

Kandidat Nummer: 518
244

Utvikling av en Webapplikasjon for TikTok Innsikt

BAO302

Bachelorprosjekt

Oppdragsgiver: Hepta AS

Innleveringsdato: 14.05.2024

Antall sider: 54

Antall ord: 10534

Tilgjengelighet: Fri ☒ Begrenset ☐

Høyskolen Kristiania



Våren 2024

Denne bacheloroppgaven er gjennomført som en del av utdannelsen ved Høyskolen Kristiania. Høyskolen er ikke ansvarlig for oppgavens metoder, resultater, konklusjoner eller anbefalinger

Innhold

Forord.....	4
Sammendrag.....	5
1. Introduksjon.....	6
1.1. Bakgrunn.....	6
1.2. Problemstilling.....	7
1.2.1. Problemstillingens relevans.....	7
1.2.3. Fordeler for bedrifter.....	8
1.2.4. Fordeler for Hepta.....	8
1.3. Mål for prosjektet.....	9
1.3.1. Om kunden.....	9
1.3.2. Om Gruppen.....	10
1.3.3. Kundens forventninger.....	10
1.3.4. Mål og ambisjoner.....	11
1.3.4. Hvordan vi oppnår vår målsetting.....	11
2. Teori og metode.....	12
2.1. Utviklingsmetode.....	12
Planlegging.....	13
Analyse.....	14
Analyse av Mediabooster's nettside.....	15
Analyse av TRYs nettside.....	16
Hvordan lage en optimal nettside basert på analysen.....	18
2.2. Design.....	19
Implementering.....	20
Agile Metode.....	21
2.3. Grunnlag & uttrykk.....	22
Frontend.....	22
Backend.....	23
Universell utforming.....	24
Responsiv design.....	25
Use Case.....	25
2.4. Språk og teknologier.....	26
JavaScript.....	26
React.....	26
HTML & CSS (Tailwind).....	27
2.5 Verktøy.....	27
Visual Studio Code.....	27
Discord.....	28
Figma.....	28
Google Docs.....	28

Trello.....	28
Supabase.....	29
Git/ Github.....	29
3. Prosjektgjennomføring.....	29
3.1 Planleggingsfasen.....	30
Valg av utviklingsmetode.....	30
Tidsplan.....	30
Arbeidsplan.....	31
Risikoplan.....	33
Målgruppen.....	34
Systemkrav.....	35
Bruksmønster.....	35
3.2 Systemarkitektur.....	36
Wireframes.....	37
Sikkerhet.....	39
Brukergrensesnitt.....	39
3.3. Implementering.....	40
Begynnelsen av Implementering.....	41
Hjemmeside.....	42
Kontaktskjema-modal.....	42
Login og Dashboard.....	43
Tiktok API integrasjon.....	45
Fakturering.....	45
Database.....	47
Siste del av implementeringen.....	47
4. Refleksjon.....	48
4.1. Bruk av programmeringsspråk og verktøy.....	48
4.2. Utviklingsmetode.....	49
4.3. Gruppens samarbeid.....	50
4.4. Applikasjonsutvikling.....	50
4.5. Sikkerhet.....	51
5. Konklusjon.....	52
Referanser.....	53
6. Vedlegg.....	54
6.1 Gruppekontrakt.....	55
6.2 Sikkerhetsvurdering av Hepta AS.....	59

Forord

Denne bacheloroppgaven er gjennomført som avsluttende del av bachelorgraden i informasjonsteknologi ved Høyskolen Kristiania våren 2025. Oppgaven markerer slutten på tre år med faglig utvikling, og har gitt oss verdifull erfaring i å arbeide med en reell kunde og et konkret utviklingsprosjekt.

Prosjektet er gjennomført i samarbeid med Hepta AS, et selskap som arbeider med digital markedsføring og AI-løsninger. Vår oppgave har vært å utvikle en moderne, responsiv og brukervennlig webapplikasjon, med særlig vekt på design, funksjonalitet og integrasjon av innsiktsdata via API. Gjennom arbeidet har vi fått anvende og videreutvikle vår kompetanse innen både frontend- og backend-utvikling, smidige metoder, designprinsipper og verktøy som Supabase, React og Next.js.

Vi ønsker å rette en stor takk til Jonathan Berg i Hepta AS for innsikt, tilbakemeldinger og tillit gjennom hele prosjektperioden. Videre takker vi vår faglige veileder, Rashmi Gupta, for verdifulle innspill, faglig veiledning og støtte når vi har møtt utfordringer underveis.

Vi vil også rette en takk til hverandre i gruppen for et sterkt samarbeid preget av høy innsats, god kommunikasjon og felles målforståelse. Med ulike faglige bakgrunner har vi komplettert hverandre og lært mye – både faglig og personlig.

Vi er stolte av sluttresultatet og de erfaringene vi tar med oss videre i vår profesjonelle karriere.

Sammendrag

Denne bacheloroppgaven er utført av fire studenter ved Høyskolen Kristiania i Oslo, i samarbeid med Hepta AS. Oppgaven tar utgangspunkt i behovet for en ny, moderne og responsiv nettside som skal styrke selskapets digitale tilstedeværelse og gjøre innsiktsdata fra sosiale medier lett tilgjengelig for brukerne.

Hovedmålet har vært å utvikle en webapplikasjon med fokus på brukervennlighet, teknisk funksjonalitet og visuell helhet. Løsningen inkluderer et interaktivt dashboard som visualiserer innsiktsdata basert på mock-data fra TikTok API, samt autentisering, fakturahåndtering og kontaktskjema. Designprosessen er gjennomført i Figma, mens Frontend er utviklet med React og Tailwind CSS. Backend-funksjonalitet, inkludert datalagring og autentisering, er implementert ved hjelp av Supabase.

Prosjektet er organisert etter en smidig utviklingsmetodikk inspirert av Scrum, og det er benyttet verktøy som Trello, GitHub og Discord for samarbeid og prosjektstyring. Arbeidet har styrket vår kompetanse innen fullstack-utvikling, teamarbeid, sikkerhet og praktisk anvendelse av teori i et kundeorientert prosjekt.

Gjennom dette prosjektet har vi erfart hvordan teknologi og design kan kombineres for å skape løsninger med reell verdi for bedrifter, og vi har utviklet ferdigheter som vil være nyttige i videre studier og arbeidsliv.

1. Introduksjon

Denne rapporten redegjør for gjennomføringen av vårt bachelorprosjekt i samarbeid med Hepta AS. Oppgaven gikk ut på å utvikle en responsiv og moderne nettside som kombinerer brukervennlig design, funksjonell teknologi og integrasjoner mot eksterne API-er, med særlig fokus på innsiktsdata fra TikTok.

Prosjektet er delt inn i tre hovedfaser: planlegging, design og implementering. I planleggingsfasen beskriver vi hvordan vi organiserte arbeidet, fordelte roller og la grunnlaget for et strukturert utviklingsløp. Designfasen fokuserer på målgruppen og hvordan vi har arbeidet for å skape en intuitiv og engasjerende brukeropplevelse basert på designprinsipper og konkurrentanalyser. I implementeringsfasen presenterer vi den tekniske oppbygningen av løsningen, hvordan vi møtte ulike utfordringer underveis, og hvordan disse ble løst.

Til slutt reflekterer vi over prosjektførsløpet som helhet, herunder valg av metode, teknologier, gruppedynamikk og samarbeid, samt hvilke faglige og praktiske innsikter vi har tilegnet oss.

1.1. Bakgrunn

Markedet for digital markedsføring og sosiale medier er i kontinuerlig vekst, og bedrifter står overfor utfordringer knyttet til hvordan de best mulig kan nå sine målgrupper. Det finnes et bredt spekter av plattformer og verktøy, noe som gjør det krevende å identifisere de mest effektive løsningene for engasjement og kundetiltrekning.

Vi har valgt denne problemstillingen fordi den adresserer et konkret og aktuelt behov i dagens markedsføringslandskap: Mange virksomheter sliter med å forstå sin målgruppe og skreddersy innholdet sitt deretter, noe som fører til at innholdet ikke treffer riktig publikum.

Med dette som bakteppe ønsker vi å utvikle en webapplikasjon som gir bedrifter tilgang til innsiktsdata og markedsføringsstrategier, basert på TikTok-data. Målet er å gjøre det enklere for brukerne å identifisere hva som fungerer på plattformen, og dermed ta bedre beslutninger basert på faktiske data.

1.2. Problemstilling

“Hvordan kan vi utvikle en brukervennlig og responsiv webapplikasjon som henter innsiktsdata fra TikTok API, visualiserer informasjonen i et interaktivt dashboard, samtidig som løsningen gir verdi for sluttbrukeren og ivaretar sikker håndtering av brukerdata og autentisering?”

Denne problemstillingen har dannet grunnlaget for prosjektets retning, og har styrt valg av teknologier, metoder og funksjonalitet gjennom hele utviklingsløpet. I oppgaven utdyper vi hvordan vi har arbeidet for å besvare problemstillingen, hvilke teknologiske løsninger som er benyttet, og hvordan sikkerhet, design og brukervennlighet er integrert i sluttproduktet.

1.2.1. Problemstillingens relevans

Utviklingen av en webapplikasjon som utnytter innsiktsdata fra TikTok er svært relevant for dagens marked, hvor datadrevet markedsføring blir stadig viktigere. Ved å samle og visualisere slik informasjon i et brukervennlig dashboard, kan både små og mellomstore bedrifter få bedre innsikt i hvordan deres innhold presterer, og dermed optimalisere sin digitale strategi.

Løsningen har potensial til å skape verdi både for sluttbrukere og for Hepta AS som tilbyder. For brukerne gir applikasjonen økt forståelse av publikum og bedre beslutningsgrunnlag. For Hepta representerer det en utvidelse av tjenestetilbudet og et verktøy som kan styrke deres posisjon i markedet for digital rådgivning.

1.2.3. Fordeler for bedrifter

Applikasjonen gir bedrifter et verktøy for å analysere og forstå prestasjonen til innhold i markedsføringen deres. Ved å tilby konkrete innsiktsdata, som visninger, engasjement og trender over tid, kan virksomheter justere sin markedsføringsstrategi for å oppnå bedre resultater.

Løsningen gjør det enklere å sammenligne ulike kampanjer, identifisere hva som fungerer, og ta mer informerte beslutninger. Dette gir økt kontroll over markedsføringsarbeidet og bidrar til mer målrettet kommunikasjon mot ønsket målgruppe. I en tid der konkurransen om synlighet er stor, kan slik datadrevet innsikt være avgjørende for å oppnå effekt og vekst.

1.2.4. Fordeler for Hepta

For Hepta AS representerer applikasjonen en mulighet til å utvide sitt tjenestetilbud med en teknologisk løsning som konkretiserer deres kjernekompetanse innen digital markedsføring. Verktøyet kan brukes aktivt i dialog med eksisterende og potensielle kunder, og demonstrerer hvordan Hepta kan tilby innsiktsbasert rådgivning gjennom datavisualisering og analyse.

Ved å tilby en plattform som gjør komplekse data tilgjengelige og forståelige, styrkes Heptas posisjon som en moderne og innovativ aktør i markedet. Applikasjonen kan også bidra til økt synlighet og merkevarebygging, og dermed tiltrekke nye kunder som søker digitale løsninger med praktisk nytteverdi.

1.3. Mål for prosjektet

Målet med prosjektet er å utvikle en responsiv, brukervennlig og funksjonell webapplikasjon og integrasjon av API-løsninger. Applikasjonen skal gjøre det mulig for brukerne å hente og analysere innsiktsdata fra TikTok via et interaktivt dashboard, og dermed gi bedre forståelse for hvordan innhold presterer på plattformen.

Løsningen skal samtidig ivareta krav til sikkerhet, personvern og brukertilgang, og være enkel å bruke på både mobil og desktop. Som gruppe har vi hatt et tydelig ønske om å utvikle et produkt som gir reell verdi for kunden, både som et teknologisk verktøy og som en støtte i deres strategiske arbeid med markedsføring.

1.3.1. Om kunden

Hepta AS er et norsk selskap som bistår mellomstore bedrifter med å styrke sin digitale tilstedeværelse. De tilbyr tjenester innen strategisk rådgivning, innholdsproduksjon, webdesign og målrettet markedsføring i sosiale medier. I tillegg utvikler Hepta AI-baserte løsninger tilpasset kundenes spesifikke behov, blant annet modeller som automatiserer og effektiviserer arbeidsprosesser.

Gjennom å kombinere teknologi, kreativitet og dataanalyse bidrar Hepta til at kundene kan jobbe mer effektivt og oppnå bedre resultater. Oppgaven vi har fått av selskapet bygger videre på dette verdigrunnlaget, og har som mål å styrke Heptas digitale verktøyportefølje.

1.3.2. Om Gruppen

Prosjektgruppen besto av fire studenter, hvor tre hadde fordypning i cybersikkerhet og én i frontend- og mobilutvikling. Denne sammensetningen ga oss et bredt kompetansegrunnlag, og vi var bevisste på å utnytte hverandres styrker gjennom hele prosjektperioden.

Allerede i oppstarten etablerte vi tydelige rammer for samarbeidet. Vi opprettet en felles kommunikasjonskanal via Discord, utarbeidet en samarbeidskontrakt og avtalte faste ukentlige møter. Dette bidro til struktur, forutsigbarhet og en felles forståelse av både ansvarsfordeling og målsetting. God gruppedynamikk og åpen kommunikasjon har vært avgjørende for fremdriften og kvaliteten på arbeidet vi har levert.

1.3.3. Kundens forventninger

Fra prosjektets begynnelse har det vært viktig for oss å forstå og imøtekomme Hepta AS sine forventninger. I oppstartsmøtene fikk vi tydelig formidlet hvilke krav og ønsker kunden hadde til løsningen. Disse ble konkretisert og dokumentert som grunnlag for videre arbeid. Applikasjonen skulle være ferdigstilt innen andre uke av mai 2025, og følgende hovedforventninger ble definert:

Forventninger	Beskrivelse
Brukervennlig	Løsningen skal være intuitiv og enkel å bruke, uavhengig av enhet eller plattform.
Sikkerhet	Applikasjonen må ivareta personvern og datasikkerhet i tråd med gjeldende regelverk (GDPR).
TikTok API	Systemet skal integrere mot TikTok API for å hente ut og visualisere relevante innsiktsdata.

Disse kravene har fungert som rettesnor for utviklingsarbeidet, og har vært sentrale i vurderingen av både teknisk løsning, designvalg og prioriteringer underveis.

1.3.4. Mål og ambisjoner

Målet vårt har vært å utvikle en sikker og brukervennlig webapplikasjon som gjør det enkelt for brukere å hente og forstå innsiktsdata fra TikTok. Vi ønsket å skape en løsning som hjelper brukerne med å se hvordan innholdet deres presterer, sammenligne ulike datapunkter og få bedre oversikt over hva som fungerer, og hva som ikke fungerer.

Samtidig har vi hatt et klart mål om å levere et produkt som Hepta AS faktisk kan ha nytte av i sitt arbeid. Vi ønsket å utvikle noe som ikke bare ser bra ut, men som fungerer godt og gir verdi, både for kunden og for sluttbrukerne.

