



IS-304: 2024

Bachelorprosjekt Phonero

Emnekode	IS-304
Emnenavn	Bacheloroppgave i Informasjonssystemer
Emneansvarlig:	Hallgeir Nilsen
Veileder	Hallgeir Nilsen
Oppdragsgiver.	Phonero

Studenter:

Etternavn	Fornavn
Lund	Laurits
Fjelltun	Markus
Worren	Ottar
Bouzidi	Zeyd
Tra	Linh
Skau	Markus

Jeg/vi bekrefter at vi ikke siterer eller på annen måte bruker andres arbeider uten at dette er oppgitt, og at alle referanser er oppgitt i litteraturlisten.	Ja X	Nei
Kan besvarelsen brukes til undervisningsformål?	Ja X	Nei
Vi bekrefter at alle i gruppa har bidratt til besvarelsen	Ja X	Nei

Kampanjemotor- webapplikasjon

Bacheloroppgaven med Phonero

Laurits Lund

Linh Phuong Tra

Markus Fjelltun

Markus Skau

Ottar Nordal Worren

Zeyd Abdulla Bouzidi

VEILEDER

Hallgeir Nilsen

Universitetet i Agder, 2024

Fakultet for samfunnsvitenskap

Institutt for IT og Informasjonssystemer

Abstrakt

Denne rapporten omhandler et studentprosjekt som ble gjort våren 2024. Prosjektet er en praktisk bacheloroppgave i samarbeid med UiA og Phonero.

Gruppen som har utført prosjektet og laget denne prosjektrapporten har vært i tett samarbeid med oppdragsgiveren for å utvikle et system som skal gjøre det lettere å distribuere kampanjer for bedriften. Dette systemet skal overtas av bedriften når semesteret er ferdig og implementeres for å bli brukt til dets formål i daglig drift. Formålet med rapporten er å gjenspeile gruppens gjennomføring, beskrive prosjektet og systemet, samt reflektere rundt valg, erfaringer og utfordringer som har oppstått.

Gjennom datainnsamling og samtaler med oppdragsgiveren tilegnet gruppen seg en oversikt over hvilke deler systemet skulle inneholde, hvilke funksjoner det skulle ha og hvordan designet skulle være. Dette var veldig viktig for å legge et godt grunnlag for resten av prosjektet. Her fikk gruppen god hjelp av ansatte i Phonero, som deltok på møter, diskuterte system detaljer og svarte på alle spørsmålene som ble stilt.

Gruppen har brukt teknologier som C#, .NET, React, TypeScript og MS SQL for å utvikle systemet. Dette var teknologier som ble valgt på bakgrunn av tidligere kjennskap, men også på grunn av hva bedriften brukte. Dette systemet skal som sagt videreutvikles og implementeres.

Mange av systemkravene ble nådd, men ikke alle. Gruppen utarbeidet en MoSCoW, i samarbeid med bedriften, der alle 'Must have's' og noen 'Could have's' ble oppnådd.

Resultatet gruppen står igjen med er en fungerende MVP med database, API og web-klient. Bacheloroppgaven har vært en utrolig lærerik reise, der samtlige medlemmer har gjort erfaringer om prosjektgjennomføring og prosjektstyring. Samtlige medlemmer har også tilegnet seg mye kunnskap om nye og kjente teknologier.

Forord

Denne rapporten gjenspeiler innsatsen til samtlige medlemmer, i utførelsen av en Bacheloroppgave i IT og Informasjonssystemer våren 2024.

Gruppen vil takke Hallgeir, veileder, for god kommunikasjonsflyt og innsats gjennom hele prosjektet. Hallgeir har vært tilgjengelig for alle spørsmål og møtt opp på kontorene til Phonero for å gi veiledning og svare på spørsmål vedrørende prosjektet.

En stor takk rettes også til oppdragsgiver, Phonero. Fra dag en har gruppen blitt tatt godt imot og følt seg som en del av bedriften. Gruppen har fått egne kontor plasser, lunsj og oppsett med utstyr, noe som har vært utrolig givende og med all sannsynlighet har hatt utfall på resultatet.



Vi vil spesielt takke Erik L. Berntsen, Kristian Hagestad og Bård Eik for oppfølging og deltakelse på møter. Samtlige personer har stilt opp og besvart alle typer spørsmål som gruppen trengte for å gjennomføre prosjektet. Dere er også en stor grunn for at vi har følt oss så godt ivaretatt når vi har vært tilstede på Phonero i alle disse ukene. Vi vil også takke Sunniva Finne, Håkon Moen, Emilie Kallhovd, Martin Blikstad og resten av de ansatte hos Phonero for deres vennlighet og deltakelse på brukertesting, møter og gode besvarelser på spørsmål.

Kristiansand

15.05.2024


Abdullah Zeyd Bouzidi


Laurits Jonas Lund


Markus Løvås Fjelltn


Markus Skau


Ottar Nordal Worren


Phuong Linh Tra

Innholdsfortegnelse

1.0 - Introduksjon.....	1
1.1 - Samarbeidsbedriften.....	1
1.2 - Definisjoner.....	1
1.3 - Gruppen.....	2
1.4 - Mål og ambisjoner.....	2
2.0 - Prosjektbeskrivelse.....	3
2.1 - Prosjektdefinisjon.....	3
2.2 - Systemkrav.....	3
3.0 - Prosjektstyring.....	4
3.1 - Agil Prosjektmetode.....	4
3.2 - Prosjektstyringsverktøy.....	5
3.2.1 - Jira.....	5
3.2.2 - Bitbucket.....	6
3.3 - Prioritering.....	7
3.4 - Kvalitet.....	8
3.4.1 Programvarekvalitet.....	8
3.4.2 - Kvalitetsfaktorer.....	9
3.4.3 - Kriterienes viktighetsgrad for Prosjektet.....	10
3.4.4 - Kvalitet og Kvalitetssikring i Prosjektgjennomføring.....	11
3.5 - Risikovurdering.....	11
4.0 - Analyse.....	13
4.1 - Datainnsamling.....	13
4.2 - Brukerhistorier.....	13
4.2.1 - Brukerhistorie #1.....	14
4.3 - FACTOR analyse.....	14
4.4 - Kvalitetssikring.....	15
5.0 - Design.....	16
5.1 - Designverktøy.....	16
5.1.1 - UXPin.....	16
5.1.2 - Figma.....	16
5.1.3 - Lucidcharts.....	17
5.1.4 - Draw.io.....	17
5.2 - Wireframe.....	17
5.3 - Mockup.....	17
5.4 - Brukertesting.....	18
5.5 - Designprinsipper.....	18
5.6 - Prototype.....	23
5.7 - Navigasjonskart.....	24
5.8 - Kvalitetssikring.....	24

5.8.1 - Kvalitetssikring av brukertesting.....	25
6.0 - Arkitektur.....	25
6.1 - Struktur.....	25
6.1.1 - Single Page Applikasjon (SPA).....	25
6.1.2 - Mappestruktur.....	26
6.2 - Database.....	29
6.3 - Bakgrunnsprosesser.....	29
6.3.1 - Funksjonalitet.....	30
6.4 - Kvalitetssikring.....	30
7.0 - Prosjektgjennomføring.....	30
7.1 - Rapportering.....	30
7.1.1 - Rapportering og Møter.....	31
7.1.2 - Fremgangsplan.....	31
7.2 - Utviklingsmiljø.....	32
7.2.1 - Programvare.....	32
7.2.2 - Programmeringsspråk.....	33
7.2.3 - Rammeverk.....	34
7.2.4 - Biblioteker.....	34
7.2.5 - Pakkebehandling.....	35
7.3 - Kompetansebygging.....	36
7.4 - Kvalitetssikring.....	37
7.5.1 - Prosjektstyring.....	37
7.5.2 - Utviklingsmiljø.....	37
7.5.3 - Enhetstesting.....	38
8.0 - Refleksjon.....	39
8.1 - Prosjektgjennomføring.....	39
8.1.1 - Agil prosjektmetodikk.....	39
8.1.2 - Gruppeinndeling.....	39
8.1.3 - Kommunikasjon.....	40
8.2 - Kvalitet og Kvalitetssikring.....	40
8.3 - utfordringer og Erfaringer.....	40
8.4 - Resultat / konklusjon.....	41
8.5 - Uttalelse fra Oppdragsgiver.....	42
Kilder.....	43
Appendiks.....	48

Figurliste

Figur 1: Diagram illustrerer Scrumban (Nearshore, 2022).....	5
Figur 2: Kanban på Jira.....	6
Figur 3: Grenstruktur på Bitbucket.....	6
Figur 4: Dashbord side.....	19
Figur 5: Kampanje-tabell.....	20
Figur 6: Kampanjeinformasjon.....	21
Figur 7: Knapper.....	22
Figur 8: ‘Stepper’ navigasjon.....	22
Figur 9: Bekreftelse for videresend til godkjenning.....	23
Figur 10: Brukere kan fortsette å fullføre kampanjen eller lagre som utkast.....	23
Figur 11: Navigasjonskart av kampanjemotor-webapplikasjonen.....	24
Figur 12: Illustrasjon av SPA.....	26
Figur 13: Mappediagram som beskriver øverste laget.....	27
Figur 14: Mappediagram som beskriver klient laget.....	27
Figur 15: Mappediagram som beskriver server laget.....	28
Figur 16: Mappediagram som beskriver Daemon laget.....	28
Figur 17: Database av kampanjemotor-webapplikasjonen.....	29

Tabelliste

Tabell 1: Definisjoner.....	2
Tabell 2: Ansvarsfordeling.....	2
Figur 2: Kanban på Jira.....	6
Figur 3: Grenstruktur på Bitbucket.....	6
Tabell 3: MoSCoW av funksjonene i webapplikasjonen.....	8
Tabell 4: ISO/IEC 25010:2023 Kvalitetsfaktorer.....	10
Tabell 5: Kritikalitetsmatrise.....	12
Tabell 6: Brukerhistorie 1.....	14
Tabell 7: FACTOR analysen.....	15
Tabell A1: Risiko og Respons.....	49
Tabell A2: Brukerhistorie 2.....	50
Tabell A3: Brukerhistorie 3.....	50
Tabell A4: Brukerhistorie 4.....	51
Tabell A5: Brukerhistorie 5.....	52

1.0 - Introduksjon

Dette kapittelet gir en forståelse av prosjektets mål, kontekst og definisjoner. Bedriften gruppen samarbeider med, Phonero, som er en ledende tele- bedrift basert i Kristiansand og andre steder i landet, vil bli introdusert. Videre vil nøkkelbegrep som blir benyttet i rapporten definert for å sikre en klar forståelse av rapportens beskrivelse av prosjektet.

Videre presenteres gruppen bak prosjektet; sammensetning og ansvarsområder. Avsluttende, beskrives målsettingene og ambisjonene som gruppen har satt for prosjektet. Dette med fokus på å levere en webapplikasjon som bedriften vil være tilfreds med. Kapittelet danner grunnlaget for resten av rapporten gjennom å etablere rammene og konteksten som er gitt for prosjektet.

1.1 - Samarbeidsbedriften

Phonero er et teleselskap med totalt 150 ansatte. De har hovedkontor i Kristiansand, med avdelinger i Oslo, Trondheim, og Stavanger. Phonero fokuserer på bedriftsmarkedet og offentlig sektor ved å tilby pakkeløsninger på mobilabonnementer til sine kunder (Phonero, u.å.). Phonero har siden 2016 vært en del av Telia Norge AS.

Phonero har gjennom hele prosjektets løp vært imøtekommende med gruppen, og tilbudt en rekke ressurser som har vært svært nyttig. Gruppen har blant annet fått tilgang til Phoneros lokaler, egen kontorplass med datamaskin for arbeidsbruk til samtlige gruppemedlemmer, og gratis lunsj hver dag. Gruppen har også fått tilgang til Phoneros utviklermiljø, med en rekke tjenester som gruppen har tatt i bruk i løpet av prosjektet.

1.2 - Definisjoner

Her ønsker gruppen å definere en rekke begrep som vil bli brukt i denne rapporten, og som kan oppleves som vanskelig å forstå for en leser som ikke allerede er godt kjent med prosjektet, eller som blir brukt på en måte spesifikt til dette prosjektet.

Begrep	Definisjon
Webapplikasjonen / Kampanjemotoren	Systemet som gruppen skal utvikle for Phonero i prosjektets løp.
Bedriften	Bedriften som gruppen har prosjekt til; Phonero.
Ditt Phonero	Phoneros kundeportal for de som har et mobilabbonnement hos Phonero. Her kan kunden administrere sitt eget abonnement.
Din Bedrift	Phoneros kundeportal for bedriftskunder (f. eks. selskap som har kjøpt mobilabbonnement for sine ansatte), hvor bedriften kan administrere sine ansattes mobilabbonnement.
Produkt(er)	Produktene Phonero selger til sine kunder og tilbyr gjennom

	kampanjer, som for eksempel mobildata.
Bruker	Brukere av webapplikasjonen; utvalgte medlemmer av Phoneros markedsføringsavdeling og IT avdeling.
Kunder	Brukere av 'Ditt Phonero' som har mobilabonnement hos Phonero, og som kan få kampanjer og tilbud av Phonero.
Modal	Et lite vindu med innhold på en nettside eller programvare som er underlagt et annet større vindu, og kan åpnes uten å navigere vekk fra dette større vinduet. Modaler inneholder ofte funksjonalitet relatert til funksjonaliteten på det større vinduet.

Tabell 1: Definisjoner

1.3 - Gruppen

Gruppen består av seks medlemmer; Laurits Lund, Linh Phuong Tra, Markus Løvås Fjelltn, Zeyd Bouzidi, Ottar Worren, og Markus Skau. Medlemmene har alle jobbet med hverandre i en rekke andre prosjekter relatert til andre emner knyttet til bachelorprogrammet IT og Informasjonssystemer ved Universitet i Agder, som samtlige medlemmer er heltids studenter ved.

Gruppen har valgt å dele seg inn i tre arbeidskategorier for dette prosjektet. Disse kategoriene er 'Fullstack', 'Frontend', og 'Backend'. Medlemmene som tilhører en kategori vil ha hovedansvar for oppgaver knyttet til den kategorien. Uavhengig om medlemmer har hovedansvar for en oppgave betyr ikke dette at medlemmer som tilhører en annen kategori, ikke kan jobbe på den samme oppgaven.

Ansvars Kategori	Oppgaver
Fullstack	Administrative oppgaver, Rapportskrivning, Kvalitetskontroll, Utvikling på begge sider av stacken
Frontend	Design, Sideoppsett, Navigasjon
Backend	Database, Databasespørringer, Databehandling, Alternativ Datalagring, Bakgrunnstjenester og sikkerhet.

Tabell 2: Ansvarsfordeling

1.4 - Mål og ambisjoner

Gruppens medlemmer ønsker å tilegne seg praktiske erfaringer og kunnskap om både programvareutvikling og prosjektgjennomføring i en profesjonell kontekst i løpet av prosjektet. Dette innebærer for eksempel å bli kjent med nye teknologier og rammeverk, tilegne seg mer kunnskap og erfaring om prosjektstyring, og opprettholde god kommunikasjon med bedriften.

Gruppens ambisjon er å utvikle en webapplikasjon med god kvalitet, som bedriften er fornøyd med, og som vil bli brukt og videreutviklet av bedriften etter levering. Gruppen er motivert for å avslutte bachelorutdanningen med høy måloppnåelse, både faglig og teknisk.

2.0 - Prosjektbeskrivelse

Dette kapittelets formål er å med nøyaktighet kartlegge og beskrive systemet gruppen skal utvikle i løpet av prosjektet, og hvilke funksjoner det skal ha i henhold til krav satt av bedriften.

2.1 - Prosjektdefinisjon

Prosjektets formål er å utvikle en webapplikasjon som skal effektivisere bedriftens nåværende prosess rundt lansering av kampanjer ved å automatisere deler av prosessen, slik at arbeid på tvers av avdelinger relatert til dette forenkles, og arbeidsmengden går ned.

I dette prosjektet har gruppen fått tydelig beskjed om hva som skal utvikles, men hvordan og hvilke deler som inngår i systemet måtte kartlegges. Dette ble gjort før selve programvareutviklingen startet. Gruppen fikk tildelt kontorplasser hos bedriften, noe som la til rette for effektiv kommunikasjon både internt i gruppen, men også med oppdragsgiver. Gruppen ble bevisstgjort av bedriften at dette systemet skulle taes i bruk og videreutvikles etter at bacheloroppgaven var gjennomført. Dette gjorde at gruppen tok valg som gjorde det enklere å overta prosjektet, og jobbe videre på det. Valg som ble gjort på bakgrunn av dette vil begrunnes under sine respektive punkter i rapporten.

2.2 - Systemkrav

Bedriften, i samarbeid med gruppen, definerte en liste for hva webapplikasjonen skal være og gjøre. I tillegg har gruppen diskutert med brukerne hva deres ønsker rundt webapplikasjonens funksjoner skal være. Listen for å beskrive systemet:

- Det skal være en webapplikasjon som skal brukes av markedsføringsavdelingen og IT/Billing avdelingen.
- Den skal kunne lage en ny kampanje og lagre informasjonen i en egen database.
 - En kampanje skal inneholde informasjon som tittel, beskrivelse, bilder, start og slutt dato, og priser.
- Grafiske elementer skal kunne knyttes til en kampanje eller flere kampanjer
 - Bilder/animasjoner med forskjellige filtyper.
- Informasjon om kampanjer skal kunne presenteres til bruker.
 - Forhåndsvisning av informasjon.
 - Statistikk om pågående og gjennomførte kampanjer.
- Det skal være mulig å redigere alle kampanjer.
- Det skal være mulig å ha flere aktive kampanjer samtidig i visning.
- Det skal være mulig å se en historikk over tidligere kampanjer.

- Verktøy og teknologier brukt i utvikling av webapplikasjonen skal passe inn i de teknologier og verktøy bedriften bruker fra før.

Fra de nevnte kravene lagde gruppen en liste av funksjoner / sider som webapplikasjonen kunne inneholde:

- Ha en hovedside (Dashboard).
 - Vise pågående kampanjer.
 - Vise statistikk for valgte kampanjer.
- Lage en ny kampanje.
 - Fylle ut kampanjeinformasjon
 - Legge til grafiske elementer til kampanjen.
 - Spesifisere hvilket kundesegment kampanjen skal gjelde for.
 - Vise en oppsummering av kampanjen.
 - Lagre uferdige kampanjer som utkast.
- Gjøre endringer på en valgt kampanje.
- Laste opp grafiske elementer.
 - Velge hvilken type det grafiske elementet tilhører (bilde, ikon, etc.).
 - Velge et antall 'tagger' som beskriver det grafiske elementet.
- Ha en side som viser alle grafiske elementer.
 - Sortere visning etter type og 'tagger'.
- Ha en side som viser alle utkast.
 - Velge et utkast for å fortsette å jobbe på det.
- Ha en side som viser alle utgåtte kampanjer.
- Ha en side som viser alle kommende kampanjer.
- Lage nye kundesegment.

3.0 - Prosjektstyring

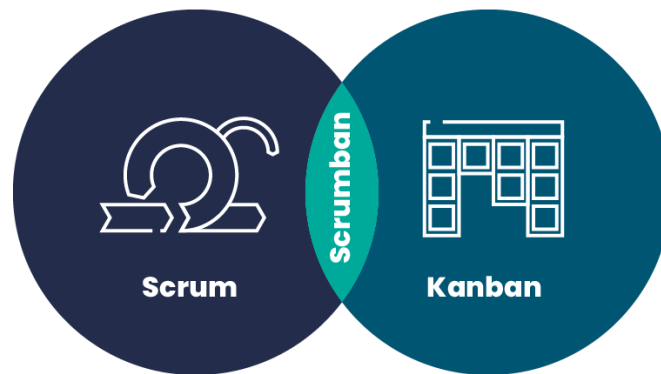
I dette kapittelet vil det presenteres hvordan gruppen har bestemt seg for å styre prosjektet, hvilke prosjektstyringsverktøy som er brukt og prioriteringer i forhold til systemets ulike deler (MoSCoW) som skal utvikles. Kapittelet inneholder også hvilke grep gruppen har gjort for å kvalitetssikre, samt risikofaktorer knyttet til prosjektet.

3.1 - Agil Prosjektmetode

Smidig prosjektledelse er en iterativ tilnærming til å administrere utviklingsprosjekter som fokuserer på kontinuerlige utgivelser og inkorporerer tilbakemeldinger fra kunder med hver iterasjon. En agil tilnærming til prosjektledelse kan øke utviklingshastigheten, utvide samarbeidet og fremme muligheten til å bedre svare på markedstrender (Drumond, u.å.).

Scrumban integrerer styrkene til Scrum og Kanban for å danne et allsidig prosjektledelse-rammeverk som vist i Figur 1. Scrumban bruker sprints og diverse møter som standups og sprint retrospektiv fra Scrum, samt den visuelle arbeidsflyten med kanban bord fra Kanban. Dette gjør Scrumban til en tilpasningsdyktig tilnærming til prosjekter av

varierende størrelse, fordi det gir et sett med smidige verktøy til å håndtere ulike prosjektbehov (Drumond, u.å.).



Figur 1: Diagram illustrerer Scrumban (Nearshore, 2022)

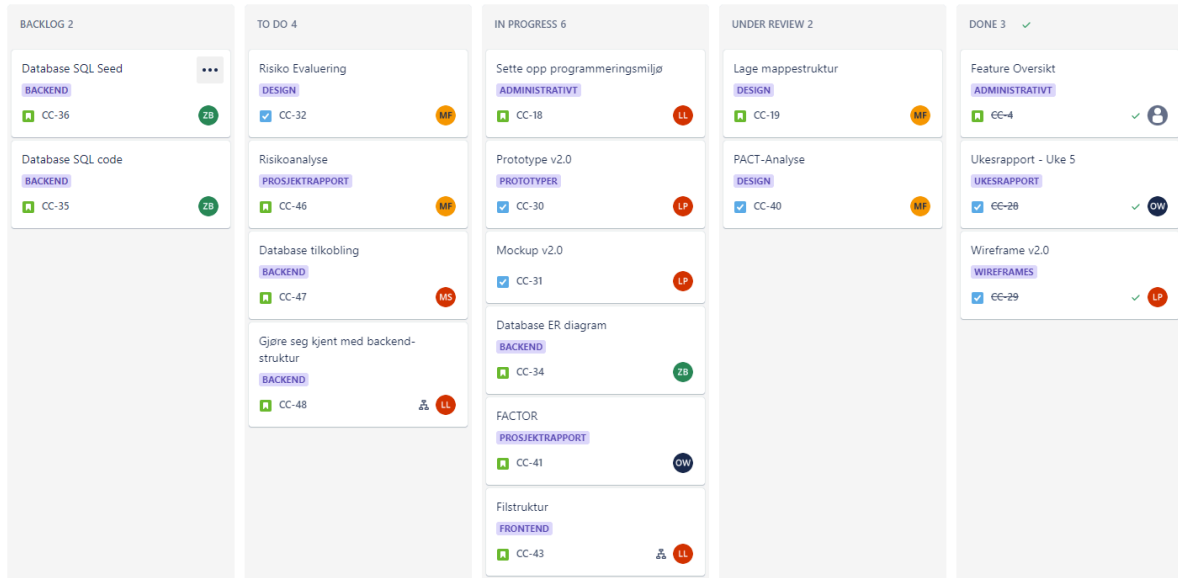
Fra starten av prosjektet ble det bestemt at gruppen skulle deles i tre. Delingen resulterte i to som jobbet fullstack, to som fokuserte på frontend og to som fokuserte på backend. Dette valget ble gjort på bakgrunn av interesser, for at medlemmene kunne spille på hverandres styrker, og for å enkelt kunne kartlegge gruppemedlemmenes arbeidsoppgaver. Det ble i tillegg avholdt faste møter hver mandag og fredag, for å allokere oppgaver og holde oversikt.

3.2 - Prosjektstyringsverktøy

Gruppen fikk tilgang til bedriftens løsning for prosjektstyring og versjonskontroll, Jira og Bitbucket. Jira for prosjektstyring, og Bitbucket for versjonskontroll. Siden gruppen brukte Scrumban som prosjektmetode, ble flere programvarer (Jira og Bitbucket) brukt for å holde oversikt over boards, sprints og datoer. Dette ble tilbudt av bedriften, og derfor brukt av gruppen gjennom hele prosjektet.

3.2.1 - Jira

Jira er en programvare utviklet av Atlassian som brukes til problem- og bugsporing, og agil prosjektstyring (Wikimedia Foundation, Inc, u.å.). Gruppen brukte Jira for å fordele arbeidsoppgaver på et Kanban bord med fem kolonner: Backlog, To Do, In Progress, Under Review, og Done. I tillegg ble alle oppgaver underlagt forskjellige 'Epics', som fungerte som kategorier for oppgavene. Oppgaver kan også bli tildelt et spesifikt medlem av prosjektet. I tilfeller hvor en oppgave ikke har blitt fordelt, ble gruppen enig om at dette var en oppgave noen påta seg. Dette gjaldt for de fleste administrative oppgaver, slik som skriving av denne rapporten. Et eksempel på hvordan gruppens kanban bord på Jira så ut gjennom prosjektets løp kan ses nedenfor i Figur 2.



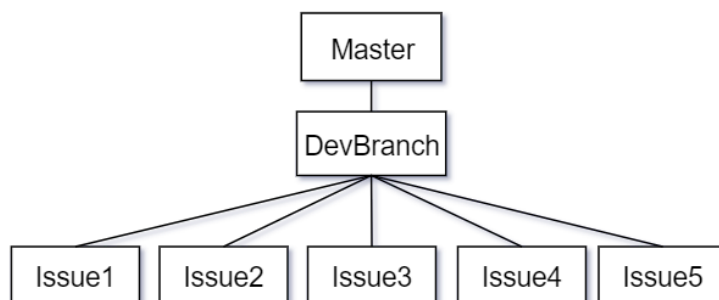
Figur 2: Kanban på Jira

Videre tillater Jira å strukturere arbeidsflyten i 'sprinter'. Hver sprint blir tildelt et kanban bord, i tillegg til start- og slutt dato. Dette gjør det mulig å sette kortsiktige mål for prosjektet, og å fordele arbeidsoppgaver på en effektiv måte. Jira ble valgt som prosjektledelses verktøy fordi bedriften bruker dette.

3.2.2 - Bitbucket

Bitbucket er en tjeneste for versjonskontroll basert på Git utviklet av Atlassian Bitbucket tilbyr skybasert lagring med git integrasjon, 'merge checks', 'code review', 'issue tracking', og 'branch management' (Wikimedia Foundation, Inc, u.å.). Bitbucket ble valgt fordi bedriften bruker dette.

Gruppen lagde versjonsstrukturen til prosjektet på Bitbucket ved å ha en hovedgren navngitt 'Master', og en gren for å 'merge' arbeid inn i navngitt 'DevBranch'. Når et medlem av gruppen ønsket å jobbe med en issue, ble en ny gren generert fra 'DevBranch' knyttet til en spesifikk arbeidsoppgave på Jira. Slik unngikk gruppen at medlemmer jobbet på en utdatert gren. En illustrering av grenstrukturen gruppen brukte på Bitbucket er illustrert nedenfor i Figur 3.



Figur 3: Grenstruktur på Bitbucket

Strukturen, sammen med bruk av de forskjellige verktøyene Bitbucket tilbyr, sørget for å minimere sannsynligheten for at problemer kunne oppstå på grunn av forskjeller mellom versjoner, konflikter i filer, eller feil i kode. Det sørget også for å minimere risikoen for tap av arbeid, ved for eksempel uheldig sletting av en gren.

Det var også muligheten til å lese ‘code reviews’. Dette er viktig for å unngå at det oppstår feil ved ‘merging’ av forgreninger. Gruppen sørget for at minst to andre medlemmer i gruppen måtte sjekke og rette opp feil og konflikter i ny kode før en gren kunne bli ført inn på DevBranch. Videre for å integrere endringer fra DevBranch til Master ble det krevd at fem av medlemmene sjekket og rettet opp feil og konflikter.

‘Git rebase’ er også et redskap som ble mye brukt for å holde koden oppdatert på Issue grener. Dette går ut på at en gren henter inn endringer som er gjort på DevBranch og automatisk føyer disse inn i grenen, før den skal ‘merge’ med DevBranch. I noen tilfeller må slike endringer manuelt føyes inn for å unngå konflikter.

