



Høgskulen
på Vestlandet

BACHELOROPPGÅVE

Entreprenør portal

Entrepreneur portal

Kenneth Gjelsvik Kvamme

Informasjonsteknologi

Institutt for data- og realfag

Fakultet for ingeniør og naturvitenskap

Innleveringsdato 04.06.2021

Eg stadfestar at arbeidet er sjølvstendig utarbeida, og at referansar/kjeldetilvisingar til alle

kjelder som er brukt i arbeidet er oppgitt, jf. Forskrift om studium og eksamen ved Høgskulen på Vestlandet, § 10.

ENTREPRENØR PORTAL

<i>Rapportens tittel:</i> Entreprenør portal	<i>Dato:</i> 18.04.2021
<i>Forfatter(e):</i> Kenneth Gjelsvik Kvamme	<i>Antall sider u/vedlegg:</i> 40
	<i>Antall sider vedlegg:</i> 2
<i>Studieretning:</i> Informasjonsteknologi	<i>Antall disketter/CD-er:</i> 0
<i>Kontaktperson ved studieretning:</i> Per Christian Engdal	<i>Gradering:</i> Ingen
<i>Merknader:</i>	

<i>Oppdragsgiver:</i> Eninvest	<i>Oppdragsgivers referanse:</i> Jan Roger Myklebust
<i>Oppdragsgivers kontaktperson:</i> Jan Roger Myklebust / Kristoffer Brandsøy	<i>Telefon:</i> 90603339 / 57009164

Sammendrag:

Målet med prosjektet er å forbedre utveksling av informasjon og data mellom Eninvest og entreprenører, ved hjelp av å utvikle en nettbasert portal. Ved utvikling og testing av løysinga vert det følgt kjente prinsipp, mønster og metodar.

Rapporten beskriver korleis dei ulike brukstilfella blir løyst, utviklingsprosessen, og resultatet av løysinga.

Summary:

The goal of this project is to improve the exchange of information and data between Eninvest and entrepreneurs, by developing a web-based portal. The solution is developed and tested by following known principles, patterns and methods.

This report describes how the different use cases are being solved, the development process and the end result of the developed solution.

Stikkord:

Agile	Exploratory testing	LAMP
-------	---------------------	------

Høgskulen på Vestlandet, Fakultet for ingeniør- og natuvitskap

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

Tlf. 55 58 75 00

Fax 55 58 77 90

E-post: post@hvl.no

Hjemmeside: <http://www.hvl.no>

FORORD

Stor takk til intern veileidar, Per Christian Engdal, for utruleg god oppfølging og tilbakemeldingar undervegs i prosjektet. Takk til Eninvest for å ha tilbudd prosjektet. Takk til prosjekteigar i Eninvest for veiledning under design og utviklinga av løysinga. Takk til diverse ansatte i Eninvest som var med på testinga under utviklinga av løysinga.

INNHALDSLISTE

1	INNLEIING	1
1.1	MOTIVASJON OG MÅL	1
1.2	KONTEKST	1
1.3	AVGRENSINGAR	2
1.4	RESSURSAR	3
2	PROSJEKTBESKRIVELSE	4
2.1	PRAKTISK BAKGRUNN	4
2.1.1	<i>Prosjekteigar</i>	4
2.1.2	<i>Tidlegare arbeid</i>	4
2.1.3	<i>Initielle krav</i>	5
2.1.4	<i>Initiell løsnings-idé</i>	6
3	DESIGN AV PROSJEKTET	8
3.1	LØYSING	8
3.2	PROSJEKTMETODIKK	9
3.2.1	<i>Utviklingsmetodikk</i>	9
3.2.2	<i>Prosjektplan</i>	10
3.2.3	<i>Risikovurdering</i>	11
3.3	EVALUERINGSPLAN	12
4	DETALJERT DESIGN	13
4.1	INTRODUKSJON	13
4.1.1	<i>Arkitekturoversikt</i>	13
4.1.2	<i>Mappestruktur</i>	14
4.1.3	<i>Pakkestruktur</i>	15
4.2	BRUKARGRENSESNITT / FRONTEND	16
4.2.1	<i>Skisser til brukardialog</i>	16
4.2.2	<i>Teknologi og bruk av rammeverk</i>	21
4.3	FORRETNINGSLAG / BACKEND	22
4.3.1	<i>Teknologistack</i>	22
4.3.2	<i>Logisk datamodell</i>	23
4.3.3	<i>Sekvensdiagram</i>	24
4.4	SIKKERHEIT	28
4.4.1	<i>Brukartilgang</i>	28
4.4.2	<i>Validering av brukarinput</i>	28

4.4.3	<i>Validering av SQL spørringar</i>	28
4.4.4	<i>Token</i>	28
4.5	INSTALLASJON OG UTRULLING	29
5	EVALUERING	30
5.1	EVALUERINGSMETODE	30
5.2	EVALUERINGSRESULTAT	31
6	RESULTATER	33
6.1	TAB: BESTILLING - BESTILLING	33
6.2	TAB: BESTILLING – ENDRINGSBESTILLING	34
6.3	TAB: ENDRINGSBESTILLINGAR	35
6.4	POPUP: PRODUKTSAMANSETNING	36
7	DISKUSJON	37
8	KONKLUSJON OG VIDERE ARBEID	38
9	REFERANSAR	39
10	VEDLEGG	41
10.1	RISIKOLISTE	41
10.2	GANTT DIAGRAM	42

1 INNLEIING

1.1 Motivasjon og mål

Enivest leiger inn entreprenørar til å utføre installasjon og vedlikehald av fibernett og radiolinjer. Ved enkelte entreprenørar ligg det i botn ei rammeavtale som seier noko om prisar, leveringstider og rutinar for samhandling mellom partane. I dag er kommunikasjonen mellom Enivest og entreprenørar via e-post, telefon og fysiske møter. Ved utveksling av data, som for eksempel material lister tilknytta ein jobb, nyttar ein Excel. Ved installasjon av enkeltkundar har entreprenørane tilgang til ein nettportal for å registrere leveransen, men dei må likevel ringe til kundesenteret for å validere at kunden er på nett.

Målsettinga til Enivest med dette prosjektet er å strukturere og effektivisere prosessane mellom Enivest og entreprenørar, gjennom betra informasjonsflyt og letta kommunikasjon, ved hjelp av ein ny nettportal.

1.2 Kontekst

Enivest jobbar med utbygging av fibernett og radiolinjer til privat- og bedriftsmarknaden. Ved utbygging leiger Enivest inn entreprenørar for å gjere all den fysiske jobben med å føre linja fram til kundane, og installere den i heimen.

For å forstå informasjonsflyten kan ein sjå på ei enkel fiber bestilling til ein bedriftskunde. Etter å ha motteke bestilling frå kunde sender Enivest oppdrag via e-post til entreprenør om å føre fiber fram til kunden. Neste utveksling er at entreprenør svarar tilbake med forventa leveringsdato. Fram mot forventa leveringsdato følger Enivest opp oppdraget via telefon og e-post for å halde oversikt over at entreprenør følger forventa tidsskjema. Her kan det skje hindringar undervegs som fører til utsetting av leveringsdato. Ved større oppdrag vil det også inngå eit Excel-skjema om nødvendig produkt som entreprenøren treng for å utføre oppdraget, og som gir grunnlag for kostnaden ovanfor Enivest.

I tillegg til oppfølging per telefon og e-post, blir det utført regelmessige møter med kvar entreprenør for å halde kontroll over alle oppdraga som er gitt.

Igjennom all informasjonsutveksling vidareformidlar Enivest relevant informasjon frå entreprenørane ut til kundane.

Slik arbeidsflyten er i dag er utvekslinga av informasjon tidkrevande og upålitelig. Dette skapar ei dårlig oppleving for Eninvest sine ansatte, entreprenørane og kundane.

For dei ansatte fører det til tidkrevande manuelle prosessar for å innhente informasjon. Her går det ekstra ressursar for å følge ei kvar bestilling, føre statusoppdateringar inn i Excel ark og vidareformidle informasjonen til kundane.

For entreprenørane er det tungvint å gi tilbakemeldingar i oppdraget ettersom all informasjon går over e-post. Dette resulterer i at Eninvest må, på eige initiativ eller på etterspørsel frå kunde, kontakte entreprenørane for å hente inn informasjonen sjølv over e-post eller telefon.

For kunden kan dagens løysing medføre ei negativ kundeoppleving, ved at ein undervegs ikkje blir oppdatert på planar og status for levering av bestilt linje.

1.3 Avgrensingar

Prosjektet blir bygd på som ein komponent i eit allereie eksisterande nettbasert system. Resultatet av prosjektet skal være ein fullverdig komponent som handtere alle brukstilfella beskrevne seinare i rapporten. Komponenten er meint til å erstatte den tidlegare nettportalen som blir brukt per i dag.

Det eksisterande systemet er hosta på ein Apache web server som kjører på ein Linux maskin. På maskina er der allereie set opp ulike miljø; for testing, utvikling og produksjon. Kodebasen ligg i eit GitHub repository, der ulike greiner blir brukt for henholdsvis testing, utvikling og produksjonsatt kode. For all lagring av data skal MySQL nyttast. Der eksisterar allereie nokre kjernekomponentar som skal gjennbrukast og byggast vidare på i løysinga.

Sidan dette prosjektet skal inngå som ein del av eit eksisterande system vil ein benytte dei allereie etablerte teknologiane. Av dei eksisterande komponentane vil prosjektet nytte seg av databasetilkopling, kundehandtering, sakshandtering og produkthandtering.

Utviklingsarbeidet i prosjektet vil bestå av å kople på nødvendig ny data og bygge på ny funksjonalitet i eksisterande komponentar, samtidig som at all eksisterande funksjonalitet forblir inntakt. Der er nokre nye databasestrukturar som må på plass, men mesterparten av arbeidet består av front end utvikling. Tidleg i prosjektet vart det avtalt at ein av Eninvest sine front end utviklarar skulle bidra i prosjektet for å komme i mål med alt.

1.4 Ressursar

Den viktigaste ressursen i prosjektet er det allerede etablerte systemet som resultatet av prosjektet skal leve i. Der føreligg det komponentar og tenester som prosjektet drar stor nytte av. Ved å benytte desse ressursane sparar prosjektet ein heil del tid som fører til at alle brukstilfella kan bli implementert som del av prosjektet.

I det eksisterande systemet er det tilgjengeleg dokumentasjon av kjernefunksjonar og element som blant anna knappar, tabellar og utfyllingsfelt som ein kan gjenbruke i frontend. I backend manglar dokumentasjon, men alle komponentane følger same struktur og mønster for koden. Dette er veldokumenterte mønster som finns nettressursar på spesifikt for PHP.

I samråd med Eninvest har prosjektet fått ein av utviklarane som tilgjengeleg ressurs på frontend utvikling. Dette blir ein viktig ressurs for å kome i mål med all kodinga. Vidare har prosjektet fått to ansatte på team utbygging i Eninvest for å utføre brukartesting og gi tilbakemeldingar undervegs i utviklinga.

I noverande corona situasjon jobbar prosjektet på heimekontor med Eninvest eigd utstyr via VPN. All møteverksemd foregår over Microsoft Teams. Prosjektet nyttar også Slack for kommunikasjon mellom utviklarane, og ein GitHub bot for varsling om nye feil og oppgåver. For prioritering og planlegging av sprintar blir det brukt ZenHub oppå GitHub for handtering av issues. For å skrive og debugge kode blir det nytta Visual Studio Code. For testing av grensesnitt blir det nytta nettlesar, og Postman for API testing.

2 PROSJEKTBESKRIVELSE

2.1 Praktisk bakgrunn

Den eksisterande nettportalen, som dekkja noko av det dette prosjektet skal dekke, er utvikla av prosjektleiaren for dette prosjektet. Design og utvikling av løysinga vil skje i dialog med blant andre han.

Underteikna har jobba i Eninvest i 4 år, starta som kundemedarbeidar på leveranseteamet, og har nyleg under studiet gått over til utviklar. Som kundemedarbeidar jobba eg blant anna med koordinering av levering av Fiber til bedriftskundar. Dette gir meg eit unikt perspektiv som utviklar der eg har erfaring frå prosessane som no skal bli dekkja av prosjektet. Dette gjer det enklare for meg å sjå moglege løysingar og diskutere løysingar med andre – frå utbyggingsansvarlege til kundemedarbeidarar.