Prosjektet skulle etter planen være ferdig innen andre uke i mai 2025. For å nå dette målet la vi stor vekt på strukturert arbeid og tydelig fremdriftsplanlegging. Vi har også vært opptatt av at løsningen skal være enkel å bruke, men samtidig ivareta personvern og datasikkerhet.

I tillegg håper vi at applikasjonen kan bidra til å styrke Hepta AS sin posisjon i markedet, og hjelpe dem med å nå ut til nye kunder som ser verdien i innsiktsbasert markedsføring.

1.3.4. Hvordan vi oppnår vår målsetting

For å nå målene vi satt, var det avgjørende å starte med en tydelig plan og sikre en felles forståelse av prosjektets omfang, krav og tidsrammer. Allerede tidlig i prosessen etablerte vi strukturer for samarbeid, inkludert faste møter, tydelig rollefordeling og et felles rammeverk for hvordan vi skulle jobbe.

Vi fordelte oppgaver basert på gruppemedlemmenes styrker og erfaring, og benyttet Trello for å holde oversikt over fremdrift og ansvarsområder. Dette ga oss en god rytme i arbeidet og gjorde det enklere å justere kursen ved behov.

Underveis har vi hatt tett dialog med både oppdragsgiver og veileder, noe som har gitt oss verdifulle tilbakemeldinger og hjulpet oss med å holde prosjektet på rett spor. Vi har jobbet iterativt, og justert både design og funksjonalitet i takt med nye innspill og erfaringer.

Ved å kombinere strukturert prosjektgjennomføring med fleksibilitet og kontinuerlig kommunikasjon, har vi klart å utvikle en løsning som lever opp til de forventningene vi og kunden hadde til prosjektet.

2. Teori og metode

I dette kapitlet presenterer vi det teoretiske og metodiske grunnlaget for prosjektet. Vi forklarer hvilke utviklingsmetoder vi har valgt å følge, og hvordan vi har planlagt og strukturert arbeidet fra idé til ferdig løsning.

2.1. Utviklingsmetode

For å sikre god struktur og fremdrift i prosjektet valgte vi å følge en etablert utviklingsmetodikk. Det gir forutsigbarhet, tydelig arbeidsdeling og gjør det enklere å koordinere innsatsen i en gruppe. I dette prosjektet tok vi utgangspunkt i System Development Life Cycle (SDLC), en metode som beskriver hele livsløpet til et informasjonssystem – fra planlegging til vedlikehold.

Modellen består av seks faser som følger hverandre i en naturlig rekkefølge:

1. Planlegging: Vi identifiserte kundens behov, definerte mål og utarbeidet en overordnet prosjektplan.

2. Analyse: Vi gjennomførte undersøkelser og kartla krav, målgruppe og funksjonelle behov.
3. Design: Basert på analysen utviklet vi skisser og prototyper som la grunnlaget for implementeringen.
4. Utvikling: Kodebasen ble bygget med utgangspunkt i designet, ved hjelp av moderne rammeverk og verktøy.
5. Implementering: Løsningen ble testet, justert og forberedt for bruk.
6. Vedlikehold: Selv om dette i liten grad var relevant innenfor prosjektets rammer, ble det lagt til rette for videre utvikling og forbedringer.

Ved å følge denne prosessen har vi kunnet jobbe strukturert og målrettet gjennom hele prosjektperioden, samtidig som vi har hatt rom for tilpasning underveis.

Planlegging

Planleggingsfasen la grunnlaget for hele prosjektgjennomføringen. Her tok vi stilling til hvilke verktøy, rammeverk og teknologier vi skulle benytte, og hvordan vi best kunne organisere arbeidet for å nå målene våre innen fristen.

Vi startet med å etablere en tydelig prosjektstruktur, hvor vi definerte roller, satte opp milepæler og fordelte ansvar internt i gruppen. I tillegg utviklet vi en fremdriftsplan som tok høyde for både teknisk utvikling og dokumentasjonsarbeid. Denne planen fungerte som et styringsverktøy gjennom hele perioden.

Vi valgte også tidlig hvilke samarbeidsverktøy vi skulle bruke, blant annet Trello for oppgavehåndtering, Discord for intern kommunikasjon og GitHub for kildekode og versjonskontroll. Dette bidro til god oversikt og effektiv samhandling, uavhengig av hvor vi befant oss fysisk.

En godt gjennomført planleggingsfase ga oss trygghet og retning i arbeidet, og gjorde det enklere å håndtere uforutsette utfordringer senere i prosjektet.

Analyse

Dette avsnittet presenterer en analyse av konkurrenters nettsider, med mål om å hente innsikt til design, funksjonalitet og brukeropplevelse i vår egen løsning. Analysen ble gjennomført ved hjelp av visuell vurdering, brukertesting og heuristisk evaluering. Ved å se på hvordan andre aktører i samme bransje utformer sine nettsider, har vi kunnet identifisere hva som fungerer godt, og hvilke områder som kan forbedres eller gjøres annerledes.

Konkurrentanalyse er en sentral del av både design- og UX-prosessen fordi det gir verdifull innsikt i bransjestandarder og brukernes forventninger. Gjennom slike analyser kan man avdekke hvilke visuelle og funksjonelle elementer som skaper verdi, og samtidig avverge vanlige svakheter eller fallgruver. Dette gir et bedre grunnlag for å utvikle en løsning som fremstår gjennomarbeidet og relevant for målgruppen.

Vi valgte konkurrenter basert på tre kriterier: aktørene måtte enten ha lignende målgruppe eller operere i samme bransje, de skulle være høytpresterende og anerkjente, og de måtte representere en viss variasjon i design og tilnærming. Fokusområdene i analysen var blant annet bruk av farger, typografi og bilder, i tillegg til layout og navigasjonsstruktur. Vi vurderte hvordan nettsidene brukte responsivitet og whitespace for å støtte brukeropplevelsen. Det ble også lagt vekt på hvor effektivt funksjoner og verktøy var plassert, hvordan innhold lastet inn, og den generelle ytelsen til nettsidene.

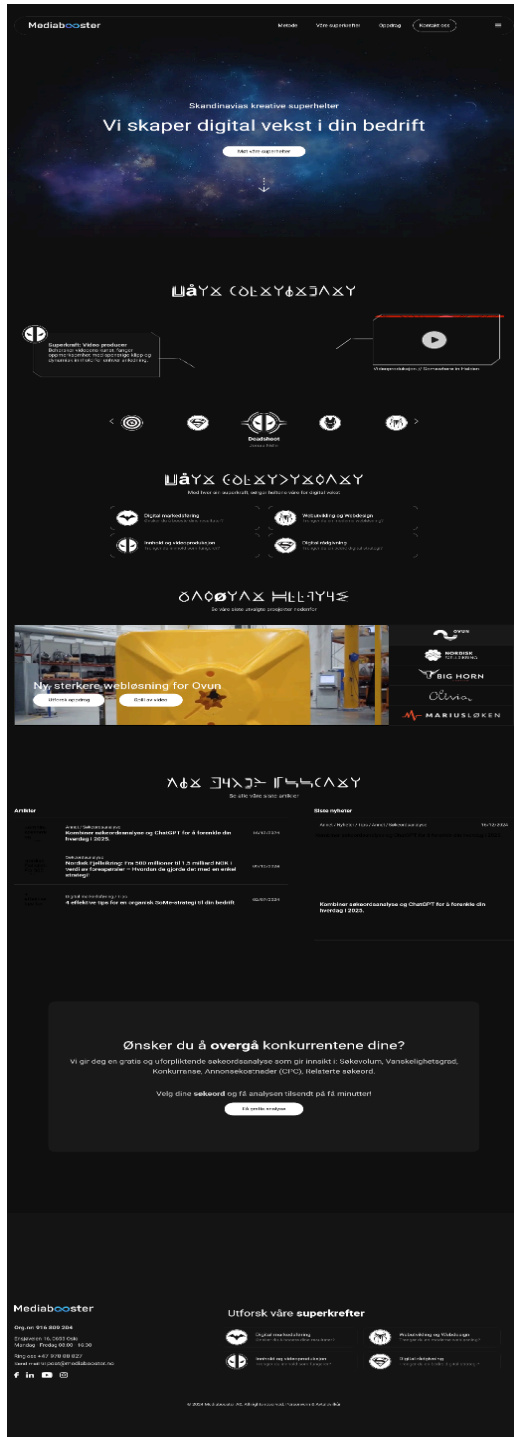
Når det gjelder brukeropplevelse, så vi på navigasjonseffektivitet, tilgjengelighetsfunksjoner og hvordan sidene fungerte på tvers av enheter. Mobilvennlighet og klarhet i innholdet var også en viktig del av vurderingen, ettersom disse faktorene påvirker brukerens inntrykk og opplevelse direkte.

Ved å sammenligne disse aspektene på tvers av konkurrentene, fikk vi et tydelig bilde av hvilke elementer som gir en intuitiv og profesjonell nettside. Innsiktene vi hentet fra analysen har vært til stor hjelp i utformingen av vårt eget design, og har bidratt til at vi har kunnet ta informerte valg som styrker både funksjon og visuell kvalitet i løsningen.

Analyse av Mediabooster's nettside

Dette avsnittet tar for seg en analyse av forsiden til Mediabooster.no, med hensikt å

hente innsikt knyttet til design, funksjonalitet og brukeropplevelse. Analysen ble gjennomført ved hjelp av visuell vurdering og heuristisk evaluering, og fokuserte på hvordan siden kommuniserer sitt budskap, bruker visuelle virkemidler og legger til rette for en effektiv navigasjonsopplevelse.



Nettsiden skaper et sterkt førsteinntrykk gjennom sin visuelle stil. Bakgrunnen med verdensromstema gir assosiasjoner til teknologi og innovasjon, og skaper en moderne og framtidsrettet atmosfære. Hovedbudskapet "Vi skaper digital vekst i din bedrift" fremstår tydelig og profesjonelt, mens undertittelen "Skandinavias kreative superhelter" gir siden et særpreg og bygger identitet på en lett gjenkjennelig måte.

Designmessig benytter siden seg av en minimalistisk layout, der oppmerksomheten rettes mot sentrale elementer som overskrift, handlingsknapp og navigasjonsmeny. Kontrasten mellom tekst og bakgrunn er stort sett god, men bakgrunnsbildets lysstyrke kan i enkelte tilfeller forstyrre lesbarheten. En svak toning av bildet kunne forbedret dette ytterligere.

Navigasjonsmenyen er plassert i øvre del av skjermen og følger brukeren nedover siden (sticky

navigation). Dette er god praksis, da det gir brukeren enkel tilgang til viktige seksjoner som “Metode”, “Våre superkrefter”, “Oppdrag” og “Kontakt oss”. Menyen er oversiktlig og bidrar til en effektiv brukerflyt.

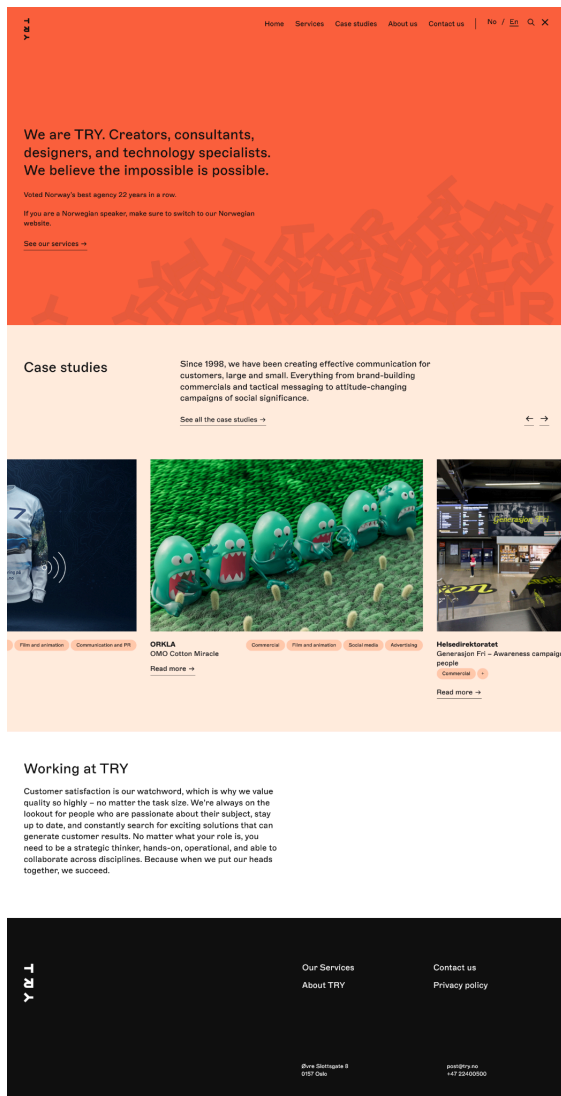
Knappen “Møt våre superhelter” er godt synlig og gir brukeren en klar inngang til videre innhold. Samtidig kunne knappeteksten vært mer handlingsorientert, for eksempel “Se våre tjenester” eller “Utforsk hva vi tilbyr”, for å skape mer konkret forventning til innholdet. Dette ville kunne styrket klikkraten og tydeliggjort brukerreisen.

Totalt sett fremstår forsiden som visuelt sterk og profesjonell, med tydelig merkevarebygging og fokus på brukeropplevelse. Den kommuniserer selskapets identitet på en klar og engasjerende måte. Likevel er det enkelte forbedringspunkter, spesielt når det gjelder tekstkontrast og handlingsorienterte elementer. Disse innsiktene har vært nyttige for vårt eget prosjekt, og har inspirert både til konkrete designvalg og bevissthet rundt visuell kommunikasjon.

Analyse av TRYs nettside

I denne analysen har vi vurdert TRY.no, en nettside utviklet av det kreative byrået TRY. Formålet var å hente inspirasjon knyttet til design, struktur og brukeropplevelse, og å identifisere hvilke grep som bidrar til et profesjonelt og engasjerende uttrykk. Analysen ble gjort med utgangspunkt i visuell vurdering og heuristisk evaluering.

TRYs forside gjør et umiddelbart inntrykk gjennom sin dristige fargebruk og en tydelig visuell identitet. Den røde bakgrunnsfargen skaper oppmerksomhet og gir siden særpreg, men kan også oppleves intens på større skjermer. Typografien er enkel, moderne og lett å lese, noe som gir et profesjonelt og ryddig inntrykk.



Designet balanserer det grafiske uttrykket med funksjonalitet på en god måte. Blant annet benyttes bokstavformer og grafiske elementer i bakgrunnen for å tilføre visuell interesse uten at det tar fokus bort fra innholdet.

Navigasjonsmenyen øverst gir rask tilgang til sentrale seksjoner som “Home”, “Services”, “Case Studies”, “About Us” og “Contact Us”.

Menyen er ryddig, men mangler visuelle tilbakemeldinger som hover-effekter eller markeringer av aktive sider – noe som kunne forbedret navigasjonsopplevelsen ytterligere.

