasjon til å tilrettelegge en tilpasset behandling for denne gruppen.

Målet med PreGO!-prosjektet er å utvikle en applikasjon som forenkler klinikernes tilgang til informasjon om gravide innvandrerkvinner, slik at de kan være bedre forberedt når de kommer i møte med denne sårbare gruppen og tilbyr en mer tilrettelagt behandling.

Denne rapporten vil omhandle utviklingen av PreGO!-prototypen, som utgjør starten på utviklingen av den endelige PreGO!-applikasjonen. Målet med prototypen er å utvikle en applikasjon som kan evalueres for sin nytteverdi i samfunnet, og som eventuelt kan fungere som inspirasjon for de påfølgende iterasjonene av PreGO!-applikasjonen. PreGO!-prototypen vil presentere data om ulike risikofaktorer, slik at klinikere kan foreta mer grundige vurderinger av gravide innvandrerkvinner helsestatus.

Stikkord:

Prototype	Risikofaktorer	Gravide innvandrerkvinner Svangerskapsomsorg
-----------	----------------	-------------------------------------------------

Høgskulen på Vestlandet, Fakultet for ingeniør- og naturvitenskap

Postadresse: Postboks 7030, 5020 BERGEN Besøksadresse: Inndalsveien 28, Bergen

Tlf. 55 58 75 00

Fax 55 58 77 90

E-post: post@hvl.noHjemmeside: <http://www.hvl.no>

FORORD

Det er med stor glede og stolthet at vi presenterer dette forordet i forbindelse med gjennomføringen av *PreGO!* prosjektet i vårsemesteret 2023. Dette prosjektet har vært utført av to dedikerte informasjonsteknologistudenter, Tommy Tran og William Pedersen, ved Høgskulen på Vestlandet (HVL).

Vi ønsker først og fremst å rette en hjertelig takk til våre oppdragsgivere, Eline og Roy, for å ha gitt oss denne unike muligheten til å utvikle en prototype for *PreGO!* og legge grunnlaget for hva det kan bli. Deres støtte, veiledning og engasjement gjennom hele prosjektperioden har vært uvurderlig. Vi er takknemlige for den tilliten de har vist oss og for den verdifulle innsikten de har delt med oss.

Vi vil også rette en spesiell takk til vår veileder, Svein-Ivar Lillehaug, for hans betydningsfulle bidrag og veiledning gjennom prosjektet. Hans visdom, faglige kompetanse og konstruktive tilbakemeldinger har vært avgjørende for vår faglige utvikling og for å sikre at prosjektet har fulgt riktige spor. Vi er takknemlige for hans dedikasjon og støtte underveis.

Ordliste

Innvandrerkvinner - Kvinner som har kommet til Norge fra et annet land og har opphold her, men som er født utenfor Norge. Begrepet brukes spesifikt i denne rapporten i forbindelse med gravide innvandrerkvinner.

Klinikere - Betegnelse for å representere jordmødre, fastleger og øvrige kommunalt helseansatte som yter helsetjenester i denne rapporten. Dette er et avvik fra den normale betydningen av begrepet «klinikere» som brukes om *leger* som undersøker og behandler pasienter.

Perinatal - Perinatal er den tiden som dekker svangerskapstiden etter 22. uke samt barnets syv første levedøgn.

Risikofaktor - et begrep som i denne oppgavekonteksten brukes til å referere til faktorer eller hendelser som kan oppstå under graviditeten for uønskede utfall eller komplikasjoner hos gravide kvinner under svangerskapet. Dette kan inkludere situasjoner som keisersnitt, lav fødselsvekt, lav folsyre og dødfødsel.

Rammeverk - en samling av kodebiblioteker, verktøy, standarder og retningslinjer som hjelper utviklere å bygge programvare mer effektivt og raskere. Rammeverket gir et grunnlag for utvikling ved å gi en struktur og organisasjon for koden som skal skrives. Det inneholder ofte også forhåndsdefinerte funksjonaliteter og moduler som kan gjenbrukes for å løse vanlige problemer i programvareutvikling.

Machine learning model - En maskinlæringsmodell er en beregningsalgoritme eller et program som er trent på et datasett for å predikere eller ta beslutninger basert på inndata. Modellen lærer å gjenkjenne mønstre og relasjoner i dataene gjennom en optimaliseringsprosess.

NoSQL - En type database som skiller seg fra tradisjonelle relasjonelle databaser som bruker SQL spørrespråket.

Parameter - Refereres til variabler som en machine learning-modell lærer fra i treningsdataene for å gjøre beslutninger.

Frontend - Grafisk brukergrensesnitt til en applikasjon

GBD - The Global Burden of Disease (GBD) er et omfattende forskningsprosjekt som tar sikte på å kvantifisere og sammenligne virkningen av ulike sykdommer og risikofaktorer på befolkningens helse rundt om i verden.

Svangerskapsforgiftning - Svangerskapsforgiftning er når blodtrykket stiger og protein lekker ut i urinen under graviditeten. Det kan føre til farlige problemer for både mor og baby.

Dødfødsel - fødsel av dødt barn etter svangerskapsuke 22

API - Application Programming Interface er et sett med regler og protokoller som tillater ulike programvaresystemer å kommunisere med hverandre.

Open source - Kode som ligger åpent med mulighet for modifiseringer og distribusjon.

Likert-skala - er en vanlig metode for å måle holdninger og meninger der respondentene velger mellom ulike svaralternativer på en skala.

Repository - Koden som ligger på GitHub

INNHOLDSFORTEGNELSE

FORORD.....	1
1 INNLEDNING	2
1.1 KONTEKST.....	2
1.2 MOTIVASJON	2
1.3 PROSJEKTEIER	3
1.4 PROBLEMBESKRIVELSE OG MÅL	3
1.5 OPPBYGGING AV RAPPORTEN	5
2 PROSJEKTBEKRIVELSE.....	6
2.1 PRAKTISK BAKGRUNN	6
2.1.1 Tidligere arbeid.....	6
2.1.2 Initielle krav	6
2.1.3 Initiell løsnings-idé.....	7
2.2 AVGRENSNINGER.....	7
2.3 RESSURSER.....	8
2.4 LITTERATUR OM PROBLEMSTILLINGEN	8
3 DESIGN AV PROSJEKTET.....	10
3.1 VALGT LØSNING	10
3.2 FORSLAG TIL VERKTØY	10
3.2.1 React.....	10
3.2.2 Andre rammeverk.....	11
3.2.3 Machine Learning	11
3.2.4 Firebase	12
3.2.5 SQL database	12
3.3 DISKUSJON AV ALTERNATIVER	13
3.3.1 Machine learning	13
3.3.2 Rammeverk.....	13
3.3.3 Database	13
3.4 VALG AV VERKTØY OG TEKNOLOGI.....	14
3.5 PROSJEKTMETODIKK	15
3.5.1 Utviklingsmetodikk.....	15
3.5.2 Prosjektplan	16
3.5.3 Risikovurdering	17
3.6 EVALUERINGSPLAN.....	17
3.6.1 Enhetstesting.....	17
3.6.2 Kontinuerlig evaluering.....	17

3.6.3	<i>Brukertesting</i>	17
4	DETALJERT LØSNING (ELLER SE FORSLAGENE UNDER)	19
4.1	UTFORMING AV BRUKERGRENSESNITT	19
4.2	BRUKER FUNKSJONER	20
4.3	ADMINISTRATOR FUNKSJONER	27
4.4	DATABEHANDLING	29
5	RESULTATER	30
5.1	EVALUERINGSMETODE	30
5.1.1	<i>Enhetstesting</i>	30
5.1.2	<i>Kontinuerlig evaluering</i>	30
5.1.3	<i>Brukertesting</i>	31
5.2	EVALUERINGSRESULTAT	33
5.2.1	<i>Brukertesting av klinikere</i>	33
5.2.2	<i>Evaluering fra oppdragsgiver</i>	33
5.3	PROSJEKTRESULTAT	34
5.4	PROSJEKTGJENNOMFØRING	35
6	DISKUSJON	37
6.1	UTVIKLING OG EVALUERING AV FUNKSJONER	37
6.2	SVAKHETER TIL PROSJEKTLØSNINGEN	37
6.2.1	<i>Svakheter i koden og potensielle fiks</i>	37
6.2.2	<i>Svakheter i utviklingsmetodikk</i>	39
6.3	STYRKER TIL PROSJEKTLØSNINGEN	39
6.3.1	<i>Styrker i utviklingsverktøy</i>	39
6.3.2	<i>Styrker i utviklingsmetodikk</i>	40
6.4	SAMFUNNSMESSIGE FORDELER OG ETISKE VURDERINGER	40
6.4.1	<i>Samfunnmessige fordeler</i>	40
6.4.2	<i>Etiske vurderinger om tilgang til ekte data</i>	41
7	KONKLUSJON OG VIDERE ARBEID	43
7.1	KONKLUSJON	43
7.2	VIDERE ARBEID	44
7.2.1	<i>Resultatsiden</i>	44
7.2.2	<i>Migrasjonsrelaterte faktorer</i>	44
7.2.3	<i>Administrator funksjoner</i>	45
7.2.4	<i>Database</i>	45
7.2.5	<i>Brukergrensesnitt</i>	45
8	REFERANSER	47
9	VEDLEGG	49

1 INNLEDNING

1.1 Kontekst

I 2021 hadde 28% av gravide kvinner i Norge innvandrerbakgrunn, en andel som har økt fra 16% i 2001. Årsaken til denne økningen er naturlig nok den økte innvandringen til Norge, hvor 15% av befolkningen i 2020(*ssb*, 2020) har innvandrerbakgrunn sammenlignet med 6% i 2001(*ssb*, 2001).

I dag kommer fødende kvinner i Norge fra rundt 200 land, og med ulike utfall av svangerskap. Svangerskapsutfallene kan variere betydelig mellom innvandrergrupper, noe som er vist av perinatale utfall og risikofaktorer som dødfødsel og svangerskapsforgiftning. Noen innvandrergrupper kan ha bedre svangerskapsutfall enn etnisk norske kvinner, mens andre grupper har et dårligere utfall (Vik et al, 2023).

Som et resultat av økt andel av innvandrerkvinner som føder i Norge, vil klinikere nå møte flere gravide innvandrerkvinner med en høy grad av variasjon i svangerskapsutfall, samtidig som de har begrenset med personlig erfaring og kunnskap for å møte disse nye problemstillingene på en mest mulig optimal måte. Dette kan føre til overbehandling eller underbehandling av pasientene (Vik et al, 2023).

1.2 Motivasjon

Dette bachelorprosjektet inngår i starten på et større forskningsprosjekt, *PreGO!*, der prosjektgruppens oppgave er å utvikle en prototype av et kunnskapsbasert verktøy planlagt brukt for å gi beslutningsstøtte i forbindelse med oppfølging av gravide innvandrerkvinner gjennom deres svangerskapsforløp. Formålet er å gjøre det enklere for klinikere å identifisere risikogrupper med behov for tilpasset omsorg, unngå overbehandling eller underbehandling av gravide med innvandrerbakgrunn, og utjevne ulikheter i dagens tilbud ved å tilby best mulig helsehjelp til hver enkelt fødende. Ved å oppgi pasientens fødselsland kan klinikerne hente inn informasjon om ulike risikoer gitt aktuell kvinnes fødeland og så på grunnlag av dette tilpasse oppfølgingen tilsvarende for å sikre best mulig behandlingsresultater. Verktøyet skal altså kunne gi klinikerne verdifull informasjon som denne skal kunne bruke for å gi en mer skreddersydd behandling.

Prototypens formål skal være å hjelpe klinikere i vurderingen av risikoen for gravide innvandrerkvinner i Norge. Etter fullføring av prototypen og prosjektinnlevering vil oppdragsgiver samle inn tilbakemeldinger fra klinikere for å evaluere prototypens funksjonalitet og relevans. Disse tilbakemeldingene vil videre bli brukt som grunnlag for

en videre utvikling av prototypen, med sikte på å forbedre og optimalisere verktøyet for fremtidig bruk. Dersom evalueringen viser at prototypen er gjennomførbar og bærekraftig, vil det bli vurdert å videreutvikle verktøyet og utforske muligheten for fullskala implementering i det kliniske miljøet.

1.3 Prosjekteier

Prosjektet er blitt initiert av Forskergruppen *Reproduktiv helse og helsefremmende arbeid i et livsløpsperspektiv* (REPROHELSEFREM). REPROHELSEFREM er en forskergruppe ved HVL som arbeider for bedre samhandling mellom spesialisthelsetjenesten og den kommunale helsetjenesten, samt for at retningslinjer og resultat fra ny forskning skal gjennomføres. REPROHELSEFREM forsker på mange områder innenfor svangerskap, helse og migrasjon.

Forskningsprosjektet *PreGO!* ledes av Eline Skirsnisdottir Vik. Vik er en erfaren jordmor og postdoktor ved HVL. Hun har forsket på mødre- og spedbarnshelse, med spesiell interesse for migrasjon. Vi vil i tillegg få hjelp av Roy Miodini Nilsen. Nilsen er en statistiker med høy kompetanse innen perinatal epidemiologi. Vik og Nilsen vil gjennom sin erfaring og ekspertise innen problemområdet, bidra med nyttig informasjon og tilbakemeldinger underveis i prosjektet.

1.4 Problembeskrivelse og mål

Gravide innvandrerkvinner kan stå overfor utfordringer med å motta best mulig behandling under svangerskapet. En av de utfordringene som klinikere møter, er å finne tilrettelagt informasjon for å kunne gi best tilpasset oppfølging til gravide innvandrerkvinner, relatert til faktorer som innvandrerbakgrunn og oppholdstid i Norge.

Manglende tilgang på tilstrekkelig informasjon kan føre til at klinikere bruker unødvendig mye tid på å undersøke et betydelig antall risikofaktorer uten tilstrekkelig forhåndskunnskap. Dette kan føre til at høyrisiko farer ikke oppdages og at gravide innvandrerkvinner ikke får den optimale oppfølgingen de trenger under svangerskapet.

Formålet med dette bachelorprosjektet er å utvikle og evaluere en prototype for et kunnskapsbasert verktøy som kan støtte klinikere med å bedre vurdere ulike risikofaktorer for gravide innvandrerkvinner i Norge. Prototypen vil innhente objektiv og systematisk data om risikoer som kan oppstå under svangerskapet for den aktuelle kvinnen, ved at klinikerens angir hjemlandet til den gravide kvinnen. Hypotesen bak *PreGO!* er at et slikt verktøy kan bidra til at klinikerens får et bedre grunnlag for å behandle gravide innvandrerkvinner og avverge eller oppdage høyrisikofaktorer som ellers kan bli oversett.

