

Maskinell Transkribering av Snapchat Skjermopptak
Automatic Transcription of Snapchat Screen Recordings
Kravdokumentasjon

Versjon 3.1

Dokumentet er basert på Kravdokumentasjon utarbeidet ved NTNU. Revisjon og tilpasninger til bruk ved IDER, DATA-INF utført av Carsten Gunnar Helgesen, Svein-Ivar Lillehaug og Per Christian Engdal. Dokumentet finnes også i engelsk utgave.

REVISJONSHISTORIE

Dato	Versjon	Beskrivelse	Forfatter
17/feb/22	1.0	Funksjonalitet, prototype og wireframes	Thea
23/feb/22	1.1	Domenemodell, wireframes	Thea
24/feb/22	1.2	Gjennomgang før innlevering	Anne
10/mar/22	2.0	Revisjon etter tilbakemelding	Vilde
20/apr/22	2.1	Gjennomgang før innlevering	Vilde
03/mai/22	2.2	Gjennomgang før innlevering	Anne
19/mai/22	3.0	Oppdaterte krav	Anne
20/mai/22	3.1	Gjennomgang før endelig innlevering	Anne

INNHALDSFORTEGNELSE

1	INNLEDNING	1
2	FUNKSJONALITET	2
2.1	KRAV FRA OPPDRAGSGIVER	2
2.1.1	<i>Prioriteringsnivå 1</i>	2
2.1.2	<i>Prioriteringsnivå 2</i>	2
2.1.3	<i>Prioriteringsnivå 3</i>	2
2.2	BRUKSTILFELLER	3
3	DOMENEMODELL	7
4	PROTOTYPER	9
4.1	WIREFRAMES	9
4.2	HTML-PROTOTYPER.....	10
5	REFERANSER	11

1 INNLEDNING

Kravdokumentasjon brukes for å beskrive en prototype for funksjonaliteten til produktet som er utviklet. Kravdokumentasjonen beskriver kravene stilt til prosjektet og hvordan de oppnås. God dokumentasjon av krav og spesifikasjoner gir oversikt slik at prosjektplanlegging skjer detaljert uten å miste det helhetlige bildet.

Dokumentasjonen inneholder beskrivelse av funksjonaliteten til produktet basert på behov gitt av oppdragsgiver. Problemet beskrives ved hjelp av elementer i utviklingen ved en domenemodell. Brukergrensesnittet beskrives ved en visuell modell av samhandling mellom bruker og programvare.

2 FUNKSJONALITET

2.1 Krav fra oppdragsgiver

Kravene satt av oppdragsgiver deles inn i tre ulike prioriteringsnivåer. Det første nivået beskriver funksjonalitet som må fungere ved endelig produkt. Nivå to inkluderer funksjonalitet som burde fungere ved endelig produkt, og nivå tre inkluderer funksjonalitet som det er ønskelig at fungerer i det endelige produktet. Dersom ikke alle kravene i nivå tre oppfylles ved endt prosjekt vil produktet fortsatt tilfredsstillende de viktigste kravene satt av oppdragsgiver.

2.1.1 Prioriteringsnivå 1

Det viktigste kravet fra oppdragsgiver er at programvaren skal kunne transkribere tekst fra skjermopptak uten å miste innhold.

For at programvaren skal være brukervennlig må brukergrensesnittet være intuitivt og uavhengig av operativsystem.

Resultater fra transkriberingen må være mulig å lese og tydelig vise forløpet i kommunikasjonen.

Et krav som ble lagt til av oppdragsgiver sent i prosjektet var muligheten til å søke etter ord som finnes i videoen og få ut de bildene hvor ordet er med.

2.1.2 Prioriteringsnivå 2

Etter at kravene i første prioriteringsnivå er oppfylt er det ønskelig å utvikle programvaren videre til å kunne transkribere videoer uavhengig av alfabet og språket brukt. Generelt innebærer dette at transkriberingen skal kunne fungere på flere spesialtegn.

Dersom det ikke er tilgang på skjermopptak, skal det være mulig å transkribere tekst fra flere skjermbilder i en mappe.

For å gjøre installasjon og bruk av programvaren enklest mulig for brukeren ønskes det et kjørbart program som installerer pakker som brukes i programvaren automatisk.

2.1.3 Prioriteringsnivå 3

Det siste prioriteringsnivået inkluderer funksjonalitet som kan utvikles hvis det er tid og kapasitet til det.