3.3 - Prioritering

Når et prosjekt inneholder et bredt spekter av arbeidsoppgaver bundet til forskjellige funksjoner som skal utvikles, er det viktig å identifisere hvilke funksjoner som har høyest prioritet. Dette er for at gruppen kan prioritere arbeidsoppgaver. Det finnes flere måter å prioritere arbeidsoppgaver på, men gruppen har valgt å ta i bruk MoSCoW-metoden. Hovedsakelig fordi denne metoden skiller ut et prosjekts kjernefunksjonalitet, og derfor er ideell for å ta avgjørelser.

“MoSCoW metoden er en metodikk for å gi forutsigbarhet til prosjekter med inkrementelle leveranser” (Miranda, 2022, s. 19). Metodikken går ut på at man identifiserer hvilke funksjoner det som skal utvikles kan ha, for så å dele disse inn i fire kategorier: Must have, Should have, Could have, og Won't have (Miranda, 2022, s. 20).

Ved å ta i bruk MoSCoW kan man kartlegge både hvilke funksjoner prosjektet som utvikles, og hvilke av disse funksjonene som har høyest prioritering. Dermed kan gruppen lettere prioritere hvilke oppgaver som skal utføres først. Gruppen brukte derfor MoSCoW metoden for å skaffe en oversikt over hvilke systemkrav som var viktigst, slik at gruppen enkelt kunne prioritere arbeidsoppgaver i utviklingsprosessen.

Must Have
Lag ny Kampanje (Detaljer, Produkt, Kundesegment, etc.)
Endre kampanje
Media Opplastning. (bilder og animasjoner)
Dashboard (Hjemmeside - oversikt over pågående kampanjer)

Historikk (Oversikt over utgåtte kampanjer)
Kommende kampanjer (Oversikt)
Utkast (Oversikt over kampanjer som fremdeles jobbes på)
Koble opp mot bedriftens API for henting av relevant data om produkter etc.
Should Have
Split kampanje
Opprette nytt Kundesegment
Dashboard (med statistikk)
Mulighet for å legge til URL til produkt
Historikk (mulighet for å legge lage ny kampanje basert på en gammel en)
Legge til ny kampanje (avansert - med flere muligheter)
Could Have
Forhåndsvisning av kampanje i Ditt Phonero
Vise antall kunder i et kundesegment
Legge til mulighet for valg av template for kampanjer

Tabell 3: MoSCoW av funksjonene i webapplikasjonen.

Gruppen formidlet også sine vurderinger om MoSCoW prioriteringene, slik at bedriften også kunne ta del i prioriteringen. Slik sikret gruppen at prioriteringene fulgte bedriftens ønsker og forventninger. Gruppen kom da frem til at hovedfunksjonalitetene til webapplikasjonen var de viktigste å prioritere, også prioriterte videre i den rekkefølgen.

3.4 - Kvalitet

Kvalitet er definert av Store Norske Leksikon som “[...] tings måte å være på, tingens beskaffenhet. [...] For en gjenstand eller tjeneste kan man enkelt si at kvalitet er evnen til å tilfredsstille brukerens krav og forventninger.” (Gundersen & Halbo, 2018).

I dette prosjektet ble flere ting lagt vekt på, for å oppnå kvalitet. Det ble lagt vekt på kvaliteten i programvaren som ble utviklet, men også på prosjektgjennomføringen.

3.4.1 Programvarekvalitet

Programvarekvalitet defineres som “graden av samsvar med spesifikke funksjonskrav, spesifiserte programvarekvalitets standarder og god programvareingeniør praksis.” (Galin, 2018, s.3). Webapplikasjonens funksjonskrav ble beskrevet i kapittel 2.3. Galin presiserer hva

som menes med 'programvarekvalitets standarder og programvareingeniør praksis' som "relevant regulering og profesjonelle konvensjoner", og legger til at kvalitet også oppnås ved å oppfylle interessenters behov og ønsker (Galin, 2018, s.3).

For å sikre at den programvaren gruppen utvikler har god kvalitet, må man derfor kvalitetssikre. "Programvare kvalitetssikring er et sett med aktiviteter som definerer og vurderer programvare prosessers tilstrekkelighet for å beviselig danne tillit til at programvare prosessene er passende til å produsere programvare produkter, av passende kvalitet til deres mente bruk, eller for deres mente operasjonelle tjenester, og at de opprettholder krav til tidsplan og budsjettering." (Galin, 2018, s.5). Her menes 'programvare prosess' som de aktiviteter som utgjør spesifisering, design, implementering, og testing av programvare systemer.

Avsluttende kan det nevnes at den grad av kvalitet gruppen sikter mot, er den grad av kvalitet bedriften har formidlet at de ønsker for kampanjemotoren og prosjektet generelt.

3.4.2 - Kvalitetsfaktorer

Siden forberedelsene for prosjektet ble påbegynt høsten 2023, har gruppen hatt høyt fokus på at arbeid i prosjektets løp, samt prosjektets resultat skulle være av en kvalitet som både gruppen og bedriften ville være fornøyd med.

Det å sørge for at en programvare blir utviklet med god kvalitet krever at utviklerne kontrollerer at programvaren oppfyller en viss standard innen en rekke faktorer. Følgende faktorer kommer fra ISO/IEC 25010 standarden (ISO/IEC, 2023): Funksjonell egnethet, Effektivitet, Kompatibilitet, Brukervennlighet, Pålitelighet, Systemsikkerhet, Vedlikeholdbarhet, Fleksibilitet, og Brukersikkerhet. Kriteriene er forklart i Tabell 4, nedenfor. Denne kvalitetsstandarden er en globalt brukt standard for programvarekvalitet.

Kriterie	Beskrivelse
Funksjonell egnethet	At programvarens funksjonalitet egner seg til å gjennomføre alle sine tiltenkte oppgaver med presisjon og korrekthet.
Effektivitet	At programvaren ikke bruker for mye tid, energi og ressurser til å gjennomføre sine oppgaver, og klarer å håndtere spesifisert kapasitet.
Kompatibilitet	At programvaren kan sameksistere og samarbeide med andre systemer uten å forårsake problemer for andre systemer i sitt miljø.
Brukervennlighet	At programvaren har et grensesnitt som er gjenkjennbar, lettlært, og tilgjengelig for brukere på en engasjerende måte, er beskyttet mot brukerfeil, og er inkluderende for brukere av forskjellige bakgrunner.
Pålitelighet	At programvaren kan kjøre feilfritt, har høy toleranse for brukerfeil, og er sterkt rustet mot tap av data.

Systemsikkerhet	At programvaren har en høy grad av konfidensialitet, systemintegritet, at bruker- ansvarlighet og autentisitet er sikret, og er motstandsdyktig mot angrep.
Vedlikeholdbarhet	At programvaren har en høy grad av modularitet, modifiserbarhet, testbarhet, og er lett å analysere.
Fleksibilitet	At programvaren er lett å tilpasse, skalere, installere, og evne til å bytte ut og bli byttet ut.
Brukersikkerhet	At programvaren har systemer som sikrer brukerens helse ved bruk, og har varsler om fare relatert med dens bruk

Tabell 4: ISO/IEC 25010:2023 Kvalitetsfaktorer

3.4.3 - Kriterienes viktighetsgrad for Prosjektet

Prosjektet i sin helhet må ha en høy standard innen faktorene Effektivitet, Kompatibilitet, Pålitelighet, Systemsikkerhet, Vedlikeholdbarhet, og Fleksibilitet. Derimot retter faktoren Brukervennlighet seg mot brukergrensesnittet, altså ‘frontend’, mens Funksjonell egnethet retter seg mest mot det som foregår ‘under panseret’ i kampanjemotoren, altså ‘backend’.

De faktorene som er spesielt viktige for dette er Funksjonell egnethet, Pålitelighet, Vedlikeholdbarhet, og Fleksibilitet. Dette kommer av at bedriftens satte krav for kampanjemotorens funksjonalitet må oppfylles med en kvalitet som bedriften er fornøyd med, og at det ikke oppstår feil under bruk. Bedriften skal også ta over prosjektet for videreutvikling når gruppens arbeid er gjennomført, derfor må kampanjemotoren ha en høy kvalitet når det kommer til muligheten for videreutvikling og tilpassing.

Funksjonell egnethet: Dette er det viktigste kriteriet i dette prosjektet, da prosjektets hovedformål er å introdusere et system med funksjonalitet som effektiviserer bedriftens arbeidsprosesser rundt lagring og lansering av kampanjer. Derfor må kampanjemotorens funksjonalitet være velegnet for dette.

Effektivitet: Dette kriteriet er av middels viktighet. Dette kommer av at kampanjemotoren kun skal brukes av noen få utvalgte brukere. Dette betyr at systemets ressursforbruk vil forbli lavt, og det vil dermed også være enkelt å håndtere kapasitetsbegrensninger.

Kompatibilitet: Dette er et kriterium av lavere viktighet, men som gruppen likevel har lagt noe vekt på. Dette er fordi kampanjemotoren har et fremtidspotensial til å bli et system som samarbeider med en rekke andre systemer. Gruppen ønsker derfor å utvikle kampanjemotoren med en kvalitet innenfor dette kriteriet som gjør at bedriften enkelt kan utvikle systemer som samarbeider med kampanjemotoren.

Brukervennlighet: Dette er et nokså viktig kriterie for dette prosjektet, da kampanjemotorens brukere må kunne forstå alle dens funksjoner uten å trenge mye opplæring. Siden prosjektets fokus er å effektivisere en arbeidsprosess, må bruk av

kampanjemotoren forenkle arbeidsprosessen. Høy kvalitet innenfor dette kriteriet vil til dels oppnå dette.

Pålitelighet: Dette er et kriterium av middels viktighet for dette prosjektet. Dette er fordi kampanjemotoren skal håndtere relativt små mengder data, som gjør at risikoen for tap av data er lav. Videre er kampanjemotorens funksjonalitet fokusert rundt relativt få funksjoner, noe som gjør at sikring mot feil ikke blir særlig komplisert.

Systemsikkerhet: Dette kriteriet er av middels viktighet for prosjektet. Dette er fordi kampanjemotoren ikke skal håndtere noe sensitiv data, og skal kun være for intern bruk av bedriften. Ved å ta i bruk bedriftens allerede eksisterende sikkerhetsløsninger vil prosjektet derfor opprettholde en tilfredsstillende kvalitet innenfor dette kriteriet.

Vedlikeholdbarhet: Dette kriteriet er meget viktig da bedriften skal ta i bruk prosjektets resultat. Det må derfor være enkelt for bedriften å analysere, teste, og modifisere alle aspekter av kampanjemotoren.

Fleksibilitet: Dette kriteriet er også meget viktig for dette prosjektet siden bedriften skal ta over resultatet fra gruppens arbeid og videreutvikle det. Dermed er det viktig at den gruppen utvikler er enkelt å forstå og bygge videre på av bedriftens ansatte.

Brukersikkerhet: Dette kriteriet er ikke relevant for dette prosjektet. Dette er fordi systemet er en programvare, og alle risikoer for brukeren er relatert til maskinvaren systemet kjører på, noe gruppen ikke kan gjøre noe med. Disse faktorene er alle til varierende grad viktig for dette prosjektet, og nøyaktig hvordan gruppen kvalitetssikrer i henhold til disse faktorene vil bli diskutert nærmere i relevante kapitler i denne rapporten. De forskjellige faktorene er også viktige i forskjellig grad når det kommer til ulike deler av prosjektet.

3.4.4 - Kvalitet og Kvalitetssikring i Prosjektgjennomføring

Gjennom planlegging før prosjektstart og kartlegging av systemet i samarbeid med bedriften, la gruppen grunnlaget for å sikre kvaliteten av prosjektgjennomføringen. Bedriften stilte seg positiv til gruppens plan, når det kom til prioriteringer og kvalitetsfaktorer.

Gjennom sprint planning og sprint retrospektiv kunne gruppen diskutere og vurdere om kvaliteten til prosjektet ble opprettholdt, og gjøre grep om nødvendig. Ansatte i bedriften ble også invitert med på flere av møtene. Dette ble gjort for å gi en oppdatering på status i prosjektet, samt for å gi bedriften muligheten til å gi tilbakemeldinger.

Inndelingen av gruppens medlemmer i frontend, backend og fullstack, ble gjort for å sikre at alle medlemmene var fornøyd med oppgavene som ble utdelt. Dette for å legge et grunnlag for god arbeidsmoral og arbeidsflyt gjennom hele prosjektet.

3.5 - Risikovurdering

Siden dette er et utviklingsprosjekt, vil det ikke her bli gjort risikovurdering av generell drift. Derimot ble en risikovurdering av prosjektets målsetting gjennomført for å kartlegge hva

konsekvensene av å ikke gjennomføre milepæler i forventet tidsrom vil være. I tillegg vil en risikovurdering av uforutsette hendelser bli gjort.

Karlsen beskriver en fem-trinns prosess for å kartlegge, analysere, og håndtere usikkerheter (Karlsen, 2021, s. 465-482). Disse trinnene er oppstart, identifikasjon, analyse, respons, og oppfølging.

Oppstart: I dette trinnet skal man klargjøre for de resterende trinnene ved å avgjøre hvordan risikovurderingen skal gjøres, og hva dens formål er. Gruppen avgjorde at alle medlemmer skulle delta i denne prosessen. Dette blir gjort for å fremme diskusjon om de forskjellige risikoene, slik at alle gruppens perspektiver om disse kan bli vurdert.

Identifikasjon: Når trinn to begynner, skal risikoer identifiseres. Gruppen avgjorde at dette skal foregå ved at fremgangsplanen skal bli vurdert i samhold med funksjons prioriteringene gruppen har gjort tidligere. Slik kan gruppen kartlegge hvilke funksjoner og arbeidsoppgaver som er viktigst, og dermed vurdere risiko for avvik fra planen. Etter dette er gjort vil gruppen diskutere forskjellige andre risikoer for å kartlegge disse.

Analyse: Når alle risikoer har blitt kartlagt skal de bli analysert. Dette blir gjort her i trinn tre, og involverer bruk av en kritikalitetsmatrise vist nedenfor i Tabell 5. Dette fungerer ved at sannsynligheten for at en gitt risiko skjer, og hvor alvorlig konsekvensen av dette er, blir vurdert ved å gi hver av de poeng fra 1 til 5. Når risikoens sannsynlighet og konsekvens begge har fått poeng, kan man se i matrisen den totale poengsummen, og dermed avgjøre hver risikos alvorlighetsgrad. Etter hvert som hver risiko blir vurdert i risikomatrisen kan de så bli skrevet inn i en tabell vist i [appendiks A1](#).

Sannsynlighet	Konsekvens				
	1	2	3	4	5
5	5	10	15	20	25
4	4	8	12	16	20
3	3	6	9	12	15
2	2	4	6	8	10
1	1	2	3	4	5

Tabell 5: Kritikalitetsmatrise

Respons: Når samtlige risikoer har blitt analysert kan gruppen fokusere på tabellen som inneholder alle risikoene. Denne tabellen inneholder også kolonner for 'Respons' og 'Handling' som nå skal fylles inn. Respons vil beskrive hvilken type respons gruppen vil ha om risikoen tilfaller. Karlsen beskriver fire typer responser (Karlsen, 2021, s. 478), unngåelse, aksept, overføring, og reduksjon.

Unngåelse handler om å velge alternative løsninger og teknologier slik at risikoen ikke lengre er en faktor. Om en risikoen er av en slik natur at ingen som deltar i prosjektet kan gjøre noe med den, må det bare aksepteres. I slike tilfeller handler det om å ha god kontroll på hva konsekvensene av et slikt tilfelle innebærer, slik at en god handlingsplan kan iverksettes. Overføring innebærer at hele eller deler av risikoen blir overført til en annen aktør, slik at de kan redusere risikoen. Til sist betyr reduksjon å aktivt gjøre tiltak for å redusere risikoen, både når det gjelder sannsynlighet og konsekvens.

Når en avgjørelse om hvilken type respons gruppen vil ta til en gitt risiko, skrives til sist en handlingsplan i den siste kolonnen 'handlin'.

Oppfølging: Når risikoer har blitt analysert og en handlingsplan er lagt, må gruppen følge opp risikoene med jevne mellomrom slik at de kan holdes oppdatert til prosjektets tilstand. Dette kan innebære alt fra å revurdere tidsestimater og alle identifiserte risikoer, identifisere nye risikoer, og iverksette handlingsplaner om nødvendig.

4.0 - Analyse

Dette kapittelets formål er å formidle hvordan gruppen arbeidet for å legge grunnlaget for systemet som skulle utvikles. Det vil si, webapplikasjonens utseende, struktur, og arkitektur. Kapittelet inneholder datainnsamlingen som er gjort, brukerhistorier, og FACTOR analyse. Avsluttende vil det bli begrunnet hvordan analysen gjort i prosjektet er av kvalitet.

4.1 - Datainnsamling

“Problemomene er den delen av en kontekst som blir administrert, monitorert, eller kontrollert av et system” (Mathiassen, et. al. 2018). Et problemomene er altså de prosessene og problemene som webapplikasjonen skal forenkle og løse. Det er derfor veldig viktig for gruppen å få god forståelse hva problemomene er, slik utviklingen av webapplikasjonen faktisk

For at gruppen skulle få full forståelse over problemomene, altså de prosessene som webapplikasjonen skal kontrollere, måtte informasjon om dette samles inn. Dette ble hovedsakelig gjort gjennom diskusjoner med ansatte i bedriftens IT avdeling og markedsføringsavdeling. På disse møtene ble det beskrevet at bedriften til nå har måttet foreta hele prosessen rundt kampanjer manuelt. Altså når markedsføringsavdelingen hadde gjort klar en kampanje, måtte de gi all informasjon om den til IT avdeling, som så manuelt måtte føre informasjonen inn i databasen og sette opp Ditt Phonero slik at kampanjen vises til kundene.

Webapplikasjonens rolle er å forenkle og effektivisere denne prosessen, slik at markedsføringsavdelingen kan jobbe med kampanjer i webapplikasjonen og legge dem inn i databasen automatisk. Dette vil gjøre jobben til både IT avdelingen og markedsføringsavdelingen enklere.

4.2 - Brukerhistorier

Brukerhistorier er et konsept som går ut på å skrive små historier rundt hvordan potensielle brukere vil ta i bruk et produkt. Bakgrunnen til at gruppen valgte å inkludere brukerhistorier i analysen er fordi det er et effektivt hjelpemiddel til å identifisere krav, samtidig som man styrker funksjonaliteten til produktet som utvikles. Dette er for å fremme kundenes interesser (Mikalsen, 2024). I dette tilfellet, er brukerhistoriene sentrert rundt brukerne av en webapplikasjon som skal fungere som en kampanje motor. Gruppen laget 5 scenarier som er koblet til forskjellige funksjoner.

Nedenfor i tabell 6 vises et eksempel på en brukerhistorie, resten av brukerhistoriene finnes i [appendiks A2](#). Alle brukerhistorie tabellene inneholder følgende; selve brukerhistorien, funksjonen som bruker historien gjelder, funksjonens formål, test-scenario som inneholder en gjennomgang av hva brukeren må gjennomføre, og til slutt en argumentasjon for hvorfor denne funksjonaliteten er viktig. Øverst på tabellen ser man hvilken prioritet funksjonaliteten har i MoSCoW analysen som er gjort.

4.2.1 - Brukerhistorie #1

#1	Prioritet: Must have
Brukerhistorie	For å kunne distribuere kampanje til kunder på Ditt Phonero, ønsker jeg som markedsførings manager å kunne lage en ny kampanje.
Funksjon	Opprette ny kampanje
Formål	Publiserer en ny kampanje på 'Ditt Phonero' webapplikasjonen
Test-scenario	Opprette en ny kampanje: 1. Gå inn "Lag ny kampanje" 2. Velg "Enkel kampanje" (Steg 1) 3. Fyll ut alle feltene til kampanje 4. Legg til grafikk 5. Se en oppsummering til kampanje 6. Klikk på "fullført"
Argumentasjon	Denne funksjonen er viktig fordi markedsføringsavdelingen gjennomfører en rekke kampanjer for distribusjon til kundene. Dette bidrar til å forbedre kundeopplevelsen knyttet til bedriftens produkter.

Tabell 6: Brukerhistorie 1

4.3 - FACTOR analyse

En FACTOR analyse er en systematisk tilnærming som er med på å gi en fullstendig forståelse av et prosjekts krav og kontekst. Denne FACTOR analysen er basert på webapplikasjonen som gruppen utvikler i samarbeid med bedriften. Ved å benytte en av de seks aspektene; Funksjonalitet, Applikasjons Domene, Forhold, Objekter og ansvar. Ved å

separerer de forskjellige aspektene dannes et helhetlig bilde av hva som kreves for å utvikle og implementere den beskrevne webapplikasjonen (Mathiassen, et al., 2018, s.40-41). Nedenfor blir FACTOR analysen presentert i Tabell 7.

Funksjonalitet	Forbedre prosessen som foretas når markedsføringsavdelingen lager og distribuerer kampanjer. I tillegg skal brukerne kunne se statistikk basert på kampanjene.
Applikasjons Domene	Applikasjonen skal gi brukerne i markedsføringsavdelingen muligheten til å konstruere kampanjer og distribuere dem ut til kunder på ulike plattformer.
Forhold	Funksjonelt og oversiktlig å operere. Systemet er designet for å brukes på pc, men skal kunne være funksjonelt på andre plattformer.
Teknologi	Systemet vil kjøre ved bruk av internett, server, datamaskin, skjerm. Systemet vil utvikles ved bruk av MSSQL, VSCode, .NET 8.0, C#, REACT TS, Kubernetes, Jira, XUnit, MOCHA, m.m.
Objekter	Ansatte og kampanjer.
Ansvar	Laste opp bilder tilhørende kampanjer, sette sammen kampanjer, generere statistikk ut ifra kampanjenes prestasjon, konstruere- og distribuere kampanjer.

Tabell 7: FACTOR analysen

Grunnen til at gruppen valgte å utføre en FACTOR analyse fremfor en PACT analyse er hovedsakelig på bakgrunn av at en FACTOR analyse er egnet for prosjekter hvor det er mest fokus på det tekniske og funksjonalitet. Denne typen analyse er mer detaljert og ikke like dyptgående i brukerens interaksjon. Analysen tar mennesker i betraktning, men legger mer vekt på det tekniske. FACTOR analysen passer derfor bedre i forhold til systemkravene som gruppen har.

4.4 - Kvalitetssikring

Alle data beskrevet i dette kapitlet ble enten innsamlet eller konstruert basert på innsamlet data. Kvalitetssikring av denne dataen har blitt gjort gjennom møter og diskusjon med bedriften. All innsamlet data, beskrevet i kapittel 2.0, ble innhentet gjennom flere møter med representanter fra bedriftens IT avdeling og markedsføringsavdeling. Her identifiserte gruppen at det viktigste i dette prosjektet er kampanjemotorens funksjonalitet. Dette er viktig da gruppens valg av analysemetodikk er rettet mot nettopp dette.

Brakerhistorier ble konstruert som følge av gruppens funksjonalitets prioritering i kapittel 3.3, og er basert på de viktigste funksjonene. Ved ikke bare å beskrive funksjonalitet, men også funksjonens formål, samt argumentere for hvorfor funksjonen er viktig, sikrer gruppen at data resultert fra brukerhistoriene er av god kvalitet i forhold til bedriftens forventninger av hver gitte funksjon.

Alt dette, samt god kommunikasjon med bedriften, gjør at gruppen kan, ved bruk av data identifisert i analysene, sikre at resultater av arbeid holder ønsket kvalitetsnivå innen kvalitets faktorene, beskrevet i kapittel 3.4.2, 'Funksjonell egnethet', 'Brukervennlighet', 'Vedlikeholdbarhet', Effektivitet og 'Fleksibilitet'.

5.0 - Design

I dette kapittelet vil designprosessen til gruppen presenteres. Dette omfatter valg av verktøy, utforming av wireframes og mockups, designprinsipper, prototype og kvalitetssikring.

Gruppen har lagt spesielt vekt på at designet skal være intuitivt og enkelt å bruke. Samtidig så vil gruppen at designet skal føre til at applikasjonen blir brukt. Applikasjonen skal ikke være "kjedelig" å bruke, den skal være inspirerende og brukervennlig.

5.1 - Designverktøy

5.1.1 - UXPin

UXPin er et designverktøy som brukes for å utvikle wireframe, mockup eller prototype. Dette verktøyet støtter samarbeid i sanntid, noe som gjør det mulig for gruppemedlemmer å jobbe samtidig med det samme (UXPin, u.å.). Tidlig i designfasen valgte gruppen å bruke UXPin til å lage mockups og prototyper. Grunnen til hvorfor gruppen valgte å ta i bruk UXPin er fordi gruppen tidligere har benyttet seg av UXPin. I tillegg tilbyr UXPin en rekke funksjoner, inkludert innebygde komponenter som input felt, knapper. Dette gjør det mulig å effektivt produsere hifi mockups og interaktive prototyper. Utover i prosjektet begynte gruppen derimot å bruke Figma til å utvikle sluttproduktet.

5.1.2 - Figma

Figma er et designverktøy som ble brukt for å utforme wireframes, navigasjonskart og lowfi prototyping (Figma, u.å.). Likt som UXPin, er Figma også et effektivt designverktøy som brukes for å designe, lage prototyper og utvikle brukergrensesnitt for et produkt eller et system. Det gir også mulighet til for at flere medlemmer i et team kan jobbe sammen på samme prosjekt i sanntid (Figma, u.å.). Grunnen til at gruppen valgte Figma er fordi bedriften også bruker Figma. Dette betyr at utviklerteam hos bedriften kan videreutvikle brukergrensesnittet for webapplikasjonen etter behov etter at den er overlevert til bedriften.

En ulempe med Figma er at det tar litt tid å lage en interaktiv prototype, fordi Figma ikke har innebygd komponenter som UXPin. Dersom gruppen ville ha et inputfelt eller en

nedtrekksmeny til prototypen, så måtte slike komponenter lages fra bunnen av. Dette ga samtidig gruppen også muligheten til å lære mer om Figma.

5.1.3 - Lucidcharts

Lucidcharts er et skybasert verktøy for å lage og dele diagrammer for systemer og prosesser. Det er et spesielt bra verktøy for å planlegge og designe arkitektur, for eksempel database-relasjoner og tabellinnhold (Lucid Software Inc., u.å.).

Lucidcharts ble valgt fordi gruppen har brukt det tidligere, og anser det som et godt verktøy for å visualisere databasen. Det gir også muligheten for å dele database 'chartet' med andre, slik at alle gruppemedlemmene får en oversikt over relasjonene i datalagringen. Dette verktøyet ble brukt til å lage et diagram som representerer databasen. Databasesdesignet inneholder både tabeller, triggere og fargekoder som hjelper med å forstå strukturen og gjør det lettere å konvertere til et MSSQL query language.

5.1.4 - Draw.io

Draw.io er et annet skybasert verktøy for å lage og dele diagrammer (JGraph Ltd., u.å.). Det er enkelt å bruke, samtidig som at det gjør det enkelt å visualisere strukturer, samtidig som man kollaborerer med andre. Gruppen valgte dette verktøyet grunnet de nevnte egenskapene. I dette prosjektet har det blitt brukt for å lage diagrammer for diverse, både for intern bruk og for presentasjon til bedriften og denne rapporten. Noen av disse diagrammene viser elementer som mappestruktur i prosjektet, visualisering av Single Page Application struktur, etc.

5.2 - Wireframe

Wireframe er et skjelett for en nettside eller en webapplikasjon uten farger eller grafikk. Ved å lage en wireframe kan det hjelpe designere til å fange opp designidéer, opprette sidestrukturen og navigasjon før man videreutvikler et detaljert design (Miro, u.å.). Etter at alle gruppemedlemmene fikk oversikt over systemdefinisjon og brukerhistorier, ble det laget to forskjellige wireframe med enkle oppsett.. Dette ga både gruppen og bedriften mulighet til å diskutere og finne ut hvilke sider som burde være med i webapplikasjonen. Wireframes som ble laget, ligger i [appendiks A3](#).

5.3 - Mockup

Mock-up er et høykvalitets design og inkluderer farger, bilder, typografi og andre grafiske elementer, men som mangler interaktiv funksjonalitet. Utviklingen av mockups gir både designere og brukere muligheten til å visualisere hvordan et sluttprodukt vil se ut på en realistisk måte (Figma, u.å.). Etter å ha gjennomført to wireframes, utviklet gruppen to mockups med distinkte design, som kan ses i [appendiks A4](#). Dette ble gjort for å gi markedsføringsteamet alternativer, slik at de kunne velge deres foretrukne design, foreslå tilpasninger og bidra til utformingen av det endelige designet. Den ene mockup ble laget i UXPin, mens den andre ble utviklet ved bruk av Figma. Begge mockupene ble presentert i et møte med markedsføringsavdelingen, og de fant tiltalende aspekter ved begge. Dette førte til

at gruppen valgte å kombinere designidéer og elementer fra begge mockupene til det endelige designet av webapplikasjonen som vises i [appendiks A4.3](#).

