

# TDT4180 MMI 2022

---

## Øving 1: Designprinsipper

---

Gruppenummer: 89

Studenter:

NTNU epost	Etternavn	Fornavn
<b>haavasny@stud.ntnu.no</b>	Solberg Nybøe	Håvard
<b>olavslo@stud.ntnu.no</b>	Lorentzen	Olav Selnes
<b>andskag@stud.ntnu.no</b>	Skagen	Andreas
<b>sigurdls@stud.ntnu.no</b>	Lie Søvold	Sigurd

Dato: 14.02.2022      Antall ord: 3 272

## Don Normans Designprinsipper i Praksis: Vipps

### Introduksjon

Det å bruke ressurser på selve utformingen til et grensesnitt for en interaktiv løsning er blitt mer og mer aktuelt ettersom teknologi har blitt en essensiell del av hverdagen. I tillegg til å gjøre løsningen tilgjengelig og lett å bruke kommer det også høye krav av at systemer som tidligere var forbeholdt eksperter og fagpersoner nå er blitt pakket inn i enkle og kompakte applikasjoner for allmenn bruk. Et særlig tydelig eksempel på denne utviklingen er tjenester som lar enkeltpersoner ta kontroll over bank og betaling. Det er kanskje ekstra viktig at riktige hensyn blir gjort med denne typen tjeneste, da personer fra alle aldersgrupper og med alle tenkelige bakgrunner ofte blir nødt til å ta stilling til denne måten å drive betalinger på, særlig når man i samfunnet opplever at denne tjenesten er blitt helt sentral i alle sammenhenger.

Den løsningen vi skal kikke nærmere på er Vipps, den enerådende betalingstjenesten som man i Norge bruker i alle mulige sammenhenger, og i enkelte tilfeller ikke kommer seg unna. Vipps er også kjent for å være en svært godt designet tjeneste (DOGA, 2021), noe som gjør at et dypdykk i grensesnittet med bakgrunn i Don Normans designprinsipper kan bli en svært verdifull øvelse.

### Vipps

Vipps er en betalingstjeneste som i utgangspunktet har en veldig godt definert oppgave. I en tid hvor måten man gjør daglige betalinger på har blitt heldigital skal Vipps gjøre det så enkelt som overhodet mulig å få gjort dette. Naturligvis er det mange spennende problemstillinger tilknyttet infrastrukturen og sikkerheten bak en slik løsning, men det stilles også svært høye krav til at applikasjonen faktisk er intuitiv og lett å bruke.

En bruker får oppgitt informasjon om en mottaker, enten via telefonnummer, Vipps-nummer, eller QR-kode. Brukeren skal så logge seg inn i Vipps, legge inn denne informasjonen, beløpet som skal betales, og så gjennomføre betalingen. Vipps tilbyr også ekstra funksjoner, som automatisk fordeling av beløp, transaksjonshistorikk, og automatisert betaling av regninger.

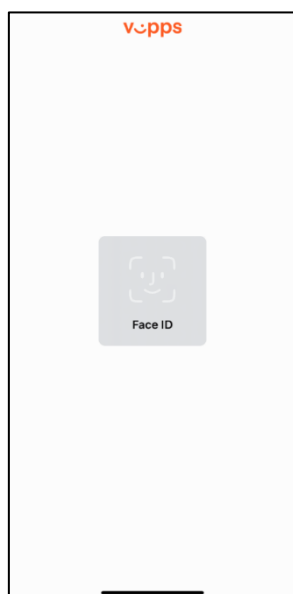
### Hvordan bruker man Vipps? Eksempel: Vennebetaling

En typisk sammenheng man bruker Vipps i er når man har vært ute og spist middag med en venn. Én person betaler regningen til spisestedet, så *vippser* den andre personen sin andel av betalingsbyrden til den som betalte. Gitt at denne personen vet telefonnummer til den andre personen, og beløpet som skal betales, så kan vi nå gå gjennom hvordan denne interaksjonen vil forløpe.

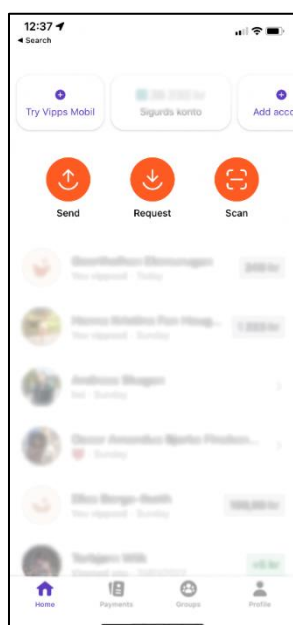
Det første brukeren møter når applikasjonen åpnes er innloggingsskjermen. Her er det tydelig at det spørres om en kode, men det kan også hende at brukeren har konfigurert Vipps til å akseptere biometri, som vist i **Figur 1**.