Et positivt trekk er muligheten for å veksle mellom norsk og engelsk språkversjon, men dette valget er plassert relativt diskret og kunne med fordel vært mer synlig.

CTA-knappene, som “See our services” og “See all the case studies”, er godt plassert og fremhever viktige brukerhandlinger. Likevel kunne de vært enda mer effektive ved å

benytte alternative farger eller animasjoner for å skape tydeligere visuell kontrast.

Helhetsinntrykket av TRYs nettside er at den fremstår som moderne, gjennomført og med høy visuell kvalitet. Nettsiden kommuniserer selskapets kreative identitet, samtidig som den gir inntrykk av profesjonalitet og erfaring. Den sterke designprofilen og den tydelige strukturen gjør nettsiden effektiv og brukervennlig, men det finnes likevel forbedringspotensial knyttet til interaktivitet og tilgjengelighet.

Analysen har gitt oss nyttig innsikt i hvordan en tydelig visuell identitet kan kombineres med funksjonell navigasjon. Flere av elementene vi har sett på her, særlig bruken av farge og typografi, har vært inspirasjonskilder for vårt eget prosjekt.

Hvordan lage en optimal nettside basert på analysen

Basert på analysene av nettsidene til Mediabooster og TRY, har vi identifisert en rekke elementer som bør kombineres for å utvikle en funksjonell og profesjonell løsning. Målet har vært å hente inspirasjon fra det beste hos begge aktørene, og samtidig ta høyde for forbedringspunktene vi avdekket. På denne måten kunne vi skape en løsning som både skiller seg ut visuelt og leverer en god brukeropplevelse.

En viktig prioritet har vært å utvikle en tydelig og konsistent visuell identitet. Det innebærer bruk av en balansert fargepalett med god kontrast, uten at fargene virker for påtrengende. Typografien skal være lett å lese, samtidig som det etableres et klart hierarki mellom overskrifter og brødtekst. Ved å kombinere lesbarhet med et moderne uttrykk, ønsket vi å skape et design som oppleves både profesjonelt og brukervennlig. Bruken av høyoppløselige bilder og grafiske elementer ble nøye vurdert, slik at de understøtter innholdet uten å forstyrre helheten.

Navigasjonen måtte være intuitiv og enkel, med en meny som er lett tilgjengelig og fremhever de viktigste sidene, som tjenester, innsikt og kontakt. Vi erfarte fra analysene at selv små detaljer som sticky-meny, hover-effekter og visuell tilbakemelding kan utgjøre en stor forskjell for brukeropplevelsen. Slike grep bidrar til at brukeren raskt forstår hvor de befinner seg, og hvordan de kan navigere videre.

Bruken av whitespace ble også ansett som et viktig virkemiddel. Å gi hvert element nok plass gir ikke bare et ryddigere uttrykk, men gjør det også lettere for brukeren å orientere seg på siden. Vi ønsket å unngå en overfylt visning, og heller prioritere klar struktur og god flyt i presentasjonen av innholdet.

Call-to-action-knapper (CTA) ble plassert strategisk for å veilede brukeren gjennom siden. Basert på det vi så hos konkurrentene, la vi vekt på at knappene ikke bare skulle være visuelt tydelige, men også ha klare og handlingsorienterte tekster. Det skal være lett å forstå hva som skjer når man klikker – for eksempel “Se våre tjenester” i stedet for “Les mer”. Knappene er gitt nok kontrast og interaktive effekter, som fargeendring ved hover, for å gjøre dem mer engasjerende og tilgjengelige.

Ved å trekke inspirasjon fra Mediaboosters struktur og tydelighet, kombinert med TRYs sterke visuelle uttrykk, har vi utviklet en nettside som balanserer estetikk, funksjonalitet og tilgjengelighet. Analysen ga oss et verdifullt beslutningsgrunnlag, og har hjulpet oss å ta gjennomtenkte valg i alt fra layout og brukerflyt til tekniske løsninger.

2.2. Design

Designfasen handler om å omsette innsikt, krav og målgrupper til et visuelt og funksjonelt brukergrensesnitt. I denne fasen jobbet vi med å utvikle wireframes, prototyper og mockups for å planlegge strukturen, interaksjonen og det visuelle uttrykket i løsningen. Hensikten var å skape en applikasjon som er intuitiv, brukervennlig og tiltalende – både estetisk og funksjonelt.

Vi la stor vekt på å sikre en helhetlig visuell identitet som gjenspeiler både Hepta AS sin profil og de tekniske ambisjonene med løsningen. Det innebar nøye vurdering av farger, typografi, ikonbruk og layout. Elementene måtte ikke bare se bra ut, men også understøtte brukerens navigasjon og forståelse av innholdet.

Underveis i designprosessen var det viktig å ta hensyn til brukervennlighet, universell utforming og responsivt design. Vi ønsket at løsningen skulle fungere like godt på mobil som på desktop, og være tilgjengelig for brukere med ulike behov. Tilbakemeldinger fra både veileder og oppdragsgiver ble integrert fortløpende, og bidro til å justere detaljer og forbedre helheten.

For å sikre kvalitet i designvalgene tok vi utgangspunkt i Donald Normans seks prinsipper for godt design: synlighet, sammenheng, respons, konsistens, begrensninger og hint. Disse prinsippene hjalp oss å evaluere løsningen med et brukerperspektiv i fokus, og fungerte som en sjekkliste gjennom hele utviklingsløpet.

Vi valgte blant annet å bruke tydelige kontraster og enkel typografi for å skape synlighet og sikre god lesbarhet. Navigasjon og innhold ble strukturert logisk for å skape

sammenheng, mens små animasjoner og visuelle endringer ved interaksjon ga brukeren respons. Designet ble holdt konsekvent på tvers av sider og komponenter for å sikre forutsigbarhet, og vi valgte å begrense antall valg og knapper for å redusere kognitiv belastning – altså begrensninger. I tillegg brukte vi visuelle virkemidler, som ikoner og fargebruk, for å gi hint om hvordan ulike funksjoner kunne benyttes.

Resultatet av designfasen var en høyoppløselig prototype som ble godkjent av kunden før implementering startet. Denne prototypen la grunnlaget for frontend-utviklingen, og sørget for at det var en tydelig linje fra idé til ferdig produkt.

Implementering

Implementeringsfasen handlet om å gjøre designkonseptet om til en fungerende applikasjon. I denne fasen ble wireframes og prototyper oversatt til faktisk kode, der både det visuelle og funksjonelle ble bygget opp parallelt. Målet var å sikre at løsningen fungerte som planlagt på tvers av enheter og skjermstørrelser, og at brukeropplevelsen samsvarte med det vi hadde skissert i designfasen.

Frontend ble utviklet med React og Next.js, og vi brukte Tailwind CSS som stilverktøy for å oppnå høy grad av fleksibilitet og konsistens. Komponenter ble bygget med tanke på gjenbruk, og det ble lagt vekt på å opprettholde en tydelig struktur i koden. Designet ble brutt ned i mindre byggeklosser, som knapper, skjemaer og kort, for å sikre god oversikt og enklere videreutvikling.

Responsivitet var et sentralt fokuspunkt under implementeringen. Det innebar at vi testet løpende på ulike skjermstørrelser, og justerte layouten slik at elementer tilpasset seg naturlig. Vi hadde særlig fokus på mobilvisning, ettersom mange brukere i målgruppen forventes å bruke løsningen på smarttelefoner.

For at implementeringen skulle bli effektiv og systematisk, opprettholdt vi tett kontakt mellom de som jobbet med design og de som kodet. Dette samarbeidet var viktig for å

sikre at den tekniske løsningen forble tro mot designintensjonene, og at det ble gjort raske justeringer der det var nødvendig.

I tillegg til frontendutviklingen ble det lagt til rette for autentisering, datainnhenting og annen funksjonalitet som beskrevet i prosjektkravene. Vi jobbet kontinuerlig med testing og små iterasjoner for å sikre at komponentene fungerte godt hver for seg – og i helhet.

Implementeringen la grunnlaget for at løsningen kunne fremstå helhetlig, pålitelig og brukervennlig, slik både vi og oppdragsgiver ønsket.

Agile Metode

I dette prosjektet valgte vi å jobbe etter en smidig utviklingsmetode inspirert av Scrum. Grunnen til dette var at vi ønsket en arbeidsform som var fleksibel, tilpasningsdyktig og som kunne håndtere endringer underveis i prosessen. Scrum er godt egnet for teamarbeid der kontinuerlig forbedring, tett samarbeid og hyppige leveranser står sentralt, noe som passet godt til både vår gruppe og prosjektets karakter.

Vi delte prosjektet inn i sprint-lignende perioder med ukentlige milepæler og leveranser. I starten av hver uke satte vi oss konkrete mål, fordelte oppgaver og planla hvordan vi skulle gjennomføre dem. På slutten av uken hadde vi gjennomgang og evaluering, der vi diskuterte hva som fungerte, hva som eventuelt måtte endres, og hvordan vi kunne forbedre samarbeidet videre.

Som prosjektstyringsverktøy brukte vi Trello, som lot oss organisere oppgavene etter “To do”, “Doing” og “Done”. Dette ga god oversikt over fremdriften, og gjorde det enkelt å tilpasse arbeidsmengden etter kapasitet og fremdrift. Vi opplevde at denne visuelle oppdelingen bidro til mer effektivt samarbeid og bedre struktur i hverdagen.

Kommunikasjonen i gruppen foregikk hovedsakelig via Discord, der vi hadde egne kanaler for daglige avklaringer, fremdriftsoppdateringer og felles beslutninger. I tillegg hadde vi ukentlige møter for mer inngående diskusjoner, hvor vi gikk gjennom status,

løste eventuelle utfordringer og tok beslutninger i fellesskap. GitHub ble brukt som versjonskontrollsystem og utviklingsplattform, noe som sikret at alle hadde tilgang til oppdatert kode og kunne jobbe parallelt uten konflikter.

Selv om vi ikke fulgte Scrum-metodikken i sin fulle form, med for eksempel dedikert Scrum Master eller faste sprintseremonier, var prinsippene bak metoden nyttige for å strukturere arbeidet og fremme godt samarbeid. Den smidige tilnærmingen gjorde det lettere å tilpasse oss endringer underveis, og ga oss en god balanse mellom planlegging og fleksibilitet.

2.3. Grunnlag & uttrykk

Frontend

Frontend er den delen av en applikasjon som brukeren interagerer direkte med, og omfatter både det visuelle grensesnittet og logikken som styrer interaksjonene (Granevang, 2023). Frontend-delen av applikasjonen er bygget med React og Next.js, to moderne og fleksible JavaScript-rammeverk som gir god ytelse og mulighet for dynamiske og responsive brukergrensesnitt. Ved å bruke Next.js fikk vi tilgang til funksjoner som server-side rendering og statisk generering, noe som gir raskere lastetid og bedre brukeropplevelse, spesielt viktig for innhold som skal presenteres oversiktlig og effektivt.

Til styling valgte vi Tailwind CSS, et utility-first-rammeverk som ga oss høy grad av kontroll og effektivitet i utviklingsprosessen. Tailwind gjør det enkelt å bygge komponenter med konsistent design, og lar oss jobbe direkte i HTML/JSX uten behov for egne CSS-filer. Dette gjorde det mulig å utvikle visuelt enhetlige komponenter raskt, samtidig som vi opprettholdt fleksibilitet og tilpasningsmuligheter.

Vi fulgte et komponentbasert utviklingsmønster der hver del av brukergrensesnittet, som knapper, skjemaer, kort og navigasjonsmenyer, ble utviklet som selvstendige

komponenter. Dette gir bedre oversikt i koden og gjør det enklere å gjenbruke og vedlikeholde funksjonalitet senere. Ved hjelp av props og state-håndtering i React kunne vi gjøre komponentene dynamiske og tilpasse dem ulike bruksområder.

Responsivt design har vært en sentral del av frontendarbeidet. Vi har sørget for at applikasjonen fungerer sømløst på ulike skjermstørrelser, med særlig vekt på mobilvennlighet. Dette ble løst ved hjelp av Tailwinds innebygde breakpoint-system og ved å teste layout og funksjoner fortløpende på tvers av enheter.

Vi la også vekt på universell utforming ved å sikre tilstrekkelig kontrast, god lesbarhet og tydelige interaksjonselementer. Dette har vært viktig for å gjøre applikasjonen tilgjengelig for et bredt spekter av brukere, og for å møte kravene til moderne webstandarder.

Frontend-delen ble utviklet parallelt med backend-integrasjon, og samarbeidet mellom de to delene har vært avgjørende for å sikre en sømløs og funksjonell løsning.

Backend

Backend omfatter serversiden av en applikasjon og håndterer logikk, databehandling og integrasjoner som driver frontend (The Codest, u.å.). Backend-funksjonaliteten i applikasjonen er hovedsakelig bygget ved hjelp av Supabase, som er en åpen kildekode-alternativ til Firebase. Supabase tilbyr et komplett backend-miljø med støtte for autentisering, database, lagring og serverfunksjoner. Det har vært et sentralt verktøy i utviklingen av prosjektet, og har gjort det mulig for oss å utvikle en sikker og effektiv løsning uten å måtte sette opp og vedlikeholde en egen backend-infrastruktur.

Til datalagring benyttet vi Supabases innebygde PostgreSQL-database, som ga oss en robust og skalerbar løsning. Vi opprettet databaser og tabeller direkte i Supabase-grensesnittet, og satte opp relasjoner og begrensninger etter behov. Bruken av relasjonelle databaser gjorde det enkelt å håndtere strukturerte data som for eksempel fakturaopplysninger, innloggingsinformasjon og innsiktsdata fra TikTok.

Autentisering ble også håndtert gjennom Supabase, som tilbyr et fleksibelt system for innlogging og brukerhåndtering. Vi brukte e-post og passord som primær innloggingsmetode, og satte opp sikkerhetstiltak som krav til sterke passord og validering ved registrering. Brukerdata ble knyttet til spesifikke ressurser i databasen, slik at hver bruker kun har tilgang til sin egen informasjon – i tråd med prinsippene for sikker datatilgang.

Vi benyttet Supabases API-tjenester for å kommunisere med databasen. Dette innebar at frontend-komponentene sendte forespørsler (GET, POST, PUT, DELETE) til spesifikke endepunkter, der data enten ble hentet eller oppdatert. Tilbakemeldingene ble håndtert med feilhåndtering og tilpasning av grensesnittet ut fra responsen – for eksempel ved suksessmeldinger, advarsler eller feilmeldinger.

Valget av Supabase ga oss en fleksibel og moderne backend som fungerte sømløst sammen med Next.js og React på frontend. Det bidro til en raskere utviklingsprosess, mindre kompleksitet og en trygg løsning for både brukerdata og applikasjonslogikk.

Universell utforming

Universell utforming har vært et sentralt hensyn i utviklingen av applikasjonen. Målet har vært å gjøre løsningen tilgjengelig for flest mulig, uavhengig av funksjonsevne, enhet eller teknologisk utgangspunkt. Dette er både et lovpålagt krav og en viktig kvalitetsfaktor i moderne webutvikling.

