



IS-304: 2024

Tittel: Defcon

Emnekode	IS-304
Emnenavn	Bacheloroppgave i informasjonssystemer
Emneansvarlig:	Hallgeir Nilsen & Geir Inge Hausvik
Veileder	Sofie Wass
Oppdragsgiver:	Phonero: Bård Eik

Studenter:

Etternavn	Fornavn
Andersen	Simon Duarte
Aspelien	Fredrik
Jahren	Pål Folmer
Johansen	Markus
Kristiansen	Tobias Kroknes
Monsen	Martin Hommeland

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

Abstrakt

Denne bacheloroppgaven tar for prosjektet Defcon som er gjort av gruppe 11 for Phonero. Målet med oppgaven har vært å utvikle systemet Defcon som har til hensikt å erstatte det tidligere systemet Data Quality Monitor. I løpet av prosjektet har Scrum-rammeverket blitt benyttet for å sikre en agil arbeidsmetode som fremmer fremdrift og kvalitet. Resultatene av denne rapporten viser til en ferdigstilt MVP basert på MoSCoW, og gruppen har gjennom prosjektet med en menneskesentrert designprosess reflektert over, og gjennomført prosjektstyring og produktutvikling som igjen har bidratt til en omfattende læringsprosess. Oppgaven belyser fordelene ved agil arbeidsmetode og menneskesentrert design, samtidig som den viser viktigheten av å følge de etablerte prinsipper på en nøye og konstruktiv måte. Denne tilnærmingen er essensiell for å sikre kvalitet og effektivitet i prosjektet. På denne måten har gruppen oppnådd tilfredsstillelse blant produkteier og ansatte samtidig som målene for prosjektet er blitt nådd.

Forord

Vår takk går til Phonero, som tok oss imot med åpne armer. Vi har fått låne utstyr, brukt deres lokaler og hatt tilgang til gratis lunsj hele vårsemesteret. Takk til Erik Linga Bertsen, Kristian Hagestad, Frode Berge-Eriksen og Sunniva Finne som har stilt opp og vært tilgjengelig til å bistå med både tips og hjelp i prosjektet.

Spesielt takk til Bård Eik som har vært produkteier og oppdragsgiver. Bård har gjennom hele semesteret vært en ressurs som vi har hatt mulighet til å lene oss på for råd og veiledning. Dette har vi satt stor pris på. Avslutningsvis ønsker vi å takke vår veileder Sofie Wass som har kommet med tilbakemeldinger og konstruktive tilbakemeldinger når det har vært behov for det. Rammene har med andre ord lagt til rette for at vi skulle gjøre det så godt som mulig.

Innholdsfortegnelse

Abstrakt.....	1
Forord.....	1
Innholdsfortegnelse.....	2
Figurliste.....	4
1. Introduksjon.....	6
1.2 Phonero.....	6
1.3 DataQualityMonitor.....	7
1.3 MVP.....	8
1.4 Teamet.....	9
1.5 Mål og ambisjoner.....	9
1.6 Om arbeidsplassen.....	10
2. Sentrale beslutninger og kvalitetssikring.....	11
2.1 Kvalitet.....	11
2.1.1 The Iron Triangle.....	11
2.2 Prosjektstyring og prosesskvalitet.....	12
2.2.1 Scrum.....	12
Scrum master.....	13
Sprint planning.....	13
Daily Scrum.....	14
Sprint review.....	14
Sprint retrospektiv.....	14
2.2.2 Risikoanalyse.....	14
2.2.3 MoSCoW.....	15
2.3 Design og kvalitet.....	15
2.3.1 Design “Thinking”, Designprosess & PACT.....	16
Personas, User stories & Use cases.....	17
2.3.2 Idémyldring.....	18
2.3.3 Idéforedling.....	18
2.3.4 Prototype.....	18
2.3.5 Brukertesting.....	19
2.4 Verktøy.....	20
2.4.1 Frontendteknologier.....	21
TypeScript.....	21
React.....	21
Tailwind.....	21
Shadcn/ui.....	22
2.4.2 Backendteknologier.....	22
C# ASP.NET Core.....	22
Hangfire.....	22

Microsoft SQL Server.....	22
2.4.3 Andre verktøy.....	23
Jira.....	23
ChatGPT.....	23
Figma.....	23
Visual Studio Code / Visual Studio.....	24
3. Gjennomføring av prosjektet.....	24
3.1 Kvalitet.....	25
3.2 Analyser.....	25
3.2.1 PACT.....	25
People.....	25
Activities.....	25
Context.....	26
Technologies.....	26
3.2.2 MoSCoW.....	26
3.2.3 Risikoanalyse.....	28
3.3 Scrum.....	30
3.3.1 Implementering av The Iron Triangle.....	30
3.3.2 Scrum master.....	31
3.3.3 Sprint planning.....	31
3.3.4 Daily Scrum.....	33
3.3.5 Sprint review.....	34
3.3.6 Sprint retrospektiv.....	36
3.4 Design.....	37
3.4.1 Designprosessen.....	38
3.4.2 Felles grunnlag med shaden/ui og Tailwind.....	39
3.4.3 Wireframes og analyser.....	40
3.4.4 Prototype.....	42
3.5 Testing.....	43
4. Produktet.....	46
4.1 Defcon.....	46
4.2 Overlevering.....	49
5. Refleksjon og lærdom.....	50
5.1 Erfaringer fra prosjektet.....	50
5.2 Konklusjon.....	51
6. Litteraturliste.....	52
Figurliste.....	55
7. Vedlegg.....	56
7.1 Uttalelse fra Phonero.....	56
7.2 ChatGPT.....	57
7.3 Gruppeevaluering/individuelle bidrag.....	57

Fredrik Aspelien.....	57
Markus Johansen.....	57
Martin Hommeland Monsen.....	57
Pål Folmer Jahren.....	58
Simon Duarte Andersen.....	58
Tobias Kroknes Kristiansen.....	58
7.4 Scenarioer til brukertesting.....	59
7.5 System Usability Scale.....	59
7.6 Brukertesting tilbakemeldinger.....	62
7.7 Low Fidelity Wireframes.....	63
7.8 High Fidelity Wireframes.....	64
7.9 Defcon logo.....	69
7.10 Personas.....	70
7.11 Risikoanalyse.....	72
7.12 PACT.....	75
7.13 Prototype i Figma.....	75
7.14 Systemarkitektur.....	76
7.15 Alternativt bilde til Designprosesen.....	77

Figurliste

Figur 1: <i>DQM Sensurert versjon</i>	8
Figur 2: <i>The Iron Triangle</i>	12
Figur 3: <i>MoSCoW siste utkast</i>	27
Figur 4: <i>Risikoanalyse</i>	29
Figur 5: <i>Backlog</i>	32
Figur 6: <i>Fibonacci-sekvensen</i>	32
Figur 7: <i>Originale pushbuttons konsept</i>	35
Figur 8: <i>Ferdigstilte pushbuttons</i>	35
Figur 9: <i>Start, stop, continue</i>	36
Figur 10: <i>“Design Thinking”</i>	37
Figur 11: <i>Persona Michelle, IT</i>	38
Figur 12: <i>Dashboard Low Fidelity Wireframe</i>	40
Figur 13: <i>Dashboard High Fidelity Wireframe</i>	41

Figur 14: <i>Brukertest “System Usability Scale” første ukast</i>	44
Figur 15: <i>Brukertest “System Usability Scale” andre ukast</i>	45
Figur 16: <i>Funksjonsliste</i>	47
Figur 17: <i>Defcon Dashboard</i>	47
Figur 18: <i>Defcon Job View</i>	48

Defcon

1. Introduksjon

Rapporten er produsert av gruppe 11 som en del av den avsluttende bacheloroppgaven i IT- og Informasjonssystemer ved UiA. Gruppen har gjennom hele det siste semesteret utarbeidet et system for Phonero som har til hensikt å forbedre og videreutvikle et allerede eksisterende verktøy i bedriften.

Kapittel én tar for seg en introduksjon av prosjektet i tillegg til en kort oppsummering av gruppen, dens mål og ambisjoner og arbeidsplassen som vi har hatt gjennom semesteret. Kapittel to tar for seg sentrale beslutninger rundt arbeidsmetoder, verktøy og kvalitet, med fokus på en teoretisk tilnærming til ulike begreper. Kapittel tre forklarer gjennomføringen av prosjektet, og hvordan gruppen har anvendt teoretiske rammeverk i prosessen. I tillegg til dette er det skrevet om refleksjoner som gruppen kontinuerlig har tatt for seg gjennom prosjektets løp. Kapittel fire omhandler selve sluttproduktet og overleveringsprosessen. Avslutningsvis er kapittel fem en konklusjon. Kapitlene er delt opp i flere underkapitler for å sørge for en oversiktlig struktur på rapporten.

1.2 Phonero

Phonero er en mobiltelefonibedrift som er med på å skape enkle og rimelige priser for norske bedrifter og offentlig sektor. De ble grunnlagt i 2008 og har i nyere tid blitt kjøpt opp av Telia. Phonero er veldig opptatt av gode kundeopplevelser og leverer i dag tjenester til både Forsvaret og Lånekassen (Phonero, u.å.-a). Noen av de tjenestene som Phonero leverer er trådløst bredbånd, mobilforsikring og Telia sitt 5- og 4G-nettverk gjennom sitt mobilabonnement (Phonero, u.å.-b). Phonero har også flere avdelinger blant annet Salg, IT, Billing, Økonomi og HR som arbeider på kontoret deres i Kristiansand sentrum, som tilsvarer rundt 160 ansatte.. De gruppen i hovedsak har samhandlet med er IT og Billing.

Gjennom dette halvåret har vi fått kontakt med flere av de som senere vil bli omtalt i oppgaven som produkteier og ansatte. Produsenter har hjulpet oss fra starten og skapt en overordnet trygghet. Ansatte i Billing- og IT-avdelingen har vært med på å gjøre prosesser enklere og har vært disponible til å hjelpe oss på veien. De har også tatt seg tid til å være med på ideutvikling rundt prosjektet og gitt oss tips om arbeidsmetodikk og rammeverk, samt brukertesting underveis og avslutningsvis.

1.3 DataQualityMonitor

DataQualityMonitor (DQM) er et internt verktøy utviklet av Phonero som brukes til å overvåke kvaliteten og integriteten på data som Phonero lagrer i sine databaser. Dataen kan være alt fra kundedata til interne applikasjonsdata som Phonero er avhengig av. Dersom et datasett inneholder ugyldig data, kan det settes opp til at DQM utløser en alarm i andre overvåkningsverktøy som Phonero bruker, og feilen kan bli håndtert. I tillegg til å sjekke integriteten på data, bruker de også DQM til å hente ut statistikk av de forskjellige datasettene de er interessert i. Et eksempel kan være hvor mange aktive kunder de har. Alarmer og statistikk blir kalt “scripts” og er Standard Query Language (SQL) spørringer som blir kjørt regelmessig. Verktøyet blir hovedsakelig brukt av IT-avdelingen og Billing-avdelingen.

DQM er et gammelt og utdatert system som ifølge utviklere hos Phonero ble laget i 2008. Fordi det er et internt verktøy, har ikke utforming av brukergrensesnittet blitt prioritert. Applikasjonen bærer preg av eldre teknologi og sentimentet om at det er et internt verktøy, som bare har til hensikt å fungere. Phonero hadde et ønske om at gruppa skulle oppdatere applikasjonen slik at den utnytter nye teknologier som støttes av de oppdaterte plattformene til Phonero, og at brukerperspektivet skal være mer i senter slik at applikasjonen både fungerer bra, og er behagelig å bruke. Under er det et eksempel på hvordan DQM ser ut pr. dags dato.



Figur 1: DQM Sensurert versjon

1.3 MVP

Etter innledende samtaler med Phonero om prosjektet og gjennomgangen av DQM hadde gruppen et utgangspunkt for å arbeide med produktutvikling. Phoneros ansatte hadde et behov for en tidsestimert oversikt over milepæler for prosjektløpet. De ønsket å være disponible med ressurser i perioder der de visste vi jobbet med komplekse utfordringer og konsepter. Gruppen diskuterte både internt, og sammen med produkteier hvor vi ble enige om felles målsettinger og satt opp en generell oversikt over milepæler. Ytterligere ble det fastslått og lagt frem et forslag til MVP (minimum viable product) av de mest sentrale funksjonene i applikasjonen. MVP'en skulle være et utgangspunkt for å kvalitetssikre produktet. Ved å inkludere tilbakemeldinger fra brukere underveis, samt i overleveringsfasen kunne vi sikre at forventninger og krav var blitt innfridd.

Grunnlaget for funksjoner, arkitektur og design av MVP'en ligger i analysene og samtalene gjort hos Phonero som vil bli tydeliggjort i rapporten. MVP'en «Defcon» ble satt som det overordnede suksesskriteriet og Scrum målet for bachelorprosjektet, se kapittel 4.2 overlevering.

1.4 Teamet

Gruppa består av 6 medlemmer hvor alle har jobbet sammen store deler av studieløpet. Dette har gjort at medlemmene er godt kjent med hverandres kompetanse, som igjen har ført til at planlegging og strukturering av arbeid ikke er like tidkrevende og usikkert som i andre settinger hvor gruppesammensetningen er en annen. Inngangen til oppgaven var åpen med tanke på arbeidsfordeling, men etter diverse kursing, prøving og feiling innledende ble det en naturlig arbeidsfordeling fordi noen satt seg ned med frontend samtidig som andre fokuserte på backend. Dette førte til at samtlige gruppemedlemmer fikk muligheten til å jobbe selvstendig samtidig som gruppemedlemmene hele tiden spilte på hverandre. Dette var mulig til dels på grunn av den fysiske konteksten vi befant oss i. Gruppen jobbet sammen hver dag på Phonero og hadde derfor tilgang på hverandre konstant. I tillegg til dette var de ulike scrum-møtene en viktig bidragsyter i kompetanseutveksling og sparring. Summen av kunnskapen gruppen besittet fra tidligere gjorde at arbeidsflyten var jevn. Rollen som Scrum-master gikk på rundgang i takt med sprinter, og alle har i stor grad fått erfaring som prosjektleder. Med dette sagt er det viktig å poengtere at det gjennom prosjektet har vært lav terskel for innspill og tilbakemeldinger, noe som har gjort beslutningstaking og vurderinger til en dynamisk prosess som alle til enhver tid har tatt del i, selv om Scrum master har hatt det siste ordet og den tyngste stemmen i vurderingsprosesser.

1.5 Mål og ambisjoner

Gruppa har gjennom de to siste semestrene hatt som mål å gjøre bacheloroppgaven sammen da alle gruppemedlemmene har en felles forståelse av ambisjonene knyttet til oppgaver. Det var et ønske om å utvikle et produkt som skulle bli brukt av en oppdragsgiver, og dette var mye grunnen til at gruppen hadde lyst til å gjøre oppgaven hos Phonero. Dette så vi for oss at skulle bidra til økt eierskap til bacheloroppgaven, noe som i retrospekt stemte veldig godt. Som et minimum var det full enighet om at gruppa skulle oppnå et MVP. I tillegg til dette var det en klar motivasjonsfaktor å tilegne seg arbeidserfaring, og ambisjonsnivået til gruppa har vært betydelig gjennom semesteret, noe som gjenspeiler seg i intens arbeidsinnsats hele veien.

Utover målene som går på karakterer og kvalitet på prosjektet har et sekundærmål vært å lære så mye som mulig, samtidig som gruppa ønsket å anvende kunnskapen som har blitt opparbeidet gjennom de tidligere semestrene. Alle gruppemedlemmene har vært motiverte til å opparbeide seg så mye kunnskap og erfaring som overhodet mulig. Å forbedre kunnskapen om koding i tillegg til drift av prosjekt og prosjektledelse har vært helt sentralt for gruppa hele veien, noe som gjenspeiler seg i den bratte læringskurven som samtlige gruppemedlemmer har kjent på.

1.6 Om arbeidsplassen

Å ha gjennomført et bachelorprosjekt hos Phonero er noe gruppa har møtt med stor entusiasme. Da gruppa fikk avklart at det var hos Phonero vi skulle være, oppstod spørsmålet om vi kunne sitte på kontoret. Det er noe Phonero var veldig positiv til og sammen med den andre gruppen fikk vi vårt eget rom i lokalet deres. Første dagen på arbeidsplassen fikk vi en omvisning og ble vist til der gruppa skulle sitte. Her var det allerede gjort klart med stoler, pulter, skjermer og datamaskiner som vi fikk låne av bedriften Phonero.

Gruppa ble veldig godt tatt imot og det ble gjort klart at vi bare måtte stille spørsmål om vi trengte hjelp med noe. Derav ble det også veldig lett å komme i kontakt med dyktige ansatte om gruppa hadde behov for det. I tillegg til den profesjonelle støtten, ble gruppen gledelig overrasket da vi fikk tilbudt gratis lunsj som en del av hverdagen på kontoret. Dette satte gruppa stor pris på og det ga oss ekstra motivasjon til å være på kontoret så ofte som mulig.

2. Sentrale beslutninger og kvalitetssikring

Dette kapittelet tar for seg de mest sentrale beslutningene som har blitt gjort for å sikre kvaliteten til prosjektet.

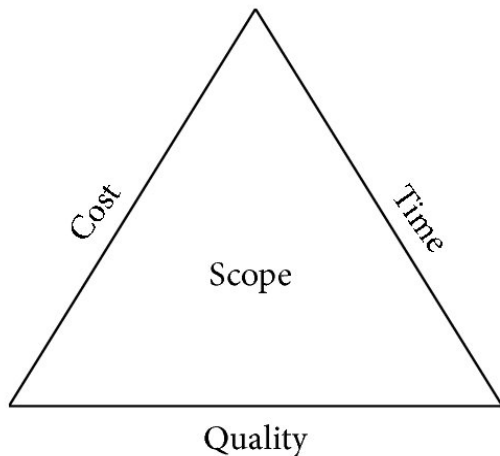
2.1 Kvalitet

Innledningsvis fant gruppa det nyttig å komme fram til en felles forståelse av kvalitet sett i lys av prosjektet som skulle leveres. Dette ble gjort fordi kvalitet i mange sammenhenger kan sees på som relativt. Elassy (2015) forklarer at konseptet kvalitet har eksistert like lenge som mennesker, og at det finnes et hav av litteratur som tar fatt på begrepet i et forsøk på å definere det. Det kan derfor argumenteres for at det er en underdrivelse å si at det finnes mange ulike definisjoner på kvalitet. Martin et al. (2020) presenterer i sin artikkel ulike definisjoner på kvalitet. De vektlegger at forståelsen av kvalitet er sterkt tilknyttet hvilken setting det skal anvendes. En av de mest fremtredende definisjonene på kvalitet kalles “*Quality-as-agreed*”. Denne tilnærmingen til kvalitet går ut på at kvaliteten på et produkt baserer seg på de standardene og kravene som blir stilt fra kunden. Ifølge denne definisjonen er det helt sentralt at produktet som leveres skal tilfredsstille kvalitetskriteriene som er lagt til grunn av kunden slik at kvaliteten samsvarer med forventningene (Martin et al., 2020). Med utgangspunkt i prosjektet kom gruppen fram til at kvalitet handler om at produktet skal innfri de kravene og standardene som er påkrevd fra oppdragsgiveren.

2.1.1 The Iron Triangle

Gjennom arbeidet med prosjektet har gruppen underveis stilt seg følgende spørsmål: “Lager vi det riktige produktet?”, “Lager vi produktet på riktig måte?” og “Holder vi oss innenfor avtalte tidsfrister?”. I arbeidet for å etterfølge disse problemstillingene ble det enighet om å utarbeide en kvalitetsplan. For å innføre og gjennomføre kvalitetsplanen fant gruppa det nyttig å lene seg på “The Iron Triangle”, som er forklart av Pollack et al. (2018). Begrepet omfatter tid, kost og kvalitet, og selv om gruppa i mindre grad var opptatt av kostnadene relatert til prosjektet var det allikevel et naturlig begrep å forholde seg til i det store bildet ettersom gruppa forholdt seg til kost som tid. Pollack et al. (2018) beskriver konseptet som en måte å forstå suksessen til et prosjekt. Dette går ut på at et prosjekt skal leveres innen en gitt tidsramme og innenfor et budsjett samtidig som det holder et avtalt kvalitetsnivå. Triangelet blir ofte sett på som en av de mest vanlige måtene å måle om et prosjekt er vellykket eller

ikke. De tre delene av triangelet henger sammen, og Pollack et al. (2018) forklarer videre at dersom det skulle være endringer i en av de tre delene vil dette kunne være med å påvirke én eller flere deler. Dersom tidsfrister ikke blir opprettholdt vil dette i praksis bety at det kan påvirke både kvaliteten på prosjektet, så vel som kostnadene. Dersom en misforstår eller feilberegner en av de tre vil det kunne resultere i at prosjektet blir kompromittert og i verste fall ikke blir gjennomført (Pollack et al. (2018)).



