

Mats Elias Davidsen
Kristian Nybakken Lund
Knut Wiig

CutWear - digitalisering av tunnelbransjen

Bacheloroppgave i Dataingeniør
Veileder: Helge Hafting
Mai 2019

Mats Elias Davidsen
Kristian Nybakken Lund
Knut Wiig

CutWear - digitalisering av tunnelbransjen

Bacheloroppgave i Dataingeniør
Veileder: Helge Hafting
Mai 2019

Norges teknisk-naturvitenskapelige universitet
Fakultet for informasjonsteknologi og elektroteknikk
Institutt for datateknologi og informatikk

Forord

Denne bacheloroppgaven er gjennomført som et ledd i den treårige bachelorgraden for dataingeniør. Oppgaven ble fremstilt som et resultat av et samarbeid mellom The Robbins Company, Seattle, USA, og Institutt for bygg- og miljøteknikk NTNU. Oppgaven ble deretter publisert for studenter ved NTNU.

Før prosjektstart hadde gruppen møter med flere potensielle oppdragsgivere. Valg av oppgave baserte seg på to ting, oppgaven i seg selv og oppdragsgiverne. Oppgaven var meget interessant da den var relevant for arbeidslivet og hadde mange spennende elementer. Førsteintrykket av oppdragsgiverne Sindre Log, Amund Bruland og Helge- Ivar Frostad var godt, og partene fikk god kjemi. Basert på dette ble valget av bacheloroppgave enkelt. Ved prosjektslutt satt gruppen fremdeles med en følelse av at dette var det rette valget.

Partene ble enig om at oppdragsgiver skulle få rettighetene til produktet og muligheten for å utnytte produktet kommersielt. På bakgrunn av dette er ikke kildekoden vedlagt dette dokumentet, se vedlegg D2 og D3. Dokumenter som vil være aktuelle for The Robbins Company å bruke for videreutvikling eller bruk av produktet er skrevet på engelsk. For tilgang til programmet må man kontakte oppdragsgiver Amund Bruland.

Underveis i prosjektet fikk gruppen forståelse for hvorfor mange digitaliseringsprosjekter i større skala ofte mislykkes. Som erfart var det vanskelig å skaffe seg en helhetlig oversikt i startfasen, noe som gjør det vanskelig å estimere arbeidsoppgavene. Dette kan føre til feilbudsjettering og feilestimering av tid.

Under hele prosessen hadde utviklingsteamet god kontakt med veileder Helge Hafting. Dette inkluderer flere oppstartsmøter før selve utviklingen begynte, samt sprint-møter underveis i prosjektet. Her fikk utviklingsteamet gode tips for arbeidet. Derfor vil gruppen gjerne rette en stor takk til Helge Hafting.

Gruppen vil også rette en stor takk til oppdragsgivere Sindre Log, The Robbins Company, professor Amund Bruland, institutt for Bygg- og miljøteknikk NTNU og Helge- Ivar Frostad. Selv med travle hverdager var oppdragsgiverne veldig behjelpelige. Før utviklingen startet hadde partene flere møter hvor oppgaven og domenet ble diskutert. Deretter hadde partene regelmessig kontakt i form av blant annet sprint-møter.

Mats Elias Davidsen

Mats Davidsen

Signature

20.05.2019

Date

Trondheim, Norge.

Kristian Nybakken Lund

Kristian N Lund

Signature

20.05.2019

Date

Knut Wiig

Knut Wiig

Signature

20.05.2019

Date

Oppgavetekst

Oppdragsgiver

Institutt for bygg- og miljøteknikk, NTNU og The Robbins Company, Seattle/Solon, USA.

Hensikten med oppgaven

Utvikling av programvare for beslutningsstøtte i forbindelse med utskifting av borverktøy på tunnelboremaskiner (TBMer) som borer tunneler i sterk og slitende bergmasse. Fagområdet Anleggsteknikk driver forskning på TBM, dvs. fullprofilboring eller mekanisk brytning av berg uten bruk av sprengstoff. Brytning skjer ved hjelp av borverktøy (kuttere) som er utformet som en ring av spesialstål, som er det mekaniske verktøyet som bryter ut bergmassen basert på aksial kraft og rotasjon.

Se aktuell link f.eks.: <https://www.youtube.com/watch?v=J0YqpZCoYEQ>. I typisk norske bergmasser (som forøvrig finnes over hele verden), er slitasje og dermed bytte av borverktøy en kritisk faktor for suksess.

Basert på eksisterende NTNU prognosemodell for estimering av tids- og kostnadsforbruk, ble det av to studenter fra Institutt for datateknologi og informatikk i 2017 utviklet en programmodul (dataverktøy) for bruk av denne prognosemodellen. Det ønskelig med ny programutvikling innen fullprofilboring av tunneler. I samarbeid med The Robbins Company i Seattle, USA, har vi identifisert et behov for digitale verktøy som støtte i produksjonsprosessen, nærmere bestemt for overvåkning av slitasjetilstand og optimal strategi for bytte av borverktøy (kuttere). Overordnede hensikter med oppgaven er å bidra til digitalisering av tunnelbransjen, samt å utvikle dataingeniører med kunnskap om tunnelbransjen.