5.4 - Brukertesting

Brukertester kan utføres på mange ulike måter, med en rekke metodikker og på ulike tidspunkt gjennom prosjektstadiet. Innen brukeropplevelses forskning, faller metodikken inn i to hovedkategorier, kvalitative og kvantitative. Kvalitative metoder skaper forståelse for hvorfor. Kvantitative metoder skaper forståelse for hvor i systemet og hvor ofte noe skjer. Ved å ha en grunnleggende forståelse rundt disse to konseptene, blir videre undersøkelser og forskning lettere å sette i perspektiv på hva som skal oppnås (Kanji, 2021, s.1). Gruppen har i prosjektets løp utnyttet tre faser innenfor brukertesting av designet.

Primærforskning/opplagelses metoder: Det tidligste stadiet i et prosjekts utvikling er der man fastslår om ideer er nyttige og levedyktige. Det er en periode som kan forme hele prosjektet og hjelpe etablere et fundament for videre utvikling (Kanji, 2021, s.2). Ved prosjektoppstart ble det derfor satt stor fokus på å knytte et samarbeid mellom gruppen, prosjekteieren og de endelige brukerne. Dette ble utført ved å bruke fokusgrupper, som er en effektiv måte å samle flere brukere samtidig, for å jobbe dynamisk fram til svar på vanskelige beslutninger og tilbakemeldinger. Nedsiden med denne metodikken er at deltakere kan bli påvirket av hverandres svar, men det var nødvendig at alle deltakerne var til stede for å ikke gå utenfor omfanget av prosjektet.

Formative metoder: Etter at primærforskning har gitt en generell retning for prosjektet og designet, er mer formative metoder brukt for å styre designet i riktig retning (Kanji, 2021, s.9). Designet av applikasjonen ble gjort i en iterativ prosess, der brukertesting ble utført for hver versjon. Formative metoder er ofte gjort iterativt sammen med evaluerende metoder for å sikre kvalitet, og var noe gruppen gjorde samtidig med brukerne under møtene. Konsept testing ble brukt i flere stadier av prosjektet, der tidlig fokus var på generelle “mockups” og senere ble mer rettet mot hele prototyper.

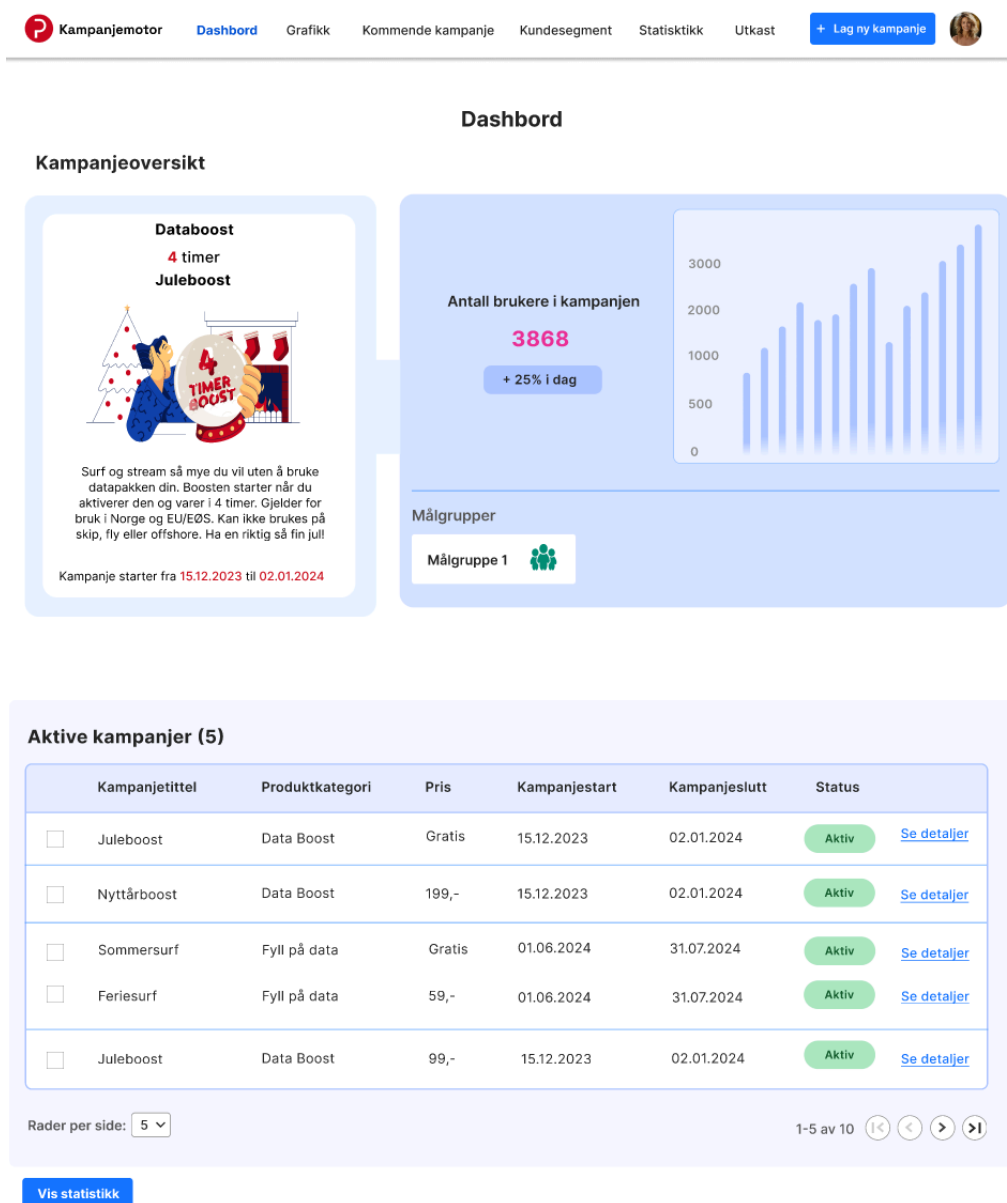
Evaluerende metoder: Som tidligere nevnt, var formative og evaluerende metoder brukt i en iterativ prosess gjennom designfasen (Kanji, 2021, s.14). Tidlig i iterasjonen var fokus satt på å gjøre valg og bygge på neste iterasjon, mens senere var det mest nyttig for å finne områder som hadde forbedringspotensial. Møtene med brukerne var semistrukturerte, og handlet mer om å ha en åpen samtale, med noen metoder for å holde noe struktur. Metodene brukt i denne fasen var blant annet A/B testing og demoer.

5.5 - Designprinsipper

For å møte markedsføringsavdelingens behov for et forståelig, rask og enkelt system for å opprette kampanjer, valgte gruppen å bruke Benyons designprinsipper som grunnsteiner for designet i prosjektet. Benyons designprinsipper gir et godt grunnlag for utviklingen av brukergrensesnitt som er brukbare, brukervennlige og effektive (Benyon, 2019, s.116). Gruppen valgte å fokusere på syv av Benyons 12 designprinsippene. Dette var fordi ikke alle

var like relevante systemet som skulle utvikles. Kampanjemotoren skal bare brukes av noen få personer som jobber hos bedriften, fra markedsføringsavdelingen og IT avdelingen. Brukerne vil være godt kjent med å bruke datamaskiner. Benyons designprinsipper er i stor grad rettet mot mer gjennomsnittlige brukere, med ulik teknologisk kunnskap. Siden bedriftens ansatte allerede har mer enn grunnleggende digitale ferdigheter, anså ikke gruppen det som nødvendig å ta i bruk alle 12 prinsipper. Nedenfor er en liste av disse syv prinsippene, supplert med eksempler:

- **‘Visibility’** - Dette prinsippet sikrer at alle elementer er synlige slik at brukere enkelt kan forstå hva systemet gjør og hvilke funksjoner som er tilgjengelig i systemet (Benyon, 2019, s.117). Med tanke på markedsføringsavdelingens ønske om at all informasjon skal være synlig på hjemmesiden, utviklet gruppen et dashbord som fungerer som en oversiktlig og informativ hjemmeside. Denne siden viser all informasjon om pågående kampanjer, samt tilhørende statistikk. Figur 4 nedenfor illustrerer oppsettet på dashbordet.



Figur 4: Dashbord side.

- ‘Consistency’**- Prinsippet handler om en konsekvent stil av alle elementer i et design. Dette inkluderer oppførsel på tvers av ulike sider og funksjoner (Benyon, 2019, s.117). Gruppen valgte å prioritere dette prinsippet fordi et konsistent system gjør det lettere for brukere å rask lære hvordan de skal samhandle og navigere gjennom systemet. Dette styrker brukbarhet og brukervennlighet til brukere (Wong, 2021). Figur 5 viser et eksempel på design av tabeller.

Kommende kampanjer (5)						
	Kampanjetittel	Produktkategori	Pris	Kampanjestart	Kampanjeslutt	Status
☐	Juleboost	Data Boost	Gratis	15.12.2023	02.01.2024	Upcoming Se detaljer
☐	Nyttårboost	Data Boost	199,-	15.12.2023	02.01.2024	Upcoming Se detaljer
☐	Sommersurf	Fyll på data	Gratis	01.06.2024	31.07.2024	Upcoming Se detaljer
☐	Feriesurf	Fyll på data	59,-	01.06.2024	31.07.2024	Upcoming Se detaljer
☐	Juleboost	Data Boost	99,-	15.12.2023	02.01.2024	Upcoming Se detaljer

Rader per side:

1-5 av 10 [<<](#) [<](#) [>](#) [>>](#)

Aktive- inaktive kampanjer (5)						
	Kampanjetittel	Produktkategori	Pris	Kampanjestart	Kampanjeslutt	Status
☐	Juleboost	Data Boost	Gratis	15.12.2023	02.01.2024	Aktiv Se detaljer
☐	Nyttårboost	Data Boost	199,-	15.12.2023	02.01.2024	Aktiv Se detaljer
☐	Sommersurf	Fyll på data	Gratis	01.06.2024	31.07.2024	Aktiv Se detaljer
☐	Feriesurf	Fyll på data	59,-	01.06.2024	31.07.2024	Aktiv Se detaljer
☐	Juleboost	Data Boost	99,-	15.12.2023	02.01.2024	Utgått Se detaljer

Rader per side:

1-5 av 10 [<<](#) [<](#) [>](#) [>>](#)

Figur 5: Kampanje-tabell.

- **‘Familiarity’**- Prinsippet refererer til bruk av språk, symboler eller designelementer som er fortrolig med brukere (Benyon, 2019. s.117). Dette kan hjelpe brukere til å raskt tilpasse seg et nytt system siden de ikke trenger å lære seg helt nye måter eller begrep i systemet for å gjennomføre oppgaver (Benyon, 2019. s.117). Ettersom at en kampanje består av mange forskjellige typer informasjon, som tittel, to beskrivelser, notis, bilder, etc., valgte gruppen å bruke terminologier foreslått av markedsføringsavdelingen. For eksempel er det to beskrivelser i en kampanje. Gruppen benyttet betegnelsen “Ingress” for den første beskrivelsen, og “Brødtekst” for den andre beskrivelsen. Dette gjør det enkelt for markedsføringsavdelingen å raskt forstå og fylle inn nødvendig informasjon når de skal opprette en ny kampanje, uten forvirring rundt terminologien. Figur 6 viser forslaget til terminologi for kampanje og prosessen for å opprette kampanjeinformasjon med disse terminologiene i systemet.

Kampanje Informasjon

Kampanjenavn *

Skriv inn et kampanjenavn. Dette er kun for internt bruk.

Kampanjetittel

Skriv inn en kampanjetittel.

Ingress

Skriv inn ingress

Brødtekst

Skriv inn en brødtekst

Målgruppe

Velg Målgruppe

Produktkategori

Velg Produktkategori

Produkt

Velg Produkt

Tilbudstekst

Skriv inn en tilbudstekst

Tilbudspris

kr 0,-

PrisInfo

Skriv inn prisInfo

Kampanjesstart

Velg dato

Kampanjesslutt

Velg dato

Bemerkning

Skriv inn en bemerkning.

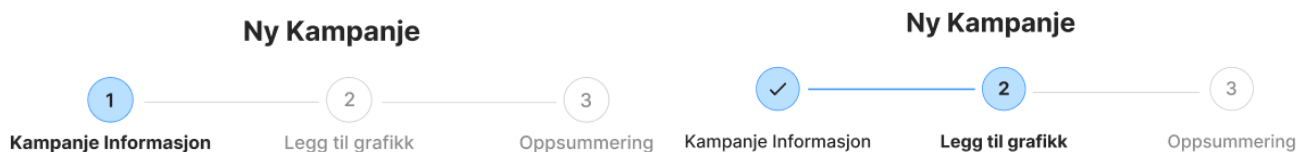
Figur 6: Kampanjeinformasjon

- **‘Affordance’** - Prinsippet omhandler at elementer i brukergrensesnittet er utformet på en måte som indikerer hva de gjør og hvordan de kan brukes (Benyon, 2019, s.117). For eksempel, hvis det er en knapp som har en “redigere” funksjon, så kan brukere trykke på knappen og få tilgang til å redigere en kampanje. Figur 7 nedenfor viser to typer av knapper. Den venstre knappen gir brukere muligheten til å avbryte en handling, og den høyre knappen brukes for å redigere en kampanje.



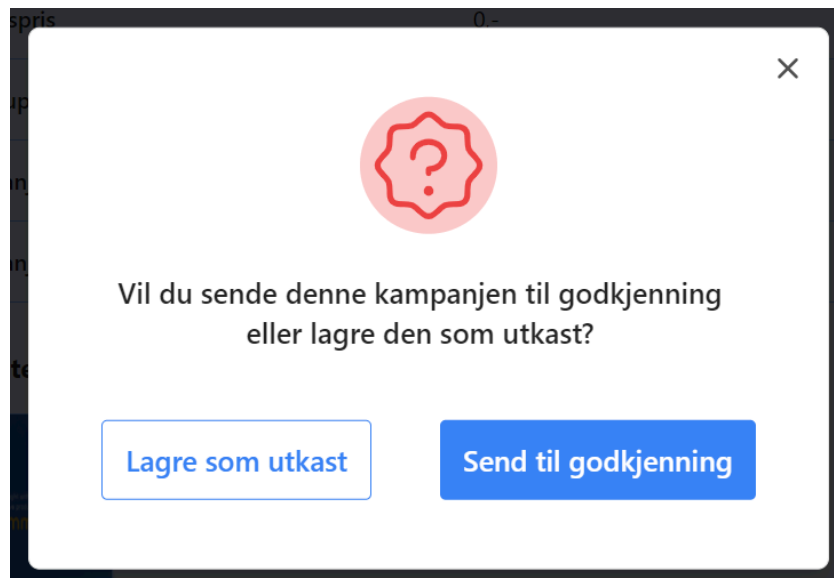
Figur 7: Knapper.

- **‘Navigation’**- Prinsippet refererer til å støtte brukere med å navigere gjennom et system. (Benyon, 2019, s.121). Gruppen valgte å bruke dette prinsippet i kampanjemotoren, fordi gruppen vil sikre at brukere vet hvor de er i systemet og hva de kan gjøre. Siden en kampanje omfatter både tekster og grafiske elementer, ble prosessen for å opprette en ny kampanje delt i tre steg, vist nedenfor i Figur 8. Dette ble gjort gjennom implementering av ‘stepper’ navigasjon, som lar brukere enkelt navigere gjennom de tre steg. For å sikre at brukere vet hvilket steg de er på i prosessen, så er det aktive steget merket med blå farge og ‘bold’ skriftformatering. De neste stegene som ikke er nådd ennå, vises i grått. Dersom brukere har fullført et steg, blir dette markert med et hake-symbol, ‘✓’, som indikerer at handlingen i det steget er fullført.



Figur 8: ‘Stepper’ navigasjon.

- **‘Feedback’** - Tilbakemeldinger er et prinsipp som gruppen anså som viktig for systemet. Dette dreier seg om at systemet gir en varsling eller tilbakemelding til brukeren, slik at brukeren forstår hvilken effekt deres handlinger har (Benyon, 2019, s.117). For å sikre at brukere sender en nylig laget kampanje til godkjenning, inkluderer systemet en modal som tilbyr to valg som bekrefter handlingen brukeren ønsker å utføre. Denne modalen kan ses i Figur 9. Dette gir brukeren mulighet til å velge om de ønsker å sende kampanjen videre til godkjenning, eller lagre den som utkast. Prinsippet er implementert flere steder i vårt system.



Figur 9: Bekreftelse for videresend til godkjenning

- **‘Constraints’**- Dette prinsippet handler om å gi en begrensning i systemet til brukere for å forhindre at de begår feil (Benyon, 2019, s.117). Gruppen valgte å implementere dette prinsippet i systemet ved å sørge for at brukere har lagt til all informasjon til en ny kampanje før det blir sendt videre til godkjenning. Dersom noen inputfelt er tomme, får ikke brukeren lov å sende kampanjen til godkjenning. Systemet gir også et alternativ til brukere å lagre kampanje i utkast hvis den ikke har blitt fullført. Dette valgalternativet kan ses i Figur 10.



Figur 10: Brukere kan fortsette å fullføre kampanjen eller lagre som utkast.

5.6 - Prototype

En prototype kan bli definert som et utkast, eller en tidlig modell som visualiserer både brukergrensesnittet og funksjonaliteten til et produkt eller system. I kontekst av en nettside

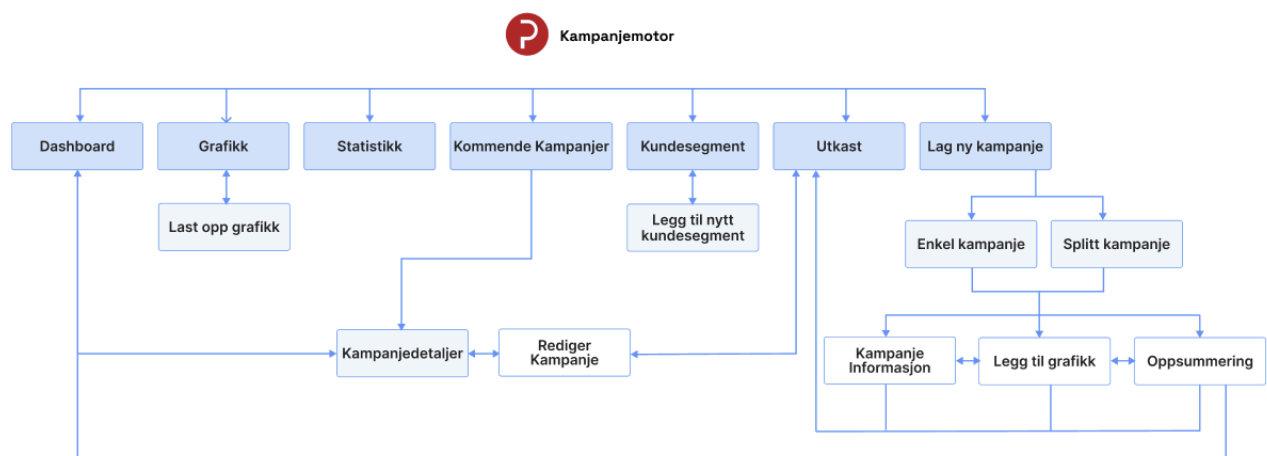
eller webapplikasjon, kan den bli brukt til å teste og evaluere designkonsepter og brukeropplevelser, slik at problemer kan identifiseres og løses tidlig før nettsiden eller applikasjonen produseres i stor skala (Ramírez, 2018).

Etter å ha utviklet to mockups med ulike design, lagde gruppen også to prototyper for å presentere til markedsføringsavdelingen i Sprint 1. Dette ble gjort slik at gruppen kunne samle tilbakemeldinger fra markedsføringsteamet om designet og oversikt over ønsket funksjonalitet i systemet. På grunnlag av tilbakemeldinger fra to intervjuer med både markedsførings- og IT-avdelingen, utviklet gruppen en tredje [prototype](#), versjon 2.0. Denne versjonen ga en detaljert forhåndsvisning av hvordan den endelige webapplikasjonen skulle se ut og fungere. Prototypen inkluderte alle funksjonaliteter som markedsføringsavdelingen ønsket. Det ble designet i henhold til åtte av Benyons 12 designprinsipper for å sikre at systemet er brukervennlig og for å skape en god brukeropplevelse.

5.7 - Navigasjonskart

Navigasjonskart omhandler en struktur på en nettside eller en webapplikasjon. Ved å utvikle et navigasjonskart kan designere få se hvordan forskjellige sider i en nettside eller en webapplikasjon blir organisert og koblet sammen. Ut fra det kan man forstå hvordan brukere kan navigere gjennom en nettside (Benyon, 2019, s.191). Gruppen lagde et navigasjonskart av webapplikasjonen for å sikre at navigasjonen gjennom applikasjonen har en gjennomgående og god flyt.

Figur 11 nedenfor illustrerer navigasjonskartet av webapplikasjonen.



Figur 11: Navigasjonskart av kampanjemotor-webapplikasjonen.

5.8 - Kvalitetssikring

Ved å forstå hovedaspektene og problemer med tilgjengelighet, prinsipper ved brukbarhet og generelle underliggende prinsipper for et bra UX design, kan man lettere kvalitetssikre design. Gjennom designprosessen har gruppen fulgt Benyons 12 designprinsipper, og sørget

for at applikasjonen er lett forståelig og brukervennlig. Brukertesting ble også brukt som en måte å kvalitetssikre designet på.

5.8.1 - Kvalitetssikring av brukertesting

Resultatene fra testing med fokusgruppene ga gruppen en god forståelse for hvordan designet skulle se ut. MoSCoW skjemaet for de viktigste funksjonalitetene ble oppdatert basert på innspill fra brukerne. Konsept-testing, A/B testing og demonstrasjoner av to mockups, ga gruppen mange tilbakemeldinger og fastsatte mål for videre utvikling.

Senere i prosjektets løp, hvor kampanjemotoren var delvis utviklet, ble det igjen gjort brukertester med representanter fra bedriftens markedsføringsavdeling og IT avdeling. Her ble det foretatt en gjennomgang av både funksjonalitet og brukergrensesnitt. Deltakeren ble tildelt en oppgave som skulle løses uten forklaring om hvordan, mens en observatør fulgte med og noterte alt som ble sagt og gjort av deltakeren. Til sist skulle deltakeren gjennomføre et spørreark, hvor de krysset av på skalerte spørsmål. Her var det en negativ og positiv versjon av hvert spørsmål. Dette var for å sikre at deltakeren evaluerte systemets brukbarhet fra både negativt og positivt perspektiv. En mal for den gjennomførte brukertesten kan ses i [appendiks A6](#). Resultater kan ses i [appendiks A7](#).

Brukertesting har altså ikke bare vært intervjuer og møter, men en pågående prosess gjennom hele bacheloroppgaven, der gruppen stadig har bedt om innspill og innsikt inn i brukernes ønsker og tanker. Ved å ha en flytende prosess, sammenslått med brukertesting, har utviklingen av applikasjonen vært betraktelig lettere å utføre.

6.0 - Arkitektur

I dette kapittelet vil webapplikasjonens arkitektur bli gjennomgått. Første delen av kapittelet vil omhandle den generelle strukturen i webapplikasjonen. Dette gjelder Single Page Application (SPA), og mappestrukturen til prosjektet. Videre vil databasen til webapplikasjonen presenteres. Til slutt vil bakgrunnsprosessene som skal kjøre bli forklart, samt hvordan gruppen har kvalitetssikret webapplikasjonen struktur og arkitektur.

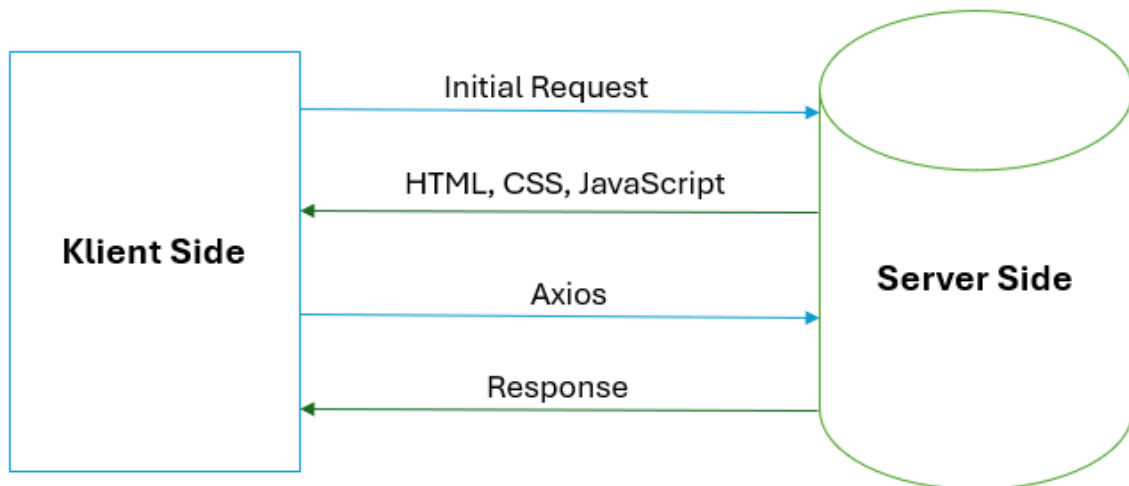
6.1 - Struktur

Dette kapittelet vil ta for seg strukturen i webapplikasjonen. Kapittelet vil først ta for seg funksjonaliteten til en (SPA) Single Page Applikasjon, og hvordan gruppens webapplikasjon benytter seg av denne teknologien. Deretter vil mappestrukturen til prosjektet bli beskrevet og illustrert for å gi innsikt i hvor de forskjellige filene ligger. Til slutt vil bakgrunnsprosessene til webapplikasjonen bli redegjort for.

6.1.1 - Single Page Applikasjon (SPA)

SPA, eller 'Single-Page Application', er en dynamisk og effektiv måte å laste inn nettsider på. Dette gjelder routing av sider, data henting og API integrasjon (Adobe, 2023). Måten SPA fungerer, er ved å be serveren om all informasjon om komponenter først, og

deretter rådata, som SPA kan tolke for å å sette på de rette stedene. Når en side skal oppdateres, vil serveren tradisjonelt printe hele siden, men med SPA sendes alle elementer som komponenter som vises på de forskjellige sidene dynamisk, avhengig av hvor brukeren befinner seg i webapplikasjonen. Oppsummert er SPAs fordeler; engangs lasting for alle HTML, CSS og JS filer, færre spørringer til serveren og til slutt en raskere og mer responsiv side oppbygging (Takada, u.å). I Figur 12 nedenfor er det illustrert hvordan en SPA opererer.



Figur 12: Illustrasjon av SPA

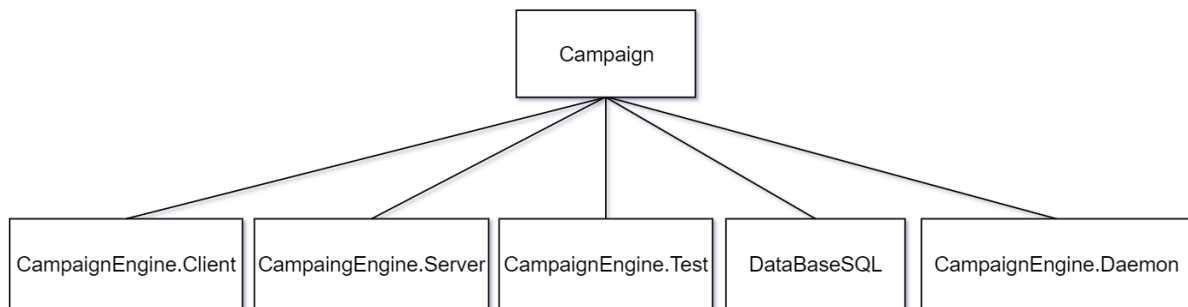
I gruppens prosjekt er SPA valgt på bakgrunn av dens fordeler som forbedret ytelse, brukeropplevelse, tilgangen på moderne utviklingsverktøy og muligheten for en smidigere utviklingsprosess. Web- applikasjonen utnytter i praksis SPA ved bruk av REACT Typescript- rammeverket. Dette rammeverket gjør det enklere å organisere koden i komponenter som lettere kan gjenbrukes og oppdateres uavhengig av hverandre. Dette er med på å gjøre web- applikasjonen mer skalerbar og strukturert. For at web- applikasjonen skal ta i bruk SPA- teknologien ble det brukt noe som heter klientside- ruting. Klientside- ruting betyr at det er en App- ruter som navigerer filbanen aktivt slik at web- applikasjonen unngår å måtte laste opp hele siden på nytt når brukeren navigerer seg på web- applikasjonen (Adobe, 2023). I tillegg gir SPA gruppen muligheten til å implementere funksjoner som datainnhenting i bakgrunnen ved hjelp av Axios- forespørsler.