Vi har fulgt retningslinjene fra WCAG (Web Content Accessibility Guidelines) og lagt vekt på kontrast, tydelig typografi, logisk struktur og lett gjenkjennelige interaksjonselementer. Fargebruk og størrelser er valgt med tanke på lesbarhet og brukervennlighet på både mobil og desktop.

Navigasjonen er bevisst enkel og konsekvent, med tydelig merking og visuell tilbakemelding ved hover og fokus. Vi har også brukt semantisk HTML og strukturert innholdet slik at det fungerer for brukere som benytter skjermlesere. Bilder og ikoner har

fått alternativ tekst der det er relevant. Løsningen er testet på ulike enheter for å sikre at tilgjengeligheten opprettholdes også på mindre skjermer.

Responsiv design

Responsivt design har vært en integrert del av utviklingsarbeidet for å sikre at applikasjonen fungerer like godt på mobil, nettbrett og desktop. Vi har benyttet Tailwind CSS for å tilpasse layout og komponenter til ulike skjermstørrelser gjennom et effektivt breakpoint-system.

Løsningen er bygget etter mobile-first-prinsippet, noe som gjør at kjernefunksjonalitet og innhold fungerer optimalt på mindre skjermer før det skaleres opp til større formater. Underveis har vi testet applikasjonen på ulike enheter for å sikre god lesbarhet, enkel navigasjon og en sømløs brukeropplevelse – uavhengig av plattform.

Use Case

Use Case er en metodikk for å definere systemkrav gjennom interaksjoner mellom brukere og systemet (Brush, 11.2022). For Hepta Dashboard kan et representativt brukstilfelle være: "Administrere kunder".

I dette scenarioet logger en kunde inn via Supabase-autentisering. Deretter navigerer vedkommende til kundeoversikten i dashboardet, en seksjon som i nåværende versjon er tilgjengelig for alle brukere, inkludert funksjoner ment for administratorer. Kunden legger så til en ny kunde via en dialogkomponent fra Radix UI. Informasjonen lagres automatisk i Supabase, og listen over kunder oppdateres umiddelbart med den nye posten.

Dette brukstilfellet illustrerer hvordan frontend (React-komponenter), backend (Supabase) og stil (Tailwind CSS) samhandler for å støtte en spesifikk funksjon i applikasjonen. Selv om intensjonen var å skille mellom kundebrukere og administratorer

med egne grensesnitt, er funksjonaliteten foreløpig åpen for alle brukere. Dette skyldes begrensninger i utviklingstiden og prioritering av andre funksjoner.

2.4. Språk og teknologier

JavaScript

JavaScript utgjør kjernen i Hepta Dashboard og brukes til å skape interaktivitet og håndtere logikk i applikasjonen (The Codest, n.d.). I dette prosjektet har vi valgt å benytte TypeScript, en utvidelse av JavaScript som introduserer statisk typing. Dette gir bedre feilhåndtering under utvikling, økt forutsigbarhet og enklere vedlikehold av koden.

React

React er et komponentbasert JavaScript-bibliotek utviklet av Meta (tidligere Facebook), og ble brukt som grunnlag for frontend-utviklingen i prosjektet. React egner seg godt til å bygge dynamiske og interaktive brukergrensesnitt, og gjør det mulig å strukturere applikasjonen i gjenbrukbare komponenter.

I Hepta Dashboard har vi delt opp grensesnittet i moduler som knapper, skjemaer, kort og navigasjon. Hver komponent håndterer sin egen logikk og stil, noe som gir god oversikt og gjør det enkelt å vedlikeholde eller skalere løsningen videre. Reacts bruk av props og state gir oss kontroll over hvordan komponentene oppfører seg og reagerer på brukerhandlinger i sanntid.

Gjennom bruk av hooks som `useState` og `useEffect`, har vi kunnet håndtere tilstand og sideeffekter på en effektiv og oversiktlig måte. Dette har vært avgjørende for funksjonalitet som datainnhenting, dynamisk rendering og brukerinteraksjon.

React har vært et naturlig valg i prosjektet, både fordi det støttes bredt i bransjen, og fordi det gir høy fleksibilitet og utviklingseffektivitet i kombinasjon med Next.js og TypeScript.

HTML & CSS (Tailwind)

I Hepta Dashboard brukes HTML til å strukturere innholdet, mens utseendet defineres med CSS gjennom Tailwind – et utility-first rammeverk som effektiviserer stilsetting (Jacobsen, 2023; Fonvig, 20XX). Ved hjelp av Next.js genereres HTML dynamisk gjennom JSX-syntaks i komponentfiler som `page.tsx`, hvor struktur og innhold defineres direkte i JavaScript-koden.

Tailwind CSS gjør det mulig å style komponentene ved å bruke klassenavn direkte i komponentene. For eksempel gir `className="bg-gray-100 p-4"` en lys grå bakgrunn og jevn innvendig margin. Denne tilnærmingen gir både konsistens og hastighet, siden det eliminerer behovet for separate CSS-filer og gjør det enkelt å holde oversikt over styling i konteksten der komponenten faktisk brukes.

2.5 Verktøy

Visual Studio Code

Visual Studio Code (VS Code) er et gratis og populært koderedigeringsverktøy utviklet av Microsoft, med støtte for en rekke programmeringsspråk og teknologier (Wikipedia, 2025). Vi brukte VS Code som hovedverktøy for både frontend- og backendutvikling i prosjektet. Redigeringsprogrammet tilbyr funksjoner som IntelliSense, Git-integrasjon og støtte for utvidelser til TypeScript, React og Tailwind.

Discord

Discord er en kommunikasjonsplattform som støtter tekst, tale og video, og brukes av millioner av brukere verden over (Discord, n.d.). Tjenesten er organisert i såkalte “servers”, altså egne rom for grupper med felles interesser. Vi brukte Discord som vårt primære kommunikasjonsverktøy i prosjektet.

Figma

Figma er et nettbasert verktøy for design og prototyping, mye brukt til UI/UX-design i moderne webutvikling (Larsen, 2023). Det gjør det mulig for flere brukere å samarbeide i sanntid, noe som har vært en stor fordel i dette prosjektet. Vi brukte Figma til å utarbeide wireframes, mockups og prototyper av applikasjonen.

Google Docs

Google Docs er en nettbasert plattform for tekstbehandling og samarbeid i sanntid (Max, 2022). Verktøyet gjør det enkelt å skrive, redigere og kommentere dokumenter samtidig, noe som har vært nyttig i forbindelse med både rapportskrivning og deling av idéer i prosjektet.

Takket være skybasert lagring og tilgangskontroll kunne gruppemedlemmene jobbe parallelt og bidra uavhengig av tid og sted, noe som effektiviserte samarbeidet og reduserte behovet for versjonskontroll manuelt.

Trello

Trello er et nettbasert verktøy for prosjektstyring, basert på en tavle- og kortstruktur (Trello, n.d.). I prosjektet brukte vi Trello til å planlegge og fordele oppgaver, holde oversikt over fremdrift og tydeliggjøre ansvar i gruppen.

Supabase

Supabase er en åpen kildekodeplattform for backend-utvikling, og tilbyr blant annet PostgreSQL-database, autentisering og RESTful API-er (Supabase, n.d.). Vi brukte Supabase som både database- og autentiseringsløsning i prosjektet.

Git/ Github

Git er et distribuert versjonskontrollsystem som gjør det mulig å spore og håndtere endringer i kode over tid. GitHub bygger på Git og tilbyr en skybasert plattform for lagring, samarbeid og versjonshåndtering (GitHub, 2024).

3. Prosjektgjennomføring

I dette kapitlet beskriver vi hvordan prosjektgruppen planla og gjennomførte arbeidet fra start til slutt. Vi forklarer og begrunner valg vi har tatt underveis, og viser hvordan prosjektet har fulgt en strukturert prosess gjennom fasene planlegging, analyse, design og implementering, slik de er definert i utviklingsmetodikken vi tok utgangspunkt i.

Vi har hatt jevnlig kontakt med både oppdragsgiver og veileder gjennom hele prosjektperioden. Dette har vært viktig for å sikre at løsningen utvikles i tråd med forventningene, og for å kunne justere kursen basert på tilbakemeldinger og nye innspill.

3.1 Planleggingsfasen

Planleggingsfasen var en av de mest avgjørende delene av prosjektet, og la grunnlaget for videre arbeid. I denne fasen hadde vi jevnlig kontakt med både veileder og oppdragsgiver, samtidig som vi hadde interne diskusjoner for å sikre felles forståelse og retning. Målet var å etablere en realistisk og strukturert plan som tok høyde for prosjektets omfang, tidsramme og ressursbruk.

Vi definerte milepæler, fordelte roller i gruppen og satte opp rutiner for kommunikasjon og fremdriftsstyring. Gjennom god planlegging sørget vi for at alle i gruppen visste hva som skulle gjøres, når det skulle være klart, og hvem som hadde ansvar for hva.

Valg av utviklingsmetode

Valget av utviklingsmetode var avgjørende for å sikre struktur og effektivt samarbeid i prosjektet. En godt tilpasset metode gir klare rammer for arbeidsprosessen, støtter fremdriftsplanlegging og bidrar til å holde tidsfrister.

Etter å ha vurdert ulike alternativer, valgte vi å benytte Scrum som rammeverk for prosjektstyring. Vi opplevde at Scrum passet godt for vår gruppe på fire, ettersom metoden er basert på teamarbeid, fleksibilitet og kontinuerlig forbedring. Rollen som Scrum Master og Product Owner ble fordelt internt, og oppgavene ble delt opp i håndterbare enheter som lot seg gjennomføre innenfor avtalte tidsperioder.

Tidsplan

Allerede i prosjektets oppstart ble det klart at god tidsstyring ville være avgjørende for å sikre fremdrift og måloppnåelse. I starten av januar kartla vi prosjektløpet sammen med oppdragsgiver, og etablerte en overordnet plan med mål om ferdigstilling innen 15. mai.

Første del av prosjektet ble viet til utvikling av nettsiden og implementering av kjernefunksjonalitet. Deretter gikk vi over i en test- og sikkerhetsfase, hvor vi gjennomførte enkle sårbarhetsvurderinger og funksjonstesting. Parallelt med dette arbeidet vi kontinuerlig med dokumentasjonen, slik at rapporten utviklet seg i takt med den tekniske fremgangen.

Arbeidsplan

Arbeidet i prosjektet ble fordelt på fem hovedområder: utvikling, møter, læring, sikkerhet og rapportskrivning. Denne strukturen bidro til tydelig oppgavefordeling og sikret fremdrift i alle deler av prosjektet.

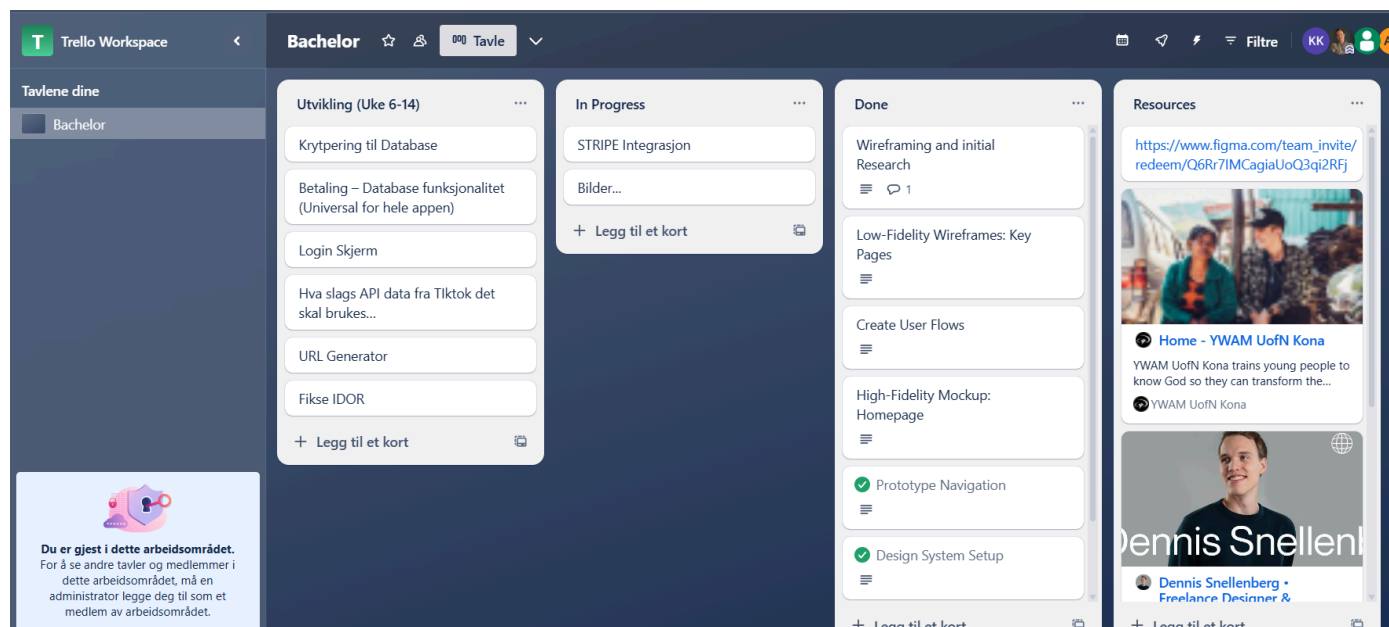
Utviklingen utgjorde den største delen av arbeidet, med fokus på å bygge applikasjonen ved hjelp av HTML, JavaScript, React, Tailwind CSS og Next.js. Gjennom GitHub kunne vi samarbeide effektivt, jobbe parallelt og dele kode. Dette gjorde det enkelt å kombinere våre individuelle styrker og implementere løpende tilbakemeldinger for å forbedre sluttproduktet.

Møter ble avholdt jevnlig, både internt og med oppdragsgiver Hepta AS og veileder. Disse møtene ga oss mulighet til å presentere fremdrift, diskutere utfordringer og motta verdifulle tilbakemeldinger og innspill.

Læring var en naturlig del av prosessen. Flere av verktøyene vi tok i bruk – som React og Supabase – var nye for deler av gruppen. Vi satte derfor av tid til å sette oss inn i disse teknologiene underveis, for å sikre korrekt og effektiv implementering.

Sikkerhet handlet primært om brukertilgang, autentisering og datasikring. Vi prioriterte trygg innlogging, sikker håndtering av brukernavn og passord, samt tiltak for å unngå eksponering av sensitiv informasjon.

Rapportskriving foregikk parallelt med utviklingsarbeidet. Vi dokumenterte fortløpende valg vi tok, hvilke verktøy vi brukte, hvordan vi løste problemstillingen, og hvordan samarbeidet i gruppen fungerte.



Som en del av Scrum-metodikken benyttet vi et visuelt tavlesystem for å organisere arbeidet. Vi brukte Trello, som ga god oversikt over status og fordeling av oppgaver. Tavlen var delt inn i fire kolonner:

Backlog inneholdt alle planlagte oppgaver for sprinten, både tekniske og dokumentasjonsrelaterte. In Progress viste hvilke oppgaver som var aktive, og hva vi jobbet med til enhver tid. Done inneholdt fullførte oppgaver, og ga en oversikt over fremdriften. Resources fungerte som støttekolonne der vi samlet lenker, dokumentasjon og relevante hjelpemidler.