PreGO!-prosjektet som prosjektgruppen utvikler er for øyeblikket en prototype, og implementeringen av denne vil ansees som vellykket når testbrukere og oppdragsgivere, Nilsen og Vik, gir positiv tilbakemelding på applikasjonens nytteverdi. En vellykket implementering av *PreGO!* kan føre til videreutvikling av en ferdig applikasjon med betydelige samfunnsmessige fordeler, inkludert en reduksjon av fosterets risiko under svangerskapet, bedre svangerskapsforløp for gravide, og økt kompetanse for klinikere innen området.

En viktig forutsetning for applikasjonen er at den skal løse et eksisterende behov uten å skape flere behov. Derfor er det sentrale forskningsspørsmålet i denne rapporten:

“Hvordan kan vi utvikle en prototype av PreGO! som fremtidige brukere vil oppleve som et nyttig verktøy?”

Gjennom bachelorprosjektet vil vi utvikle en prototype av applikasjonen og evaluere dens nytteverdi og relevans ved å teste den på klinikere som arbeider med gravide innvandrerkvinner. Hvis resultatene er positive og applikasjonen oppleves som et nyttig verktøy, vil dette styrke dens mulige verdi for klinikere og bidra til å møte behovene deres på en bedre måte.

1.5 Oppbygging av rapporten

1. **Innledning:** Beskriver konteksten og målet til prosjektet.
2. **Prosjektbeskrivelse:** Her beskrives nærmere bakgrunnen for prosjektet, og evt. tidligere arbeid, resultat, kunnskap og teori som prosjektet skal bruk og evt. bygge videre på. Her presenteres også prosjekteiers sine opprinnelige krav og ideer.
3. **Design av prosjekt:** Beskriver teknologier og metoder som er gjort bruk av under arbeidet med prosjektet. Her vil det også presenteres og diskuteres alternative løsninger.
4. **Detaljert løsning:** Beskriver hvordan resultatene til prosjektet ble oppnådd.
5. **Resultater:** Beskriver hvordan resultatene evalueres og hva slags metoder som ble brukt for evaluering.
6. **Diskusjon:** Drøfting av valgt løsning, positive og negative sider rundt løsningen og hva som kunne blitt gjort annerledes.
7. **Konklusjon og videre arbeid:** Konklusjonen til problemstillingen og hva som kan jobbes videre med.
8. **Referanser:** Referanser som er brukt i rapporten.
9. **Vedlegg:** Diverse dokumenter knyttet til bacheloroppgaven. Dette inkluderer prosjekthåndbok, visjonsdokument, kravdokumentasjon og systemdokumentasjon.

2 PROSJEKTBSKRIVELSE

2.1 Praktisk bakgrunn

2.1.1 Tidligere arbeid

På verdensbasis har det vært en betydelig økning i bruken av graviditetsrelatert informasjon og risikovurderingsverktøy som tar sikte på å forbedre omsorgen og svangerskapsforløpet for gravide kvinner (Vik et al, 2023). Mange av disse nye verktøyene overser de ulike behovene til kvinner med innvandrerbakgrunn. Få verktøy er utviklet for å være sensitive til kulturelle og språklige barrierer, og de eksisterende verktøyene retter seg mot å øke kompetansen til den gravide kvinnen, og ikke for klinikerens.

Vi vil få utlevert fiktivt datamateriale fra Nilsen som simulerer ekte data fra *Medisinsk Fødselsregister og Statistisk sentralbyrå*. Formålet med dette datasettet er å danne et solid grunnlag for utviklingen av systemet som det skal arbeide med.

2.1.2 Initielle krav

Prosjektet er utformet for å imøtekomme kravene til både klinikere og administratorer som skal bruke applikasjonen. For å dekke disse aktørenes behov, er det identifisert to hovedaktører: *Kliniker* som arbeider med gravide innvandrerkvinner, og *administrator* som vil vedlikeholde og oppdatere applikasjonens databaser. Det er utarbeidet følgende krav for å sikre at applikasjonen oppfyller grunnleggende behov:

Kliniker sine krav:

- Søke etter land ved å bruke en intuitiv søkefunksjon som tillater klinikerens å finne og velge ønsket land raskt og enkelt.
- Se graviditetsrelatert informasjon for det valgte landet, inkludert risikofaktorer.
- Gi mulighet for å gruppere land med lav forekomst av risikofaktorer relatert til graviditet ved hjelp av Global Burden of Disease (GBD).

Administrator sine krav:

- Endre grunnleggende informasjon for valgt land
- Laste opp nytt datasett med statistisk datamateriale over gravide innvandrerkvinner i Norge

2.1.3 Initiell løsnings-idé

Webapplikasjon *PreGO!* skal primært utvikles for bruk på datamaskiner. Applikasjonen skal inneholde en startside hvor brukere kan logge inn og velge et land. Videre vil det være en resultatside som viser risikofaktorer basert på land. Disse vurderingene vil bli utarbeidet ved å analysere fiktive data som skal simulere ekte data fra Medisinsk Fødselsregister og Statistisk sentralbyrå.

For å sikre at produktet er tilpasset behovene til klinikere som behandler gravide innvandrerkvinner, har vi valgt å utvikle et brukervennlig grensesnitt som gir klinikere enkel tilgang til relevant grunnlagsinformasjon for hver kvinne. Dette vil bidra til å avdekke risiko og gi en bedre tilpasset oppfølging. Gruppens fokus har vært å utvikle et grensesnitt som er enkelt å bruke selv for klinikere med begrensede dataferdigheter, og som vil spare dem tid ved å presentere all relevant informasjon på en oversiktlig måte.

I tillegg skal det være en administratorfunksjon tilgjengelig, hvor det skal være mulig å logge seg inn som administrator. Administratoren har privilegier til å laste opp ny data og endre informasjonen knyttet til et land etter behov.

2.2 Avgrensninger

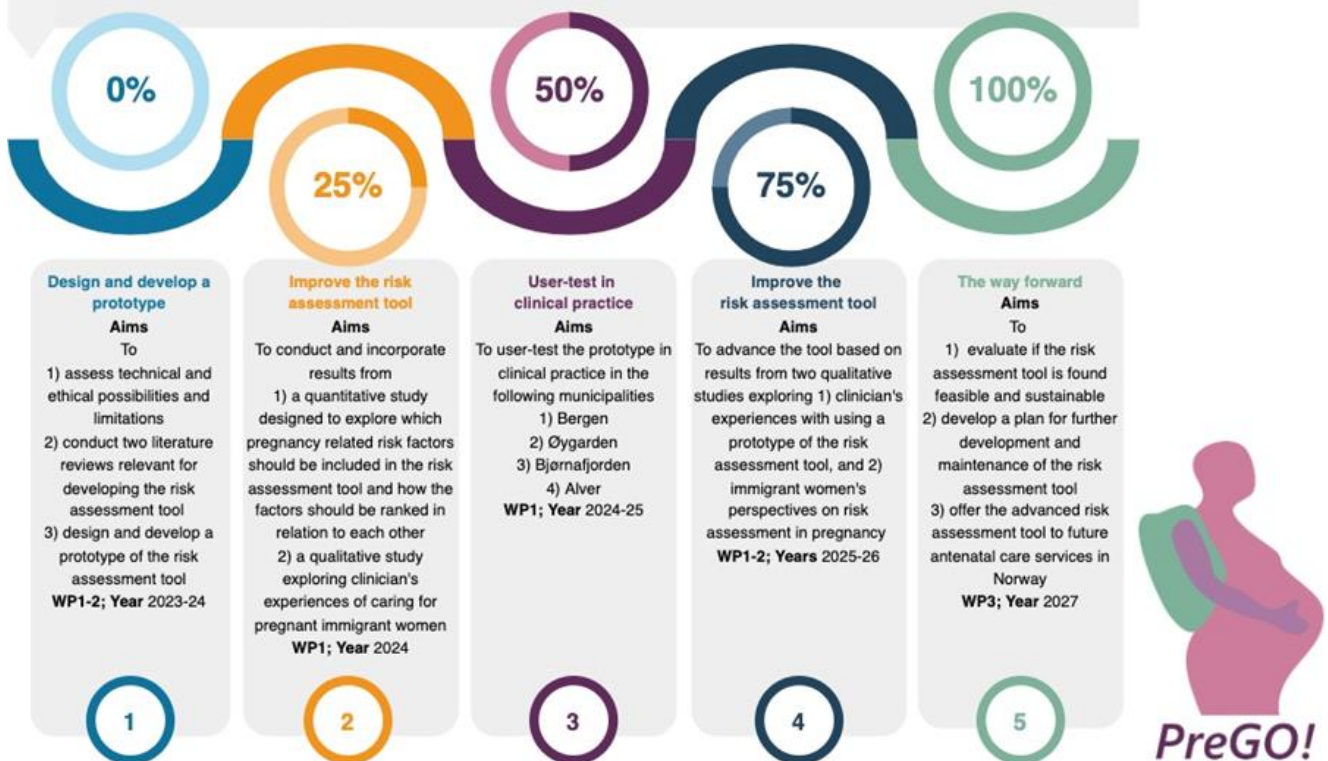
Som nevnt i 2.1.3 Initiell løsnings-idé er *PreGO!* applikasjonen som skal utvikles kun en prototype. Som det illustreres i figur 2.1, er tanken til oppdragsgiver Vik og Nilsen at denne prototypen skal være den første av fem milepæler, som vil tillate videreutvikling av produktet til en fullt fungerende løsning som kan tas i bruk ca. 2027. Det er viktig å påpeke at prototypen har visse begrensninger som vil bli adressert og utbedret i senere faser av utviklingen.

Siden *PreGO!* er en prototype som skal brukertestes av klinikere, vil hovedfokuset være på å utvikle en funksjonell applikasjon fremfor en fullstendig løsning. I utviklingsfasen av prototypen vil sikkerhet bli gitt en lavere prioritet og begrenset oppmerksomhet.

Som beskrevet i avsnitt 2.1.1 om Tidligere arbeid vil prosjektet få tilgang til fiktivt datamateriale fra oppdragsgiver Nilsen, noe som beskrives nærmere i avsnitt 2.3 om ressurser.

Immigrant women in Norway and risk assessment in antenatal care

Aim: to design, develop and test a risk assessment tool



Figur 2.1 - PreGO! milepæl laget av oppdragsgiver Vik

2.3 Ressurser

Vi er gitt tilgang til ressurser som vil bidra til å optimalisere oppgaveløsningen. Disse inkluderer en prosjektsøknad fra Vik (Vik et al, 2023) og aggregerte fiktivt data utlevert av Nilsen som simulerer ekte data. Prosjektsøknaden er utarbeidet av oppdragsgiver Vik, som har søkt og fått godkjent midler fra *Norsk Revmatikerforbund* for å utvikle verktøyet *PreGO!*. Søknaden gir verdifull informasjon og referanser som vil bidra til å øke vår kompetanse innenfor et fagfelt vi ikke er så spesielt kjent med. De fiktive dataene som gruppen har tilgang til inkluderer detaljer om 6 land med 3 100 108 fødsler. Dataene er representert i form av CSV-filer i forskjellige format.

Oppdragsgiver Vik og Nilsen vil også være en viktig ressurs i seg selv. Vik vil hjelpe med tematikken og Nilsen vil støtte med metode.

Veileder Svein-Ivar Lillehaug fra IDER/HVL vil veilede oss på det datafaglige ved å gi oss tilbakemeldinger på statusmøter og innleverte rapporter.

2.4 Litteratur om problemstillingen

For at vi skulle få kunnskap innenfor et fagfelt, som vi i utgangspunktet hadde begrenset kjennskap til, ble det gjennomført en litteraturstudie. Som beskrevet i 2.3 Ressurser, fikk vi tilgang til en prosjektsøknad skrevet av oppdragsgiver Vik som inkluderte omfattende referanser.

En av de undersøkte studiene var "*The Rise of Pregnancy Apps and the Implications for Culturally and Linguistically Diverse Women: Narrative Review*" (Hughson *et al.*, 2018). Denne studien gir en dyptgående analyse av bruken av graviditetsapper blant kvinner, og ga oss innsikt i dagens bruk av slike apper i samfunnet.

3 DESIGN AV PROSJEKTET

Etter å ha diskutert med oppdragsgiverne om hvilke krav de har til prototypen som skal utvikles, har det blitt funnet noen krav. De initielle kravene var veldig løse med tanke på hvordan prototypen skal utvikles og med stor grad av frihet til hvordan prototypen skal utvikles. Det eneste kravene som ble gitt på starten var hva slags funksjonalitet prototypen skulle ha. Det var opptil gruppen og velge løsning, teknologi og verktøy for prototypen.

3.1 Valgt løsning

Ettersom prototypens hovedfunksjon er å hente data om et land fra et datasett som kan oppdateres og brukes av flere samtidig, er det hensiktsmessig å implementere en database-løsning. Dette vil tillate brukerne å hente data fra det samme datasettet og gi administrator mulighet til å oppdatere datasettet. På denne måten kan alle klienter få tilgang til det oppdaterte datasettet så raskt som mulig.

Siden det i vårt prosjekt skal utvikles en prototype som oppdragsgiverne så skal utvurdere for mulig videreutvikling og bruk i henhold til visjon, vil det være mest produktivt å utvikle en prototype som er enkel og egnet for å få tilbakemelding på og deretter tilpasse etter tilbakemeldinger.

3.2 Forslag til verktøy

3.2.1 React

Når en skal lage et brukergrensesnitt er det viktig å tilgang til verktøy som lar en forme grensesnittet slik en ønsker det. React er et kjent rammeverk som gir tilgang til et stort bibliotek til hjelp for å lage en nettside. React er også kjent for å være lett å lære med mye gratis stoff på nettet i form av Youtube videoer og dokumentasjoner.

Det skal også være ganske lett å designe dynamiske webapplikasjoner som er tilpasset til forskjellige enheter. Dette kan være en utfordring å designe med kompleks kode, men React løser dette ved hjelp av JSX - Javascript extension (javapoint, u.å).

React er også et av de meste kjente rammeverkene idag og blir tatt i bruk av store selskaper som blant annet *Facebook*, *Netflix* og *Uber* (Melnyk, 2022).

3.2.2 Andre rammeverk

Det finnes også andre rammeverk enn React som kan bli benyttet til å utvikle et brukergrensesnitt som for eksempel Vue og Angular. Vue er et framework som ligner ganske på React, men er lettere å lære, har mindre bibliotek og bruker også en komponent drevet arkitektur (bacancytech, 2022). Angular er et annet kjent rammeverk og som også bruker komponent drevet arkitektur med lang tids støtte fra Google og brukt av mange store bedrifter som IBM, Forbes og PayPal for utvikling av nettsider, men kan ha en litt større læringskurve enn Vue og React (Saini og Sakshi, 2021).