2.1.1 Prosjekteigar

Eigar av prosjektet er Eninvest. Eninvest er ein internettleverandør med rundt 60 tilsette, og har hovudkontor i Førde. Eninvest bygger ut og leverer fiber- og radiolinjer i nærområdet til privat- og bedriftskundar. Eninvest tilbyr også wifi løysingar, digital-TV, breibandstelefonti, e-post, webhotell og datalagring til kundane sine.

Prosjektleiar i Eninvest er Jan Roger Myklebust som er teamleiar for utviklingsteamet. Han driv også med programmering på lik linje med utviklarane. Det gjer det enkelt å diskutere moglege løysingar i prosjektet ettersom han har realistiske ambisjonar om kva som er mogleg å få til.

Prosjektet er viktig for å effektivisere og heve kvaliteten på arbeidet som utførast i tilknytning til utbygging, for dernest å gi ei betre kundeoppleving for Eninvest sine kundar.

2.1.2 Tidlegare arbeid

Det eksisterer i dag ein nettportal der entreprenørane kan logge seg inn, finne kundar og registrere at dei har ferdiginstallert tenesta hos dei. Her er det begrensningar i forhold til kva informasjon som blir lagra, og er meint for å opparbeide ei liste som Eninvest ansatte kan bruke til å validere leveransane.

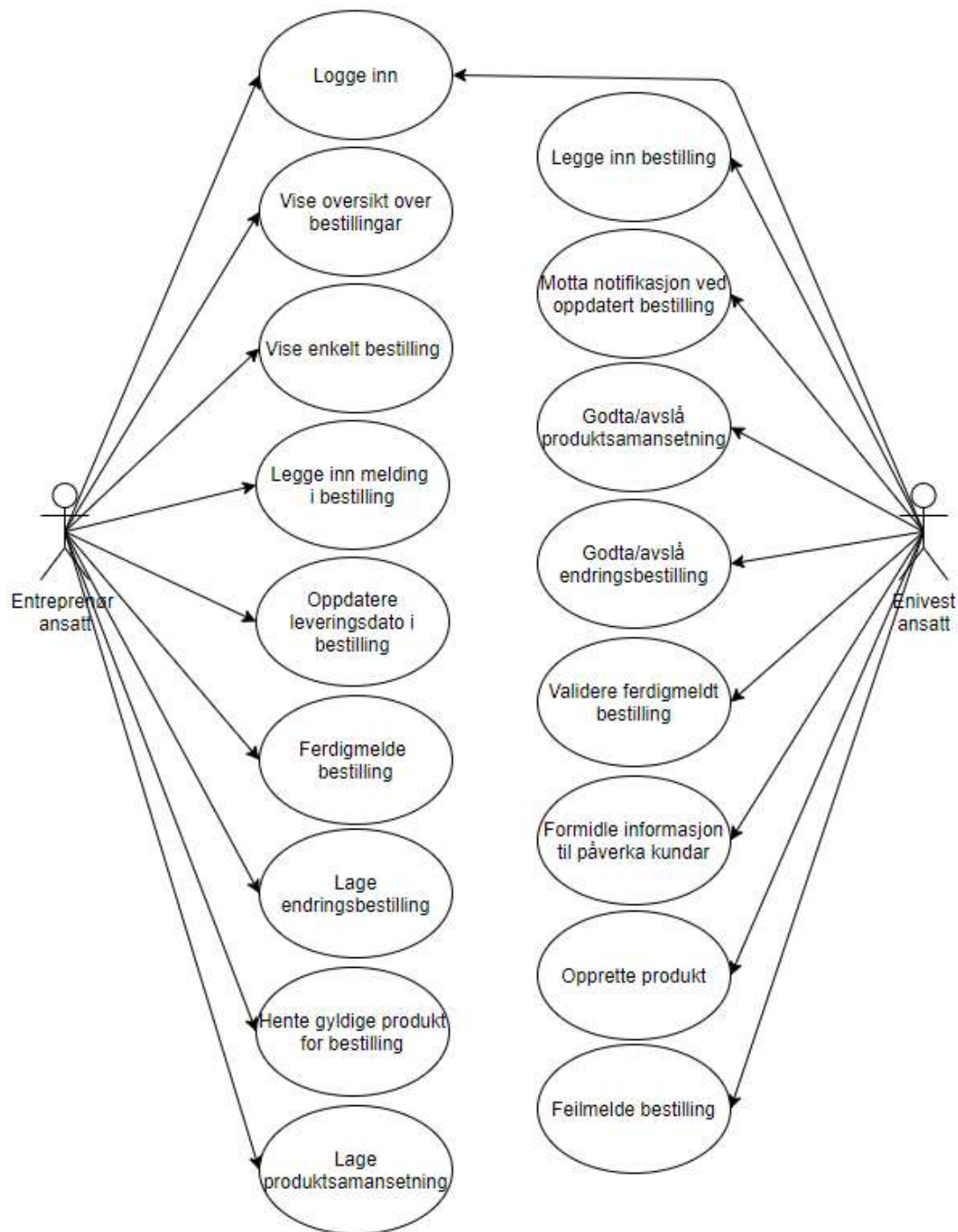
I det nye systemet er det utvikla komponentar som blant anna handterer saker, bestillingsløp og tenesteovervaking. Det føreligg altså fundamentale byggeklossar som gjer det enklare å komme i gang med prosjektet. Underteikna har tidlegare implementert andre komponentar i systemet, og har dermed god kjennskap til gjeldande strukturar og moglegheiter som finnast.

2.1.3 Initielle krav

Dei initielle krava til prosjektet er å tene alle prosessar som inngår i bestillingsløpet til eit oppdrag inn mot entreprenør, samt å støtte rammeavtalar. Krava er nedfelt i brukstilfella som entreprenørane og Eninvest sine ansatte har for å kunne levere ei gitt teneste. Alle brukstilfella som prosjektet skal dekke er gjengitt i Figur 1 – brukstilfeller.

Brukstilfella skal enten utførast via ein brukar av Eninvest sin nettportal, eller via API kall frå eksterne system. Ved bruk av Eninvest sin nettportal skal ein kunne utføre brukstilfella via pc, nettbrett og mobiltelefon.

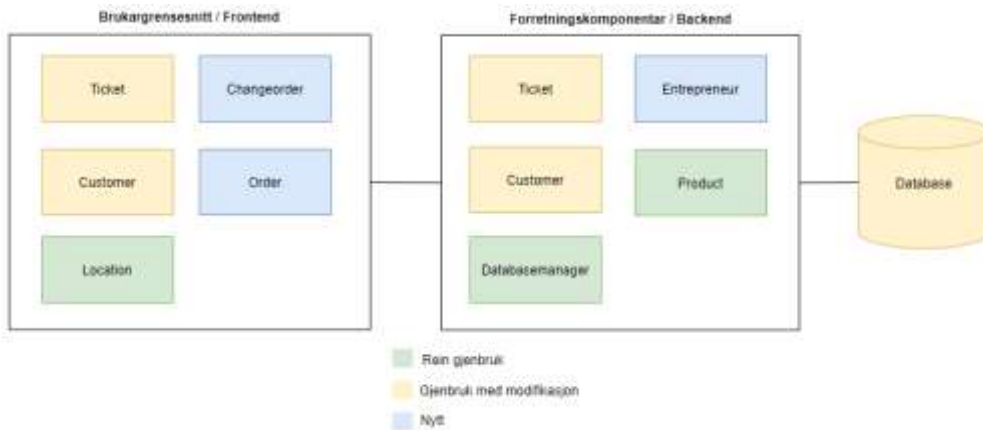
Krav til løysinga er at den skal inngå som en del i Eninvest si eksisterande løysing og infrastruktur, og nytte den teknologi, rammeverk og tekniske løysinga ein har frå før.



Figur 1 – brukstilfeller

2.1.4 Initiell løsnings-idé

Den initelle løysinga går ut på å kunne oppfylle alle brukstilfella ved å ta i bruk eksisterande komponentar og funksjonalitet, og kun lage nytt der det er nødvendig. Løysinga består difor av nokre komponentar som er rein gjenbruk, nokre som må modifiserast og nokre heilt nye.

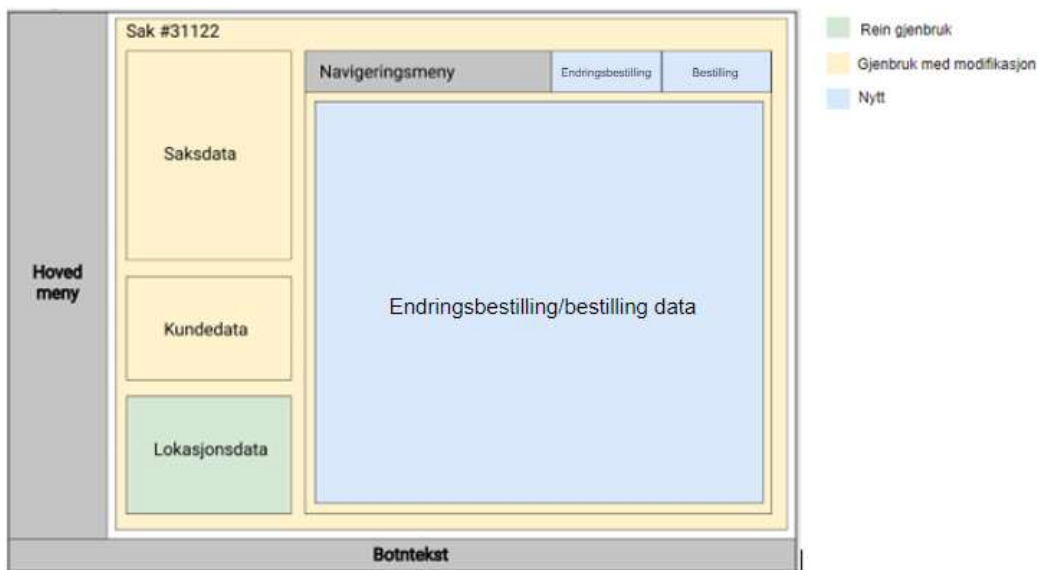


Figur 2 - hovudkomponentar i løysinga

Ved rein gjenbruk vil det seie at det ikkje er behov for å endre i kode eller struktur for komponenten. Desse komponentane vil bli brukt til å holde på nødvendig data. Den nødvendige dataen vert konvertert slik at det passar med eksisterande komponentar.

Komponentane som blir modifisert vil bli tillagt ny funksjonalitet og data for å passe inn med løysinga. Komponentane som er nye må utviklast frå botnen av. I brukargrensesnittet er det to nye komponentar, bestilling og endringsbestilling. I forretningslaget er entreprenør komponenten ny som skal tene dei nye komponentane i brukargrensesnittet.

Figur 3 – komponentar i brukargrensesnittet viser ein eksempel sak med korleis dei ulike komponentane vil framkome i brukargrensesnittet. Figuren viser brukargrensesnittet for ein pc bruker, men nettsida er responsive og blir dynamisk generert for å tilpasse seg nettbrett og mobiltelefonar også.



Figur 3 – komponentar i brukargrensesnittet

3 DESIGN AV PROSJEKTET

3.1 Løysing

Krav til innhald, løysing og bruk av teknologi og verktøy er i dette prosjektet nokså klart og detaljert definert frå oppdragsgjevar. Det har difor ikkje vore naudsynt å vurdere ulike løysningsalternativ mot kvarandre, men heller sjå på korleis brukstilfella kan implementerast i det eksisterande systemet. Oppgåva har dermed vore å vurdere kva eksisterande komponentar og tenester som kan gjenbrukast, og kva som eventuelt må lagast nytt. I vurderinga er det krav om at løysinga skal harmonisere med det eksisterande, og at ein ikkje brukar meir utviklingstid enn nødvendig.

Hovedkomponenten i løysinga er sakskomponenten/ticket. Den blir i dag brukt til å blant anna å handtere bestillingane som Eninvest tek i mot frå kundane. Her vil det bli bygd på funksjonalitet for å kunne opprette ei ny sak som blir bestillinga mot ein gitt entreprenør for å levere bestilt produkt til kunden. Når sakshandterar i Eninvest opprettar bestilling til entreprenør, vil den nye saken bli automatisk fylt inn med all nødvendig informasjon basert på bestillinga frå kunden. Det vil i tillegg gå ut ein e-post til entreprenøren for å varsle om at det er oppretta ei ny bestilling med referanse og direkte lenke til saken i Eninvest sitt system.

I kvar sak har entreprenørane moglegheit til å fylle ut nødvendig data, legge inn meldingar og fylle ut nødvendige skjema. For Eninvest sine ansatte finns det allereie funksjonalitet som gjer at ein mottar varsel når det skjer ei endring i ei sak. På den måten vil Eninvest ha moglegheit til å varsle kunden sin så snart entreprenøren har gitt ny informasjon i saken.