Denne strukturerte tilnærmingen gjorde det enklere å samarbeide effektivt, holde fokus og sikre at alle områder av prosjektet fikk nødvendig oppfølging.

Risikoplan

Vi har utarbeidet en risikoplan som viser ulike risikoer som ofte kan oppstå når man jobber i prosjekter. Vi har tatt utgangspunkt i tidligere prosjekter som vi har gjort på skolen og tenkt på hva som vil ha store konsekvenser for vårt bachelorprosjekt. Vi har delt dette inn i hvor høy sannsynlighet det er for at hver risiko kan skje, med tiltak som må gjøres for å unngå risikoene. Til slutt har vi utarbeidet noen strategier som skal utføres dersom noen av risikoene skjer.

Sannsynlighet:

Veldig sannsynlig

Sannsynlig

Lite sannsynlig

Usannsynlig

Konsekvens:

0-10

Risiko	Sannsynlig	Konsekvens	Tiltak for å unngå	Strategi
Frafall/Sykdom	Sannsynlig	8	Ha godt samarbeid og miljø i gruppen. Oppfølging av sykdom over tid.	Fordele arbeid mellom resterende medlemmer.
Miste data eller arbeid	Usannsynlig	10	Lagre dokumenter jevnlig. Sikkerhetskopiere og lagre på sky.	Tilpasse arbeidsmengde etter hvor mye som er mistet.
For stor arbeidsmengde og dårlig tid.	Lite sannsynlig	8	Legge mye fokus på å planlegge. Lage en plan som viser "fremdrift". Passe jevnlig på at vi ligger godt an i forhold til planen.	Ta et møte med oppdragsgiver der man sammen fokuserer på hva som burde prioriteres.

Uenigheter i gruppen	Usannsynlig	5	Respektere alle sine meninger/tanker, og komme frem til enigheter	Oppfordre alle i gruppen til å dele tanker og meninger og lytte til hva andre har og si.
Tekniske feil	Usannsynlig	8	Delegere oppgaver mer effektivt, planlegge mer og ha dialog med oppdragsgiver	Teste løsninger underveis og ha alternative løsninger på oppgaven.
Misfornøyd kunde	Usannsynlig	9	Holde dialogen med oppdragsgiver jevnlig og sørge for at kravene blir møtt.	Be om tilbakemeldinger underveis og justere arbeidet etter behov.

Målgruppen

Målgruppen definerer hvilke brukere applikasjonen skal utvikles for, og det var viktig for oss å avklare dette tidlig i prosjektet. Å forstå målgruppens behov og forventninger er avgjørende for å kunne lage en løsning som faktisk skaper verdi.

I samarbeid med vår kontaktperson hos Hepta AS, Jonathan Berg, kom vi fram til at målgruppen primært består av mellomstore bedrifter som har behov for rådgivning, innholdsproduksjon og strategisk markedsføring for å styrke sin tilstedeværelse i sosiale medier.

I tillegg tilbyr Hepta utvikling av nettsider og AI-baserte løsninger, der de blant annet hjelper kunder med å trene opp modeller spesifikt til deres behov. Målet er å effektivisere prosesser slik at virksomheter kan spare tid og kostnader gjennom automatisering.

Ved å ha denne målgruppen klart definert, har vi kunnet utvikle en løsning som er tilpasset deres behov og utfordringer, med funksjonalitet som understøtter innsikt, effektivitet og digital vekst.

Systemkrav

Systemkravene ble utarbeidet i tett samarbeid med oppdragsgiver Hepta AS, og har fungert som en styrende ramme for utviklingsprosessen. Kravene omfatter både funksjonelle og tekniske spesifikasjoner som systemet måtte oppfylle for å imøtekomme kundens behov og forventninger. Alle kravene ble gjennomgått og godkjent av oppdragsgiver før implementeringen startet.

Krav nr	Type Krav	Krav Beskrivelse
1	Frontend utvikling	Nettlesere, full støtte for desktop og mobil, javascript for full funksjonalitet, stabilt internett for API_kommunikasjon og e-postfunksjon
2	Backend (Server)	Next.js,
3	Database, Sikkerhet	Supabase, kryptering av brukernavn og passord,
4	API-integrasjoner	Tiktok data, Stripe
5	Brukergrensesnitt	Design Språk (Norsk), Målgruppe

Bruksmønster

Etter at vi hadde kartlagt behovene til Hepta AS og definert systemkravene, utviklet vi et bruksmønster for applikasjonen. Bruksmønstre viser til gjentakende og observerbare handlinger, vaner eller trender brukere har når de samhandler med en programvare (Application Usage Patterns, 2024). Å forstå slike mønstre har hjulpet oss med å holde brukerens forventninger i fokus gjennom hele utviklingsløpet.

Nedenfor vises et sentralt bruksmønster som tar utgangspunkt i applikasjonens viktigste funksjon: visualisering av innsiktsdata fra TikTok via API.

Navn	Aktør	Prioritering
Datavisualisering via API	Bruker	Høy

Forutsetninger

1. Brukeren åpner nettsiden via mobil eller desktop.
2. Vedkommende navigerer til seksjonen for innsiktsdata.
3. Systemet henter sanntidsdata via en forhåndsdefinert API-integrasjon.

Hendelse

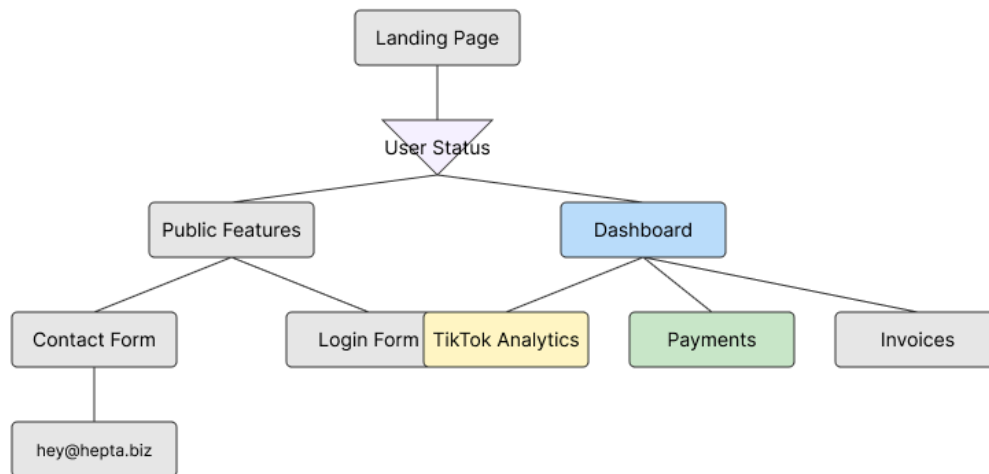
1. Brukeren blir presentert for en interaktiv graf eller tabell med oppdatert informasjon.
2. Det er mulig å filtrere data basert på ulike kriterier, som dato eller kategori.
3. Brukeren kan laste ned rapporten eller dele innsikten med andre.

3.2 Systemarkitektur

Systemarkitekturen beskriver strukturen og samspillet mellom de tekniske komponentene i applikasjonen. Hensikten er å gi en oversikt over hvordan systemet er bygget opp, hvordan data flyter mellom ulike deler, og hvilke teknologier som inngår i løsningen.

Et sentralt element i systemet er databasen, som benyttes til å lagre informasjon om brukere, autentisering og data hentet fra eksterne integrasjoner. Supabase fungerer både som en PostgreSQL-database og som autentiseringssystem. Det tilbyr ferdige

tjenester for brukerhåndtering og tilgangskontroll, noe som har gjort det mulig å implementere sikker og effektiv databehandling uten å utvikle en egen backend.



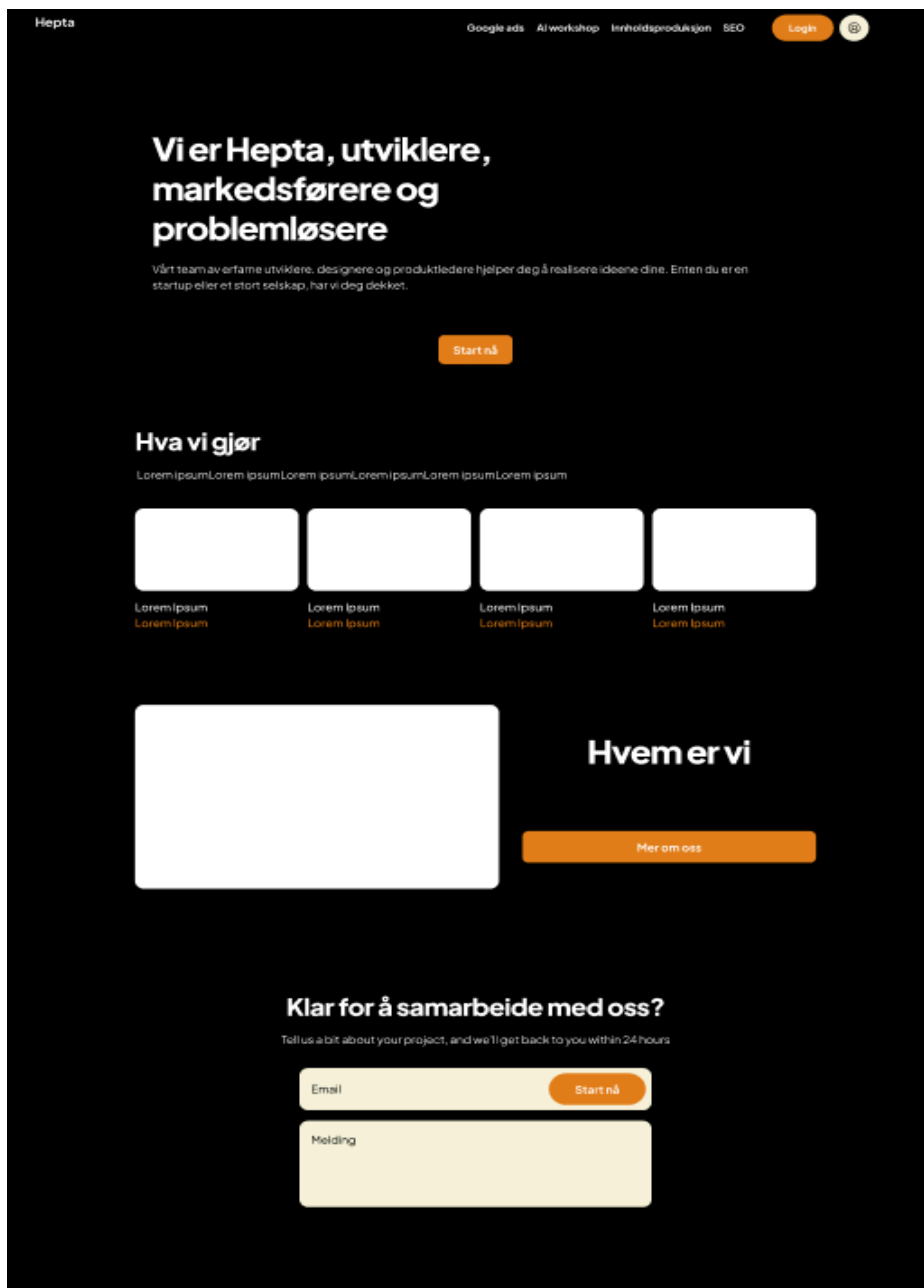
Figuren viser hvordan bruker navigerer gjennom systemet. Landing Page fungerer som et inngangspunkt for alle brukere, og rutes videre basert på om de er innlogget eller ikke:

- U-innloggende brukere får tilgang til “Public Features”, hvor de kan benytte funksjoner som kontakt- og innloggingsskjema.
- Innloggende brukere får tilgang til Dashboard, som samler funksjonalitet som TikTok Analytics, betaling og fakturavisning.

Wireframes

«A wireframe is a rough schematic created in the early stages of digital product design to help visualize and communicate the structure of a product or website» (Guilizzoni, u.å). Wireframes gir utviklere og designere en oversikt over hvordan ulike elementer på en side skal organiseres, og fungerer som et visuelt utgangspunkt før implementering starter.

I vårt prosjekt brukte vi Figma som hovedverktøy for å lage wireframes. Valget falt naturlig ettersom alle i gruppen hadde erfaring med verktøyet fra tidligere prosjekter. Sammen utarbeidet vi flere utkast og gjennomførte justeringer underveis, i tett dialog med kunden. Endelig versjon av wireframen ble valgt etter at vi hadde vurdert struktur, plassering av innhold og brukervennlighet.



Vi benyttet både low-fidelity og high-fidelity wireframes i løpet av utviklingsprosessen. Low-fidelity-versjonene ble brukt tidlig for raskt å skissere idéer og teste struktur, uten fokus på visuell stil. Disse var nyttige for å få oversikt og innspill fra gruppen og oppdragsgiver. Senere utviklet vi high-fidelity wireframes med høyere detaljeringsgrad, inkludert farger, typografi og plassering av faktiske komponenter, for å gi et mer realistisk inntrykk av det endelige produktet. Dette arbeidet gjorde at vi kunne gå inn i implementeringsfasen med en tydelig visjon og

et godt grunnlag for utviklingen av brukergrensesnittet.

Sikkerhet

Sikkerhet har vært et sentralt fokusområde i utviklingen av applikasjonen, både teknisk og med tanke på brukeropplevelse. Informasjonssikkerhet er avgjørende for å bygge tillit, spesielt når det gjelder håndtering av personopplysninger og autentisering. Dette har også hatt betydning for våre vurderinger knyttet til fremtidig videreutvikling av løsningen.

For å sikre etterlevelse av gjeldende regelverk, har vi lagt til grunn prinsippene i GDPR, og utarbeidet utkast til både vilkår for bruk (Terms of Service) og en personvernerklæring (Privacy Policy). Disse dokumentene definerer hvordan persondata behandles, hvilke rettigheter brukeren har, og hvordan data eventuelt deles eller slettes.

Et viktig prinsipp i dette arbeidet har vært at brukeren skal ha kontroll over egne data, inkludert rett til innsyn, retting og sletting. Gjennom tydelig dokumentasjon og sikker teknisk implementering har vi lagt et godt grunnlag for å ivareta både personvern og datasikkerhet i løsningen.

Brukergrensesnitt

Brukergrensesnittet er utformet for å gi en enkel, oversiktlig og brukervennlig opplevelse. Både for gruppen og oppdragsgiver Hepta AS var det viktig at nettsiden oppleves moderne, ryddig og intuitiv, uten unødvendig kompleksitet. Mange digitale løsninger har i dag tunge og uoversiktlige grensesnitt, og dette ønsket vi bevisst å unngå.