Etter brukeren har logget inn blir brukeren møtt med forsiden, som har knapper som tilsvarer hovedfunksjonene, betalingshistorikk, og koblinger til annen funksjonalitet. Brukeren skal sende et beløp, og trykker på 'Send', markert med pil i **Figur 2**.

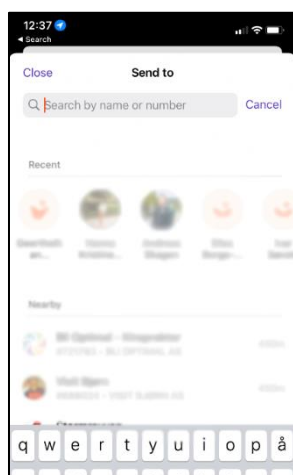
Brukeren skal nå oppgi mottaker, og får muligheten til å skrive inn mottakerens informasjon i søkefeltet, eller å velge en nylig kontakt. Personen skriver telefonnummer i feltet der det står 'Søk på navn eller nummer', og velger mottaker. Dette er markert i **Figur 3**.



**Figur 1**  
Innloggingsskjerm.



**Figur 2** Forside.



**Figur 3** "Send til"-skjerm.

Brukeren velger nå beløpet som skal betales, og legger til meldingen «Spleise på middag». Brukeren trykker seg så videre. Dette er vist i **Figur 4**.

Brukeren får nå oppgitt et kort sammendrag av betalingen, som inkluderer mottaker, beløp, og hvor pengene kommer fra (kontonummer). Brukeren trykker så 'Send', og autoriserer betalingen på samme måte som brukeren logget inn i Vipps. Dette er vist i **Figur 5**.

Betalingen er nå gjennomført, og man får en kvittering. Brukeren blir så returnert til forsiden igjen (**Figur 1**) med den nye betalingen nå listet i historikken.

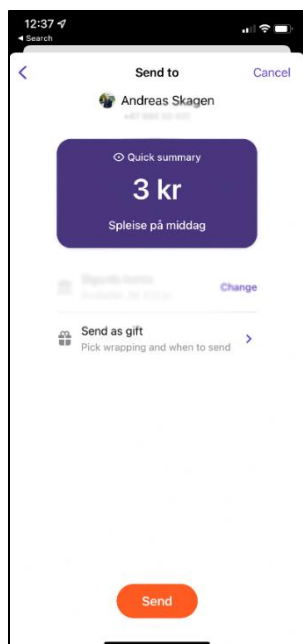
## Vipps og Don Normans' Designprinsipper

### Visibility

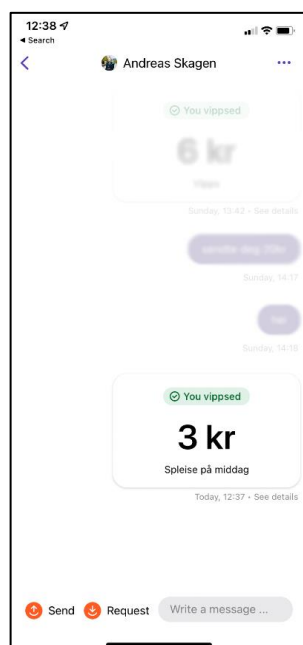
For at Vipps skal kunne påta seg rollen som enerådende betalingstjeneste er det kritisk at applikasjonen har et intuitivt brukergrensesnitt. Det er tydelig at Vipps har vært nøye i sine designvalg både med tanke på det estetiske, og det brukervennlige. De har sørget for at man bare kan interagere med en type tjeneste av gangen, og bruker inndeling av undersider for å oppnå dette. Denne typen designvalg går under det Norman definerer som *visibility*. Tanken bak dette designprinsippet er å tydeliggjøre bruksområdene til et produkt gjennom hva slags elementer og informasjon som vises, ofte ved hjelp av farger, relative størrelser, og plassering (Sharp *et. al.*, 2019).