Figur 2: The Iron Triangle

2.2 Prosjektstyring og prosesskvalitet

Gruppen bestemte seg tidlig for å rette seg etter et eksisterende prosjektrammeverk som samtlige gruppemedlemmer var kjent med gjennom tidligere prosjekter. Grunnen til dette var for å kvalitetssikre arbeidsprosessen under prosjektet. Oppdragsgiver og veileder hadde tips og anbefalinger til valg av rammeverk, som var med å påvirke valget av Scrum som arbeidsmetodikk. Dette blir videre begrunnet i kapittel 3.

2.2.1 Scrum

Scrum er et rammeverk som legger opp til kontinuerlig læring av tidligere sprinter ved å reflektere rundt hva som fungerte og hva som ikke fungerte. Det som blir resultatet av refleksjonen kan videre bli tatt inn i betraktning gjennom kommende sprinter for å kunne styrke arbeidsprosessen mot et produkt (Drumond, u.å.). Videre valgte gruppen å gjennomføre en agil Scrum-prosess. En agil arbeidsmetode ved bruk av Scrum-rammeverket kan forklares som en metode som legger til rette for at et prosjekt kan deles opp i mindre deler, nemlig sprinter. Til forskjell fra en fossefallmetodikk hvor et produkt blir igangsatt,

utviklet og levert, legger den agile strukturen til rette for en iterativ prosess hvor målet med de ulike sprintene er å prioritere det som blir ansett som mest prekært først. Dette gjøres blant annet for å etterstrebe å kunne ha noe å vise til i sprint reviews med produkteier og ansatte. En annen grunn til å drive et prosjekt på en agil måte er for å få kontinuerlige tilbakemeldinger fra produkteier og ansatte som igjen fører til en kontinuerlig kvalitetssikring (Peek, 2023). De ulike aspektene ved den agile Scrum-metoden vil bli forklart videre i dette kapittelet.

Scrum master

Scrum master er en rolle som har ansvar for tilretteleggelse av Scrum metodikk. En Scrum master blir ofte omtalt som en “trener” istedenfor en prosjektleder på bakgrunn av at han leder gjennom veiledning i motsetning til en prosjektleder som leder gjennom å styre arbeidsprosessen. En Scrum master har flere interne ansvarsområder som blant annet gjennomføring av daily standup, sprint review, sprint retrospektiv og andre administrative oppgaver som hindrer gruppen i utføre ulike arbeidsoppgaver (Rehkopf, u.å.).

Sprint planning

En sprint syklus starter ved å gjennomføre en sprint planning. Her samles samtlige gruppemedlemmer i lag med produkteier for å diskutere hva hovedmålene for sprinten skal være gjennom å sette sprint mål. Før gruppen kan begynne å sette ulike mål er det nødvendig å bli enige om hvor lenge sprinten skal vare. Videre kan utførelsen av en god sprint planning bidra til å holde gruppemedlemmer motiverte og fokuserte gjennom prosessen mot målet. (West, u.å.). Videre er det hensiktsmessig å kunne lene seg på estimering av hvor krevende oppgaver er. Gruppen valgte å ta utgangspunkt i Fibonacci-sekvensen. Fibonacci-sekvensen i forhold til Scrum er en måte å beskrive omfanget av arbeidet som skal gjøres, for eksempel relatert til en oppgave. Summen av et nummer i sekvensen er summen av de to numrene som kom før. Dette vil altså si at sekvensen går som følger: 0, 1, 1, 2, 3, 5, 8, 13, 21 og så videre (Kathuria, u.å.). Det ble enighet i gruppen om å ta utgangspunkt i en skala som strakk seg fra null til 21. Grunnen til dette var for å avgrense omfanget av oppgaver slik at de ikke skulle bli for store. Mer om dette vil bli beskrevet i kapittel 3, under sprint planning.

Daily Scrum

Daily Scrum er et kort daglig møte som har til hensikt å fokusere på fremdrift i forhold til sprint-målet som er satt for sprinten. Dette gjøres ved å legge en plan for dagen som fører til effektivitet og økt selvstyring (Schwaber, u.å. -a). Møtet blir også brukt til å snakke om utførelsen av arbeidsoppgaver og hva som eventuelt stopper den enkelte fra å utføre dem, samtidig som det er mulighet for å ta opp ulike temaer man vil diskutere med resterende gruppemedlemmer. Resultatet av en effektiv daily Scrum kan være at alle gruppemedlemmer føler seg mer oppdatert og kalibrert gjennom prosessen (Radigan, u.å. -a). Daily Scrum har blitt brukt hyppig av gruppen for å sikre kvalitet gjennom arbeidsdagen, samtidig som at det er en hyggelig og produktiv start på dagen.

Sprint review

Sprint review blir gjort for å markere slutten på en sprint. Her inviteres produkteier til en gjennomgang av hva som har blitt produsert under sprinten. Videre er det viktig for gruppemedlemmer å initiere til diskusjon for å finne ut av hva produkteier syntes om det som har blitt produsert slik at man kan gjøre justeringer dersom det ikke står i stil med det som ble diskutert under sprint planningen. (Radigan, u.å. -b)

Sprint retrospektiv

Sprint retrospektiv er en avsluttende sprint evaluering som handler om hvordan gruppemedlemmene har arbeidet under sprinten. Under retrospektiv fikk gruppen mulighet til å reflektere rundt hvilke prosesser gruppen var fornøyd eller misfornøyd med. Videre ble resultatet av retrospektivet tatt i betraktning inn i fremtidige sprinter (Schwaber, u.å. -b)

2.2.2 Risikoanalyse

Risikoanalyse utgjør en essensiell prosess innenfor prosjektledelse, som systematisk identifiserer og analyserer mulige hendelser som kan forårsake uønskede konsekvenser for prosjektet. Denne prosessen er nødvendig for å informere gruppen om mulige trusler og deres eventuelle skadeomfang. Sentralt i analysen er identifikasjon av bestemte hendelser som kan skyldes menneskelige feil, systemsvikt eller eksterne faktorer utenfor prosjektets kontroll. Videre involverer analysen en vurdering av konsekvensene til hver identifisert risikohendelse, samt alvorlighetsgraden til disse konsekvensene. (Yasar, u.å.)

Gjennom analysen fikk gruppen en felles forståelse for hvordan mulige trusler skal bli håndtert. Fordelen ligger i hvordan det bidrar til å øke bevisstheten om mulige scenarioer som kan oppstå i løpet av prosjektgjennomføringen, hvilke tiltak som må tas for at det ikke skal skje, og hvilke tiltak som må tas dersom det skulle skje. Videre muliggjør analysen en effektiv allokering av ressurser ved å sette fokus på tiltak som bidrar til å minimere tap av tid og ressurser om en hendelse skulle forekomme. (Yasar, u.å.)

2.2.3 MoSCoW

MoSCoW-analyse er en metode innen prosjektledelse, som benyttes for å prioritere krav og funksjoner i prosjektet. Analysen gjør at prosessen med å etablere tydelige mål og oppnå felles forståelse for gruppen blir enklere. Dette gjør at man best mulig kan bruke ressursene på det som er mest hensiktsmessig i henhold til prosjektets behov. MoSCoW står for “Must have”, “Should have”, “Could have” og “Won’t have”, som representerer de fire prioriteringsnivåene i analysen (Køster, 2022)

Hensikten med analysen er å kartlegge, prioritere og rangere alle funksjoner og aspekter av systemet som skal implementeres, samt å avgjøre hva som skal utelates (Køster, 2022). De høyeste prioriteringsnivåene i analysen, “Must have” og “Should have”, er hva gruppen setter som grunnlag for hvilke funksjoner som innebærer et minimum viable product (MVP). Ved å tydeliggjøre hvilke prioriteringer alle funksjoner har, kan gruppen bedre konsentrere seg om kjernefunksjonaliteten og å oppnå målsetninger innenfor de gitte tidsrammene og ressursbegrensningene. Dette mener gruppa resulterer i en mer kvalitetsdrevet utviklingsprosess som er mer i tråd med brukerens behov og vårt kompetansenivå. (Benyon, 2019, s.148-149).

2.3 Design og kvalitet

Gjennom utdrag og prinsipper fra Benyon (2019) sin bok om menneske-maskin interaksjon og UX design skal gruppen i dette kapittelet skrive om hvorfor vi valgte å vektlegge en "menneskesentrert" designprosess (s.21-22). En "menneskesentrert" designprosess krever at designerne involverer og gjør seg godt kjent med brukergruppen, noe som kan kreve mye ressurser med tanke på tid og kostnader i “The Iron Triangle”, men ofte en viktig faktor for å kvalitetssikre datainnsamling og sluttproduktet (Benyon, 2019, s.21-22). Videre i rapporten

skal gruppen reflektere om hvorfor ulike prosesser, rammeverk og metoder har blitt valgt og inkludert i bachelorprosjektet.

2.3.1 Design “Thinking”, Designprosess & PACT

Design “thinking” blir beskrevet som en sentral del av menneskesentrert designprosess (IDEO 2024). I boken av Benyon (2019, s.47-77) er metoden representert gjennom å løse brukerbehovet ut ifra iterative prosesser som inkluderer ideutvikling, prototyping, evaluering og testing. Denne metodikken er sentral i «menneskesentrert UX (User experience) designprosess», og er et utgangspunkt for å utvikle brukervennlige, universelt utformede og mer behagelige interaksjoner med produkter/applikasjoner (Benyon, 2019, s.43).

Ifølge designprosessen til Benyon. D, anbefales det å følge PACT rammeverket og se nærmere på fire hovedelementer: People, Activities, Context and Technologies. PACT skal hjelpe UX designere å systematisere og få et tydelig inntrykk av produktet som utvikles (Benyon, 2019, s.25-44). I PACT-analysen skal UX designeren studere hvem brukerne er, hvilke aktiviteter de utfører, i hvilken kontekst og med hvilke teknologier som er involvert (Benyon, 2019, s.26-42).

I bachelorprosjektet har designprosessen vært med å styrke produktet. Design “thinking” og PACT-analysen har bidratt til å oppnå et tydelig bilde på kvalitetssikring av brukerperspektivet i design- og utviklingsprosessene. Det ble tydelig at det var behov for å ta i bruk et strukturert rammeverk for å forstå interaksjoner mellom mennesker og systemer som kom av prosjektbeskrivelsen til Phonero. I tillegg ble Phonero en viktig medspiller i designprosessen da deltagende design var essensielt i gruppas tilfelle. Med deltagende design ble Phonero en del av designteamet som gjorde at gruppen fikk et unikt innsyn i perspektiver og funksjonalitet på applikasjonen. Gjennom erfaring ble det tydelig at deltagende design ga merverdi for produktet ved tilbakemeldinger fra ansatte i Billing og IT. Gruppen så fordelene av å kvalitetssikre produktet ved å inkludere brukerperspektivet så sentralt (Sandnes, 2022, s.272). En av fordelene erfart gjennom deltagende design var et høyere nivå av tilfredsstillelse. Samtidig så la deltagende design til rette for økt innovasjon ved å bruke ansattes perspektiver og ideer til å løse design og funksjonelle utfordringer. (IDF, 2017).

Gruppen hadde også bekymringer for at PACT-analysen som et større rammeverk kunne skape utfordringer i planleggingsdelene av bachelorprosjektet. PACT kan være en kilde til å finne kompleksitet utover den innledende prosjektbeskrivelsen og forskyve estimert tidsbruk.

Samtidig kan det også skape splittelser i gruppen om hva resultatet av analysen faktisk er, og hvordan det skal vektlegges i videre utvikling. Likevel var det tydelige fordeler ved PACT, som var helt sentrale for prosjektet. Gruppen vektla verdien av PACT, spesielt det å skape en felles forståelse om forventninger, krav og begrensninger som følger av produktet. PACT ville også gi tyngde og trygghet i avgjørelser som ble gjort med tanke på diskusjoner om funksjoner, tjenester og design, der produkteier eller ansatte fra Phonero hadde motstridende ønsker eller behov. Noe som ble diskutert var hvordan enkelte funksjoner kan gå utover sikkerheten og om det da var lurt å implementere dem.

Personas, User stories & Use cases

Personas, user stories og use cases er viktige verktøy i design “thinking” og kan forbedre det analytiske arbeidet fra PACT rammeverket. Personas er fiktive karakterer som representerer demografi, verdier, behov og smertepunkter fra enkelte brukergrupper av produktet (Benyon, 2019, s.55-56) Å utvikle personas kan være med på å styrke forståelsen av brukere som blir etablert i "People" delen av PACT og gruppen kan lettere gjenkjenne og personifisere behov og smertepunkter i designprosessen.

User stories og Use cases knytter seg tett til Activities ved PACT. User stories beskriver en generell bruk av produktet og skrives med naturlig språk (Benyon, 2019, s.67-69). Historien som fortelles gjøres fra et brukerperspektiv og gir et innsyn i verdiskapning med produktet. Use cases vil være med å gi et strukturert bilde på interaksjonen mellom menneske og system og beskriver steg for steg gjennomgang og et forventet resultat (Benyon, 2019, s.65-66).

Det ble utarbeidet og jobbet kontinuerlig med personas, user stories og use cases. Dette tilførte prosjektet en tydelig struktur og grunnlag for diskusjon med fokus på design "thinking". Den analytiske verdien og gjenbruksmulighetene av disse verktøyene i Scrum sprintene var med å forsterke og personifisere behovene som kommer gjennom design og testing av produktet. Samtidig ble det tydelig at det var to sentrale brukergrupper som hadde ulike behov hos Phonero, noe som var avgjørende for å inkludere både avdeling "Billing" og "IT" i designprosessen. Dette grunnlaget gjorde at gruppen fikk et tydelig bilde på hvor skoen trykket hos de ansatte i Phonero som senere skulle ta over og ta i bruk produktet etter overleveringsfasen av bachelorprosjektet.

2.3.2 Idémyldring

I den tidlige designfasen er det vanlig å ha mange ideer i henhold til hva prosjektet dreier seg om. Det kan derfor være lurt å sette seg sammen som en gruppe for å komme opp med alle ideene man har for så å luke ut de mindre relevante senere. Dette blir også kalt for idémyldring. Sandnes (2022) forklarer at myldringsprosessen dreier seg om å lufte ut alle ideene som teamet har i hodet og det er flere ulike måter å gjøre det på (Sandnes, 2022, s.265). Blant annet legger Sandnes vekt på at det kan lønne seg å gjøre idémyldring i grupper siden de aller fleste har forskjellige tanker om hvordan noe kan løses, og andre sine ideer kan være med å inspirere til å tenke annerledes. Dette bygger også videre på det faktum at det er mange måter og metoder å komme på nye ideer på, men det er ingen fasit. Derav kan man ta i bruk de metodene som selv passer gruppa best (Sandnes, 2022, s.266). I idémyldringsfasen er det viktig å prioritere kvantitet over kvalitet. Ingen ideer er dumme, så det bygger heller på at man skal få ut sine tanker gjennom skissering eller muntlig deltakelse (Sandnes, 2022, s.267). Basert på hva idémyldring kan gi, så valgte gruppa å lene seg på dette designkriteriet. Det kan være verdifullt å høre flere ideer og få tatt det sammen i plenum slik at alle kan dra inspirasjon fra hverandre.

2.3.3 Idéforedling

For å ta i bruk ideene som ble produsert i idémyldringsprosessen på en effektiv måte, så er det viktig å gå videre til idéforedling. Hovedpoenget her er å videreføre de gode ideene fra myldringsprosessen og forkaste ideer man ikke trenger (Sandnes, 2022, s.268). Sandnes (2022) poengterer at de fleste ideer er dårlige og at man må forvente å få kritiske tilbakemeldinger på sine ideer. Hvis ikke så er det vanskelig å finne noen ideer å bygge videre på. Ofte kommer de beste ideene av et godt samarbeid og samspillet i gruppen, noe som kan være vanskelig å få frem hvis man er redd for å fornærme et annet gruppemedlem (Sandnes, 2022, s.269). Ved å ha gruppediskusjoner som ser på idégrunnlaget vil man kunne finne de beste ideene via å sette noen utslagsgivende kriterier, og dermed øke kvaliteten på ideene som blir diskutert (Sandnes, 2022, s.268).

2.3.4 Prototype

Benyon definerer en prototype som en sentral del av designprosessen, og at den skal være konkret, men ikke nødvendigvis en fullverdig representasjon eller implementasjon av system designet (2019, s.195). Med utgangspunkt i analysen gjort med PACT, personas, user stories

og MoSCoW var det verdifullt for gruppen å konstruere en prototype som dannet en felles enighet internt også mellom gruppen og Phonero sine ansatte. Gruppen diskuterte også fordelene og ulempene av å bruke ressurser til å utvikle wireframes utover sketches og mockups. Wireframes defineres av Benyon som en bredere skissering av et software-system (2019, s.194-195). Verdien av wireframes kommer ofte i den grad av at et konsept raskt kan bli fremstilt uten at det investeres store ressurser i en utdypende analyse og tiltenkt visuelt design. Fokuset er å fremstille essensen av konseptet og funksjonaliteten (Benyon, 2019, s.194-195). Likevel kan tidsbruken ved skissering av wireframes overgå estimerer. Det kan også brukes unødvendig tid i samtaler med kunder angående detaljer og mindre sentrale valg som ikke er gjennomarbeidet i et utkast med tanke på skisser og wireframes.

Gruppen har erfart fra tidligere prosjekter at det gir utslag i utviklingsprosessen å ha en strukturert og tydelig prototype. Verdien dette har gitt oss ligger i hvordan dette har bidratt til å redusere antall misforståelser innad i gruppen. En felles enighet i gruppen og direkte tilbakemeldinger fra Phonero på prototype med interaksjoner i sentrum, ga et utgangspunkt for å avgrense arbeidsoppgaver før utviklingen av produktet begynte. Avgrensningene gjort gjennom MoSCoW med tanke på MVP var også en tydelig avklaring av prosjektets omfang. Dette gjorde at gruppen fikk en bedre oversikt over allokering av ressurser og tid. Samtidig åpnet MoSCoW opp for å diskutere implementasjoner og funksjonalitet som ikke i utgangspunktet ville være kritisk for å oppnå en fungerende MVP. Tilbakemeldingene fra prototypen var i tillegg en anledning for brukerne å gi klar beskjed om hva de mente kunne bli tatt bort, og hva som kunne forbedres. Avgrensningene, tilbakemeldingene og designprosessen generelt var med å skape et stabilt og velformet prototype som gjorde programmeringsfasen mer streamlined (Sandnes, 2022, s.292).

2.3.5 Brukertestning

Brukertestning og tilbakemeldinger var et viktig steg for å sikre kvaliteten på produktet gruppen har laget. Brukere tester produktet ved å gi dem tilgang til systemet og de gjennomfører oppgaver ved for eksempel å navigere seg rundt i produktet for å bygge et vurderingsgrunnlag. Testingen vil i seg selv også gi objektiv informasjon både positivt og konstruktivt som vil lede til høyere kvalitet ved forbedring (Sandnes, 2022, s.316). En metode for å vurdere systemet gjennom testing er å lage konkrete scenarioer. Disse burde helst være så reelle som mulig og skal være med på å gi brukerne et klarere bilde av hva de

skal gjøre i testen og gi tilbakemeldinger på hva som er et forventet resultat (Sandnes, 2022, s.318-319). Gjennom disse scenarioene vil man også avdekke eventuelle feil eller mangler samt positive sider med systemet som blir testet (Sandnes, 2022. s.318).

For å sikre at kvaliteten var så høy som mulig ville gruppa også ta i bruk System Usability Scale (SUS). SUS er et standardisert skjema for brukervennlighet som vil være med på å gi målinger over hvor høy kvaliteten på produktet er. Hver bruker vil bli tildelt et skjemaet på slutten av hver brukertest hvor de svarer på ulike påstander (Sandnes, 2022, s.324). Ettersom SUS kun er med på å gi enkle målinger på brukervennlighet, ville ikke gruppa bare bruke det som metode, men heller ha en kombinasjon sammen med flere tester.

2.4 Verktøy

Prosjektet omhandlet å oppdatere et gammelt prosjekt hvor teknologien ikke var støttet av de nye plattformene som Phonero har. Det var da viktig at Defcon ble laget med moderne teknologier slik at Phonero sine systemer støtter den og at den er fremtidsrettet. Vedlikehold og videre utvikling var også noe vi måtte legge til rette for, så teknologivalget måtte også basere seg på hva Phonero allerede bruker slik at det ikke var behov for utviklere å tilegne seg ny teknologisk kompetanse bare for dette prosjektet. Dette går videre hånd i hånd med definisjonen til Martin et al. (2020) som tidligere er nevnt hvor grunnlaget for kvalitet ligger i forventningene til en kunde. I tillegg til gruppens teknologivalg som tar utgangspunkt i Phonero sin eksisterende teknologistack, ble det enighet om å bruke verktøy som utviklere hos Phonero bruker og er kjent med slik at overleveringsprosessen ville gå så lett som mulig.

Defcon består av en nettside (frontend) og en bakgrunnstjeneste (backend). Disse er to separate applikasjoner med sin egen teknologistack, men de kommuniserer med hverandre gjennom en Application Programming Interface (API). En API er en måte for å få to tjenester til å kommunisere med hverandre og dele data (Amazon, u.å.).