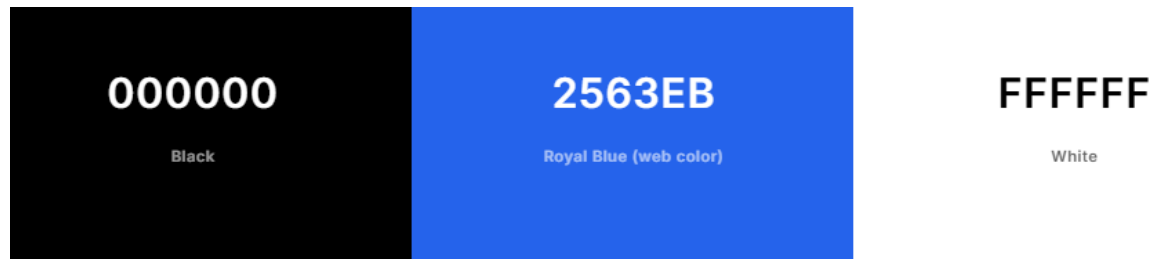
Designet følger en minimalistisk stil som reduserer visuell støy og fremmer en mer fokusert brukeropplevelse. Vi valgte en mørk bakgrunn kombinert med hvit tekst for å sikre høy kontrast og god lesbarhet. Interaktive elementer, som knapper og lenker, er markert med blå for å gjøre dem tydelige og lette å identifisere. Dette skaper en

helhetlig visuell identitet og bidrar til at brukeren enkelt forstår hvor og hvordan de kan samhandle med applikasjonen.

Vi har også hatt universell utforming som en viktig føring i designprosessen.

Tilgjengelighet for alle brukere, inkludert personer med nedsatt syn eller andre funksjonsnedsettelse, har vært prioritert. Vi har derfor valgt løsninger som gir tilstrekkelig kontrast mellom tekst og bakgrunn, logisk og intuitiv navigasjon, samt en romslig layout som gjør det lettere å skille mellom ulike innholdselementer. I tillegg er innholdet strukturert slik at det enkelt kan tolkes av skjermlesere, noe som gjør nettsiden mer tilgjengelig for brukere med behov for assistive teknologier.

Som en del av dokumentasjonen er det inkludert en visuell fremstilling av fargekombinasjonene som er benyttet. Disse består hovedsakelig av svart som bakgrunn, hvit tekst, blå aksenter og blå som signalfarge for interaksjon.



3.3. Implementering

I implementeringsfasen utviklet vi nettsiden for Hepta AS og organiserte arbeidet etter Scrum-metoden. Dette ga oss en strukturert tilnærming, der oppgaver ble fordelt tydelig og fremdrift fulgt opp gjennom sprintene. Samarbeidet foregikk tett, med jevn kommunikasjon og hyppige justeringer underveis.

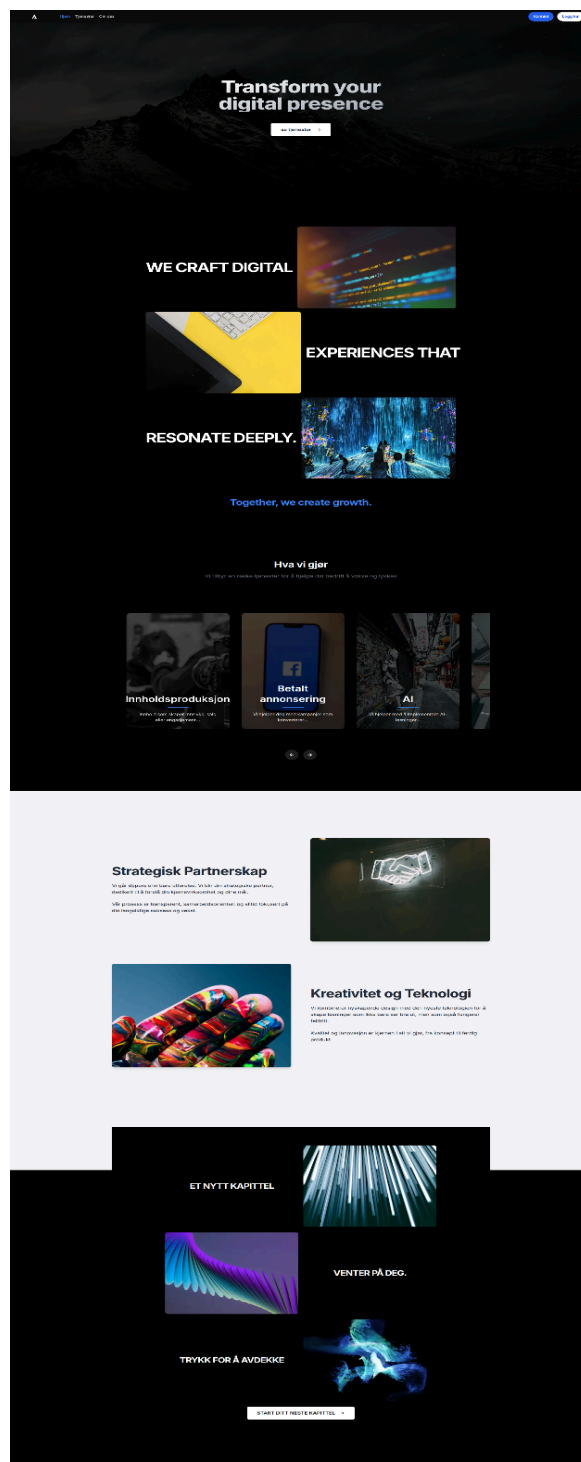
Vi utviklet både frontend og backend, integrerte API-er og implementerte autentisering og datalagring med Supabase. Designet ble tilpasset basert på tidligere utarbeidede wireframes og prototyper. Parallelt gjennomførte vi løpende testing og en enkel sikkerhetsgjennomgang, med fokus på personvern, trygg innlogging og sikker databehandling.

Implementeringen la grunnlaget for en funksjonell og brukervennlig løsning som samsvarte med kravene fra oppdragsgiver.

Begynnelsen av Implementering

Implementeringsfasen startet med en grundig planlegging der vi kartla hvilke funksjoner nettsiden skulle inneholde. Vi samarbeidet tett med Hepta AS for å få en tydelig forståelse av deres behov, målgruppe og forventninger. På bakgrunn av dette utviklet vi tidlige prototyper i Figma, med fokus på brukervennlighet, navigasjonsflyt og visuell struktur.

Designet ble justert løpende gjennom regelmessige møter og tilbakemeldingsrunder med oppdragsgiver. Denne dialogen sikret at både funksjonelle og visuelle valg var i tråd med kundens visjon. Det tidlige arbeidet i denne fasen la et viktig grunnlag for den tekniske



implementeringen og bidro til en mer helhetlig og målrettet utviklingsprosess.

Hjemmeside

Målet med hjemmesiden var å gjøre den så enkel, tydelig og brukervennlig som mulig. Hepta AS skulle fremstå profesjonelt, med klar visuell identitet og en struktur som gjør det enkelt for brukeren å finne frem til relevant innhold. Vi la vekt på tydelig tittel, oversiktlig navigasjon og en logisk oppbygning av siden.

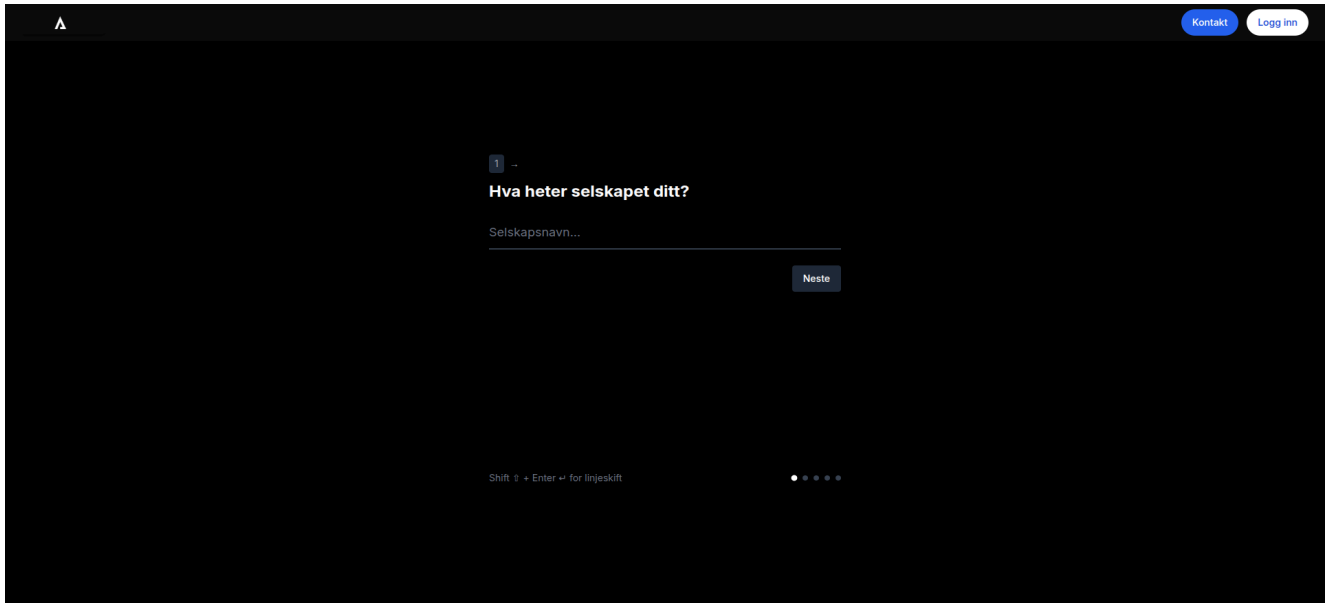
Til utviklingen benyttet vi Tailwind CSS, som gjorde det mulig å bygge responsive og fleksible komponenter som fungerer sømløst på både mobil og desktop. Designet er minimalistisk, med fokus på lesbarhet og ryddighet. Vi brukte oransje som aksentfarge på knapper og interaktive elementer, noe som både forsterker merkevaren og gjør det lettere for brukeren å forstå hvor handling kan utføres.

Kontaktskjema-modal

For å gjøre det enkelt for brukere å kontakte Hepta AS, utviklet vi en kontaktskjema-modal som åpnes direkte på nettsiden. Komponentet er bygget i React med Next.js, og benytter tilstandshåndtering for validering og innsending av data. Skjemaet inneholder feltene navn, e-post, telefonnummer og en valgfri melding, med innebygd logikk for å validere e-post og telefonformat.

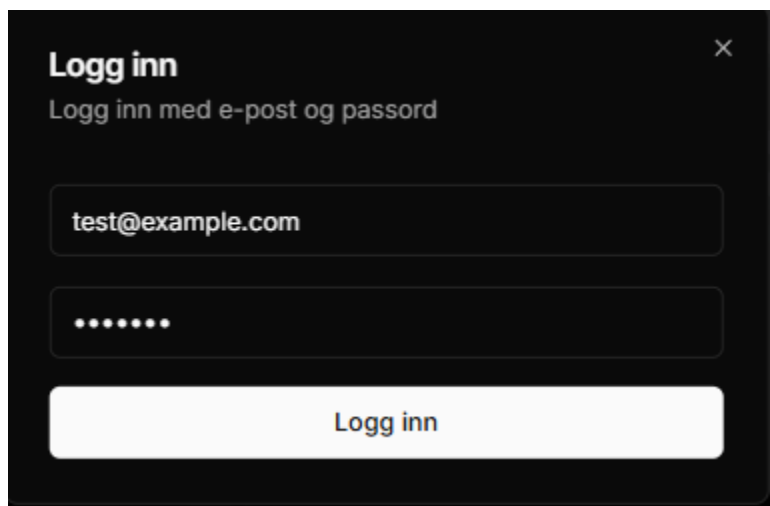
Modalen rendres som en dialog med støtte for klient-side-hydrering, og lukkes automatisk dersom brukeren navigerer bort fra forsiden. Vi har lagt inn visuell tilbakemelding under innsending, inkludert en lastespinner og en bekreftelsesanimasjon ved vellykket innsending. I denne versjonen er skjemaet ikke koblet til en aktiv e-posttjeneste, men innsending simuleres med en forsinkelse (mock submission). Strukturen er imidlertid klargjort for enkel integrasjon med en SMTP-tjeneste eller et API-endepunkt i senere faser.

Skjemaet er responsivt, tilpasset både desktop og mobil, og følger samme designprofil som resten av nettsiden. Det er bygget med Tailwind CSS og bruker komponenter fra det felles UI-biblioteket, noe som sikrer konsistens, vedlikeholdbarhet og en profesjonell brukeropplevelse.



Login og Dashboard

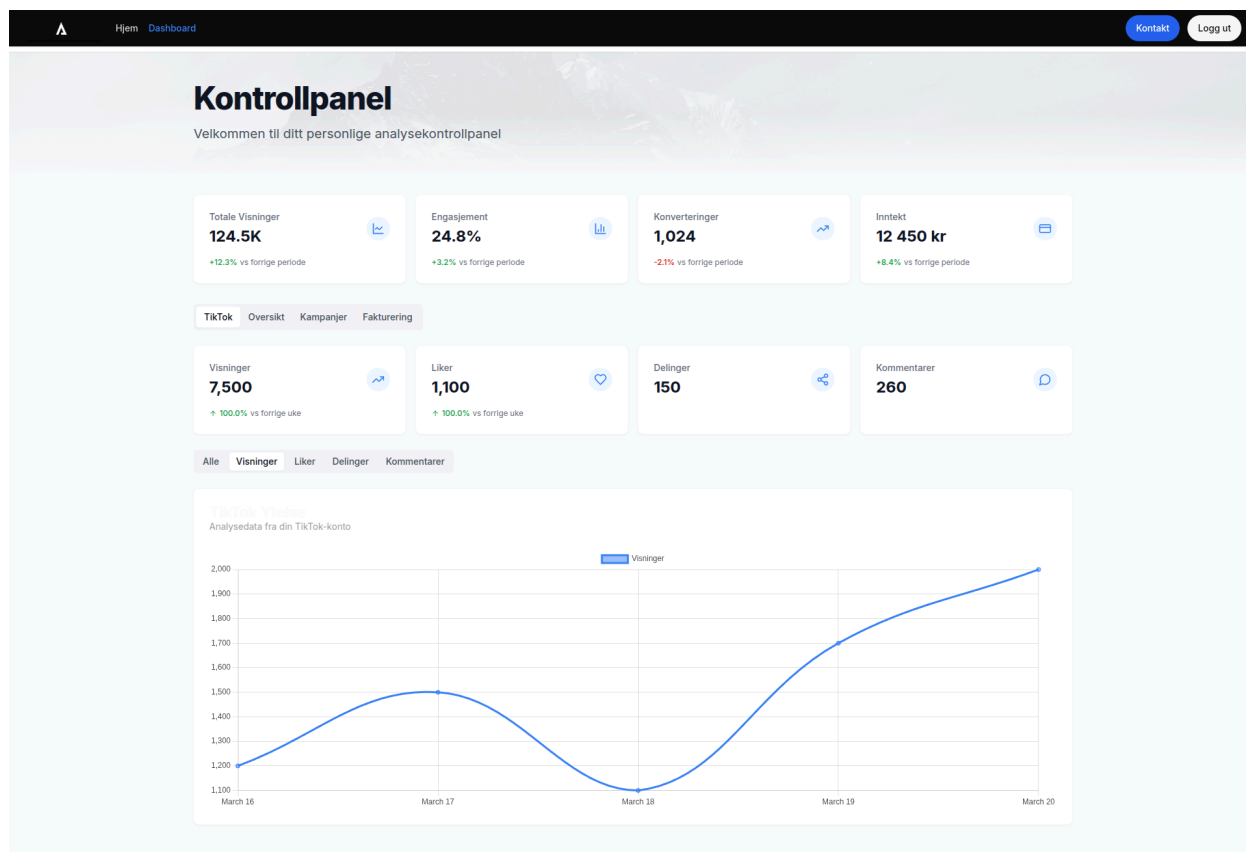
For å gi brukerne en oversiktlig og funksjonell administrasjonsflate, utviklet vi et interaktivt dashboard med innlogging. Dashboardet er bygget i React og Next.js, og organiserer innholdet i faner for TikTok-analyse, kampanjer, fakturaer og generell oversikt. TikTok-seksjonen visualiserer data gjennom komponenten TikTokAnalytics, som foreløpig benytter mock-data i påvente av



full API-integrasjon. De øvrige seksjonene gir brukeren tilgang til kampanjeadministrasjon, fakturavisning og betalingshistorikk, der Stripe-støtte er forberedt gjennom egne komponenter.