Selv om programvaren kan håndtere ulike språk og alfabet betyr ikke det at brukeren kan tyde det. Et tillegg som kunne forenklet forståelsen av resultater er en oversettelsesfunksjon som gir den originale teksten og oversatt tekst ved siden av hverandre.

I chat-samtaler er det vanlig å bruke emoji'er, animasjoner og å sende bilder. Å hente ut disse elementene sammen med transkriberingen slik at de plasseres korrekt i forløpet hadde gitt dypere forståelse for den transkriberte teksten.

Emoji'er brukes ofte for å representere andre betydninger og politiet vil ofte finne disse igjen. Å inkludere at søkefunksjonaliteten gir ut bilder hvor man finner spesifikke emoji'er ville vært svært hjelpsomt.

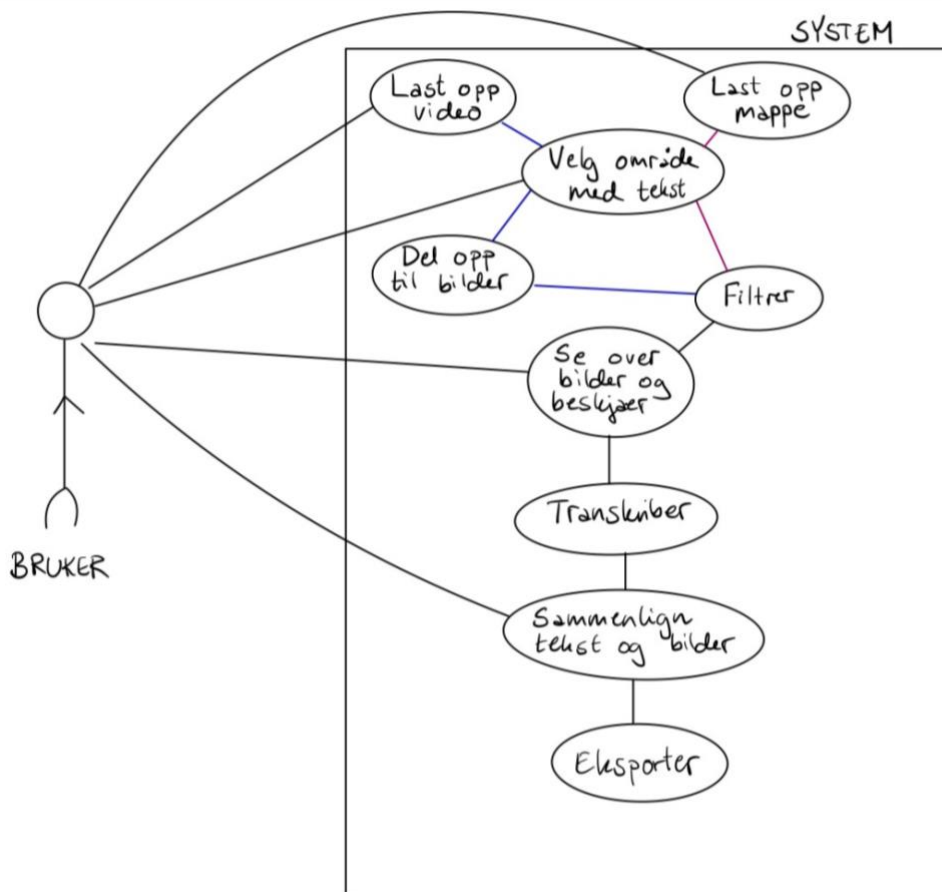
2.2 Brukstilfeller

Brukstilfellemodeller brukes for å beskrive funksjonalitet til programvare basert på brukerinteraksjon.

Den visuelle representasjonen av brukstilfellet for programvaren med de originale kravene er vist i Figur 1. Her vises hvilke funksjonaliteter som er avhengig av brukeren og hvilke som utføres av systemet. Brukeren er ansvarlig for å velge mellom å laste opp en video eller en mappe med bilder. Videre må brukeren velge området med tekst som skal transkriberes slik at ikke overflødig informasjon blir inkludert i systemet sine prosesser. Etter at systemet har delt opp videoen til bilder, ved valg av video, og filtrert bilder for å finne mer unike bilder, må brukeren igjen se over de utvalgte bildene og beskjære eller hoppe over dem ved overlappende tekst.