3.2.3 Machine Learning

Machine learning handler om å bygge et system som lærer fra data, identifiserer mønstre og gir et resultat basert på det. Supervised machine learning bruker merket data eller "labels" som treningsdata med kjent utdata for å finne mønsteret for å lære seg opp til å predikere fremtidige input data. Dette krever en god del data, gode parametere og stor mengde data for å få et mest mulig nøyaktig resultat (Tayo, 2023).

Det finnes veldig mange gode verktøy for å sette opp en machine learning-modell spesielt fra python-biblioteket som gir tilgang numPy, matplotlib og pandas. Disse tre bibliotekene blir brukt til å håndtere dataen før man setter dem inn i en machine learning-modell. Dette inkluderer innhenting, manipulering, visualisering og analysing av data (Micro Focus, u.å).

For selve trening og testing av en machine learning modell vil sickit-learn, tensorflow og pyTorch være nyttig. Disse bibliotekene tilbyr en rekke verktøy for testing, algoritmer og modifiering av modeller (Tayo, 2023).

Pycaret er et annet bibliotek som fungerer fint for machine learning og er lett å ta i bruk. Biblioteket vil automatisere store deler av prosessen for å utforme en machine learning-modell med veldig lite kode.

3.2.4 Firebase

Siden det skal utvikles en prototype for oppdragsgiverne som et artefakt for å undersøke om den endelige applikasjonen til *PreGo!* vil være nyttig for klinikere som muligens skal ta i bruk produktet i fremtiden. Vi har derfor bestemt å prøve å holde koden så simpel som mulig slik at funksjonaliteten til prototypen er i fokus og ikke ha altfor mye komplisert backend kode for å få funksjonalitetene til å fungere som de skal.

Følgelig vurderes det å bruke Firebase sin Firestore Database som database løsning for prototypen. Firestore er en cloud nosql database som tilbyr en lett integrasjon av databasen i koden med en rekke nyttig funksjoner dersom behov for dette under utviklingen av koden til prototypen, noen av funksjonaliteten som Firestore tilbyr er blant annet skalerbarhet, sanntidsoppdatering og sikkerhetsfunksjoner (Firebase, u.å).

3.2.5 SQL database

Den mest kjente type database som blir brukt i dag er SQL. Noen av grunnene til at SQL blir brukt som en database løsning i dag er at den støtter en del kodespråk, med et stort samfunn med mange tilgjengelige ressurser. Det er mange forskjellige type SQL databaser en kan ta i bruk, som eksempelvis Microsoft SQL, MySQL og PostgreSQL. Mange av disse er «open-source» og støtter en rekke forskjellige kodespråk. Det kan være litt krevende å implementere og vedlikeholde en SQL-database, men med et stort samfunn og mye ressurser på nettet vil ikke dette være et problem i det lange løpet.

SQL er en av de mest kjente og brukte typene databaser i dag. Det er flere grunner til at SQL er populært som database-løsning. For det første støtter det en rekke programmeringsspråk, noe som gjør det fleksibelt og kompatibelt med ulike teknologier. Videre har SQL et stort samfunn og et omfattende utvalg av tilgjengelige ressurser, som tutorials, dokumentasjon og forum, som gir støtte og hjelp til brukere. Det finnes også forskjellige typer SQL-databaser å velge mellom, for eksempel Microsoft SQL, MySQL og PostgreSQL, hvor mange av dem er «open-source» og tilbyr ulike funksjonaliteter. Selv om implementering og vedlikehold av en SQL-database kan være utfordrende, er tilgjengeligheten av et aktivt samfunn og omfattende nettbaserte ressurser en fordel som hjelper brukere med å håndtere eventuelle problemer og oppnå suksess på lang sikt (Chand, 2022).

3.3 Diskusjon av alternativer

3.3.1 Machine learning

Ettersom vi har fått utlevert en aggregert, fiktiv versjon av dataen som simulerer den ekte dataen fra Medisinsk Fødselsregister og Statistisk sentralbyrå, er bruk av machine learning lite relevant. Machine learning-modeller krever tilgang til store mengder data og et mangfold av parametere. Dette begrenser mulighetene for machine learning-modellen til å oppdage mønstre eller skape meningsfulle resultater.

Det kan være mulig å lage en machine learning-modell til en av de senere versjonene i *PreGO!* prosjektet som muligens har tilgang til flere parametere og større mengder data. Dette kan hjelpe klinikere å få mer informasjon om pasienten som et individ istedenfor å hovedsakelig gruppere dem etter land.

3.3.2 Rammeverk

Det finnes mange ulike rammeverk å velge mellom med ulike verktøy, svakheter og styrker. Derfor er det viktig å velge et rammeverk som er egnet for utviklingen av prototypen.

Vi har fått mye frihet av oppdragsgiverne når det gjelder valg av verktøy for å utvikle prototypen. Med lite samlet kunnskap om utvikling når det gjelder utvikling av å bruke grensesnitt, samt lite kjennskap til de forskjellige rammeverkene vi har undersøkt, er det viktig at det er mange tilgjengelige ressurser som kan hjelpe oss å lære rammeverket vi velger å ta i bruk for å oppnå det ønskede resultatet til oppdragsgiverne.

Etter å ha undersøkt de tre forskjellige rammeverkene gruppen vurderte å bruke, er det React som virker mest egnet for dette prosjektet. Den viktigste grunnen til at vi lener mest mot React er tilgangen til ressurser for å lære rammeverket og at det er det vi kjenner mest til.

3.3.3 Database

For prototypen tenker vi at det vil være best å ha en database som trenger minst mulig vedlikehold, oppsett, lettest integrasjon i koden og god sikkerhet. Ved bruk av React vil Firebase være et veldig solid valg med lett oppsett, lite vedlikehold, innebygd sikkerhetsfunksjoner og et godt gratis tilbud for små prosjekter. For Vue og Angular vil det kreve litt mer å integrere Firebase i koden. Ulempen med Firebase er at man ikke har noe annet valg enn å benytte av Google sine databaser, og at Firebase ikke er veldig fleksibel med hva slags kodespråk som blir støttet.

SQL er den meste brukte databasen og blir tilbudt av mange store bedrifter og det finnes til og med «open-source» SQL databaser og støtter av mange forskjellige typer kodespråk. Ulempen med SQL er at det kan være krevende å sette opp en SQL-database, krevende å integrere i koden og vedlikeholde sammenlignet med Firebase (Chand, u.å).

Hvis vi velger å benytte React som rammeverk for prototypen vil Firebase være det databaseverktøyet som blir mest sannsynlig brukt ettersom Firebase har flere av de kvalitetene vi ser etter. Dersom vi hadde valgt et annet kodespråk som Java eller C ville det ha vært en større diskusjon mellom Firebase og SQL ettersom noen SQL-databaser kan være lettere å integrere i C eller Java.

3.4 Valg av verktøy og teknologi

Oppdragsgiverne ga ikke noe spesielle krav for bruk av teknologi og verktøy for utviklingen av prototypen. Med lite erfaring i frontend utvikling var det viktig å velge teknologi som var lett å lære, med tilgang til mange ressurser på nettet. Vi bestemte oss derfor for å bruke React som rammeverk for prototypen. Ved bruk av React er det mulig å enten kode i TypeScript eller JavaScript, vi valgte å kode i JavaScript siden det er det kodespråket vi er mest kjent med.

Visual Studio Code er det integrerte utviklingsmiljø (IDE) vi har valgt som kodeeditor for utvikling av prototypen ettersom vi har mye erfaring med editoren og det er vår foretrukne editor for koding.

For at vi skal kunne samarbeide om koden benytter vi oss av GitHub som et samarbeidsverktøy. Dette gjør det mulig for oss å utvikle innen samme prosjekt på hver vår pc ved å ta i bruk push og pull funksjonene som GitHub tilbyr. Push gjør det mulig for et individuelt gruppelem å gjøre endringer på repositoryen mens pull gjør det mulig å hente nye endringer fra repositoryen og jobbe videre på koden. Google Docs er et annet verktøy vi benytter for å øke samarbeidet når det gjelder rapportskrivning. Google Docs gjør det mulig for oss å skrive på rapporten og dokumenter samtidig med sanntidsoppdatering.

For å lage en skisse av brukergrensesnittet vil vi benytte oss av Figma, slik at vi kan få raske tilbakemeldinger fra oppdragsgiverne om brukergrensesnittet.

Som enhetstester for prototypen vil vi gjøre bruk av Vitest, som tilbyr et veldig lett oppsett for enhetstesting.

Som kommunikasjonsplattform vil gruppen ta i bruk Discord for å kommunisere under arbeid

3.5 Prosjektmetodikk

3.5.1 Utviklingsmetodikk

Vi har valgt å jobbe iterativt slik at vi kan tilpasse oss til endringer og problemer underveis. Arbeidet vil bli delt opp i mindre oppgaver som skal bli fullført i løpet av en iterasjon. Underveis vil vi ha møter med oppdragsgiver og veileder for å få svar på spørsmål og tilbakemeldinger til arbeidet som har blitt gjort. Dette vil bidra til en smidig arbeidsprosess der vi kan kontinuerlig få tilbakemeldinger og tilpasse oss etter problemer som kan oppstå underveis.

Starten av utviklingsarbeidet gikk ut på å skaffe oversikt over alt arbeidet som skulle gjøres og dele det opp i iterasjoner. Vi valgte å dele utviklingen i iterasjoner etter de forskjellige statusmøtene som har blitt satt opp på forhånd med oppdragsgiver. For hver iterasjon vil gruppen ha et kort møte der arbeidsoppgaver blir delt opp. Siden det bare er to personer i gruppen, er det enkelt å kommunisere og endre arbeidsoppgaver etter behov.

Når vi starter på utviklingen av prototypen, vil de presentere den for oppdragsgiverne og veileder annenhver uke for å få tilbakemeldinger. Dette vil tillate gruppen å gjøre endringer underveis og ha en smidig utvikling av prototypen. Gruppen kunne ha hatt flere møter for å få flere tilbakemeldinger og innspill fra oppdragsgiverne for å optimalisere prototypen enda mer.

Utviklingsarbeidet av prototypen deles inn i sider og funksjoner prototypen skal ha etter oppdragsgiver Vik sine krav for applikasjonen. Etter noen møter ble det også enighet om å legge til en logg inn side, noen administrasjonsfunksjoner og grafer som representerer risiko data fra tidligere år. Oppgavene i gruppen ble fordelt etter sider og funksjonalitetene prototypen skulle ha.

Vi ble enig med veilederen å prøve å ha statusmøter annenhver uke på fredager i løpet av prosjektperioden. Under disse møtene vil alt som angår prosjektet kunne bli tatt opp, som tips til skriving av dokumenter og rapport, tilbakemelding på prototypen og annet arbeid i bacheloroppgaven. De ukene det ikke var møte med veileder ble brukt til å ha møter med oppdragsgiver der veileder også deltok. Disse møtene ble hovedsakelig brukt til å forsikre at gruppens arbeid var på linje med oppgaven oppdragsgiverne hadde gitt. Gruppen hadde også mulighet til å sende spørsmål gjennom mail til oppdragsgiverne og veilederen om det skulle være behov for det.

Gruppen kunne ha benyttet seg av en spesifikk utviklingsmetode som Scrum, noe gruppen har tatt inspirasjon av for utviklingsmetodikken de tok i bruk. Scrum er et rammeverk innenfor prosjektledelse som ofte brukes i programvareutvikling. Scrum tar i

bruk sprinter eller iterasjoner. For hver sprint blir bestemte oppgaver utført, og på slutten av hver sprint blir det holdt en sprint evaluering der en oppsummering og evaluering av sprinten blir gjennomført. Arbeidsgruppen og andre interessenter i prosjektet deltar på dette møtet, slik at gruppen kan få kontinuerlige tilbakemeldinger underveis i prosjektet (scrum, u.å).

Grunnen til at gruppen ikke har benyttet seg av Scrum som utviklingsmetode, men heller brukt Scrum som inspirasjon er fordi gruppen består av to gruppemedlemmer og ikke har behov for roller som Scrum master og behovet for daglig møter er heller ikke nødvendig siden gruppen kan kommunisere lett med hverandre underveis i arbeidet.

Gruppemedlemmene vil heller ta et felles ansvar for at arbeidet for hver iterasjon blir gjennomført.

3.5.2 Prosjektplan

Gruppen har utviklet et Gantt-diagram for å planlegge fremdrift av prosjektet. Gantt-diagrammet viser planlagt fremdrift for utviklingen av prototypen, rapportskrivning og dokumentering av bachelorprosjektet. Planlagt fremdrift vil kunne bli endret underveis ettersom gruppen går gjennom iterasjoner. Den nyeste versjonen av Gantt-diagrammet vil være tilgjengelig i prosjekthåndboken til prosjektet (Se vedlegg 9.1). Gantt-diagrammet blir brukt for å gi gruppen en oversikt over hvilke arbeidsoppgaver som skal jobbes med og når oppgavene skal være ferdig. Grafen vil også vise en forventet lengde for hver fase og iterasjon.

Gantt diagrammet består av følgende faser:

- **Oppstart:** Dette er den første fasen i prosjektet og innebærer at gruppen får kontakt med oppdragsgiver og veileder, samtidig få en forståelse av oppgaven og problemstillingen.
- **Prosjektplanlegging:** Planlegge utviklingen av prototypen, dette innebærer datainnhenting, design, planlegging for bruk av teknologier og verktøy. Samtidig som iterasjon 2 av rapportskrivningen og dokumentering starter.
- **Utvikling:** Her vil hovedfokuset være å utvikle prototypen med rapportskrivning og dokumentering som handler om utviklingen av prototypen.
- **Hovedrapport og fullføring av støttedokumenter:** Evaluering av prototypen samt dokumentering. Skrive ferdig hovedrapport og oppdatere støtte dokumenter.

De forskjellige fasene kan gå litt i hverandre. De forskjellige aktivitetene for hver fase vil være spesifisert i prosjekthåndboken (se vedlegg 9.1) i kapittel 1.

3.5.3 Risikovurdering

Risikovurdering er en avgjørende prosess for et vellykket gjennomført prosjekt. I gruppens bachelorprosjekt vil risikovurdering, som er inkludert i prosjekthåndboken (se vedlegg 9.1) som et vedlegg, hjelpe oss å identifisere og evaluere potensielle risikoer og farer som kan påvirke prosjektet på en negativ måte.