For å gi sikker tilgang til dashboardet er det implementert en login-løsning med Supabase-autentisering, hvor brukeren logger inn med e-post og passord. Ved innlogging får brukeren tilgang til et personlig kontrollpanel, der en brukermeny gir muligheter for å administrere profil, innstillinger, fakturering og utlogging. Login-komponenten gir direkte tilbakemelding ved feil eller manglende input, og er integrert sømløst i resten av applikasjonen.

Brukergrensesnittet er responsivt og bygget med Tailwind CSS, med kortbasert visning, ikoner og tydelig fargebruk som gjør informasjonen lett å forstå. Designet balanserer funksjonalitet og enkelhet, og er utviklet med tanke på fremtidig utvidelse og vedlikehold.



TikTok API integrasjon

TikTok-integrasjonen er en sentral del av dashboardet og er implementert gjennom komponenten `tiktok-analytics.tsx`. Formålet er å gi brukeren oversikt over hvordan innholdet deres presterer på TikTok, presentert gjennom visuelle fremstillinger av nøkkeltall som visninger, engasjement og konverteringer. Til visualiseringen bruker vi `Chart.js`, et fleksibelt og populært bibliotek for datavisualisering i React.

Foreløpig benytter komponenten mock-data som er strukturert på samme måte som faktisk respons fra TikToks API. Dette gir oss mulighet til å demonstrere funksjonalitet og design uten å være avhengig av ekstern tilkobling i utviklingsfasen. Når API-tilgangen åpnes, er systemet allerede tilrettelagt for å hente og vise sanntidsdata via backend-integrasjon.

Brukeren kan filtrere data etter parametere som dato eller kampanjetype, søke i datasettet og laste ned en tilpasset rapport. Vi har inkludert ulike diagramtyper – som stolpediagram, linjediagram og doughnut-diagram, for å vise både totalvolum og utvikling over tid. I tillegg benyttes visuelle indikatorer, som fargekoder og piler, for å gjøre det enkelt å tolke trender og endringer.

Målet med denne funksjonaliteten er å gjøre avanserte innsiktsdata lett tilgjengelig og forståelig for brukeren, slik at de kan ta bedre, datadrevne beslutninger i sin markedsføring. Løsningen er også bygget med tanke på videre utvidelse, slik at den i fremtiden kan kobles mot andre plattformer eller flere datastrømmer ved behov.

Fakturering

Som en del av dashboardets funksjonalitet har vi utviklet en betalingsløsning integrert med Stripe. Brukeren får tilgang til en oversikt over fakturaer og kan gjennomføre betaling direkte via et eget skjema, hvor transaksjonen håndteres gjennom Stripe sitt SDK. Løsningen er implementert med `@stripe/react-stripe-js` og `loadStripe`, og er bygget inn i en trygg og brukervennlig React-komponent.

Fakturainformasjon hentes inn og vises med detaljer som beløp, fakturanummer og beskrivelse. I denne versjonen er dataene simulert med mock-informasjon, men strukturen speiler reelle betalingsdata. Når brukeren går videre til betaling, rendres Stripe sitt PaymentElement, som tilpasser seg valgt betalingsmetode og gir en trygg og velkjent opplevelse. Transaksjonen bekreftes med stripe.confirmPayment, og brukeren får løpende tilbakemelding underveis – inkludert feilhåndtering og bekreftelse ved fullført betaling.

Løsningen er responsiv, sikker og teknisk klar for produksjon. Selv om vi i utviklingsfasen bruker et simulert klientSecret, er systemet klart for tilkobling mot ekte betalingsdata. Arkitekturen er også fleksibel, og kan enkelt utvides med funksjoner som kvitteringsutsendelser, abonnementsstøtte eller integrasjon med backend-baserte fakturasystemer.

Ved å bruke Stripe som betalingsplattform får vi tilgang til en sikker og moderne betalingsinfrastruktur uten å måtte håndtere sensitiv informasjon direkte. Dette gjør løsningen både brukervennlig og i tråd med GDPR, og gir Hepta AS et solid utgangspunkt for fremtidig fakturering direkte gjennom dashboardet.

Fakturering

Administrer fakturaer og betalingshistorikk

Betalingsmetoder

Ny betaling

Oversikt

Betalingsmetoder

Abonnement

Aktivt abonnement

Pro Plan - 1 499 kr månedlig

Neste faktura: 15. juni 2025

Administrer

Betalingsmetode

VISA **** 4242 12/2025

Endre

Søk faktura...

Alle statuser

Faktura # t1	Dato	Forfallsdato	Status	Beløp	Handlinger
INV-001	13.05.2025	13.06.2025	Åpen	100.00 NOK	Betal nå
INV-002	14.04.2025	29.04.2025	Betalt	250.00 NOK	
INV-003	09.05.2025	21.05.2025	Åpen	150.00 NOK	Betal nå
INV-004	15.03.2025	30.03.2025	Betalt	350.00 NOK	
INV-005	12.05.2025	28.05.2025	Utkast	50.00 NOK	

Database

Som database- og autentiseringsløsning valgte vi Supabase, som er bygget på PostgreSQL og tilbyr innebygde API-er, sanntidsoppdateringer og sikker datalagring. Selv om flere i gruppen hadde erfaring med MySQL, valgte vi Supabase fordi det gir en mer integrert og effektiv løsning sammen med Next.js, samtidig som det reduserer behovet for manuell backend-konfigurasjon.

All brukerdata, innsiktsdata og betalingsinformasjon fra Stripe lagres i Supabase med kryptering, noe som gir høy grad av datasikkerhet. Valget av Supabase ga oss både skalerbarhet og trygg databehandling, og bidro til å forenkle utviklingsprosessen og spare verdifull tid.

Siste del av implementeringen

Etter at vi hadde implementert de mest sentrale funksjonene i tråd med kravspesifikasjonen fra oppdragsgiver, ønsket vi å presentere en fungerende demo for både vår interne veileder og Hepta AS. Denne demonstrasjonen inkluderte hjemmesiden, fakturasiden og dashboardsiden, der vi viste frem innsiktsvisualisering basert på mock-data fra TikTok API.

Formålet med demoen var å dokumentere fremdriften, få konkrete tilbakemeldinger og forsikre oss om at løsningen svarte til kundens forventninger før videre finpuss og ferdigstilling. Presentasjonen markerte en viktig milepæl, og ga både oss og oppdragsgiver trygghet i at applikasjonen nærmet seg et gjennomarbeidet og funksjonelt sluttprodukt.

4. Refleksjon

I dette kapittelet deler vi våre erfaringer fra prosjektet, og reflekterer over teknologivalgene vi har tatt, samarbeidet i gruppen, arbeidsmetodikken vi har brukt, og utfordringene vi har møtt underveis. Å jobbe med et reelt oppdrag for en ekstern oppdragsgiver har gitt oss verdifull praktisk erfaring og innsikt i hvordan teori fra studiene anvendes i praksis.

Gjennom prosjektet har vi utviklet både tekniske ferdigheter og evnen til å samarbeide effektivt som gruppe. Vi har fått økt forståelse for kravhåndtering, løpende tilpasning i utviklingsløpet og viktigheten av strukturert kommunikasjon. Erfaringene vi har gjort oss, har vært med på å styrke både vår faglige trygghet og vår evne til å gjennomføre komplekse prosjekter i fellesskap.

4.1. Bruk av programmeringsspråk og verktøy

Da vi startet prosjektet i samarbeid med Hepta AS, vurderte vi nøye hvilke språk, rammeverk og verktøy som best kunne møte kravene til funksjonalitet, sikkerhet og brukeropplevelse. Valget falt på en teknologistabel bestående av HTML, JavaScript og CSS, kombinert med Tailwind CSS for å oppnå en responsiv og konsistent layout. Som applikasjonsrammeverk valgte vi Next.js, som bygger på React og støtter både server-side og client-side rendering, noe vi så på som viktig for ytelse og skalerbarhet.

Valgene var delvis basert på tidligere erfaringer, men også på et ønske om å lære nye teknologier. React og Supabase var nye for enkelte i gruppen, men ved hjelp av dokumentasjon, opplæring og praktisk implementering klarte vi å tilegne oss nødvendige ferdigheter i løpet av prosjektet.

Selv om vi vurderte MySQL som database på grunn av tidligere erfaring, endte vi opp med å bruke Supabase. Denne løsningen ga oss både database, autentisering og

sanntidsfunksjoner i ett system, noe som forenklet backend-utviklingen og reduserte behovet for manuell konfigurasjon.

Til utviklingsarbeidet brukte vi Visual Studio Code, som alle i gruppen var godt kjent med. Kildekoden ble versjonskontrollert gjennom GitHub, mens Trello ble brukt til oppgavestyring i tråd med Scrum-metodikken vi fulgte.

4.2. Utviklingsmetode

Som beskrevet i kapittel 3.2.2, valgte vi Scrum som rammeverk for utviklingsprosessen, og dette viste seg å være et godt valg både for prosjektets natur og gruppens sammensetning. Metoden ga oss en kombinasjon av struktur og fleksibilitet, noe som var avgjørende ettersom behov og krav utviklet seg underveis. Gjennom sprintbasert arbeid kunne vi kontinuerlig justere prioriteringer og mål uten å miste oversikten over helheten.

Trello ble brukt aktivt for å sette Scrum ut i praksis. Oppgavene ble organisert i kolonnene “Backlog”, “In Progress” og “Done”, noe som ga god kontroll over arbeidsflyten og sikret at alle visste hva som skulle gjøres, når og av hvem. Dette bidro til en effektiv arbeidsfordeling og god oversikt gjennom hele prosjektperioden.

Vi gjennomførte jevnlige statusmøter og korte, daglige avklaringer, som styrket samarbeidet og gjorde det enklere å oppdage og løse utfordringer tidlig. Den smidige tilnærmingen ga rom for forbedringer underveis, og bidro til at prosjektet kunne leveres innen fristen med ønsket funksjonalitet og kvalitet.

4.3. Gruppens samarbeid

Samarbeidet i gruppen har fungert godt gjennom hele prosjektet. Allerede i oppstarten avklarte vi forventninger og fordelte roller basert på hver enkelt sine ferdigheter og interesser. Dette skapte engasjement og eierskap til oppgavene, og sørget for at alle kunne bidra der de hadde mest å tilby.

Vi etablerte et åpent og inkluderende arbeidsmiljø, hvor det var lav terskel for å komme med innspill, forslag og konstruktiv kritikk. Selv om vi støtte på utfordringer underveis, både tekniske og organisatoriske, klarte vi å løse disse i fellesskap gjennom god kommunikasjon og samarbeid.

Prosjektet har tydelig vist hvor viktig det er med klare roller, gjensidig respekt og strukturert samhandling for å oppnå fremdrift og kvalitet i et utviklingsarbeid.

Erfaringene vi har gjort oss, har styrket vår evne til å samarbeide effektivt i gruppe, en kompetanse vi tar med oss videre i både studier og arbeidsliv.

4.4. Applikasjonsutvikling

Da vi startet utviklingen, hadde gruppemedlemmene ulike tekniske forutsetninger og erfaringer, noe som i utgangspunktet kunne virke utfordrende. Enkelte hadde mest erfaring med frontend-utvikling, mens andre tok hovedansvar for backend- og databasehåndtering. Denne variasjonen viste seg likevel å være en styrke, da vi kunne fordele oppgaver etter kompetanse og samtidig lære av hverandre underveis.

Gjennom prosjektet fikk vi innsikt i hele utviklingsprosessen, fra tidlig design og prototyping til implementering, testing og distribusjon av applikasjonen via Vercel. Denne helhetlige tilnærmingen ga oss bedre forståelse for hvordan ulike deler av en moderne webapplikasjon henger sammen og påvirker hverandre.

Ved å lansere og teste funksjonalitetene tidlig, og hente inn tilbakemeldinger underveis, kunne vi forbedre løsningen kontinuerlig. Dette bidro til høyere kvalitet på sluttproduktet og en mer tilfredsstillende leveranse for oppdragsgiveren.

4.5. Sikkerhet

Sikkerhet har vært et gjennomgående fokus gjennom hele utviklingsprosessen, og vi har oppnådd økt forståelse for hvordan tekniske løsninger må utformes for å håndtere brukerdata på en trygg og forsvarlig måte. Allerede tidlig i prosjektet ble det klart at tillit var en forutsetning for at brukerne skulle benytte systemet, og vi valgte derfor å gjøre sikkerhet til en integrert del av systemarkitekturen, ikke noe som legges til i etterkant.

Dette arbeidet har gitt oss innsikt i både tekniske sikkerhetstiltak og hvordan brukerperspektivet må tas med i utviklingen av digitale tjenester. Prosessen har også vist oss verdien av tverrfaglig samarbeid, der vi har kombinert grunnleggende cybersikkerhet med brukersentrert design. Resultatet er en løsning der sikkerhet ikke går på bekostning av brukervennlighet, og der hele gruppen har fått styrket sin kompetanse i sikker applikasjonsutvikling i praksis.

5. Konklusjon

Gjennom dette bachelorprosjektet har vi utviklet en moderne, responsiv og funksjonell nettside for Hepta AS, med mål om å styrke selskapets digitale tilstedeværelse og gjøre det enklere for brukere å få innsikt i TikTok-data. Prosjektet har omfattet både frontend- og backendutvikling, brukerautentisering, API-integrasjon, datasikkerhet og interaktiv visualisering, og har gitt oss verdifull erfaring med hele utviklingsprosessen, fra idé til ferdig produkt.

Ved å benytte teknologier som Next.js, React, Tailwind CSS og Supabase, har vi utviklet en løsning som kombinerer teknisk robusthet med god brukervennlighet. Gjennom bruk av Scrum-metodikk har vi samarbeidet effektivt som gruppe, og lært mye om prosjektstyring, versjonskontroll og bruk av verktøy som Figma, Trello og GitHub.