6.1.2 - Mappestruktur

I webapplikasjonen har mappestrukturen en kritisk rolle hvis man tar i betraktning viktigheten med å vedlikeholde-, skalere- og organisere kode. Dette er med på å nå prosjektmålene. Derfor ble det dedikert god tid på å utforme mappestrukturen. I tillegg til dette krever SPA flere komponenter som fungerer uavhengig av hverandre, og derfor er en godt strukturert mappestruktur nødvendig for å administrere disse komponentene.

Nedenfor i Figur 13 er det øverste laget av prosjektets mappestruktur illustrert. Hovedmappen, 'Campaign', inneholder fem andre mapper. Disse mappene inneholder

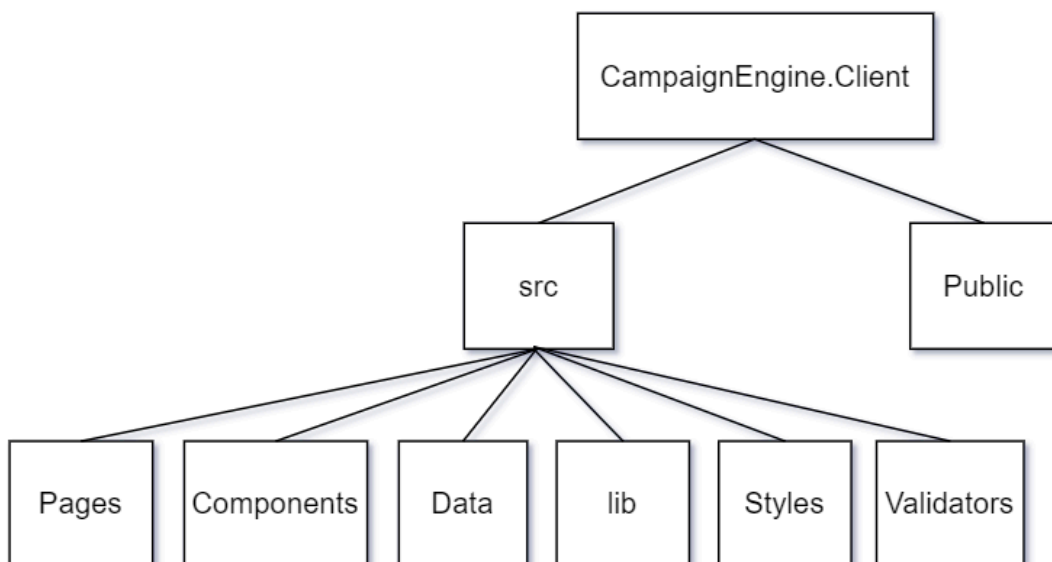
forskjellige aspekter ved prosjektets funksjonalitet. Prosjektets 'frontend' befinner seg i 'CampaignEngine.Client'. Mens forskjellige aspekter av prosjektets 'backend' befinner seg i de resterende mappene. 'CampaignEngine.Server' inneholder webapplikasjonens server. 'CampaignEngine.Test' inneholder alle enhetstester. 'DataBaseSQL' inneholder sql kode for oppsett av databasen. 'CampaignEngine.Daemon' inneholder prosjektets bakgrunnsprosesser.



Figur 13: Mappediagram som beskriver øverste laget

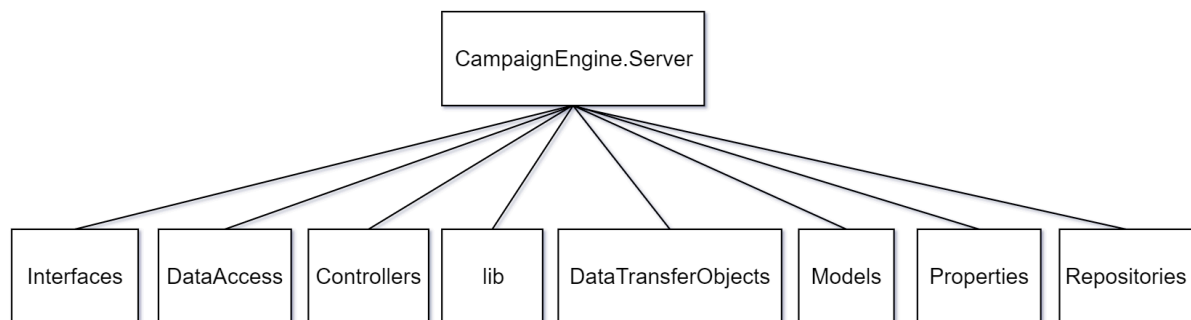
I 'CampaignEngine.Client' finnes det to hovedmapper; 'src' - mappen og 'public' - mappen. 'src' - mappen inneholder kildekoden til prosjektet. Her ligger det mapper som inneholder alle sider som skal vises til brukeren, komponentene sidene er bygget opp av, data, lib, stiler og validatorer. Dette er videre organisert i mapper ut ifra sammenhenger mellom komponenter og sider.

'Public' - mappen inneholder informasjon som serveres direkte til brukeren uten noe prosessering. Nedenfor i Figur 14 er strukturen til "Campaign Engine.Client"- mappen illustrert.



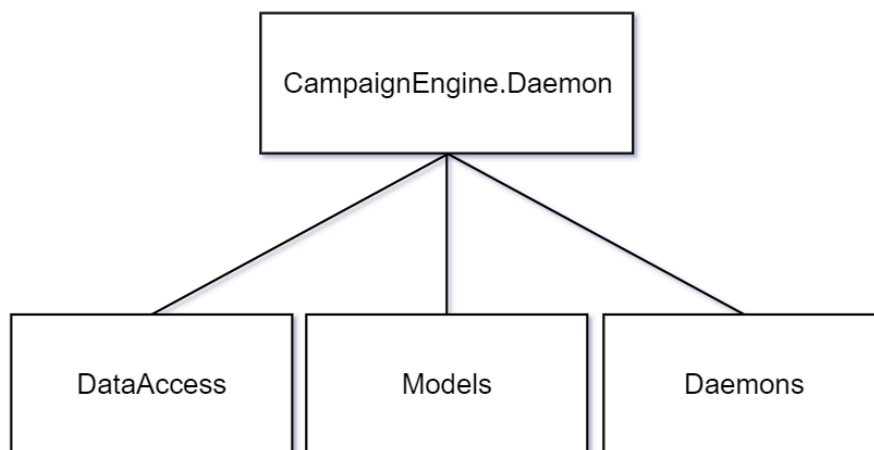
Figur 14: Mappediagram som beskriver klient laget

Mappen 'CampaignEngine.Server' inneholder webapplikasjonens server. Her har gruppen benyttet seg av en .NET CORE 'backend' arkitektur. Hver type kode blir lagt inn i sine egne mapper. Dette går ut på at alle dataobjekter som skal håndteres blir strukturert av modeller og 'data transfer objects' som hver har sine respektive mapper. Når en klient sender en forespørsel til serveren, blir den tatt imot av en 'controller' som kjører metoder på en 'interface' til en 'repository' som alle har sine egne mapper. Det 'repository' som sender spørringer til databasen ved hjelp av kode i 'DataAccess'- mappen. 'lib' mappen inneholder andre klasser som blir brukt av serveren. Denne strukturen er illustrert nedenfor i Figur 15.



Figur 15: Mappediagram som beskriver server laget

Avsluttende kommer 'CampaignEngine.Daemon'- mappen som tidligere nevnt, håndterer bakgrunnsprosesser som blir brukt av webapplikasjonen. Denne mappen inneholder tre undermapper som er; 'DataAccess', 'Models' og 'Daemons'. Disse fungerer som i 'CampaignEngine.Server'- mappen, bortsett fra 'Daemons' som inneholder koden for selve bakgrunnsprosessene.

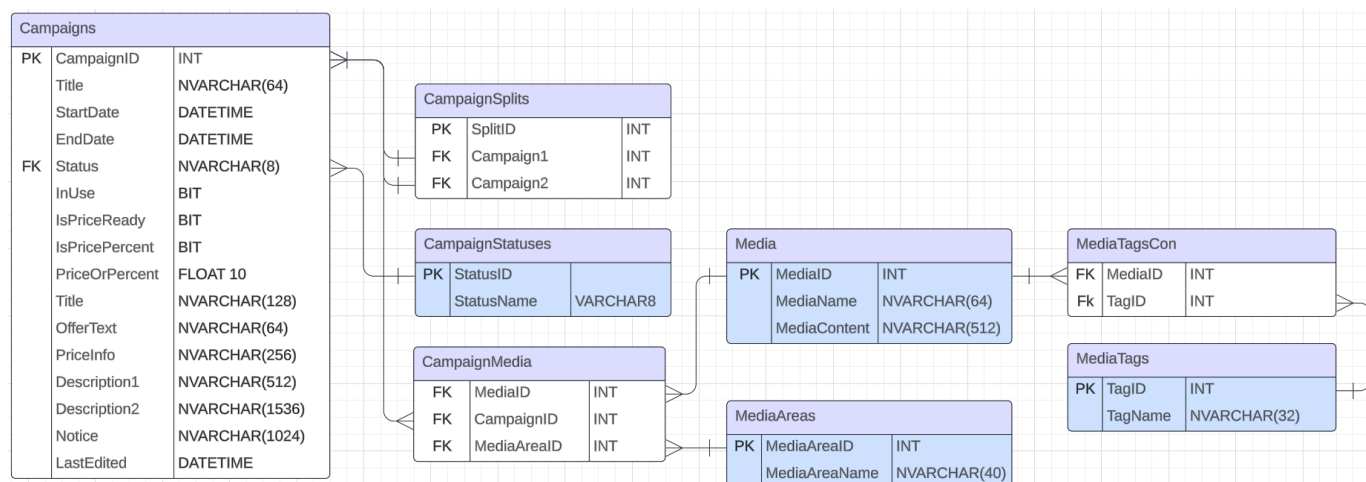


Figur 16: Mappediagram som beskriver Daemon laget

Erfaringene gruppen har hatt ved å bruke denne mappestrukturen, er at det gir muligheten for å enkelt navigere til ønskede filer. I tillegg gjør det muligheten for skalering større ettersom at mappene for hver enkelt fil er strukturert. Utfordringene gruppen har hatt med mappestrukturen var hovedsakelig det å komme frem til hvordan det er strukturert. Det er

gjort endringer underveis i prosjektet for å forbedre mappestrukturen, ettersom gruppen benyttet seg av prinsippet ‘learning by doing’.

6.2 - Database



Figur 17: Database av kampanjemotor-webapplikasjonen

Databasen inneholder dataen som genereres av webapplikasjonen. Under designet av databasen ble det lagt vekt på videreutvikling, optimalisering og skalerbarhet. Databasesdesignet ble endret i løpet av prosjektet på grunn av tekniske utfordringer i frontend. Det første designet inneholdt flere tabeller. Noen av disse tabellene ble vurdert som komplisert å bruke, og ble derfor fjernet eller integrert i databasen Figur 17. ovenfor. Det første databasesdesignet finnes i [appendiks A5](#). Endringene som ble gjort førte til en lettere data flyt implementasjon mellom frontend og backend.

Databasen inneholder hovedtabellen Campaigns, som er koblet opp mot tabeller som lagrer informasjon med bilder, statuser og tags.

6.3 - Bakgrunnsprosesser

Webapplikasjonen er avhengig av å kjøre visse operasjoner automatisk i bakgrunnen. For å oppnå dette finnes det flere metoder med sine fordeler og ulemper. Gruppen bestemte seg for å fokusere på motstandskraft og optimalisering. Dette betyr at det var to alternativer for hvilken teknologi som skulle brukes for å utvikle dette aspektet av webapplikasjonen. Disse alternativene var innebygd funksjonalitet i Windows Services og Task Scheduler, eller å lage alt fra bunnen av. Siden bedriften foretrekker å bruke Linux'Containers' for å kjøre servere, betydde dette at gruppen ikke kunne bruke Windows Services og Task Scheduler. Videre ble valg som ble tatt basert på de teknologier og tjenester bedriften allerede bruker, og til slutt ble en 'Console App Daemon' valgt som den beste løsningen å utvikle (systemd, u.å.). En 'Daemon', kort for 'Console App Daemon', er et program eller prosess som kjører som en bakgrunnsprosess (Wikimedia Foundation, Inc, u.å.).

6.3.1 - Funksjonalitet

Denne daemonen skal kjøre bakgrunnsprosesser som sjekker om det er noen kampanjer som skal endre status til 'Aktiv' eller 'Utgått', og endre de til den status ved det klokkeslett som er satt. Den må være presis og kunne korrigere seg selv, samt skrive logg om feil og kunne gjenopprette etter krasj.

Dette betyr at den må bestå av en hovedprosess som vil kjøre ved midnatt hver dag, og starte en tidsmåler for hver kampanje som skal endre status. Den sjekker også for kommende oppgaver i løpet av dagen og tapte oppgaver dagen før, og gjennomføre disse. Dette sikrer at hvis en feil skulle skje, ville det ikke føre til at hele tjenesten låser seg eller data blir håndtert feil over lengre tid.

Den skal også kunne sende påminnelser til relevante ansatte i bedriftens IT avdeling og markedsføringsavdeling når dette trengs, for eksempel når en kampanje er klar til godkjenning, eller blir aktiv.

6.4 - Kvalitetssikring

Gruppen har tatt i bruk en rekke tiltak for å sørge for at kampanjemotorens arkitektur opprettholder et ønsket nivå av kvalitet. Av de kvalitetskriteriene beskrevet i kapittel 3.4.2 er det her 'Effektivitet', 'Kompatibilitet', 'Pålitelighet', 'Systemsikkerhet', 'Vedlikeholdbarhet' og 'Fleksibilitet' som er viktigst.

Gjennom møter med ansatte i bedriftens IT avdeling har gruppen fått en forståelse for hvordan kampanjemotoren skal passe inn i bedriftens miljø, og har tilpasset webapplikasjonens arkitektur deretter. Dette var viktig fordi den måtte kunne kommunisere med andre systemer i bedriftens miljø. At bedriften skal ta over og videreutvikle prosjektet gjorde det viktig for gruppen å sørge for at arkitekturen er strukturert på en måte som gjør at den som skal videreutvikle raskt kan sette seg inn i prosjektet.

7.0 - Prosjektgjennomføring

Dette kapittelets formål er å vise gruppens prosedyrer for arbeid innad i prosjektet. Kapittelet tar for seg emner som rapportering, utviklingsmiljø, kompetansebygging og hvordan gruppen har sørget for at gjennomføringen følger samme kvalitet som resten av arbeidet som er utført i prosjektet.

7.1 - Rapportering

Rapporter er et nødvendig aspekt av ethvert prosjekt og gruppen har hatt fokus på å skrive ned detaljer for å kartlegge framgangen i prosjektet. Både for å ha et retrospektivt innsyn i progresjonen og fremtidige mål for oppnåelse.

7.1.1 - Rapportering og Møter

Fra start hadde gruppen stand-up møter hver mandag og fredag. Det ble også avholdt sprint planning og sprint retrospektiv. Gruppen var på kontorene til bedriften alle ukedager, fra 09:00 til 15:00, utenom onsdager.

Hver uke ble det skrevet ukesrapporter, som inneholdt hva gruppen hadde jobbet med. Dette ga en oversikt over oppgaver som hadde blitt utført, og hva vært medlem bidro med.

Ukesrapportene finnes i [appendiks A10](#).

Som tidligere nevnt har gruppen også hatt kontakt med bedriften hver uke, angående prosjektets fremgang og diverse spørsmål knyttet til prosjektet. Denne gode kontakten har ført til at detaljer rundt prosjektet har vært enkelt å fange opp, samt vært en sikring på at prosjektet har vært i rute.

7.1.2 - Fremgangsplan

For å sikre en effektiv prosjektgjennomføring og legge en plan for fremgang, hadde gruppen møter som heter 'Sprint planning'. På møtene ble det satt mål for sprinten, og fordelt oppgaver. For å videre følge opp de satte målene, hadde gruppen to ukentlige stand-up møter, som nevnt tidligere. Her var hovedformålet å se framgang, sette nye mål og tidsestimering av visse oppgaver. Tidsestimeringen finnes i [appendiks A8](#). Etter enhver sprint utførte gruppen en 'Sprint retrospective', for å få oversikt over fremgangen til prosjektet og sammenligne mot fremgangsplanen satt under sprintplanleggingen, se [appendiks A9](#).

Utover prosjektgjennomføringen ble det gjort noen endringer, og gruppen har erfart hvor vanskelig prosjektstyring og oppfølging kan være. Ukentlige stand-up møter ble gjort om til daglige, for å møte de mer dynamiske tekniske problemene gruppen prøvde å løse.

Tidsestimering ble også gjort om til story points, for å kunne beskrive kompleksiteten av oppgavene bedre. Oppsummert har fremgangen av prosjektet vært jevn, og grep ble gjort for å opprettholde god fremgang.

7.2 - Utviklingsmiljø

En rekke teknologier og rammeverk har blitt brukt i prosjektet, og utgjør sammen utviklingsmiljøet. Disse kan deles inn i følgende kategoriene vist i tabellen til høyre:

1.	Programvare
2.	Programmeringsspråk
3.	Rammeverk
4.	Biblioteker
5.	Annet

Hver av disse kategoriene inneholder en eller flere teknologier som alle blir brukt i prosjektet. Når hver teknologi og verktøy skulle velges ble ofte flere vurdert etter forhold som:

1. Hvor godt kjent med teknologien/verktøyet er gruppens medlemmer?
2. Hvor godt kjent med teknologien/verktøyet er bedriften?
3. Hva innebærer implementasjon av teknologien/verktøyet for prosjektet?
4. Hvordan vil teknologien/verktøyet påvirke prosjektets ressursforbruk?

I de aller fleste tilfeller har gruppen brukt teknologier og verktøy som er godt kjent av gruppens medlemmer, slik at gruppen ikke skulle bruke for mye tid på å tilegne seg kompetanse. Det var også et mål for gruppen å bruke hovedsakelig teknologier og verktøy som bedriften er kjent med, og allerede bruker i sine egne systemer.

Dette kommer av at kampanjemotoren skal videreutvikles av bedriften etter prosjektet er gjennomført av gruppen, og fordi at kampanjemotoren lettere skal kunne integreres med bedriftens andre systemer. De teknologier som oppfyller begge disse kriteriene er: Visual Studio Code, C#, SQL, ASP.NET, Dapper ORM, npm, og NuGet. De teknologier som ble valgt fordi bedriften bruker det i sine egne løsninger, men som gruppen selv ikke kjenner godt er: MS SQL og Server Management, MinIO, Keycloak, Kubernetes, TypeScript, og React. Om en teknologi eller verktøy ble valgt på annet grunnlag er dette forklart nedenfor med den teknologi eller verktøy det gjelder.

7.2.1 - Programvare

Visual Studio Code

Samtlige av gruppens medlemmer bruker Visual Studio Code (Visual Studio, u.å.) som IDE (Integrated Development Environment) til å utvikle og skrive koden som utgjør webapplikasjonen.

Microsoft SQL Server (MS SQL)

MS SQL er et system for håndtering av relasjonelle databaser utviklet av Microsoft (Wikimedia Foundation, Inc, u.å.), bygd på SQL. Gruppen valgte dette systemet fordi webapplikasjonen trengte en database for håndtering av data og informasjon.

Server Management

Server Management er en programvare laget av Microsoft som brukes til å administrere windows baserte servere (Microsoft, 2022). I dette prosjektet har Server Manager hovedsakelig blitt brukt for å sette opp en lokal server på hvert gruppemedlems datamaskin

slik at de kunne ha en testversjon av prosjektets database tilgjengelig. Programmet har også blitt brukt for å inspisere databasens tilstand, og hvordan den reagerer på kommandoer fra webapplikasjonens backend. Denne programvaren ble brukt fordi den følger med MS SQL.

MinIO

MinIO er et lagringssystem som kan bearbeide ustrukturerte data som bilder, video og loggfiler, etc (Wikimedia Foundation, Inc, u.å.). I prosjektet blir MinIO brukt til lagring av mediafiler som skal brukes av webapplikasjonen. MinIO ble valgt av to grunner. Den første grunnen er at gruppen trengte en måte å lagre bildefiler på en webbasert løsning utenfor databasen. Den andre grunnen er at dette er en teknologi bedriften selv har begynt å implementere i andre løsninger.

Keycloak

Keycloak er en webbasert tjeneste som håndterer og administrerer bruker-autentisering for webapplikasjoner og tjenester (Keycloak, u.å.). Dette er et system som blir brukt av bedriften selv for å sørge for at bare kunder og ansatte har tilgang til de forskjellige tjenestene de skal ha tilgang til. For eksempel, når en kunde logger seg inn i Ditt Phonero, blir de autentisert av Keycloak slik de får tilgang til bare de tjenester de skal ha tilgang til. Derimot vil en ansatt hos bedriften bli autentisert av Keycloak på en slik måte at de får tilgang til en del interne tjenester, avhengig av hvilken tilgang de har blitt tildelt som ansatt.

Siden webapplikasjonen skal være et internt verktøy som bare skal bli brukt av noen få ansatte i bedriften, og fordi bedriften allerede bruker Keycloak, valgte gruppen å integrere Keycloak som autentiserings verktøy for å oppnå dette.

Kubernetes og docker

Kubernetes er et system for automatisering av programvare distribusjon, skalering, og håndtering av containeriserte applikasjoner (Kubernetes, u.å.). Webapplikasjonen ble i løpet av prosjektets løp distribuert på en kontainer i bedriftens eget kubernetes miljø, slik at den ble tilgjengelig for testing og bruk av bedriftens ansatte.

Docker er en tjeneste som bruker OS nivou virtualisering til levere programvare inni containers, den følger i principe “vært OS kjører et programvare” men det er fremdeles mulig flere apps inne i samme container (Wikimedia Foundation Inc., u.å.). Phonero bruker Kubernetes som en orchestrator til å deploye Docker containers.

7.2.2 - Programmeringsspråk

C#

C# er et høynivå objektorientert programmeringsspråk, og er laget av Microsoft (Wikimedia Foundation, Inc, u.å.). Gruppen valgte å bruk C# som programmeringsspråk på webapplikasjonens backend.

TypeScript

TypeScript er et språk laget av Microsoft som er basert på Javascript. TypeScript legger til typer og annoteringer til Javascript. Til tross for at gruppen selv er lite kjent med TypeScript ble dette språket valgt, fordi det tilbyr en rekke fordeler. Disse fordelene er:

- Type- sjekking
- Enklere å skalere
- Type- sikkerhet
- Lettere å refaktorere kode
- Bedre dokumentasjon og verktøy
- Javascript Inter operasjonelt
- Lesbarhet og vedlikehold

(Microsoft, u.å.). Bedriften bruker også denne teknologien selv.

Structured Query Language (SQL)

SQL er et programmeringsspråk som brukes for av relasjonelle databaser til å lagre og prosessere informasjon (Wikimedia Foundation, Inc, u.å.). MS SQL bruker SQL som språk for å håndtere relasjonelle databaser. Siden gruppen har valgt å bruke MS SQL til håndtering av databasen, bruker derfor gruppen også SQL som programmeringsspråk for databasen.

7.2.3 - Rammeverk

ASP.NET

ASP.NET er et rammeverk for bygging av webapplikasjoner og webtjenester med .NET (Microsoft, u.å.). Dette rammeverket ble valgt av gruppen fordi gruppen skulle lage en webapplikasjon, og det er et av de mest brukte rammeverk for webapplikasjoner i dag.

Tailwind CSS

Tailwind CSS er et rammeverk for css som skiller seg fra andre css rammeverk ved at istedenfor å bygge css strukturen basert på definerte klasser rundt elementer som for eksempel knapper, bygger Tailwind CSS strukturen rundt klasser basert på bruk. Dette gjør at stil tilbudt av en klasse kan brukes av alle forskjellige elementer (Wikimedia Foundation, Inc, u.å.). Gruppen valgte å benytte seg av Tailwind CSS for brukergrensesnittet i utviklingsprosessen på grunn av dets unike 'utility-first' tilnærming til styling, som både fleksibel og effektiv (Material Tailwind, 2023). Ved å implementere dette rammeverket i prosjektet, gjorde det mulig til gruppen å legge til stil direkte i HTML-elementer, uten behov for separate CSS-filer (Material Tailwind, 2023). Dette bidrar til en enkel og rask design implementasjon i utviklingen av webapplikasjonen. I tillegg tilbyr Tailwind CSS en rekke av forhåndsdefinert av klasser som gruppen kan benytte og gjenbruke på tvers av prosjektet som er enklere å redigere og vedlikeholde uten å bekymre om CSS-oppblåsthet (Rajak, 2023).

7.2.4 - Biblioteker

React

React er et Javascript-bibliotek for å bygge opp brukergrensesnitt ved hjelp av komponenter (React, u.å.). React forenkler utviklingen av dynamiske webapplikasjoner ved å krever mindre koding samtidig som det tilbyr mer funksjonalitet (Guide & Deshpande, 2023). Ved bruk av React, blir gjenbruk av komponenter mer effektiv, siden hver komponent er selvstendig med egen logikk og kontroll, som kan bli brukt på tvers av hele applikasjonen. Dette bidrar til å redusere utviklingstiden for applikasjonen (Guide & Deshpande, 2023). Selv om gruppen ikke har erfaring med React, valgte vi å bruke det i utviklingen av kampanje-webapplikasjonen. Gruppen gjorde dette valget fordi det gir oss muligheten til å lære og bli kjent med React, som er veldig populært brukt til frontend-utvikling. I tillegg benytter bedriften også React, noe som vil forenkle deres videreutvikling av kampanje-webapplikasjonen etter at vi har levert produktet til dem. Det gjør også enklere for gruppen å søke hjelp og lære fra bedriftens erfarne utviklingsteam, siden de har mye erfaring med React.

Shadcn/ui

I dette prosjektet benyttet gruppen seg av komponenter fra Shadcn/ui, en omfattende samling av gjenbrukbare komponenter (Shadcn/ui, u.å.). Gruppen valgte å bruke komponenter fra Shadcn/ui, ettersom dette biblioteket tilbyr mange utvalg av komponenter som passer veldig bra i vårt prosjekt. Komponentene fra Shadcn/ui ble effektivt gjenbrukt på tvers av ulike sidene i kampanje-webapplikasjonen. På den måten unngikk gruppen kode-duplisering, sparte tid sammenlignet med å utvikle egne komponenter, og sikret et konsistent oppsett i hele applikasjonen. I tillegg tillot Shadcn/ui komponentene gruppen å endre stil ved bruk av Tailwind CSS, slik at det blir helt likt som vårt design i Figma.

Dapper ORM

Dapper er et ORM (Object Relational Mapping) bibliotek lagd for bruk innenfor .NET rammeverket (Wikimedia Foundation, Inc, u.å.). Spesifikt brukes det til å sende og hente data mellom en server og tilknyttede databaser. Dapper krever mye mindre ressursforbruk i forhold til andre ORM, slik som for eksempel Entity Framework Core. Bakdelen med bruk av Dapper er at man ikke kan genere SQL basert på kode, men må skrives manuelt og brukes som variabler. Dette betyr derimot at ved å kontrollere brukeres tilgang til variabler og funksjons parametre, kan SQL injisering unngås (Learn Dapper, 2023). Dette er også grunnene til hvorfor gruppen valgte å bruke Dapper.

Axios.js

Axios.js er et Javascript bibliotek som tilbyr en HTTP klient for webapplikasjoner (Axios, u.å.). Fordi kampanjemotoren bruker React, som er et Javascript basert bibliotek, betyr dette at den også må bruke en HTTP klient som støtter Javascript. Axios var også tilgjengelig via npm, beskrevet nedenfor. Dette gjorde at gruppen valgte Axios fremfor andre HTTP klienter.