Hensikten med oppgaven er hentet direkte fra oppgaveteksten som ble utlyst, se vedlegg G.

Kort beskrivelse av oppgaveforslag

1. Programmet skal brukes til å registrere tilstand på alle borverktøy på borhodet ved periodisk inspeksjon (1 eller flere ganger i døgnet). Flere innparametere.
2. Programmet skal brukes til å registrere bytter av borverktøy. Flere innparametere.
3. Innsamlede data skal overføres til en egnet database (nettbasert?).
4. Registrerte data skal analyseres for å gi real-time tilbakemeldinger og beslutningsstøtte til operatør som foretar inspeksjon og/eller bytte av borverktøy.
5. Registrerte data skal også brukes til mer omfattende analyser av f.eks. borverktøylevetid og TBM-design.
6. Programmet skal tilpasses bruk i krevende tunnelmiljø ved inspeksjon og bytte av borverktøy. Programmet må kunne brukes på industristandard nettbrett, f.eks. IP67-kvalitet.
7. Det er ønskelig at brukergrensesnittet fullt ut er basert på touch-screen.
8. Programmet utvikles i samarbeid med Institutt for bygg- og miljøteknikk ved NTNU og The Robbins Company ved deres representant i Norge/Trondheim Sindre Log.
9. Det må påregnes besøk til tunnelanlegg i Norge eller Europa som introduksjon til teknologien for fullprofilboring av tunneler i sterk og slitende bergmasse.
10. Studentene vil få grundig og regelmessig veiledning og oppfølging ved gjennomføring av oppgaven.

Den korte beskrivelsen av oppgaveforslaget er hentet direkte fra oppgaveteksten som ble utlyst, se vedlegg G.

Systemets krav

Systemets krav var under stadig endring gjennom hele prosjektet. Ordinært var det ikke noe spesifikke systemkrav, kun et forslag. Deretter gjennom en iterativ prosess ble oppdragsgiver og utviklingsteam enige om kravspesifikasjonen for den digitale løsningen. Kravspesifikasjonen ble utarbeidet som user stories i prioritert rekkefølge. Tilsammen ble det utarbeidet tre versjoner. Se vedlegg C.

Sammendrag

Denne bacheloroppgaven svarer på oppgaven fra Institutt for bygg- og miljøteknikk, NTNU og The Robbins Company, Seattle/Solon, USA. Den overordnede hensikten med oppgaven var å bidra til digitalisering av tunnelbransjen. På bakgrunn av oppgaveteksten utviklet gruppen hovedproblemstillingen:

Kan innføring av et digitalt system øke beslutningsstøtten for en eksisterende arbeidsprosess i forbindelse med utskifting av boreverktøy?

Denne problemstillingen ble valgt fordi et digitalt system som øker beslutningsstøtten for en eksisterende arbeidsprosess vil bidra til digitalisering av tunnelbransjen. I teorien ser man nærmere på digitalisering, hvorfor digitaliseringsprosjekter ofte feiler og hvilke forhåndsregler man kan ta for å unngå dette.

For å løse denne problemstillingen valgte gruppen å utvikle produktet ved hjelp av den agile utviklingsprosessen Scrum. Dette innebar regelmessige møter med oppdragsgiver og veileder der gruppen fikk gode tilbakemeldinger, dette gjorde at produktet ble utviklet i riktig retning. I utvikling av programmet valgte utviklingsteamet å bruke JavaScript. Dette er fordi det er godt dokumentert, relevant for arbeidslivet, lett å ta i bruk og enkelt å videreutvikle. Applikasjonen ble utviklet med fokus på å være en brukervennlig applikasjon som effektiviserer arbeidsprosessen i forbindelse med utskifting av boreverktøy. Noe av programmets funksjonalitet avhenger av om applikasjonen har internett-tilgang, hvis den ikke har dette vil brukeren få begrenset beslutningsstøtte.

Det ferdige produktet tilbyr økt beslutningsstøtte til prosessen rundt utskifting av boreverktøy. Til tross for manglende realistisk testmiljø, er det i stor grad forsøkt å legge til rette for at applikasjonen skal kunne brukes under de vanskelige forholdene i en TBM. Med disse egenskapene tatt i betraktning vil applikasjonen være mer effektiv og tilby mer beslutningsstøtte enn den nåværende løsningen. Gruppen kom frem til at disse resultatene i stor grad løser hovedproblemstillingen, samtidig som applikasjonen bidrar til digitalisering av tunnelbransjen.

Nøkkelord: Digitalisering, programvareutvikling, touch applikasjon, tunnelboring og Scrum.

Summary

This bachelor's thesis aims to resolve the assignment given by Institutt for bygg- og miljøteknikk, NTNU and The Robbins Company, Seattle/Solon, USA. The purpose of this assignment is to contribute to the digitalization of the tunneling business. Based on this the group developed the following main problem statement:

Can a digital system increase the decision support for the already existing process of changing cutters in a tunnel boring machine?

This problem statement was chosen because solving this would contribute to the digitalization of the tunneling business. The theory chapter in the thesis takes a closer look at digitalization, why digitalization projects often fail and which precautions one can take to prevent these projects from failing.