2.4.1 Frontendteknologier

TypeScript

TypeScript er et programmeringsspråk utviklet av Microsoft som et slags “overbygg” av JavaScript som legger til flere hjelpemidler for utviklere. Et eksempel på dette er typedefinering av variabler. TypeScript er ment til å løse kjente problemer med JavaScript, som kompleksitet i større prosjekter og raskere oppfanging av feil under utvikling (TypeScriptLang, u.å.). Gruppen valgte TypeScript fordi Phonero bruker det i alle sine prosjekter, og kompleksiteten av prosjektet virker mer overkommelig med bakgrunn i at koden kan virke “selvdokumenterende”.

React

React er et Javascript rammeverk som er utviklet av Meta Open Source. Rammeverket brukes til utvikling av brukergrensesnitt samtidig som det lar utviklere bygge gjenbrukbare byggesteiner til nettsiden din, også kalt komponenter. Disse komponentene kan ha sine egne tilstander som tilsier hvordan de skal vises i en nettleser, og React tar ansvar for hyppig oppdatering av nettsidens innhold. (React, u.å.). Ifølge W3Techs (u.å), bruker 4.3% av alle kjente nettsider i verden React som rammeverk, men dette inkluderer ikke interne nettsider. I likhet med TypeScript valgte gruppen å bruke React fordi Phonero sine kommersielle produkter også bruker denne teknologien. Den omfattende bruken av React-rammeverket gjør videre at det er en teknologi som er svært relevant å opparbeide seg kunnskap i, med tanke på dagens marked.

Tailwind

Tailwind er et rammeverk for Cascading Style Sheets (CSS) som tilbyr flere ferdiglagde CSS klasser. Disse kalles hjelpeklasser og de lar deg designe nettsider uten å måtte lage egne CSS filer. Tailwind kommer med ferdiglagde fargepaletter og andre designvalg, som skygger og enkel responsiv design. (Tailwind CSS, u.å.). Gruppen bestemte seg for å bruke Tailwind for å redusere behovet for antall CSS filer, og for å legge til rette for muligheten til bruke felles klasser istedenfor at hver enkel utvikler lager egne klasser som ofte gjør det samme som det noen andre har skrevet.

Shadcn/ui

Shadcn/ui er et sett av ferdiglagde, universelle utformet komponenter laget med React, TailwindCSS og komponentbiblioteket Radix UI (shadcn/ui, u.å.). Disse komponentene kan virke som byggesteinene til en nettside, og man kan argumentere for at det å bruke ferdiglagde komponenter sparer mye tid fordi en ikke må utvikle og designe dem selv. I tillegg får man oppdaterte versjoner når rammeverkene endres og språkene utvikles, så det blir mindre utvikling og testing for teamet. Derfor ble det enighet om å benytte shadcn/ui, og komponentene gjør slik at nettsiden har en felles stil.

2.4.2 Backendteknologier

C# ASP.NET Core

C# ASP.NET Core er et rammeverk av Microsoft for webutvikling som inneholder flere teknologier til å utvikle alt innen web, som nettsider og API. En av teknologiens styrker er at den kan kjøres på mange forskjellige plattformer, og Microsoft omtaler rammeverket som allsidig og moderne. (Microsoft, u.å.-a). C# ASP.NET Core er grunnlaget for backend til Defcon, og har ansvar for all forretningslogikk. For at Defcon skulle bli støttet av Phonero sine plattformer, var det også et krav å velge denne teknologien.

Hangfire

Hangfire er et C# rammeverk for håndtering og prosessering av bakgrunnsjobber i .NET. Hangfire lar deg definere jobber som du vil at skal kjøre regelmessig uten tilsyn av administratorer. (Hangfire, u.å.). Hangfire er en sentral del av Defcon siden den har ansvar for å kjøre jobber regelmessig, som i Defcon sitt scenario er alarmer. Med Hangfire kunne det unngås å lage denne logikken selv, noe som tillot gruppen å legge større vekt på andre deler av forretningslogikken.

Microsoft SQL Server

Microsoft SQL Server (MSSQL) er en relasjonsdatabase utviklet av Microsoft brukt til å lagre data (Microsoft, 2024b). En relasjonsdatabase er en database hvor lagrede datapunkter kan referere eller relateres til andre datapunkter i databasen (Oracle, u.å.). MSSQL er en kjerneteknologi for Defcon fordi informasjon som brukere genererer må lagres for senere visning.

2.4.3 Andre verktøy

Jira

Jira er et prosjektstyringsverktøy utviklet av Atlassian. Jira tilbyr en et mangfold av verktøy for å gjøre blant annet utvikling av programvare strømlinjeformet, og legger opp for effektiv kommunikasjon mellom avdelinger. (Atlassian, u.å.). Jira ble valgt på bakgrunn av at Phonero har tatt det i bruk og at de allerede hadde lagt opp til at gruppen kunne bruke det før prosjektet ble startet.

ChatGPT

ChatGPT er et program som bruker kunstig intelligens og avanserte maskinlæring algoritmer som har evnen til å tolke og forstå både skriftlig og muntlig språk. Programmets kontekstbevissthet gjør at det kan opprettholde en dialog med brukerne og håndtere oppfølgingsspørsmål. For gruppen ligger verdien av ChatGPT i evnen den har til å forstå menneske-maskin- interaksjon, samt potensialet til å forbedre effektiviteten gjennom dens innsikt i problemløsning. (University of Central Arkansas, u.å.)

Integreringen av ChatGPT i praksis kan by på både fordeler og utfordringer. Programmet fungerer som et godt verktøy for spesielt idemyldring og oppsummering av tekst og kode, men kan noen ganger produsere feilaktige svar på bakgrunn av dens manglende tilgang til ekstern informasjon fra internett. Dette blir spesielt tydelig i sammenheng med programmering, hvor det krever betydelig mye kontekst for å generere nøyaktige og brukbare svar (OpenAI, u.å.). Fordelen for gruppen ved bruk av dette verktøyet lå i hvordan man kan få teoretiske løsninger og et innblikk i hvordan en oppgave kan løses som grunnlag når man løser oppgaven i praksis.

Figma

Figma er et verktøy som ble en sentral bit av gruppas prosjekt og er beskrevet som et godt designverktøy som var med å gjøre disse prosessene lettere. Figma ble valgt ettersom det tillot å ha en felles plattform hvor alle kunne jobbe på samme sted på flere forskjellige maskiner (Figma, u.å.).

Visual Studio Code / Visual Studio

Visual Studio Code (VS Code) er en kodeeditor utviklet av Microsoft som er veldig fleksibel og har et omfattende økosystem. (Visual Studio, u.å.). I en spørreundersøkelse for utviklere utført av Stack Overflow (2023), svarte 73% at de brukte VS Code som sin kodeeditor. Individuelle gruppemedlemmer sto fritt til å velge hvilken kodeeditor de ville bruke, men alle valgte VS Code til frontendutvikling.

Visual Studio (VS) er et integrert utviklingsmiljø også utviklet av Microsoft. VS sikter på å samle hele utviklingssyklusen inn i ett program, og lar deg gjøre alt fra å skrive kode til feilsøking og bygging av kode. (Microsoft, 2023c). Gruppen brukte VS til backendutvikling siden verktøyet har en rekke med hjelpemidler som gjør utvikling i C# og .NET lettere.

3. Gjennomføring av prosjektet

For å sikre kvalitet på produktet og i selve prosessen valgte gruppen som tidligere nevnt å forholde seg til ulike teoretiske rammeverk rettet mot systemutvikling. Bruk av Scrum var helt essensielt for kvalitetssikring i form av at den tillot grundig planlegging og evaluering av sprinter. Ved hjelp av sprint planning, daily standup, sprint retrospektiv og sprint review var det mulig å kartlegge de ulike delene av prosjektet som gruppen kontinuerlig måtte forholde seg til. Ulike analyser ble grundig planlagt i startfasen og etterfulgt i løpet av prosessen. MoSCoW ble brukt til å klarlegge og rangere ulike funksjoner som var planlagt å implementere. Risikoanalysen ble utarbeidet med bakgrunn i å kartlegge mulige hendelser som i ulik grad kunne kompromittere prosjektet, og utarbeide potensielle løsninger på disse problemene. Videre ble det gjennomført en PACT-analyse av prosjektet for å analysere ulike kontekster som gruppen så det hensiktsmessig å ta utgangspunkt i. I tillegg til disse analysene og rammeverkene ble det foretatt kontinuerlige tidsestimeringer på hvor lang tid som var nødt til å bli satt av på enkelte deler av prosjektet og for å lage et bilde på hvor lang tid ting ville ta totalt. Summen av dette var en omfattende løsning på usikkerhet og endringer som oppstod under prosjektets løp. Senere i dette kapittelet vil gjennomføringen av de ulike rammeverkene og analysene bli forklart.

3.1 Kvalitet

Som tidligere nevnt bestemte gruppen seg for å komme fram til en felles forståelse av kvalitet. Dette var fordelaktig fordi det ga gruppen et hjelpemiddel for å navigere gjennom alle de ulike forståelsene av kvalitet i tillegg til at det førte til en enighet internt i gruppa rundt begrepet. Videre førte dette til en solid forventningsavklaring og forankring hos samtlige gruppemedlemmer som alle kunne forholde seg til. Under initielle diskusjoner ble det enighet om å ta utgangspunkt i den ovennevnte definisjonen som baserer seg på standardene og kravene som blir stilt fra kunden, nemlig “*Quality-as-agreed*”. For å påse at kvalitet hele tiden var sentralt i prosjektet valgte gruppen å lene seg på eksisterende rammeverk og analyser. Dette var hensiktsmessig for å ha klare rammer det var mulig å lene seg tilbake på dersom det skulle oppstå uforutsette hendelser eller endringer i løpet av prosjektet.

3.2 Analyser

3.2.1 PACT

Som nevnt i kapittel 2.3.1 er PACT forklart som et rammeverk gruppen tok nytte av for å forstå brukerne, aktivitetene deres, hvilken kontekst systemet skal bli brukt i, og hvilke teknologiske krav det er. Dette har vært en viktig del av gruppens tilnærming til prosjektarbeidet. Videre i kapitlet blir det forklart hvordan PACT- analysen har blitt anvendt i praksis for å forme en forståelse av prosjektets begrensninger og krav.

People

Mennesker er det første punktet i analysen og gruppen startet med å identifisere de viktigste brukergruppene som skulle ta i bruk systemet. Dette var hovedsakelig IT- og Billing avdelingene på Phonero. Gjennom tidligere diskusjoner innad i gruppen og med produkteier og ansatte hadde brukergruppens tekniske kompetanse og behov blitt kartlagt. Dette bidro til en bedre forståelse av hvem som skulle anvende det nye systemet og hvilke forventninger de hadde til det.

Activities

Ved å analysere hvordan teknologien skal bli tatt i bruk, ble de viktigste aktivitetene som brukerne skulle gjøre identifisert. Det inkluderer registrering, kommentering og redigering av saker, samt opprettelse av databaser og avdelinger. Det ble etablert at teknologien skulle bli

tatt i bruk hver dag hele tiden, det var derfor viktig at spørringer i databasen var raske for å hindre lang ventetid. Brukerne skulle også vite hvor nødvendig det er å løse en alarm, hvor ofte alarmene gikk, og å kunne se hvem som jobber på en alarm.

Context

For å best kunne forstå når og i hvilken sammenheng teknologien skulle bli tatt i bruk, ble det foretatt en analyse av brukernes arbeidsmiljø og behov. Det innebar å identifisere at systemet vil hovedsakelig bli tatt i bruk på arbeidsplassen, men med mulighet for å bruke på hjemmekontor med PC. Det ble også tatt hensyn til brukernes behov for å kunne kommunisere gjennom dokumentering i hver alarm for en mer praktisk løsning sammenlignet med fysisk kommunikasjon på tvers av avdelinger.

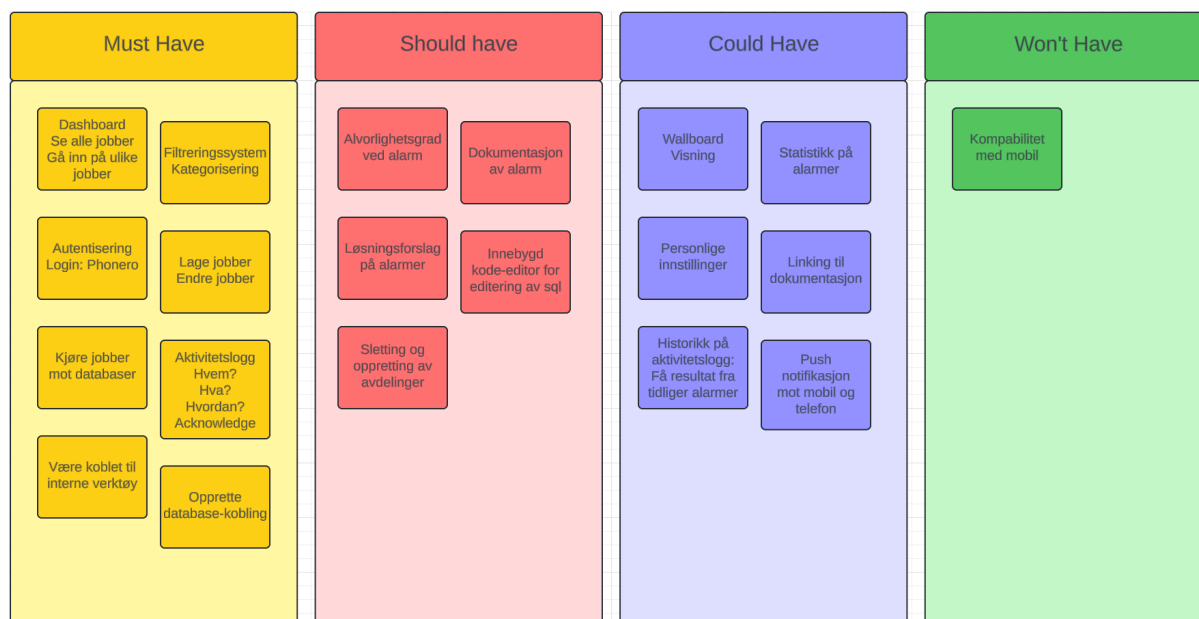
Technologies

Systemets teknologier som ble tatt i bruk ble undersøkt, og deres påvirkning på brukeropplevelsen ble vurdert. Dette inkluderte å vurdere bruken av smarttelefoner, PC-er og en Wallboard-visning, samt å velge en god teknologisk stack for å oppnå ønsket funksjonalitet, design og ytelse. Smarttelefoner, PC-er og wallboard er alle betydelig forskjellige når det kommer til layout og design av systemet, noe som kunne gi en utfordring i utviklingsprosessen, dette var et tema som senere ble tatt opp i MoSCoW- analysen.

Gjennom bruken av PACT-analysen fikk gruppen en god forståelse av brukernes behov og krav, noe som ble viktig for å kunne utforme og utvikle systemet. Analysen ga et godt grunnlag for å kartlegge systemets formål og for å kunne ta beslutninger med god samvittighet for å sikre at produktet møter brukernes forventninger og behov.

3.2.2 MoSCoW

For å komme i gang med MoSCoW-analysen hadde gruppen et møte hvor alle funksjonene som kom frem under idéforedlingen skulle bli implementert i systemet. Disse funksjonene ble deretter gitt underpunkter som beskrev hva hver funksjon innebærer. Eksempelvis ble funksjonen "håndtere scripts" delt inn i underpunkter som "kjøre scripts", "stoppe scripts", "pause scripts" og "frekvens på hvor ofte scripts kjøres".



Figur 3: MoSCoW siste utkast

Gjennom ulike møter i Scrum prosessen hvor produkteier var involvert, ble funksjonslisten kontinuerlig oppdatert. Nye funksjoner ble lagt til og eksisterende funksjoner ble justert basert på tilbakemeldinger fra testing og diskusjoner med produkteier. Oppdateringene og justeringene ble gjort innenfor rammene av de etablerte prioritetsnivåene i analysen, hvor fokuset lå på å imøtekomme kravene som foreligger i systemets MVP.

Til tross for at det oppsto endringer underveis i prosjektet, spesielt med funksjonslisten i minne, ble de høyest rangerte funksjonene i “Must have” opprettholdt. Endringene var hovedsakelig relatert til utvidelser og justeringer av funksjonene som var “Should have” og “Could have” kategoriene. Etterhvert ble noen funksjoner lagt til i MoSCoW modellen, og et fåtall ble prioritert opp.

Et eksempel på dette er hvordan funksjonen “innstillinger” som var listet under “Could have”, ble utvidet og justert. “innstillinger” er en side innad i systemet som originalt sett kun inneholdt underpunktet “legge til database”. Dette ble utvidet til å inneholde “darkmode” og “legge til department” i tillegg. Noe som senere ble justert nok en gang på bakgrunn av tilbakemeldinger fra brukerne i brukertesting. Gruppens reaksjon på dette var at funksjonen “innstillinger” skulle bli gjort om til “preferences/personlige innstillinger” og inneholde to nye preferanse-alternativer relatert til brukervennligheten i systemet. Videre skulle innholdet i “innstillinger” flyttes til en ny side som heter “manage”, dette på bakgrunn av at plasseringen

av “legge til database” og “legge til department” samsvarte mer med hva brukerne allerede var vant til og dermed ville forhindre forvirring rundt navigering. Som vist i siste utkast av MoSCoW modellen (figur 3), ble to funksjoner i manage siden lagt til i MoSCoW-modellen og prioritert. Her var det hensiktsmessig å gi funksjonene nye navn som var mer beskrivende. "Legge til departement" ble endret til “sletting og oppretting av avdelinger” og ble satt til “Should have”. “legge til database” ble om til "opprette database-kobling” og satt til “Must have”.

3.2.3 Risikoanalyse

Risikoanalysen baserte seg på en mal hvor det ble lagt til nye punkter som bestod av tidligere erfaringer fra andre gruppeprosjekter og nye potensielle hendelser som ble utarbeidet og diskutert med innspill fra produkteier. Hendelser og forslag ble deretter gjennomgått i fellesskap for å vurdere om de var relevante for prosjektet. Hver enkelt risikohendelse ble så analysert ved å vurdere sannsynligheten for at det vil skje og konsekvensene det vil ha for prosjektet, på en skala fra 1-5.

En del av prosessen var utformingen av en tilpasset risikomatrise med flere nivåer representert av fargene grønt, gult, oransje og rødt. Disse nivåene bidro til å gi et visuelt bilde av risikonivået for hver hendelse. I figur 4 (risikoanalyse) blir nivåene representert i matrisen. Hendelser som endte opp innenfor grønn og gul sone ble ansett som innenfor akseptable rammer, mens hendelser i oransje og rød sone ble ansett som mer alvorlige og i behov av mer oppmerksomhet og tiltak for å unngås. I praksis ble risikoanalysen kontinuerlig brukt som et verktøy for å veilede gruppen i prosjektarbeidet. Gjennom prosjektløpet hadde gruppen de identifiserte risikohendelsene i bakhodet, noe som bidro til økt bevissthet og gjorde medlemmene utrustet for potensielle utfordringer. Hvis en hendelse oppsto, kunne man raskt henvise til risikoanalysen for veiledning om hvilke tiltak som kunne iverksettes. Selv om analysen ikke nødvendigvis ble brukt i den daglige driften av prosjektet, var den et verdifullt verktøy som hjalp gruppen med å forutse og håndtere potensielle problemer på en god måte.

SANNSYNLIGHET	Svært Sannsynlig	5	5	10	15	20	25
	Meget Sannsynlig	4	4	8	12	16	20
	Sannsynlig	3	3	6	9	12	15
	Lite Sannsynlig	2	2	4	6	8	10
	Usannsynlig	1	1	2	3	4	5
			1	2	3	4	5
			Ubetydelig	Mindre Alvorlig	Betydelig	Alvorlig	Svært Alvorlig
KONSEKVENNS							

Figur 4: Risikoanalyse. Basert på “Risikomatrise”, 2016, Regjeringen

(<https://www.regjeringen.no/no/dokumenter/nou-2016-19/id2515424/sec3?q=nou%202016:19>)

Et eksempel hvor risikoanalysen ble brukt var i forbindelse med en av de identifiserte risikohendelsene som omhandlet størrelsen på gruppen. Gruppens ambisjonsnivå var høyt fra starten av, men det var usikkerhet rundt om det var nok arbeidsoppgaver for alle gruppemedlemmene. Det ble senere tydelig at omfanget til prosjektet var så stort at det ville være mer enn nok arbeid for alle i gruppen. Imidlertid, i tråd med en annen risikohendelse som var knyttet til sykdom, opplevde gruppen flere situasjoner hvor gruppemedlemmer ble syke. Et medlem ble også borte over lengre tid grunnet sykemelding, noe som førte til en betydelig økning i arbeidsbelastning for resten av gruppen. Denne hendelsen avdekket en uventet konsekvens rundt arbeidsbelastning som tidligere ikke ble tatt nok i betraktning, og førte til at gruppa periodevis hadde utfordringer med tidsfrister og arbeidsplanen. I og med at en agil arbeidsmetodikk har blitt praktisert, var det innenfor akseptable rammer at slike hendelser forekom, men det var likevel noe gruppen ikke var godt nok forberedt på.