7.2.5 - Pakkebehandling

Node Package Manager (npm)

Node Package Manager er en pakkebehandler for JavaScript som er laget av og vedlikeholdt av Microsoft. Hovedsakelig brukes denne som en kommandolinje klient til å installere og håndtere modulpakker til prosjektløsninger (Wikimedia Foundation, Inc, u.å.). Det er også nettopp dette npm blir brukt til i dette prosjektet. Hovedsakelig ble npm valgt fordi dette gjør det enkelt for gruppen å installere pakker til prosjektet som kan brukes av de deler av prosjektet som er skrevet i TypeScript. Men npm tillater også noen pakker brukt av C# delen av prosjektet.

NuGet

NuGet er en pakkebehandler og programvare distributør hovedsakelig knyttet til .NET rammeverket (Wikimedia Foundation, Inc, u.å.). I dette prosjektet har NuGet blitt brukt til å installere en rekke pakker og bibliotek som blir brukt av webapplikasjonen. Grunnen til at gruppen bruker to pakkebehandlere (npm og NuGet), er at mange pakker bare blir distribuert på enten den ene eller andre, avhengig av hvilket rammeverk de er knyttet til og rett og slett hvordan skaperne av pakken distribuerer den.

7.3 - Kompetansebygging

Ved utvikling av et prosjekt er det nødvendig at gruppen tilegner seg riktig teknisk kompetanse for å imøtekomme prosjektets systemkrav. I prosjektet skulle gruppen ta i bruk teknologier som var kjente og ukjente fra tidligere av. Teknologiene som gruppen var ukjente med var React, Typescript, TailwindCSS, Figma, Dapper, Axios og MSSQL. Teknologiene som gruppen har erfaring med fra før av var C#, SQL og .NET. Måten gruppen tilnærmet seg erfaring med de ukjente teknologiene var gjennom en variasjon av læringsteknikker.

Mye av læringen foregikk ved å lære gjennom å gjøre. Det går ut på at medlemmene prøvde å utføre oppgaver, også underveis finne løsninger. Dette ble gjort gjennom å lese dokumentasjon, se videoer og lese på forumer som StackOverflow underveis i utviklingen. Gruppen fikk også hjelp med å tilnærme seg kunnskaper gjennom å kommunisere med utviklerne som jobber hos bedriften.

I tillegg til individuell innsats, fikk også gruppen nytte av å lære av hverandres erfaringer og kunnskaper med forskjellige teknologier. Dette foregikk gjennom møter og felles problemløsning. Gruppen arbeider i et åpent kontorlandskap som gjør det enkelt å spørre hvis en av medlemmene trenger hjelp til å løse et problem.

De nevnte måtene å tilnærme seg kunnskap på har utgjort en stor rolle gjennom hele prosjektgjennomføringen, og derfor har kommunikasjon hatt en stor betydning for resultatet av prosjektet. Erfaringene vi har fått gjennom å lære gjennom de nevnte teknikkene er at man lærer på en god måte, men at det til tider kan være tidkrevende ettersom man i noen tilfeller er nødt til å bruke mye tid på å finne informasjon. Andre erfaringer gruppen har hatt er at hvert medlem står ansvarlig selv for å kommunisere med andre medlemmer hvis man har vanskeligheter med å finne fram til svar. Hvis ikke kan dette føre til at man bruker lengre tid enn antatt på å lære, og det er ikke optimalt tidsbruk.

På bakgrunn av erfaringene vil gruppen konkludere med at læringsteknikkene mer eller

mindre har vært effektivt i prosjektgjennomføringen, men at det også kunne blitt introdusert andre, mer kontrollerte former for opplæring for å sørge for lik læring.

7.4 - Kvalitetssikring

Det å kvalitetssikre gjennomføringen av prosjektet er særdeles viktig for å levere et solid produkt (Nair, 2013) . Leddene som inngår i prosjektgjennomføringen er prosjektstyring, der rapportering, møter, fremgangsplan og målsetting har en stor rolle i det å sette rammene i prosessen. Det andre leddet er selve prosessen som inngår i systemutviklingen, også til slutt risikovurderingen som inneholder punkter som bruker- og enhetstesting.

7.5.1 - Prosjektstyring

Innen prosjektstyring har vår gruppe implementert en grundig tilnærming for å sikre kvalitet i gjennomføringen av prosjektet. Tilnærmingen gruppen har tatt er å ha en definert struktur og organisering innad i prosjektet. Det som er inkludert, er å ha definerte roller og ansvarsroller for hvert medlem i gruppen, samt å fremme at medlemmene tar selvstendig ansvar i å individuelt sikre fremdrift. Dette bidrar til å sikre at alle er opplyste om hva som forventes av hvert enkelt medlem og hvilke oppgaver som skal gjøres. I tillegg til å definere ansvarsroller og stille forventninger til medlemmene, har vi også implementert regelmessige risikovurderinger for å lettere identifisere potensielle utfordringer tidlig i prosjektet.

Gruppen har også hatt fokus rundt det å ha en klar og effektiv kommunikasjon innad i gruppen. Det som inngår i å oppnå dette er å holde regelmessige møter med en planlagt agenda. Dette er også med på å sikre at medlemmene i gruppen er oppdatert i prosjektets fremgang samtidig som alle har mulighet til å delta i å diskutere beslutninger som blir tatt. Møtene som blir holdt fungerer som en plattform der medlemmene kan diskutere eventuelle problemer, løsninger og beslutninger.

I tillegg til møter og kommunikasjon innad i gruppen, ble det også rapportert om mål, fremdrift og møter med oppdragsgiver. Rapporteringen som er blitt gjort gjennom prosjektet er av typen ukesrapporter, møtenotater og målsettinger. Det er med på å gi alle medlemmer oversikt over hvordan gruppen ligger i prosessen med prosjektet.

Kort oppsummert har tiltakene bidratt til å sikre kvaliteten på prosjektstyringen gjennom å sikre en solid organisering, effektiv risikohåndtering og effektiv kommunikasjon.

7.5.2 - Utviklingsmiljø

En del av kvalitetssikring av utviklingsmiljøet er å sørge for at de valgte teknologier og verktøy samhandler, er trygge i bruk, og at de ikke krever for mye ressurser i bruk. Utviklingsmiljøet inneholder også en rekke mindre teknologier og verktøy som ikke har blitt diskutert i denne rapporten, og som heller ikke vil bli diskutert da dette risikerer å gjøre rapporten alt for teknisk. Derimot er det viktig å forklare hvordan det ble sørget for at disse oppfylte krav for kvalitet.

De faktorer introdusert i kapittel 3.4.2 som er viktigst å bruke ved valg av teknologier og verktøy som skal brukes i utviklermiljøet er Effektivitet, Kompatibilitet, Pålitelighet, Systemsikkerhet, Vedlikeholdbarhet, og Fleksibilitet. Til en mindre grad er også funksjonell egnethet viktig her, men dette dreier seg mer om at de forskjellige teknologiene og verktøyene egner seg, både for seg selv og sammen, til å skape den ønskede funksjonaliteten.

Hovedsakelig har gruppen fokusert på at valgte teknologier skal være sikre i bruk, og at de ikke skal introdusere unødvendig kompleksitet til kampanjemotoren. Det er viktig at systemet oppfyller bedriftens krav til sikkerhet, slik at ingen av deres, eller deres kunders data kommer på avveie. Det er også viktig at når bedriften skal påbegynne videreutvikling av kampanjemotoren, så påfører ikke gruppens valg av teknologier unødvendig behov for kompetansebygging på bedriftens side, samt at de er godt egnet til integrering med bedriftens eksisterende systemer.

7.5.3 - Enhetstesting

Enhetstesting er en metode som ofte brukes gjennom utviklingsfasen av et prosjekt og kan hjelpe med å kvalitetssikre og identifisere problemer før utgivelse. Det handler altså om å isolere spesifikke deler av koden, eller seksjonere komponenter, så de kan bli testet individuelt. Målet med enhetstesting er å opprettholde prosjektet under vekst av kode-basen. Dette vil si at man kan ha en stabil progresjon underveis, istedenfor å oppdage store feil med arkitekturen av koden på slutten av utviklingsfasen. Ved lite implementasjon av enhetstesting kan man risikere fenomenet “software entropy”, altså uorden eller kaos i koden. Enhetstesting fungerer derfor som et sikkerhetsnett, for å unngå at koden faller sammen (Khorikov, 2020).

Gruppen valgte å teste gjennom å bruke verktøyet NUnit, som er et etablert verktøy for testing innen .NET. Hovedformålet ble å segmentere metodene knyttet til opprettelse, sletting, redigering og henting av kampanjer fra databasen og teste dem individuelt. Dette ble gjort med bruk av verktøyet MOQ, som avspeiler eller forfalsker metodene i repository. Gjennom denne metoden kan man kjøre testene hver gang kode-basen endres og forstå koden bedre, som deretter kan føre til refaktorering av de opprinnelige komponentene.

Enhetstesting validerer disse kvalitetsfaktorene for gruppens prosjekt: funksjonell egnethet, pålitelighet og vedlikeholdbarhet.

8.0 - Refleksjon

I dette kapittelet vil gruppens refleksjoner rundt ulike temaer som gjennomføringen av prosjektet, kvalitetssikring og resultater presenteres. Alle i gruppen har drøftet de ulike temaene sammen, slik at refleksjonene som presenteres, tilhører gruppen som en helhet.

8.1 - Prosjektgjennomføring

8.1.1 - Agil prosjektmetodikk

Gruppen bestemte seg for å arbeide med den agile prosjektmetodikken “ScrumBan”, som beskrevet i kapittel 3.1. Som tidligere nevnt holdt gruppen faste møter hver mandag og fredag, for kartlegging og allokering av oppgaver. Dette fungerte godt i en god periode. Oppgaver ble løst innen rimelig tid, og prosjektet var på riktig vei i forhold til satte frister/milepæler.

Litt før påskeferien, merket gruppen at det lå en del oppgaver igjen fra sprinten før (Sprint x). Dette mente gruppen kunne ha sammenheng med at de tekniske oppgavene som ble utdelt, gradvis ble mer komplekse. For å få en bedre oversikt over oppgaver som ble utdelt, innførte gruppen daglige stand-ups fra og med sprint x. Dette var for å bedre allokere oppgaver, men også gi alle medlemmer muligheten til å få innspill på hvordan de komplekse oppgavene kunne løses. Endringen førte til at oppgaver ble løst raskere og prosjektet gled inn på riktig spor igjen. Hovedfordelen med denne endringen var at kommunikasjonen rundt hver enkelt oppgave ble mye bedre, noe som førte til mer forståelighet og effektivitet.

8.1.2 - Gruppeinndeling

Som beskrevet i kapittel 1.3 delte gruppen seg inn i tre ulike grupper: To Frontend, to Backend og to Fullstack. De administrative oppgavene ble fordelt utover i gruppen. Dette gjorde gruppen på grunn av å spille på hverandres styrker, interesser, og for å gjøre det lettere å allokere oppgaver.

Dette fungerte meget godt gjennom hele prosjektet. Inndelingen førte til at medlemmene raskt kunne kartlegge oppgaver innenfor sine domener. Samtidig ble det opprettholdt god kommunikasjon mellom alle gruppemedlemmene. Dette gjorde at oppgaver som krever samarbeid mellom Front- og Backend ble løst på en god måte. Alle gruppemedlemmene hadde erfaring fra å jobbe med C# og .NET fra et tidligere prosjekt. Dette gjorde at oppgavene som tilhørte de nevnte teknologiene var lettere å gjennomføre. Grunnet lite erfaring med React og Typescript førte til at noen av oppgavene på FrontEnd tok noe lengre tid enn oppgaver som skulle løses på Backend. Hvis gruppen hadde vanskeligheter med å utføre oppgaver innenfor Front - eller Backend, var medlemmene fleksible nok til å ta på seg oppgaver utenfor deres ‘domene’.

8.1.3 - Kommunikasjon

Gjennom hele prosjektet har gruppen opprettholdt god kommunikasjonsflyt med bedriften. Det har vært en stor fordel å få egne kontorplasser, med oppsett, hos bedriften. Gruppen inviterte personer fra bedriften på retrospektiver, som ble holdt i slutten av hver sprint. Dermed fikk bedriften oppdatering på prosjektets fremgang. Den gode kommunikasjonsflyten førte også til at spørsmål knyttet til prosjektet ble oppklart tidlig, samt at gruppen fikk diskutert prosjektutviklingen regelmessig, for å sikre fremgang og kvalitet.

Gruppen bestod av seks medlemmer gjennom hele prosjektet. Dette antallet gjør at en god kommunikasjonsflyt ble nødvendig. Grunnen til dette er slik at alle medlemmer skulle være oppdatert på oppgavene som skal gjennomføres, og status på oppgaver som gjenstår. Gruppen har opprettholdt god kommunikasjon internt, gjennom hele prosjektet. Dette har ført til at gruppen har fått en stor fordel når det kommer til hvordan samarbeidet i gruppen har vært. Alle medlemmer har sett og snakket med hverandre nesten daglig (ukedager), gjennom hele prosjektets løp. Dette har gjort det enkelt å kommunisere rundt oppgaver som skal løses, og dermed økt effektiviteten.

8.2 - Kvalitet og Kvalitetssikring

Spørsmålene rundt kvalitet og kvalitetssikring var noe gruppen var meget oppmerksomme på, gjennom hele utførelsen av bachelorprosjektet. Spørsmål som, hvordan de enkelte komponentene som ble utviklet var “gode” nok, eller hvordan gruppen skulle kvalitetssikre designet? Det ble ofte diskutert rundt hvordan koden som ble utviklet og hvordan prosjektstyringen bidro i forhold til kvalitetssikring av gjennomførelsen og produktet, nevnt i kapittel 3.4. Dette førte til at alle gruppemedlemmer var inneforstått og med på valg som ble gjort underveis i prosjektet, for å sikre kvaliteten.

Alle kapitlene som omhandler prosjektgjennomføring og webapplikasjonen i rapporten er dedikert et underkapittel om hvordan gruppen har kvalitetssikret hver ‘del’ av prosjektet.

Gruppen mener at alle kvalitetssikringstiltak gjort før og i løpet av prosjektet har ført til at kvaliteten på både prosjektgjennomføringen og produktet som er utviklet, er god. Det har tidligere blitt definert hva kvalitet er, og dette innebærer at en gjenstand eller tjeneste har evnen til å tilfredsstille brukerens krav og forventninger. Dette er noe gruppen mener er gjort, og som også gjenspeiles i uttalelse fra oppdragsgiveren. Uttalelsen finnes i kapittel 8.5.

8.3 - Utfordringer og Erfaringer

Under arbeidet med bachelorprosjektet møtte gruppen på en rekke utfordringer, hvorav de største var knyttet til bruken av React og Typescript som nevnt tidligere i kapittel 7.3. Siden begge teknologiene var nye for gruppens medlemmer, var det ganske utfordrende i begynnelsen av utviklingen. Gruppen brukte mye tid på å bli kjent med React rammeverket, komponenter og TypeScript syntaks. Positivt, ga denne prosessen gruppen mulighet til å lære mer om hvordan man kan bygge opp en egen komponent som passer til prosjektet, samt å integrere og tilpasse eksisterende komponenter fra forskjellige kilder på nettet.

For å støtte gruppens arbeid med kodingen, benyttet gruppen seg av ChatGPT som et hjelpeverktøy. ChatGPT hjalp til med å tolke og feilsøke koden, spesielt når utfordringer rundt forståelse av kode skrevet av andre medlemmer i gruppen eller fra andre kilder som ble funnet på nettet. Ved bruk av dette verktøyet forbedret gruppen sin forståelse av React, og det hjalp til med å raskere løse problemer som dukket opp under utviklingen.

Gruppen har også hatt utfordringer med utvikling av bakgrunnsprosesser og å finne nok ressurser og informasjon til å lære hvordan dette skal foregå. Gruppen fant at læringsressurser hadde mangler når det kom til detaljerte beskrivelser om funksjonalitet og kompatibilitet. Dette gjorde at det tok lang tid å få nødvendig forståelse for å gjennomføre oppgaven.

8.4 - Resultat / konklusjon

Gruppen hadde mål som ble satt internt i gruppen, før prosjektets start. Samtidig ble noen mål satt med oppdragsgiver i startfasen til prosjektet. En generell oversikt over mål og ambisjoner ble presentert i kapittel 1.4. En kort oppsummering av dette var å avslutte bachelorutdanningen med et godt resultat, tilegne seg ny kunnskap og levere et produkt av god kvalitet.

Et overordnet mål som ble satt med bedriften i starten av prosjektet var å kunne lage, og distribuere en kampanje til 'development' miljøet til Ditt Phonero.

Gruppen kom ikke så langt at applikasjonen kunne distribuere kampanjer til utviklingsmiljøet til Phonero. Dette var på grunn av komplikasjoner som oppstod i forhold til integrering av applikasjonen til miljøet. Det måtte mer tid og utvikling til på både systemene til bedriften og webapplikasjonen før den kunne integreres helt. Dette fikk dessverre ikke gruppen eller bedriften mulighet til å gjøre før semesteret var over - men planen til bedriften er å implementere dette etter at de har overtatt prosjektet.

Gruppen er stolt over egen innsats og mener at målet som ble satt internt i gruppen er nådd. Alle i gruppen har tilegnet seg veldig mye kunnskap, når det kommer til nye og kjente teknologier. Det har blitt gjort mange erfaringer rundt prosjektstyring og prosjektgjennomføring. Etter tilbakemeldinger fra oppdragsgiver og god innsats kan det også argumenteres for at gruppen har levert et produkt av god kvalitet. Det har vært et lærerikt og engasjerende semester, der erfaringer og kunnskap vil bli tatt med videre til arbeidsliv eller masterutdanning.

8.5 - Uttalelse fra Oppdragsgiver



Kristiansand 15.05.2024

Til rette vedkommende

Studentene Markus Løvås Fjelltn, Linh Phuong Tra, Laurits Lund, Zeyd Bouzidi, Markus Skau og Ottar Worren utførte sitt Bachelor prosjektet hos oss i Phonero. Vi ønsker i den forbindelse å skrive en uttalelse om hva studentene skulle levere samt vår opplevelse av måten gruppen leverte på.

Oppgaven gruppen fikk fra oss omhandlet i korte trekk å designe og utvikle en kampanjemotor for å forenkle måten Phonero lager og lanserer kampanjer ut mot våre kunder. Dette er tenkt som et internt verktøy for avdelingene Produkt og Marked, hvor de fritt skal kunne opprette og lansere kundekampanjer basert på produkter fra porteføljen vår. Oppgavespesifikasjonen omhandler blant annet presentasjon av kampanjeproduktet med priser, kampanjenavn, beskrivelse, til/fra dato mm.

Vi erfarte at studentene, på en meget god måte, utførte følgende oppgaver/leveranser:

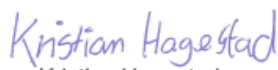
- Designet ny applikasjon – studentene hadde ingen restriksjoner eller konkrete krav på GUI fra vår side, men de måtte finne ut hva brukerne ønsket og levere på disse kravene
- De var begrenset til å bruke verktøyene og teknologier som vi besitter og bruker for at vi enklere skulle kunne ta over videreutviklingen
- Brukeravklaringer og brukertesting mot de avdelingene som skal ta i bruk ny applikasjon
- Utviklet fungerende MVP med Database, API og Web-klient
- Designet databasemodell for lagring av all informasjon på en fornuftig måte
- Integrert Minio (Kubernetes S3 lagring) for lagring av grafikk til bruk i kampanjene som lages i forbindelse med applikasjonen

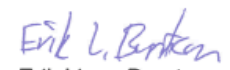
Vi opplevde også at studentene leverte på en svært god, ryddig og selvstendig måte. Dette i tillegg til at studentene holdt oss godt orientert med god kommunikasjon og at de tok initiativet underveis i prosjektet. Vi er trygge på at studentene kunne ledet fremdriften i et nytt prosjekt på en forbilliglig måte. De tok også hensyn til når de tok kontakt med oss i Phonero og vi opplevde en lavere belastning enn vi kunne ha forventet av en studentgruppe. De satte av tid til møter med oss kun når det var nødvendig og de hadde tydelige agendaer på møtene.

Vi opplevde som nevnt at studentene jobbet meget selvstendig og de var veldig engasjerte med et sterkt ønske om at sluttproduktet skulle bli bra. I tillegg var de ikke redde for å vise sårbarhet overfor hverandre med et brennende ønske om å lære så mye som mulig.

Vi har vært involvert i planleggingen, teknologivalg (hvor vi informert om hva vi bruker og kjenner). Vi har bidratt i integrasjoner når det trengs, eksempelvis laget vi et API mot vår database slik at studentene fikk ett endepunkt mot oss for å finne produkter, priser osv. Vi deltok selvsagt også på statusmøter med studentene og koordinator fra UiA. I dette prosjektet var vi mer en ren produkteier, noe som er en god øvelse for oss slik at vi får mer innsikt i hvordan det er å være produkteiere. Selvsagt bidro vi med teknisk bistand på teknologiene vi kjenner.

Vi ønsker å takke for et meget bra levert prosjekt! Nå tar vi finpussen før vi bringer det ut i produksjon! Lykke til videre med studier og/eller arbeidssøking.


Kristian Hagestad
Utvikler


Erik Linga Berntsen
Teamleder og utvikler


Bård Eik
Head of development

En bedre hverdag - for hele bedriften

Kilder

Adobe. (2023). *Single-page applications (SPAs) — what they are and how they work*.

<https://business.adobe.com/blog/basics/learn-the-benefits-of-single-page-apps-spa>

Axios. (u.å., hentet 15. mai 2024). *Getting Started - Promise based HTTP client for the browser and node.js - What is Axios?* <https://axios-http.com/docs/intro>

Benyon, D. (2019). *Designing User Experience: A Guide to HCI, UX and Interaction Design* (4. utgave.). Pearson Education Limited.

Drumond, C. (u.å.). *Agile Project Management for Software Teams*. Atlassian. Hentet 10.mai, 2024, fra <https://www.atlassian.com/agile/project-management>

Figma. (u.å.). *Free Design Tool for Websites, Product Design & More*. Figma. Hentet 25.mars, 2024, fra <https://www.figma.com/design/>

Figma. (u.å.). *Wireframe Vs. Mockup—What's The Difference?* Figma. Hentet 30.mars, 2024, fra <https://www.figma.com/resource-library/wireframe-vs-mockup/>

Galín, D. (2018). *Software Quality - Concepts and Practice* (1. utg.). IEEE Press.
<https://nibmehub.com/opac-service/pdf/read/Software%20quality%20%20concepts%20and%20practice.pdf>

Guide, S., & Deshpande, C. (2023, 5.oktober). *What Is React? [Easily Explained]*. Simplilearn.com. Hentet 10.mai, 2024, fra <https://www.simplilearn.com/tutorials/reactjs-tutorial/what-is-reactjs>

Gundersen, D., Halbo, L. (2018, 28. mai). kvalitet. I *Store norske leksikon*.
<https://snl.no/kvalitet#:~:text=Kvalitet%20er%20tings%20m%C3%A5te%20%C3%A5,tilfredsstille%20brukerens%20krav%20og%20forventninger>.

ISO/IEC. (2023). *ISO/IEC 25010:2023(en) Systems and Software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Product quality model*. iso.org. <https://www.iso.org/obp/ui/en/#iso:std:iso-iec:25010:ed-2:v1:en>

JGraph Ltd. (u.å.). *Features of draw.io*. draw.io. Hentet 15.mai, 2024, fra <https://www.drawio.com/features>

- Kanji, F. (2021). UX Research & Testing Techniques. Akendi.com.
<https://www.akendi.com/include/common/downloads/whitepapers/Akendi-UX-Research-Testing-Techniques.pdf>
- Karlsen, J. (2021). *Prosjektledelse* (5. utg.). Universitetsforlaget.
- Keycloak. (u.å., hentet 25. april 2024). *Keycloak*. keycloak.org. <https://www.keycloak.org>
- Khorikov, V. (2020). Unit Testing Principles, Practices, and Patterns. Manning Publications.
- Kubernetes. (u.å., hentet 30. april 2024). *Production-Grade Container Orchestration*.
kubernetes.io. <https://kubernetes.io>
- Learn Dapper. (2023, 1 august). *Welcome To Learn Dapper*. learndapper.com.
<https://www.learndapper.com/#:~:text=Dapper%20is%20an%20open-source,need%20to%20write%20tedious%20code.>
- Lucid Software Inc. (u.å.). Lucidchart: Intelligent Diagramming. Hentet 10.mai, 2024, fra
<http://www.lucidchart.com>
- Material Tailwind. (2023, 11.juni). *7 Reasons Why You Should Use Tailwind CSS Right Now*.
Material Tailwind. Hentet 10.mai, 2024, fra
<https://www.material-tailwind.com/blog/7-reasons-why-you-should-use-tailwind-css>
- Mathiassen, L., Munk-Madse, A., Nielsen, P. A., & Stage, J. (2018). *Object Oriented Analysis & Design* (2. utg.). Metodica ApS.
- Microsoft. (2022, 6. august). *Server Manager*. learn.microsoft.com.
<https://learn.microsoft.com/en-us/windows-server/administration/server-manager/server-manager>
- Microsoft. (u.å., hentet 30. april 2024). *TypeScript is JavaScript with syntax for types*.
typescriptlang.org. <https://www.typescriptlang.org>
- Microsoft. (u.å., hentet 25. april 2024). *What is ASP.NET?*. dotnet.microsoft.com.
<https://dotnet.microsoft.com/en-us/learn/aspnet/what-is-aspnet>
- Miranda, E. (2022). *Moscow Rules: A Quantitative Exposé*. Agile Processes in Software Engineering and Extreme Programming.
https://doi.org/10.1007/978-3-031-08169-9_2

Mikalsen, M. *Brukerhistorie* i Store norske leksikon på snl.no. Hentet 25. april 2024 fra

<https://snl.no/brukerhistorie>

Miro. (u.å.). *Hva er Wireframe?* Miro. Hentet 25.mars, 2024, fra

<https://miro.com/nb/wireframe/what-is-a-wireframe/>

Nair, G. J. (2013). Ensuring quality in the coding process: A key differentiator for the accurate interpretation of safety data. *Perspectives in Clinical Research*, 4(3).

<https://doi.org/10.4103/2229-3485.115383>

NAOB. (u.å., hentet 3. mai 2024). kvalitet. I *Det Norske Akademis Ordbok*.

<https://naob.no/ordbok/kvalitet>

Nearshore. (2022, 28.juli). *Scrumban methodology.... What is it? Who is it for?* Nearshore. Hentet 15.mai, 2024, fra

<https://www.nearshore-it.eu/articles/project-management-leadership/scrumban/>

Phonero. (u.å., hentet 15. mai 2024). Rekruttering. phonero.no.

<https://rekruttering.phonero.no/>

Rajak, S. (2023, 27.april). *5 reasons to use Tailwind*. LinkedIn. Hentet 10.mai, 2024, fra

<https://www.linkedin.com/pulse/5-reasons-use-tailwind-saurav-rajak/>

Ramírez, V. (2018, 4.august). *What is a Prototype?. A prototype is an early sample, model... | by Virginia Ramírez | NYC Design*. Medium. Hentet 3.mai, 2024, fra

<https://medium.com/nyc-design/what-is-a-prototype-924ff9400cfd>

React. (u.å.). React. Hentet 10.mai, 2024, fra <https://react.dev/>

Shadcn/ui. (u.å.). *Introduction*. shadcn/ui. Hentet 10.mai, 2024, fra <https://ui.shadcn.com/docs>

systemd. (u.å.). *System and Service Manager*. Hentet 15.mai, 2024, fra <https://systemd.io/>

Takada, M. (u.å.). *Single Page App Book*. Single Page App Book. Hentet 9.feb, 2024, fra

<https://singlepageappbook.com/single-page.html>

UXPin. (u.å.). UXPin | UI Design and Prototyping Tool. Hentet 01.mai, 2024, fra

<https://www.uxpin.com/>

Visual Studio. (u.å., hentet 25. april 2024). *Visual Studio Code*. code.visualstudio.com.

<https://code.visualstudio.com>

Wikimedia Foundation, Inc. (u.å., hentet 6. februar 2024). *Bitbucket*. wikipedia.org.

<https://en.wikipedia.org/wiki/Bitbucket>

Wikimedia Foundation, Inc. (u.å., hentet 30. april 2024). *C Sharp (programming language)*. wikipedia.org.

[https://en.wikipedia.org/wiki/C_Sharp_\(programming_language\)#:~:text=C%23%20\(%2F%CB%8Csi%CB%90%20%CB%88,C%23](https://en.wikipedia.org/wiki/C_Sharp_(programming_language)#:~:text=C%23%20(%2F%CB%8Csi%CB%90%20%CB%88,C%23)

Wikimedia Foundation, Inc. (u.å., hentet 15. mai 2024). *Daemon (computing)*. wikipedia.org.