Videre vil systemet transkribere områdene valgt av brukeren, og til slutt kan brukeren sammenligne den transkriberte teksten og bildet for å utføre endringer ved feil transkribering av tekst. Den endrede transkriberte teksten eksporteres så til en PDF.

Proessen er også beskrevet i brukstilfelletabell, leselig i Tabell 1.

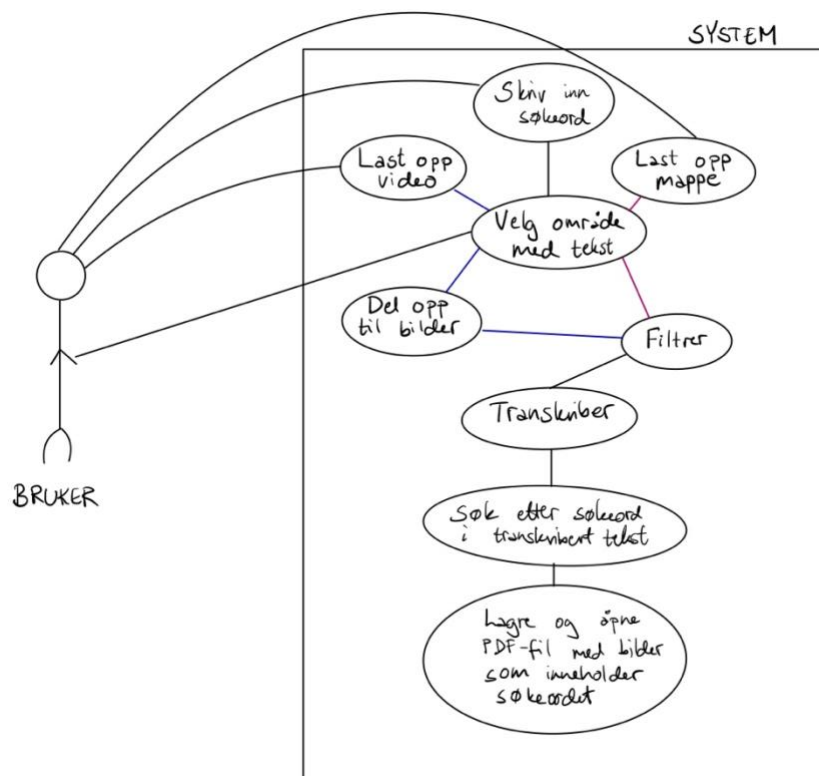


Figur 1 Brukstilfellediagram for opprinnelige krav

Tabell 1 Brukstilfellebeskrivelse for opprinnelige krav

Navn:	Transkribering av skjermopptak eller skjernbilder
Aktører:	Bruker av systemet
Hensikt/Målsetting:	Brukstilfellet viser hva som skjer når aktøren bruker systemet. Det er mulig å velge mellom å laste opp en videofil eller flere bilder som skal transkriberes før resultatet eksporteres.
Normalflyt:	<ol style="list-style-type: none"> 1. Brukeren velger å laste opp en videofil 2. Brukeren velger hvilken del av videoen som inneholder tekst 3. Systemet deler videofilen opp i skjernbilder og filtrerer disse 4. Brukeren ser over de utvalgte bildene og kan beskjære eller fjerne overflødige elementer 5. De prosesserte bildene transkriberes av systemet 6. Brukeren sammenligner den transkriberte teksten med det originale bildet og kan redigere hvis det er nødvendig 7. Resultatet eksporteres til PDF format
Alternativ flyt #1:	<ol style="list-style-type: none"> 1. Brukeren velger å laste opp flere bilder lagret i en mappe 2. Brukeren velger hvilken del av hvert bilde som inneholder tekst 3. Systemet transkriberer hvert bilde 4. Brukeren sammenligner den transkriberte teksten med det originale bildet og kan redigere hvis det er nødvendig 5. Resultatet eksporteres til PDF format

De nye kravene satt av oppdragsgiver, hadde fokus på søkefunksjonalitet med redusert fokus på nøyaktighet i transkriberte resultater. Brukstilfellediagram for denne funksjonaliteten er vist i Figur 2 og beskrevet i Tabell 2. I denne nye løsningen er det kun to punkter hvor programvaren krever samhandling med brukeren. Når programmet startes, må det lastes opp en fil eller en mappe og brukeren må skrive inn et søkeord. Etter at systemet har mottatt informasjonen fra dette punktet må brukeren velge området i et skjermbilde som inneholder tekst slik som beskrevet i brukstilfellet beskrevet i Figur 1.