Dei nye komponentane, bestilling/order og endringsbestilling/changeorder, vert tilgjengeleg via sakskomponenten.

Bestillingskomponenten er eit skjema for å legge inn kontaktperson hos entreprenør, ulike datoar, og produkt som entreprenør treng for å utføre jobben. Summen av produkta utgir kostnaden ovanfor Eninvest for at entreprenør utførar bestillinga. Skjemaet blir initiert av Eninvest og fylt ut av entreprenørane. Eninvest ansatte kan godta eller avslå bestillinga og dei valte produkta.

Endringsbestilling er eit skjema for når oppdraget går utover forventa kostnad og/eller tid. Her kan ein også fylle ut nødvendige produkt ved behov. Skjemaet blir sendt inn av entreprenørane, og Eninvest ansatte kan godta eller avslå skjemaet.

For å halde kontroll på alle entreprenørane og deira kontaktpersonar, blir dei registrert via kundekomponenten/customer. Her vil det bli bygd på ein ny kundetype, entreprenør, som kan ha informasjon om eventuelle rammeavtaler.

Dei ulike komponentane vil ha ulik visning og tilgjengeleg funksjonalitet, alt etter om ein er ansatt eller entreprenør. Dette er basert på miljø og rolle til brukaren som er innlogga. For eksempel kan ein entreprenør lage ny eller endre eksisterande endringsmeldingar, medan ein Eninvest ansatt kan kun godta eller avslå endringsmeldingar.

For entreprenørane som vil nytte sitt eige system for å samhandle med Eninvest, vil det bli tilgjengeleggjort API for å kunne nytte same funksjonalitet som ein kan som ein brukar i Eninvest sitt system. Forretningslogikken/backend tilbyr allereie eit API ut mot brukargrensesnittet/frontend. Det er då kort veg for å få eksterne inn på å bruke API'et ved å tildele nøklar, og tilgjengeligjere nødvendig dokumentasjon

3.2 Prosjektmetodikk

3.2.1 Utviklingsmetodikk

I prosjektet er det valt metodikk og rammeverk som passar med prosjektet sin størrelse og egenart. I starten av prosjektet var det naturleg å nytte ein vannfallsmodell (Guru99, ukjent) for å planlegge stegvis tiltak før ein kan starte på utviklinga. Ved utviklinga av løysinga blir det brukt smidige metodikkar som er sentrert rundt ein iterativ utviklingsyklus (Cprime, Ukjent). Kjernen i ein smidig utviklingsmetodikk er å kunne levere fungerande kode ofte, få tilbakemeldingar og tilpasse funksjonaliteten fortløpande. Målet ved å bruke ein smidig metodikk er å kunne tilfredsstille Eninvest sine behov ved at ein raskt kan omstille seg etter Eninvest sine fortløpande tilbakemeldingar.

Prosjektet nyttar element frå Scrum (Cprime, Ukjent), og kvar utviklingsiterasjon definerast som sprintar. Kvar sprint består av aktivitetar om planlegging, design, utvikling, testing og tilbakemelding.

Kvar sprint varer ei veke der planlegging startar kvar måndag med kva som skal inngå i kommande sprint. Her nyttar prosjektet ZenHub til å fylle sprinten med ei rekke oppgåver som skal bli løyst den veka, der kvar oppgåve har ei estimert utviklingstid. Så snart planlegginga er utført, startar ein å designe korleis funksjonaliteten skal passe inn i systemet.

Vidare under utviklinga kodast og testast sjølve innhaldet i det som er prioritert for sprinten. Kvar fredag er det scrummøte med deltagarane i prosjektet frå Enivest si side for å vise fram det som er utvikla, her blir det rom for Enivest til å gi tilbakemelding på løysinga. Dersom det kjem innspel vil det bli oppretta oppgåver i ZenHub som vil gå inn i seinare sprintar.



Figur 4 – sprint

I forhold til skriving av kode blir det følgd anarkjente prinsipp og mønster for å ha ein funksjonell, effektiv og lesbar kodebase. Prinsipp som KISS og DRY (Baghel, 2018) er essensielle for utviklinga der ein forsøkar å ha små og konkrete klassar som har eit spesifisert ansvar. For å oppnå høg samhörighet og løys kopling så delar ein logikken inn i mindre gjenbrukbare metodar for å minske kodeduplisering.

3.2.2 Prosjektplan

For å ha oversikt over dei ulike elementa i prosjektet, er det laga eit GANTT diagram.

Diagrammet er delt delt inn i to fasar, oppstart og utvikling, med definerte hovudaktivitetar. På tvers av fasane ligg rapportskriving der det er planlagt kva kapittel ein kan skrive etter kor langt ein er komen med hovudaktivitetane.

Oppstartsfasen pågår i tidsrommet 17.01.2021 til 31.03.2021. Fasen følgjer ein vannfallsmodell der ein startar med å gjere rede for omfang og innhald i prosjektet, og klargjering av nødvendig data. Deretter går ein over på overordna design av løysing der ein ser på utforminga av nye og eksisterande komponentar, og korleis dei skal harmonisere. Til slutt vert utviklingsfasen planlagt der ein tek ei vurdering av kva rekkefølge dei ulike brukstilfella skal bli implementert i.

Rekkefølga vert bestemt basert på at ein klarar å kode og teste løysinga til ei kvar tid.

Utviklingsfasen pågår i tidsrommet 12.04.2021 til 21.05.2021. I denne fasen er det nytta Scrum der utviklingsarbeidet er delt inn i seks sprintar:

1. I kundedata og produkt sprinten er det tiltenkt å lage klart data for entreprenørane som kundar, og tilhøyrande produkt. Her blir det ein del registreringar for å persistere data inn i det eksisterande systemet sin database.
2. I sakskomponent og bestillingskjema sprinten vert den eksisterande sakskomponenten vidareutvikla for å passe dei nye brukstilfella. I tillegg vert det utvikla eit bestillingskjema for produkta som vart oppretta i førre sprint. Det finns allereie eit bestillingskjema i brukargrensesnittet som kan takast som utgangspunkt, men forretningslogikken må utviklast.
3. I bestilling sprinten skal der utviklast ny funksjonalitet for å støtte bestillingsdata. Her vil det bli laga eit skjema som kan fyllast ut i saken og lagrast i databasen. I tillegg blir det laga funksjonalitet for å kunne godkjenne eller avslå bestillingar, og kopla på bestillingskjema frå førre sprint.
4. I endringsbestilling sprinten skal der utviklast ny funksjonalitet for å støtte endringbestillingsdata. I sprinten vil det bli det laga eit skjema som kan fyllast ut og lagrast i databasen. I tillegg vert det kopla på godkjenning frå førre sprint, og bestillingskjema.
5. I ende til ende logikk sprinten skal der sjåast på manglar i logikk for brukaren å kunne navigere seg fritt mellom dei ulike brukstilfella. Denne veka vil det være endå større fokus på testing og tilbakemeldingar.
6. Under produksjonsetting skal produksjon databasen oppdaterast, koden produksjonsettast, og API bli tilgjengeliggjort.

Sjå vedlegg 10.2 GANTT diagram for ein detaljert prosjektplan.

3.2.3 Risikovurdering

For å vurdere risiko blir det sett på blant anna kva påverknad prosjektet kan ha i det eksisterande systemet ved å implementere løysinga. Vurderinga er basert på kva påverknad kvar komponent som prosjektet vil nytte seg av kan ha på resten av verdikjeden til det eksisterande systemet.

For å vurdere risiko er det nytta risikomatrise der det er rekna ut prioriteringsgrad til alle identifiserte risikoar. Prioriteringsgraden er produktet av sannsyn for at risikoen skjer og påverkningsgraden dersom risikoen oppstår. Kvar risiko har til slutt forslag til korleis ein kan handtere risikoen dersom den oppstår.

Den mest kritiske risikoen i prosjektet går ut på manglande automatisk testing i det eksisterande systemet. Dette kan forårsake feil i nye og eksisterande komponentar under utviklinga av løysinga. Prosjektet har vurdert å skrive automatiske testar for all eksisterande funksjonalitet som blir brukt i løysinga, men det var sett på som ikkje oppnåeleg i løpet av utviklingstida satt av prosjektet. For å handtere risikoen er det innført brukartesting av eksisterande funksjonalitet, regresjonstesting, i tillegg til test av ny funksjonalitet, i kvar sprint.

Sjå vedlegg 10.1 Risikoliste for ei oversikt over alle identifiserte risikoar i prosjektet.

3.3 Evalueringsplan

Enivest sine ansatte på team utbygging, som er tilbydd som testarar til prosjektet, vil være med på å teste og gi tilbakemeldingar til prosjektet undervegs i kvar sprint. Ein av testarane er og tidlegare ansatt i eit entreprenør firma som Enivest har nytta. Hos testarane føreligg det kunnskapar om alt som trengst for at nettportalen skal fungere optimalt for brukarane. All strukturert testing av løysinga vil foregå som brukartesting. Enhet- og integrasjonstesting vil også bli utført, men som manuelle testar, då Enivest ikkje har rutinar og verktøy for automatisert testing. Undervegs i sprintane vil det i tillegg bli utført regresjonstesting av eksisterande funksjonalitet for å sikre at det ikkje har blitt påverka av prosjektet. Prosjektet har i tillegg god kommunikasjon med dei andre utviklarane om eventuelle endringar som blir gjort i same komponentane som blir nytta i prosjektet.

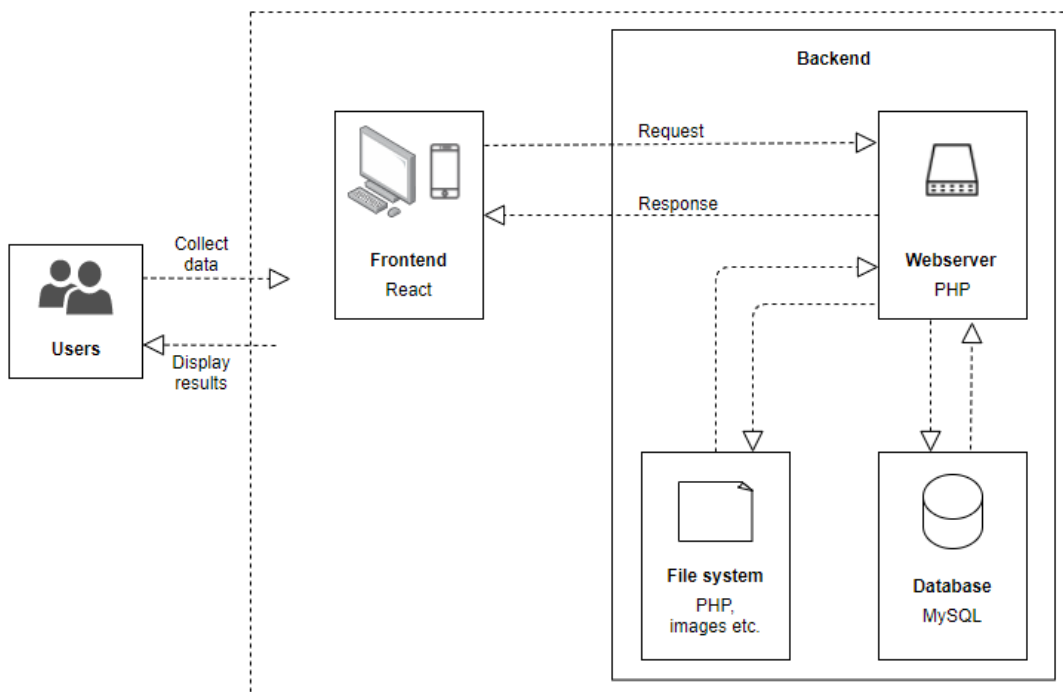
4 DETALJERT DESIGN

4.1 Introduksjon

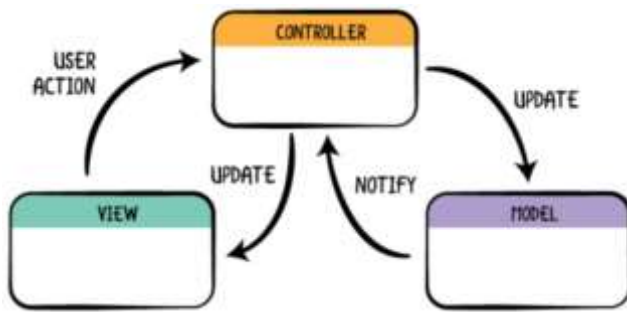
4.1.1 Arkitekturoversikt

Det eksisterande systemet til Eninvest er ein web applikasjon. Figur 5 - overordna arkitektur syner dei ulike delsystema, kva teknologi som nyttast og korleis delsystema brukar kvarandre. I brukargrensesnittet er det nytta React, og i forretningslaget er det nytta PHP som kjører på ein webserver. For lagring av data er det nytta MySQL.