For å forhindre at ovennevnte eksempel ville få følger for resten av prosjektløpet, ble det diskutert løsninger i sprint retrospektiv. Flere gruppemedlemmer kjente på ubehag for ikke å komme i et endelig mål med ønsket produkt og det ble enighet om å være i forkant av skippertak mot avslutningen av prosjektet. Løsningen var å øke antall obligatoriske dager for oppmøte på Phonero fra 3 dager til 5 dager. Som tidligere nevnt vil slike endringer være innenfor akseptable rammer når man bruker et agilt rammeverk og arbeidsmetode ettersom det tillater gruppen å være tilpasningsdyktige. Sett i lys av arbeidsrelatert erfaring, ville dette

tilsagt at gruppen iverksatte overtidsarbeid for å nå midlertidige tidsfrister av milepæler i prosjektet. Med risikoanalysen i betraktning ble det diskutert om det var hensiktsmessig å øke arbeidsmengden i så stor grad. Gruppen drøftet tematikk rundt "Scope creep", det å ukontrollert legge til funksjoner eller arbeidsoppgaver uten å vurdere tidsestimering og kostnader. Om hvordan det ville påvirke det endelige produktet, utbrenthet og samhold i gruppen. Selv om det ble frigitt ressurser til å løse oppgaver, var det en fare for at vi fikk nye krav og ønsker fremover i prosessen og at problematikken gjenoppsto. Gruppen konkluderte med at det var fornuftig med en økt arbeidsmengde midlertidig, for å sikre at kravene som ble satt i MoSCoW og MVPen ble møtt innenfor projektrammene.

3.3 Scrum

Som nevnt tidligere var det en naturlig avgjørelse for gruppen å anvende Scrum som projektrammeverk. Den agile metodikken ved bruk av Scrum la opp til kontinuerlig læring og vurdering av prosessen var fordelaktig av flere grunner. Som nevnt var gruppen kjent med rammeverket fra før av fordi det tidligere hadde blitt brukt i annet prosjektarbeid. Med bakgrunn i manglende erfaring fra prosjekter av dette omfanget var det hensiktsmessig med et rammeverk som tillater å gjøre justeringer ofte. Det ble tidlig gjort et valg om at sprintene skulle vare i to uker. Motivasjonen for dette var at gruppen ønsket kontinuerlige tilbakemeldinger fra produkteier som ga innspill på design og videre utvikling og prioritering av funksjoner. Hadde det blitt tatt utgangspunkt i lengre sprinter kunne dette ha ført til mindre rom for endringer. Samtidig ville kortere sprinter ført til en dårligere sprint retrospektiv og review fordi gruppen ikke ville hatt like mye å vise til og få tilbakemeldinger på.

3.3.1 Implementering av The Iron Triangle

En gjennomgående diskusjon i gruppa var balansen av "*The Iron Triangle*". Kost, tid og kvalitet var sentrale byggesteiner i prosjektet. Kost per se var mindre relevant da økonomi i seg selv ikke var relevant. Som tidligere beskrevet bestemte allikevel gruppen å trekke linjer mellom kost og tid for å likevel kunne forholde seg til begrepet. Det kan argumenteres for at selve triangelet mister sin betydning når en av de tre delene ikke blir tatt i betraktning, men idèen om at Phonero som en bedrift satt av ressurser og tid til å bistå i arbeidet med oppgaven ble av gruppen ansett som nok grunn til at begrepet "kost" allikevel var relevant. Phonero stilte blant annet opp med lokale og lån av PC-er som ble brukt av gruppa under hele

prosjektet. I tillegg til dette fikk alle gratis lunsj hver dag. Phonero sto dessuten til disposisjon med kompetanse gjennom ansatte som ofte var tilgjengelige om det ble behov for innspill og hjelp. Det var også mulighet for lengre møter med de ansatte dersom det var noe som var usikkert, eller noe gruppen trengte hjelp med. Sprint planning-møtene som ble gjennomført ble en gyllen mulighet til å drøfte justeringer rundt rammene for "The Iron Triangle". Selv om det var her mesteparten av vurderingene rundt kost, tid og kvalitet fant sted, var det også en kontinuerlig prosess, gjennom blant annet analyser som blir nevnt videre i rapporten.

3.3.2 Scrum master

Et bevisst valg gruppen gjorde tidlig i prosessen var at samtlige gruppemedlemmer skulle være Scrum master minst én gang. Grunnen til dette var flerfoldig, hvor den mest framtreddende grunnen var et ønske om at alle skulle få erfaring som leder av prosjektet. Som tidligere nevnt er hovedrollen til Scrum master å fungere som veileder, men det ble enighet om å gjøre rollen til en mer prosjektleder-relatert rolle. En annen grunn til å rullere på Scrum master rollen var å balansere ansvarsbyrden for det å være Scrum master. Som nevnt har Scrum master ansvar for at de ulike delene av Scrum-prosessen blir gjennomført i tillegg til å tilrettelegge for en god flyt i arbeidet. Som Scrum master har alle på gruppen hatt dette ansvaret i tillegg til å gjøre annet administrativt arbeid, som møteinnkallelser og utarbeiding av referater fra møtene. Selv om Scrum master hadde siste ordet i alle avgjørelser som ble tatt, var det gjengs oppfatning at alle problemer eller utfordringer som oppstod skulle løses i plenum. Stor takhøyde med rom for tilbakemeldinger og diskusjon gjorde at problemløsning utartet seg på en konstruktiv måte gjennom hele prosjektets løp.

3.3.3 Sprint planning

Før begynnelsen av en ny sprint satt gruppen seg ned for å planlegge hvordan den neste sprinten skulle se ut. Gjennom planleggingen ble det utarbeidet et overordnet mål for sprinten og fikk dermed et overblikk over de kommende to ukene. Denne tidsrammen bidro til et oversiktlig helhetsbilde som det var hensiktsmessig å lene seg tilbake på dersom det skulle oppstå usikkerhet rundt målet i sprinten. I tillegg ble både nye oppgaver, og oppgaver fra backloggen (figur 5) ført inn under den nye sprinten.

☐ Backlog (15 issues)	7 0 0	Create sprint
☒ DAT-233 Har egt tobias gjort dette i en annen oppgave? : Virtualisering av Aktivitetslogg	TO DO	-
☒ DAT-220 Backend: Rewrite text -> JSON format : Tobias	BACKEND	TO DO
☒ DAT-211 Destructive farge i dark mode er ikke riktig	TO DO	-
☒ DAT-196 Dashboard: Pushbuttons får tilbake feil antall alarmer når det filtreres med andr...	FRONTEND	TO DO
☒ DAT-242 Dashboard: Result row -> might be issues with to large amouint for row results	TO DO	-
☒ DAT-195 Test at ting ikke crasher helt dersom backend ikke kjører/er utilgjengelig.	TO DO	-
☒ DAT-93 Epostnotifikasjoner for Watchers	BACKEND	TO DO
☒ DAT-202 Dashboard: Lage hovercard på hver enkelt alarm/counter i data-table	FRONTEND	TO DO
☒ DAT-203 CORS oppsett til dev/prod	TO DO	-
☒ DAT-117 Frontend Dashboard: Gjennom søkefelt gjøre det mulig å søke etter tags og få ...	FRONTEND	TO DO
☒ DAT-108 Dokumentasjon av Teknologier	RAPPORT - BACHELOR...	IN PROGRESS
☒ DAT-57 Lag README til backend oppsett	BACKEND	TO DO
☒ DAT-37 Slett gamle dataquality-web repository	FRONTEND	TO DO
☒ DAT-70 (Handover) Mulighet for "Currently watching" på backend, med Signal R eller we...	FRONTEND	TO DO
☒ DAT-160 (HANDOVER) Dashboard: View og Filter blir plassert riktig ved skalering / mobil?	FRONTEND	IN PROGRESS

+ Create issue

Figur 5: Backlog

Den agile arbeidsmetoden som ble praktisert la til rette for en kontinuerlig gjennomgang og vurdering av design og utvikling hvor de viktigste aspektene ble prioritert i de inneværende sprintene. I retrospekt er det tydelig å se at planlegging tidvis var for optimistisk da det var gjennomgående at oppgaver fra backloggen stadig måtte flyttes til den kommende sprinten. Som tidligere nevnt anvendte gruppen Fibonacci-sekvensen for å estimere tidsbruken på de enkelte oppgavene. Grunnen til ønsket om å estimere poeng var for å videreføre kvalitet i form av tidsbruk, slik at oppgaver i like stor grad ikke ble forskjøvet til senere sprinter. Måten sekvensen ble brukt på blir beskrevet i tabellen (figur 6):

Poeng	Tidsbruk
0 poeng	Under 1 time
1 poeng	Ca 1 time
2 poeng	Ca 2 timer
3 poeng	Mellom 2 timer og en halv dag
5 poeng	Over en halv dag, til en dag
8 poeng	Ca en dag
13 poeng	Mer enn en dag
21 poeng	Mer enn to dager

Figur 6: Fibonacci-sekvensen

Utformingen av denne tabellen (figur 6) og tiden som poengene representerer ble innledningsvis nøye vurdert, noe gruppen tjente godt på under prosjektets løp. Hvis en person tok på seg en oppgave var det opp til den enkelte å estimere hvor lang tid den kom til å ta, og dette igjen førte til en gjengs forståelse av arbeidsomfanget av de ulike oppgavene i sprintene. Det er verdt å nevne at det ble satt et tak på 21 poeng per oppgave fordi det var ønskelig å forholde seg til konkrete oppgaver som var mer overkommelige å ta fatt på. Hvis en oppgave ble vurdert til å tilsvare mer enn 21 poeng ble det satt som et krav internt at den skulle deles opp i mindre underoppgaver. Dette var hensiktsmessig for å få et lavere abstraksjonsnivå på oppgavene slik at ingen oppgaver ble for store.

Ved utarbeiding av oppgaver i Jira var det ønskelig å produsere selvforklarende beskrivelser. Hensikten med dette var blant annet relatert til kvalitetssikring av tidsestimater. Innledningsvis var ikke oppgavene beskrevet til et tilfredsstillende nivå, noe som gjorde at det oppstod utfordringer ved fravær blant gruppemedlemmene. Som et resultat av dette ble det bestemt at oppgavene skulle være mer deskriptive, og det skulle bli mer flittig bruk av underoppgaver. De beskrivende og veldokumenterte oppgavene var en stor fordel, også med tanke på størrelsen til gruppa. Som en gruppe på seks personer kunne det helt i starten av prosjektet virke som det skulle bli utfordrende å organisere og strukturere arbeidet på en effektiv måte. Dette ble ikke tilfellet da gruppen fokuserte på å fordele arbeid til alle, og måten dette ble gjort på var blant annet å lage mer konkrete oppgaver i Jira. Derav ble ingen oppgaver overflødige på bakgrunn av nøye planlagte og grundige gjennomførte sprintplanlegginger.

3.3.4 Daily Scrum

Gjennom prosjektet løpet ble det gjennomført daily Scrum hver dag, så fremt det var fysisk oppmøte. Møtene varte i alt fra 10 til 30 minutter, og noen ganger enda lenger, alt ettersom hvor mye det var å ta opp. I starten av prosjektet bestemte gruppen seg for 3 spørsmål alle skulle svare på, og som Scrum master noterte ned svarene på. Spørsmålene var : 1. ”Hva gjorde du i går?”, 2. ”Hva skal du gjøre i dag?” og 3. ”Har du noe å ta opp?”. De to første spørsmålene ga gruppemedlemmene en konstruktiv start på dagen hvor det ble klart for samtlige hva den enkelte gjorde forrige dag og hva som var planen samme dag. Dette var hensiktsmessig i forhold til tidsestimering hvor det ble klargjort et estimat på hvor lang tid den enkelte kom til å bruke på dagens arbeidsoppgaver. Spørsmål nummer tre ga

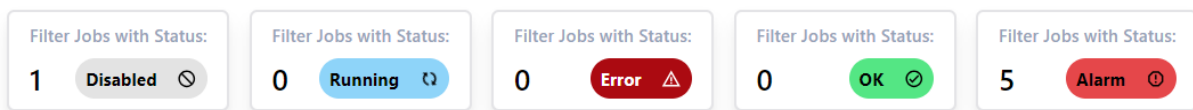
gruppemedlemmene mulighet til å ta opp smått og stort. Hvis noen hadde støtt på utfordringer foregående dag var dette en gylden mulighet til å ta det opp og potensielt få hjelp til å løse utfordringen. Spørsmålet ble også brukt for å finne ut av mye annet også, være seg motivasjonsnivå eller dagsformen på gruppemedlemmene.

Tidlig i prosjektet ble det tatt i bruk et element i daily Scrum som gikk ut på å gi en hyggelig kommentar til et av de andre medlemmene av gruppa. Dette ble gjort fordi det ga en hyggelig start på dagen hvor samtlige fikk et smil om munnen og en god følelse i kroppen. Igjen hadde dette en positiv effekt på motivasjonsnivået og vi kom i forskudd på ting som prosjekt-tretthet. Selv om daily Scrum enkelte ganger kunne bli et tidkrevende element bidro det til en god struktur på dagene i tillegg til at alle ble satt i en slags arbeidsmodus. Det ble også en lavere terskel for å be om hjelp og lettere å forholde seg til tidsestimeringer. Det å sette ting på dagsorden gjorde at vi kom utfordringer i forkjøpet, som igjen bidro til en konstruktiv måte å jobbe på som en gruppe.

3.3.5 Sprint review

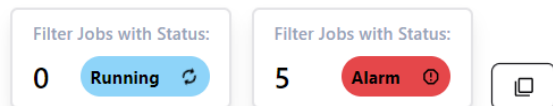
Som nevnt tidligere i rapporten tilrettelegger implementeringen av agile Scrum metodikk mulighet for kontinuerlig tilbakemeldinger fra produkteier og ansatte gjennom utførelsen av sprint reviews. Sprint reviews ble gjennom prosjektløpet holdt i enden av hver sprint som for vår del ofte falt på en mandag. Varigheten på en review ble bestemt ut ifra hvor mye innhold gruppen ønsket å få tilbakemelding på, og som regel varte disse møtene alt fra 15 til 40 minutter. Strukturen på gjennomføringen av sprint review var relativt lik hver gang. Gruppen startet med å introdusere en agenda for møtet, etterfulgt av en framvisning av hva som hadde blitt produsert i form av demo visning. Videre ble det framvisning av backloggen hvor produkteier hadde mulighet til å få innblikk i hvor mange arbeidsoppgaver som hadde blitt løst. Møtet ble avsluttet med en gjennomgang av sprint målet og milepæler som tidligere ble satt under sprint planningen.

I begynnelsen av prosjektet var usikkerhet rundt hvor mye produkteier skulle involveres på bakgrunn av at de hadde sine egne prosjekter å drive. Etter tilbakemelding fra produkteier vedrørende for lite involvering valgte gruppen deretter å invitere produkteier til samtlige Scrum aktiviteter som planning, review og retrospektiv. Dersom produkteier ikke var tilgjengelig ble det utarbeidet et referat slik at han var oppdatert på tidligere og videre prosesser. Gruppen så seg nødt til å mer eller mindre justere seg etter kommentarer fra produkteier vedrørende funksjonalitet eller designvalg som ble tatt. Rollen produkteier spilte under et review ble derfor essensiell for å sikre framdrift og kvalitet i prosjektet. Eksempler på justeringer innenfor designvalg etter tilbakemelding fra produkteier, kan vi se gjennom implementeringen av “pushbuttons”.



Figur 7: Originale pushbuttons konsept

Under et review presenterte gruppen denne utformingen av dashboard display med pushbuttons. Det ble gitt tilbakemelding fra produkteier vedrørende at brukere som regel kun ønsker å se jobber med status “Alarm” og "Running". Dette ble tatt i betraktning inn i neste sprint hvor det ble designet et nytt konsept rundt pushbuttons.



Figur 8: Ferdigstilte pushbuttons

Følgende sprint review presenterte gruppen dette designvalget hvor pushbuttons skjules ved hjelp av filtrering. Her fikk brukeren mulighet til å skru av og på hvilken pushbuttons de ønsket å se. Dette ble gjort på bakgrunn av at produkteier og gruppen fant det hensiktsmessig å beholde de ulike filtreringsknappene da statuser som “Ok” og “Error” kun ble brukt i enkelte tilfeller. Dette er kun ett av flere eksempler gruppen kunne presentert. Gjennom hele prosjektet har produkteier etter ønske fra gruppen, hatt mulighet til å gi tilbakemelding på både arbeidsprosess, arbeidsoppgaver og resultat.

3.3.6 Sprint retrospektiv

Gjennomføring av sprint retrospektiv var for gruppen en god måte å kontinuerlig forbedre de interne prosessene som ble kjørt gjennom prosjektløpet. I dette møtet deltok alle gruppemedlemmer i lag med produkteier. Hensikten med å involvere produkteier var et ønske om å få et utenfra-perspektiv på sprintprosessen. Varigheten på møtet ble satt ut ifra hvor mange diskusjoner eller temaer gruppen ønsket å ta opp. En normal sprint retrospektiv varte mellom 15 og 40 minutter.

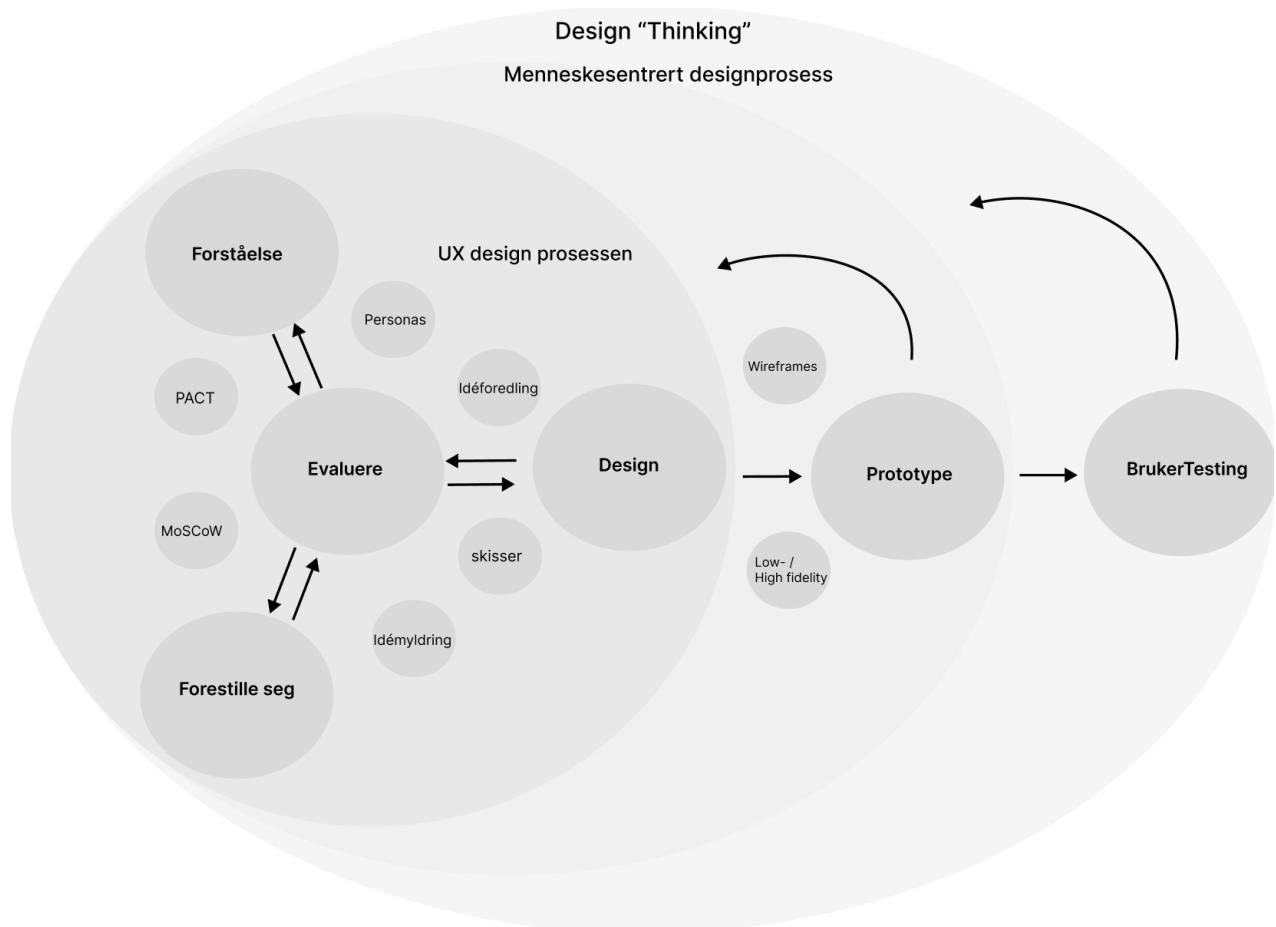
En sentral del av sprint retrospektiv gjelder refleksjon rundt arbeidsprosesser, estimeringer, gruppedynamikk og ellers andre temaer som ble sett på som viktige. Dette skjer gjennom diskusjon av 3 forskjellige spørsmål. Spørsmålene var: 1. “Hva syntes du gikk bra under sprinten?”, 2. “Hva syntes du gikk dårlig under sprinten?” og 3. “Hva ønsker vi å oppnå i videre sprinter?”. I starten fungerte dette bra da det ble tatt opp mange temaer som gruppen så seg nødt til å diskutere. Videre inn i prosjektet ble det tydelig at enkelte diskusjoner stagnerte. Resultatet av dette ble at gruppen brukte en god del tid på refleksjon uten at konkrete tiltak ble innført. Dette ble tatt opp under et møte med produkteier fordi det var ønskelig å høre om hvordan hans team hadde løst dette problemet.