En gjennomsnittlig bruker av Vipps er ikke nødvendigvis interessert i mange spesialfunksjoner. Derfor velger utviklerne å lage en enkel hovedside med relativt begrenset funksjonalitet. I **Figur 2** ser man Forsiden, som er det første brukeren møter når appen åpnes. Man får da umiddelbart presentert den sentrale funksjonaliteten. 'Send'-knappen brukes til å sende penger, 'Request'-knappen brukes til å be om penger og 'Scan'-knappen brukes til å skanne QR-koder eller legge inn kvitteringer i ditt personlige arkiv. Utover dette får brukeren oversikt over hvilken konto som er koblet til tjenesten, og man kan også legge til både nye kontoer og vipps sitt eget mobilabonnement. Vi har altså det som kan betraktes som nøkkelfunksjoner og nøkkelinformasjon på Forsiden, og lite annet.

En fordel med å ha en såpass enkel hovedside har fordelen av at flere brukere blir i stand til å bruke tjenestene, samtidig som at det går for å lære seg hvor ting er, og hva det gjør. Er man interessert i andre funksjoner kan man navigere applikasjonen ved hjelp av ikonmenyen på bunnen av skjermen (**Figur 2**). Når man aktivt må lete etter de andre funksjonene i appen, så sørger dette for at brukere som er interessert i mer avanserte funksjonene kan finne dem ved behov,



**Figur 4 Sammen drag, betaling.**



**Figur 5 Betalingshistorikk.**

uten at en typisk bruker kun interessert i nøkkelfunksjoner trenger å ta stilling til disse. Det å begrense informasjonen vi har på hver side, i tillegg til sterke kontraster mellom aktiverbare og passive element bidrar til at Vipps følger tett opp mot visibility-prinsippet. Vipps er en applikasjon som er lett å navigere, samtidig som at ytterlig funksjonalitet gjemmes bort uten at det blir vanskelig for en brukere og oppsøke disse.

### Feedback

Vipps tilbyr tjenester som i utgangspunktet utsetter brukeren for stor risiko, da ansvaret faller på dem når det gjelder korrekthet i betalingsinformasjon og beløp. Det er derfor særlig viktig at brukeren får så mye hjelp som mulig til å forstå hvilke funksjoner som er aktivert, og hvordan applikasjonen svarer på input. Derfor stilles det særlig høye krav til at prinsippet om *feedback* ivaretas, slik at det aldri rår usikkerhet over hva som skjer når brukeren gir input.

Feedback er nært relatert til prinsippet om visibility, og handler om å gi brukeren visuell bekreftelse på handlingene den utfører (Sharp *et. al.*, 2019). Det handler altså om endringer som indikerer for bruker at noe har skjedd, enten ved hjelp av endringer blant annet i utseende (visibility), lyd, vibrasjon, og animasjon. Om en handling ikke gir respons før etter en god stund, vil det føles håpløst for bruker å interagere med programmet, da bruker trenger nødvendig respons for å være sikker på at programmet fungerer som det skal. Feedbacken bør også stemme overens med handlingen den er knyttet til. For eksempel vil et høyt «pling» hver gang en knapp trykkes ned, kun være distraherende og plagsomt for bruker, mens en liten vibrasjon sammen med små endringer i farge og størrelse vil i mange tilfeller være nok bekreftelse for bruker.

Det kan være slitsomt å måtte autorisere seg hver gang man bruker appen, selv om dette har et viktig sikkerhetsgrunnlag. Det å opprettholde høy sikkerhet samtidig som man skaper en god brukeropplevelse kan derfor være krevende. På innloggingssiden (**Figur 1**) har brukeren valgt mellom å logge inn med kode eller “Face-ID”. Hvis brukeren har valgt innlogging med kode, dukker det opp et siffer-tastatur og fire tomme, grå bokser. Siffer-taster som trykkes ned får en mørkere gråtone, og for hvert trykk får en av de grå boksene en hvit sirkel i seg som signaliserer at et noe er tastet inn. Her treffer man gjerne første gang på hvordan applikasjonen håndterer innlasting ved hjelp av en dominerende animasjon. Denne består av noen prikker spretter opp og ned og fluktuere i oransjetone, for å gi brukeren bekreftelse på at noe skjer. Feedback på innlasting er noe man stort sett bare finner i software, men det er samtidig blitt en helt sentral del av opplevelsen da de fleste brukere har som forventning at man får umiddelbar feedback på handlinger, og at i mangel på dette så kan det mistenkes at en teknisk feil har skjedd. Hvis koden er feil, tas man tilbake til innloggingssiden, hvor det nå også dukker opp en liten rød setning som forteller brukeren at koden var gal, som gir både god feedback og instruks på hvordan brukeren kan gå videre.