Dette kan inkludere teknologiske problemer, kommunikasjonsproblemer eller økonomiske utfordringer. Gjennom risikovurdering vil gruppen kunne evaluere sannsynligheten for at en risiko inntreffer og konsekvensene dette vil ha på prosjektet. Deretter kan gruppen utvikle en plan for å håndtere risikoen, slik at gruppen kan minimere eller eliminere truslene for prosjektets suksess. Ved å bruke risikovurdering vil gruppen kunne planlegge forventede utfordringer og ta hensyn til uforutsette hendelser som kan påvirke prosjektets progresjon. Dette vil hjelpe gruppen å sikre at prosjektet går som planlagt og at gruppen når målet innenfor tidsrammen som er satt. Risikovurdering er derfor en viktig prosess for å sikre at prosjektet har en høy sannsynlighet for suksess. Med en grundig risikovurdering kan gruppen redusere sjansene for uforutsette problemer og øke evnen til å håndtere dem hvis de oppstår.

3.6 Evalueringsplan

3.6.1 Enhetstesting

Enhetstesting vil bli brukt for å teste applikasjonens logikk og sikre at den fungerer som forventet. Men, det kan ta mye tid og være en repeterende oppgave å lage tester for hver funksjon.

3.6.2 Kontinuerlig evaluering

Gruppen vil implementere kontinuerlig evaluering i prosjektet, ved å bruke flere evalueringsteknikker for å sikre høy kvalitet på applikasjonen. Gruppen vil bruke Scrum-metodikken og ha Vik og Nilsen, gruppens oppdragsgivere, sammen med veilederen Svein i statusmøter, der funksjonaliteten til applikasjonen vil bli presentert og demonstrert. Tilbakemeldinger fra oppdragsgiverne og veilederen vil gi verdifulle innspill til forbedring og optimalisering av applikasjonen.

3.6.3 Brukertesting

Planlagt brukertesting vil involvere oppdragsgiverne Vik og Nilsen, som vil evaluere gruppenes applikasjon og gi tilbakemeldinger på funksjonalitet og brukervennlighet. En vellykket evaluering fra oppdragsgiverne vil kunne føre til videre distribusjon av applikasjonen til klinikere. Dette vil igjen gjøre det mulig å gjennomføre en grundigere

brukertesting og intervju med ekte klinikere, der de vil få mulighet til å teste applikasjonen og gi tilbakemeldinger på brukervennlighet og funksjonalitet.

4 DETALJERT LØSNING (eller se forslagene under)

I dette kapitlet vil gruppen presentere en grundig beskrivelse av prototypen *PreGO!* og dens viktigste funksjoner. Vi vil gi en detaljert oversikt over hvordan dataene blir representert, samt hvordan selve databasen fungerer. I tillegg vil vi også gå gjennom noen administrative funksjoner som er tilgjengelige i *PreGO!*. Samlet vil denne gjennomgangen gi en dypere forståelse av *PreGO!* og dens funksjoner.

4.1 Utforming av brukergrensesnitt

Brukergrensesnittet til gruppens prototype skal være brukervennlig og intuitivt, med et fokus på å utvikle en fungerende prototype. Vi erkjenner betydningen av et godt brukergrensesnitt for å skape en positiv opplevelse for brukerne. Derfor vil vi implementere en enkel og tydelig navigasjonsstruktur, slik at brukerne enkelt kan finne nødvendig informasjon og bruke de ulike funksjonene.

Selv om hovedfokuset er på å sikre brukervennlighet, vil gruppen prioritere utviklingen av en prototype med fungerende logikk og kjernefunksjonalitet. Grundig testing og feilretting vil være viktig for å sikre sømløs funksjonalitet og pålitelige resultater. Dette vil sikre at prototypen gir brukerne en pålitelig og nyttig opplevelse.

Gruppen vil også ta hensyn til designelementer som tydelig tekst, riktig bruk av farger og kontraster for å forbedre lesbarhet og visuell appell. Knapper og interaksjonselementer vil være gjenkjennelige og enkle å bruke. Ved å følge grunnleggende prinsipper for god brukergrensesnittutforming, som konsistens og tilbakemelding på handlinger, vil prosjektgruppen skape en positiv og intuitiv opplevelse for brukerne.

Selv om funksjonalitet og logikk prioriteres, vil gruppen fortsatt vurdere tilbakemeldinger og innspill fra oppdragsgivere og veileder gjennom iterative statusmøter for å forbedre brukergrensesnittet over tid. Gruppen vil være åpen for justeringer og forbedringer basert på oppdragsgiver og veileder sine tilbakemeldinger og behov, slik at prototypen blir stadig mer brukervennlig og tilpasset målgruppens krav.

Gjennom en kombinasjon av funksjonalitet, brukervennlighet og kontinuerlig forbedring, vil prosjektgruppen skape en prototyp som ikke bare fungerer pålitelig, men også gir en god brukeropplevelse for klinikere som benytter verktøyet i møte med gravide innvandrerkvinner.

4.2 Bruker funksjoner

4.2.1 Land-velger

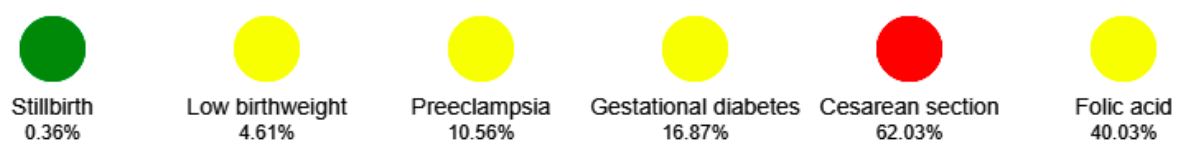
Når brukeren logger seg inn, vil land-velgeren være det første elementet de ser. Land-velgeren gir brukeren muligheten til å velge sitt eget land ved å skrive inn navnet manuelt eller ved å velge et land fra en alfabetisk sortert liste. Etter at brukeren har valgt et land og trykket på Enter-knappen i skrivefeltet, blir de videreført til en resultatside for det valgte landet.



Figur 4.1 - Bilde av land-velger funksjonen (Skjerm bilde ble tatt i mørk modus for å skille det mer ut i rapporten)

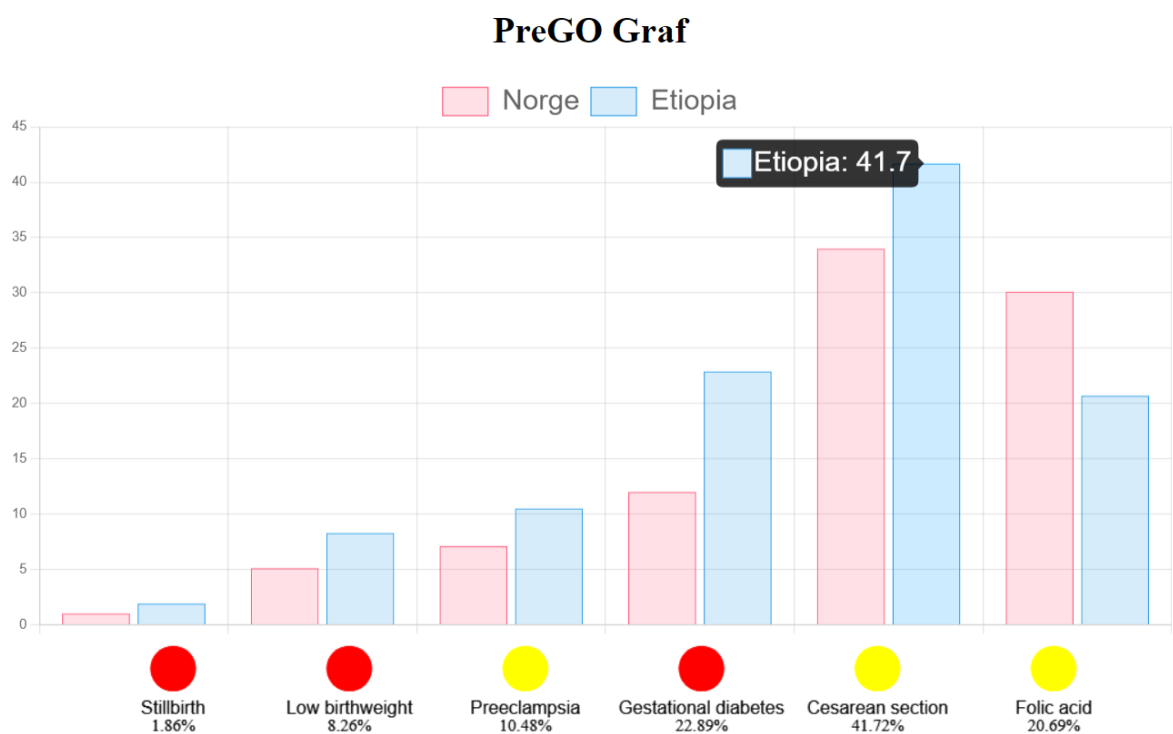
4.2.2 Resultatside

Etter at brukeren har valgt et land i land-velgeren (se pkt. 4.2.1), vil brukeren bli videreført til en resultatside som presenterer dataene fra det valgte landet. På denne siden vil det være seks sirkler som representerer ulike risikoer som kan oppstå hos gravide kvinner. Fargen til sirklene vil indikere hvorvidt risikoen er bedre (grønn), omtrent lik (gul) eller verre (rød) sammenlignet med Norge. Disse sirklene ligger nederst på resultatsiden.



Figur 4.2 - Risikofaktor sirkler fra på resultatsiden

Over de seks sirklene vil det bli presentert en graf som presenterer den samme data som sirklene på en annen måte. Hensikten med grafen er å gi en mer presis sammenligning av risikofaktorene i det valgte landet med Norge, og dermed øke forståelsen av eventuelle forskjeller. Mens sirklene gir en visuell indikasjon på risikonivået, kan grafen gi en mer detaljert presentasjon av tallene. Grafen vil ha to linjer, en blå linje som representerer data for det valgte landet fra land-velgeren, og en rød linje som representerer data for Norge. Brukeren vil ha mulighet til å se nøyaktige tallverdier ved å føre musepekeren over grafen.



Figur 4.3 - Stolpe graf på resultatsiden

Dersom brukeren trykker på en av de nevnte sirklene, vil det framkomme en boks som inneholder en grafisk representasjon av dataene for den aktuelle risikofaktoren. Grafen

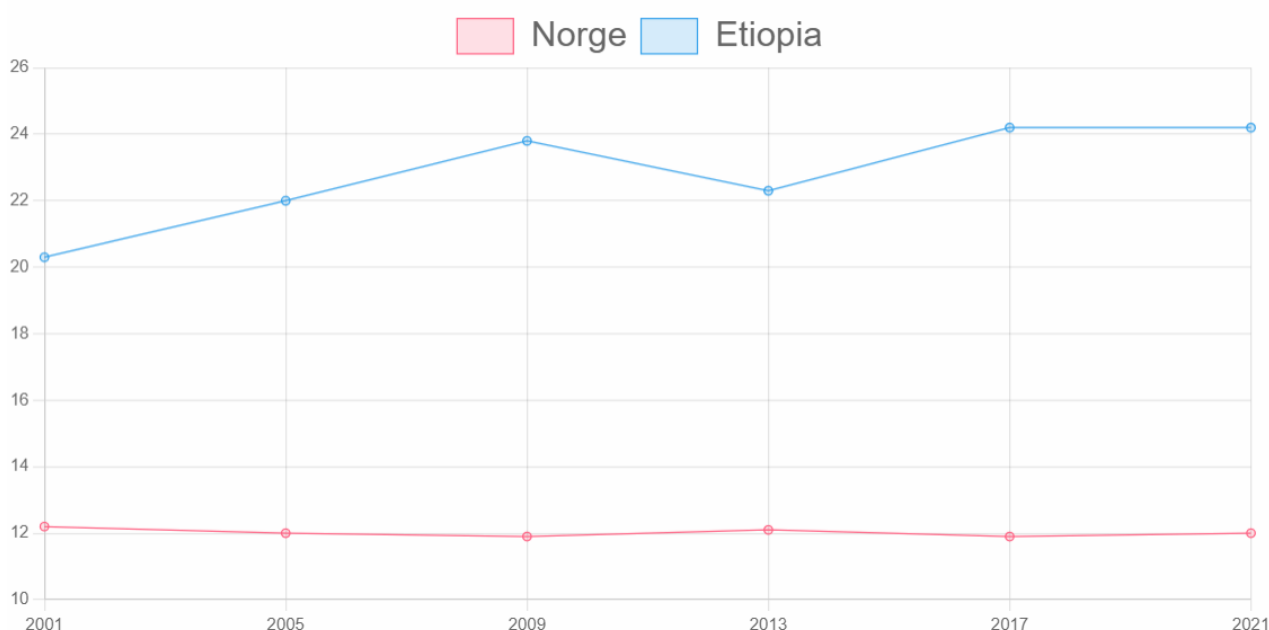
vil gi en visuell fremstilling av endringer i dataene for den valgte risikofaktoren mellom perioden 2001 til 2021 for både det valgte landet og Norge. Som et eksempel, dersom brukeren trykker på sirkelen for Preeklampsi (svangerskapsforgiftning), vil grafen vise endringer i dataene for denne risikofaktoren for begge landene over tidsperioden nevnt.

Merk at i en fullstendig versjon vil det være inkludert fakta om risikoen og tiltak som kan gjøres. Imidlertid, ettersom dette er utenfor gruppens fagfelt og gruppens oppgave er bare å presentere konseptet, har gruppen valgt å la en placeholder være i boksene.

Gestational diabetes



Fakta om Gestational diabetes?



Figur 4.4 - linjediagram på resultatsiden

Øverst på siden står det 3 punkter, Inntektsgruppe, GBD (global burden of disease) og risikoscore. GBD bidrar til å gi en bedre oversikt over risikofaktorene i land med begrenset data, slik at tiltak kan settes inn for å forbedre helsen i disse landene. I gruppens verktøy vil det komme opp en rød skrift på toppen som sier om GBD blir brukt hvis du har valgt et land som har lite data. Risikoscore er en numerisk verdi som beregnes ved å kombinere ulike risikofaktorer som kan påvirke helsen til gravide kvinner. Denne summen vil bli tildelt hvert enkelt land, og landene vil deretter bli rangert etter risikoscore fra høyest risikoscore til lavest. Rangeringen av landene gir en indikasjon på alvorligheten og sannsynligheten for at uønskede utfall oppstår hos gravide kvinner.