Figur 2 Brukstilfellediagram for oppdaterte krav

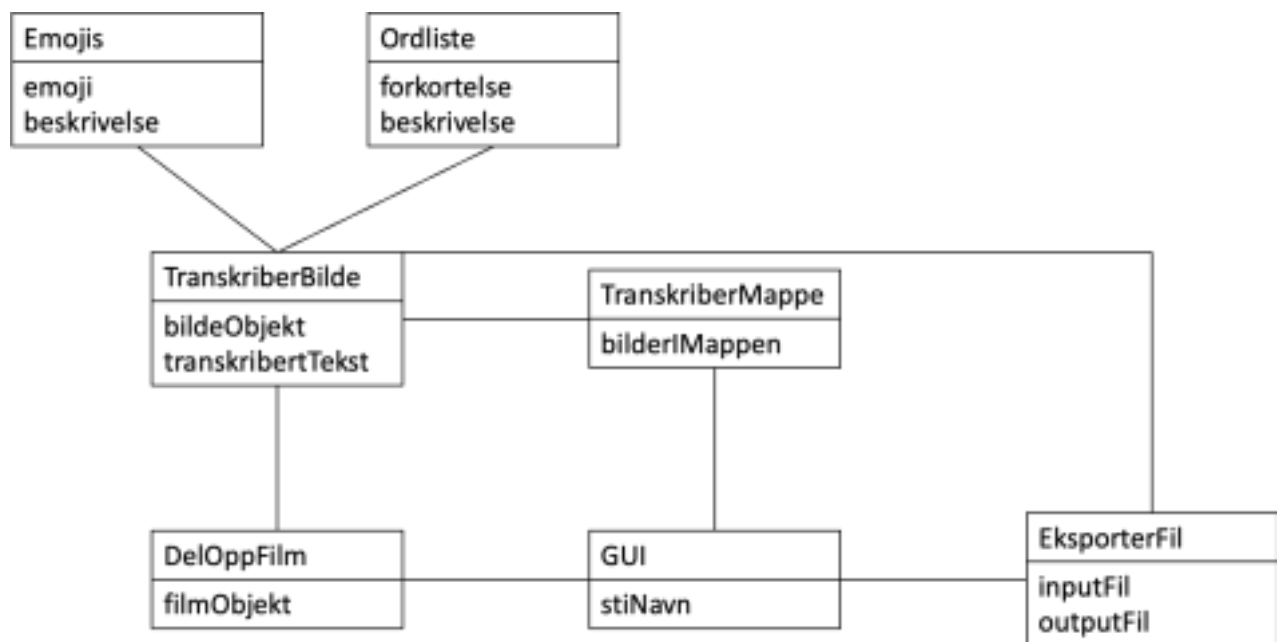
Tabell 2 Brukstilfellebeskrivelse for oppdaterte krav

Navn:	Transkribering av skjermopptak eller skjermbilder inkludert ordsøk
Aktører:	Bruker av systemet
Hensikt/Målsetting:	Brukstilfellet viser hva som skjer når aktøren bruker systemet. Det må oppgis et søkeord som skal finnes igjen i den transkriberte teksten. Det er mulig å velge mellom å laste opp en videofil eller flere bilder som skal transkriberes og søkes i før resultatet eksporteres.
Normalflyt:	<ol style="list-style-type: none"> 1. Brukeren velger å laste opp en videofil 2. Brukeren skriver inn søkeordet som skal finnes igjen 3. Brukeren velger hvilken del av videoen som inneholder tekst 4. Systemet deler videofilen opp i skjermbilder og filtrerer disse 5. De filtrerte bildene transkriberes av systemet 6. Systemet søker etter søknadsordet oppgitt av brukeren i transkribert tekst 7. Systemet finner bildene hvor søknadsordet ble funnet 8. Bildene med søkeordet i lagres til PDF og PDF-filen åpnes
Alternativ flyt #1:	<ol style="list-style-type: none"> 1. Brukeren velger å laste opp flere bilder lagret i en mappe 2. Brukeren skriver inn søkeordet som skal finnes igjen 3. Brukeren velger hvilken del av bildene som inneholder tekst 4. Systemet filtrerer bildene for å ikke få med duplisert innhold 5. Systemet transkriberer hvert bilde 6. Systemet søker etter søknadsordet oppgitt av brukeren i transkribert tekst 7. Systemet finner bildene hvor søknadsordet ble funnet 8. Bildene med søkeordet i seg lagres til PDF og PDF-filen åpnes