På Forsiden (**Figur 2**) har vi mye av funksjonaliteten i Vipps. Dersom bruker ønsker å ta i bruk en funksjon, så er gjerne det første de gjør å trykke på det som ser ut som knapper. Bekreftelsen på dette kommer når knappen får en mørkere oransjetone, samtidig som den minker i størrelse. Når knappen slippes, gir mobilen fra seg en liten vibrasjon, og alt dette skal gjenskape følelsen av å trykke på en fysisk knapp. Brukeren får deretter klar beskjed om at funksjonen er aktivert ved at en ny skjerm skir opp fra bunnen (**Figur 3**). Hvis brukeren trykker på søkefeltet dukker det opp en blinkende, vertikal markør i starten av søkefeltet og et tastatur på nedre halvdel av skjermen. Resten av siden blir uklar, slik at bruker får sterk feedback på at hva som har skjedd, og hva man skal gjøre videre (visibility). Etter at mottaker er valgt skir betalingssiden (**Figur 4**) inn fra høyre, som igjen gir et kraftig signal om progresjon. Tekstfeltet hvor bruker skriver inn beløp er allerede markert, noe som bruker ser ved at vi igjen har markøren i feltet og ved at et siffer-tastatur har dukket opp på skjermen. Før man faktisk sender pengene til motparten, er det flere sider med bekreftelse, som eksplisitt forteller brukeren hva som er i ferd med å skje.

Gjennom hele denne prosessen møtes brukeren av knapper som indikerer at de blir trykket, og applikasjonen responderer med tydelige animerte skjermbytter, animert innlasting, lyder, vibrasjoner, og flere instanser av forklarende tekst som gjør det åpenbart for brukeren hva som har skjedd, og hvilken input som fikk dette til å skje. Det er med andre ord tatt godt hensyn til feedback-prinsippet, noe brukeren gjerne ikke legger merke til med mindre det er mangelfullt.

### Affordance og Mapping

Når man tar for seg hva objektene i grensesnittet inviterer til og i hvilken grad de er selvforklarende til hvordan de fungerer, så er det viktig å ta i betraktning hvordan man er opplært til å interagere med en smarttelefon med touch-display. Vi kan anta at når brukeren først tar i bruk applikasjonen, så tar man i bruk de samme prinsippene som man møter i bruk av enheten ellers. Altså kan man forvente at hvis man trykker bestemte steder på skjermen, eller drar langs skjermen i en retning, så er dette måter man kan gi input til applikasjonen. Dette er grunnlaget for det Norman kaller *perceived affordances*, altså den selvforklarende egenskapen virtuelle objekter har til hvordan de fungerer (Sharp *et. al.*, 2019). I motsetning til *real affordances* som fysiske gjenstander gjerne har, så er man i stor grad avhengig av at brukeren følger konvensjoner for hvordan man kan forvente at virtuelle objekter kan interageres med. *Mapping* er et prinsipp som ligger veldig nært affordance, men der affordance tar utgangspunkt i forventningene til brukeren fra tidligere erfaring, handler mapping om de faktiske hendelsene som oppstår når brukeren interagerer med applikasjonen (Sharp *et. al.*, 2019). Et tegn på god mapping er at funksjonene i applikasjonen svarer til brukerens forventninger ut ifra hva man oppfatter at applikasjonen skal gjøre og hva man er kjent med fra liknende applikasjoner. Riktige forventninger kan også indikere god affordance.

Gitt at man blir presentert med forsiden (**Figur 2**), så skiller det seg ut enkelte objekter. Sentralt ser man tre store oransje sirkler med ikoner og tekst under. Her får vi et perfekt eksempel av hvilke konvensjoner det er ment at brukeren følger. Sirklene inviterer til at man trykker på dem, da de ser ut som knapper både i utseende, og fra erfaring for hvordan knapper ser ut i applikasjoner. De har en kraftig farge, og er omringet av mye tom plass. Om man antar at brukeren kjenner til den sentrale tjenesten Vipps tilbyr, så blir dette bekreftet i form av tre punkter som ligner på knapper midt på skjermen. Brukeren forventer å finne funksjonene, ser det som ser ut som knapper, som brukeren av konvensjon tenker at er knapper, og som oppfører seg som knapper når de trykkes på. Knappene har også flere måter å utføre samme funksjon på, som kan være til nytte for brukeren hvis den er kjent med ulike metoder fra andre

tjenester. På nylige kontakter (**Figur 2**) kan man for eksempel enten trykke, dra til siden eller holde inne for å få en meny for å velge hvordan man vil interagere med kontakten.