Under diskusjonen ble det nevnt at teamet hans kjører en metode som baserer seg på begrepene Start, Stop og Continue. Dette tok gruppen inn i betraktning videre inn i senere sprinter, hvor det igjen ble utarbeidet 3 nye spørsmål. Spørsmålene var START; ”Hva burde vi starte å gjøre?”, STOPP; ”Hva burde vi stoppe å gjøre?” og CONTINUE; ”Hva burde vi fortsette å gjøre?”. Ved å benytte disse definerte begrepene ble det lettere å komme fram til konkrete løsninger. Det er flere eksempler på løsninger implementert ved hjelp av start, stop og continue. Disse illustreres i figuren (9) under:



Figur 9: Start, Stop, Continue

3.4 Design



Figur 10: «Design Thinking»

Som referert i kapittel 2 "Design og Kvalitet", var det flere faktorer å ta hensyn til i utviklingsprosessen til Defcon. For dette prosjektet, samt andre prosjekter er designfasen en essensiell del av prosjektstyringen. Samtidig som det ble skissert arkitektur og design av produktet, ble det også diskutert hvilken tilnærming som skulle bli brukt underveis. Figur 10 representerer designprosessen som har blitt etterstrebet i prosjektet, og gjennom flere iterasjoner har vi bygd på UX-prosessen gjennom design "thinking" og et menneskesentrert perspektiv. Dette dannet grunnlaget for vår tilnærming til designprosessen.

3.4.1 Designprosessen

Med ansatte og produkteier i Phonero gjennomførte gruppen en idémyldringsprosess og lagde en funksjonsliste som baserte seg på hvilke ønsker og krav Phonero hadde for det nye produktet. I myldringsprosessen ble PACT metodikken brukt til å bygge en forståelse av hvem brukerne var og hvilke utfordringer de hadde. Videre brainstormet teamet med noen ansatte, hvor også teamet gjennomførte individuelle møter der “Must- og Should have” funksjoner ble skissert.

Gruppen opplevde at ansatte hos Phonero var engasjerte og villig til å delta, og ga konstruktive tilbakemeldinger til det som ble tatt opp. Helhetlig fikk gruppen utløp for mange ideer og design skisser som kunne passe til produktet. Med produkteier fikk gruppen fastslått hvilke av de ideene som var verdt å bygge videre på. Innholdet fra datainnsamling og samtalene, samt krav og ønsker, ble systematisert og fylt inn i rammeverket MoSCoW.

Michelle IT



ALDER

32

UTDANNELSE

Bachelor i IT og Informasjonssystemer

STATUS

Gift

STILLING

System utvikler

BOSTED

Kristiansand

TEKNISK KUNNSKAP

Høy

“ Jeg liker å bruke DQM når jeg sitter på kontoret. Er raskere å komme i kontakt med de som er relevante hvis en alarm slår ut.

Personlighet

Ekstrovert

Ansvarlig

Åpen og omgjengelig

Teknisk

Biografi

Michelle bor i kvadraturen og sitter lange dager på kontoret. Er gift og er kjent for å strukturere dagen godt. Endel av planleggingen går på kjennskap til arbeidsoppgaver nå og i fremtiden. Michelle er litt kjent med interne verktøy, men ennå ny i jobben etter bare 1.5år hos phonero.

Kjerne behov

- Avhengig av en stabil DQM for å jobbe effektivt.
- Lett tilgjengelig tjeneste som ikke er for avansert, da det allerede eksisterer andre verktøy for å håndtere utslag fra alarmene andre steder.
- Ønsker å legge meg til som watcher på en alarm så jeg får beskjed når det er en oppdatering på en utslått alarm
- Ønsker å se dokumentasjon på alarmer og ligende scrips sånn at det blir lettere å løse utslagene.

Frustrasjoner

- Utdatert nettside som er klønete å håndtere.
- Bekymret for sikkerhet med muligheter for mange mulige scripts i alarmsystemet.
- Irriterende å ikke kunne se tilbake på hvem som har gjort endringer i alarmer og gjort fixes.

Plattformer

 Web

 Mobil App

 Wallboard

Figur 11: Persona Michelle, IT

Ut ifra idéforedlingen og analysene ble det utviklet personas basert på ansatte i IT og Billing som nevnt, og det ble tatt utgangspunkt i hvilke arbeidsoppgaver de hadde. Gruppen opplevde gjennom samtaler og analysen at det var forskjellige behov på tvers av avdelingene, og at det var naturlig å skille dem fra hverandre. Fra designprosessen fikk gruppen et tydeligere bilde av hvordan systemet ville bli utformet og hvilke funksjoner det var behov for. Det ble da lettere å begynne med low fidelity wireframes og fremstille design forslagene visuelt.

3.4.2 Felles grunnlag med shadcn/ui og Tailwind

Fra tidligere erfaringer med utvikling av web-applikasjoner har Uutilsynet.no og WCAG blitt brukt til å jobbe med standardiserte krav til universell utforming (UU). Kravene inkluderer blant annet tekststørrelser, fargekontraster og grunnleggende støtte for skjermleser. I samtaler med Phonero hadde de ikke UU som krav, men “Nice to have” ettersom Defcon prosjektet skulle være til et internt produkt. Samtidig så gruppen på det som fremtidig læring og “best practice” å inkludere UU så tidlig som mulig.

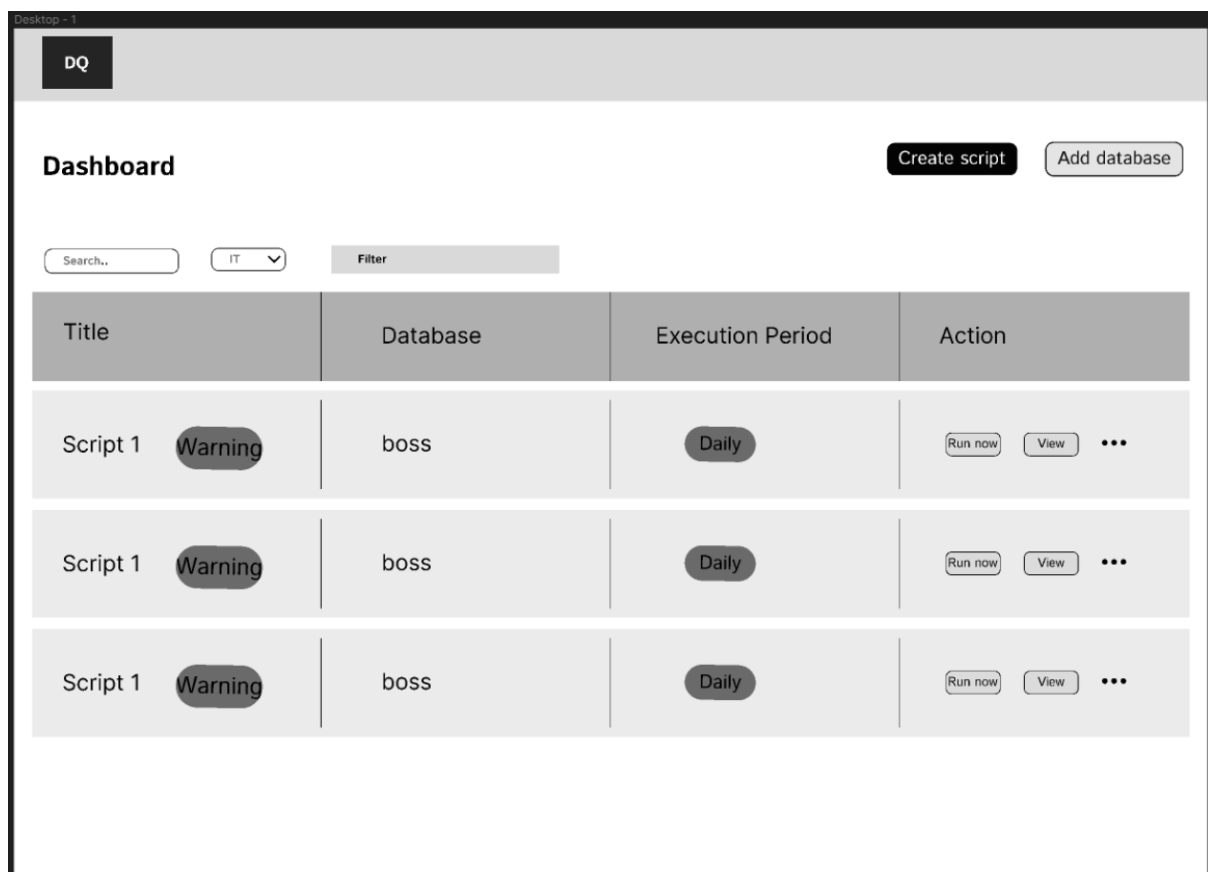
For å samkjøre arbeidet med design av systemarkitektur og universelt utformet brukergrensesnittet, ble det enighet om å bruke komponentbiblioteket shadcn/ui. Gjennom PACT analysen, spesielt delen om “Technologies”, fikk gruppen en grundig oversikt over hvilke teknologier som kunne ha innflytelse i utformingen av Defcon. Biblioteket var en styrende faktor for utformingen av komponenter til brukergrensesnittet som knapper, tabeller og overordnet innhold, samtidig tillot utforskning og redesign. Kombinasjonen med Tailwind sin “styling” både med tekst, farger, former og ferdigstilte komponenter (shadcn/ui) tillot det grundig kartlegging av fordeler og begrensninger tidlig i prosjektet.

Gruppen diskuterte fordelene av å jobbe med abstraksjoner, og at shadcn/ui ville spare tid på innledende utforming av brukergrensesnittet og implementering av kode. Samtidig ville det gi fordeler av å være i forkant av en felles opplæring i gruppen, utforske kjente rammeverk og jobbe med god dokumentasjon. Gruppen så positivt på abstraksjonsnivåene som kom av React arkitekturen og shadcn/ui-komponentene. De ga tilgang til kildekoden og mulighet til å endre på det som det var behov for. Samtidig tillot det raskere implementering av komponentene og det å sette opp grunnleggende funksjonalitet med relevante eksempler å henvende oss til.

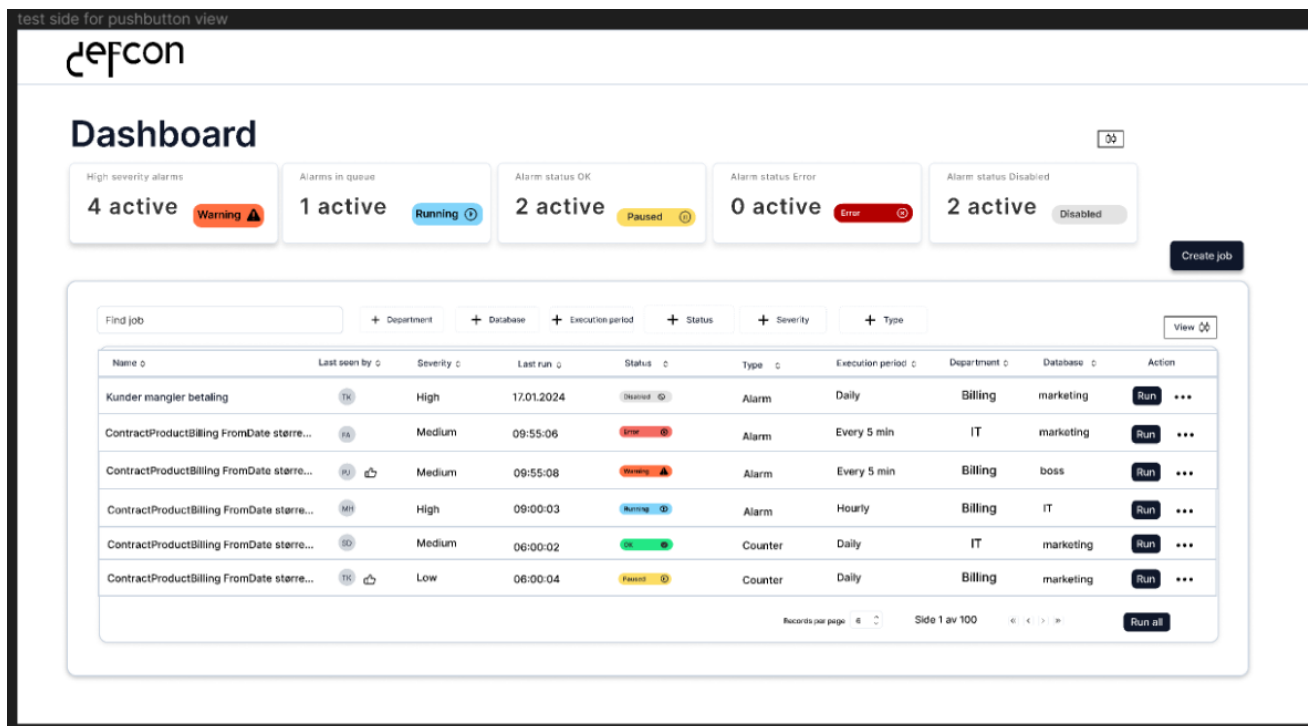
Som en følger av dette kom gruppen til enighet om at kvaliteten på produktet ville øke, ettersom shadcn/ui sine komponenter har grunnleggende funksjonalitet for universell utformede komponenter til en web-applikasjon. I tillegg har komponentbiblioteket et profesjonelt design som igjen blir styrket av Tailwind CSS sin styling av brukergrensesnittet. Shadcn/ui ga et sterkt fundament å bygge videre på i form av high fidelity wireframes, prototyping og implementasjon.

3.4.3 Wireframes og analyser

Deltakende design, gruppediskusjoner og analysene satte standarden for hvordan designprosessene til gruppen skulle gjennomføres. Utgangspunktet for diskusjon med produkteier og Phonero var en samling av low fidelity wireframes som gruppen hadde utarbeidet. Low fidelity ga et enkelt design som tok utgangspunkt i de ideene som hadde blitt foredlet og plassert i MoSCoW.



Figur 12: Dashboard Low Fidelity Wireframe



Figur 13: Dashboard High Fidelity Wireframe

Design verktøyet Figma ble tatt i bruk, noe som gjorde det lett å holde alt på ett sted, og samtidig sørge for at hvert medlem enkelt kunne ta del i designprosessen. Ettersom planen var å gå fra low fidelity til high fidelity var det viktig å jobbe systematisk med tilbakemeldinger og strukturere arbeidsflyt som en gruppe på seks personer. Gruppen delte seg inn i forskjellige “frames” og gikk løs på design forslag basert på diskusjoner, tilbakemeldinger og begrensninger vi hadde analysert tidligere. Gruppen opplevde en fin arbeidsflyt med design og kompetansehevning, samtidig som enkeltpersoner hadde et overordnet ansvar for å se over utkastene. Sammen ble det avklart hvilke design som skulle forkastes og hvilke som skulle videreutvikles.

Gruppen så det også nødvendig å ha en kontinuerlig diskusjon angående design med produkteier og ansatte Phonero for å sikre kvalitet. Overgangen til high fidelity wireframes dro også nytte av dette og prosessen var sømløs i stor grad. Som figur 12 tydeliggjør, har low fidelity wireframes inspirert high fidelity designet. Etterhvert som det også ble produsert flere high fidelity wireframes for andre sider som skulle implementeres, så ble det vektlagt interaksjoner mellom sidene for å utvikle en mer fullverdig prototypen. Designvalg og funksjoner som prioriteres som “Could have” i MoSCoW kommer ikke like tydelig frem i

prototypen, men var viktige samtaler i sprint reviews og demo møter med produkteier og ansatte.

3.4.4 Prototype

Underveis som high fidelity utkastene ble ferdigstilt og interaksjonene falt mer på plass, utformet gruppen en horisontal prototype. En horisontal prototype er med å vise bredden av systemet som skulle representere gruppens tanker, og utforming av funksjoner godt nok uten at alt trenger å være ferdigstilt (Sandnes, 2022, s.294). Det var ønskelig å komme i gang utvikling av et kodet "skall" for applikasjonen, noe som vil tillate fremvisning til sprintene og en mer strukturert arbeidsmetodikk. Dette var også viktig å få frem ettersom designprosessen allerede hadde tatt mye tid av de første sprintene. Etter en demo som gruppen presenterte for Phonero hvor de var svært fornøyde med det de fikk se, kom de også med konstruktive tilbakemeldinger om hva som kan gjøres annerledes. Derav fortsatte prosessen med deltagende design, ettersom Phonero kom med nye ideer som måtte vurderes og tas i betraktning videre.

Designprosessen har alltid blitt sett på som flytende med design “thinking”, noe den agile Scrum arbeidsmetodikken har tillatt. Det har medført at tid og kvalitet har trengt å bli veid opp mot hverandre. Blant annet har det gått mye tid til idéforedling og å produsere en god MoSCoW og PACT analyse. Likevel har dette vært avgjørende funksjoner som har forbedret kvaliteten selv om det har krevd mye tid. Det er tilbakemeldinger som hovedsakelig har gjort designprosessen flytende siden gruppen har tatt for seg nye ideer med behov fra Phonero for å finne ut om de var verdt tiden som trengtes for å gjennomføre. Et eksempel er at gruppen sammen med Phonero kom frem til at mobilvisning var nedprioritert til "Won't have". Dermed ble tiden investert her høy i forhold til kvalitetsaspektet ettersom det falt helt bort. Gruppen merket også at Figma hadde spist mye tid i de tidlige fasene. Det har senere blitt diskutert om det ble brukt for mye tid på små ting i Figma som ikke nødvendigvis ga utslag for kvaliteten, men det er ikke lett å konkludere med. Et tema som har blitt diskutert i ettertid er om tid og kvalitet vokser parallelt med hverandre. Vår opplevelse er at ettersom man dykker dypere i enkelte prosesser så vokser tid, mens kvaliteten stagnerer.

3.5 Testing

Designprosessen som er beskrevet i 2.3 og 3.4 har vært tidkrevende, men ytterst nødvendig. Testing i ulike former har vært med på å sikre designet og produktets kvalitet. Gjennom demoer og møter med Phonero har gruppa fått testet produktet med de som skal ta det i bruk. Deres evaluering har vært essensielt for å bygge videre og ferdigstille funksjoner og designvalg. Den kontinuerlige testingen fungerte godt i praksis og kom naturlig ettersom sprintene ble fullført og teamet holdt review møter med fremvisning av demo, prototyper og utkast til web-applikasjonen. Utover i sprintene kunne gruppen også vise et mer fullverdig produkt. Dette ga verdifull input på hva som ikke var ønsket å utvikle eller hva som burde implementeres. Produkteier satte også opp et fellesmøte med IT og Billing hvor gruppa skulle ha en demo av MVP'en «Defcon». Demoen ble møtt med god feedback og positivitet, samt noen forslag fra ansatte. Deretter ble det lagt en plan for å sende ut en survey sammen med Defcon slik at flere ansatte kunne prøve seg i det nye designet.

Selv om den kontinuerlige testingen ga gruppa mye bra, ble det likevel nødvendig å kjøre standardiserte brukertester for å ta i bruk et rammeverk som kan gi en pekepinn på sammenhengen i Defcon. Når man har designet et system er det ikke lett å se hva som ikke fungerer ettersom man ofte vet hvor alle funksjoner er plassert. For å gjennomføre brukertesting ble en test guide som inneholdt syv scenarioer utarbeidet. Scenarioene var basert på hvordan Defcon skal bli brukt og disse skulle bli gjennomført fra A til B av to ansatte i IT og to i Billing. Den første ansatte som gjennomførte testen var fra IT og etter brukertesten fikk gruppen etablert hva som burde endres i fremgangsmåten til testingen. En endring tillot testerne å “leke” seg litt inne på Defcon før gjennomgangen av scenarioer ble gjennomført, noe som reduserte tidsbruken per oppgave drastisk bare etter et par minutter med frie tøyler. Et scenario som virkelig ble påvirket var hvor du skulle filtrere ut ifra push-buttons. Under brukertesting prøvde testerne å ha en god samtale med de som gjennomførte. Det var viktig for testerne å etablere hva brukernes tanker om Defcon var, og fremgangsmåtene de hadde når de navigerte seg rundt scenarioene.

Selv om scenarioene var den største delen av brukertesting, ble det også gjennomføre en System Usability Scale (SUS)-test. Dette skulle være med på å gi et enda klarere bilde på kvaliteten av Defcon. Som nevnt i del 2 er SUS et standardisert skjema, gruppa, så det likevel

nødvendig å gjøre noen endringer. Disse endringene skulle gjøre at skjemaet passet bedre vårt produkt og brukertest.