For å ha klare roller for brukargrensesnittet og forretningslaget er det nytta model-view-controller mønster (Shvets, 2018). Dette mønsteret blir nytta for å skilje mellom presentasjonslogikk og forretningslogikk i applikasjonen (Fowler, 2006). Alle komponentar i forretningslaget har ein controller med funksjonar som vert utført ved bruk av modellar. For å bruke funksjonane må brukargrensesnittet sende ein request der ein ved hjelp av URL'en rutar requesten til rett controller og funksjon. Når controlleren har behandla requesten, sender den tilbake ein response med data som presentasjonslogikken brukar til å gjere endringar i visninga.



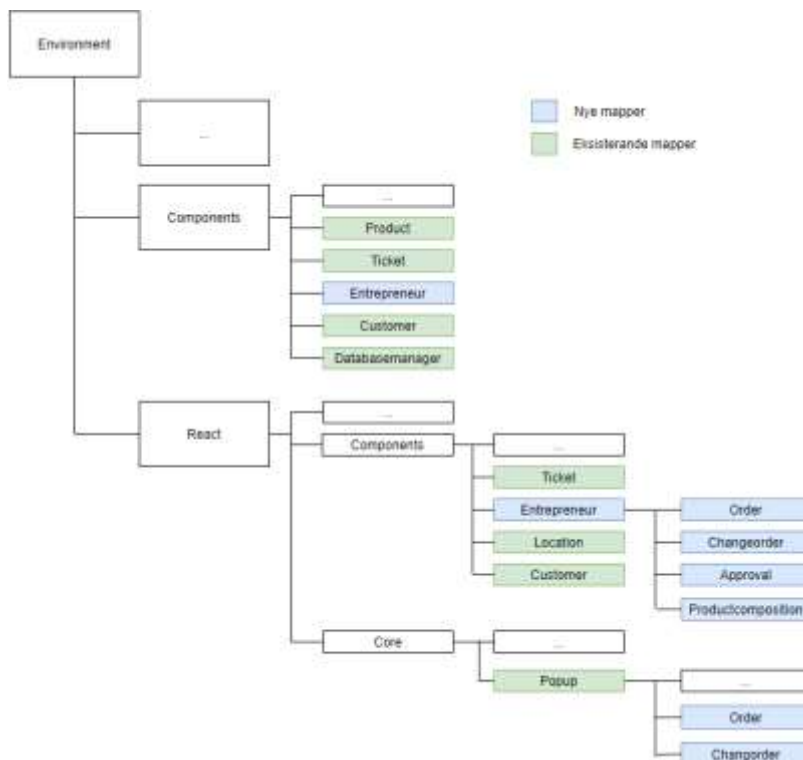
Figur 5 - overordna arkitektur



Figur 6 - model view controller

4.1.2 Mappedstruktur

Oppbygginga av kodebasen til brukargrensesnittet/frontend og forretningslaget/backend er komponentbasert. Figur 7 - mappestruktur per miljø viser korleis kvart miljø/environment sin mappestruktur er bygd opp med fokus på komponentane som vert viktig for prosjektet. Miljøet i figuren kan for eksempel være eit utviklarmiljø der ein kan modifisere kode utan å påverke produksjonsmiljøet, som er miljøet som systemet køyrer i. Frå rota i «Environment» ligg mappa «Components», som utgir forretningslaget med komponentar der kvar komponent er bygd opp med ein felles fast pakkestruktur. Nedanfor «Components» ligg mappa «React» som utgir brukargrensesnittet, der bygg ein opp komponentane ved hjelp av fleire mindre komponentar.

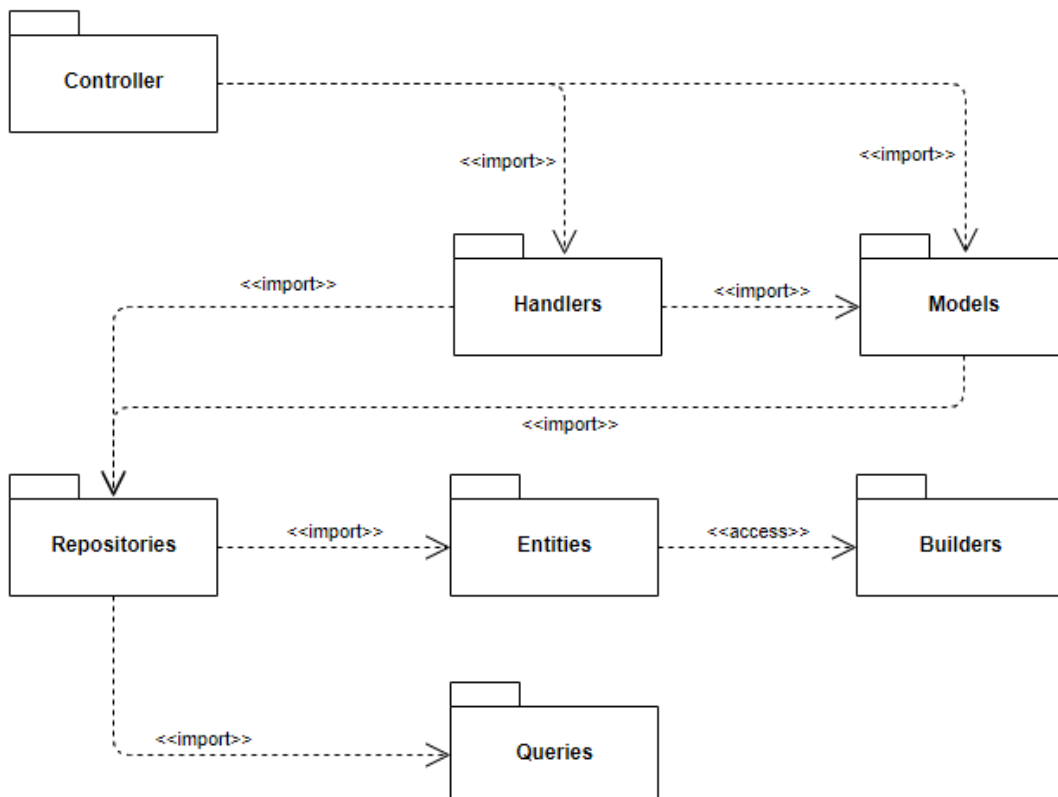


Figur 7 - mappestruktur per miljø

4.1.3 Pakkestruktur

Alle komponentane i forretningslaget har ein fast oppbygging og namngiving av pakkar som vert nytta for å ha ein oversiktleg kodebase. Individuelt for kvar komponent vert den strukturert opp med dei pakkane som vert nødvendig for å utføre dei ulike controller funksjonane.

Figur 8 - pakkediagram viser oppbygginga av pakkane for Entreprenør komponenten, og korleis dei heng saman. Ytterst ligg controlleren der alle request kjem inn. Der får ein inn eit container objekt som inneheld all dataen i requesten. Dersom container objektet har brukarinput, så går ein vidare til ein handler for å validere og uskadeleggjere dataen. Frå handler kan ein vidare til models der ein har spesifikke klassar for å utføre det controller funksjonen skal gjere. Ein kan også komme til models direkte via controlleren dersom ein ikkje har brukarinput. Frå models og handlers kan ein gå vidare til repository for å hente ut, endre, slette eller legge til data i databasen. Kvar tabell i databasen har ein tilsvarande entity som repositorien brukar for å gjere operasjonar i rett tabell. Repository kan også bruke queries for å utføre meir kompliserte spørringar. Ein entitet kan bruke ein builder for å legge på ekstra data frå andre tabellar der dei er kopla saman i databasen.



Figur 8 - pakkediagram

4.2 Brukargrensesnitt / frontend

4.2.1 Skisser til brukardialog

For å skissere opp korleis brukargrensesnittet skal sjå ut, er designverktøyet Figma nytta. I designet er det teke høgde for eit responsivt design, ved at ein skal kunne dynamisk endre på oppbygginga av skjermbildet etter oppløysning for å støtte mindre skjermar, for eksempel mobiltelefon.

I skissene er det lagt opp visning av to nye tabbar, «Bestilling» og «Endringsbestillingar» i den allereie eksisterande sakskomponenten. I «Bestilling» tabben er det tenkt å samle all informasjon ein treng for det som er aktuelt for ei bestilling. I «Endringsbestillingar» tabben finn ein historikk på alle endringsbestillingar som er motteken.

For å presentere data nyttast accordion boksar som ein kan trykke på for å vise eller skjule innhaldet i. Ved endring av data er det oppretta popup boksar som ein kan gjenbruke kor som helst i applikasjonen. For eksempel ved «Opprett/Endre ordre» knappen kjem det opp ein popup der du kan fylle inn data. Dersom ein opprettar ein ordre utanfor ein sak som ikkje har sakstype «Bestilling entreprenør», så vil det bli oppretta ei ny «Bestilling entreprenør» sak som blir åpna i ei ny fane med dataen som er fylt ut i popup. For mesteparten er popup'ane enkle skjema for å kunne endre datainnhaldet. Den mest avanserte popup'en er «Opprett/Endre produktsamansetning» der ein kan velje inn produkt, endre antall og lagre valde produkt.

4.2.1.1 Tab: Bestilling – ordre data

Skissa viser korleis visninga er når ein kjem inn i bestilling tabben. Ordre accordion'en er åpen, og siste endringsordre accordion er lukka. I bilde kan ein endre på alt, men dette vil bli avgrensa etter kva miljø ein befinn seg i, og brukaren si rolle. For eksempel kan ein i Eninvest sitt miljø endre ordre og status, medan i entreprenør miljøet kan ein endre på ordre og produksamansetning.

Name	Antall	Pris	Eining	Sum
Sum				

Figur 9 - ordre data

4.2.1.2 Tab: Bestilling – endringsordre data

Her er det skjult ordre data og henta fram endringsordre data ved å trykke på «Siste endringsordre». På same måte som ved ordre viser det her alt ein kan endre på, men blir seinare restrikkert per miljø og rolle.

The screenshot shows a web application interface for order management. On the left is a sidebar with a 'Hoved meny' section containing buttons for 'Sak #31122', 'Følgarar', 'Kunde', and 'Adresse'. The top navigation bar has tabs for 'Chat', 'E-post', 'Sms', 'Bestilling', and 'Endringsbestillingar'. The main content area displays order information for 'Sak #31122'. It shows '+ Ordre' with status 'Godkjent' and '- Siste endringsordre' with status 'Ny'. Below this are two sections: 'Endre ordre' with fields for 'Ny antatt ferdig dato:', 'Vedlegg:', and 'Beskrivelse:'; and 'Endre produktsammansetning' with a table for items. The table has columns for 'Namn', 'Antall', 'Pris', 'Eining', and 'Sum'. At the bottom of the main area, there is a 'Status: Ny' dropdown and a 'Lagre' button. The footer contains the text 'Botntekst'.

Figur 10 - endringsordre data

4.2.1.3 Tab: Endringsbestillingar

I endringsbestillingar tabben viser alle endringsordrar som har skjedd for ordren i saka. Endringsordrane blir sortert etter sist endra dato med nyaste først. Den øvste accordion er alltid åpen når ein først trykkar inn på tabben. Her vil det også være mogleg å gjere same endringane som i bestilling tabben.

The screenshot shows a web application interface for managing change orders. On the left is a sidebar with 'Hoved meny' and buttons for 'Sak #31122', 'Følgarar', 'Kunde', and 'Adresse'. The main area has navigation tabs: 'Chat', 'E-post', 'Sms', 'Bestilling', and 'Endringsbestillingar'. The 'Endringsbestillingar' tab is active, showing a list of change orders. The top order is expanded, displaying a table for 'Endre produktsamansetning' with columns: 'Name', 'Antall', 'Pris', 'Eining', and 'Sum'. The status is 'Ny' and the last change date is visible. Below the table are buttons for 'Endre ordre' and 'Endre produktsamansetning', and a 'Lagre' button. The bottom of the page has a 'Botntekst' section.