To solve the main problem statement the group chose to use the agile framework Scrum. This included regular meetings with the group's advisor and the customer where the group got feedback on their progress, as a result of these meetings the product was developed in the right direction. The team developed the software using JavaScript. This is because the language is well documented, relevant for future work, easy to use and simple to further develop. The software was developed to be a user friendly application that streamlines the process of changing cutters in a tunnel boring machine. Some of the program's functionality depends on having a internet connection, without a connection the user will get limited decision support.

The finished product offers increased decision support to the process of changing cutters in a tunnel boring machine. The application is developed to be used in the tough environment inside a TBM, despite the lack of a realistic test environment. When considering this the team concludes that the application will be more efficient and offer more decision support than the previous solution. The team concludes that the results of this bachelor's thesis solves the main problem statement, the application also contributes to the digitalization of the tunneling business.

Keywords: Digitalization, software development, touch application, tunneling and Scrum

Innhold

Figur og tabelliste	1
1 Introduksjon og relevans	2
1.1 Resultatmål	2
1.2 Forretningsmessig mål	2
1.3 Prosessmål	2
1.4 Problemstilling	2
1.5 Målgruppe	3
1.6 Rapportens struktur	3
1.7 Akronymer og forkortelser	4
1.8 Begrepsavklaringer	5
2 Teori	6
2.1 Digitalisering	6
2.1.1 Definisjon av digitalisering	6
2.1.2 Suksessfaktorer	6
2.1.3 Hvorfor feiler digitaliseringsprosjekter?	8
2.1.4 Gevinster ved digitalisering	9
2.1.5 Negativ side ved digitalisering	10
2.1.6 Digitaliseringsprosjektet CutWear	10
2.2 Brukervennlighet	11
2.2.1 MMI	12
2.2.2 Brukbarhet	12
2.2.3 Hvordan oppnå høy brukbarhet?	13
2.2.4 Brukeropplevelse	13
2.2.5 Prototyper	13
2.2.6 Brukervennlighet i CutWear	14
2.3 Metodikk	14
2.3.1 Agil utviklingsprosess	14
2.3.2 Native kontra Ikke-native applikasjoner	15
2.3.3 Rammeverk	15
2.3.4 Container	15
2.3.5 Kontinuerlig Integrasjon	16
3 Valg av teknologi og metode	17
3.1 Windows	17
3.2 Utviklingsmiljø	17
3.3 ReactJS	17
3.4 Electron	17
3.5 Node.js med Express	17
3.6 MySQL	18
3.7 Docker Compose	18
3.8 Electron-builder	18
3.9 Github	18
3.10 Travis CI og Codecov	18
3.11 Winston	18
3.12 jest	18
3.13 jsDoc	19
3.14 Onsen UI	19
3.15 Utviklingsprosess	19

4	Resultater	20
4.1	Vitenskapelige resultater	20
4.2	Ingeniørfaglige resultater	23
4.2.1	Forretningsmessige mål	23
4.2.2	Resultatmål	24
4.2.3	Prosessmål	25
4.2.4	Tester	26
4.3	Administrative resultater	26
4.3.1	Timeregnskap	26
4.3.2	Utviklingsprosess	26
5	Diskusjon	31
5.1	Drøfting av vitenskapelige resultater	31
5.2	Drøfting av ingeniørfaglige resultater	33
5.3	Drøfting av administrative resultater	34
5.3.1	Timeregnskap	34
5.3.2	Utviklingsprosess	35
5.4	Etiske problemstillinger	37
5.4.1	Profesjonsetiske spørsmål	37
5.4.2	Økonomiske konsekvenser	37
5.4.3	Miljømessige konsekvenser	38
6	Konklusjon og videre arbeid	39
6.1	Konklusjon	39
6.2	Videre arbeid	40
	Referanser	41
7	Vedlegg	43
A	Møter	43
A1	Oppstartsmøter	43
A2	Sprint 1	49
A3	Sprint 2	54
A4	Sprint 3	59
A5	Sprint 4	63
A6	Sprint 5	67
A7	Sprint 6	73
A8	Sprint 7	75
A9	Sprint 8	79
B	Brukbarhetstest	81
C	Kravdokumentasjon	83
C1	Introduksjon	84
C2	User Stories	84
	C2.1 Version 1	84
	C2.2 Version 2	91
	C2.3 Version 3	99
C3	Domenemodell	109
C4	ER-diagram	110
	C4.1 Versjon 1	110
	C4.2 Versjon 2	110
C5	Prototyper	112
	C5.1 Prototype 1	112
	C5.2 Prototype 2	139
	C5.3 Prototype 3	149
D	Korrespondanse og kontrakter	161

E	Visjonsdokument	165
F	Systemdokumentasjon	184
G	Ordinær oppgavebeskrivelsen	221
H	Risikoanalyse	224
I	Retrospective	229
J	Timelister	233
K	Further work	236
L	Trello board	236
M	User manual	238