3 DOMENEMODELL

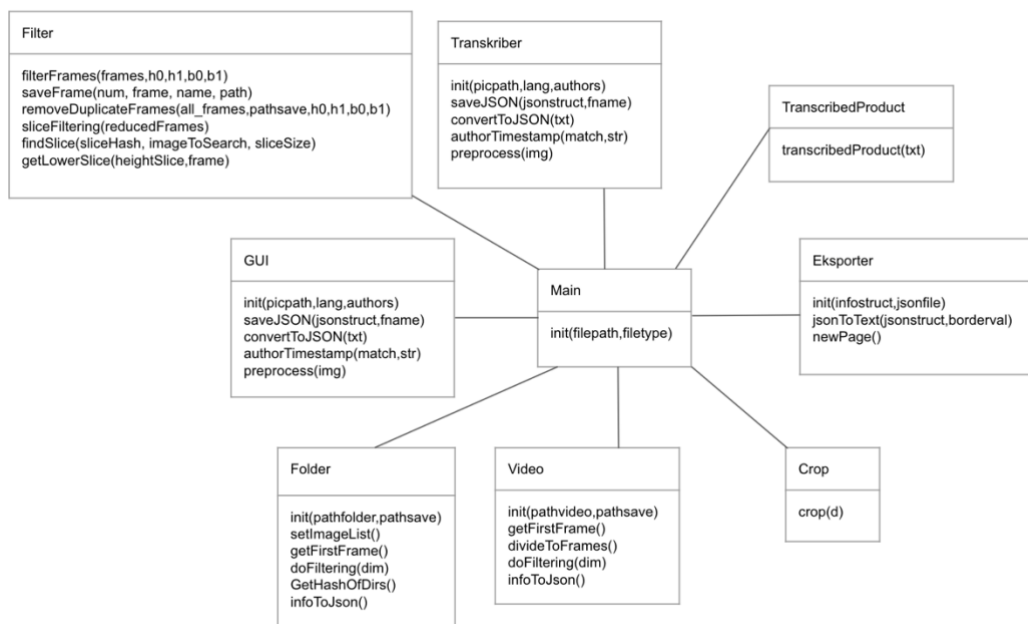
Domenemodellen lages for å få en oversikt over hvordan ulike konsepter i programvaren henger sammen. For dette prosjektet er det laget tre domenemodeller. En konseptuell domenemodell som ble opprettet i starten av prosjektet, en domenemodell for programvaren før de nye kravene kom, og til slutt en domenemodell som representerer programvaren med de nye kravene.

Den konseptuelle domenemodellen er vist i Figur 3 og inneholder klasser og klassevariabler som var forventet nødvendig for å løse oppgaven.