4.2.3 risikoscore beregning

Risikoscoren for et land beregnes ved å ta alle risikofaktorer som påvirker helsen til gravide kvinner, og rangere dem fra lavest til høyest sannsynlighet for at risikoen oppstår i hvert land. Den eneste faktoren som vurderes annerledes er «folic acid» (folsyre), hvor høy sannsynlighet for folsyremangel vil bli rangert øverst på grunn av den positive effekten av tilstrekkelig folsyre. Når landene er rangert etter risikoscore, blir deres plassering i ranglisten multiplisert med et tall som er gitt av oppdragsgiver Nilsen. Dette tallet er justert etter hvor alvorlig risikofaktoren er; for eksempel vil «stillbirth» (dødfødsel) vanligvis bli vurdert som mer kritisk enn «Ceaseran section»(keisersnitt). Denne metoden for beregning av risikoscore er en uvitenskapelig måte å kalkulere risikoscoren på, men blir brukt for å demonstrere konseptet for risikoscoren i prototypen og vil mest sannsynlig endre seg i den fullstendige versjonen av *PreGO!* applikasjonen.

Nedenfor i tabell 2 er tallene for alvorlighetsgrad gitt til hver enkelt risikofaktor. Poenget er da at hvis vi eksempelvis har 7 land på prototypen og antallet «stillbirth» i Etiopia er høyest av alle landene, da vil det bli rangert nederst og få tallet 7. Det rangerte tallet 7 vil da bli ganget sammen med alvorlighetsgraden til «stillbirth» som er 12. Den summen vil da bli lagret og samme prosedyre vil bli gjort på resten av risikofaktorene, utenom «folic acid» der rangeringstallet vil være omvendt siden det er positivt for en gravid kvinne å ha en høy folsyre mengde.

Merk at prosjektgruppens beregning er ikke en vitenskapelig beregning gitt av oppdragsgiver Nilsen, men er heller en måte å demonstrere konseptet til risikoscore i prototypen.

Tabell 2 - Alvorlighetsgrad til risikoer

Risikofaktor	Alvorlighetsgrad
Stillbirth	12
Low birthweight	3
preeclampsia	4
gestational diabetes	2
caesarean section	2
Folic acid	1

Risiko score:

157 rangert som 1 av 7 i Norge

Figur 4.5 - Presentering av risikoscore i prototypen (Skjerm bilde ble tatt i mørk modus for å skille det mer ut i rapporten)

4.2.4 Andre migrasjonsrelaterte faktorer

Siden "Andre migrasjonsrelaterte faktorer" er en side i verktøyet der klinikerer har muligheten til å undersøke ulike migrasjonsrelaterte faktorer hos pasienten, for eksempel årsaken til deres migrasjon til Norge. Basert på pasientens svar på disse spørsmålene om migrasjonsrelaterte faktorer, skal siden kunne indikere hvordan disse faktorene vil påvirke resultatet for risikofaktorene. Selv om denne siden ikke er nødvendig for klinikerer å bruke i alle scenarier, vil den være tilgjengelig i tilfelle klinikerer ønsker å sjekke om migrasjonsrelaterte faktorer spiller en viktig rolle for risikofaktorene. Gruppen har utviklet denne siden med 12 spørsmål der klinikerer kan vurdere pasientens svar som grønt, gult eller rødt. Grønt indikerer et positivt svar på spørsmålet, gult indikerer nøytralt, mens rødt indikerer et negativt svar. På slutten av siden vil det vises hvordan migrasjonsrelaterte faktorer påvirker risikofaktorene ved hjelp av grønt, gult eller rødt. Grønt indikerer at migrasjonsrelaterte faktorer kan ha en positiv innvirkning på risikofaktorene, gult betyr at innvirkningen er nøytral, og rødt betyr at innvirkningen er negativ.

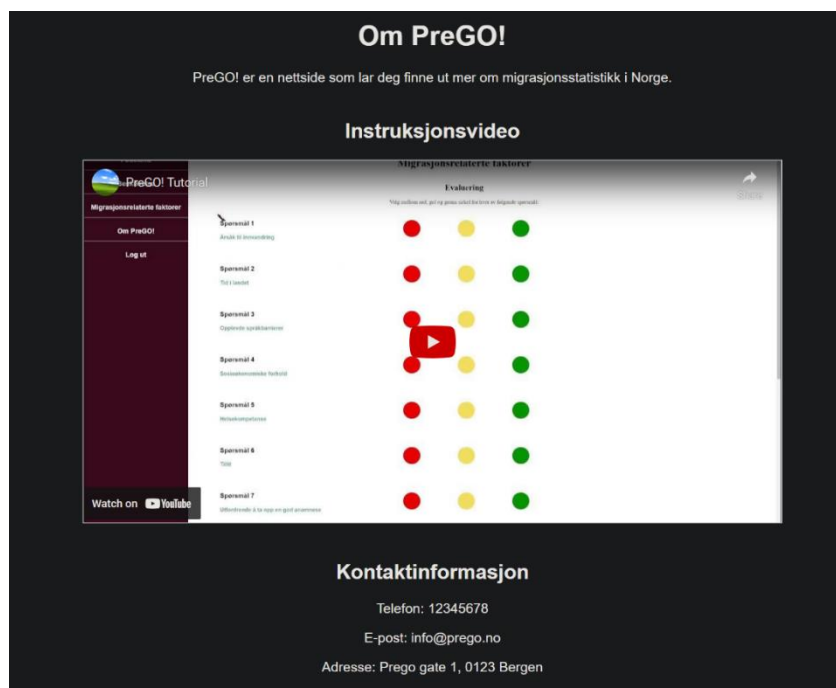


Figur 4.6 - Viser Andre migrasjonsrelaterte faktorer til prototypen (Skjerm bilde ble tatt i dark mode for å skille det mer ut i rapporten)

Denne siden hadde lav prioritet hos gruppen og oppdragsgiverne, derfor er logikken veldig enkel, men den demonstrerer konseptet som oppdragsgiverne hadde tenkt. En av logikkene som gruppen ikke har implementert, er forskjellig vekting av de ulike migrasjonsrelaterte faktorene. Dette skyldes utfordringer med å beregne hvor mye hver faktor skal veie opp mot de andre.

4.2.5 Informasjonsside

Her vil det stå generelt om *PreGO!* prosjektet og kontaktinformasjon til de som har ansvaret for prosjektet i fremtiden, men for nå har gruppen noen dummy tekster. Det vil også være en kort instruksjonsvideo som viser hvordan man skal bruke prototypen. Videoen er ment å være en instruksjonsvideo for brukere og vil derfor ikke vise administrator funksjonene. Videoen ligger vedlagt (se vedlegg 9.6).



Figur 4.7 - Viser informasjonsside side med fungerende video (Skjerm bilde ble tatt i mørk modus for å skille det mer ut i rapporten)

4.2.6 Land beskrivelse

I land beskrivelse vil det stå relevant informasjon som kan hjelpe klinikere i å blant annet forstå situasjonen i landet og helsepraksis i landet. Denne funksjonen var originalt i resultatsiden, men ble flyttet bort etter tilbakemelding fra oppdragsgiver siden det ikke skal være behov for denne funksjonen i alle situasjoner for klinikere.

I prototypen har det blitt brukt generell informasjon om landet for å demonstrere dette konseptet siden det kan være litt for komplekst ettersom vi har lite kompetanse i dette feltet og dømme hva slags informasjon som kan være relevant for klinikere.

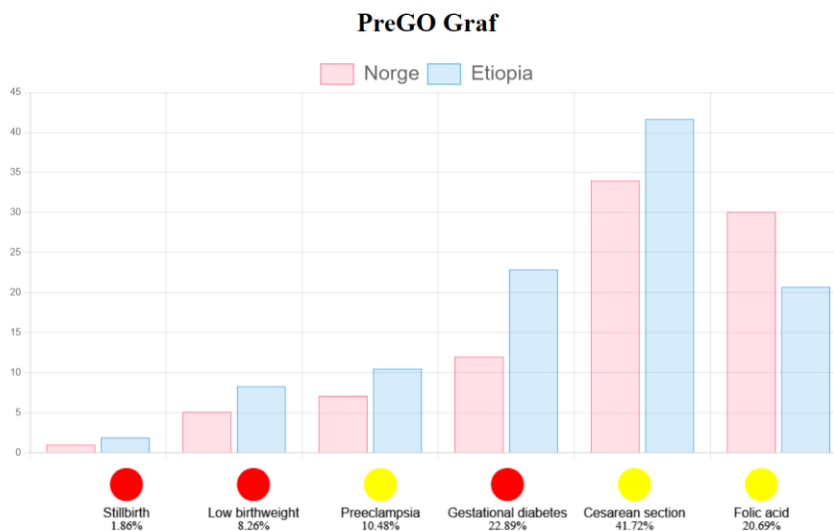
4.2.7 Navigasjonsmeny

Som sett på Figur 4.8 nedenfor, er navigasjonsmenyen en vertikal stripe som finnes på venstre side av hver side i verktøyet. Denne navigasjonsmenyen er et viktig element i brukergrensesnittet, og formålet er å gjøre det enkelt for klinikerne å navigere rundt i verktøyet.

Fødeland
Beskrivelse
Migrasjonsrelaterte faktorer
Om PreGO!
Log ut

Etiopia

Inntektsgruppe: Lav
 GBD: Sub-Sahara afrika
 Risiko score: 179 rangert som 1 av 8 i Norge



Figur 4.8 - Resultatsiden med Navigasjonsmenyen til Venstre

4.3 Administrator funksjoner

En av de initielle forslagene til prosjektet var en administrator bruker med tilgang til spesielle funksjoner. Kravene til administrator funksjonen var at det skulle være mulig for en administrator å oppdatere eller legge til nye datasett i databasen. Det var opp til oss å bestemme hvordan disse funksjonene skulle implementeres.

4.3.1 Innlogging

Ved brukerens eller administrators første tilgang til *PreGo!* prototypen vil logg inn-siden være det første som møter dem. Hovedformålet med innloggingssiden til *PreGo!* prototypen er å autentisere brukeren og differensiere mellom vanlige brukere og administratorer. Dette oppnås ved å bruke Firebase Auth som autentisering.

Firebase Auth gir enkel integrasjon med mange innebygde verktøy, inkludert bransjestandard sikkerhetsprotokoller for sikker autorisering, beskyttelse av brukerdata, passord kryptering samt funksjonaliteter for brukeradministrasjon som registrering, håndtering av brukerprofiler og autentisering basert på tokens (Firebase, u.å).

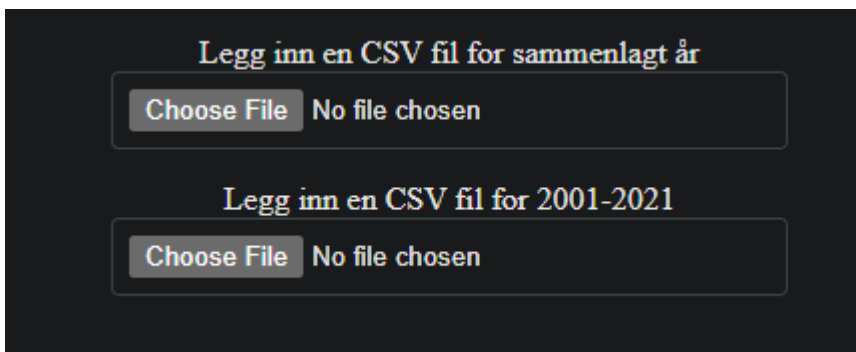
Med Firebase Auth var det lett å skille brukere fra hverandre ved å hente data om den nåværende bruker som er logget inn ved hjelp av Firebase Auth sine innebygde metoder som `auth.currentUser` og `user.uid`. Dette gjorde det lett for oss å skille de forskjellige

brukerne fra hverandre og gi brukeren forskjellige funksjoner basert på hvem brukeren er logget inn som.

4.3.2 laste opp ny eller endre data

Muligheten til å laste opp ny eller oppdatere data i databasen var den viktigste funksjonen som skulle være tilgjengelig for Administrator-brukeren. Vi valgte å implementere denne funksjonaliteten ved å benytte to ulike tilnærminger. Den første metoden involverer opplasting av en CSV-fil til prototypen, som inneholder ny data som skal legges til eller oppdateres i databasen. Den andre metoden er å foreta direkte endringer i prototypen for å oppdatere dataene.

Opplasting av CSV-fil ble delt opp i to forskjellige funksjoner, en for generell data som samler alt data om et land fra 2001 - 2021 og en annen funksjon der dataene fra landet er delt opp i årene fra 2001 - 2021. Se delkapittel 4.4 for mer informasjon om CSV-filene.



Figur 4.8 - Viser de to forskjellige måtene å laste opp CSV-fil til databasen på (Skjerm bilde ble tatt i mørk modus for å skille det mer ut i rapporten)

Endringer som blir gjort direkte på prototypen vil være veldig begrenset til å kun være mulig å oppdatere inntektsgruppen og GBD til valgt land.