Figur og tabelliste

Figurer

1	Suksessfaktor for prosjekter.	7
2	Prosjekttriangelets dimensjoner.	7
3	Gevinster fra digitaliseringsprosjekter.	9
4	Kotlers femtrinns modell for kjøpsbeslutningsprosessen.	10
5	Eksisterende løsning hos The Robbins Company.	11
6	Ny løsning ved bruk av CutWear.	11
7	Oversikt over Scrum.	15
8	TBM Inspector hovedside.	20
9	TBM Inspector, hovedside for posisjon.	21
10	TBM Inspector, registrer slitasje.	21
11	TBM Inspector, registrer bytte.	22
12	TBM Supervisor, inspeksjonsdata.	22

Tabeller

1	Forhåndsregler tatt for å oppnå et suksessfullt digitaliseringsprosjekt	24
2	Oversikt over user stories	24
3	Versjon 1 vs. Versjon 3, kravspesifikasjon	25
4	Økonomiske konsekvenser ved CutWear	38

1 Introduksjon og relevans

I dette kapittelet blir målene for prosjektet, problemstilling og målgruppen presentert. Deretter følger en kort beskrivelse av rapportens struktur etterfulgt av en liste med akronymer, forkortelser og begrepsavklaringer.

1.1 Resultatmål

Programmet skal være en fungerende applikasjon som kan installeres og kjøres på Windows nettbrett, samt kan ha mulighet kjøres i en nettleser. Applikasjonen skal:

- Fungere som et beslutningsverktøy for en TBM Inspector.
- Digitalisere prosessen knyttet til registrering av slitasje og bytting av kuttere.
- Gi The Robbins Company og Institutt for bygg- og miljøteknikk, NTNU kvalitetsdata til videre prognoser og analyser.

1.2 Forretningsmessig mål

The Robbins Company har som forretningsmessig mål å digitalisere tunnelbransjen. Som en del av dette har de identifisert et behov for digitale verktøy for støtte i produksjonsprosessen, nærmere bestemt for overvåkning av slitasjetilstand og optimal strategi for bytte av boreverktøy, se vedlegg G.

1.3 Prosessmål

Basert på de forretningsmessige målene til The Robbins Company, vil det komme flere lignende utviklingsprosjekter i fremtiden. Ved å gjennomføre denne oppgaven vil The Robbins Company få høyere kompetanse på hvordan de skal utføre slike utviklings - og digitaliseringsprosjekter.

Institutt for bygg- og miljøteknikk, NTNU og The Robbins Company har inngått et gjensidig samarbeid. De ønsker å forsterke denne relasjonen. Gjennom denne oppgaven vil partene styrke dette samarbeidet ved utveksling av data. Her vil NTNU få tilgang til data som kommer fra den digitale løsningen som skal utvikles.

Prosessmålet for utviklingsteamet vil primært være kompetansebygging på flere områder. Teamet skal sitte sammen gjennom hele perioden. Her vil teamet få nyttig erfaring med prosjektarbeid, hvor hvert medlem vil få en bedre forståelse av hvilke rolle de naturlig passer inn i. Her vil man også lære hva som er effektive og mindre effektive arbeidsmetoder. Utviklingsteamet vil også opparbeide seg verdifull kunnskap om Scrum, noe som er en utbredt arbeidsmetodikk.

Teamet vil også tilegne seg kunnskaper på hvordan man skal skrive litterære dokumenter, dette innebærer blant annet:

- Hvordan komme frem til en god problemstilling.
- Undersøke og vurdere vitenskapelig litteratur.
- Komme frem til gode løsninger av problemstillingen.

Opgaven kommer fra en reell oppdragsgiver som ønsker å benytte seg av resultatet på et kommersielt nivå. Dette betyr at teamet vil få masse relevant kunnskap om hvordan arbeidslivet kan være. Dette inkluderer alt fra planlegging av en større applikasjon, utviklingen, dialog med kunden og å opprettholde tidsfrister.

1.4 Problemstilling

Problemformulering

Gruppen har valgt å undersøke om en eksisterende arbeidsprosess i industribransjen kan effektiviseres ved hjelp av en samlet digitalisert løsning. Den eksisterende prosessen går ut på å registrere data med penn og

papir, deretter skrives dette inn på en datamaskin og sendes til ledelsen som skriver dette inn på et felles excel-ark. Den største delen av denne arbeidsprosessen, selve registreringen, gjøres under vanskelige forhold inne i selve borhodet.

De vanskelige forholdene vektlegges når den digitale løsningen skal utvikles, og dette er en av de større utfordringene tilknyttet utviklingen.

Hovedproblemstilling

Hovedproblemstilling er som følger:

Kan innføring av et digitalt system øke beslutningsstøtten for en eksisterende arbeidsprosess i forbindelse med utskifting av boreverktøy?

Under hovedproblemstilling brukes uttrykket “øke beslutningsstøtten”. Økt beslutningsstøtte måles ut fra disse punktene:

1. Tiden det tar for informasjon å komme seg fra første ledd til siste ledd i arbeidsprosessen. Dette vil være fra en inspektør måler kutterslitasje, til inspeksjonsdata er registrert i en felles database.
2. Hvordan fremvisning av lagret data vil hjelpe inspektørene med å ta de rette avgjørelsene under inspeksjonene.
3. Hvordan applikasjonen hjelper inspektørene med å optimalisere arbeidsprosessen under inspeksjon av borhodet.
4. Hvordan lagring av data vil hjelpe ledelsen med videre analyser og prognoser.