Figur 11 - endringsbestillingar

4.2.1.4 Popup: Produktsamansetning

I oppretting og endring av produktsamansetning popup er visninga delt i to. På den eine sida listar ein opp alle produkt og grupperar dei etter type, og til høgre viser ein produkta som ein har valt. Ein vel produkt ved å trykke på produkta i lista, for å fjerne produkt trykker ein på same produktet som er valt. I visninga til høgre kan ein endre antall per produkt. Summen per produkt og samanlagt sum vil oppdatere seg etter kvart som ein gjer endringar.

Endre produktsamansetning	
Graving	Utbygging
#1produkt 499,-	
#2produkt 499,-	
#3produkt 499,-	
#4produkt 499,-	
#5produkt 499,-	
#6produkt 499,-	

Hoved meny

Timar

Namn	Antall	Pris	Eining	Sum
#4produkt	1	499	Stk	499
Sum				499,-

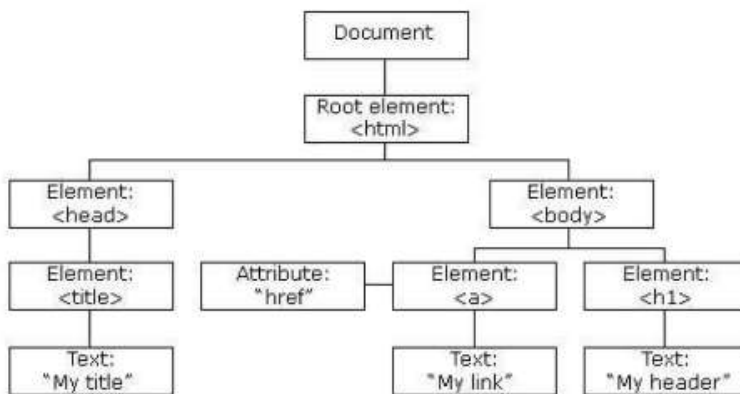
Lagre

Botntekst

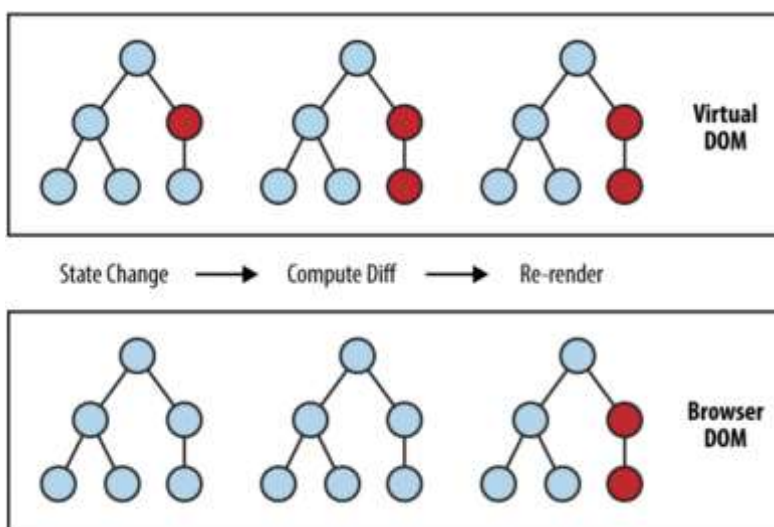
Figur 12 - produktsamansetning popup

4.2.2 Teknologi og bruk av rammeverk

I brukargrensesnittet er det nytta Javascript rammeverket React (Alttexsoft, 2020). Hovedstyrken til React er at ein kan dele funksjonalitet inn i mindre gjenbrukbare komponentar og oppdatere kun påverka komponentar ved endringar i grensesnittet. Tradisjonelt må ein oppdatere heile DOM treet når ei endring har skjedd. Som vist i Figur 14 - React oppdatering av DOM, har React ein virtuell kopi av DOM'en. Ved oppdateringar av data i applikasjonen finn React ut kva komponentar som må oppdaterast i den virtuelle DOM'en, og deretter oppdaterar kun dei som er påverka i den verkelege DOM'en. Dette fører til at oppdateringar av element går raskt og skapar eit kjapt og dynamisk UI.



Figur 13 - eksempel DOM tre



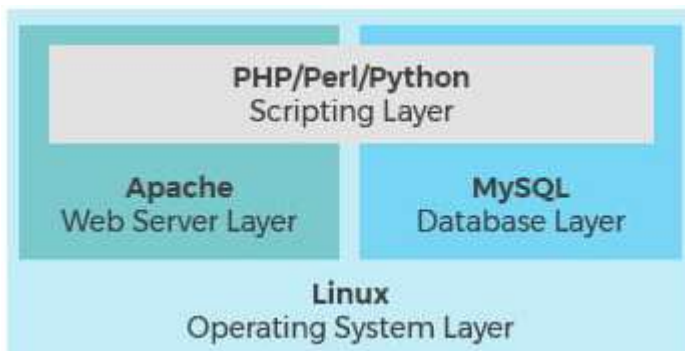
Figur 14 - React oppdatering av DOM

4.3 Forretningslag / backend

4.3.1 Teknologistack

I forretningslaget er det nytta ein teknologistack som består av Linux, Apache, PHP og MySQL (LAMP). I Figur 15 - LAMP arkitektur viser korleis dei ulike teknologiane er lagvis avhengig av kvarandre. På botnen er operativsystemet som tenar Apache, MySQL og alt av filer som for eksempel PHP filer. Apache er web serveren som har ansvar for å behandle http requests. For eksempel viss det er ei PHP fil request, så sender Apache requesten vidare til PHP som lastar fila og kjører koden i fila. På same nivå ligg MySQL som PHP kan nytte til lagring av data. På det ytterste nivået ligg PHP som er programmeringsspråket som er nytta for å utføre ønska operasjonar og gi tilbake ønska data.

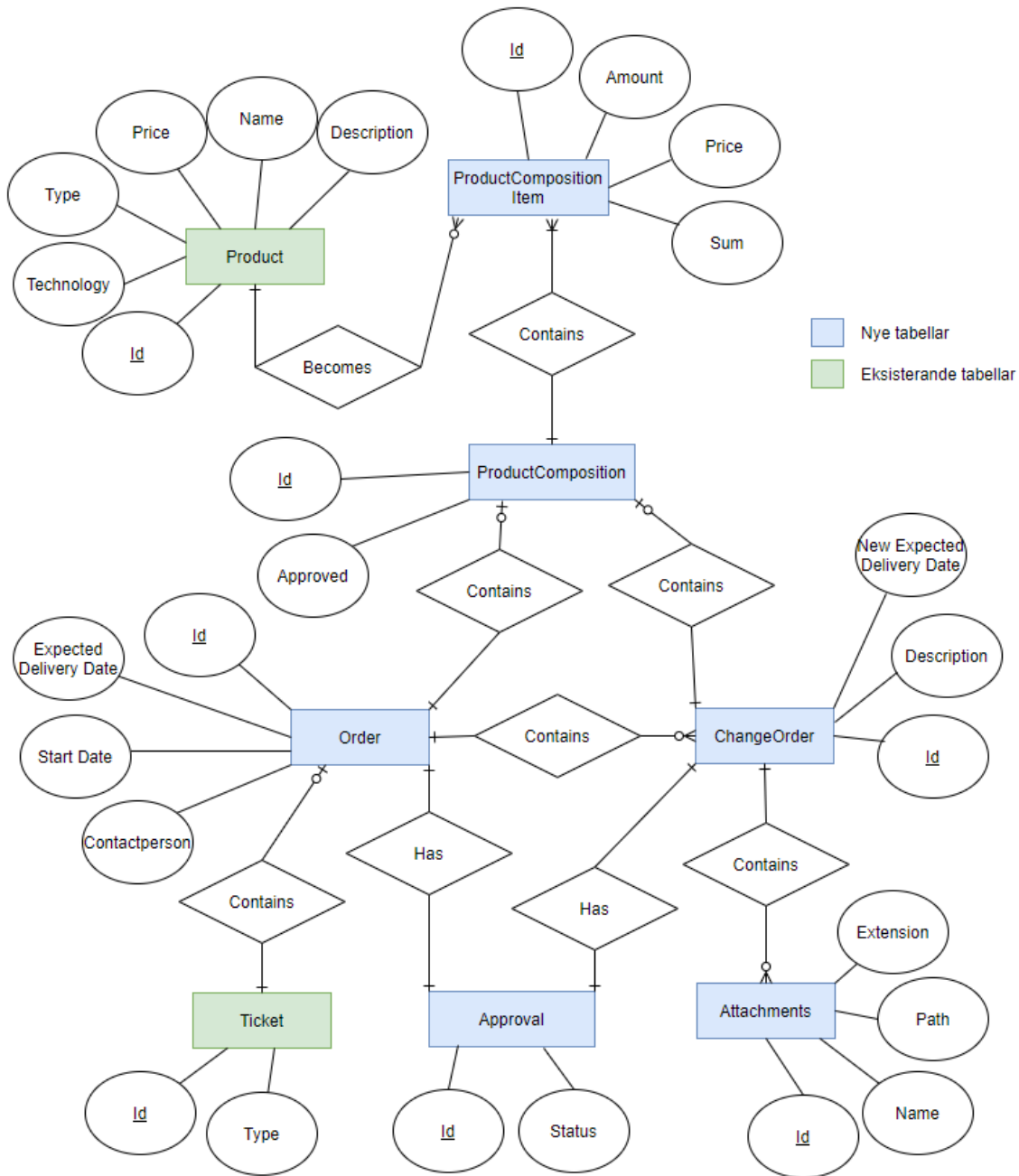
Generelt sett er LAMP ein av dei mest vanlege stackane for å kjøre open source web applikasjonar (IBM, 2019). Spesifikt for Eninvest er det sida oppstart av bedrifta alltid nytta LAMP til alle web applikasjonar som dei tenar, det er alt frå interne verktøy til heimesida for bedrifta. Det har gjort Eninvest godt kjent og komfortabel med moglegheit og drifting av ein slik teknologistack.



Figur 15 - LAMP arkitektur

4.3.2 Logisk datamodell

Figur 16 - EER diagram viser struktur over alle aktuelle tabellar og koplingar i prosjektet. Her vises nye tabellar og koplingar mot eksisterande tabellar i databasen. Det er lagt opp til at ein gjenbrukar «Approval» og «ProductComposition» i både «Order» og «ChangeOrder» for å unngå fleire like tabellar.