Figur 4.9 - Viser direkte endringer av data i prototypen (Skjerm bilde ble tatt i mørk modus for å skille det mer ut i rapporten)

4.4 Databehandling

I delkapittel 4.3.2 var beskrevet hvordan en administrator bruker, laster opp nye data eller endrer data i databasen. Her vil prosessen bli beskrevet, som det ble nevnt i delkapittel 4.3.2 bruker prototypen hovedsakelig to type CSV formater en der dataene fra årene 2001-2021 (Figur 4.10) og en der dataene for årene er delt opp (Figur 4.11). Gjennom kode i prototypen vil disse dataene bli lest og lagret i et objekt som blir sendt til databasen til prototypen. Se kapittel 5 i Systemdokumentasjonen (se vedlegg 9.4) for mer informasjon om databasemodellen. Disse dataene er fiktive, men skal simulere ekte data fra *Medisinsk Fødselsregister og Statistisk sentralbyrå*.

ctry	N	n_sb	pct_sb	n_lbw	pct_lbw	n_pet	pct_pet	n_gdm	pct_gdm	n_cs	pct_cs	n_fa	pct_fa	gbd
Etiopia	51222	952	1,858576	4233	8,264027	5370	10,48378	11723	22,88665	21371	41,72231	10596	20,68642	Sub-Sahara afrika
India	58157	208	0,357653	2680	4,608216	6142	10,56107	9813	16,87329	36075	62,03037	23278	40,02614	Sør-asia
Norge	2779367	27824	1,001091	141967	5,10789	198028	7,124932	333939	12,01493	943921	33,96173	835676	30,06713	Norge
Polen	85829	277	0,322735	2225	2,592364	5439	6,337019	6380	7,433385	32750	38,15727	18647	21,72576	Høy-inntekt
Somalia	39712	247	0,621978	1651	4,157434	2273	5,723711	5254	13,23026	21492	54,11966	12617	31,77125	Sub-Sahara afrika
Sri-Lanka	85821	747	0,870416	3818	4,448795	7005	8,162338	8459	9,856562	41108	47,8997	28739	33,48714	Sør-asia
tester	85821	747	0,870416	3818	4,448795	7005	8,162338	8459	9,856562	41108	47,8997	28739	33,48714	Sør-asia

Figur 4.10 - CSV format der data fra 2001-2021 er samlet

ctry	year	N	n_sb	pct_sb	n_lbw	pct_lbw	n_pet	pct_pet	n_gdm	pct_gdm	n_cs	pct_cs	n_fa	pct_fa
Etiopia	2001	543	12	2,209945	45	8,287293	48	8,839779	110	20,25783	233	42,90976	118	21,73112
Etiopia	2002	744	15	2,016129	56	7,526882	79	10,61828	148	19,89247	306	41,12903	165	22,17742
Etiopia	2003	922	14	1,518438	66	7,158351	106	11,49675	197	21,36659	387	41,97397	187	20,282
Etiopia	2004	1121	20	1,784121	113	10,08029	119	10,61552	238	21,23104	455	40,58876	235	20,96343
Etiopia	2005	1284	27	2,102804	108	8,411215	134	10,43614	283	22,0405	557	43,38006	264	20,56075
Etiopia	2006	1486	24	1,615074	112	7,537012	167	11,23822	332	22,34186	621	41,79004	287	19,31359
Etiopia	2007	1689	32	1,894612	144	8,525755	173	10,24275	372	22,02487	715	42,33274	342	20,24867
Etiopia	2008	1876	38	2,025586	171	9,115139	227	12,10021	430	22,92111	786	41,89765	380	20,25586
Etiopia	2009	2046	46	2,248289	162	7,917889	215	10,50831	487	23,80254	863	42,17986	431	21,06549
Etiopia	2010	2240	30	1,339286	178	7,946429	213	9,508929	487	21,74107	902	40,26786	469	20,9375
Etiopia	2011	2431	55	2,262443	203	8,350473	237	9,749074	556	22,87125	1045	42,98643	480	19,74496
Etiopia	2012	2631	56	2,128468	216	8,209806	275	10,4523	610	23,1851	1046	39,75675	561	21,32269
Etiopia	2013	2814	61	2,167733	228	8,102345	285	10,12793	628	22,31699	1201	42,67946	564	20,04264
Etiopia	2014	3012	70	2,324037	251	8,333333	323	10,72377	717	23,80478	1244	41,30146	625	20,75033
Etiopia	2015	3192	51	1,597744	262	8,20802	332	10,401	706	22,11779	1323	41,44737	634	19,86216
Etiopia	2016	3389	64	1,888463	277	8,173503	395	11,65536	769	22,69106	1458	43,02154	715	21,09767
Etiopia	2017	3586	55	1,533742	297	8,282209	367	10,23424	867	24,17736	1545	43,08422	749	20,88678
Etiopia	2018	3782	70	1,850873	315	8,328926	390	10,312	870	23,0037	1557	41,16869	775	20,4918
Etiopia	2019	3961	62	1,565261	330	8,331229	396	9,997475	949	23,9586	1654	41,75713	837	21,13103
Etiopia	2020	4140	79	1,908213	354	8,550725	426	10,28986	920	22,22222	1718	41,49758	872	21,0628
Etiopia	2021	4333	71	1,638588	345	7,962151	463	10,68544	1047	24,1634	1755	40,50312	906	20,9093

Figur 4.11 - CSV format der data fra 2001-2021 er delt opp etter år

I koden blir det brukt et av bibliotekene til React for å lese av CSV-filene og lagrer dataene i et objekt som kan bli sendt til databasen gjennom funksjoner som Firestore tilbyr gjennom deres API.

5 RESULTATER

5.1 Evalueringsmetode

Prosjektet vil benytte seg av tre evalueringsmetoder som omfatter enhetstesting, kontinuerlig evaluering gjennom statusmøter med veileder og oppdragsgiver samt en avsluttende brukertest utført av klinikere. Disse metodene vil bli brukt for å gi en helhetlig evaluering av prosjektets gjennomføring i henhold til funksjonalitet og målsettinger. Sammen vil disse evalueringsmetodene gi et omfattende overblikk over prosjektets fremgang og eventuelle avvik som må korrigeres for å sikre at målene blir oppnådd.

5.1.1 Enhetstesting

Enhetstesting er en form for programvare testing som brukes til å validere en del av koden, vanligvis en funksjon eller metode for å sjekke om den fungerer som forventet isolert fra resten av koden. Målet er finne eventuelle feil i koden så tidlig som mulig (*The Practical Test Pyramid*, u.å.).

Ettersom koden består av komponenter som skal visuelt presentere data fra en database vil det være store deler av koden som vil være litt krevende å teste med enhetstesting, men det ble fortsatt brukt til å teste logikken for henting av data og beregning av en type score på prototypen. En bedre løsning for automatisert testing av produktet ville ha vært "end to end" testing som blir brukt til å simulere brukeropplevelse, men dette er krevende å sette opp og det tar en del tid å kjøre testene. Mer om testing står vedlagt i systemdokumentasjon (se vedlegg 9.4).

5.1.2 Kontinuerlig evaluering

I løpet av prosjektet vil det bli gjennomført jevnlig møter med både veileder og/eller oppdragsgiver for å sikre en god fremdrift. Disse møtene gir dem muligheten til å gi verdifulle tilbakemeldinger på arbeidet som er utført så langt, samt å planlegge videre aktiviteter i prosjektet. Kontinuerlig evaluering spiller en viktig rolle i å opprettholde fokus og sikre at prosjektet holder seg på rett spor i forhold til prosjektmålene. Evalueringen gir også mulighet til å få svar på eventuelle spørsmål eller behandle viktige temaer under møtene. Selv om kontinuerlig evaluering kan kreve tid og ressurser, har det vist seg å være en effektiv metode for å oppnå høy kvalitet i prosjektarbeidet. Ved å motta tilbakemeldinger underveis har vi kunnet tilpasse og justere retningen etter behov, og dermed levere et mer skreddersydd produkt ved prosjektets avslutning.

5.1.3 Brukertesting

Brukertesting vil bli brukt som den endelige evalueringsmetoden, og vil gi klinikere mulighet til å teste gruppens prototype. Vi har vurdert to metoder for å utføre brukertesting: SUS-skala og semistrukturert intervju.

SUS-skala

System Usability Scale (SUS) er en standardisert skala som er utviklet for å måle brukervennlighet. Skalaen består av 10 spørsmål som respondentene svarer på ved å indikere i hvilken grad de er enige eller uenige med utsagnene på en 5-punkts Likert-skala. Poengene for hvert spørsmål blir deretter omregnet til et resultat mellom 0 og 100, hvor høyere poengsum indikerer en høyere grad av brukervennlighet (Brooke, 1995).

Semistrukturert intervju

Semistrukturerte intervjuer er en evalueringsmetode som er kjent for å være mer åpen og fleksibel sammenlignet med standardiserte spørreskjemaer, som for eksempel SUS-skalaen. Mens SUS-skalaen har faste spørsmål og svaralternativer som brukes i alle intervjuer, gir semistrukturerte intervjuer intervjueren mer frihet til å utforske spørsmål og emner som dukker opp underveis i intervjuet (Hove og Anda, 2005).

Semistrukturerte intervjuer gir mer fleksibilitet for intervjuere og respondenter sammenlignet med standardiserte spørreskjemaer, da spørsmålene som stilles kan variere basert på intervjuerens skjønn og respondentens svar. Dette kan føre til mer omfattende og detaljert informasjon enn standardiserte spørreskjemaer.

Valgt løsning

Vi besluttet å anvende SUS-skalaen som en evalueringsmetode fremfor semistrukturerte intervjuer. Gruppens beslutning var i stor grad basert på tidsbegrensninger og begrensede ressurser som ikke ville tillate en grundig og tidkrevende intervjuundersøkelse med hver kliniker. SUS-skalaen var mer passende da den muliggjør en effektiv og rask innsamling av data uten å kreve mye tid og ressurser. Selv om SUS-skalaen er standardisert, er det lov å gjøre små endringer for å tilpasse den til spesifikke behov eller kontekster, så lenge endringene ikke endrer meningen av spørsmålene og ikke påvirker validiteten og påliteligheten til skalaen (Hägglund og Scandurra, 2021), som sett på Figur 5.1 nedenfor. I tillegg spurte vi brukerne om de hadde noen forslag til forbedringer.

PreGO! Brukervennlighetsevaluering

Navn: _____

1 = veldig uenig

5 = veldig enig

		1	2	3	4	5
1	Jeg tror at hvis PreGO! Bli et ferdig program, så vil jeg bruke det regelmessig					
2	Jeg fant PreGO! unødvendig komplisert å bruke					
3	Jeg synes PreGO! var lett å bruke					
4	Jeg tror at jeg trenger støtte fra en teknisk person for å kunne bruke PreGO!					
5	Jeg synes de ulike funksjonene i PreGO! fungerer godt sammen.					
6	Jeg syntes det var for mye inkonsekvens (mangel på logisk sammenheng) i PreGO!					
7	Jeg kan tenke meg at de fleste ville lære å bruke PreGO! veldig raskt					
8	Jeg syntes PreGO! var veldig tungvint å bruke					
9	Jeg følte meg veldig sikker og trygg på hva jeg gjorde når jeg brukte PreGO!					
10	Jeg trengte å lære meg mye før jeg kunne forsøke å bruke PreGO!					

Figur 5.1 - PreGO! SUS-skala

5.2 Evalueringsresultat

5.2.1 Brukertesting av klinikere

For å utføre brukertesting av prototypen samlet oppdragsgiver Vik klinikere som skulle delta i brukertesten. Klinikere som skulle delta i brukertesten som brukere var alle sammen jordmødre fra HVL. De fleste klinikere deltok fysisk, mens én kliniker deltok via Zoom.

Opprinnelig var planen å vise en kort instruksjonsvideo og deretter la klinikere teste prototypen, etterfulgt av utfylling av SUS evalueringsskjemaet. Imidlertid, etter den første klinikeren, ble det klart at det var mer intuitivt å gi en kort forklaring om prototypen og la testbrukere gå gjennom prototypen selv. Under brukertesting ga vi instruksjoner om hva klinikere kunne gjøre i prototypen og støttet dem hvis de trengte hjelp eller hadde spørsmål underveis.

Oppdragsgiver Vik var til stede under brukertesten for å hente klinikere og for å gi støtte hvis det oppstod kliniske spørsmål som var for avanserte for oss. Under brukertesten opplevde vi at klinikere var fornøyde med brukergrensesnittet og at de hadde god forståelse for hva som skjedde i prototypen, spesielt på resultatsiden for fødeland. Imidlertid var det uventet at brukere måtte trykke på skrivefeltet og deretter på Enter-knappen for å få tilgang til resultatsiden når de hadde valgt fødeland. Dette var ikke så intuitivt for klinikere, men prosjektgruppen, som hadde mer IT-erfaring, syntes det var enkelt. Denne tilbakemeldingen vil være svært verdifull i den videre utviklingen av *PreGO!*.

Gruppen utførte brukertesten med totalt fire klinikere. Selv om flere klinikere ville vært ideelt, var det vanskelig å samle mange på grunn av tidsbegrensninger. Gruppen hadde bare 15 minutter til å la dem teste prototypen, evaluere den ved hjelp av SUS-skalaen og gi forslag til forbedringer. Gjennomsnittlig SUS-score for de fire klinikere var overraskende høy på 90 poeng av totalt 100 mulige poeng. Alle klinikere var enige om at de ville bruke *PreGO!* regelmessig hvis det ble et ferdig program. Dette er lovende ettersom klinikere er målgruppen for en fullstendig versjon.

5.2.2 Evaluering fra oppdragsgiver

Siden det var kort med tid i slutten av prosjektet fikk vi en skriftlig sluttevaluering der gruppen satt disse fire spørsmålene:

- **Hva er ditt generelle inntrykk av prototypen slik den fremstår nå?**

Oppdragsgivere er svært fornøyde med prototypen slik den fremstår nå og mener at den oppfyller alle de elementene de hadde bedt om i henhold til gruppens

bacheloroppgave. De understreket imidlertid at videreutviklingen av en fullversjon innebærer arbeid som ligger utenfor rammen av vår oppgave. Dette inkluderer ytterligere testing, implementering av ekte data og eventuelle justeringer og finpuss.

- **I hvilken grad mener du at prototypen oppfyller de krav og forventninger du hadde til den ved start for prosjektet?**

Oppdragsgivere uttrykker stor takknemlighet for at gruppen ble involvert tidlig i prosessen med å utvikle prototypen, som er en del av et større prosjekt med en tidsramme mot 2027 (se Figur 2.1). De mener at det arbeidet gruppen har utført så langt er svært verdifullt, og at prototypen som er levert oppfyller både krav og forventninger på en tilfredsstillende måte.

- **Dersom mangler i forhold til spørsmål 2 (initielle krav som ikke er oppfylt), vennligst list disse.**

Selv om det ikke ble identifisert noen mangler i prototypen, ønsket oppdragsgivere å nevne viktigheten av tilgang til ekte data. De uttrykte at hvis prosjektets tidsramme hadde tillatt det, ville det vært svært verdifullt å arbeide med ekte data. Dette ville ha gitt prosjektgruppen muligheten til å håndtere autentiske og realistiske scenarier. Selv om denne muligheten ikke var tilgjengelig i gruppens bacheloroppgave, anerkjenner oppdragsgiver betydningen av ekte data og ser det som en potensiell vei for videre utforskning og utvikling av prosjektet

- **Dersom du har konkrete forslag til videre forbedringer for prototypen utover mangler oppgitt i pkt.3, vennligst list disse i prioritert rekkefølge (viktigst først).**

Oppdragsgiver har gitt følgende forslag til videre forbedring av prototypen:

- Inkludere ekte data: Det er ønskelig å integrere ekte data i prototypen for å gjøre den mer autentisk og relevant. Dette vil gi brukerne en bedre opplevelse og mer pålitelige resultater.
- Utvikle migrasjonsrelaterte faktorer: Migrasjonsrelaterte faktorer har blitt implementert som et konsept i prototypen, men oppdragsgiver mener at det er viktig å utvikle dette videre. Dette vil muliggjøre en dypere innsikt i helsesituasjonen til innvandrerkvinner og bidra til å tilpasse verktøyet bedre til deres behov.