Figur 3 Konseptuell domenemodell

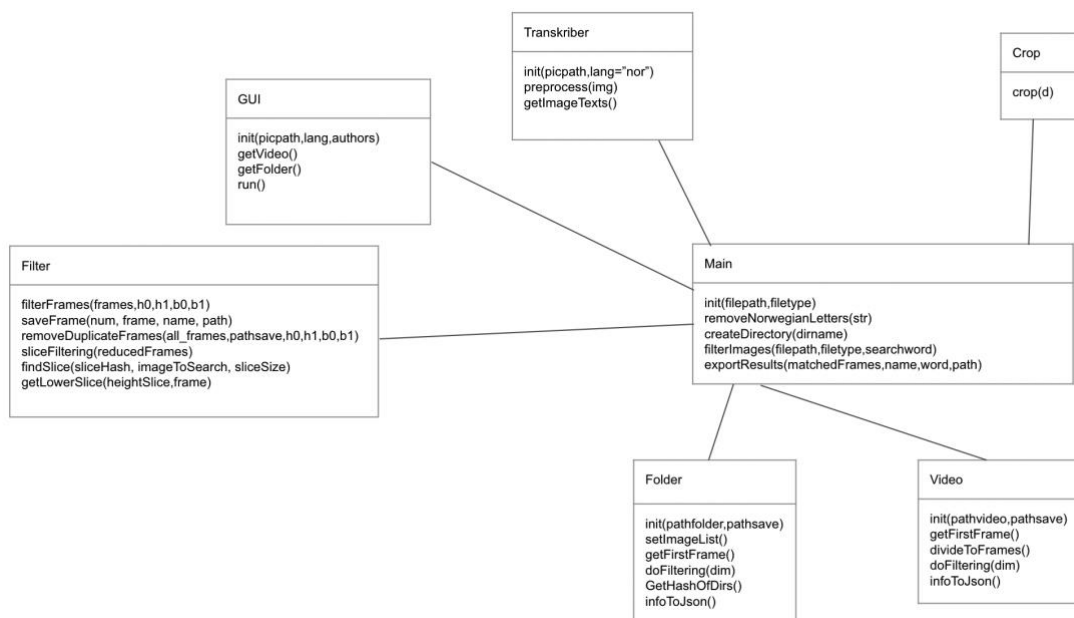
Figur 4 viser domenemodellen for programvaren som løser de originale kravene, og har flere forskjeller fra den konseptuelle domenemodellen. Det er blitt gjort endringer på klasser i løpet av prosjektet hvor oppgaver krevde flere klasser for å kunne løses. I tillegg ble klassene Emojis og Ordliste ikke med i programvaren grunnet tidsbegrensninger.



Figur 4 Domenemodell for opprinnelige krav

Domenemodell for den endelige programvaren til de oppdaterte kravene er vist i Figur 5.

Domenemodellen har mange likheter med domenemodellen i Figur 4. De største endringene gjelder forandringer i funksjonaliteten til GUI, Transkriber og at klassene TranscribedProduct og Eksporter ikke brukes lenger.