[https://en.wikipedia.org/wiki/Daemon_\(computing\)](https://en.wikipedia.org/wiki/Daemon_(computing))

Wikimedia Foundation, Inc. (u.å., hentet 25. april 2024). *Dapper ORM*. wikipedia.org.

<https://en.wikipedia.org/wiki/Dapper ORM>

Wikimedia Foundation Inc. (u.å., hentet 15. mai 2024). *Docker (software)*. wikipedia.org.

[https://en.wikipedia.org/wiki/Docker_\(software\)](https://en.wikipedia.org/wiki/Docker_(software))

Wikimedia Foundation, Inc. (u.å., hentet 6. februar 2024). *Jira (software)*. wikipedia.org.

[https://en.wikipedia.org/wiki/Jira_\(software\)](https://en.wikipedia.org/wiki/Jira_(software))

Wikimedia Foundation, Inc. (u.å., hentet 11. april 2024). *Microsoft SQL Server*.

wikipedia.org. https://en.wikipedia.org/wiki/Microsoft_SQL_Server

Wikimedia Foundation, Inc. (u.å., hentet 30. april 2024). *MinIO*. wikipedia.org.

<https://en.wikipedia.org/wiki/MinIO>

Wikimedia Foundation, Inc. (u.å., hentet 25. april 2024). *npm*. wikipedia.org.

<https://en.wikipedia.org/wiki/Npm>

Wikimedia Foundation, Inc. (u.å., hentet 1. mai 2024). *NuGet*. wikipedia.org.

<https://en.wikipedia.org/wiki/NuGet>

Wikimedia Foundation, Inc. (u.å. hentet 10. mai 2024). *SQL*. wikipedia.org.

<https://en.wikipedia.org/wiki/SQL>

Wikimedia Foundation, Inc. (u.å., hentet 1. mai 2024). *Tailwind CSS*. wikipedia.org.

https://en.wikipedia.org/wiki/Tailwind_CSS

Wong, E. (2021, 2.januar). *Principle of Consistency and Standards in User Interface Design*.

The Interaction Design Foundation. Hentet 2.mai, 2024, fra

<https://www.interaction-design.org/literature/article/principle-of-consistency-and-standards-in-user-interface-design>

Appendiks

A1 - Risikovurdering

Nr .	Risiko	Sannsynlighet	Konsekvens	Innvirkning	Respons	Handling
1	Feil tidsestimering	4	4	16	Reduksjon	Revurdere tidsestimater
2	Lengre Fravær (sykdom)	1	5	5	Reduksjon	Re-fordele arbeidsoppgaver
3	Forsovelser	4	4	16	Reduksjon	Bot - 50kr - 100kr
4	Kort Fravær (sykdom)	2	3	10	Reduksjon	Re-fordele arbeidsoppgaver
5	Sløsing av tid	3	5	15	Reduksjon	Oppstramming i plenum
6	Tap av Arbeid (data)	1	5	5	Unngåelse	Gjøre oppgavene på nytt
7	Teknologi oppdatering som innfører kompatibilitetsproblemer	1	5	5	Aksept	Vurdere om oppdatering er hensiktsmessig / mulig
8	Dårlig kode/skikk	2	4	8	Reduksjon	Møte - bedre instruksjoner
9	Feil i Scope	2	5	10	Reduksjon	Revurdere
10	Feil bruk av Teknologi	1	3	3	Reduksjon	Lage regler for hvilke teknologier som skal brukes Gjøre om evt. feil
11	Dårlig kommunikasjon m/ Phonero	1	3	3	Reduksjon	Prøve å opprette bedre kommunikasjon
12	Dårlig kommunikasjon i gruppen	1	5	5	Reduksjon	Opprette bedre kommunikasjonsflyt
13	Dataangrep	2	4	8	Aksept	Lite vi kan gjøre med dette da Phonero står for sikkerhet
14	Dårlig Dokumentasjon	2	4	8	Reduksjon	Gjøre om dokumentasjon - stramme inn regler for dokumentasjon

15	Utvendige trusler	1	5	5	Aksept	Følge sikkerhetstiltak
16	Dårlig Enhetstesting	3	3	9	Reduksjon	Code review
17	Feature Creep	3	3	9	Reduksjon	Jevnlig oppfølging og kommunikasjon om limiteringer
18	Phonero leverer ikke API	1	5	5	Aksept	Bruk test data for databasen i stedet
19	Maskinfeil	1	4	4	Overføring	Phonero skaffer ny
20	Merging feil	2	4	8	Reduksjon	Code review og kommunikasjon
21	Bottle necking i arbeidsflyt	3	3	9	Reduksjon	Kommunisere når man forventer ferdig feature for videre arbeid mellom backend/frontend

Tabell A1: Risiko og Respons

A2 - Brukerhistorier

Brukerhistorie #2

#2	Prioritet: Must have
Brukerhistorie	For å kunne få bedre brukergrensesnitt på en kampanje, ønsker jeg som markedsføringsmanager å kunne legge til bilde, ikon, animasjon og banner på en kampanje.
Funksjon	Legge til grafiske elementer
Formål	Design kampanje
Test-scenario	1. Fyll ut steg 1 - Kampanje informasjon 2. Gå til steg 2 ved å trykke på “Neste” knappen for å legge til grafiske elementer. 3. Velg grafiske elementer
Argumentasjon	Denne funksjonen er “Must have”, for dette gir bedre brukergrensesnitt til hvordan en kampanje ser ut.

Tabell A2: Brukerhistorie 2

Brukerhistorie #3

#3	Prioritet: Must have
Brukerhistorie	Som en ansatte i markedsføringsavdeling ønsker jeg å redigere eksisterende kampanje for å gjenbruke den kampanjen.
Funksjon	Redigere kampanje
Formål	Endring av kampanje informasjon og gjenbruk av kampanje
Test-scenario	1. Gå til kampanje 2. Redigere kampanje informasjon 3. Legg til nye grafiske elementer ved behov 4. Se oppsummering av kampanjen 5. Klikk på “Fullført”
Argumentasjon	Denne funksjonen er prioritert som “Must have”, for det gir mulighet til de ansatte i markedsføringsavdeling å redigere kampanje informasjon og gjenbruke kampanjen neste gang.

Tabell A3: Brukerhistorie 3

Brukerhistorie #4

#4	Prioritet: Must have
Brukerhistorie	Som en produktsjef ønsker jeg å se oversikt over prestasjonen av hvilke kampanjer som oppnår best engasjement blant kundene knyttet til tilbud.
Funksjon	Statistikk av kampanjer
Formål	Oversikt over hvilke kampanjer som har truffet målgruppen(e) best
Test-scenarior	1. Gå til hjemmeside/Dashboard 2. Velg en kampanje eller to kampanjer for å se oversikt over prestasjonen av forskjellige kampanjer.
Argumentasjon	For å kunne se hvilke kampanjer som tiltrekker mest kundene for å ta en beslutning på å velge en kampanje som skal ta i bruk neste gang. Dette gir en mulighet til å forbedre kundeopplevelsen til bedriftens kundene. Derfor er denne funksjonen prioritert som "Must have".

Tabell A4: Brukerhistorie 4

Brukerhistorie #5

#5	Prioritet: Should have
Brukerhistorie	For å teste hvilken kampanje som er mest brukt av kundene, ønsker jeg som ansatte i markedsføringsavdelingen å opprette to kampanjer samtidig.
Funksjon	Lag en split-testing av kampanjer (Splitt kampanje).
Formål	Få en oversikt over hvilke kampanjer som er mest brukt av kundene.
Test-scenarior	Opprette splitt kampanje 1. Klikk på "Lag ny kampanje" 2. Velg "Splitt kampanje" (Steg 1) 3. Velg målgruppe 4. Fyll ut informasjon til kampanje A og kampanje B 5. Legg til grafikk til kampanje A og kampanje B 6. Se en oppsummering til to kampanjer 7. Klikk på "Fullført" for å fullføre to kampanjer.
Argumentasjon	Split-testing av kampanjer kan gi muligheten til markedsføringsavdeling for å se hvilke kampanjer som får mest engasjement fra kundene. Gjennom dette kan markedsføringsavdelingen samle inn data for å ta beslutningen på hvilken kampanje som skal tas i bruk for neste gang. Derfor er denne funksjonen prioritert som "Must have".

A3 - Wireframe v.1.0

A3.1 - Wireframe 1

Ny Kampanje

Mine Utkast

Legg til Grafikk

Hovedside

Pågående Kampanjer △

Bilde

Kampanjetittel

Startdato

Sluttdato

Detaljer

Endre

Bilde

Kampanjetittel

Startdato

Sluttdato

Detaljer

Endre

Bilde

Kampanjetittel

Startdato

Sluttdato

Detaljer

Endre

Kommende Kampanjer ▽

Utgåtte Kampanjer ▽

Hovedside

Ny Kampanje

Mine Utkast

Last Opp Grafikk

Velg fil...

Bilde

Gi grafikken ett navn.

Hvilken type grafikk er dette?

Bilde

Bilde

Ikon

Banner

Logo

Tekst

Avbryt

Godkjenn

Hovedside

Mine Utkast

Legg til Grafikk

Lag Ny Kampanje - Steg 1

Lag Ny Kampanje

Detaljer

Kampanjetype ▾

Priskutt

Tillegg

Produkt ▾

Produkt 1

Produkt 2

Målgruppe ▾

Kundetype 1

Kundetype 2

Pris ▾

Pris 1

Pris 2

Tillegg ▾

Mer Data

Annet

Startdato

Kalender

Sluttdato

Kalender

Avbryt

Videre

Hovedside

Endre Kampanje

Slett Kampanje

Legg til Grafikk

Lag Ny Kampanje - Steg 2

Lag Ny Kampanje

Grafiske Elementer

Velg Type ▾

Bilde

Bilde

Ikon

Banner

Logo

Tekst

Bilder

Bilde

Bilde

Bilde

Riktn

Riktn

Riktn

Valgte Elementer

Bilde 1

Bilde 2

Bilde 3

Logo 1

Ikon 1

Tilbake

Avbryt

Videre

53

Lag Ny Kampanje - Steg 3

Hovedside

Endre Kampanje

Slett Kampanje

Legg til Grafikk

Lag Ny Kampanje

Forhåndsvisning

Produkt: Navn

Kampanjetype: Type

Målgruppe: Gruppe X

Startdato: xx.yy.20zz

Sluttdato: xx.yy.20zz

Pris: xyz

Valgte Elementer

Bilde 1

Bilde

Bilde 2

Bilde

Bilde 3

Bilde

Logo 1

Bilde

Ikon 1

Bilde

Tilbake

Avbryt

Fullfør

Endre Eksisterende Kampanje - Steg 1

Hovedside

Mine Utkast

Legg til Grafikk

Endre Eksisterende Kampanje

Nåværende Detaljer

Produkt: Navn

Kampanjetype: Type

Målgruppe: Gruppe X

Startdato: xx.yy.20zz

Sluttdato: xx.yy.20zz

Pris: xyz

Endringer

Nytt Produkt

Produkt 1

Produkt 2

Ny Type

Priskutt

Tillegg

Ny Målgruppe

Kundetype 1

Kundetype 2

Ny Pris

Pris 1

Pris 2

Ny Tillegg

Mer Data

Annet

Ny Startdato

Kalender

Ny Startdato

Kalender

Avbryt

Videre

Endre Eksisterende Kampanje - Steg 2

Hovedside

Endre Kampanje

Slett Kampanje

Legg til Grafikk

Endre Eksisterende Kampanje

Grafiske Elementer

Velg Type

Bilde

Bilde

Ikon

Banner

Logo

Tekst

Bilder

Bilde

Bilde

Bilde

Bilde

Bilde

Valgte Elementer

Bilde 1

Bilde

Bilde 2

Bilde

Bilde 3

Bilde

Logo 1

Bilde

Ikon 1

Bilde

Tilbake

Avbryt

Videre

Endre Eksisterende Kampanje - Steg 3

Hovedside

Endre Kampanje

Slett Kampanje

Legg til Grafikk

Endre Eksisterende Kampanje

Forhåndsvisning

Produkt: Navn

Kampanjetype: Type

Målgruppe: Gruppe X

Startdato: xx.yy.20zz

Sluttdato: xx.yy.20zz

Pris: xyz

Valgte Elementer

Bilde 1

Bilde

Bilde 2

Bilde

Bilde 3

Bilde

Logo 1

Bilde

Ikon 1

Bilde

Tilbake

Avbryt

Fullfør

54

Mine Utkast

Hovedside

Ny Kampanje

Legg til Grafikk

Mine Utkast

Bilde

Kampanjetittel

Startdato

Sluttdato

Detaljer

Endre

Kampanjetittel

Detaljer

Endre

Bilde

Startdato

Sluttdato

Detaljer

Endre

Kampanjedetaljer

Kampanjetittel

Status

Hovedside

Ny Kampanje

Mine Utkast

Legg til Grafikk

Produkt: Navn

Kampanjetype: Type

Målgruppe: Gruppe X

Startdato: xx.yy.20zz

Sluttdato: xx.yy.20zz

Pris: xyz

Grafiske Elementer

Bilde 1

Bilde

Bilde 2

Bilde

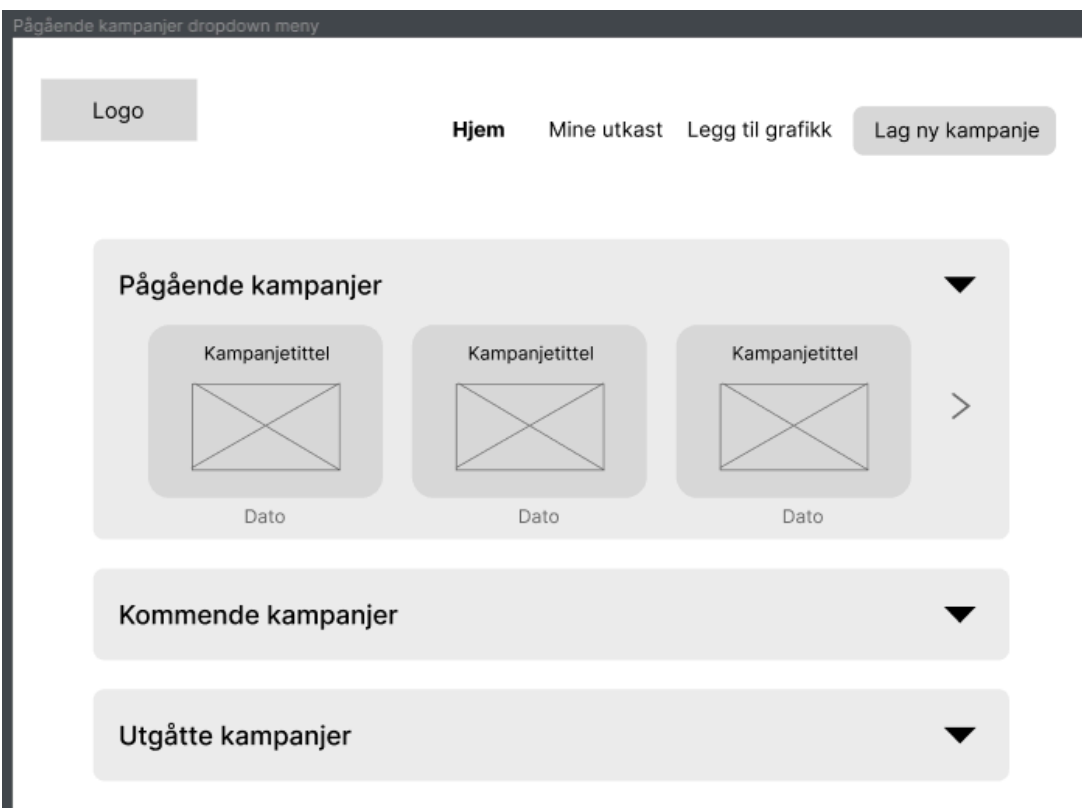
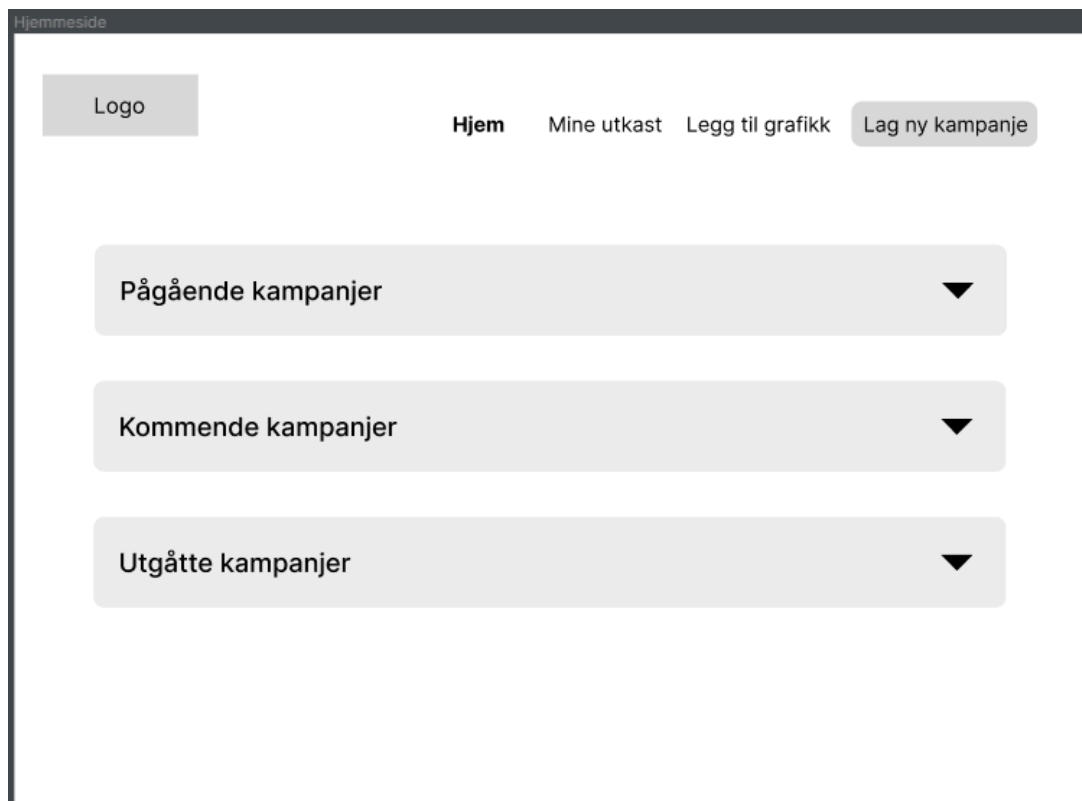
Bilde 3

Bilde

Tilbake

Gjør Endringer

A3.2 - Wireframe 2



Ny Kampanje

LogoHjemMine utkastLegg til grafikkLag ny kampanje

Ny Kampanje

Kampanje navn

Kampanje beskrivelse

Type

Målgruppe

Produkt

Pris

Kampanjestart *

Kampanjestart *

Lastopp fil/bildet

Avbryt

Lagre

Kampanjedetaljert

LogoHjemMine utkastLegg til grafikkLag ny kampanje

Kampanje navn

Redigere

Teskt

Teskt

Teskt

Logo

Hjem

Mine utkast

Legg til grafikk

Lag ny kampanje

Redigere Kampanje

Kampanje navn

Kampanje beskrivelse

Type

Målgruppe

Produkt

Pris

Kampanjestart *

Kampanjestart *

Lastopp fil/bildet

Avbryt

Lagre

Logo

Hjem

Mine utkast

Legg til grafikk

Lag ny kampanje

Mine utkast

Kampanje 1

Redigere

Slett

Kampanje 2

Redigere

Slett

Kampanje 3

Redigere

Slett

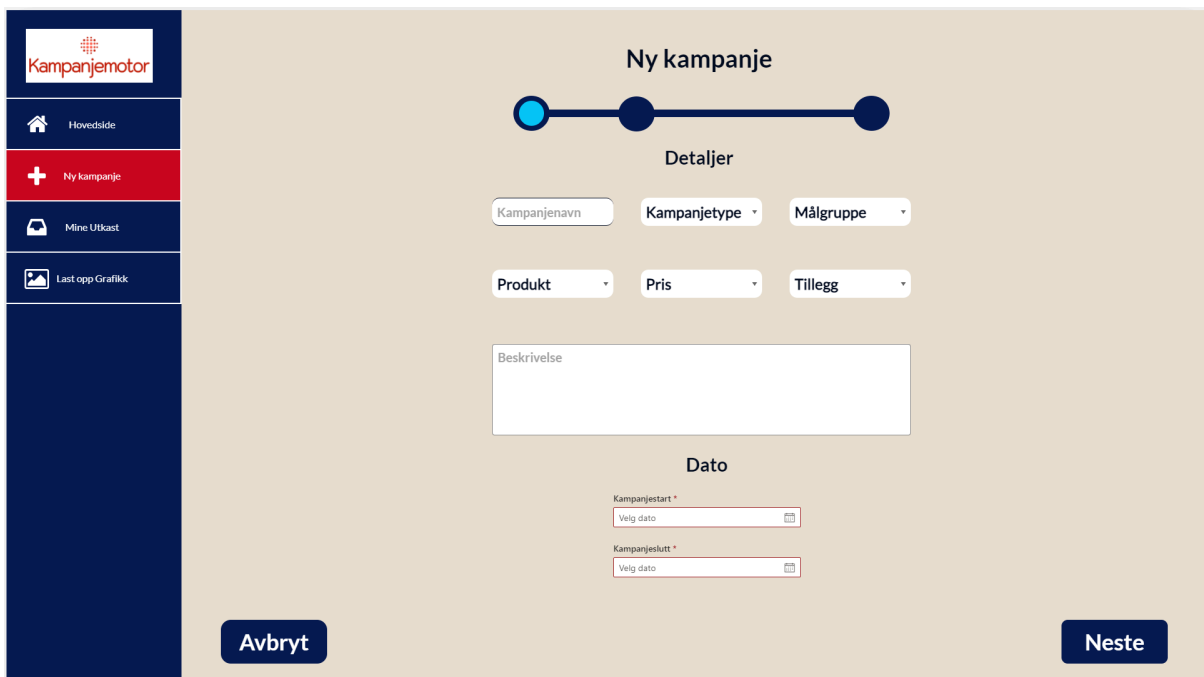
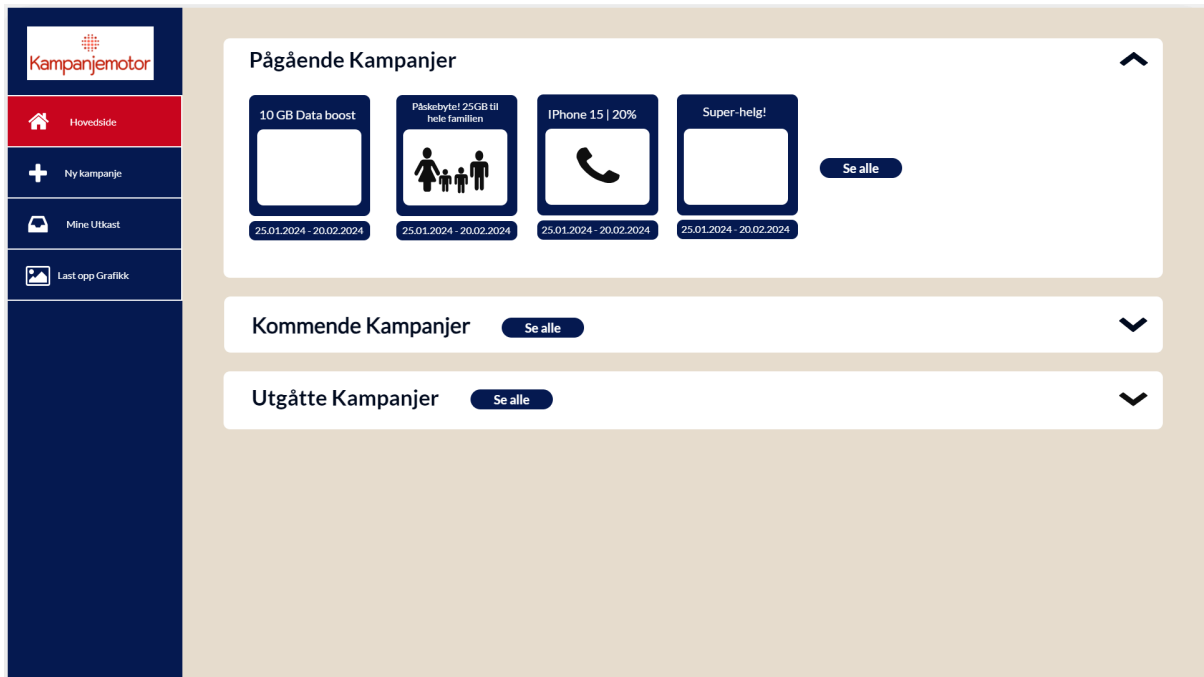
Kampanje 4

Redigere

Slett

A4 - Mockups

A4.1 - Mockup 1 - v1.0



Kampanjemotor

Hovedside
Ny kampanje
Mine Utkast
Last opp Grafikk

Ny kampanje

Banner

Tekst

Valgte elementer

Tilbake

Avbryt

Neste

Kampanjemotor

Hovedside
Ny kampanje
Mine Utkast
Last opp Grafikk

Ny kampanje

Kampanjenavn

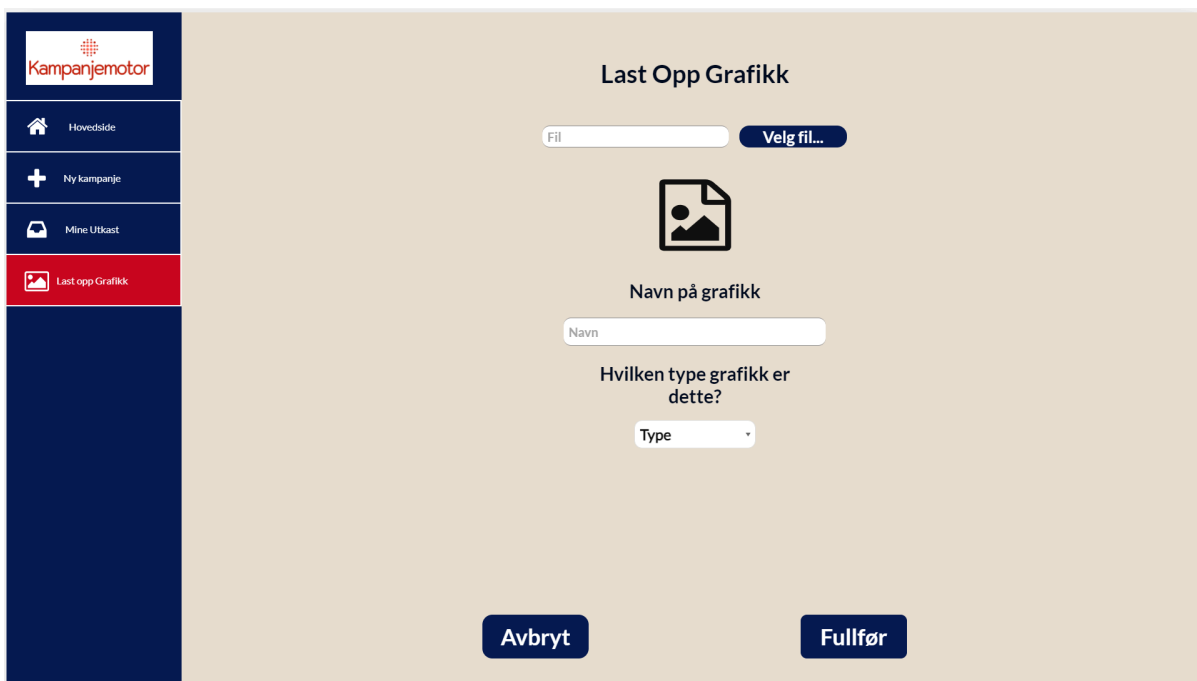
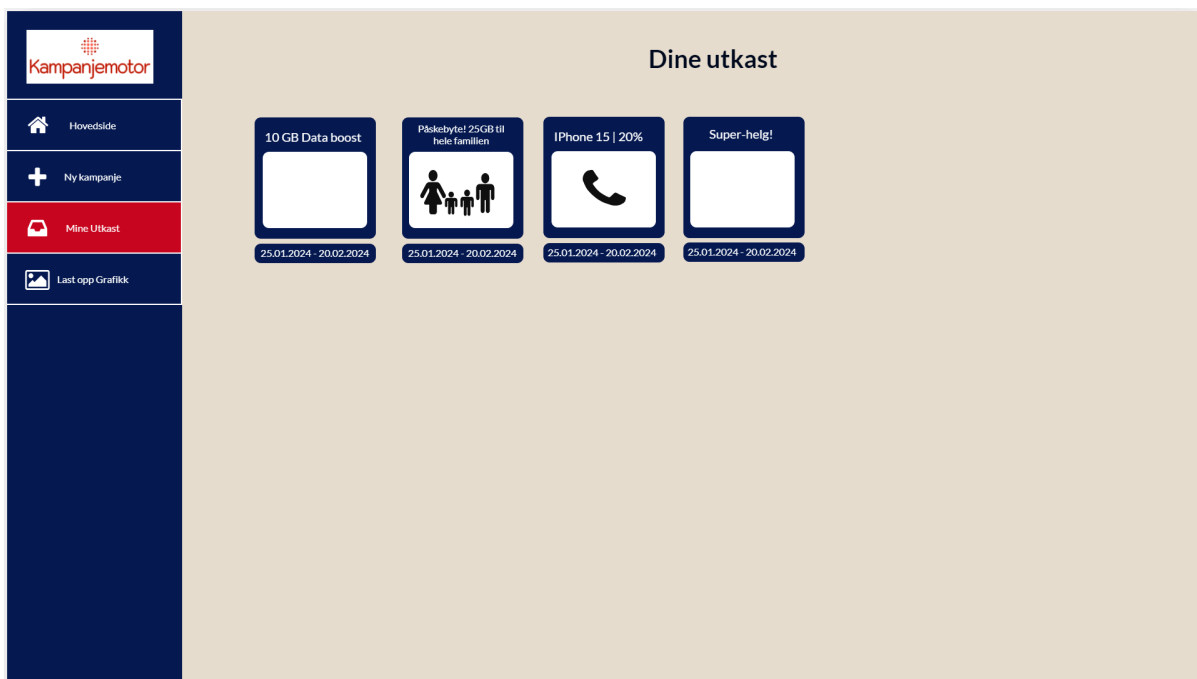
Produkt: Produktnavn
Kampanjetype: Type
Målgruppe: Kundesegment
Pris: xyz NOK
Tillegg: x GB

Beskrivelse
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.