Den neste konvensjonen kommer til nytte når brukeren skal navigere informasjon som tar mer plass enn det er tilgjengelig på en skjermstørrelse, som for eksempel betalingshistorikken vist under knappene i **Figur 2**. På samme måte som man navigerer nettaviser, eller leter seg fram i lister over innstillinger på telefonen, så kan man da tenkes at for å undersøke innholdet i betalingshistorikken, så drar man i den delen av skjermen. At denne konvensjonen gjelder for dette objektet blir også umiddelbart bekreftet av umiddelbar feedback i form av at listen flytter seg og følger bevegelsen til brukeren. Dette illustrerer godt hvordan bevisst hensyn til flere designprinsipper drastisk forbedrer kvaliteten til objekter i et grensesnitt.

### Constraints

*Constraints* handler om, i ordets rette forstand, å begrense brukerens valg med tanke på handlinger, for å gjøre det lettere å bruke applikasjonen slik den er ment og styre brukeren i riktig retning (Sharp *et. al.*, 2019). En god begrensning er med på å øke brukeropplevelsen, og gjør at brukeren lettere forstår hvordan applikasjonen skal brukes. En dårlig begrensning hindrer brukeren i å gjøre en handling de ser som innlysende, og kan svekke brukeropplevelsen sterkt.

Vipps har noen intensjonelle begrensninger, som bestemmer hva brukeren kan og ikke kan gjøre. For eksempel får man bare opp et *tall-tastatur* når man enten skal betale eller skrive inn et kontonummer. Selv om man prøver å tvinge frem et vanlig tastatur får man ikke til å tase inn bokstaver der det skal være tall. Denne begrensningen hjelper brukeren med å finne ut av hva man kan skrive hvor, og hva som skal fylles ut i tekstfeltet. En annen begrensning er at når man skal taste inn et kontonummer stopper feltet å ta imot tall etter 11 siffer. Denne begrensningen, i likhet med den forrige nevnte, gjør brukeren mer oppmerksom på hva man kan og ikke kan taste inn i feltet.

### Consistency

Begrepet *consistency* handler om at elementer som av brukeren oppfattes like, burde ha samme funksjon, og at programmet burde være konsekvent med seg selv på tvers av ulike plattformer som f.eks. web, iOS og Android (Sharp *et. al.*, 2019).

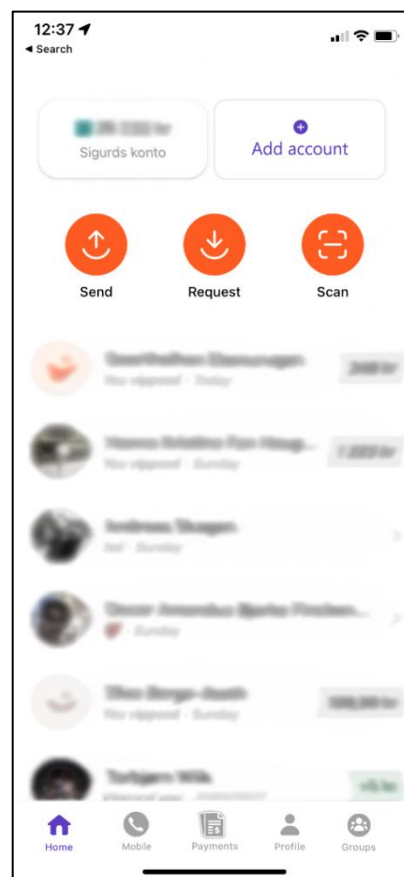
På tvers av plattformene (Android og iOS) har applikasjonen noen små ulikheter som kan ha påvirkning på brukeropplevelsen. I Android-versjonen får man blant annet feedback i form av en *skygge-boble* når man når toppen eller bunnen mens man blar, mens man i iOS har muligheten til å få opp en meny med kontakter når man holder inne enten 'send'- eller 'be om'-knappen. Disse forskjellene er ikke av veldig stor betydning når det gjelder Vipps, men slike ting kan påvirke brukeropplevelsen for brukere som følger konvensjoner som de tror er universelle, men som egentlig er spesielle for en plattform. Dette kan ha tilknytning til små forskjeller i hvordan brukere interagerer på iOS og Android, som i så fall utgjør consistency gjennom plattformen.