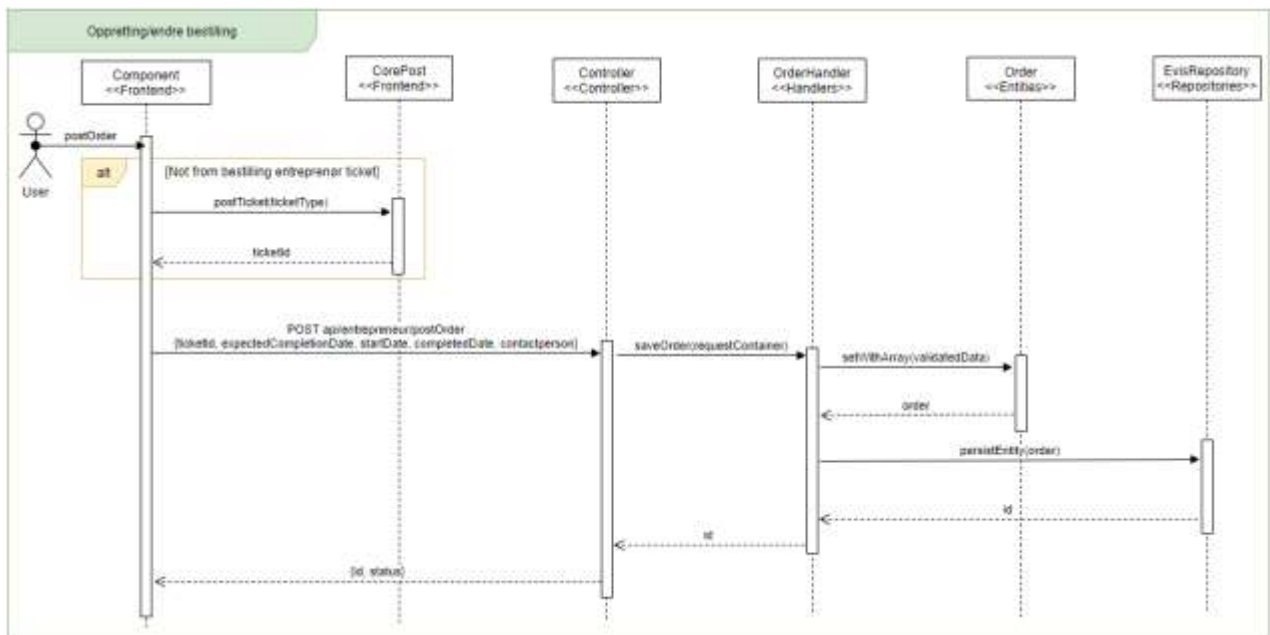


Figur 16 - EER diagram

4.3.3 Sekvensdiagram

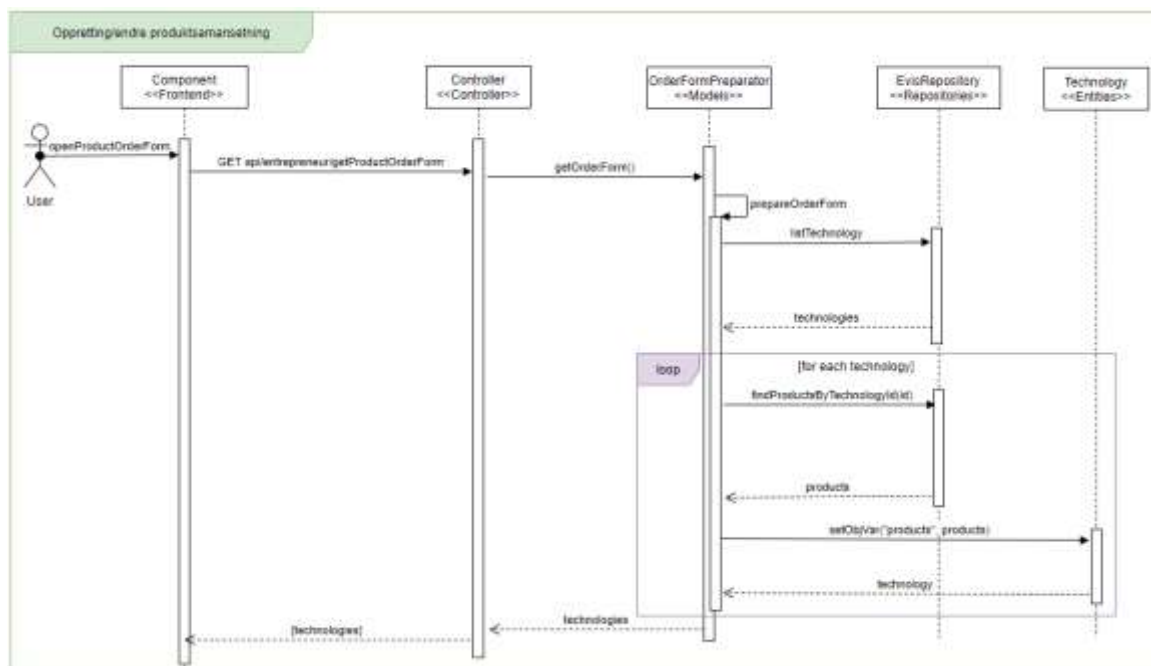
I dette avsnittet presenterast eit detaljdesign for komponentar og tenestar i løysinga ved hjelp av UML sekvensdiagram. Diagramma er utvikla og dokumentert for dei viktigaste brukstilfella.

4.3.3.1 Brukstilfelle: Opprette og endre bestilling



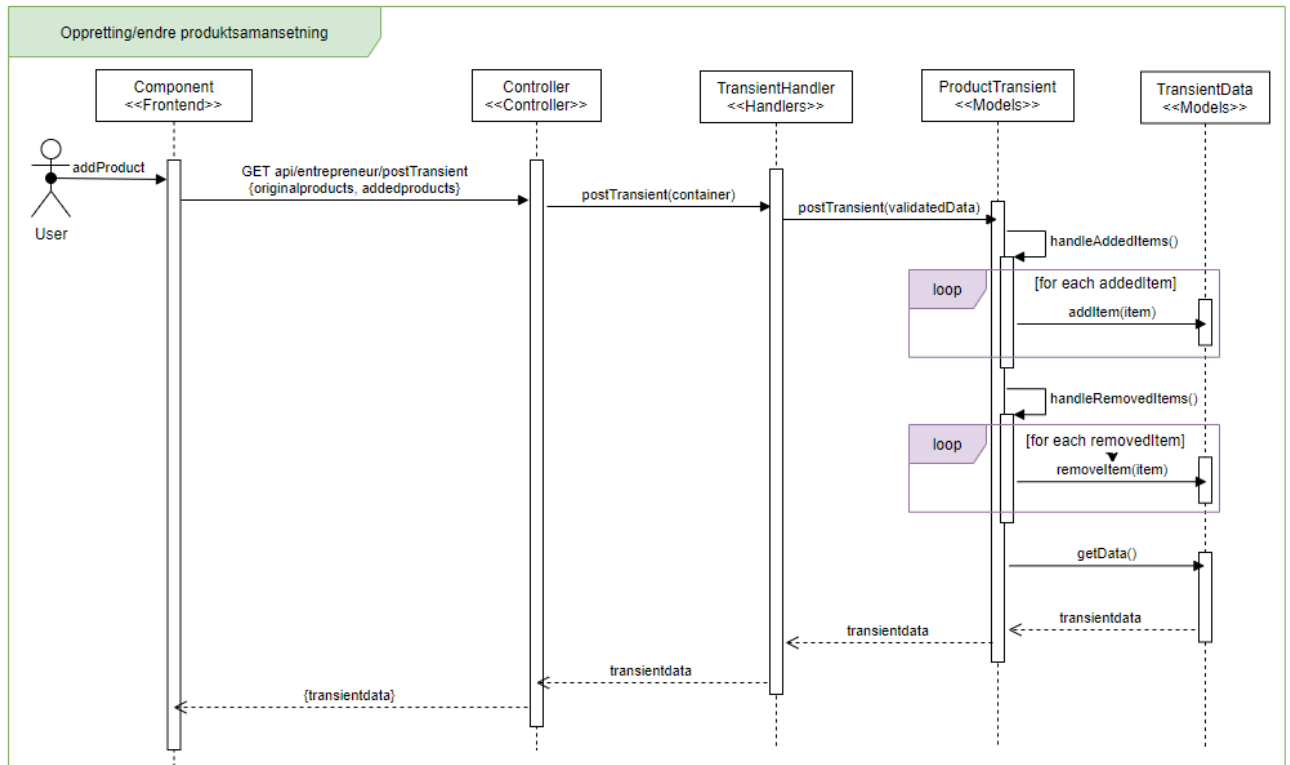
Figur 17 - sekvensdiagram for oppretting og endring av bestilling

4.3.3.2 Brukstilfelle: Opprette og endre produksamansetning – ved åpning av popup



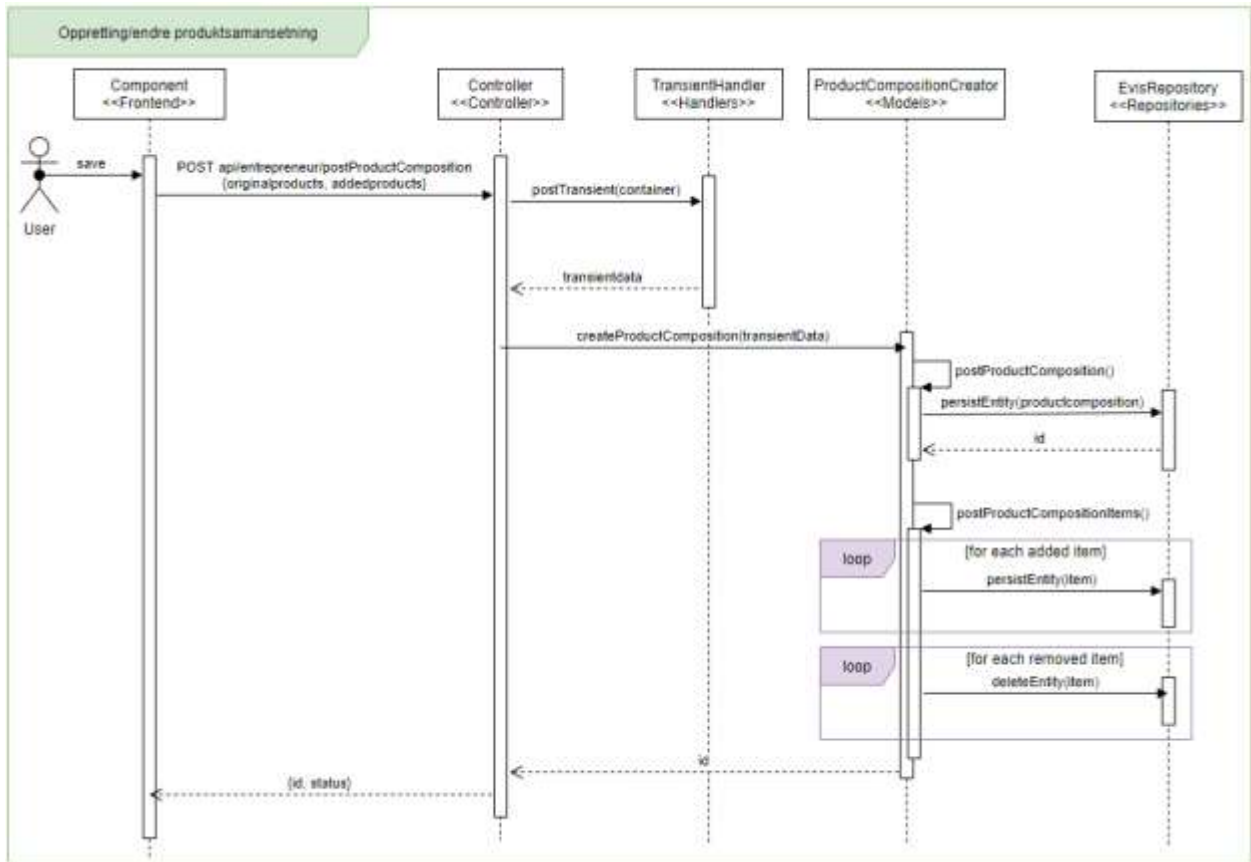
Figur 18 - sekvensdiagram ved åpning av produksamansetning popup

4.3.3.3 Brukstilfelle: Opprette og endring av produksamansetning – ved tillegging av produkt



Figur 19 - sekvensdiagram ved tillegging av produkt i produksamansetning popup

4.3.3.4 Brukstilfelle: Opprette og endring av produksamansetning – ved lagring av valte produkt



Figur 20 - sekvensdiagram ved lagring av valte produkt i produksamansetning popup

4.4 Sikkerheit

4.4.1 Brukartilgang

Alle brukarar i systemet skal opprettast av Eninvest, med brukarnamn og passord. Vidare har alle brukarane som standard to faktor autentisering der ein sender ut SMS med eingangskode som brukaren må taste inn for å fullføre innlogginga. For kvar brukar vert det definert miljø og roller. Basert på det er der reglar for kva URL og funksjonar som brukaren kan sjå og bruke. Dette blir validert i alle controller funksjonar som første steg før ein går vidare med å utføre requesten.

4.4.2 Validering av brukarinput

I alle controller funksjonar med brukarinput i requesten har ein handler klassar som ein skal bruke for å validere og uskadeleggjere dataen før ein tek den i bruk. Dette er for å forhindre ulike typar angrep som Cross Site Scripting og injections som SQL injection (OWASP, 2020). Dette oppnår ein ved å fjerne kjente teikn og taggar ved slike angrep frå brukarinputen. Dersom ein ikkje sikrar seg mot slike angrep kan ein person manipulere visninga av data på skjermen, kjøre skadelege script og eksponere data i databasen.

4.4.3 Validering av SQL spørringar

For å unngå SQL injection vidare, i tillegg til validering av brukarinput, blir alle SQL spørringar med brukarinput parameterisert (OWASP, 2020). Dette er innebygde funksjonar som alle bibliotek som handterer databaseoperasjonar har. For PHP blir det nytta eit bibliotek som heiter PDO der ein kan forberede SQL'en ved å erstatte kvar brukarinput med ein nøkkel. Ved utføring av spørringa sender ein inn både SQL'en og ein array av nøklar med tilsvarande brukarinput. PDO har deretter funksjonalitet for at det ikkje er mogleg å gjere endringar i dei parameteriserte verdiane for å gjere om på korleis spørringa vert køyrt.