Tilbake

Avbryt

Lagre



A4.2 - Mockup 2 - v1.0

[Hjem](#)[Mine utkast](#)[Grafikk](#)[+ Lag ny kampanje](#)

Pågående kampanjer ▼

Kommende kampanjer ▼

Utgåtte kampanjer ▼

[Hjem](#)[Mine utkast](#)[Grafikk](#)[+ Lag ny kampanje](#)

Mine utkast

5 timer fri data

[Rediger](#) [Slett](#) 

3 timer fri data

[Rediger](#) [Slett](#) 

4 timer fri data

[Rediger](#) [Slett](#) 

50 GB databonus

[Rediger](#) [Slett](#) 

< Forrige 1 2 3 ... Neste >

Ny Kampanje

Kampanjenavn

Kampanje beskrivelse

Kampanjetype

Produkt

Pris

Tillegg

Målgruppe

Kampanjestart *

Kampanjeslutt *

Last opp bilder

Avbryt

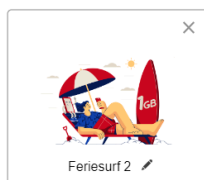
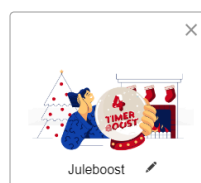
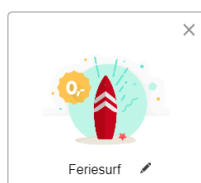
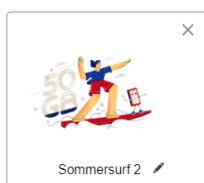
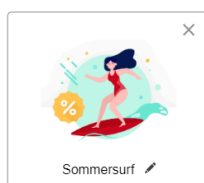
Lagre

Grafikk

Bilder

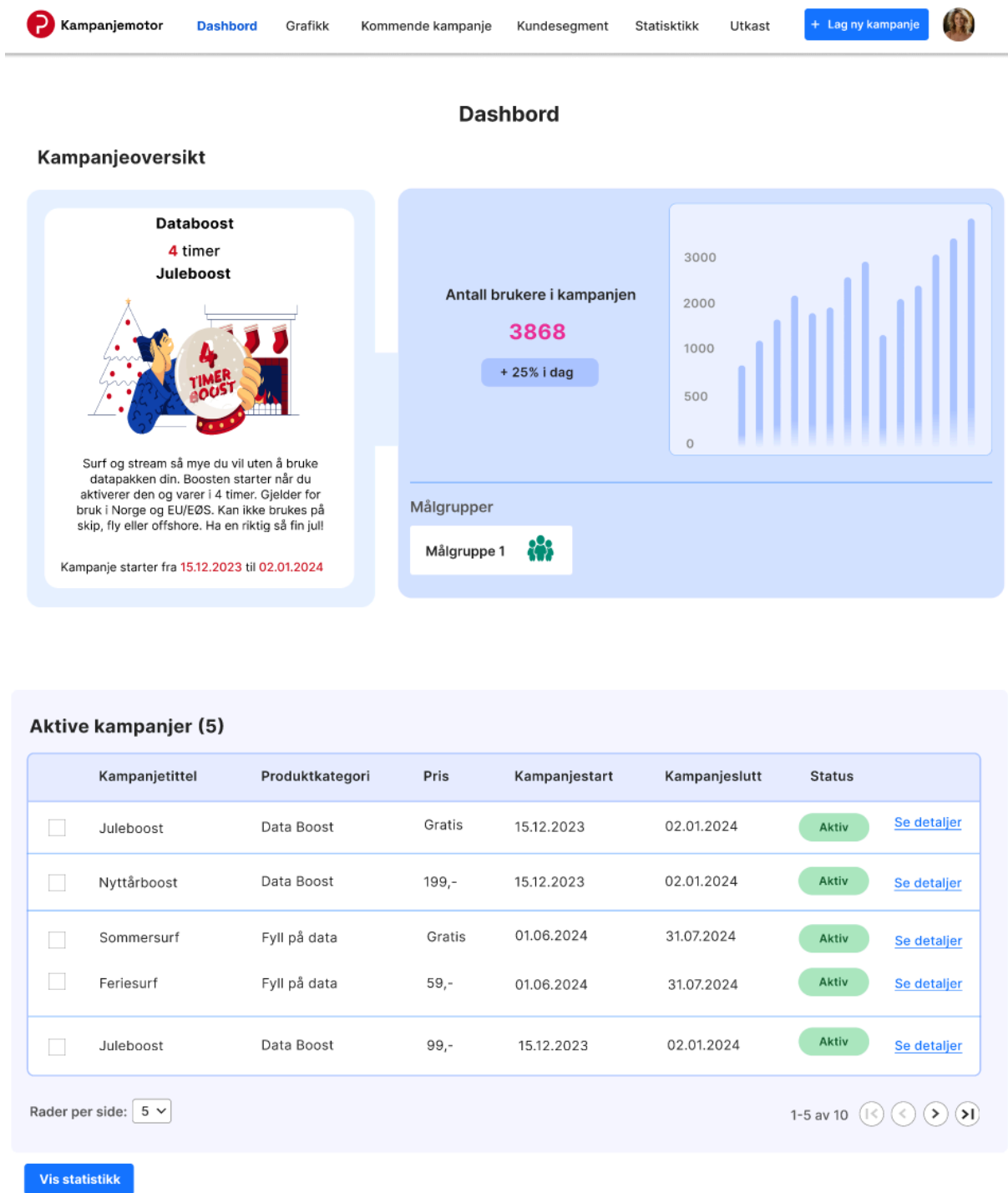
Banner

Ikoner



+ Legg til bilder

A4.3 - Mockup 3 - v2.0

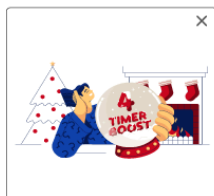




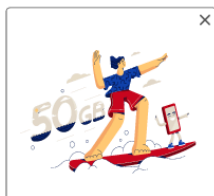
Grafikk



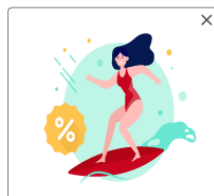
Alle grafiske elementer ▾



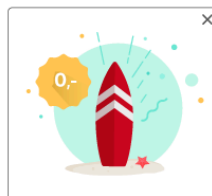
Juleboost



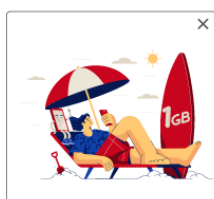
Sommersurf



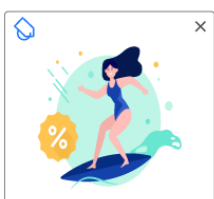
Sommersurf-2



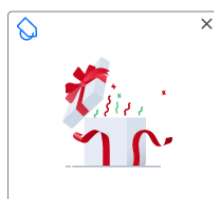
Feriesurf



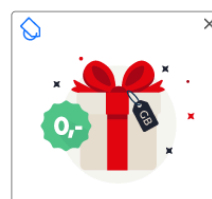
Feriesurf-2



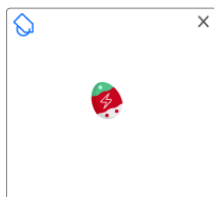
Sommersurf-blå-jente



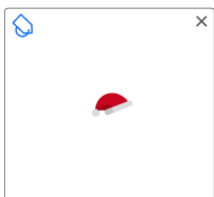
gratis-gave



gratis-gave-2



påske-egg



nisse-jule

< Forrige 1 2 3 ... Neste >

[↑ Last opp grafikk](#)

Kommende kampanjer

Forespørsler (2)

	Kampanjetittel	Produktkategori	Pris	Kampanjestart	Kampanjeslutt	Status	
☐	Juleboost	Data Boost	Gratis	15.12.2023	02.01.2024	Pending	Se detaljer
☐	Nyttårboost	Data Boost	199,-	15.12.2023	02.01.2024	Pending	Se detaljer

Rader per side: 5

1-5 av 10

Kommende kampanjer (5)

	Kampanjetittel	Produktkategori	Pris	Kampanjestart	Kampanjeslutt	Status	
☐	Juleboost	Data Boost	Gratis	15.12.2023	02.01.2024	Upcoming	Se detaljer
☐	Nyttårboost	Data Boost	199,-	15.12.2023	02.01.2024	Upcoming	Se detaljer
☐	Sommersurf	Fyll på data	Gratis	01.06.2024	31.07.2024	Upcoming	Se detaljer
☐	Feriesurf	Fyll på data	59,-	01.06.2024	31.07.2024	Upcoming	Se detaljer
☐	Juleboost	Data Boost	99,-	15.12.2023	02.01.2024	Upcoming	Se detaljer

Rader per side: 5

1-5 av 10



Statistikk

Kampanjeoversikt

Databoost

4 timer

Juleboost



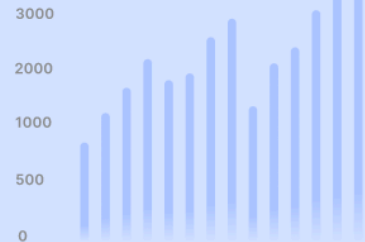
Surf og stream så mye du vil uten å bruke datapakken din. Boosten starter når du aktiverer den og varer i 4 timer. Gjelder for bruk i Norge og EU/EØS. Kan ikke brukes på skip, fly eller offshore. Ha en riktig så fin jull!

Kampanje starter fra 15.12.2023 til 02.01.2024

Antall brukere i kampanjen

3868

+ 25% i dag



Målgrupper

Målgruppe 1



Aktive- inaktive kampanjer (5)

Søk etter navn



	Kampanjetittel	Produktkategori	Pris	Kampanjestart	Kampanjeslutt	Status	
☒	Juleboost	Data Boost	Gratis	15.12.2023	02.01.2024	Aktiv	Se detaljer
☐	Nyttårboost	Data Boost	199,-	15.12.2023	02.01.2024	Aktiv	Se detaljer
☐	Sommersurf	Fyll på data	Gratis	01.06.2024	31.07.2024	Aktiv	Se detaljer
☐	Feriesurf	Fyll på data	59,-	01.06.2024	31.07.2024	Aktiv	Se detaljer
☐	Juleboost	Data Boost	99,-	15.12.2023	02.01.2024	Utgått	Se detaljer

Rader per side: 5

1-5 av 10

[Vis statistikk](#)



Utkast

	Kampanjetittel	Produktkategori	Pris	Kampanjestart	Kampanjeslutt	Status	
☐	Juleboost	Data Boost	Gratis	15.12.2023	02.01.2024	Utkast	
☐	Nyttårboost	Data Boost	199,-	15.12.2023	02.01.2024	Utkast	
☐	Sommersurf	Fyll på data	Gratis	01.06.2024	31.07.2024	Utkast	
☐	Feriesurf	Fyll på data	59,-	01.06.2024	31.07.2024	Utkast	
☐	Juleboost	Data Boost	99,-	15.12.2023	02.01.2024	Utkast	

Rader per side:

1-5 av 10





Ny Kampanje



Hvilken kampanjetype vil du lage?



Enkel kampanje

Lage en kampanje som gjelder flere målgrupper.



Splitt kampanje

Lage to kampanjer som gjelder spesifikke målgrupper.

Kampanje informasjon

Kampanjenavn *

Skriv inn kampanjenavn. Dette er kun for internt bruk.

Kampanjetittel

Skriv inn kampanjetittel

Ingress

Skriv inn en Ingress

Bredtekst

Skriv inn en bredtekst

Produktkategori

Velg produktkategori

Produkt

Velg produkt

Målgruppe

Velg målgruppe

Tilbudstekst

Skriv inn tilbudstekst

Tilbudspris

kr -

Prisinfo

Skriv inn tilbudstekst

Kampanjestart

Velg dato



Kampanjeslutt

Velg dato



Notis

Skriv inn notis

Avbryt

Neste



Ny Kampanje



Kampanje Informasjon



Legg til grafikk



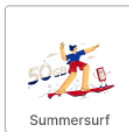
Oppsummering

Søk etter navn eller tags

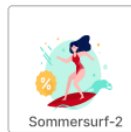
Alle grafiske elementer



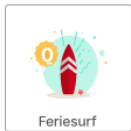
Juleboost



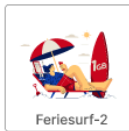
Summersurf



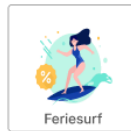
Sommersurf-2



Feriesurf



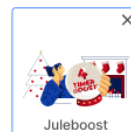
Feriesurf-2



Feriesurf

[↑ Last opp grafikk](#)

Valgte grafiske elementer



Juleboost

[Tilbake](#)[Neste](#)



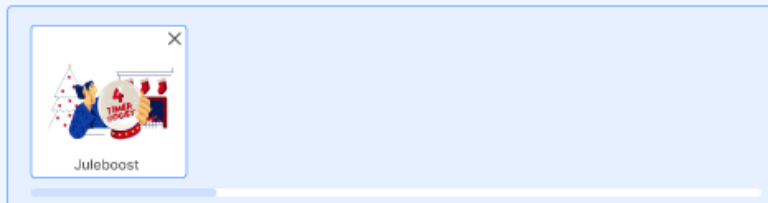
Ny Kampanje



Kampanje informasjon

Kampanjenavn	Test
Kampanjetittel	JuleBoost
Ingress	Surf og stream så mye du vil uten å bruke datapakken din. Boosten starter når du aktiverer den og varer i 4 timer. Gjelder for bruk i Norge og EU/EØS. Kan ikke brukes på skip, fly eller offshore. Ha en riktig så fin jull!
Brødtekst	Kan brukes i følgende land: Asorene, Belgia, Bulgaria, Danmark, England, Estland, Finland, Frankrike, Fransk, Guiana, Franske Antiller...
Produktkategori	Data Boost
Produkt	4 timer
Tilbudstekst	fri data
Målgruppe	Målgruppe 1
Tilbudspris	0,-
PrisInfo	Kan kun benyttes en gang
Kampanjestart	15.02.2023
Kampanjeslutt	02.01.2023
Notis	NB! Starter fra aktivering og kan kun brukes en gang.

Valgte grafiske elementer

[Lagre som utkast](#)[Tilbake](#)[Bekreft](#)



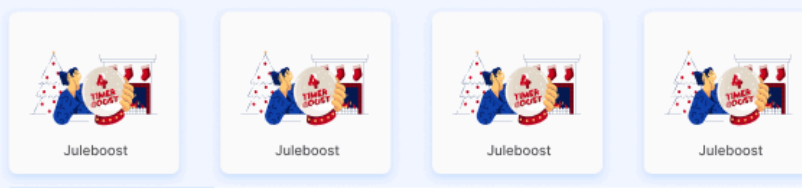
Kampanjedetaljer

Kampanjen er for målgruppe 1

Aktiv

Kampanjetittel	JuleBoost
Ingress	Surf og stream så mye du vil uten å bruke datapakken din. Boosten starter når du aktiverer den og varer i 4 timer. Gjelder for bruk i Norge og EU/EØS. Kan ikke brukes på skip, fly eller offshore. Ha en riktig så fin jull!
Bredtekst	Kan brukes i følgende land: Asorene, Belgia, Bulgaria, Danmark, England, Estland, Finland, Frankrike, Fransk, Guiana, Franske Antiller...
Notis	NBI Starter fra aktivering og kan kun brukes en gang.
Produktkategori	Data Boost
Produkt	4 timer
Tilbudstekst	fri data
Tilbudspris	0,-
Målgruppe	Målgruppe 1
Kampanjestart	15.02.2023
Kampanjeslutt	02.01.2023

Grafikk



Link for testmiljø

<https://kampanjemotor/123456>

Kopi

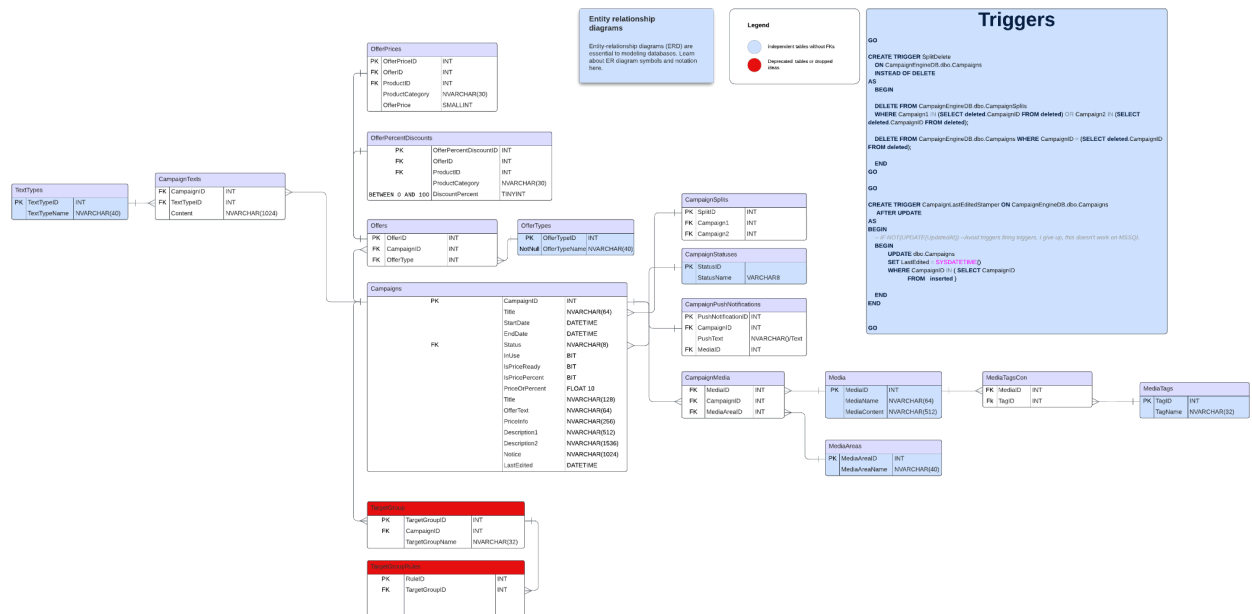
Sist oppdatering 03.april 2024

Avbryt

Rediger

A5 - Database Diagram

Database ER diagram For CE





A6 - Brukertest Mal

Introduksjon

Forklar til den som skal teste at:

1. De vil få en rekke oppgaver de skal løse, men bare én av gangen.
2. Det vil ikke bli forklart hvordan oppgaven skal gjennomføres, testeren må finne ut det selv ved å prøve seg frem.
3. Oppgaven er ferdig når testeren enten gir opp, eller selv mener oppgaven er fullført.
4. Mens testeren gjennomfører en oppgave, ønskes det at de verbalt forklarer sin tankegang og hva de holder på med.
5. Når en oppgave er gjennomført skal testeren besvare noen spørsmål relatert til oppgaven.

Test 1

#1	Prioritet: Must have
Brukerhistorie	For å kunne distribuere kampanje til kunder på Ditt Phonero, ønsker jeg som markedsførings manager å kunne lage en ny kampanje.
Funksjon	Opprette ny kampanje
Kriterier	Få en oversikt over en kampanje
Test-scenario	Opprette en ny kampanje: 1. Gå inn “Lag ny kampanje” 2. Velg “Enkel kampanje” (Steg 1) 3. Fyll ut alle feltene til kampanje 4. Legg til grafikk 5. Se en oppsummering til kampanje 6. Klikk på “fullført”
Argumentasjon	Denne funksjonen er viktig fordi markedsføringsavdelingen gjennomfører en rekke kampanjer for distribusjon til kundene. Dette bidrar til å forbedre kundeopplevelsen knyttet til bedriftens produkter.

I denne oppgaven skal brukeren lage en ny kampanje som et utkast. Start på Dashboard og be brukeren å lage en ny enkel kampanje uten bilder, mens de beskriver hva de gjør og tenker underveis. Noter brukerens kommentarer, problemer som oppstår underveis, bugs om de dukker opp, og ting brukeren ser ut til å synes er vanskelig.

Notater fra observasjon av brukeren som gjennomførte oppgaven.

Spørsmål 1: Hvordan syns du det gikk?

Spørsmål 2: Var det noe du syns var vanskelig med å gjennomføre oppgaven?

Spørsmål 3: Er det noen ting du ikke har nevnt enda som du syns burde endres på?

Spørsmål 4: Var det noen ting du syns manglet i brukergrensesnittet? Noe som burde legges til?

Spørsmål 5: På en skala fra 1 til 5, hvor 1 er dårlig og 5 er bra, hvordan vil du rangere funksjonen du testet i denne oppgaven?

Test 2

#3	Prioritet: Must have
Brakerhistorie	Som en ansatte i markedsføringsavdeling ønsker jeg å redigere eksisterende kampanje for å gjenbruke den kampanjen.
Funksjon	Redigere kampanje
Kriterier	Endring av kampanje informasjon og gjenbruk av kampanje
Test-scenario	1. Gå til valgt kampanje 2. Redigere kampanje informasjon 3. Legg til nye grafiske elementer ved behov 4. Se oppsummering av kampanjen 5. Klikk på "Fullført"
Argumentasjon	Denne funksjonen er prioritert som "Must have", for det gir mulighet til de ansatte i markedsføringsavdeling å redigere kampanje informasjon og gjenbruke kampanjen neste gang.

I denne oppgaven skal brukeren endre den første kampanjen de laget. Start på Dashboard og be brukeren å gjøre endringer på den første kampanjen de laget, gjerne alle punkter på kampanjen (navn, bilder, dato, etc.). Be om at de beskriver hva de gjør og tenker underveis. Noter brukerens kommentarer, problemer som oppstår underveis, bugs om de dukker opp, og ting brukeren ser ut til å synes er vanskelig.

Notater fra observasjon av brukeren som gjennomførte oppgaven.

Spørsmål 1: Hvordan syns du det gikk?

Spørsmål 2: Var det noe du syns var vanskelig med å gjennomføre oppgaven?

Spørsmål 3: Er det noen ting du ikke har nevnt enda som du syns burde endres på?

Spørsmål 4: Var det noen ting du syns manglet i brukergrensesnittet? Noe som burde legges til?

Spørsmål 5: På en skala fra 1 til 5, hvor 1 er dårlig og 5 er bra, hvordan vil du rangere funksjonen du testet i denne oppgaven?

Systembrukbarhetsskala

1. Jeg opplevde systemet som unødvendig komplisert.

Helt enig ----- Helt uenig

5 4 3 2 1

--	--	--	--	--

2. Jeg synes systemet var enkelt å bruke.

Helt enig ----- Helt uenig

5 4 3 2 1

--	--	--	--	--

3. Jeg synes at de forskjellige funksjonene i systemet er godt integrert.

Helt enig ----- Helt uenig

5 4 3 2 1

--	--	--	--	--

4. Jeg synes det var dårlig sammenheng i systemet.

Helt enig ----- Helt uenig

5 4 3 2 1

--	--	--	--	--

5. Jeg synes systemet var tungvint å bruke.

Helt enig ----- Helt uenig

5 4 3 2 1

--	--	--	--	--

6. Jeg følte meg selvsikker når jeg brukte systemet.

Helt enig ----- Helt uenig

5 4 3 2 1

--	--	--	--	--

A7 - Brukertest Resultater

A7.1 - Brukertest #1 (Representant fra Markedsføringsavdelingen)

Test 1

Notater fra observasjon av brukeren som gjennomførte oppgaven:

- Finner lett frem til siden som kan opprette en ny kampanje.
- Brukeren forstår hvilke tekster som er hva, men brukeren er litt redd for å gjøre feil.
- Brukeren ønsker å få mer informasjon/forklaringer på hvilke tekst som er til kunder eller for internt bruk.
- Brukeren ønsker at kalender forsvinner når en dato er valgt, samt automatisk rulle opp når det går videre til neste steg og mulighet til å trykke på steg i 'stepper' navigasjon.

Spørsmål 1: Hvordan syns du det gikk?

"Jeg synes det gikk veldig greit, følte jeg kom inn og klarte å lage det jeg skulle. Men sitter igjen med spørsmål rundt hvordan resultatet vil se ut. Jeg var litt uklar om noen ting var for internt bruk eller for kunder."

Spørsmål 2: Var det noe du syns var vanskelig med å gjennomføre oppgaven?

"Nei, jeg synes det var veldig enkelt å opprette en ny kampanje, ingen tvil om hvilke knapper jeg skulle trykke på. Fint strukturert og oppsett."

Spørsmål 3: Er det noen ting du tenker burde endres på?

Mer informasjon/forklaringer om status og hvilken tekst som er for kunder eller for internt bruk. Mulighet til å trykke på 'stepper' navigasjon og fix rullet opp for neste steg.

Spørsmål 4: Var det noen ting du syns manglet i brukergrensesnittet? Noe som burde legges til?

"Visuell representasjon om hva som er hva i forhold til applikasjonen. Mangler en 'Preview' funksjon til en kampanje. Litt uklart om hvordan kampanjen blir lagret eller publisert".

Spørsmål 5: På en skala fra 1 til 5, hvor 1 er dårlig og 5 er bra, hvordan vil du rangere funksjonen du testet i denne oppgaven?

"Jeg ga 5. Alt fungerte som det skulle, men unntak av de små tingene som jeg nevnte tidligere."

Test 2

Notater fra observasjon av brukeren som gjennomførte oppgaven:

- Brukeren kan merke at kampanjen ligger i utkast.
- Brukeren er litt usikker på om utkast er det beste.
- Brukeren er usikker på hvilken status må velge
- Brukeren vet ikke hvordan kampanjen blir publisert etter å ha endret den.

Spørsmål 1: Hvordan syns du det gikk?

“Dette var hakket vanskeligere enn å lage en ny kampanje, siden jeg slet litt med å finne rett kampanje. Jeg synes at det burde vært en litt tydeligere knapp. Mer informasjon eller forklaringer om hva hver status gjør. Jeg var litt usikker på hvilke valg jeg skulle ta for å komme fram.”

Spørsmål 2: Var det noe du syns var vanskelig med å gjennomføre oppgaven?

“Ja, fordi jeg ikke fant den knappen som jeg leter etter.”

Spørsmål 3: Er det noen ting du ikke har nevnt enda som du syns burde endres på?

“Litt i tvil med alle de forskjellige statusene. Er de nødvendige? Kanskje mer informasjon om det.”

Spørsmål 4: Var det noen ting du syns manglet i brukergrensesnittet? Noe som burde legges til?