System Usability Scale, 1 IT	Helt enig 4	3	2	1	Helt uenig 0
Jeg opplevde systemet som unødvendig komplisert					x
Jeg synes systemet var enkelt å bruke	x				
Jeg tror jeg trenger hjelp for å bli bedre kjent med systemet	x				
Jeg synes at de forskjellige funksjonene i systemet er godt integrert	x				
Jeg synes det var dårlig sammenheng i systemet					x
Jeg tror de fleste vil lære seg å bruke systemet fort	x				
Jeg synes systemet var tungvint å bruke					x
Jeg følte meg selvsikker når jeg brukte systemet		x			
Jeg tror dette systemet vil funke bedre enn det gamle	x				

Figur 14: Brukertest "System Usability Scale" første utkast

Det første utkastet av skjemaet figur 14, men dette ble endret før de tre neste brukertestene ettersom det skulle tillate lettere utregning av poengsummen til SUS senere. Endringene som ble lagt til var et spørsmål samt omrokkere litt på de andre slik at de skulle følge samme oppsett med tanke på en positiv og negativ struktur (Sandnes, 2022, s.324).

System Usability Scale, 1 Billing	Helt enig 4	3	2	1	Helt uenig 0
Jeg har lyst til å bruke dette systemet	x				
Jeg opplevde systemet som unødvendig komplisert					x
Jeg synes systemet var enkelt å bruke	x				
Jeg tror jeg trenger hjelp for å bli bedre kjent med systemet		x			
Jeg synes at de forskjellige funksjonene i systemet er godt integrert		x			
Jeg synes det var dårlig sammenheng i systemet					x
Jeg tror de fleste vil lære seg å bruke systemet fort	x				
Jeg synes systemet var tungvint å bruke					x
Jeg følte meg selvsikker når jeg brukte systemet	x				
Jeg liker bedre det eldre systemet					x

Figur 15: Brukertest “System Usability Scale” andre utkast

Brukertestingene ga gruppa mye innsikt i form av interaksjoner og sammenhenger i systemet. Selv om tilbakemeldingene i stor grad var positive, så ble det enighet om å gjøre noen endringer basert på gruppas helhetsinntrykk av testene. Som nevnt tidligere ble funksjonen som tillot det å legge til databaser og departments flyttet til en egen manage-button for å forhindre forvirring rundt navigering. Det ble også foretatt noen endringer på utformingen av farger på skrift for å sikre bedre synlighet i systemet. System Usability Scale viser også at kvaliteten på systemet slår høyt ut. Ved å bruke uttrykket $P + (20 - N)$ kan det regnes ut at gjennomsnittet av SUS testen ble 35 på de tre testene som brukte andre utkastet av SUS (Sandnes, 2022, s.324). Dette ansees for å være et veldig høyt snitt og styrker derfor troverdigheten av kvaliteten på Defcon ytterligere. Likevel er gruppa klar over at ulike feilkilder kan ha vært med på å spille inn. Blant annet kan vårt nærvær under testingen være med på at de vil naturlig gi flere positive kommentarer.

4. Produktet

Dette kapitlet tar for seg en gjennomgang av produktet vi har utviklet og hva den tilbyr, samt hvordan gruppen håndterer overleveringsprosessen av produktet til Phonero for videre bruk og utvikling.

4.1 Defcon

Tidligere beskrev gruppen Phonero sitt gamle verktøy, DQM, og viste til at verktøyet består av utdaterte teknologier og utdatert UX, men at den har vært robust og pålitelig. Gruppens oppgave var å oppdatere DQM med bruk av ny teknologi og en bedre brukeropplevelse. Defcon er en stor forbedring av DQM og er innstilt på å være en stabil og pålitelig erstatning med en mange UX-forbedringer og nye funksjoner for å øke livskvaliteten til brukerne av verktøyet, samt bruken av en moderne teknologistack som beskrevet i kapittel 2.4 Med bakgrunn i grundige analyser av krav fra ansatte og produkteier har gruppen utviklet en applikasjon som tar de kvalitetene DQM spiller godt på, og bygger de videre til en moderne standard.

Defcon er et system som med utgangspunkt i forgjengeren DQM. Selv om flere av funksjonene er videreført, er hele applikasjonen skrevet fra bunnen. Det nye systemet innehar minimum samme funksjonalitet som DQM. En bruker kan opprette en jobb (tidligere kalt et “script”) som enten er en alarm eller statistikk, skrive en SQL-spørring for jobben og bestemme hvor ofte den skal kjøre. Defcon vil i tillegg håndtere regelmessig kjøring av jobben og alarmutløsning dersom det finnes dataintegritetsfeil, noe som resulterer i en lettere og mer oversiktlig prosess. Som nevnt har vi forbedret eksisterende funksjoner og implementert nye. Videre viser vi til en funksjonsliste som nevner det viktigste for vår MVP, både av nye funksjoner og forbedringer. Vi tar for oss noen generelle aspekter av produktet og beskriver hvordan de har blitt forbedret eller implementert, med MoSCoW-analyse som grunnlag.

Nye funksjoner	Forbedret eksisterende funksjoner
• Tabelloversikt av jobber med og sortering og forbedret filtrering • Mer informasjon om en jobb ◦ Fikser til alarmer ◦ Relaterte jobber ◦ Tags ◦ Linker • Legg til/fjern databaser og avdelinger i nettsiden • Dark mode • Wallboard mode • Acknowledge • Aktivitetslogg	• Navigering til forskjellige deler av nettsiden har blitt enklere • Bedre oversikt over status på jobber • Synkronisering av alarmkjøringstider • Informasjon og handlinger er lettere og raskere tilgjengelig

Figur 16: Funksjonsliste

En av de større forbedringene har vært en helt ny oversiktlig forside som inneholder en tabell av alle eksisterende jobber med ny informasjon. I bildet under ser du hovedsiden “Dashboard” med noen jobber lagt til. Her ser du blant annet jobbens status, når jobben sist kjørte og hvilken database den kjører mot. I tillegg kan du filtrere og sortere jobber i tabellen.

The screenshot shows the Defcon Dashboard interface. At the top, there's a header with the Defcon logo, a 'Manage' button, and a user profile icon 'SD'. Below the header, the title 'Dashboard' is displayed. There are two filter boxes: 'Filter Jobs with Status: 0 Running' and 'Filter Jobs with Status: 0 Alarm'. A 'Create Job' button is on the right. Below the filters, there's a table with columns: Name, Acknowledged by, Status, Severity, Last run at, Result, Database, Department, and Actions. The table contains three rows of job data. At the bottom, there's a pagination bar showing '0 of 3 row(s) selected', 'Rows per page 20', and 'Page 1 of 1'.

Name	Acknowledged by	Status	Severity	Last run at	Result	Database	Department	Actions
Aktive kunder	TN	Error	Medium	Today at 10:53:18	0 rows	dev_db	IT	Run Edit
Betalinger med manglende datoer		OK	Medium	Today at 10:55:44 (0.03 seconds)	0 rows	dev_db	IT	Run Edit
SIM betalt men deaktivert		Disabled	Medium	2024-05-07 12:46:46	0 rows	dev_db	IT	Run Edit

Figur 17: Defcon Dashboard

En annen viktig side av Defcon er jobbvisningssiden som viser brukere all tilgjengelig informasjon om jobben. Denne siden viser jobbens status, ekstrainformasjon som fikser, dokumentasjon og relaterte jobber, samt en tabell for resultatet av SQL-spørring. Det vises også en aktivitetslogg hvor alle bruker -og systemmeldinger blir loggført og vist fra det nyeste til eldste. Siden inneholder også flere handlinger, som “Run now” for å tvinge en jobb til å kjøre, eller “Edit” for å redigere en jobb.

defcon Manage SD

ALARM

Aktive kunder Alarm

Last ran at 2024-05-08 12:17:41 • dev_db • IT • Hourly

Alarm
At 2024-05-08 12:17:41, the script returned **3 rows**

Overview Fixes 0 Documentation 0 Related jobs 0

Description ^

asd

SQL v

Run now Acknowledge Edit ...

Activity

Edited Job was updated • x2
 • TODO • 2024-05-08 12:17:38
 Edited by TODO

Error Unexpected error
 • System • 2024-05-08 12:07:13
 Incorrect syntax near 'TestData'.

Edited Job was updated
 • TODO • 2024-05-08 12:07:10
 Edited by TODO

Error Unexpected error
 • System • 2024-05-08 12:05:17
 Could not find stored procedure 'asd'.

View

ContractId	ContractTypeDescription	ProductName	RegistrationDate
1	Mobil	Test	2020-10-05T00:00:00
2	Mobil	Test	2020-10-05T00:00:00

Figur 18: Defcon Job View

Satt opp mot DQM er Defcon en betydelig forbedring. I introduksjonen av DQM i kapittel 1.2.2 vises det til et skjermbilde av den gamle forsiden, og sammenlignet med bildene vist over har produktkvaliteten økt betraktelig. Sammen med de visuelle endringene, har også Defcon ført til forbedringer internt med bruk av nye teknologier. Defcon presenteres som et robust og pålitelig produkt, noe som var målet for prosjektet fra starten.

4.2 Overlevering

Scrum målet for bachelorprosjektet har vært ambisiøst og omfattende. Å utvikle en MVP, med mer funksjonalitet enn en applikasjon som har overlevd siden 2008, var en utfordring. Samtidig har vi jobbet konsekvent med oppgaven og kvalitetssikret prosjektet etter beste evne. Som en del av den avsluttende prosessen ønsket gruppen å avholde en overlevering fase og avklare med produkteier hvordan det passet best for Phonero. Ettersom planen for produktet etter at vi er ferdig vil bli forbedret og integrert inn i Dev-miljøet til Phonero, ønsker vi å sette opp en plan med fokus på prosjektoverføring som inkluderer dokumentasjon over overdragelse og vedlikehold.

Med utgangspunktet til Karlsen (2021) sitt kapittel 15 "Avslutning og gevinstrealisering" hadde vi en forlenget sprint retrospektiv med produkteier (s.513-533). Der diskuterer vi prioriteringer i den avsluttende fasen av prosjektet og overleveringsprosessen. Under møtet kom det fram at implementeringen av automatisk testing var noe produkteier ikke anså som nødvendig. Han ønsket at gruppen skulle ferdigstille de avtalte funksjonskravene, og deretter prioritere retting av bugs. Etter felles enighet om kodeløstopp 1.Mai, skulle gruppen fokusere på å skrive godt dokumenterte arbeidsoppgaver relatert til overlevering inn i Jira. Vi snakket om hvordan dette skulle fremlegges best mulig. Et eksempel på en godt dokumentert arbeidsoppgave skulle inneholde en relevant tittel, en detaljert oppgave- og problembeskrivelse, samt annen generell informasjon som er viktig for utførelsen av oppgaven.

Mange av teknologiene brukt i løpet av prosjektet er kjent for Phonero ettersom de tar de i bruk selv. Likevel er det viktig å dokumentere de andre teknologiene vi har tatt i bruk for å kjøre prosjektet og videreutvikle det i Dev-miljø. Dette vil også medføre mindre arbeid for teamet som skal ta over da de kan lese en brukermanual på hvordan Defcon er utviklet fra bunnen av. Ettersom Phonero er eksperter på området ønsket de ikke omfattende dokumentasjon, men overordnet innsyn i hvordan vi har tenkt oss frem til logikk og implementasjoner.

5. Refleksjon og lærdom

Dette kapittelet tar for seg refleksjon rundt erfaringer fra prosjektet. Avslutningsvis konkluderer gruppen om prosjektet har vært en suksess.

5.1 Erfaringer fra prosjektet

Gjennom prosjektløpet har gruppen som en helhet og som individer hatt en stor vekst og utvikling i forbindelse med prosjektstyring, dokumentering, analysering, rapportering og koding. Det har vært en bratt læringskurve som har vært givende i form av nye ferdigheter og kunnskap, men også fra erfaringer med problemløsning og håndtering av endringer som har oppstått.

Selv om gruppen tidligere hadde kjennskap til prosesser og rammeverk som er nyttige i konteksten av utviklingsprosjekter, var det gjennom praktisk erfaring at betydningen og nyttiligheten de hadde ble tydeliggjort. Gruppens evne til å tilpasse seg til, og reagere på ulike endringer og vendinger ble styrket i løpet av hver sprint. Læringsutbyttet av dette var i henhold til bruken av analyser og rammeverk. Erfaringer fra hvordan det hjelper å håndtere eventuelle utfordringer, som oppstår når man utvikler et prosjekt av dette omfanget.

Problemløsning og kommunikasjon har vært sentrale temaer i arbeidet med prosjektet. Det har blitt oppnådd en bedre forståelse for viktigheten av å ha klare og åpne plattformer for kommunikasjon både internt i gruppen og med oppdragsgiver. Gjennom diskusjoner, samarbeid og deling av erfaringer og kompetanse har gruppen klart å håndtere utfordringer som har oppstått på en strukturert og god måte. Dette ble gjort ved å bruke de etablerte analysene som risikoanalysen for å ha klare tiltak som kan iverksettes og Scrum rammeverket som har åpnet for en sikker plattform for kommunikasjon. Det har for øvrig blitt praktisert en agil arbeidsmetodikk som skal bidra til å gjøre gruppen mer tilpasningsdyktig.

En annen utfordringer gruppen har møtt på var en felles enighet om at kodeløp skulle tre i kraft 1. mai. Det ble vanskelig å holde ettersom live-versjonen av Defcon fikk byggefeil og det måtte løses raskt så systemet kunne brukes og testes i overleveringsfasen. Derav ble det en del ekstra arbeid hvor gruppa var avhengig av hjelp fra Phonero ettersom vi hadde lite kjennskap med byggefeilen som Defcon fikk. Dette går også inn i en annen utfordring i

henhold til hjelp. At det ofte tok lengre tid enn vi beregnet å få hjelp til utfordringer vi ikke har kan løse selv. De ansatte i Phonero har sin egen arbeidsdag og må prioritere sine problemer først. Sett positivt på utfordringsbildet er at vi blir tvunget til å være mer agile og tilpasse oss nye oppgaver for å dra prosjektet videre. Samtidig har vi måtte være klar til å omstille oss veiledningsmøter for å så løse utfordringen selv.

5.2 Konklusjon

I prosjektstyringen av Defcon har vi hatt et fokus på kvalitetssikring og vært bundet til vurderinger av kvalitet, tid og kostnader. Vi bestemte oss sammen med Phonero for å vurdere suksessen av produktet med *“Quality as agreed”*, og at suksesskriteriet for produktutviklingen ville være å ferdigstille en MVP basert på MoSCoW.

For å måle kvaliteten på produktet har vi gjennom analyser, design, samtaler og testing fått tilbakemeldinger på implementasjon, interaksjon og generelt innhold. Tilbakemeldingene fra Phonero sine ansatte og produkteier har vært positive. MVP ble ferdigstilt innenfor prosjektets tidsramme. Ressursfordeling med tanke på kostnader for oss og Phonero, samt læringsutbyttet er fungert godt innenfor prosjektrammene og vi opplever å ha gjennomført etter beste evne.

Suksess kan også måles i andre former utover kvalitet, tid og kostnader. For eksempel har vi vektlagt suksessen av å overlevere produktet med dokumentasjon, der teknologivalg, implementasjon og ufikset bugs er godt beskrevet. Ut ifra samtaler, brukertesting og overleveringsprosessen av Defcon vurderer vi utgangspunktet for overlevering som godt. Teknologiene er lignende miljøet til Phonero og kodestandard skal være innenfor norm. Vi mener derfor at Phonero er godt utrustet til å overta prosjektet og integreringen av Defcon i sitt Dev-miljø. Nytteeffekten kan potensielt bli at DQM blir helt erstattet av Defcon og at applikasjonen er den nye standarden for Phonero sitt alarmsystem.

Vi kan konkludere med at dette semesteret har vært ekstremt givende. Det har vært en smakebit på arbeidsmarkedet hvor mye kunnskap har blitt tilegnet på relativt kort tid hvor hvert gruppemedlem har fått spesialisert seg etter sitt ønsket arbeid. Å være seks stykker i én gruppe har vist seg å fungere godt om man samhandler og utarbeider en god prosjektplan. Her har også ferdighetene til hvert gruppemedlem spilt inn til å skape en god helhet. I tillegg har en fast rutine hatt en positiv innvirkning på kvaliteten i produktet. Ut ifra disse kriteriene vil vi kunne si at prosjektet har vært en suksess.

6. Litteraturliste

Amazon. (u.å.) *What is an API? - Application Programming Interface Explained - AWS*.

AWS. Hentet 16. April, 2024, fra <https://aws.amazon.com/what-is/api/>

Atlassian. (u.å.). *Welcome to Jira*. Atlassian Jira. Hentet 2. Mai, 2024, fra

<https://www.atlassian.com/software/jira/guides/getting-started/introduction#dig-into-specific-features>

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

Elassy, N. (2015, Juli 6). *The concepts of quality, quality assurance and quality enhancement*.

Emerald. Hentet April 16, 2024, fra

<https://www.emerald.com/insight/content/doi/10.1108/QAE-11-2012-0046/full/html>

Figma. (u.å.). *Free Design Tool for Websites, Product Design & More*. Figma. Hentet

17. April, 2024, fra <https://www.figma.com/design/>

Hangfire. (u.å.). Hangfire – Background jobs and workers for .NET and .NET Core. Hentet

17. April, 2024, fra <https://www.hangfire.io/>

IDEO. (u.å.). *DESIGN THINKING DEFINED*. IDEO DESIGN THINKING. Hentet 06.05.

2024, fra <https://designthinking.ideo.com/>

Interaction Design Foundation. (2017, Mars 17). *What is Participatory Design?* Hentet 17.

April, 2024, fra

<https://www.interaction-design.org/literature/topics/participatory-design#:~:text=Participatory%20design%20is%20a%20collaborative,applying%20their%20knowledge%20and%20experiences>.

Karlsen, J. T. (2021). *Prosjektledelse - fra initiering til gevinstrealisering* (5th ed.).

Universitetsforlaget.

Kathuria, M. (u.å.). *A Guide to Using the Fibonacci Sequence in Scrum* | Resource Library.

Scrum Alliance Resources. Hentet 30.April, 2024, fra

<https://resources.scrumalliance.org/Article/guide-using-fibonacci-sequence-Scrum>

Køster, C. (2022, Mars 3). *Slik gjør du enklere prioriteringer ved bruk av MoSCoW-metoden*.

Smidigbloggen. Hentet 16.April, 2024, fra

<https://www.smidigbloggen.no/slik-gjor-du-enklere-prioriteringer-ved-bruk-av-moscow-metoden>

Martin, J., Elg, M., & Gremyr, I. (2020, November 15). (PDF) *The Many Meanings of Quality: Towards a Definition in Support of Sustainable Operations*. ResearchGate.

Hentet 16.April, 2024, fra

https://www.researchgate.net/publication/345942395_The_Many_Meanings_of_Quality_Towards_a_Definition_in_Support_of_Sustainable_Operations

Microsoft. (u.å.-a). *What is ASP.NET Core?* | .NET. Microsoft DotNet. Hentet 16.April, 2024, fra <https://dotnet.microsoft.com/en-us/learn/aspnet/what-is-aspnet-core>

Microsoft. (2023c, Oktober 24). *Visual Studio*. What is Visual Studio?

<https://learn.microsoft.com/nb-no/visualstudio/get-started/visual-studio-ide?view=vs-2022>

Microsoft. (2024b, Februar 14). *What is SQL Server?* Microsoft Learn. Hentet 29.April, 2024, fra

<https://learn.microsoft.com/en-us/sql/sql-server/what-is-sql-server?view=sql-server-ver16>

OpenAI. (u.å.). *What is ChatGPT?* OpenAI Help Center. Hentet 16.April, 2024, fra

<https://help.openai.com/en/articles/6783457-what-is-chatgpt>

Oracle. (u.å.). *What Is a Relational Database*. Oracle. Hentet 29.April, 2024, fra

<https://www.oracle.com/database/what-is-a-relational-database/>

Peek, S. (2023, December 20). *What Is Agile Scrum Methodology?* - *businessnewsdaily.com*.

Business News Daily. Hentet 25. April, 2024, fra

<https://www.businessnewsdaily.com/4987-what-is-agile-Scrum-methodology.html>

Phonero. (u.å.-a). *Om Phonero*. Phonero. Hentet 23. April, 2024, fra

<https://www.phonero.no/informasjon/om-phonero>

Phonero. (u.å.-b). *Produkter og tjenester*. Phonero. Hentet 14. Mai, 2024, fra

<https://www.phonero.no/produkter-og-tjenester>

Pollack, J., Helm, J., & Adler, D. (2018, April 13). *What is the Iron Triangle, and how has it changed?* Emerald. Hentet April 17, 2024, fra

<https://www.emerald.com/insight/content/doi/10.1108/IJMPB-09-2017-0107/full/html>

Radigan, D. (u.å. -a). *What is a Stand Up Meeting? [+ Expert Tips]*. Atlassian. Hentet 17.

April, 2024, fra <https://www.atlassian.com/agile/Scrum/standups>

Radigan, D. (u.å. -b). *Mastering Agile Sprint Reviews: A Guide for Agile Teams*. Atlassian.

Hentet 17. April, 2024, fra <https://www.atlassian.com/agile/Scrum/sprint-reviews>

React. (u.å.). *React – A JavaScript library for building user interfaces*. Hentet 16. April,

2024, fra <https://legacy.reactjs.org/>

Rehkopf, M. (u.å.). *Scrum Master Roles and Responsibilities*. Atlassian. Hentet 17. April,

2024, fra <https://www.atlassian.com/agile/Scrum/Scrum-master>

Sandnes, F. E. (2022). *Universell utforming av IKT-systemer* (3rd ed.). Universitetsforlaget.