Metodene for innlogging er et godt eksempel på consistency på tvers av alle applikasjoner med autorisering ved at det tas i bruk standardiserte metoder, altså PIN-kode, Touch-ID eller Face-ID. Hvis bruker har aktivert en av disse innloggingsmetodene er det sannsynlig at bruker er kjent med disse fra før, og dermed vet hvordan disse skal brukes. Det at autorisering foregår nærmest identisk gjennom alle applikasjoner indikerer at det ligger dype konvensjoner i hvordan man designer grensesnitt på telefoner med touch-display.



## Hva kan bli bedre?

Gitt at man er så avhengig av konvensjoner i hvilke objekter som kan interageres med og hva man kan forvente at de gjør på en smarttelefon, så er det utrolig viktig at man er konsistent i dette. Vi har påpekt at vi har to typer knapper i applikasjonen. Først har vi knapper som tar oss gjennom selve Vipps tjenesten, markert i oransje. Ellers så har man noen litt mindre synlige knapper som lar brukeren navigere mellom forskjellige menyer, men som kritisk ikke aktiverer betalings-funksjonaliteten. Derimot så har man funksjoner som man direkte kan aktivere fra forsiden, men som har et utseende ulikt begge de overnevnte knappetyperne. I båndet øverst (**Figur 2**) har man noen rektangler som både viser informasjon, men og har tekst som inviterer til at brukeren skal trykke. Dette er ikke sentrale tjenester man forventer at Vipps har (eksempelvis 'Se saldo'), så det er naturlig at de ikke har samme utseende som 'Send' osv. Likevel så er det ikke tydelig at dette er objekter som tar oss til en meny, eller at de overhodet kan trykkes på. Dersom man kan tillate at rektangelet som beskriver områder hvor 'Try Vipps Mobil' faller bort, så kan man i stedet vise kun vise kontonummer og saldo. Det er videre naturlig at funksjonaliteten tilknyttet Vipps Mobil får sin egen kategori på bunnen ved siden av annen øvrig funksjonalitet. Dersom brukeren ønsker flere kontoer på toppen, så bevarer vi 'pluss'-knappen.



Figur 6 Alternativ Forside.

Videre kan man kikke på ikonografien som blir brukt. Ikonene er ofte det eneste man tar stilling til før man gjør et valg for hvor man ser etter en funksjon. Derfor er det særlig viktig at disse er så tydelige som mulig for hva funksjonen er. Det eksisterende ikonet for funksjonaliteten tilknyttet 'Payments' (nederst på **Figur 2**) viser et ark med noe tekst. Dette kan bli for generisk når man er interessert i én type dokumentasjon, altså regninger. Vårt forslag er å ha noe som viser at det er snakk om finansielle dokumenter, så en logo med et valutasymbol kan gjøre funksjonaliteten mer åpenbar. I kombinasjon med et ryddigere bånd på toppen illustrerer **Figur 6** en Forside vi mener er mindre rotete, og mer relevant for de fleste brukere.

## Konklusjon

Vipps omfavner eksisterende konvensjoner for hvordan man bruker touch-baserte applikasjoner. Det er tydelig at det er tatt hensyn til designprinsippene, men det er uklart i hvor stor grad Vipps har ansvar for gode designvalg. Design av mobile grensesnitt virker å være standardisert på tvers av applikasjoner, fordi så mye av brukerens forventninger og vaner kommer av konvensjon. Videre er det åpenbare problemstillinger tilknyttet presentasjon og navigasjon på et så lite display, noe som gir grunnlag for at denne typen grensesnitt nærmest har en optimalisert formel. Vipps har likevel noe designfrihet i en sfære såpass preget av konvensjon. Dette er organisering av funksjoner, valg av farger, kontraster, og ikoner, og plassering av (ellers individuelt optimaliserte) objekter. På disse områdene finner vi lite som kan gjøres bedre, og vi håper vi har gjort det klart at Vipps representerer utmerket anvendelse av Normans designprinsipp, samtidig som det kaster lys på hvordan disse prinsippene har formet design for smarttelefoner på tvers av plattformen.

## Referanser

Design og arkitektur Norge (2021). *Vipps til topps i inkluderende innovasjonskonkurranse*. Tilgjengelig via <https://doga.no/aktuelt/vipps-mottar-pris-for-inkluderende-tjenstedesign/> (sist besøkt 09.02.2022).

Sharp, H., Rogers, Y., Preece, J. (2019). *Interaction Design, 5th ed.* Wiley.