Bare synlighet når det kommer til knappen

Spørsmål 5: På en skala fra 1 til 5, hvor 1 er dårlig og 5 er bra, hvordan vil du rangere funksjonen du testet i denne oppgaven?

“Denne gangen ga jeg ikke full score. Bare 4. Det fungerer jo som det skal, men det er noen som mangler som jeg har nevnt tidligere.”

TIL SIST

Systembrukbarhetsskala

1. Jeg opplevde systemet som unødvendig komplisert.

Helt enig ----- Helt uenig

5 4 3 2 1

				X
--	--	--	--	---

2. Jeg synes systemet var enkelt å bruke.

Helt enig ----- Helt uenig

5 4 3 2 1

X				
---	--	--	--	--

3. Jeg synes at de forskjellige funksjonene i systemet er godt integrert.

Helt enig ----- Helt uenig

5 4 3 2 1

	X			
--	---	--	--	--

4. Jeg synes det var dårlig sammenheng i systemet.

Helt enig ----- Helt uenig

5 4 3 2 1

			X	
--	--	--	---	--

5. Jeg synes systemet var tungvint å bruke.

Helt enig ----- Helt uenig

5 4 3 2 1

			X	
--	--	--	---	--

6. Jeg følte meg selvsikker når jeg brukte systemet.

Helt enig ----- Helt uenig

5 4 3 2 1

		X		
--	--	---	--	--

A7.2 - Brukertest #2 (Representant fra IT avdelingen)

TEST 1

Notater fra observasjon av brukeren som gjennomførte oppgaven:

- Brukeren kan finne fram knappen for å lage en ny kampanje.
- Brukeren er usikker på alle teksttypene, men forventet siden brukeren jobber i en annen avdeling.
- Forventer “stor kunde, etc.” under målgruppe.
- Kalenderboksen endrer størrelse og overrasker. Trykke enter i søkefeltet for bilder på ny kampanje manøvrerer til annen side.

Spørsmål 1: Hvordan syns du det gikk?

Bortsett fra å søke etter grafikk gikk det ganske greit. Også er jeg litt usikker på hva alle de forskjellige tingene faktisk betyr.

Spørsmål 2: Var det noe du syns var vanskelig med å gjennomføre oppgaven?

Syns rekkefølgene på tabulator var litt ulogisk.

Spørsmål 3: Er det noen ting du tenker burde endres på?

Buggen burde fikses. Datovelgerne var litt snodige, endret størrelse og hoppet opp og ned.

Spørsmål 4: Var det noen ting du syns manglet i brukergrensesnittet? Noe som burde legges til?

Hjelpeikoner med forklaring.

Spørsmål 5: På en skala fra 1 til 5, hvor 1 er dårlig og 5 er bra, hvordan vil du rangere funksjonen du testet i denne oppgaven?

4.

TEST 2

Notater fra observasjon av brukeren som gjennomførte oppgaven:

- Brukeren kan raskt finne frem til hvor han skal trykke for å redigere en kampanje.
- Bemerkning heter notis på endre kampanje som ikke gir en tydelig mening til brukeren.
- Brukeren likte lagre bekreftelsen.
- Bug for dato inputs på endre kampanje.

Spørsmål 1: Hvordan syns du det gikk?

“Bra. Forutsett at jeg gjorde riktig.”

Spørsmål 2: Var det noe du syns var vanskelig med å gjennomføre oppgaven?

“Nei.”

Spørsmål 3: Er det noen ting du ikke har nevnt enda som du syns burde endres på?

“Notis. Fiks kalender for at det ikke hopper opp eller ned. Ellers nei.”

Spørsmål 4: Var det noen ting du syns manglet i brukergrensesnittet? Noe som burde legges til?

“Nei.”

Spørsmål 5: På en skala fra 1 til 5, hvor 1 er dårlig og 5 er bra, hvordan vil du rangere funksjonen du testet i denne oppgaven?“5”.

TIL SIST

Systembrukbarhetsskala

1. Jeg opplevde systemet som unødvendig komplisert.

Helt enig ----- Helt uenig

5 4 3 2 1

				X
--	--	--	--	---

2. Jeg synes systemet var enkelt å bruke.

Helt enig ----- Helt uenig

5 4 3 2 1

	X			
--	---	--	--	--

3. Jeg synes at de forskjellige funksjonene i systemet er godt integrert.

Helt enig ----- Helt uenig

5 4 3 2 1

	X			
--	---	--	--	--

4. Jeg synes det var dårlig sammenheng i systemet.

Helt enig ----- Helt uenig

5 4 3 2 1

				X
--	--	--	--	---

5. Jeg synes systemet var tungvint å bruke.

Helt enig ----- Helt uenig

5 4 3 2 1

				X
--	--	--	--	---

6. Jeg følte meg selvsikker når jeg brukte systemet.

Helt enig ----- Helt uenig

5 4 3 2 1

	X			
--	---	--	--	--

A8 - Tidsestimering

Oppgave	Tidsestimat (Man Hours)
Rapportskriving	x timer
• Kapittel 1 • Kapittel 2 • Kapittel 3 • Kapittel 4 • Kapittel 5 • Kapittel 6 • Kapittel 7 • Kapittel 8	• x timer • x timer • x timer • x timer • x timer • x timer • x timer • x timer
Personas	2 timer
Brukerhistorier	2 timer
PACT-analyse	1 time
FACTOR- analyse	1 time
Wireframe 1 - v1.0	8 timer
Wireframe 2 - v1.0	8 timer
Wireframe 3 - v1.0	8 timer
Mockups	10 timer
Prototype 1 - v1.0	10 timer
Prototype 2 - v1.0	10 timer
Prototype 3 - v1.0	10 timer
Wireframe - v. 2.0	4 timer
Mockup - v. 2.0	35 timer
Prototype - v. 2.0	5 timer
Sette opp utviklermiljø	15 timer
Risikoanalyse	5 timer
Kartlegge systemkrav	15 timer
Lage Mappestruktur	1 time
Lage grenstruktur for versjonskontroll	1 time

ER Diagram	16 timer
Backend - Sette opp database	3 timer
Backend - Database seed	6 timer
Frontend - Lage Dashboard siden (no styling)	
Frontend - Lage Opprett Ny Kampanje siden (no styling)	
Frontend - Lage Utkast siden (no styling)	
Frontend - Lage Utgåtte Kampanjer/Historikk siden (no styling)	
Frontend - Lage Last opp Grafikk siden (no styling)	
Frontend - Lage Redigere Kampanjer siden (no styling)	
Frontend - Lage Slett Kampanjer siden (no styling)	
Frontend - Lage Kampanje Detaljer siden (no styling)	
Backend - CRUD	15 timer
Backend - Database connection	5 timer
Funksjon - Opprette ny enkel kampanje	24 timer
Funksjon - Opprette split kampanjer	12 timer
Backend - Data Sanitisation Tools	12 timer
Funksjon - Dashboard	15 timer
Funksjon - Vise Statistikk	20 timer
Funksjon - Utkast	30 timer
Funksjon - Kommende kampanjer	20 timer
Funksjon - Utgåtte kampanjer (historikk)	10 timer
Funksjon - Last opp Grafikk	18 timer
Funksjon - Rediger kampanjer	14 timer
Funksjon - Slett kampanjer	3 timer
Funksjon - Kampanje detaljer	2 time
Backend - External Data Access mot REST API	5 timer
Push Notifications	2 timer

Funksjon - Kundesegmentering	17 timer
Fronted - Styling (Dashboard)	20 timer
Fronted - Styling (Opprette ny kampanje)	20 timer
Fronted - Styling (Utkast)	15 timer
Fronted - Styling (Kommende kampanjer)	20 timer
Fronted - Styling (Utgåtte kampanjer/historikk)	20 timer
Fronted - Styling (Last opp Grafikk)	30 timer
Fronted - Styling (Redigere kampanjer)	15 timer

A9 - Fremgangsplaner (sprint)

Pre-sprint

Ferdig med Wireframes, Mockups, Forarbeid. Finne ut av regler for koding. Risikovurdering.

Sprint 1 (9.2):

Struktur: Ferdig med å sette opp alle mapper/filer til FrontEnd.

Frontend: Begynne å sette opp noen av sidene / hjemmeside, lag ny kampanje.

Backend: Ferdig med DB struktur og valg av backend teknologier.

Annet: Begynne på rapporten.

Sprint 2 (23.02):

Backend: Ferdig med DB. Lage database-seed. Ferdig med å lage ny kampanje - enkel (som blir lagret i databasen).

Frontend: Design på navbar, hjemmeside (enkel), lag ny kampanje. Mulighet for å legge til nye bilder i databasen.

Annet: Være ferdig med 1/4 av rapporten

Sprint 3 (09.03):

Backend: Ferdig med lage ny kampanje, inkludert bilde og grafikkopplasting.

Frontend: Ferdig med de fleste sider (lag ny kampanje, hjemmeside (dashboard), legg til grafikk, historikk)

Annet: Ferdig med 2/4 av rapporten

Sprint: 4 (22.03):

Backend:

Frontend:

Annet: Ferdig med 3/4 rapporten

Sprint 5 (08.04):

A10 - Ukesrapporter

Uke 2 - 08.01 til 12.01

Tirsdag 9.1 ble det holdt et møte med Phonero. Dette møtet fungerte som en introduksjon til miljøet i bedriften, og hvilke redskaper de bruker i dag, blant annet Bitbucket og Jira.

Torsdag 11.1 ble det holdt enda et møte. Her ble prosjektbeskrivelsen tatt opp, dette inkluderte hvilke funksjoner Phonero ønsker at skal være med. Det var kun medlemmer av IT-avdelingen som deltok her.

Denne uken fokuserte gruppen på å kartlegge prosjektets krav, bli kjent med bedriften, og starte prosessen med å lage Wireframes for produktet.

Kartlegging av prosjektets krav ble gjort ved å skrive et “design dokument” som inneholder en prosjektbeskrivelse, hvilke funksjoner produktet skal ha, og en beskrivelse av utviklingsmiljøet; hvilke teknologier som skal brukes.

Dokumentet inneholder også personas lagd av gruppen. Disse skal bli brukt til å skrive brukerhistorier, og for å gi gruppen perspektiv til å kunne forstå hvordan brukere vil ta i bruk produktet.

Uke 3 - 15.01 til 19.01

Denne uken fokuserte gruppen på å komme i gang med prototypene. Disse ble basert på wireframes gruppen hadde laget forrige uke. Samtidig som prototypene ble utviklet ble også maler for farger laget, slik at gruppen kunne lage flere forslag til hvilke farger webapplikasjonen skulle bruke.

I tillegg har gruppen lagt seg inn i rammeverket React i sammenheng med å sette opp utviklingsmiljøet.

Utviklingen på prototypene kom langt og en av de ble nesten ferdigstilt, med ferdig valgte farger og placeholder funksjoner.

Uke 4 - 22.01 til 26.01

25.1 var det tre møter. Først et obligatorisk sikkerhetskurs, som omhandlet Phoneros sikkerhetsrutiner og hva som forventes av gruppen i henhold til dette. Så et møte med markedsføringsavdelingen om hva de ønsket rundt applikasjonens funksjonalitet. Erik og Kristian fra IT avdelingen deltok også på dette møtet. Det siste møtet var et opplæringsmøte med Phonero, med omvisning.

Denne uken fokuserte gruppen på å klargjøre prototypene til møte med markedsføringsavdelingen. På møtet ble ønskede funksjoner kartlagt, og prototyper ble vist frem.

Informasjonen samlet inn på møtet ble kollert og planer for videre utvikling ble lagt. Disse planene involverer å videreutvikle den valgte prototypen ved å gjøre endringer på dens

wireframe basert på informasjonen samlet inn på møtet. Disse endringene skal så overføres til selve prototypen.

Uke 4 - 5.02 til 9.02

Denne uken ble design ferdigutviklet og frontend teknologier utvalgt, blant annet Tailwind. Det ble gjort et generelt tidsestimat av alle oppgavene som er en del av prosjektet, i tillegg til at en risikovurdering ble gjennomført. På backend ble det jobbet videre på databasens design, og et møte med Erik ble holdt 8.02 for å diskutere både dette og ... Stor fremgang ble også gjort på Prosjektrapporten.

Uke 5 - 29.01 til 02.02

Mandag 29.1 hadde vi et møte med Erik fra IT-avdelingen angående Database. Hvordan Phonero strukturerer sin database i dag, og hva slags forventninger de har til hva gruppen produserer. Tirsdag 30.01 hadde vi et oppsummeringsmøte av pre-sprinten. Vi hadde også et planleggingsmøte der vi strukturerte en plan for milepæler som skal nås i prosjektet. Torsdag 01.02 hadde vi et møte der vi presenterte det endelige designet til brukerne, og diskuterte løsningene på funksjonaliteten til programmet. Fredag 02.02 hadde vi et møte der vi oppsummerte uka. (Hva gikk som planlagt - Hva som ikke gikk som planlagt).

Denne uken hadde gruppen fokus på å ferdigstille designet til webapplikasjonen, og danne grunnlaget slik at vi kunne begynne å programmere. Vi konstruerte filstrukturen, samt laget modeller for å bedre kartlegge utviklingen. Det ble også skrevet mer på bacheloroppgaven.

Uke 6 - 05.02 til 09.02

Vi startet mandagen med et standup-møte hvor vi la grunnlaget for ukens arbeid og satte opp våre mål og oppgaver.

Tirsdag var dedikert til et møte med Erik, hvor vi dykket ned i statistikk og databaseelementer som er avgjørende for prosjektets fremgang. Diskusjonene vi hadde bidro til å skape klarhet og retning i hvordan vi skal håndtere statistikken som skal visualiseres, samt feedback på vår database.

Mot slutten av uken, på torsdag, brukte vi tid på å lage en presentasjon som skulle oppsummere våre fremskritt og planer så langt. Denne skulle presenteres for veilederne.

Fredagen ble viet til å oppsummere sprinten og evaluere vår fremdrift. Vi identifiserte hva som hadde fungert bra og hva som kunne forbedres i kommende perioder.

Våre hovedaktiviteter i uken inkluderte å finalisere prototypen og wireframes basert på innsikt fra møtene, definere en klar oversikt over ønskede funksjoner, oppsett av vårt programmeringsmiljø, samt grundig risiko- og PACT-analyse. Vi strukturerte også en solid mappe- og filstruktur for prosjektet, utviklet en passende databasestruktur med tilhørende ER-diagram, og satt opp en fremdriftsplan for kommende arbeid.

Denne uken har satt et sterkt fundament for prosjektet vårt, og vil hjelpe oss med videre utvikling.

Uke 7 - 12.02 til 16.02

Denne uken har vært en intensiv periode med fokus på møter, praktisk arbeid og kompetanseoppbygging. Vi startet mandagen med et standup-møte, der vi la grunnlaget for ukens arbeid og satte klare mål for hva vi ønsket å oppnå. Deretter fortsatte dagen med et veiledningsmøte med UiA (Hallgeir, Kristian og Erik). Der presenterte vi det vi hadde gjort så langt i prosjektet, og mottok tilbakemelding.

I løpet av uken gjennomførte vi flere sentrale aktiviteter. Vi startet på å etablerte Git-Ignore for å sikre en effektiv versjonskontroll og filbehandling. Vi begynte også å designe Dashboard- og Enkel kampanje siden for å skape et funksjonelt grensesnitt. Videre implementerte vi stepper- funksjonen til Enkel Kampanjesiden. og fullførte NavBar for å gi en helhetlig opplevelse.

I tillegg startet vi arbeidet med å etablere databaseforbindelsen. Vi viet også tid til å forstå bakenden av systemet og planlegge implementasjonsstrategier for å sikre en robust og skalerbar løsning. Samtidig satte vi av tid til å lære Tailwind CSS, en verdifull ressurs for å optimalisere frontend-designet vårt og øke brukerens opplevelse.

Denne uken har vært preget av både handling og læring, med hver aktivitet designet for å styrke prosjektets fundament og øke vår tekniske dyktighet. Med en solid start er vi klare til å fortsette arbeidet neste uke, med fokus på videreutvikling av funksjonalitet, design og vår tekniske ekspertise.

Uke 8 - 19.02 til 23.02

Vi startet uken med et grundig standup-møte på mandag (19.2), hvor vi gikk gjennom statusen på våre pågående oppgaver og prioriterte de kommende aktivitetene.

Sentrale fokusområder for uken inkluderte arbeidet med vår database-forbindelse. Vi dedikerte tid til å etablere en solid forbindelse og gjennomføre finpuss på selve databasen.

Samtidig jobbet vi også med utvikling av modeller og spørringer (queries) for å effektivt kunne håndtere dataene våre. Denne innsatsen bidro til å legge grunnlaget for en smidig og effektiv datahåndteringsprosess.

I tillegg til prosjektarbeidet, viet vi også tid til å skrive på bacheloroppgaven. Vi fortsatte arbeidet med vår bacheloroppgave ved å skrive og utarbeide modeller til oppgaven.

Vi finaliserte også designet på Dashboard- og Enkel Kampanje sidene. Vi møtte på problemer når alle skulle pushe det de hadde arbeidet med. Dette var grunnet conflicts som oppstod fordi ingen hadde oppdatert branchen sin på en stund og branchene var utdaterte. Dette medførte at vi brukte store deler av fredagen til å sende de arbeidet til hverandre slik at en i gruppa med oppdatert branch kunne pushe. Deretter lagde vi en klar plan for hvordan vi videre skulle pushe slik at en lignende situasjon oppstår igjen. Da kom vi frem til at alle skal

lage en branch ut ifra issues, og der skal man kun jobbe på det issuen beskriver. Man skal også maks jobbe på en issue i 2 dager.

Uke 9 - 26.02 til 01.03

Denne uken har vært preget av flere viktige hendelser og arbeidsoppgaver. Mandagen startet med et møte for å gjennomgå og evaluere våre arbeidsmetoder og fremgang gjennom en retrospektiv analyse.

Mot slutten av uken, på fredag, ble det avholdt et styremøte hvor Hallgeir fra UiA deltok . Dette møtet var en mulighet til å diskutere strategier, vurdere fremdrift og ta avgjørelser om fremtidige retninger for prosjektet.

Sentrale arbeidsoppgaver denne uken inkluderte intensiv innsats på frontend-utviklingen av vår webapplikasjon. Vi fokuserte på å jobbe videre på frontend-delen for de fleste sidene i applikasjonen, og samtidig sikre at backend-systemet kunne sende data sømløst til frontend. En betydelig del av tiden ble også brukt på å forbedre koden gjennom refaktoring, slik at kommunikasjonen mellom frontend og backend ble i større grad optimalisert.

I tillegg til å implementere funksjonalitet og optimalisere ytelsen, var en annen viktig oppgave denne uken å identifisere og velge de mest hensiktsmessige komponentene som vi skulle bruke videre i vårt prosjekt. Alt i alt har denne uken vært travel og produktiv, med viktige møter og intensivt arbeid med å forbedre vår webapplikasjon.

Uke 10 - 04.03 til 08.03

Denne uken har vært preget av møter og utviklingsarbeid. Vi startet med et møte på tirsdag sammen med Erik, hvor vi diskuterte hvordan lagring av bilder og hvordan APIet skal utformes. Neste møte var på torsdag sammen med Erik og Håkon, der fikk vi et innblikk i MinIo, og hvordan vi kan utnytte dette verktøyet til å lagre media- filer.

Utviklingsmessig har vi lagt til flere komponenter som er essensielle for prosjektet vårt, og fokusert på validering av skjemaer og media opplasting. Det ble også brukt tid til å sette opp MinIo klient og server. Der fikk vi mer innsikt i hvordan håndteringen av filer og bilder vil foregå. Videre har vi jobbet med funksjonaliteter knyttet til opprettelse og redigering av kampanjer. I tillegg til dette er det laget et modal- skjelett som skal laste opp media- filene.

Kort oppsummert har uken vært preget av et høyt tempo og fremskritt rundt hvordan vi skal håndtere media- filene.

Vi avsluttet uken med et sprint retrospektiv på fredag, hvor vi kunne reflektere over våre prestasjoner, identifisere suksesser og områder for forbedring i kommende sprinter.

Uke 11 - 11.03 til 15.03

I denne uken har det vært fokus på videre arbeid med issues, design og skolearbeid.. Først og fremst har gruppen jobbet med å prøve å få til nedlasting av mediefiler, men dette ble ikke fullført denne uken. I tillegg jobbet gruppen med å implementere full CRUD-funksjonalitet

hovedsakelig på backenden i webapplikasjonen, bortsett fra Create og Update. Videre har vi fortsatt arbeidet med utviklingen av skjelettet til flere sider i webapplikasjonen. Dessuten har gruppen gjennomført noe re-design av sider, grunnet endringer i visnings innhold og funksjonalitet som ble nødvendig. Til slutt har det vært en del arbeid med skoleoppgaver da det var innleveringsfrister som nærmet seg.

Kort oppsummert har uken vært fylt med prøving, feiling og mye læring. Gruppen har fått til mye, og vi satser på at vi kommer i mål med media nedlasting, og media- visning i løpet av neste uke.

Uke 12 - 18.03 til 21.03

Denne uken har vært en hektisk periode med framgang og utvikling på flere fronter. Vi har jobbet hardt med å forbedre systemets funksjonalitet og ytelse, og resultatene begynner å ta form.

Først og fremst har vi fullført implementeringen av media opplasting på backend. Dette vil videre gi våre brukere muligheten til å enkelt laste opp medieinnhold til plattformen, etter at vi blir ferdig med front- end delen. Vi har også gjennomført viktige endringer i databasen for at det skal bli enklere å arbeide med databasen. Dette vil bidra til raskere og mer pålitelige operasjoner i systemet.

På frontend-siden har vi gjennomført omfattende refaktorering av koden, dette gjelder tabeller, nav- bar og logo.. Dette vil gjøre koden mer funksjonell, lesbar og brukeropplevelse.

En annen milepæl denne uken er at vi har fikset funksjonaliteten for å opprette og oppdatere kampanjer. Dette betyr at brukerne nå kan redigere og tilpasse kampanjene sine etter behov, samtidig som de kan opprette nye kampanjer, noe som vil gi dem større kontroll og fleksibilitet.

Vi har også startet arbeidet med å utvikle frontend for split-kampanjer. Dette er en spennende ny funksjon som vil tillate brukerne å opprette kampanjer med ulike variasjoner for å måle effektiviteten og optimalisere resultatene.

I tillegg er det fortsatt å skrive på bachelor- rapporten for å dokumentere fremdriften vår og analysere eventuelle utfordringer.

Uke 13 - 25.03 til 29.03 - PÅSKEFERIE

Uke 14 - 01.04 til 05.04

Denne uken har vi har konsentrert oss om å forbedre brukeropplevelsen og funksjonaliteten på grafikk-siden ved å fullføre utformingen for å gjøre den mer brukervennlig. I tillegg har vi introdusert en nyttig funksjon som gjør det mulig å slette mediefiler, noe som vil forenkle administrasjonen av innholdet betraktelig. Videre neste uke skal vi jobbe med å få til opplasting, sletting og henting av mediefiler.

En annen viktig oppdatering vi har gjort denne uken er retting av feil i oppretting prosessen for nye kampanjer. Dette sikrer at all nødvendig informasjon blir sendt riktig til databasen. Vi har også gjort betydelige forbedringer slik at redigering av kampanjer fungerer mer optimalt.

For å bedre organisere og estimere arbeidsmengden har vi implementert storypoints i Jira, som vil gi oss bedre kontroll over tidsrammer og ressursallokering. I tillegg har vi begynt å holde daglige stand-ups, noe som vil bidra til å opprettholde en jevn kommunikasjonskanal og sikre at teamet er godt informert om progresjonen og eventuelle hindringer.

På toppen av dette har vi også delt opp skriveoppgaver for både bachelor- og bærekraftsoppgaven for å sikre fremgang på disse viktige områdene. Og sist, men ikke minst, har vi gjennomført mindre databaseendringer for å forbedre kontrollen og effektiviteten i systemet vårt.

1 = Lett

2 = Litt lett

3 = Mellomliggende

5 = Litt vanskelig

8 = Vanskelig

13 = Veldig vanskelig

Uke 15 - 08.04 til 12.04

Denne uken har vi implementert daglige stand-up-møter for å bedre kontrollere prosjektets fremdrift før levering. Et høydepunkt var presentasjonen av webapplikasjonen for de ansatte i Phonero. Vi har også hatt et veiledningsmøte med Hallgeir, der vi diskuterte vårt bachelorprosjekt og strategier for å forberede oss til den kommende eksamenen. Dette har gitt oss klarhet og retning i vårt videre arbeid.

Det har også blitt lagt innsats i å forbedre designet av webapplikasjonen, spesielt gjennom finpussing og utvikling av kampanjedetalj- siden. Vi har arbeidet med å implementere universelle knapper for å vise modals og sikre at input fra nye kampanjer sendes korrekt til backenden.

Videre har vi fortsatt å utvikle grafikk siden og startet arbeidet med å implementere tjenester for å støtte bakgrunnsfunksjonalitet. Oppdateringer av databasestrukturen har også vært en prioritet denne uken.

For å sikre full funksjonalitet når vi integrerer API-et, har vi også jobbet med å inkludere statistikk i systemet. Alt i alt har det vært en travel uke med fremgang på flere fronter, og vi ser frem til å bygge videre på denne progresjonen neste uke.

Uke 16 - 15.04 til 19.04

Denne uken startet med å implementere funksjonalitet for å hente og vise mediefiler fra databasen til frontend. Samtidig har vi også fokusert på muligheten for brukerne å laste opp egne mediefiler til databasen. Denne funksjonaliteten er nå på plass, og

Videre har vi optimalisere og forbedre KampanjeDetaljer-siden, og sikret at den fungerer som forventet. Som en del av kontinuerlig forbedring har vi også startet en refaktoring av submit- og campaignAsDraft-funksjonene. Dette arbeidet bidrar til å forbedre koden vår ved å gjøre den renere, mer strukturert og enklere å vedlikeholde på lang sikt.

På toppen av dette hadde vi et viktig møte om KeyCloak på fredag. Dette møtet markerte starten på integreringen av KeyCloak-løsningen for autentisering og autorisasjon i prosjektet vårt. Dette vil være en nøkkelkomponent for å sikre systemets sikkerhet.

Uke 17 - 20.04 til 24.04

Denne uken har vært preget av fremgang og fokus på flere viktige områder. En av hovedprioriteringene har vært å forbedre brukeropplevelsen ved å muliggjøre opplasting av flere mediefiler samtidig. Dette medfører et behov for en grundig refaktoring av vår nåværende kode for mediefil opplasting. I tillegg er det jobbet med å forbedre designet av utkast og kommende kampanjer.

En annen viktig oppgave denne uken har vært å jobbe med ferdigstilling av en rapport om bærekraft i sammenheng med IS- 305. Dette har involvert forskning, dataanalyse og skriving.

Selv om vi har gjort betydelige fremskritt på flere fronter, har det dessverre vært en glipp i oppfølgingen av sprintplanleggingen. Vi har ikke avsluttet sprint 5, så den har pågått over 5,2 uker. Vi har derfor nå avsluttet denne og startet på sprint 6.

Videre har det vært arbeid pågående med integreringen av Keycloak, men vi venter på deployment med Kubernetes.

Til slutt har det vært arbeid med utviklingen av services, men det tar tid å utvikle.

Uke 18 - 27.04 til 31.04

Denne uken har fokuset vært på å ferdigstille eksamensoppgaven i IS-304, og skrive en god del av bacheloroppgaven slik at vi kan sende den til Hallgeir for veiledning. I tillegg har vi gjort fremgang på programmeringen. Tirsdagen startet med et møte med Hallgeir, hvor vi diskuterte viktige detaljer i forhold til bacheloroppgaven, og utspørring av bacheloroppgaven.

På kodefronten har vi gjort følgende;

Vi har forbedret koden på både Dashboard og Kommende kampanje for å sikre bedre ytelse og funksjonalitet. Det er også implementert en ny funksjonalitet denne uken, med muligheten til å hente bilder fra databasen for å legge til i trinn 2 når en ny kampanje opprettes. Arbeidet med å vise de valgte bildene i kampanjeoppsummeringen er også påbegynt. I tillegg til dette ble det lagt en plan for funksjonaliteten til Upload Graphics Modal, samtidig startet vi implementeringen av dette. Pagination ble også lagt til Kommende Kampanjer-delen for å

forbedre brukeropplevelsen. Det har også vært tid til å arbeide videre på tjenestene og utføre smallfixes på media for å sikre en forbedret funksjonalitet. Til slutt ble det laget en fix for å gjøre det mulig å lagre en ny kampanje som utkast uten å ha skrevet inn en dato.