Schwaber, K. (u.å. -a). *What is a Daily Scrum?* Scrum.org. Hentet 6. Mai, 2024, fra

<https://www.Scrum.org/resources/what-is-a-daily-Scrum>

Schwaber, K. (u.å. -b). *What is a Sprint Retrospective?* Scrum.org. Hentet 26. April, 2024, fra

<https://www.Scrum.org/resources/what-is-a-sprint-retrospective>

Shadcn/ui. (u.å) *shadcn/ui documentation*. shadcn/ui. Hentet 16. April, 2024, fra

<https://ui.shadcn.com/docs>

Stack Overflow. (2023, Juni 13). *Stack Overflow Developer Survey 2023*. Developer Survey.

Hentet 17. April, 2024, fra

<https://survey.stackoverflow.co/2023/#most-popular-technologies-new-collab-tools>

Tailwind CSS. (u.å.). *Tailwind CSS - Rapidly build modern websites without ever leaving*

your HTML. Hentet 16. April, 2024, fra <https://tailwindcss.com/>

TypeScriptLang. (u.å.). *Why does TypeScript exist?*. TypeScriptLang. Hentet 16. April, 2024,

fra <https://www.typescriptlang.org/why-create-typescript>

University of Central Arkansas. (u.å.). *Chat GPT: What is it?* University of Central Arkansas.

Hentet 16 April, 2024, fra <https://uca.edu/cetal/chat-gpt/>

Visual Studio (u.å.). *Documentation for Visual Studio Code*. Visual Studio Code. Hentet

17. April, 2024, fra <https://code.visualstudio.com/docs>

West, D. (u.å.). *Sprint Planning*. Atlassian. Hentet 17. April, 2024, fra

<https://www.atlassian.com/agile/Scrum/sprint-planning>

W3Techs. (u.å.). *Usage Statistics and Market Share of React for Websites*. W3Techs. Hentet

16. April, 2024, fra <https://w3techs.com/technologies/details/js-react>

Yasar, K. (u.å.). *What is Risk Analysis? | Definition fra TechTarget*. TechTarget. Hentet 16.

April, 2024, fra <https://www.techtarget.com/searchsecurity/definition/risk-analysis>

Figurliste

Figur 2: Leong, T., Zakuan, N., Saman, M., Md Ariff, M. S., & Tan, C. (2014). *Using project performance to measure effectiveness of quality management system maintenance and practices in construction industry*. TheScientificWorldJournal, 2014, 591361.

<https://doi.org/10.1155/2014/591361>

Figur 4: Regjeringen. (2016). *Risikomatrise* [Figur].

<https://www.regjeringen.no/no/dokumenter/nou-2016-19/id2515424/sec3?q=nou%202016:19>

7. Vedlegg

7.1 Uttalelse fra Phonero

Kristiansand 15.05.2024

Til rette vedkommende

phonero

Studentene Fredrik Aspelien, Tobias Kroknes Kristiansen, Simon Duarte Andersen, Pål Folmer Jahren, Martin Hommeland Monsen og Markus Johansen utførte sitt Bachelor prosjektet hos oss i Phonero. Vi ønsker i den forbindelse å skrive en uttalelse om hva studentene skulle levere samt vår opplevelse av måten gruppen leverte på.

Opgaven gruppen fikk fra oss omhandlet i korte trekk å erstatte et eksisterende system som hjelper oss å fange opp inkonsistente data i databasen samt feilsituasjoner i databasen. Dette er et internt verktøy for avdelingene IT og Billing. Oppgavespesifikasjonen omhandler å analysere dagens funksjonalitet i gammelt system samt gjøre videre undersøkelser for å avdekke behov som systemet også kan og bør dekke.

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

- Designet ny applikasjon – studentene hadde ingen restriksjoner eller konkrete krav på GUI fra vår side, men de måtte finne ut hva brukerne ønsket og levere på disse kravene
- De var i hovedtrekk begrenset til å bruke verktøyene og teknologier som vi besitter og bruker for at vi enklere skulle kunne ta over videreutviklingen
- Brukeravklaringer og brukertesting mot de avdelingene som skal ta i bruk ny applikasjon
- Utviklet fungerende MVP med Database, Tidsstyrte prosesser, API og Web-klient
 - Studentene valgte en teknologi for tidsstyrte kall som heter Hangfire, og denne har vi nå adoptert i andre applikasjoner på IT – meget godt funnet!
- Designet databasemodell for lagring av all informasjon på en fornuftig måte

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

Vi opplevde som nevnt at studentene jobbet meget selvstendig og de var veldig engasjerte med et sterkt ønske om at sluttproduktet skulle bli bra. I tillegg var de ikke redde for å være ærlige og åpne seg imellom og det skinte igjennom at de hadde et sterkt ønske om å lære så mye som mulig. Studentene var også bevisste på å gjøre gode retrospektiv møter etter hver sprint og fulgte dette opp som gruppe.

Vi har vært involvert i planleggingen, teknologivalg (hvor vi informert om hva vi bruker og kjenner). Vi har bidratt til å sette applikasjonen ut i live, det vil si at vi bidro til å bruke våre verktøy for å bygge og legge ut funksjonaliteten i produksjon, noe vi ikke så det hensiktsmessig at studentene skulle sette seg inn i. Vi deltok selvsagt også på statusmøter med studentene og koordinator fra UiA. I dette prosjektet var vi mer en ren produkteier, noe som er en god øvelse for oss slik at vi får mer innsikt i hvordan det er å være produkteiere. Selvsagt bidro vi med teknisk bistand på teknologiene vi kjenner.

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

Kristian Hagestad
Kristian Hagestad
Utvikler

Erik L. Berntsen
Erik Linga Berntsen
Teamleder og utvikler

Bård Eik
Bård Eik
Head of development

En bedre hverdag - for hele bedriften

7.2 ChatGPT

Under skriveprosessen brukte gruppen tidvis ChatGPT versjon 3.5 og 4 til tekstforbedring. Det ble blant annet brukt til forslag til strukturering og rettskriving. Forslagene til ChatGPT har bidratt til en bedre sammenheng i teksten i tillegg til å holde på en rød tråd.

7.3 Gruppeevaluering/individuelle bidrag

Gjennom prosjektet har gruppen vokst som både individuelle bidragsyttere og som en samlet enhet. Vi har anvendt Scrum som en plattform for struktur og kommunikasjon for å oppnå felles enighet og forståelse. Dette er spesielt nyttig når vi er i en stor gruppe med ulike meninger og perspektiver. Med forskjellige grader av erfaring og kompetanse har vi tatt nytte av hverandres styrker og oppnådd et godt samarbeid og er stolte av arbeidet vi har gjennomført. Alle har deltatt i prosjektstyring og fått forståelse for hele omfanget av prosjektet til tross for at vi har i forskjellig grad drevet med koding, analysering og rapportering. Den felles høye arbeidsmoralen har vært et godt fundament for evnen vår til å levere kvalitetsarbeid.

Fredrik Aspelien

Dette semesteret har jeg i bachelorprosjektet vært med på å drive prosjektet fremover og hatt en overordnet oversikt over prosjektstyring i design-, utvikling- og rapportskrivning fasen. Jeg har også hatt et ansvar for front-end design og utvikling, samt på å implementere funksjonalitet på dashboard siden og bistå gruppemedlemmer som sitter på uløste oppgaver i Jira.

Markus Johansen

Under dette prosjektet har mitt hovedarbeid vært utvikling av back-end funksjonalitet. Jeg har vært med på å lage samtlige endepunkter som blir brukt i blant annet dashbordet og andre sider i applikasjonen. Gjennom en 4 ukers periode hadde jeg rollen som Scrum master, hvor jeg bidro til generell prosjektstyring og opprettholding av kontakt mellom gruppen og produkteier.

Martin Hommelands Monsen

I prosjektet har min rolle i stor grad gått ut på utvikling av frontend skall og implementering av funksjonalitet. Jeg jobbet med skallet til jobview, manage og preference- sidene. Videre

implementerte jeg funksjonalitet i manage og preferences. Jeg har også kontinuerlig gjort mikrojusteringer på frontend for å sikre en god brukeropplevelse. Som alle medlemmene har jeg også bidratt til prosjektstyring og fremdrift gjennom rollen som Scrum master.

Pål Folmer Jahren

I bachelorprosjektet har jeg hovedsakelig bidratt med implementering av ulike frontend-løsninger. Dette har involvert deltakelse i designprosessen og utviklingen av både dashbordet og "create job"-siden. I tillegg til dette har jeg hatt en aktiv rolle i prosjektledelsen, både i mindre og større omfang, med ansvar for overvåkning og koordinering av prosjektets fremdrift. Som Scrum-master har jeg også hatt ansvaret for kommunikasjonen med produkteieren, og har utført denne rollen ved to separate anledninger.

Simon Duarte Andersen

Under bachelorprosjektet har jeg vært med på mye forskjellig. Som alle andre har jeg også vært med på å skyve prosjektet fremover ved å ta del i flere prosesser. Blant annet har jeg vært med på å legge til funksjoner i front-end som dark mode, vært med på å kode litt backend gjennom parprogrammering og tatt del i designprosessen. Jeg kan si at jeg har vært der hvor det trengtes. I tillegg har jeg fått prøve meg som Scrum-master hvor jeg blant annet fikk ansvar for styringsmøte samt de andre oppgavene en Scrum-master har. Jeg hadde også hovedansvar for brukertesting.

Tobias Kroknes Kristiansen

I dette prosjektet har jeg vært teknisk ansvarlig for utviklingsprosessen og implementeringen, og har bidratt til gjennomførelsen av frontend og backend. Jeg har vært til bisto for gruppemedlemmer og hatt ansvar for godkjenning av pull-request kode som skulle legges til. I starten av prosjektet var jeg innledende Scrum-master og jobbet mye med design, både low-fidelity og high-fidelity.

7.4 Scenarier til brukertesting

Hva som skal gjennomføres

For å gjennomføre denne brukertesting skal du gå gjennom noen scenarier som er knyttet til DEFCON. Disse scenariene er basert på ulike oppgaver og funksjoner som skal eller kan bli brukt.

Scenario 1

Du skal finne alarmen “kunde mangler betaling” og sette deg selv som acknowledge på alarmen slik at folk kan se at du skal jobbe med den. På samme alarmen skal du apply en fix.

Scenario 2

Det er behov for å opprette en ny database og du får oppgaven å gjøre det. Naviger deg frem ved hjelp av profiles tabben slik at du kan opprette en database.

Scenario 3

Du er ikke en stor fan av hvordan light mode ser ut. Bytt til dark mode.

Scenario 4

Du er nødt til å tvinge jobben “kunde har mistet mitt phonero” til å kjøre.

Scenario 5

Du har ikke noe behov for å se jobber som er disabled, paused eller er ok. Filtrer slik at du ikke får dem opp.

Scenario 6

En ny job må bli lagd og du må bruke databasen du allerede har laget. Den har også behov for å bli kjørt hvert 5 minutt og er en del av IT. Den blir sett på som kritisk.

Scenario 7

Du trenger å lese dokumentasjonen på en job. Finn frem til dokumentasjonen.

7.5 System Usability Scale

System Usability Scale, 1 IT	Helt enig 4	3	2	1	Helt uenig 0
Jeg opplevde systemet som unødvendig komplisert					x
Jeg synes systemet var enkelt å bruke	x				
Jeg tror jeg trenger hjelp for å bli bedre kjent med	x				

systemet					
Jeg synes at de forskjellige funksjonene i systemet er godt integrert	x				
Jeg synes det var dårlig sammenheng i systemet					x
Jeg tror de fleste vil lære seg å bruke systemet fort	x				
Jeg synes systemet var tungvint å bruke					x
Jeg følte meg selvsikker når jeg brukte systemet		x			
Jeg tror dette systemet vil funke bedre enn det gamle	x				

System Usability Scale, 1 Billing	Helt enig 4	3	2	1	Helt uenig 0
Jeg har lyst til å bruke dette systemet	x				
Jeg opplevde systemet som unødvendig komplisert					x
Jeg synes systemet var enkelt å bruke	x				
Jeg tror jeg trenger hjelp for å bli bedre kjent med systemet		x			
Jeg synes at de forskjellige funksjonene i systemet er godt integrert		x			
Jeg synes det var dårlig sammenheng i systemet					x
Jeg tror de fleste vil lære seg å bruke systemet fort	x				
Jeg synes systemet var tungvint å bruke					x
Jeg følte meg selvsikker når jeg brukte systemet	x				
Jeg liker bedre det eldre systemet					x

System Usability Scale, 2 IT	Helt enig 4	3	2	1	Helt uenig 0
Jeg har lyst til å bruke dette systemet	X				
Jeg opplevde systemet som unødvendig komplisert				x	

Jeg synes systemet var enkelt å bruke		x			
Jeg tror jeg trenger hjelp for å bli bedre kjent med systemet				x	
Jeg synes at de forskjellige funksjonene i systemet er godt integrert		x			
Jeg synes det var dårlig sammenheng i systemet				x	
Jeg tror de fleste vil lære seg å bruke systemet fort	x				
Jeg synes systemet var tungvint å bruke					x
Jeg følte meg selvsikker når jeg brukte systemet		x			
Jeg liker bedre det eldre systemet			x		

System Usability Scale, 2 Billing	Helt enig 4	3	2	1	Helt uenig 0
Jeg har lyst til å bruke dette systemet	x				
Jeg opplevde systemet som unødvendig komplisert					x
Jeg synes systemet var enkelt å bruke		x			
Jeg tror jeg trenger hjelp for å bli bedre kjent med systemet				x	
Jeg synes at de forskjellige funksjonene i systemet er godt integrert	x				
Jeg synes det var dårlig sammenheng i systemet				x	
Jeg tror de fleste vil lære seg å bruke systemet fort	x				
Jeg synes systemet var tungvint å bruke					x
Jeg følte meg selvsikker når jeg brukte systemet	x				
Jeg liker bedre det eldre systemet					x

7.6 Brukertesting tilbakemeldinger

Person 1, IT

- Bruker mye piltaster for navigasjon
- Ikke intuitivt ved bruk av filterknapper type pushbuttons
- Refresh siden for å få opp database
- Mulighet til å runne fix knapp under fix
- Pushbutton skrift vanskelig å lese i light mode
- Fix burde standard ha dropdown funksjon på

Person 1, Billing

- Sortering burde standard sortere etter alvorlighetsgrad
- Hover info når musepekeren er på en job

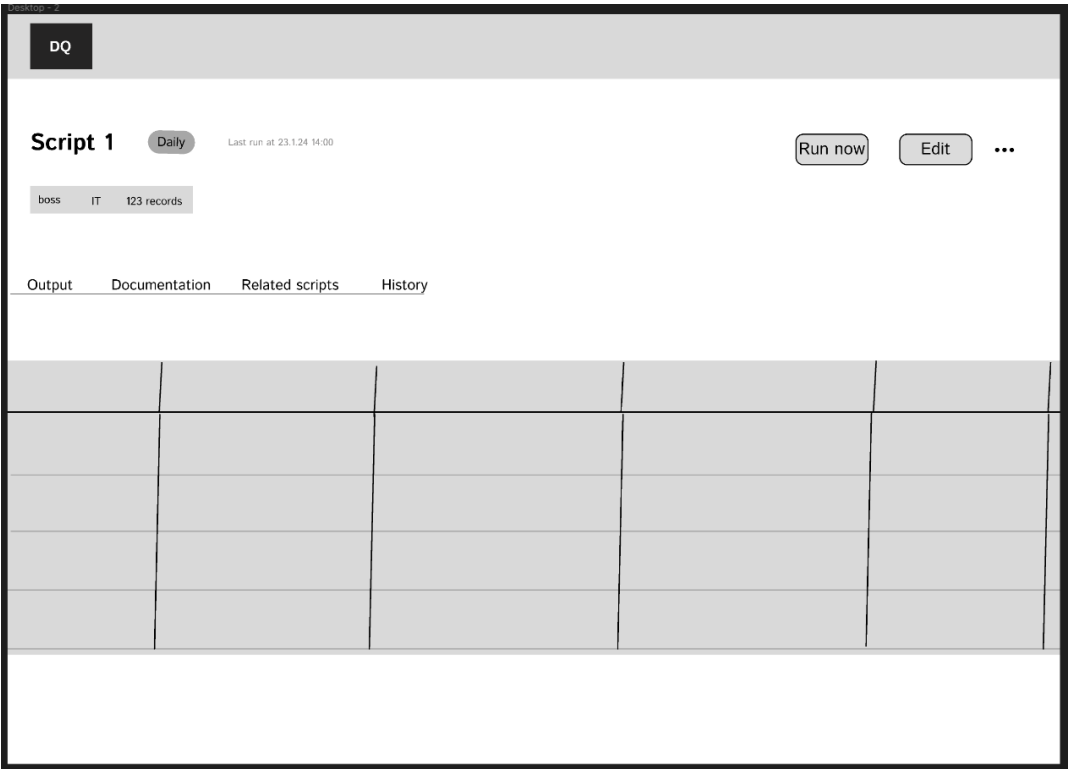
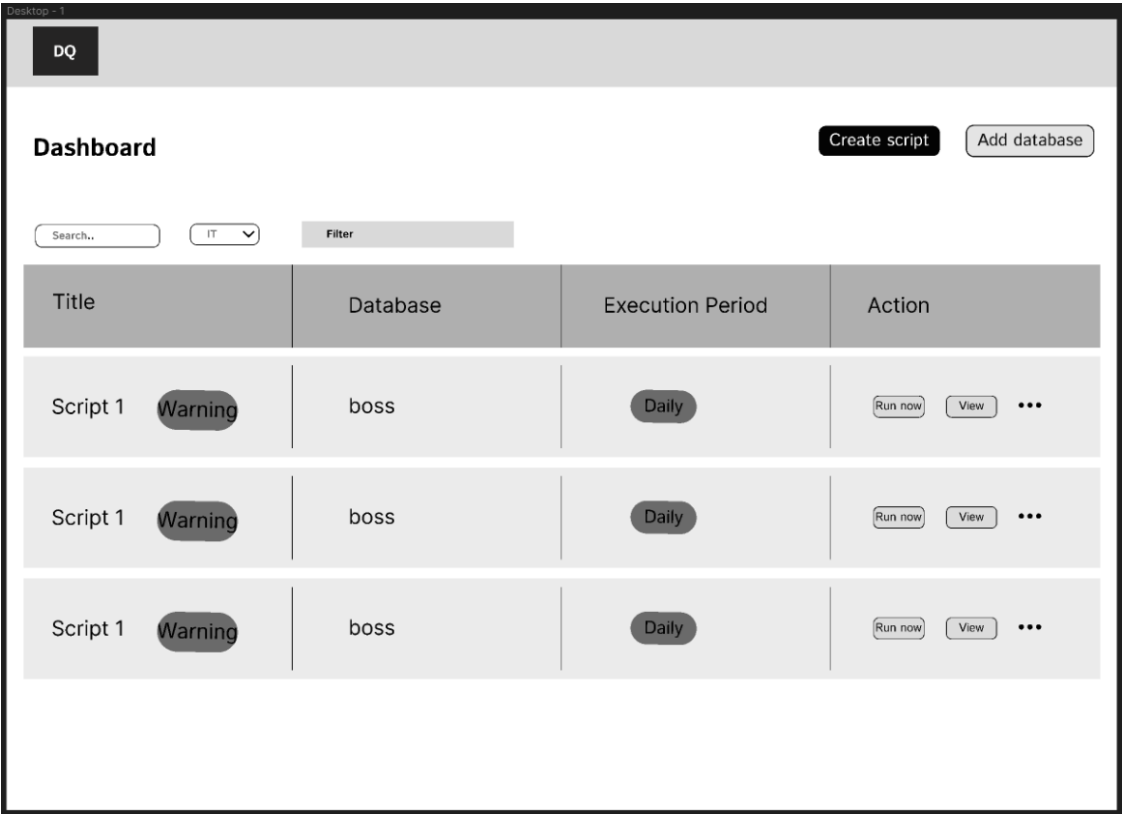
Person 2, IT

- Trykker på “change theme” isteden for avatar
- Pop-up hvor det står hva man mangler i create job
- Vil også se fixes med en gang

Person 2, Billing

- Filtrering med pushbuttons virker “omvendt”. Du må trykke på hva du vil beholde istedenfor fjerne
- Pushbuttons burde ha tydeligere highlighting
- Savner filtrering av tags

7.7 Low Fidelity Wireframes



Create script

DQ

Create script

General

Name

Description

Database

Add database

Department

Add department

Tags

+

Script settings

Script

7.8 High Fidelity Wireframes

test side for pushbutton view

defcon

Dashboard

High severity alarms

4 active

Warning

Alarms in queue

1 active

Running

Alarm status OK

2 active

Paused

Alarm status Error

0 active

Error

Alarm status Disabled

2 active

Disabled

Create job

Find job

+ Department

+ Database

+ Execution period

+ Status

+ Severity

+ Type

View

Name	Last seen by	Severity	Last run	Status	Type	Execution period	Department	Database	Action
Kunder mangler betaling	TK	High	17.01.2024	Disabled	Alarm	Daily	Billing	marketing	Run ...
ContractProductBilling FromDate sterre...	TK	Medium	09:55:06	Error	Alarm	Every 5 min	IT	marketing	Run ...
ContractProductBilling FromDate sterre...	TK	Medium	09:55:08	Warning	Alarm	Every 5 min	Billing	boss	Run ...
ContractProductBilling FromDate sterre...	TK	High	09:00:03	Running	Alarm	Hourly	Billing	IT	Run ...
ContractProductBilling FromDate sterre...	TK	Medium	06:00:02	OK	Counter	Daily	IT	marketing	Run ...
ContractProductBilling FromDate sterre...	TK	Low	06:00:04	Paused	Counter	Daily	Billing	marketing	Run ...