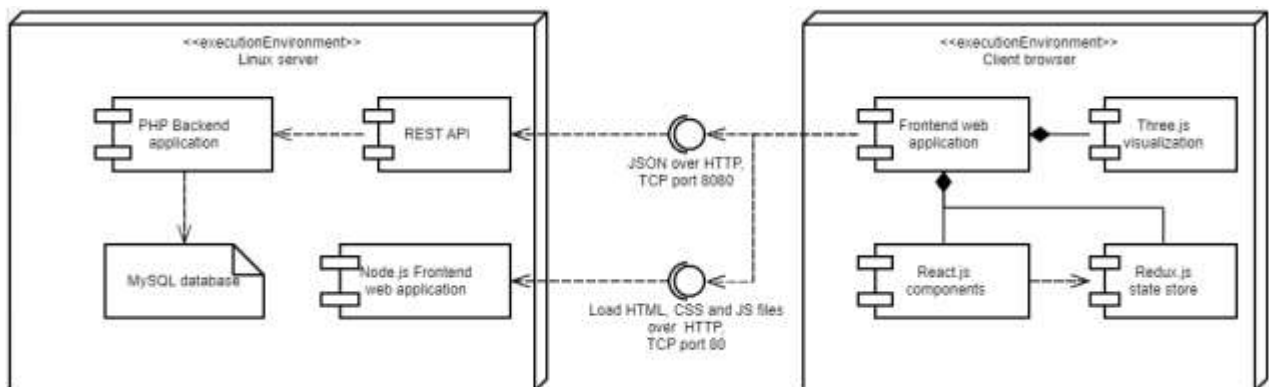
4.4.4 Token

For å sikre seg mot «man in the middle attacks» og «token sidejacking» blir det nytta JSON Web Tokens gjennom eit tredjeparts bibliotek (OWASP, 2020). Ved autentisering får klienten ein hash som identifikasjon, i tillegg får payloaden ein hash som utgir dataen i requesten. Desse to hashane

i tillegg til ein hemmelig nøkkel utgir ein token som blir validert av backend. Dette sikrar at klienten som sender requesten er den som han blei autentisert som, og at dataen i requesten ikkje har blitt endra på i etterkant. I tillegg har tokens ein varigheit slik at brukaren må autentisere seg på nytt etter så og så lang tid.

4.5 Installasjon og utrulling

Figur 21 - deployment diagram viser korleis dei ulike programvareelementa blir installert og køyrt. Enivest driftar eigne dedikerte serverar som køyrer ulike tenestar. Til venstre i figuren viser Linux serveren som systemet køyrer på, og til høgre viser klienten sin nettlesar der frontend applikasjonen kjører.



Figur 21 - deployment diagram

5 EVALUERING

5.1 Evalueringsmetode

For å evaluere løysinga er det i hovudsak nytta testing og tilbakemeldingar frå prosjekteigar og andre brukarrepresentantar i dei forskjellige fasane.

Under design aktiviteten i oppstartsfasen hadde prosjektet og prosjekteigar eit møte om den initielle løysingsideen og korleis løysinga skal konkret passe saman i systemet. Dette var sett på som eit viktig steg i dei vidare prosessane for å sikre at den eksisterande funksjonaliteten forblir intakt, og samtidig at ein kan bygge vidare på løysinga til prosjektet. Seinare i design aktiviteten vart det haldt ein workshop med prosjekteigar og ansatte på team utbygging for å gå igjennom korleis grensesnitta og prosessane er tenkt skal henge saman.

I utviklingsfasen blei det lagt opp til korte sprintar for å ha hyppige møter med prosjekteigar etter kvart som dei ulike brukstilfella vart implementert. Etersom prosjekteigar har lang erfaring som utviklar såg prosjektet på tilbakemeldingane som ein god måte å evaluere kvaliteten av arbeidet på.

I forhold til testing av løysinga har Eninvest eit fråverande forhold til automatisk testing. Det finns ikkje noko eksisterande arbeidsmetodar for å drive testdrevne utvikling, som har ført til at mesteparten av dei eksisterande komponentane ikkje har automatisk testing. Med det som utgangspunkt har ikkje prosjektet laga automatisk testar i form av UI testing eller enhetstestar. Prosjektet kunne laga det spesifikt for entreprenør komponenten, men ettersom prosjektet støttar seg på eksisterande komponentar får ein ikkje testa alt. For å få automatisk testing på plass måtte prosjektet ha skreve testar for både løysinga og alle komponentane som er involvert, men det var sett på som ikkje oppnåeleg i løpet av utviklingsperioden.

I staden for automatisk testing nytta prosjektet brukartesting gjennom utviklingsfasen. Ved brukartestinga vart det bestemt å bruke exploratory testing som metode ettersom testarane, både utviklarane og ansatte på team utbygging tatt inn som testarar, hadde gode kunnskapar om prosessane som skulle bli dekkja. Samtidig opnar dette opp for at testarane kan være meir kreative i interaksjonane med løysinga som kan føre til fleire og betre innspel til forbetringar.

Brukartestinga blei utført ved at ein først gjekk igjennom funksjonaliteten som skulle testast. Deretter fekk ein teste funksjonaliteten i vilkårlege rekkefølger etter egen logikk og erfaring. Eventuelle feil vart meldt inn fortløpande via egen funksjon i systemet som lagar oppgåver til

prosjektet i ZenHub. Til slutt vart det gått igjennom korleis testinga gjekk ved å diskutere erfaringar og eventuelle innspel til forbetringar.

Under utviklinga blei all ny funksjonalitet brukartesta av prosjektet for å kontrollere at grensesnitta oppfører seg som forventa, og at backend gjer dei rette operasjonane og sender rett data. For å kontrollere at prosjektet ikkje øydelagde for funksjonalitet i dei eksisterande komponentane, blei det utført regresjonstesting ved at eksisterande funksjonalitet blei retesta samtidig som ein testa ny funksjonalitet. Testing av eksisterande funksjonalitet blei gjort på basis av tidlegare erfaringar av korleis komponentane skal fungere.

I tillegg til prosjektet si brukartesting, brukte prosjektet også Eninvest ansatte til å brukarteste og gi tilbakemeldingar i utviklingsfasen. Prosjektet følgde med på eventuelle feil som var meldt inn, og fiksa dei fortløpande. Etter gjennomgåing av kvar test periode, gjekk prosjektet igjennom innspel og implementerte det som var gjennomførbart. Prosjektet brukte testarane si tilbakemelding til å evaluere funksjonalitet og brukaropplevinga til løysinga.

5.2 Evalueringresultat

I starten av design aktiviteten fekk prosjektet god nytte av dialogen med prosjekteigar. Det førte til at strukturen bak løysinga vart forbered på å kunne vidareutviklast med meir planlagt funksjonalitet i seinare versjonar. Samtidig fekk dei viktigaste prosessane, som vist i sekvensdiagramma, ein gjennomgang i forhold til avhengigheit og ansvarsfordeling mellom klassane, som også er eit viktig moment i å kunne vidareutvikle. I workshopen fekk prosjektet tidleg tilbakemeldingar frå brukarane på korleis løysinga er designa. Etter innspel i workshopen klarte prosjektet å treffe godt på brukargrensesnitt og logikken i løysinga under implementeringa, og førte til mindre behov for endringar i sprintane.

Ved brukartestinga som prosjektet utførte fann ein at alle funksjonar og logikkar i frontend og backend oppførte seg som forventa utan nemneverdige feil. I bestillingskjema for produktsamansetningar er der forbetringspotensiale i koden for å forbetre tidsbruken systemet brukar frå ein trykker på eit produkt til det kjem opp i handlekurven. Dette vart oppdaga i etterkant at ein kunne forenkla datastrukturane på dataen som flyttar seg mellom frontend og backend for å gjere operasjonane raskare. Prosjektet fekk ikkje nokon kommentarar på det frå brukarane, og ser på det som at det fungerer godt nok.

Ved brukartestinga som ansatte i Eninvest utførte vart det innrapportert feil som del av testinga, men ikkje fleire enn at det var mogleg å rette dei som del av utviklingsfasen. Innspel til forbetringar gjekk ut på visning og namngiving av datafelt, dette var mindre endringar som blei sett på som gjennomførbare. Det vart nemnt betydelege tidsbesparingar i erfaringane deira under testinga i samanlikning med korleis dei jobbar per i dag. Etter siste sprinten var brukarane/testarane fornøgd med funksjonaliteten som var levert av prosjektet, og gav samtykke for produksjonsetting.

6 RESULTATER

Prosjektet kom i mål med å implementere alle brukstilfella. Resultatet av det som er nyutvikla er definerte skjemaer og datastrukturer for å holde på dataen som skal inngå i prosessane mellom Eninvest og entreprenørane. Vidare er det teken i bruk eksisterande funksjonalitet i systemet som blant anna automatisk utsending av e-post og varsling av brukarane.

6.1 Tab: Bestilling - bestilling

Tidlegare vart dataen i bestilling ført inn som laustekst i ein e-post. Produktsamansetninga vart laga i ei Excel-fil, og blei enten limt inn som eit skjermbilde i e-posten eller lagt ved som vedlegg. Ved endringar av denne dataen under oppstart av bestillinga måtte ein sende e-post fram og tilbake med endringar. I den utvikla løysinga får entreprenørane tilsendt ein automatisk e-post med lenke til saken når Eninvest opprettar den. Entreprenørane kan så trykke på lenka og registrere eller endre dataen i dette bildet. Eninvest ansatte som følger saken, typisk den som oppretta den, får varsel når entreprenør har gjort endringar. Eninvest kan godkjenne eller avslå bestillinga som sender ein automatisk e-post til entreprenøren om at dei kan enten starte på jobben, godkjent, eller må endre på bestillinga, avslått. Dersom den blir godkjent vert også bestillingsdataen låst mot å gjere vidare endringar. Einaste måten å gjere endringar i bestillingsdataen er for entreprenøren å legge inn ei endringsbestilling der dei legg inn ny forventa ferdig dato og eventuelle nye produkt.

Kategori	Bestilling	Utbbygging	Fris	Anskaff	Eining	Sum
Type	Småttbestilling 071407H	1-Rørkanale jord (2 ubet)			1 95	
Start	1. mai 2021	Gjæving			1 95	
Forventa ferdig	7. juni 2021	28-Graevning i løst 40 cm			1 95	
Ferdig		Arbeid			1 95	
		TSP/berarbeid skjøtt 1 filar			1 95	
		V1 ODF munnstykke			1 95	
		Sum				

Figur 22 - Resultat tab: bestilling

6.2 Tab: Bestilling – endringsbestilling

Tidlegare vart endringsbestillingane fylt ut i ei Excel-fil og sendt som vedlegg via e-post. I same trend som bestilling måtte ein sende e-postar med oppdaterte vedlegg ved endringar, og ein mista fort kontroll på kva som var siste endringane. I løysinga kan entreprenørane opprette endringsbestillingar i saken under tabben endringsbestillingar. Her legg dei inn blant anna ny forventa ferdig dato, ved utsetting av leveranse, og ny produksamansetning, ved auka kostnad. Ved oppretting eller endring av endringsbestilling får Enivest ansatte som følger saken varsling. Enivest kan godkjenne eller avslå endringsbestillinga som igjen sender ein automatisk e-post til entreprenøren. I tillegg vil ein ved godkjent overskrive forventa leveringsdato og slå saman valte produkt i produksamansetninga inn i bestillingsdataen.

The screenshot displays a software interface for managing orders. On the left, a sidebar shows order details for 'Entreprenør Bestilling - Ferdigevegen 3'. The main area shows a list of orders with two entries: a 'Bestilling' (Status: Godkjent) and a 'Sole endringsbestilling' (Status: Avslått). Below this, there are two tables: 'Endre produksamansetning' and 'Endre produktamansetning'. The 'Endre produksamansetning' table has columns for 'Graving', 'Pris', 'Antall', 'Eining', and 'Sum'. The 'Endre produktamansetning' table has columns for 'Type', 'Pris', 'Antall', 'Eining', and 'Sum'. At the bottom, there is a 'Status' bar with 'Avslått' and 'Begrunding' fields, and a 'Lagre' button.

Figur 23 - Resultat tab: endringsbestilling

6.3 Tab: Endringsbestillingar

Tidlegare hadde ein ikkje ei god oversikt over alle endringsbestillingane som var sendt inn av entreprenør. Dei hamna blant andre e-postar der ein måtte bruke tid på å finne att ei gitt endringsbestilling, og ikkje minst siste versjonen av den. I løysinga er det ein eigen tab som viser alle endringsbestillingar som er registrert i kronologisk rekkefølge med den siste først. Her har entreprenørane også moglegheit til å opprette ny endringsbestilling.