Sekundær problemstilling

Siden store deler av den eksisterende arbeidsprosessen foregår under vanskelige forhold, må man også adressere følgende problemstilling:

Vil en digitalisert løsning kunne brukes under de vanskelige forholdene for selve registreringen av kutterslitasje?

1.5 Målgruppe

Applikasjonen

Applikasjonen CutWear ble laget for The Robbins Company, slik at de videre kunne tilby dette programmet til sine kunder. Brukerne av dette systemet vil variere i stor grad. Her har man alt fra administratorer internt hos The Robbins Company, ansatte ved NTNU som vil analysere data og ulike bruker grupper ute hos kunden. Dette skal altså ikke lanseres mot forbrukermarkedet.

Rapporten

Rapporten er primært skrevet for de ansatte ved Institutt for datateknologi og informatikk, samt eventuelle utviklere som skal videreutvikle produktet. På tross av dette vil rapporten bli lagt ut offentlig, slik at den vil også kunne være behjelpelig for andre som er interessert i dette emne.

1.6 Rapportens struktur

Under kommer det en oversikt over de resterende kapitlene, samt forventet utbytte av å lese dem.

Kapittel 2 - Teori

Dette kapitlet skal dannet grunnlaget for leserens teoretiske kunnskap om tema i rapporten. Her vil leseren få en dypere forståelse av digitalisering og hvorfor god brukervennlighet er essensielt for et IKT- prosjekt.

Kapittel 3 - Valg av teknologi og metode

Her skal leseren få et dypere innblikk i hvilke teknologier og metoder som har blitt brukt og hvorfor akkurat disse ble brukt.

Kapittel 4 - Resultat

Her får leseren et innblikk i resultatet av prosjektet. kapitlet er delt inn i tre deler, (1) det vitenskapelige resultatet, (2) det ingeniørfaglige resultatet og (3) det administrative resultatet.

Kapittel 5 - Diskusjon

For å få en oversiktlig rapport er dette kapitlet delt inn i samme tre deler som kapittel 4 - resultat. Her kan leseren få et innblikk i tankene rundt de respektive resultatene.

Kapittel 6 - Konklusjon og videre arbeid

Her finner leseren konklusjonen av problemstillingen, kravene i visjonsdokumentet og diskusjonen i forrige kapittel. Dette kapitlet vil være ekstra interessant for potensielle lesere som skal videreutvikle applikasjonen, fordi her finner man hvilke videreutviklinger av produktet som er påtenkt og hva som er lagt til rette i systemet for videreutvikling.

1.7 Akronymer og forkortelser

- **CutWear** Navnet på programmet produsert i denne bacheloroppgaven.
- **TBM** Tunnelboremaskin.
- **Kutter** En mekanisk del som brukes i TBM'er til å slite ned berg.
- **Borhode** Fremsiden på en TBM, utstyrt med flere kuttere.
- **Housing** Plassen som kuttere blir plassert i på borhodet. Disse byttes svært sjeldent og sitter fastmontert på borhodet.
- **TBM Inspector** Dette er mannskapet som gjør avlesning-, registrering- og bytter av kuttere.
- **TBM Supervisor** Dette er brukere med et overordnede ansvar i et prosjekt.
- **Super User** Dette er brukere med interesse av å hente ut data for analyse, de har ikke mulighet til å endre på eller legge til data.
- **Admin** En administrator for systemet, sørger for at nye prosjekter og brukere blir opprettet.
- **MMI** Menneske-maskin-interaksjon.
- **ERP** Enterprise resource planning.
- **IKT** Informasjons- og kommunikasjonsteknologi.
- **IT** Informasjonsteknologi.
- **Difi** Direktoratet for forvaltning og IKT.
- **CRM-systemer** Customer relationship management systemer.
- **CI** Continuous Integration.
- **GUI** Graphical User Interface
- **I/O** Input/Output

1.8 Begrepsavklaringer

Under kommer en liste med begreper som blir omtalt i rapporten. IT-fagfeltet er dominert av engelske terminologier, samtidig som det er endel uttrykk som må forklares.

- **Kognitive prosesser** Prosessene som ligger til grunn for oppfattelse, tenkning og kunnskapservervelse [1].
- **Emperisk analyse** Analyser på data som er samlet inn ved hjelp av systematiske observasjoner og undersøkelser [2].
- **Fagfeltet antropologi** Dette er studiet om mennesker. [3].
- **Fagfeltet sosiologi** Dette er en generell samfunnsvitenskap som 'studerer forutsetninger for, oppkomsten av og samspillet mellom samfunnets ulike elementer' [4].
- **Trelloboard** Dette er et digitalt verktøy der man kan organisere arbeidsoppgaver i forskjellige kolonner, behjelpelig ved gjennomføring av Scrum.
- **Fagfeltet industriell design** Dette er studiet rettet mot design av produkter, varer, tjenester, samt utvikling av grensesnitt og designstrategier [5].
- **Utviklingsteam/Gruppen/Team** Mats Elias Davidsen, Knut Wiig og Kristian Nybakken Lund.
- **Veileder** Helge Hafting.
- **Oppdragsgiver/Kunde** Amund Bruland og Sindre Log.