Underveis har vi møtt både tekniske og organisatoriske utfordringer, men løst dem i fellesskap gjennom god kommunikasjon og strukturert problemløsning. Arbeidet med ekstern oppdragsgiver har gitt oss innsikt i viktigheten av tydelig dialog og evne til å tilpasse løsningen etter reelle behov.

Prosjektet har vært krevende, men svært lærerikt. Vi avslutter med et produkt vi er stolte av, og en dypere forståelse for hvordan gode teknologivalg, effektivt samarbeid og målrettet utvikling kan føre til et konkret og verdiskapende resultat.

Referanser

Visual Studio Code (Wikipedia) - https://en.wikipedia.org/wiki/Visual_Studio_Code

About Discord (Discord, u.å.) - <https://discord.com/company>

Webdesign med Figma: en knallstart på designfasen (Kevin Larsen, 14.11.2023) -

<https://www.mediaveien.no/aktuelt/webdesign-med-figma-en-knallstart-pa-designfasen>

Den ultimate guiden til Google Dokumenter (Max, 8.3.2022) -

<https://www.ecwid.com/no/blog/guide-to-google-docs.html>

Trello (Trello u.å.) - <https://trello.com/nb>

JavaScript: Det Dynamiske Programmeringsspråket (TheCodest, u.å.) -

<https://thecodest.co/nb/ordbok/javascript/>

Hva er React.js (loadfocus, u.å.) - <https://loadfocus.com/no-no/glossary/react-js>

Hva er HTML? En guide for nybegynnere (Webmakeriet, Tom Jacobsen, 9.9.2023) -

<https://webmakeriet.no/nettside-101/hva-er-html/>

Hva er CSS? (Web Guruen, Steffen Fonvig, 28.04.2023) -

<https://webguruen.no/hva-er-css/>

Frontend (Merethe Granevang, 31,7,2020) - <https://snl.no/frontend>

Backend-Utvikling: En Omfattende Definisjon (The Codest, u.å.) -

<https://thecodest.co/nb/ordbok/backend-utvikling-2/>

Universell utforming (16,02,2024) - https://no.wikipedia.org/wiki/Universell_utforming

Hva er responsivt design? (Forklaringsvideo og infografikk) (Douglas Karr, 09.10.2024) -

<https://no.martech.zone/what-is-responsive-design/>

Use Case (Kate Brush, 11.2024) -

<https://www.techtarget.com/searchsoftwarequality/definition/use-case>

How do I make a wireframe? (Peldi Guilizzoni, u.å.) -

<https://balsamiq.com/learn/articles/what-are-wireframes/>

Supabase - <https://supabase.com/>

Git og GitHub essentials for Microsoft Learn-dokumentasjon () -

<https://learn.microsoft.com/nb-no/contribute/content/git-github-fundamentals>

5 Feil bedrifter gjør i sosiale medier (Christina Andersen, 14.11.2019) -

https://no.journeyagency.com/blogg/digital-markedsforing/5-feil-bedrifter-gjor-i-so-siale-medier/?utm_source=chatgpt.com

Application Usage Pattern (u.å) -

<https://www.prohance.net/glossary/what-is-application-usage-patterns.php>

6. Vedlegg

6.1 Gruppekontrakt

Gruppemedlemskontrakt for Bacheloroppgaven

Prosjekt: Bachelor 2025

Gruppenavn:

Dato: 12.10.2024

Medlemmer av gruppen:

1. Alexander Amundsen
2. Khushal Khan
3. Marius Sperre
4. Ruben C. Tötterman

1. Formål med kontrakten

Denne kontrakten er opprettet for å sikre en strukturert og rettferdig arbeidsprosess i bachelorgruppen. Den skal regulere samarbeid, ansvar, tilstedeværelse og kvalitet i arbeidet. Den inneholder også en prosess for konfliktløsning, inkludert en demokratisk metode for å ekskludere medlemmer som ikke følger opp sine forpliktelser.

2. Tilstedeværelse og oppmøte

Alle gruppemedlemmer **må være til stede** på avtalte møter, både fysisk og digitalt, med mindre det er gitt gyldig beskjed på forhånd. For at gruppen skal lykkes med oppgaven er det avgjørende at alle bidrar til felles diskusjoner og arbeidsprosessene.

- **Forventet oppmøte:** Det kreves at alle møter opp til minst 90% av de planlagte møtene.

- **Fraværsvarsel:** Dersom et medlem ikke kan møte, må det gis beskjed minst 24 timer i forveien. Ved plutselige hendelser (sykdom, nødstilfelle) skal dette meldes så raskt som mulig.
 - **Konsekvenser ved fravær:** Gjentatt fravær uten gyldig grunn eller uten varslings vil kunne føre til vurdering av videre deltakelse i gruppen (se punkt 8 om ekskludering).
-

3. Arbeidsfordeling og ansvarsområder

Arbeidet i gruppen skal fordeles rettferdig basert på kompetanse, interesse og prosjektets krav. Hver person forplikter seg til å utføre sine tildelte oppgaver til avtalt tid og i tråd med kvalitetskravene som gruppen og veileder setter.

- **Ansvarsområder:** Gruppen vil i fellesskap avgjøre ansvarsfordelingen, men det forventes at hvert medlem tar ansvar for sitt område og bidrar til å hjelpe andre ved behov.
 - **Frister:** Alle gruppemedlemmer skal levere sitt arbeid innen de fastsatte fristene for å unngå forsinkelser i prosjektet. Ved problemer med å overholde fristen må dette kommuniseres så tidlig som mulig.
-

4. Kvalitetssikring og revisjon

Alle medlemmer forplikter seg til å opprettholde høy kvalitet på sitt arbeid og delta i revisjon og forbedringer av både individuelle og felles leveranser. Det er viktig at gruppen støtter hverandre i dette arbeidet, ved å gi konstruktive tilbakemeldinger og være åpne for forslag til forbedringer.

- **Kollektiv revisjon:** Før innlevering av større deler av arbeidet, skal gruppen gjennomgå det sammen for å sikre at alle er enige om innhold og kvalitet.
 - **Kvalitetssikring av egne bidrag:** Medlemmene skal være ansvarlige for at eget arbeid er gjennomgått og klart til fremvisning eller innsending før felles revisjon.
-

5. Kommunikasjon

For å sikre effektiv fremdrift skal gruppen opprettholde kontinuerlig og tydelig kommunikasjon gjennom avtalt kommunikasjonsplattform (Discord). Alle medlemmer

forplikter seg til å være tilgjengelige og svare på meldinger innen rimelig tid, normalt innen 4-5 timer maksimalt.

- **Plattform:** Gruppen skal benytte Discord som hovedmiddel for diskusjon og informasjonsdeling.
 - **Oppfølging:** Alle medlemmer forplikter seg til å holde resten av gruppen oppdatert om status på sine oppgaver og melde fra dersom utfordringer oppstår.
-

6. Konfliktløsning

Uenigheter og konflikter som oppstår i gruppen skal løses på en profesjonell og respektfull måte. Først bør diskusjoner tas internt i gruppen, hvor alle medlemmene får mulighet til å gi sine synspunkter. Dersom det ikke finnes en løsning gjennom diskusjon, vil veileder bli informert for hjelp til å mekle.

- **Intern diskusjon:** Før veileder trekkes inn, skal gruppen prøve å finne løsninger på egne utfordringer gjennom åpen kommunikasjon.
 - **Veiledermekling:** Veileder involveres kun dersom konflikten ikke kan løses internt.
-

7. Brudd på kontrakten og konsekvenser

Dersom et medlem bryter kontrakten ved gjentatt fravær, unnlattelse av å levere arbeid, eller annen misligholdelse, kan det vurderes å ekskludere medlemmet fra gruppen. Prosessen for ekskludering skjer gjennom en demokratisk avstemning.

Brudd som kan føre til ekskludering inkluderer, men er ikke begrenset til:

- Uteblivelse fra møter uten gyldig grunn og uten forvarsel (mer enn 2 tilfeller). Medlemmet **skal** varsle om fravær.
 - Manglende bidrag til prosjektet eller gjentatte forsinkelser.
 - Lav kvalitet på arbeidet som påvirker fremdriften til prosjektet negativt.
 - Dårlig kommunikasjon eller uprofesjonell opptreden.
-

8. Prosess for ekskludering av et gruppemedlem

Dersom et gruppemedlem anses som uegnet for videre deltakelse i prosjektet på grunn av gjentatt brudd på kontrakten, kan resten av gruppen ta opp spørsmålet om ekskludering. Denne prosessen vil følge følgende trinn:

1. **Varsel:** Medlemmet det gjelder skal varsles skriftlig om at gruppen vurderer å stemme over en eventuell ekskludering. Medlemmet skal gis mulighet til å forklare sin situasjon og eventuelle årsaker til misligholdelse.
2. **Diskusjon:** Gruppen skal ha en felles diskusjon, inkludert det aktuelle medlemmet, for å vurdere mulige løsninger eller endringer før en avstemning gjennomføres.
3. **Avstemning:** Dersom det ikke oppnås enighet etter diskusjonen, kan en avstemning om ekskludering gjennomføres.
 - **Flertall:** For at ekskluderingen skal være gyldig, må et flertall av de gjenværende medlemmene (minimum 2 av 4) stemme for at medlemmet skal fjernes fra gruppen.
 - **Resultat:** Hvis flertallet stemmer for ekskludering, vil det aktuelle medlemmet ikke lenger være en del av prosjektgruppen. Gruppens veileder vil bli informert om beslutningen, og medlemmet vil være ansvarlig for å rapportere sin individuelle status til veilederen.
4. **Konsekvens:** Et medlem som ekskluderes, vil ikke lenger ha rett til å delta i gruppens arbeid og vil ikke bli kreditert for gruppens innsats i oppgaven. Videre vil medlemmet bli informert om veilederens avgjørelse om hvordan han eller hun kan fortsette sin bacheloroppgave individuelt eller i en annen gruppe.

9. Signaturer

Ved å signere denne kontrakten forplikter hvert medlem seg til å følge de oppsatte reglene og bidra til gruppens suksess i bacheloroppgaven. Dette inkluderer å delta i fellesarbeidet, opprettholde god kommunikasjon og møte alle forventningene til kvalitet og tilstedeværelse.

Marius Sperre

Ruben C. Tøtterman

Dato: 12.10.2023

6.2 Sikkerhetsvurdering av Hepta AS

Prosjektbeskrivelse

Oppdragsbeskrivelse

I samsvar med Hepta AS skal vi gjennomføre en penetrasjonstest på deres nettside. Formålet med testen er å kartlegge eksponert infrastruktur og tjenester utenfra og avsløre svakheter og sårbarheter.

Deltagere

Testen ble gjennomført av bachelorgruppen 101, studenter på Høyskolen Kristiania.

Klassifikasjon av funn

Alle sårbarheter som er oppført i denne rapporten er vurdert ved hjelp av et poengsystem. Vi benytter bransjestandarden Common Vulnerability Scoring System (CVSSv3). CVSS gir et system for å måle alvorlighetsgraden av sårbarheter uavhengig av programvare/maskinvareplattform eller funksjon til tjenesten. Hver sårbarhet tildeles en poengsum mellom null og ti, der 0 representerer ingen risiko og 10 i en alvorlig risiko. Ved å tildele hver oppdaget sårbarhet en score, bidrar jeg til å identifisere de mest sårbare systemene og prioritere løsningen på hvert problem. CVSS-systemet brukes av National Vulnerabilities Database (NVD) for å beregne score for nesten alle kjente sårbarheter, og disse poengsummene refereres til i denne rapporten.

Risikonivå	CVSS	Beskrivelse
Kritisk Risiko	10	Sårbarheten utgjør en trussel mot organisasjonen, og utbedring bør prioriteres <i>Angrep krever ofte stor ekspertise</i>
Høy Risiko	7-9	Vellykket utnyttelse er mulig og kan føre til betydelige forstyrrelser av forretningsfunksjonaliteten. Denne sårbarheten bør utbredes når det er mulig. <i>Angrep krever mer ekspertise for gjennomførelse</i>

Medium Risiko	4-6	Disse sårbarhetene bør tas hånd om så snart man har anledning, da det kan utføre en fare. <i>Angrep er vanskeligere å utføre</i>
Lav Risiko	1-3	Sårbarheten utgjør en ubetydelig/minimal trussel for organisasjonen. Tilstedeværelsen av disse sårbarhetene bør noteres og utbedres hvis mulig.
Informasjon	N/A	Ingen sårbarhet. Disse funnene er kun ment som informasjon og utgjør sannsynligvis ikke umiddelbar fare.

Oppsummering av sårbarhets funn

Sårbarhetsfunn

Sårbarhetsfunn-01 Uautorisert tilgang til Supabase REST API

CVSS 4.3

Medium Risiko

Observasjon

Ved bruk anon-nøkkelen returnerer REST API-et en gyldig, men tom respons.

CVSS

4.3

Beskrivelse

REST API-et til Supabase returnerer en gyldig, men tom liste [] når man gjør kall til tabellen users med apikey for anon-rollen. Selv om det ikke returneres sensitiv data direkte, gir dette bekreftelse på at tabellen eksisterer og kan kartlegges. Dette er uheldig dersom hensikten er å skjule API-strukturen eller tabeller som inneholder brukerdata.

Anbefalinger

Aktiver og kontroller Row Level Security (RLS) på alle tabeller, spesielt users, profiles, etc.

Begrens anon-rollen sin tilgang til kun nødvendige tabeller og felter.

Hvis REST API ikke aktivt brukes av klienten, vurder å deaktivere rest-tilgang fra Supabase dashboard.

Bruk Supabase sine funksjoner (rpc) med eksplisitte policies for mer kontrollert datatilgang.

Referanse

<https://supabase.com/docs/guides/database/postgres/row-level-security>

<https://owasp.org/www-project-api-security/>

Proof Of Concept (PoC)

```
(kali@kali)-[~]  
$ curl -X GET "https://pwcibfnmndbdwdjmdj.supabase.co/rest/v1/profiles?limit=1" \  
-H "apikey: eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpc3MiOiJzdXBhYmFzZSIsInJlZiI6InB3Y2liZm5tbW5kYmR3eGRqbWRqIiwicm9sZSI6ImFub24iLCJpYXQiOiE3NDEwMDUxMTYsImV4cCI6IjU4MTEyOTU0MTExNn0.r7yJNeA4AWA2dd1y4MmS_unB7ERpSKNjWxNKHRRGo6zU"  
[ ]
```

Metodikk

Supabase REST API-endepunkter ble kartlagt manuelt med curl

Tabeller som users, profiles og status ble testet for respons

Ulike autentiseringsnivåer og header-varianter ble brukt for å vurdere tilgang

Konklusjon

Selv om ingen data lekkes direkte, gir API-et informasjon om tabellstruktur og eksistens, noe som i kombinasjon med dårlig RLS-konfigurasjon eller senere feil kan føre til alvorlige datalekkasjer. Dette funnet indikerer en svakhet i tilgangskontrollen og bør adresseres for å redusere angrepsflaten.