The screenshot shows a web application interface for 'Endringsbestillingar' (Change Orders). The interface is divided into several sections:

- Left Sidebar:** Contains filters for 'Status' (oppn), 'Prioritet' (høgaste), 'Tilgangsrätt' (alle), 'Avdeling' (interiør), and 'Sakstype' (interiørrenovasjon). It also shows 'Oppretta av' (Knutth Kaurane), 'Oppretta' (1 juni 2021, kl. 15:08:18), and 'Sist endra' (1 juni 2021, kl. 15:08:18).
- Main Content Area:** Displays a list of change orders. The selected one is 'Endringsbestilling' with status 'Avsluttet', description 'Begravning av belysningsanlegg', and 'Sist oppdatert: 2021-06-01 15:47:57'. Below this, there is a table of items with columns for 'Navn', 'Prio', 'Antall', 'Eining', and 'Sum'. The table shows two items: '36 Rigg grøntprodukt' and '80 Tenserts monteringsbord'. A 'Sum' row is also present.
- Bottom Section:** Shows the status 'Avsluttet' and description 'Begravning av belysningsanlegg' for another change order, with 'Sist oppdatert: 2021-06-01 15:44:44'.

Figur 24 - Resultat tab: endringsbestilling

6.4 Popup: Produktsamansetning

Tidlegare vart produkta som utgav kostnaden for bestillinga ført inn i ei excel-fil og sendt via e-post. Ved endringar fekk ein same problematikk som påpeika før med leiting i e-post og versjonering. I løysinga er det oppretta ein type bestillingskjema der entreprenørane får opp alle produkta som ein kan trykke på for å legge til eller fjerne i høgre margin. Vidare i høgre margin kan ein endre antall per produkt, før ein eventuelt vel å lagre produktsamansetninga. Den vil deretter kople seg på bestillinga eller endringsbestillinga, alt etter kor ein opna skjemaet. Ved endringar vert tidlegare valte produkt lasta inn i skjemaet når ein opnar skjemaet.

Graving	Arbeid	Id	Graving	Pris	Antall	Eining	Sum
25:Graving i las-masse 40cm overdekkning	64:Fiberarbeid etablering G4	36	Rigg graveprodukt		1	Stk	
26:Graving i las-masse 60cm overdekkning	65:Fiberarbeid etablering G12		Time				
27:Microtrenching	66:Fiberarbeid etablering G24	93	Tirepris monterarbeid		8	Timar	
28:Graving i asfalt 40 cm	67:Fiberarbeid etablering G48		Sum				
29:Graving i asfalt 60 cm	68:Fiberarbeid etablering G96		Lagre produktsamansetning				
30:Håndgraving	69:Fiberarbeid etablering G144						
31:Punkt graving	70:Fiberarbeid etablering G192						
32:Nedsetting av 1-lokks betongkum	71:Fiberarbeid etablering G216						
33:Nedsetting av 1-lokks komposittkum	72:Fiberarbeid etablering G288						
34:Nedsetting av 2-lokks betongkum	73:Fiberarbeid skjøting 1 fiber						
35:Nedsetting av 3-lokks betongkum	74:Fiberarbeid skjøting av opp til 4 fiber						
36:Rigg graveprodukt	75:Fiberarbeid skjøting av opp til 12 fiber						
37:Rigg punkt graving	76:Fiberarbeid skjøting av opp til 24 fiber						
38:Pigging	77:Fiberarbeid skjøting av opp til 48 fiber						
39:Oppsetting av fiberskap pr. stk	78:Fiberarbeid skjøting av opp til 96 fiber						
40:Rigg trench	79:Fiberarbeid skjøting av opp til 144 fiber						
41:Grunnrei-vgrunnrettsavklaring	80:Fiberarbeid skjøting av opp til 196 fiber						
42:Skiltplan	81:Fiberarbeid skjøting av opp til 216 fiber						
43:Blåsing av kabel pr. meter G1 t.o.m. G24	82:Fiberarbeid skjøting av opp til 288 fiber						
44:Blåsing av kabel pr. meter G48 t.o.m. G288	83:Opphenging av luftkabel G4 t.o.m. G96						
45:Subling av microtrå 8 t.o.m. 14 mm rår pr. meter	84:Opphenging av luftkabel G144 t.o.m. G288						
46:Subling av microtrå 20 mm rår pr. meter	85:Etablering av nedføring						
47:Asfaltering pr. kvadratmeter t.o.m. 5 cm	86:OTDR måling						
48:Asfaltering pr. kvadratmeter t.o.m. 6 t.o.m. 10 cm	87:Montasje stolpe						
Drift	88:Utsifting av stolpe						
97:Garantiarbeid	89:Skogrydding						

Figur 25 – Resultat popup: produktsamansetning

7 DISKUSJON

Prosjektet klarte å løyse bruken av nye og eksisterande komponentar i Eninvest sitt system utan å påverke eksisterande funksjonalitet. Ei merkbar utfordring for prosjektet var den manglande automatiske testinga i dei eksisterande komponentane.

Grunna mangel på automatisk testing brukte prosjektet meir tid i sprintane på brukartesting. Det er vanskeleg å teste alle moglege utfall ved exploratory testing ettersom det blei også brukt til mangel for enhetstesting og integrasjonstesting. Dersom prosjektet hadde hatt meir utviklingstid ville det ha blitt prioritert å lage automatiske testar for dei eksisterande komponentane, og vidare i dei nye komponentane som blei utvikla.

Prosjektet sine erfaringar kring nyutvikling i det eksisterande systemet utan automatiske testar er spelt vidare til Eninvest. Ettersom løysinga skal bli vidareutvikla på eit seinare tidspunkt, har prosjektet tilråda Eninvest om å få på plass eit godt test regime.

8 KONKLUSJON OG VIDERE ARBEID

Målsettinga til prosjektet var å strukturere og effektivisere prosessane mellom Eninvest og entreprenørar, gjennom betra informasjonsflyt og letta kommunikasjon, ved hjelp av ein ny nettportal.

Prosjektet kom i mål med brukstilfella som var planlagt å implementere. Brukarane er blitt godt kjent med korleis dei skal bruke løysinga for å gjere jobben sin. Løysinga er klar til å bli teken inn i produksjon, men det er avventa på grunn av at superbrukarane skal opprette nye arbeidsrutinar for å ta i bruk komponenten. For å sjå eksakt i kor stor grad løysinga har forbetra arbeidsprosessane, må dette målast når komponenten vert teken i bruk. Etter testinga som blei utført, var inntrykket frå testarane at løysinga kjem til å effektivisere prosessane deira og føre til store tidssparingar i arbeidsdagen, både for Eninvest og entreprenørane. Det kan også forventast at kundeopplevinga vert forbetra ettersom ein vil raskare få informasjon frå entreprenørane, og i kontrollerte dataformat.

Utan å ha handfaste data på i kor stor grad løysinga har forbetra informasjonsflyten og letta kommunikasjonen, så kan det tenkast seg til ved å samanlikne tidlegare prosessar opp mot resultatet av prosjektet at ein har i stor grad forbetra informasjonsflyten. Noko som tilbakemeldingane frå testarane kan også tyde på.

Eninvest har planlagt vidare versjonar av komponenten der dei skal legge på meir funksjonalitet. Prosjektet har tatt høgde for dette og Eninvest ser positivt på moglegheitane som er lagt fram til at dei kan vidareutvikle løysinga.

9 REFERANSAR

Alttexsoft, 2020. *The good and the bad of React*. [Internett]

Tilgjengeleg på: <https://www.altexsoft.com/blog/engineering/the-good-and-the-bad-of-reactjs-and-react-native/>

[Funnen 30 03 2021].

Baghel, A. S., 2018. *Software design principles DRY and KISS*. [Internett]

Tilgjengeleg på: <https://dzone.com/articles/software-design-principles-dry-and-kiss>

[Funnen 23 03 2021].

Cprime, Ukjent. *What is Agile what is Scrum*. [Internett]

Tilgjengeleg på: <https://www.cprime.com/resources/what-is-agile-what-is-scrum/>

[Funnen 23 03 2021].

Fowler, M., 2006. *GUI Architectures*. [Internett]

Tilgjengeleg på: <https://martinfowler.com/eaDev/uiArchs.html>

[Funnen 23 03 2021].

Fowler, M., 2006. *Separated Presentation*. [Internett]

Tilgjengeleg på: <https://martinfowler.com/eaDev/SeparatedPresentation.html>

[Funnen 23 03 2021].

Guru99, ukjent. *What is SDLC or waterfall model*. [Internett]

Tilgjengeleg på: <https://www.guru99.com/what-is-sdlc-or-waterfall-model.html>

[Funnen 24 03 2021].

IBM, 2019. *Lamp stack explained*. [Internett]

Tilgjengeleg på: <https://www.ibm.com/cloud/learn/lamp-stack-explained>

[Funnen 01 04 2021].

OWASP, 2020. *Cross Site Scripting Prevention*. [Internett]

Tilgjengeleg på:

[https://cheatsheetseries.owasp.org/cheatsheets/Cross Site Scripting Prevention Cheat Sheet.htm](https://cheatsheetseries.owasp.org/cheatsheets/Cross_Site_Scripting_Prevention_Cheat_Sheet.htm)

1

[Funnen 02 04 2021].

OWASP, 2020. *Injection Prevention*. [Internett]

Tilgjengeleg på:

https://cheatsheetseries.owasp.org/cheatsheets/Injection_Prevention_Cheat_Sheet.html

[Funnen 02 04 2021].

OWASP, 2020. *JSON Web Token*. [Internett]

Tilgjengeleg på:

https://cheatsheetseries.owasp.org/cheatsheets/JSON_Web_Token_for_Java_Cheat_Sheet.html

[Funnen 02 04 2021].

Ravichandran, A., 2018. *React Virtual Dom Explained*. [Internett]

Tilgjengeleg på: <https://programmingwithmosh.com/react/react-virtual-dom-explained/>

[Funnen 30 03 2021].

Shvets, A., 2018. *Design patterns*. [Internett]

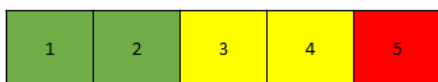
Tilgjengeleg på: <https://refactoring.guru/design-patterns>

[Funnen 24 03 2021].

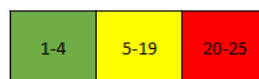
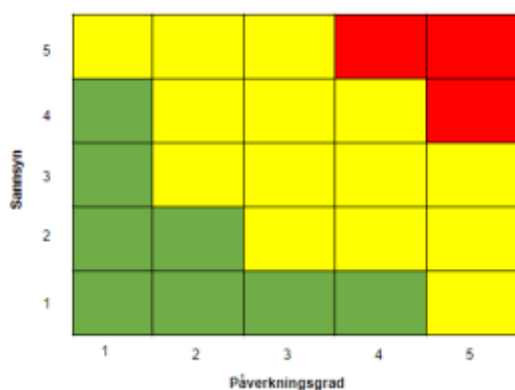
10 Vedlegg

10.1 Risikoliste

Sannsyn / påverkningsgrad skala



Prioriteringsgrad heatmap



Risiko	Sannsyn	Påverkningsgrad	Prioriteringsgrad	Handtering
Sakskomponent sluttar å fungere i eksisterande system	2	5	10	Må sjå på kva som kræskar med eksisterande funksjonalitet. Vurdere tiltak for å tilpasse begge verdenar.
Produktstruktur er ikkje god nok til å lage produksamansetning	2	4	8	Må sjå på manglande logikk for å få det til å fungere, enten via å utvide eksisterande produktstruktur eller lage eigen struktur.
Kundestruktur er ikkje god nok til å halde på informasjon om entreprenør	2	4	8	Må sjå på å utvide kundestrukturen med ein substruktur for leverandørar.
Manglande automatisk testing kan forårsake feil i nye og eksisterande komponentar	4	5	20	Det må dokumenterast kva den nye funksjonaliteten påverkar så ein kan brukarteste all funksjonaliteten den bærer under utvikling og før produksjonsetting
Tap av data	2	5	10	Det blir teken nattlege databasekopiar for å minske skade ved tap av data.
Tap av kode	3	5	15	Må committe ofte og ha god kontroll på kva branch ein er aktiv på ved bruk av git funksjonalitetar.
Manglande moglegheit for personleg oppmøte i planlegging og diskusjonar	5	2	10	Må vurderast med gjeldande smittetilak. Kan muligens ha personleg oppmøte ved redusert antall deltagarar. Ellers nytte zoom til møteverksamheit.

10.2 GANTT diagram