5.3 Prosjektresultat

Resultatet av prosjektet er en fungerende prototype som gir tilgang til data om gravide kvinner i et valgt land og sammenligner disse dataene med dataene om norske fødte

kvinner. Dataene er lagret inne på en database som prosjektgruppen har opprettet. Prototypen tilbyr også en risikoscore-funksjon som beregner alvorlighetsgraden av risikoen i det valgte landet og rangerer landene i henhold til denne risikoscoren. En migrasjonsrelaterte faktorer side inkluderes også i prototypen, som vil presentere andre faktorer som kan påvirke risikoscoren.

I tillegg til disse funksjonene, inkluderer prototypen en instruksjonsvideo som viser hvordan brukere kan bruke prototypen på en effektiv måte. Dette gjør det enklere for nye brukere å lære seg å bruke prototypen og øker brukervennligheten. Prototypen inneholder også administratorfunksjoner som tillater administratorer å legge til nye data for land eller oppdatere eksisterende data for land. Disse funksjonene gir administratorer muligheten til å oppdatere dataene regelmessig og sørge for at prototypen alltid inneholder de nyeste dataene.

Nærmere detaljer om utviklingen av prototypen, inkludert valg av teknologi og metoder, er beskrevet i kapittel 4.

5.4 Prosjektgjennomføring

Prosjektet ble implementert ved hjelp av en iterativ utviklingsmetodikk som tok inspirasjon fra SCRUM-metoden. Utviklingsprosessen ble strukturert gjennom bruk av fire distinkte faser, som blir nærmere beskrevet i delkapittel 3.5.1 *Prosjektplan* og 3.5.2 *Utviklingsmetodikk*.

I starten av prosjektet møtte prosjektgruppen på en betydelig utfordring med å forstå kompleksiteten i oppgaven, siden svangerskapsomsorgen var et relativt nytt område for gruppen. Likevel valgte gruppen å bruke tid og ressurser på å utforske relevant litteratur innen fagfeltet, noe som bidro til økt innsikt og bedre forståelse av temaet. Videre opplevde gruppen noen utfordringer med å tydeliggjøre oppdragsgiverens krav og forventninger i begynnelsen, da *PreGO!* prosjektet var i en enda tidligere utviklingsfase enn den er nå, og kravene ennå ikke var fullstendig definerte.

I utviklingsprosessen ble det implementert en oppgavefordeling som tildelte individuelt ansvar til hvert medlem av prosjektgruppen. Med tanke på at gruppen bestod av kun to medlemmer, skapte dette en effektiv kommunikasjonskanal mellom deltakerne. Særlig når gruppen benyttet seg av digitale verktøy for å jobbe sammen og utføre utviklingen av prototypen, muliggjorde dette en sømløs utveksling av informasjon og innspill mellom gruppemedlemmene. Denne tilnærmingen resulterte i en smidig utviklingsprosess hvor gruppen kunne bidra med innspill på alle utviklingsaspekter og kontinuerlig tildeles nye arbeidsoppgaver, slik at det ikke oppsto perioder med inaktivitet.

Prosjektet ble gjennomført med suksess ved hjelp av en iterativ utviklingsmetodikk og effektiv kommunikasjon internt i gruppen. Gruppen leverte en fungerende prototyp som møtte oppdragsgiverens krav og forventninger. Gjennom grundig litteraturstudie fikk gruppen økt forståelse for svangerskapsomsorgen. Til tross for noen utfordringer i starten med å klargjøre oppdragsgiverens krav, opprettholdt gruppen tett dialog og oppnådde en smidig utviklingsprosess. Totalt sett var prosjektgjennomføringen positiv og resulterte i en vellykket leveranse.

6 DISKUSJON

6.1 Utvikling og evaluering av funksjoner

Målet for det aktuelle bachelorprosjektet var å utvikle og evaluere en prototype for et kunnskapsbasert verktøy som skal kunne hjelpe klinikere i deres arbeid med gravide innvandrerkvinner. I utgangspunktet var ikke alle funksjoner for prototypen spesifisert, da prosjektet befant seg i en tidlig fase av utviklingen. Etter en rekke møter med oppdragsgiver og veileder ble det besluttet hvilke funksjoner som skulle inkluderes i prototypen underveis.

De initielle kravene fra oppdragsgiver for prototypen omfattet en webapplikasjon med en påloggingsside, en søkefunksjon for land og en resultatside som presenterte data fra en database. Dette beskrives nærmere i delkapittel 2.1.2. I løpet av prosjektet ble det gjennom møter med oppdragsgiver og veileder identifisert flere funksjoner som skulle inkluderes eller endres i prototypen. Disse inkluderte blant annet en linjegrav for hver risikofaktor (delkapittel 4.2.2), en risikoscore (delkapittel 4.2.3), andre migrasjonsrelaterte faktorer (delkapittel 4.2.4) og flytting av beskrivelsen fra resultatsiden (delkapittel 4.2.5). Endringene fra de initielle kravene ble diskutert enten før utviklingsprosessen startet eller under iterasjons møter, hvor vi viste iterasjoner av prototypen til oppdragsgiver og mottok tilbakemeldinger som ble tatt med videre til neste iterasjon.

6.2 Svakheter til prosjektløsningen

Dette er gruppens første prosjekt der rammeverket React blir brukt siden dette er en av de større frontend prosjektene vi har jobbet med og ville derfor lære nytt teknologi som var egnet for dette prosjektet. Med de nye kunnskapene vi har fått er det en del ting vi ville ha gjort annerledes om det hadde vært mulig å starte på nytt igjen.

6.2.1 Svakheter i koden og potensielle fiks

I gruppens første prosjekt med bruk av React-rammeverket er det forventet at prototypen vil ha visse svakheter. I denne delen vil vi presentere de identifiserte svakhetene i prototypen, potensielle løsninger for å fikse dem, samt en forklaring på hvorfor disse svakhetene oppstår.

Land-velgeren (delkapittel 4.2.1): Dersom man ser bort fra tilbakemeldingen gruppen mottok under brukertesting, slik det er beskrevet i delkapittel 5.2 om brukertesting av klinikere, kan det observeres en svakhet i land-velgeren når det gjelder henting av land fra databasen. Denne svakheten oppstår ved at alle land hentes ut samtidig fra databasen. En bedre løsning ville være å kun hente ut de landene som skal vises i

rullegardinmenyen. På grunn av gruppens begrensede kunnskap ble det valgt å ikke implementere denne logikken, ettersom det å hente ut alle landene fra databasen ville være tilstrekkelig for prototypen.

Resultatsiden (delkapittel 4.2.2): Denne siden presenterer hovedfunksjonene til prototypen, som inneholder en rekke ulike funksjoner. En betydelig svakhet i koden er den manglende integrasjonen mellom graf-komponentene og risiko-sirkelen. Som en konsekvens av denne mangelen må disse komponentene foreta separate spørringer mot databasen. Dette resulterer i økt trafikk mellom klientene og databasen, noe som kan påvirke prototypens ytelse og kostnader. Vi ble oppmerksomme på dette problemet sent i utviklingsprosessen når vi fikk mer kompetanse. På grunn av begrenset tid og behovet for omfattende endringer for å løse dette problemet, har vi valgt å fortsette uten å fikse denne svakheten. Til tross for dette fungerer funksjonaliteten fortsatt godt i prototypen.

Andre migrasjonsrelaterte faktorer (delkapittel 4.2.4): Svakheten til denne funksjonen er logikken noe vi var klare over veldig tidlig i utviklingsprosessen, dette kan leses mer om i delkapittel 4.2.4. Vi fikk også tilbakemelding under brukertesting at denne funksjonen ikke var så brukervennlig, se delkapittel 7.2.

Land beskrivelse (delkapittel 4.2.6): Land beskrivelsen blir hentet fra en database der teksten ikke blir formatert ordentlig, dette fører til at teksten ikke har avsnitt, noe som fører til at teksten kan være tung å lese. Dette kan være en lett fiks ved å formatere teksten før den blir lastet opp i databasen. Vi har valgt å ikke bruke altfor mye tid på denne funksjonen ettersom denne funksjonen var den minst prioriterte funksjonen og demonstrerte det oppdragsgiverne ønsket av funksjonen.

laste opp ny eller endre data (delkapittel 4.3.2): Vi har delt funksjonaliteten i to basert på CSV-fil typen administratoren ønsker å laste opp, se delkapittel 4.3.2 for informasjon om CSV-filene.

Det var egentlig unødvendig å gjøre dette. En mer optimal løsning som vi senere kom på var å bruke en CSV-fil-type der dataene er fordelt på årene, og deretter ha en funksjon i koden som samler dataene fra alle årene og legger dem sammen for å presentere en generell visning av dataene i databasen.

Årsaken til denne funksjonen er delt i to var at ved begynnelsen av utviklingen var det nok for oss å bruke den generelle CSV-fil typen for å vise sirkler og grafen på resultatsiden, som beskrevet i delkapittel 4.2.2. Senere fikk vi tilbakemelding fra oppdragsgiver som ønsket å se utviklingen av de ulike risikofaktorene over tid på resultatsiden, slik at klinikere kunne observere endringer over tid.

På grunn av en lavere prioritering av denne funksjonen og begrenset tid, tok vi ikke nok tid til å revaluere hele denne funksjonaliteten og bestemte heller å utvikle en ny filopplastning funksjon for den nye CSV-fil typen.

Rammeverk og database: Vi har generelt vært veldig fornøyd med valg av rammeverket React og databasen Firebase sin database Firestore for prototypen. Noe vi kunne ha vurdert var å bruke typescript istedenfor JavaScript. Gruppen utforsket ikke dette valget, ettersom vi var mer kjent med JavaScript enn typescript og ville komme i gang med utviklingen så fort som mulig. Med mer tid kunne vi ha vurdert valget om å bruke typescript ovenfor JavaScript.

6.2.2 Svakheter i utviklingsmetodikk

Bachelorprosjektet utgjør det mest omfattende prosjektet gruppen har jobbet med hittil. På grunn av begrenset erfaring med et slikt prosjekt, er det naturlig at det er rom for forbedringer når det gjelder utvikling metodikken som brukes av oss. En annen vesentlig faktor som har påvirket vår utviklingsmetodikk er at gruppen kun består av to medlemmer.

Siden gruppen bestod bare av to gruppemedlemmer var det mange arbeidsoppgaver gruppemedlemmene måtte sette seg inn spesielt under utviklingen når gruppemedlemmene måtte lære seg nye verktøy som firebase og react. Det var krevende for gruppemedlemmene å studere de nye verktøyene og samtidig utvikle funksjonene til prototypen. Med tidspresset for å levere noe til hver iterasjon i koden har hovedfokuset vært å få funksjonene til å funke uten å tenke så mye på den samla integrasjonen i koden.

6.3 Styrker til prosjekløsningen

6.3.1 Styrker i utviklingsverktøy

Verktøyene som ble brukt for å utvikle prototypen har vært veldig behagelig å bruke og lett å lære. React sin simple komponent drevet arkitektur har bidratt til en oversiktlig kodeprosess som har gjort det lett for oss å lære og ta i bruk for å utvikle prototypen. Med det brede biblioteket React tilbyr har det ikke vært noe problem å finne en måte å utvikle de forskjellige funksjonene i prototypen.

Firestore database er databasen som har blitt brukt for denne prototypen. Det har vært veldig lett for oss å integrere Firestore sin API inn i koden og gjøre spørringer mot databasen. Det var ikke behov for å lage en DAO (data access object) ved bruk av Firestore med React siden Firestore tilbyr innebygde funksjoner for å gjøre spørringer

mot databasen gjennom API til Firestore, kan leses mer om i Systemdokumentasjonen (se vedlegg 9.4) (kapittel 2).

6.3.2 Styrker i utviklingsmetodikk

Som tidligere nevnt i delkapittel 6.2.1 består gruppen av kun to medlemmer. Dette har vært både en svakhet og en styrke for oss.

Gjennom den iterative utviklingsmetoden har vi vært i stand til å tilpasse seg og levere en prototype som oppdragsgiverne er tilfredse med. Dette skyldes hovedsakelig de regelmessige statusmøtene vi har hatt med oppdragsgiveren og veilederen, der vi har mottatt tilbakemeldinger på arbeidet som ble utført etter hver iterasjon. Dette har muliggjort justeringer og tilpasninger i prosjektet etter oppdragsgiverens behov og endringer i de opprinnelige kravene.

Det at gruppen består av kun to medlemmer har bidratt til visse fordeler i prosjektarbeidet. En vesentlig fordel ved det lave antallet gruppe-medlemmer er at kommunikasjonen mellom medlemmene er svært effektiv, spesielt under lengre utviklings- og skriveøkter. Dette har gitt oss en høyt fleksibel arbeidsprosess der det er mulig å bytte arbeidsoppgaver underveis, raskt å få innspill fra det andre gruppe-medlemmet og eliminere behovet for formelle gruppe-møter der arbeidsfordeling, progresjon og spørsmål blir diskutert. Dette kan enkelt kommuniseres mens gruppen jobber.

En annen fordel med kun to medlemmer i gruppen er arbeidsfordelingen mellom gruppe-medlemmene. Med et så begrenset antall medlemmer er det alltid arbeid å utføre i prosjektet. Det er også enkelt for hvert medlem å følge med på den andre medlemmens fremgang for å forsikre seg at arbeid blir gjort og at oppgavene som medlemmene gjør ikke overlappes.

6.4 Samfunnsmessige Fordeler og Etske Vurderinger

6.4.1 Samfunnsmessige fordeler

Gruppens prototype har potensial til å gi betydelige samfunnsmessige fordeler hvis den blir videreutviklet til en fullstendig versjon som sett på Figur 2.1 som reflekterer visjonen til oppdragsgiver Vik og Nilsen. Ved å tilby et brukervennlig verktøy som gir klinikere og jordmødre innsikt i helsesituasjonen til gravide innvandrerkvinner, kan prototypen bidra

til følgende samfunnsmessige fordeler:

Forbedret helseutkomst: Ved å gi relevante og pålitelige data kan prototypen bidra til at klinikere tar bedre informerte beslutninger og tilpasser omsorgen til den enkelte kvinne. Dette kan føre til bedre helseutkomst for gravide innvandrerkvinner, med redusert risiko for komplikasjoner og forbedret fødselsresultater.

Mer skreddersydd omsorg: Prototypen gir klinikere muligheten til å identifisere spesifikke helseutfordringer og behov hos gravide innvandrerkvinner. Dette legger grunnlaget for en mer skreddersydd og individualisert omsorg, som kan bidra til bedre behandlingsresultater og økt trivsel for den enkelte kvinne.