Records per page

Side 1 av 100

Run all

Job view - Overview regular

TK

defcon

AI ARM

Kunder mangler betaling

Daily

Run now

Acknowledge

Edit

Last ran at 17.01.2023 13:34:01

Markedsføring

IT

Warning

At 16.01.2021 11:34:01, the script returned 13 rows

Overview

Fixes

Documentation

Related scripts

Description

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum

Activity

OK

Alarm "Kunde mangler betaling"

Missed

08.01.2024 14:00:10

Count by 4

Alarm "Kunde mangler betaling"

System

01.01.2024 09:09:54

Count by 1

Alarm "Kunde mangler betaling"

Proble Ang...

01.01.2024 09:09:54

Failed

Alarm "Kunde mangler betaling"

System

02.01.2024 12:23:16

Count by 3

Warning

Alarm "Kunde mangler betaling"

System

02.01.2024 09:00:16

OK

Alarm "Kunde mangler betaling"

Missed

01.01.2024 09:09:54

CustomerName	Customer ID	Product Name	ActivationDate
Jim Halpert	123123123123	5 GB + roaming	20.10.2024
Dwight Schrute	123123123123	5 GB + roaming	13.3.2024
Stanley Hudson	123123123123	Fri bruk	20.10.2024
Angela Martin	123123123123	Fri bruk	9.10.2024
Kevin Malone	123123123123	10 GB + roaming	29.2.2022
Oscar Martinez	123123123123	Fri bruk	20.10.2024

TK

defcon

Dashboard

High severity alarms

4 active

Error

Settings

Light mode

Log out

Create script

Find Alarm

+ Department

+ Database

+ Execution period

+ Status

+ Severity

Alarm type

Name	Department	Database	Execution period	Status	Last run	Severity	Action
Kunder mangler betaling	Billing	marketing	Daily	Error	17.01.2024	High	Run Edit
ContractProductBilling FromDate sterre enn ToDate	Billing	marketing	Every 5 min	OK	17.01.2024	Medium	Run Edit
ContractProductBilling FromDate sterre enn ToDate	Billing	marketing	Every 5 min	OK	17.01.2024	Medium	Run Edit
ContractProductBilling FromDate sterre enn ToDate	Billing	marketing	Every 24 hour	Running	17.01.2024	High	Run Edit
ContractProductBilling FromDate sterre enn ToDate	Billing	marketing	Daily	Error	17.01.2024	Low	Run Edit
ContractProductBilling FromDate sterre enn ToDate	Billing	marketing	Daily	Error	17.01.2024	Low	Run Edit

Records per page 6Side 1 av 100

65 av 79

General

General settings of your new script

Name

The public name of your script

Description

Auto saved

A description of what your script does

Database

+ Add new database

The database this script should target

Department

+ Add new department

+ boss

+ marketing

✓ billing

s script is used by

Tags

🔍 Kundesøtte Faktura Billing Type to add tags...

Tags

Apply tags to your script to categorize and relate scripts together

Script

Write and configure your script to be executed

SQL script

```
SELECT
  c.customer_name,
  c.customer_email,
  o.order_date,
  o.order_amount
FROM
  customers c
JOIN
  orders o ON c.customer_id = o.customer_id
WHERE
  o.order_date >= '2023-01-01'
ORDER BY
  o.order_date DESC;
```

The SQL script to be executed

Dry run

Fixes

Name

Name

SQL script

```
SELECT
  c.customer_name,
  c.customer_email,
  o.order_date,
  o.order_amount
FROM
  customers c
JOIN
  orders o ON c.customer_id = o.customer_id
WHERE
  o.order_date >= '2023-01-01'
ORDER BY
  o.order_date DESC;
```

+ Add fix

Documentation

Provide documentation and link to related documents or other scripts

Articles

www.example.com



+ Add article

Related scripts

Type to add tags...

Scripts

- ✓ Billing har feil dato Ola Nordmann
- ✓ Manglende status Karl Nordmann
- ✓ Brukere med samme nummer Ola Nordmann

Link other related scripts to display them in the Script details view

Alarm settings

Settings for script and alarm behavior

Execution period

Every 24 hours

How often this script should be executed

Set hour

Default (06:00)

00:00

01:00

02:00

✓ Default (06:00)

Alarm action

☒ Raise alarm

☐ Counter

The action of this alarm

Set Severity of alarm

☐ Critical

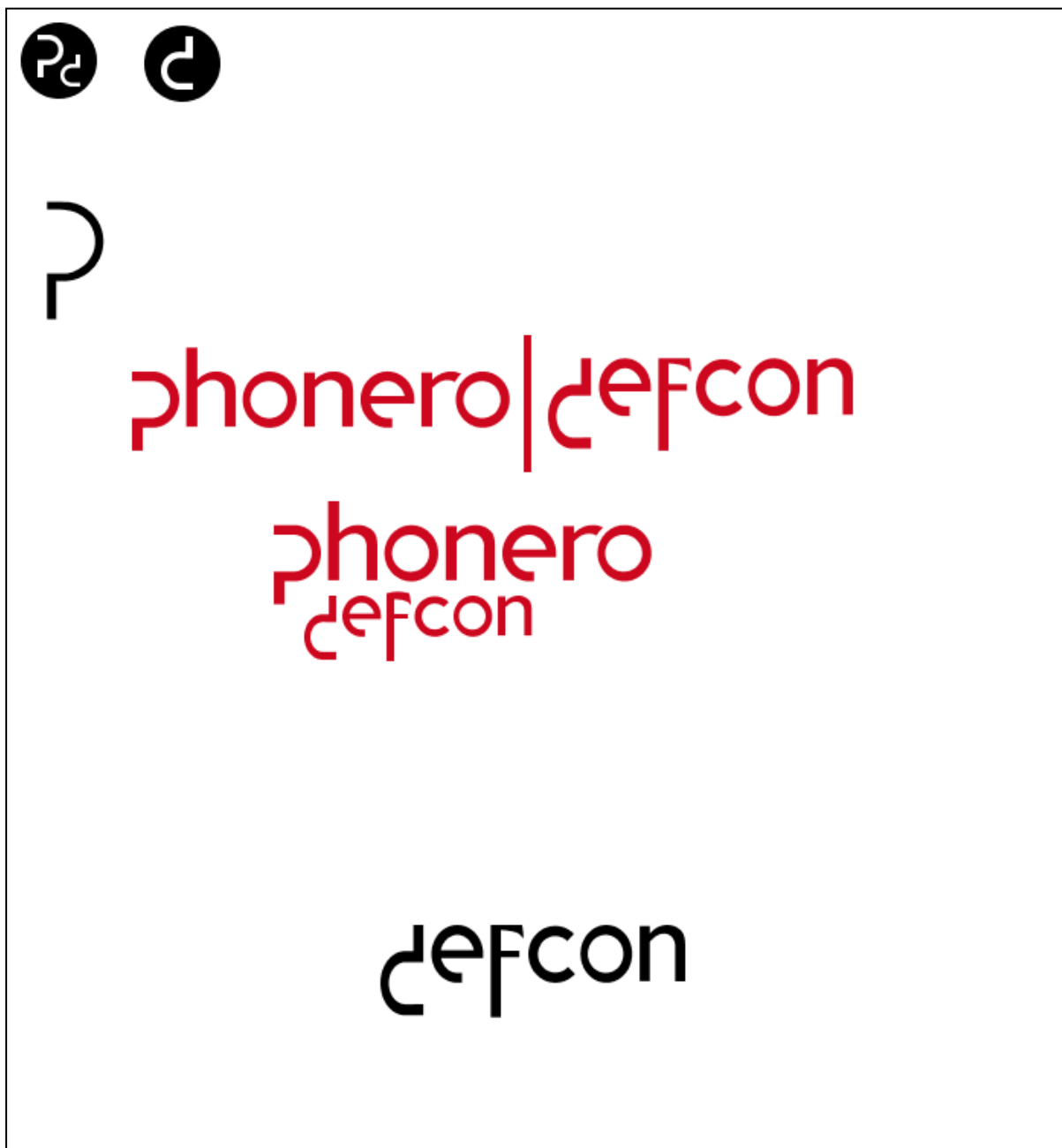
☒ High

☐ Medium

☐ Low

The severity of the alarm

7.9 Defcon logo



7.10 Personas

Michelle IT



ALDER	32
UTDANNELSE	Bachelor I IT og Informasjonssystemer
STATUS	Gift
STILLING	System utvikler
BOSTED	Kristiansand
TEKNISK KUNNSKAP	Høy

“ Jeg liker å bruke DQM når jeg sitter på kontoret. Er raskere å komme i kontakt med de som er relevante hvis en alarm slår ut.

Personlighet

Ekstrovert

Ansvarlig

Åpen og omgjengelig

Teknisk

Biografi

Michelle bor i kvadraturen og sitter lange dager på kontoret. Er gift og er kjent for å strukturere dagen godt. Endel av planleggingen går på kjennskap til arbeidsoppgaver nå og i fremtiden. Michelle er litt kjent med interne verktøy, men ennå ny i jobben etter bare 1.5år hos phonero.

Kjerne behov

- Avhengig av en stabil DQM for å jobbe effektivt.
- Lett tilgjengelig tjeneste som ikke er for avansert, da det allerede eksisterer andre verktøy for å håndtere utslag fra alarmene andre steder.
- Ønsker å legge meg til som watcher på en alarm så jeg får beskjed når det er en oppdatering på en utslått alarm
- Ønsker å se dokumentasjon på alarmer og ligende scripts sånn at det blir lettere å løse utslagene.

Frustrasjoner

- Utdatert nettside som er klønete å håndtere.
- Bekymret for sikkerhet med muligheter for mange mulige scripts i alarmsystemet.
- Irriterende å ikke kunne se tilbake på hvem som har gjort endringer i alarmer og gjort fixes.

Plattformer



Web site Mobil App Wallboard

William Billing



ALDER	39
UTDANNELSE	Master i Business
STATUS	Gift
STILLING	Billing analytiker
BOSTED	Kristiansand
TEKNISK KUNNSKAP	Medium - Høy

“ Jeg er liker godt å bytte mellom hjemmekontor og i byen, og er derfor avhengig av en brukervennlig DQM.

Personlighet

Fleksibel Nysgjerrig
Analytisk Karismatisk

Biografi

Hei, bor et lite stykke unna kristiansand sentrum. Er gift, liker miljøet på Phonero og de ekstra trenings aktiviteten som blir arrangert. Jobbet her i 8 år og er forholdsvis godt kjent med interne tjenester på huset.

Kjerne behov

- Ønsker å ta ibruk DQM mer for å løse kjerne utfordringer.
- Ønsker å få en oversiktlig og mer informativ liste med de aller viktigste alarmene.
- At alarmer som har gått er tydelig i alvorlighetsgrad, og det er lett å fordele ansvarsområde er viktig for meg.

Frustrasjoner

- Alarmer kan ligge lenge på grunn av lite relevans.
- Lite kjennskap til:
 - Når alarmen gikk første gang
 - Hvem som har sett den, og hvem skal gjøre noe med det
- Alarmfeeden er relativ uoversiktlig så noe informasjon kan bli borte og koste penger for selskapet.

Plattformer



Webside Wallboard

7.11 Risikoanalyse

RisikoTabell:

Risiko	Sannsynlighet	Konsekvens	RISIKO	Tiltak for at det ikke skal skje	Tiltak for å redusere skade
Usikkerhet rundt krav, funksjonalitet	2	4	8	Hyppig kontakt med oppdragsgiver. Tidlig etablere prioriteringer gjennom analyser.	Planlegge møter med oppdragsgiver. Gå gjennom analysene igjen for å sikre at man utvikler i tråd med kravene etablert i analysene.
Manglende kompetanse i prosjektet	4	3	12	Utføring av kurs prosjektet, aktivt be om hjelp av gruppemedlemmer /veileder/oppdragsgiver	Parprogrammering og tett oppfølging av personene det gjelder. Eventuelt gi dem andre oppgaver som de er bedre på
Ikke svar, oppfølging fra oppdragsgiver	1	4	4	Fysisk oppmøte/ planlegging av møter med oppdragsgiver	Opprette dialog mellom de 3 partene: UIA, Gruppen, og oppdragsgiver
Uklart i overgangen fra analyse/design til implementering	3	4	12	God kommunikasjon Kontinuerlig bruke design mockups i utviklingsfasen.	Ta et skritt tilbake og ha en grundig gjennomgang av analyse og design mockups.
Dårlig gruppedynamikk	2	4	8	Konstruktiv kritikk, hyppige tilbakemeldinger, Teambuilding.	Dialog innad i gruppen, teambuilding aktiviteter
Prosjektet/arbeidet stopper opp pga. mangel på motivasjon	2	4	8	Innføre milepæler, forhindre utbrenning ved å: Innføre pauser, koble av når man forlater arbeidsplassen.	Snakk med arbeidsgiver/ Veileder. Redusere målsetning og justere forventninger
Brudd i sikkerhet (uønskede får tilgang til)	2	5	10	Låse pc hver gang man forlater pulten, bruke vpn dersom man	Gi beskjed med en gang til oppdragsgiver og innføre strengere sikkerhetsrutiner..

kildekode)				jobber hjemme, lage sterke passord.	
Uenigheter i gruppen eller med oppdragsgiver	2	4	8	God dialog og hyppige tilbakemeldinger til hverandre og oppdragsgiver	Avtale oppklaringsmøte hvor forventninger gjennomgås og justeres
Sykdom på arbeidsplassen	5	3	15	Holde seg hjemme ved smittebærende sykdom.	Legge opp til hjemmekontor ved lettere sykdom
Ubalansert arbeidsfordeling og forståelse av prosjektet	3	3	9	Daglige Scrum møter og kontinuerlig ha dialog om hva man syns er for mye eller lite arbeid	Komme til enighet om hva som må gjøres for at noen enten skal få flere eller færre arbeidsoppgaver
Dårlig kodestandard : duplication, high cohesion, etc.	4	4	16	Aktivt be om hjelp. Reviewer i jira vil godkjenne om det følger standarden	Gjennomgang av endringene i branchen med andre gruppemedlemmer
Forskjellig kodestandard	2	2	4	Felles gjennomgang av kodestandard. Gode oppgavebeskrivelser i jira	Gå gjennom det som har blitt pushet opp, sørge for at alt er funksjonelt og ikke går utover UX. Evt
Sitter fast på enkeltoppgaver	5	2	10	Sørge for at det er en lav terskel for å be om hjelp	Be om hjelp fra Scrum master, evt. delegere oppgaver videre
For lite arbeid å gjøre mtp. gruppens størrelse	3	2	8	Daily Scrum→ sikre at alle gjør noe hver dag	Samarbeide og jobbe mer i felleskap. Jobbe mer med “should have” og “could have”
Vi møter ikke MVP	2	4	8	Daily Scrum og innføring av milepæler. Øke arbeidsmengde.	Informere veileder og produkteier om at vi ikke når målet. Forklare og drøfte om det i rapporten

Risikomatrise V1

SANNSYNLIGHET	Svært Sannsynlig	5					
	Meget Sannsynlig	4					
	Sannsynlig	3					
	Lite Sannsynlig	2					
	Usannsynlig	1					
			1	2	3	4	5
			Ubetydelig	Mindre Alvorlig	Betydelig	Alvorlig	Svært Alvorlig
KONSEKVENNS							

Risikomatrise V2:

SANNSYNLIGHET	Svært Sannsynlig	5	5	10	15	20	25
	Meget Sannsynlig	4	4	8	12	16	20
	Sannsynlig	3	3	6	9	12	15
	Lite Sannsynlig	2	2	4	6	8	10
	Usannsynlig	1	1	2	3	4	5
			1	2	3	4	5
			Ubetydelig	Mindre Alvorlig	Betydelig	Alvorlig	Svært Alvorlig
KONSEKVENNS							

7.12 PACT

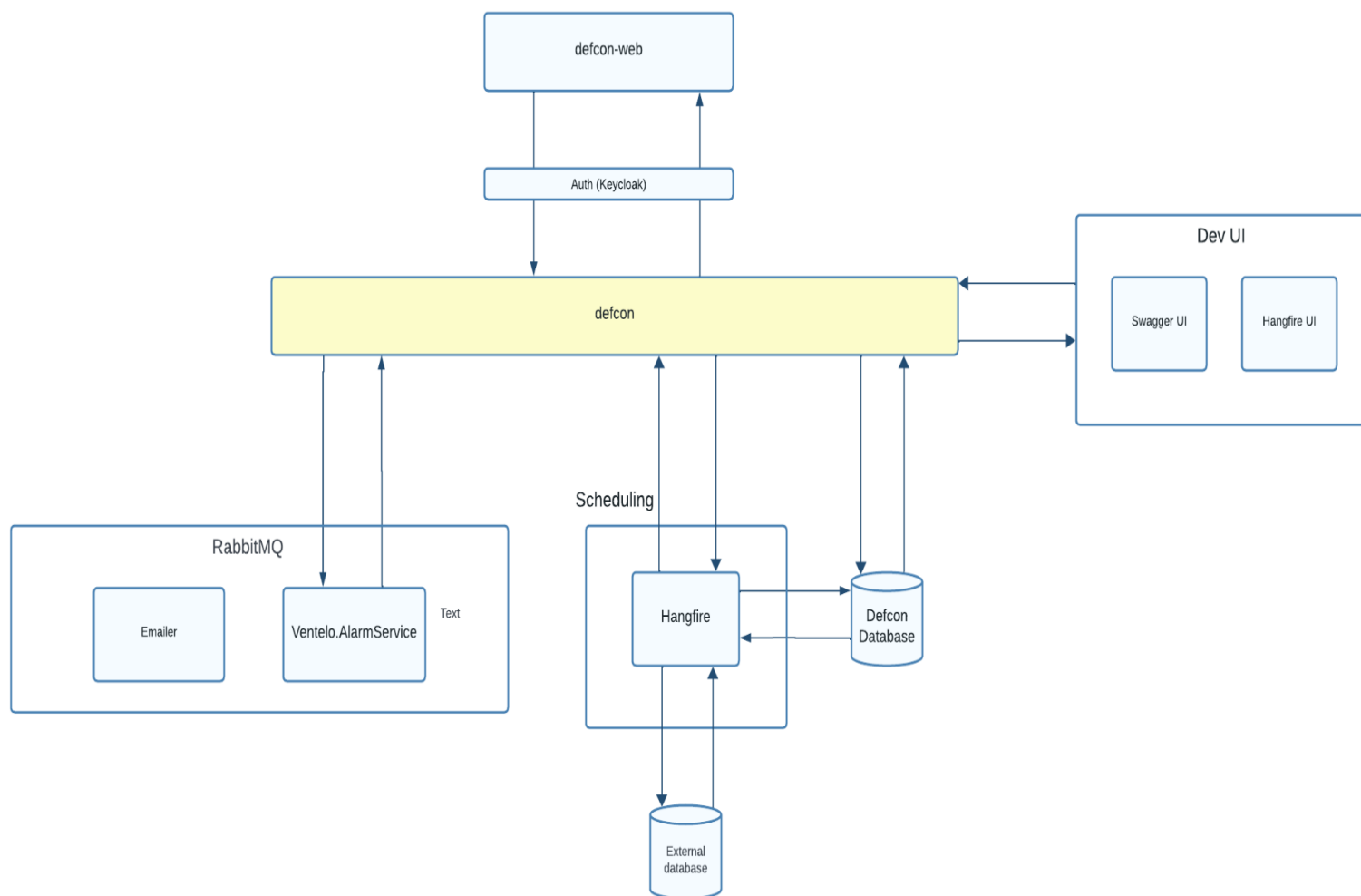
People	- God teknisk kompetanse - Varierende alder på ansatte - Norsktalende ansatte - Ulike arbeidsteam fra flere avdelinger
Activities	- Registrere, kommentere, oppdatere og redigere saker - Opprette og dele brukere inn i team - Innlogging
Context	- Applikasjonen skal bli brukt på arbeidsplassen - Applikasjonen skal kunne brukes hjemme(?)
Technology	- Smarttelefon - PC - Wallboard

7.13 Prototype i Figma

Url:

<https://www.figma.com/proto/nb0X7BFYNkp801QLHQA24x/dataquality-web-mockups?type=design&node-id=458-5153&t=qp7THqz96lATfKz0-1&scaling=scale-down&page-id=0%3A1&starting-point-node-id=433%3A3001&mode=design>

7.14 Systemarkitektur



7.15 Alternativt bilde til Designprosessen