Figur 5 Domenemodell for oppdaterte krav

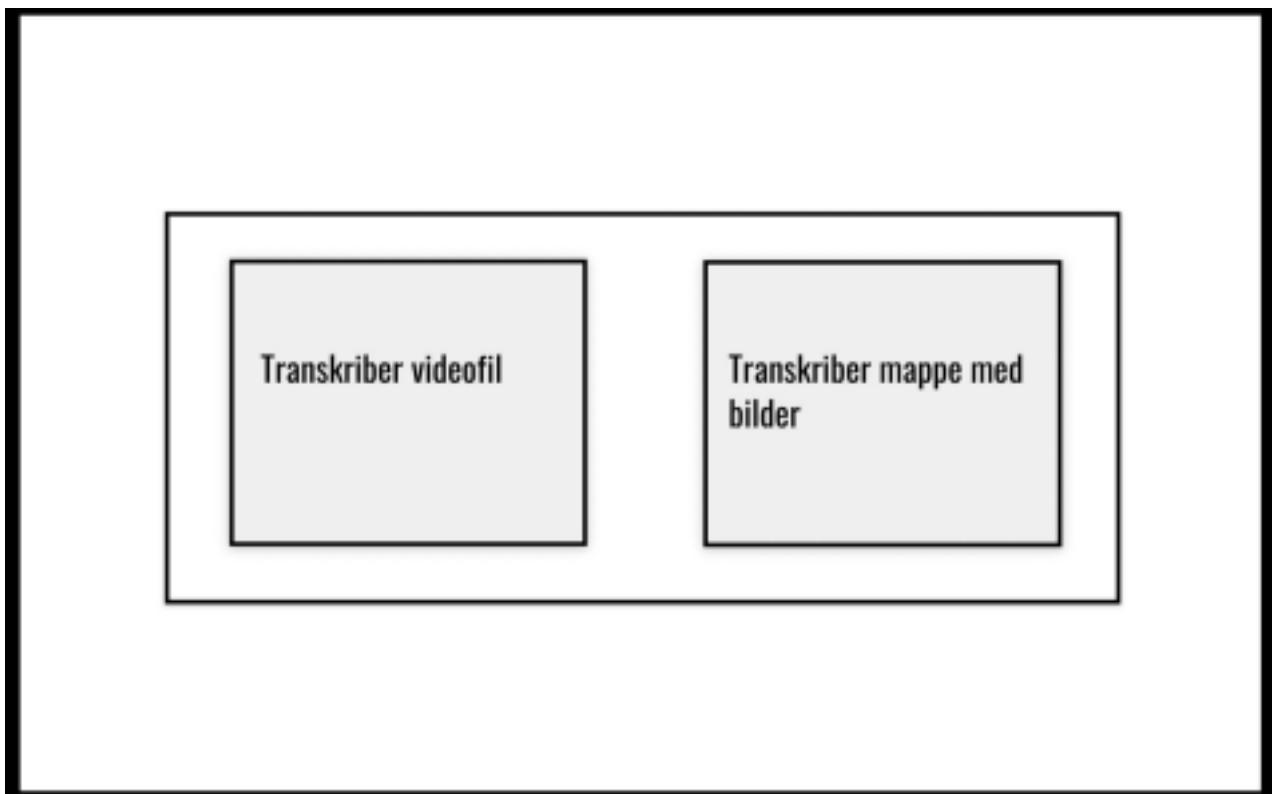
4 PROTOTYPER

4.1 Wireframes

Et av kravene til produktet er et brukervennlig brukergrensesnitt. Prototypen til brukergrensesnittet ble produsert med uizard, et wireframe verktøy. En interaktiv versjon av brukergrensesnittet er tilgjengelig på nettsiden til uizard¹.

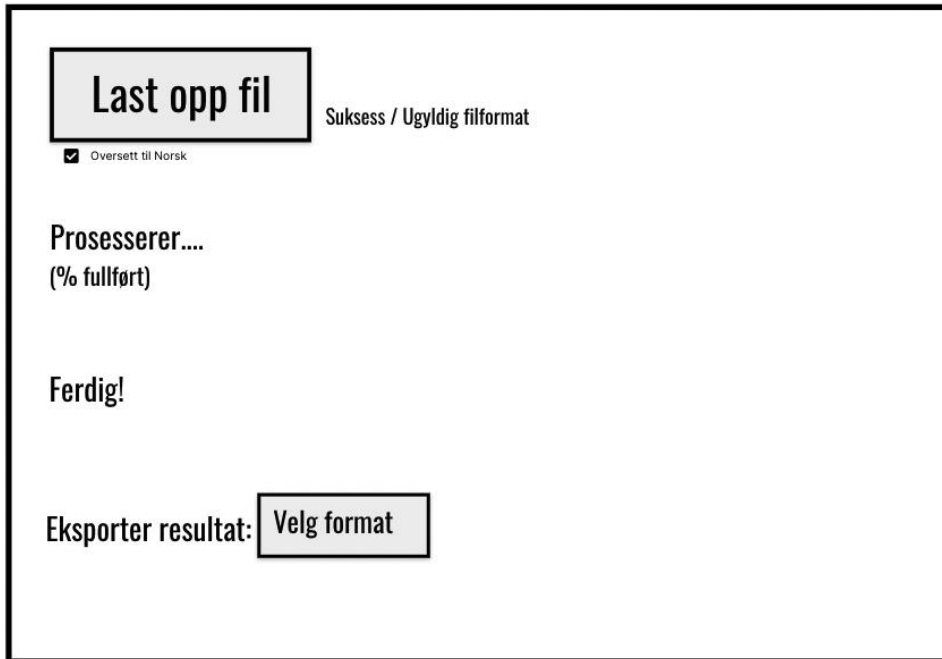
Prototypen beskrevet her er for programvaren til de originale kravene. Det er ikke blitt laget prototype for programvaren til de nye kravene da disse kom så sent i prosjektet at det ikke var tid til det.

Startsiden til brukergrensesnittet er vist i Figur 6. Dette er det første møtet brukeren har med programvaren og valgmulighetene er som beskrevet i brukstilfellet. De to alternativene brukeren kan velge mellom fører til like oppsett, vist i Figur 7, hvor eneste forskjellen er teksten på knappen “Last opp fil” som for mapper blir “Last opp mappe”.



Figur 6 Startsiden til brukergrensesnittet

¹ <https://app.uizard.io/p/548c4482>



Figur 7 Siden som man kommer til hvis det velges å laste opp video

4.2 HTML-prototyper

Det var ikke relevant for prosjektet å lage HTML-prototyper da Wireframes prototypene beskriver brukergrensesnittet. Det er og ikke relevant da programvaren som utvikles ikke er en web-applikasjon.

5 REFERANSER

Det er ikke brukt referanser i kravdokumentasjonen.