2 Teori

Kapittel to er delt i tre hoveddeler. Første del tar for seg digitalisering. Her går man gjennom hva det er, hvorfor så mange digitaliseringsprosjekter ofte feiler og hvordan man kan unngå dette. Andre del av kapitlet tar for seg viktigheten av god brukervennlighet i systemapplikasjoner. Siste del av teorien tar for seg metoder og teknologi som er brukt under utviklingen av applikasjonen.

2.1 Digitalisering

Digitaliseringsprosjekter har kommet som en følge av den eksplorative utviklingen av teknologi. Parviainen *et al.* har identifisert digitalisering som en av de største trendene som forandrer samfunnet og forretninger [6]. De mener også at digitalisering kommer til å ha en enorm påvirkning inn mot fremtiden. Det er viktig å påpeke forskjellen på digitalisering og digitisering. Her definerer Parviainen *et al.* digitisering som transformeringen av analog data til digital data, mens Brennen og Kreiss [7] definerer digitalisering som “*the adoption or increase in use of digital or computer technology by an organization, industry, country, etc.*” I denne rapporten skal fokuset være på digitalisering. Siden dette er et så vidt begrep kommer det en mer detaljert definisjon under.

2.1.1 Definisjon av digitalisering

Begrepet digitalisering har et bredt spekter av betydninger. Det er vanlig å beskrive begrepet ut ifra fire nivåer.

Det første nivået handler om å endre prosesser, innhold og objekter, som har vært manuelle, fysiske eller analoge, og transformere de til å bli digitale [8]. Det er den mest grunnleggende og utbredte forståelsen av begrepet. Dette er på lik linje som begrepet ‘digitisering’, men innenfor rammene av en operativ organisasjon [6]. Eksempel på denne formen for digitalisering er Amazons nye strategi innen klesindustrien. I nær fremtid vil Amazon bytte ut sine manuelle arbeidsprosesser, ansatte i store produksjonshaller, med høyteknologiske roboter som svar på den høye etterspørselen av skreddersydde klesplagg [9].

Neste steg ser på digitalisering som en sosioteknisk prosess, som anvender teknikker på tvers av næringer og kontekst på måter som påvirker og former deres underliggende infrastruktur for opprettelse, lagring, distribusjon av innhold, applikasjoner og services [10]. Eksempler på denne formen for digitalisering er hvordan store IT-systemer, eksempelvis ERP-systemer, kan transformere og optimalisere hele forretningsprosesser. Dette nivået berører altså forretningen som en helhet i større grad enn ved det første steget.

Det tredje steget omhandler hvordan digitalisering kan transformere hele bransjer og bryte ned anerkjente forretningsmodeller [11]. Dette kan blant annet komme av at ny, revolusjonerende teknologi forandrer hele forretningsmodeller. Et eksempel på dette er hvordan film distribusjons-bransjen har gått fra utleie av fysiske filmer til streaming, dette var teknologidrevet.

Parviainen *et al.* mener det siste steget ligger på et samfunnsnivå, hvor det skjer endringer av samfunnsstrukturer. Et eksempel på dette er hvordan teknologi har forandret jobber, kommunikasjon, infrastruktur, etc, som har massiv påvirkning på samfunnet [6].

2.1.2 Suksessfaktorer

I følge regjeringen investeres det bare i Norge milliarder av kroner inn i IKT-prosjekter for den offentlige sektor. Et IKT-prosjekt er i de aller fleste tilfeller også et digitaliseringsprosjekt. Nyere undersøkelser fra 2015 viste at det kun var 5% av prosjekter som lyktes på alle kriterier og ca 50% som var lite tilfredsstillende på minst et av kriteriene. Dette gjaldt både for privat og offentlig sektor. Regjeringen benyttet seg av disse kriteriene som mål for graden av suksess [12]:

- Levert nytte.
- Teknisk kvalitet av produktet.

- Kostnadskontroll av prosjektet.
- Tidskontroll av prosjektet.
- Effektivitet av prosjektet.

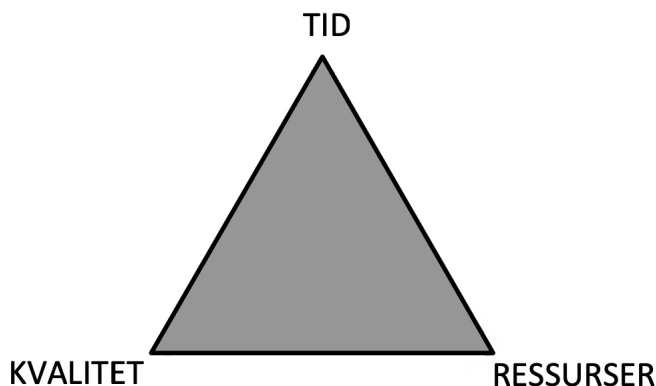
I 2014 gjennomførte Jørgensen en undersøkelse for regjeringen, med 80 respondenter fra offentlig og privat sektor. Her skulle de rapportere inn hvordan deres siste prosjekter hadde gått, og her er prosjektene som ble kansellert før de ble fullført ikke inkludert. Figur 1 viser resultatet av undersøkelsen.