Økt bevissthet og kunnskap: Gjennom bruk av prototypen blir det skapt økt bevissthet om helseutfordringer knyttet til innvandrerkvinner og deres spesifikke behov. Dette kan bidra til at både helsepersonell og samfunnet som helhet får en bedre forståelse av de utfordringene denne gruppen står overfor, og dermed legge til rette for målrettede tiltak og ressurser.

Effektiv ressursutnyttelse: Prototypen kan bidra til en mer effektiv ressursutnyttelse i helsetjenesten. Ved å gi klinikere rask tilgang til relevant informasjon og verktøy for vurdering, kan det redusere behovet for unødvendige undersøkelser og bidra til en mer målrettet bruk av ressurser.

6.4.2 Ethiske vurderinger om tilgang til ekte data

Under utviklingen av prototypen ble gruppen stilt spørsmål i forbindelse med etiske vurderinger vedrørende bruk av ekte data fra *Medisinsk Fødselsregister og statistisk sentralbyrå*. I sluttevalueringen fra oppdragsgiver ble det uttrykt ønske om at ekte data hadde vært inkludert dersom tidsrammen hadde tillatt det. Dette understreker betydningen oppdragsgiver la på autentiske data for å oppnå en mer realistisk representasjon av helsesituasjonen til innvandrerkvinner.

I tillegg til tidsbegrensningen var det også andre viktige etiske grunner til at gruppen ikke benyttet ekte data i prototypen. Et vesentlig aspekt var personvernet til individene i *Medisinsk Fødselsregister*. Det var viktig for oss å respektere personvernet til disse individene og sikre at deres data ikke ble brukt uten samtykke og riktig juridisk ramme. Det må også bemerkes at dersom vi skulle utviklet en løsning som muliggjør sikker bruk av ekte pasientdata, da måtte vi også ha ivare tatt alt det sikkerhetsmessige, og prosjektet *PreGO!* måtte ha søkt Datatilsynet og andre offentlige organ om de nødvendige rettigheter. Dette vil være en nødvendig utfordring for en senere videreutvikling av verktøy.

En ytterligere etisk bekymring var risikoen for mulig misbruk av dataene. Med tilgang til detaljerte opplysninger om gravide kvinner fra minoritetsland eller land med lav representasjon i Norge, var det en reell risiko for utilsiktet eller bevisst misbruk av informasjonen. Som en ansvarlig prosjektgruppe var gruppen forpliktet til å beskytte personvernet og sikre at sensitive opplysninger ikke ble utsatt for utilsiktet spredning eller misbruk.

7 KONKLUSJON OG VIDERE ARBEID

7.1 Konklusjon

Målet for dette prosjektet var å utvikle og evaluere et kunnskapsbasert verktøy som klinikere kan bruke i møte med gravide innvandrerkvinner. Gruppens sluttresultat ble et verktøy som inneholder relevant kunnskap og informasjon som er spesifikt rettet mot klinikere som jobber med gravide innvandrerkvinner.

Vi er overbevist om at prototypen som er blitt utviklet kan være en solid grunnmur og kan være nyttig for videreutvikling av en fullverdig versjon. Forbedringsforslagene som er nevnt i neste delkapittel kan bli brukt for å bygge videre på verktøyet. Ettersom de opprinnelige kravene i kravdokumentet (se vedlegg 9.3) og de funksjonelle egenskapene beskrevet i visjonsdokumentet (se vedlegg 9.2) er blitt oppfylt, kan vi konkludere med at målet om å utvikle et kunnskapsbasert verktøy er nådd.

Videre, med tanke på forskningsspørsmålet som ble formulert i starten av prosjektet i kapittel 1:

- **"Hvordan kan vi utvikle en prototype av PreGO! som fremtidige brukere vil oppleve som et nyttig verktøy?"**

Evalueringen av prototypen gjennom brukertesting, der jordmødrene fra HVL deltok og vurderte verktøyet på SUS-skalaen, resulterte i enighet blant alle deltakerne om at hvis *PreGO!* hadde blitt en ferdig programvare, ville de ha brukt det regelmessig. Basert på svarene på SUS-skalaen og tilbakemeldingene fra målgruppen av fremtidige brukere, konkluderer vi med at prototypen i høy grad oppfyller kravene til å være et nyttig verktøy for klinikere.

I tillegg til den vellykkede brukertesting har vi også mottatt en sluttevaluering fra oppdragsgiverne, som i stor grad var tilfredse med prosjektet. Selv om det ble påpekt at bedre tidsrammer ville ha muliggjort implementering av ekte data, ble det gitt positiv tilbakemelding på arbeidet utført av oss.

Vi kan konkludere med at den andre delen av målsetningen, nemlig evalueringen av prototypen, er oppnådd. Dette skyldes de positive tilbakemeldingene fra oppdragsgivere og den vellykkede brukertesting som har blitt gjennomført. Denne suksessen bidrar til at det overordnede målet for prosjektet også er oppnådd.

7.2 Videre arbeid

7.2.1 Resultatsiden

På resultatsiden var det opprinnelig planlagt å inkludere en funksjon for landsflagg, der hvert land skulle bli representert med sitt eget flagg. Imidlertid ble implementeringen av denne funksjonaliteten utelatt på grunn av tidsbegrensninger. Å integrere individuelle landsflagg viste seg å være mindre intuitivt, da det ville krevd manuell tilføyning av bilder for hvert enkelt land. I tillegg ble det ikke funnet en hensiktsmessig metode for å lagre og hente flaggbilder fra en database, noe som også bidro til beslutningen om å utelate denne funksjonaliteten.

En annen forbedring som kunne ha blitt implementert hadde vært en presentering av beregningsmetoden for risikoscoren. Dette kunne ha involvert en tydelig representasjon av alvorlighetsgraden knyttet til hver enkelt risikofaktor. En slik tilnærming ville ha gitt brukerne en bedre forståelse av hvilke faktorer som bidrar mest til den totale risikoscore. I tillegg ville det vært gunstig å inkludere en funksjon som gjør det mulig for brukerne å vurdere om risikoscoren for et spesifikt land er under en definert lavere eller høyere grense. Ved å etablere tydelige grenser for hva som betraktes som en "god" eller "dårlig" risikoscore, kan brukerne enkelt tolke resultatene og forstå hvorvidt risikoen er akseptabel eller krever ytterligere oppfølging.

Når det gjelder representasjonen av ulike risikofaktorer på resultatsiden, ville det vært hensiktsmessig å implementere en mer fleksibel måte å integrere nye data på. Det nåværende datasettet vi har inneholder kun 6 risikofaktorer, mens den virkelige dataen fra *Medisinsk Fødselsregister og Statistisk sentralbyrå* inneholder mange flere. Det ville vært en forbedring å utvikle en mer skalerbar og tilpasningsdyktig kode som kan håndtere en større mengde risikofaktorer.

7.2.2 Migrasjonsrelaterte faktorer

I delkapittel 4.2.4 "Andre migrasjonsrelaterte faktorer" ble det påpekt at denne siden hadde lav prioritet, og den ble utviklet for å demonstrere konseptet av hvordan den skulle fungere. Imidlertid mangler mye av logikken på denne siden. Under brukertesting mottok vi tilbakemeldinger som indikerte at det ville vært mye mer brukervennlig hvis spørreskjemaet hadde benyttet en rullegardinmeny med spesifikke valg for hvert spørsmål, i stedet for å bare tillate brukeren å velge mellom 3 forskjellige fargede sirkler for spørsmålet "Årsak til innvandring". Vi var enige i at dette ikke var optimalt.

7.2.3 Administrator funksjoner

I prototypen fungerer disse funksjonene som den skal, utenom de mulige forbedringer som ble nevnt i delkapittel 6.2.1 kan det være greit å oppdatere prototypen slik at den får sanntidsoppdatering fra databasen etter at en admin har gjort endringer på data eller last opp ny data. Dette er mulig med Firestore databasen som vi har tatt i bruk, men vi hadde ikke nok tid til å prioritere denne funksjonen for prototypen

7.2.4 Database

En mulig forbedring eller forslag til videre arbeid ville være å vurdere å bytte til en annen database tjeneste i stedet for Firebase, som ble brukt i utviklingen av verktøyet. Selv om Firebase har sine fordeler og funksjonaliteter, kan det være begrensninger knyttet til avhengighet av Google-tjenester. Ved å velge en annen database tjeneste kan man oppnå flere fordeler:

Uavhengighet fra spesifikke leverandører: Ved å bytte til en annen database tjeneste kan man unngå å være begrenset til kun å bruke tjenester fra én bestemt leverandør. Dette åpner opp muligheten for å velge den beste løsningen for prosjektets behov, uavhengig av leverandør.

Tilpasningsmuligheter: En alternativ database tjeneste kan gi flere tilpasningsmuligheter og fleksibilitet i hvordan data organiseres og behandles. Dette kan være spesielt nyttig når det gjelder komplekse datamodeller eller spesifikke krav til databehandling.

Kostnadsoptimalisering: Ved å evaluere ulike database tjenester kan man også vurdere kostnadsaspektet. Noen tjenester kan være mer kostnadseffektive enn andre, spesielt når det gjelder skalerbarhet og datalagring. Ved å velge en optimal løsning kan man redusere driftskostnader og ressursbehov.

7.2.5 Brukergrensesnitt:

I henhold til delkapittel 4.1 *Utforming av brukergrensesnitt*, var hovedfokuset til gruppen å sikre at logikken prototypen fungerte, og mindre vekt ble lagt på det visuelle designet av brukergrensesnittet. Imidlertid kan det være en forbedring eller et forslag å implementere en offisiell hovedside etter at brukeren har logget inn på prototypen. Denne hovedsiden vil inkludere Land-velgeren og andre essensielle funksjoner som skal implementeres videre. På denne måten kan brukeren få en oversiktlig og intuitiv startside som gir tilgang til viktige verktøy og funksjoner, og samtidig opprettholde fokus på prototypens funksjonalitet.

Vi støtte på utfordringer knyttet til skjermoppløsning da prototypen skulle brukes på ulike enheter med varierende skjermoppløsninger. Selv om utviklingen ble utført på stasjonære PC-er, ble brukertesting gjennomført på både bærbare datamaskiner og TV-

skjermer. Dette medførte behovet for å zoome inn verktøyet for å tilpasse det til den aktuelle enhetens skjermstørrelse. En forbedring på dette området kunne vært å implementere en løsning som sikrer riktig skalerbarhet av verktøyet uavhengig av hvilken enhet det brukes på. Dette vil sikre en gunstig visuell opplevelse og brukervennlighet på tvers av ulike enheter.

8 REFERANSER

bacancytech (2022) 'Why Use Vue Js: Unleash The Frontend Framework Head To Toe', 16 September. Available at: <https://www.bacancytechnology.com/blog/why-use-vue-js> (Accessed: 8 May 2023).

Brooke, J. (1995) 'SUS: A quick and dirty usability scale', *Usability Eval. Ind.*, 189.

Chand, M. (2022) *Most Popular Databases In The World (2023)*. Available at: <https://www.c-sharpcorner.com/article/what-is-the-most-popular-database-in-the-world/> (Accessed: 11 May 2023).

Firebase (u.å) *Firestore*, *Firebase*. Available at: <https://firebase.google.com/docs/firestore> (Accessed: 6 May 2023).

Firebase (u.å) *Firebase Authentication*, *Firebase*. Available at: <https://firebase.google.com/docs/auth> (Accessed: 16 May 2023).

Fowler, M. (u.å) *The Practical Test Pyramid*, *martinfowler.com*. Available at: <https://martinfowler.com/articles/practical-test-pyramid.html> (Accessed: 11 May 2023).

Hägglund, M. and Scandurra, I. (2021) 'User Evaluation of the Swedish Patient Accessible Electronic Health Record: System Usability Scale', *JMIR Human Factors*, 8(3), p. e24927. Available at: <https://doi.org/10.2196/24927>.

Hove, S.E. and Anda, B. (2005) 'Experiences from conducting semi-structured interviews in empirical software engineering research', in *11th IEEE International Software Metrics Symposium (METRICS'05)*. *11th IEEE International Software Metrics Symposium (METRICS'05)*, p. 10 pp. – 23. Available at: <https://doi.org/10.1109/METRICS.2005.24>.

Hughson, J.-A.P. *et al.* (2018) 'The Rise of Pregnancy Apps and the Implications for Culturally and Linguistically Diverse Women: Narrative Review', *JMIR mHealth and uHealth*, 6(11), p. e189. Available at: <https://doi.org/10.2196/mhealth.9119>.

javatpoint (u.å) *Pros and Cons of ReactJS - javatpoint*, *www.javatpoint.com*. Available at: <https://www.javatpoint.com/pros-and-cons-of-react> (Accessed: 22 February 2023).

Melnyk, O. (2022) *Top 10 Big Companies Using React*, *Career Karma*. Available at: <https://careerkarma.com/blog/companies-that-use-react/> (Accessed: 22 February 2023).

Micro Focus (u.å) *What Is Machine Learning and Why Is It Important?* | Micro Focus. Available at: <https://www.microfocus.com/en-us/what-is/machine-learning> (Accessed: 8 May 2023).

Saini, S. and Sakshi, A. (2021) *8 Reasons Why You Need Angular Today*, Grazitti Interactive. Available at: <https://www.grazitti.com/blog/8-proven-reasons-you-need-angular-for-your-next-development-project/> (Accessed: 8 May 2023).

scrum (u.å) *What is Scrum?*, Scrum.org. Available at: <https://www.scrum.org/learning-series/what-is-scrum> (Accessed: 9 May 2023).

ssb (2001) *Oppdaterte tall om innvandrere 2001*, ssb.no. Available at: <https://www.ssb.no/befolkning/artikler-og-publikasjoner/oppdaterte-tall-om-innvandrere--42215> (Accessed: 22 May 2023).

ssb (2001) *Oppdaterte tall om innvandrere 2001*, ssb.no. Available at: <https://www.ssb.no/befolkning/artikler-og-publikasjoner/oppdaterte-tall-om-innvandrere--42215>

Vik, E. S., Nilsen, R. M. med flere. Designing, developing and testing a risk assessment tool for clinicians caring for pregnant immigrant women in antenatal care (PreGO!). Prosjektsøknad sendt Regionalt Samarbeidsorgan for forskning og innovasjon, februar 2023. Intern rapport IHO/FHS/HVL.

9 VEDLEGG

Vedlegg 9.1 – Prosjekthåndbok

Vedlegg 9.2 – Visjonsdokument

Vedlegg 9.3 – Kravdokument

Vedlegg 9.4 – Systemdokumentasjon

Vedlegg 9.5 - Brukertesting av klinikere

Vedlegg 9.6 - Instruksjonsvideo: <https://www.youtube.com/watch?v=MI5AJFxxfBU>