Suksessfaktor	Suksess	Akseptabel	Lav suksess/fiasco
Nytte	36%	59%	5%
Teknisk kvalitet	24%	66%	10%
Budsjettkontroll	38%	40%	22%
Tidskontroll	33%	40%	30%
Prosjekteffektivitet	19%	57%	24%

Figur 1: Suksessfaktor for prosjekter.
Bilde hentet fra regjeringsrapporten [12]

Ut i fra tallene over ser man betydningen av å finne ut hvorfor prosjektene feiler, og ikke minst hva som skal til for å øke sannsynligheten for suksess. Mange organisasjoner blir skremt av slike tall, men som seksjonssjefen for digital forvaltning, Ellen Strålberg påpeker, ligger det også risiko i det å ikke gjennomføre digitaliseringsprosjekter. Hun mener det er essensielt for å møte både dagens og morgendagens utfordringer [13].

Det er prosjekttriangelets dimensjoner som er fundamentet til ethvert digitaliseringsprosjekt. Tid, kvalitet og ressurser, Se figur 2.



Figur 2: Prosjekttriangelets dimensjoner.

Hvert prosjekt er unikt, hvor de har forskjellige prioriteringer i henhold til prosjekttriangelets dimensjoner. I et prosjekt hvor det er tilknyttet stor risiko er fokuset rettet mot kvalitet. Her kan prosjektet anses som suksessfullt om kvaliteten og sikkerheten er ivaretatt, selv om både tid og ressurser går over sine gitte rammer.

I regjeringsrapporten kommer det frem åtte tiltak organisasjoner bør følge for å øke sannsynligheten for at et prosjekt blir vellykket [12].

1. Redusere prosjektstørrelser. Dele opp store komplekse prosjekter inn i flere mindre prosjekter. Dette

vil øke innsikten i hver prosjekt og gjøre de mer gjennomførbare. Ulempen med dette kan være at man fort kan miste oversikten. Her kreves det en god programleder som holder orden på samtlige prosjekter.

2. Arbeide agilt, med hyppige leveranser underveis.
3. Gjennomgående nyttestyring, fra konseptanalyse, gjennom prosjektet og hos mottager av leveransen. Undersøkelser gjort av HIT-nettverket tilsier at prosjekter med gode prosesser for nyttestyring lykkes cirka tre ganger oftere enn prosjekter uten slike prosesser [14].
4. Større fokus på bedre forarbeid med evaluering og valg av leverandører.
5. Om mulig, velge kontraktstyper som ikke gir en fastpris. Slike IKT/digitaliseringsprosjekter er ekstremt vanskelig å estimere, så en fastpris er ikke optimalt.
6. Involvere kunden i stor grad. Kunden sitter med viktig og relevant kompetanse.
7. Bruk av smidig utviklingsprosesser, slik at endringer av krav og målsetninger muliggjøres. Dette er ekstra viktig i IKT/digitaliseringsprosjekter da verken kunden eller leverandør nøyaktig vet hva som skal lages ved prosjektstart.
8. Vektlegging av risiko og usikkerhetsanalyser, slik at involverte aktører blir risikobeviste.

2.1.3 Hvorfor feiler digitaliseringsprosjekter?

I følge Jørgensen kom det i en omfattende rapport fremstilt av McLeod og MacDonell frem at for å lykkes, kreves det tilstrekkelig kunnskap på følgende områder [12]:

- En felles forståelse av målsetninger av samtlige interessenter.
- En felles forståelse av domene mellom kunde og leverandør, hvor begge parter innehar høy kompetanse innen lederskap og teknisk forståelse, hvor kommunikasjonen er på topp.
- Effektive prosesser og gode verktøy for produksjon, testing og realisering av målsetninger og nytteeffekter.
- Prosesser for prosjektledelse og risikostyring.
- Prosjektene får nok tid og ressurser til å realisere målsetningene.
- En god kultur for endring, slik at man lett kan håndtere endringer og læring underveis i prosjektet.

Disse elementene er ganske vidt kjent, grunnen til at mange digitaliseringsprosjekter fortsatt feiler kommer trolig av flere årsaker.

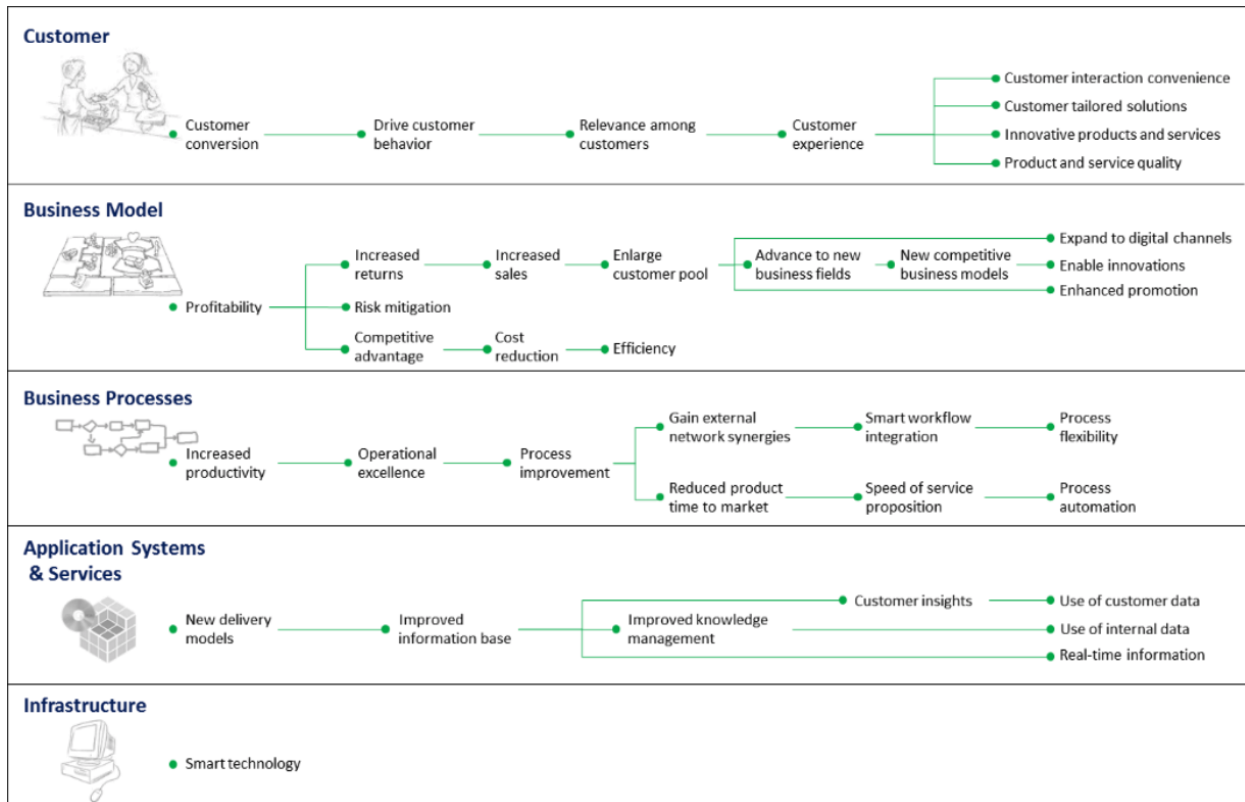
For det første kan organisasjonene ha for liten forståelse av hva som kreves for elementene over. Dette gjør at det ofte startes prosjekter som har alt for høy kompleksitet og fra starten av innehar høy risiko for fiasko. Det kan også skyldes at man overvurderer egen og andres kompetanse på et eller flere områder nevnt over, samt overvurderer sine valg og prosesser.

En annen grunn til fiasko kan være at man ikke klarer å overføre kunnskapen sin til praktiske handlinger. Oppdragsgiver og leverandør vet hva som skal lages og hvordan, men har ikke god nok kompetanse til selve gjennomføringen. Her handler det om å velge rett leverandør og gjøre noe med manglende kompetanse. Det kan også være organisasjonsmessige, politiske eller strukturelle forhold som setter en stopper for suksessen for prosjektet. Eksempel på dette kan være at tidsfrister må settes tidligere enn først antatt, eller manglende lederkapasitet når det kjøres mange parallelle krevende prosjekter. Til slutt er det viktig å se på sammenheng mellom elementene. Det er ikke tilstrekkelig å se på dem individuelt. Det er ofte en dårlig forstått interaksjon mellom dem, noe som kommer av dårlig prosjektledelse [12].

2.1.4 Gevinster ved digitalisering

Difi har et stort ønske om å få flere vellykkede digitaliseringsprosjekter i Norge. Som et tiltak ønsker de å opprette en rådgivnings- og kvalitetssikringsordning som skal hjelpe organisasjoner med å tilfredstille prosjekttriangelets dimensjoner, slik at de planlagte gevinstene kan realiseres [13]. Så hvilke gevinster oppnår typisk slike digitaliseringsprosjekter?

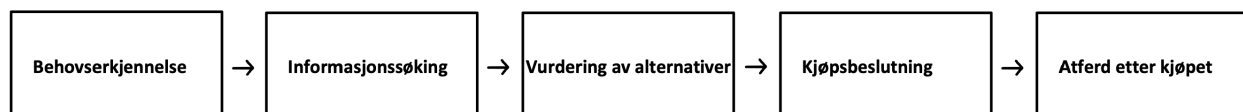
Som Neumeier, A., Wolf, T. og Oesterle, S., adresserer i sin artikkel [15], så er det ekstremt vanskelig å finne nøyaktige mål på gevinstene. Men de har klart å identifisere generelle gevinster knyttet til de ulike leddene i verdikjeden. Figur 3 viser disse gevinstene.



Figur 3: Gevinster fra digitaliseringsprosjekter.
Bilde hentet fra [15], side 491

Som man ser ut ifra figuren skaper digitalisering verdi i alle ledd i verdikjeden. Dette støttes av Parviainen et al. som mener organisasjoner kan kutte hele 90% av sine kostnader ved å benytte seg av digitalisering. Samtidig som de vil gjøre organisasjonen mer automatisk styrkes deres beslutningsgrunnlag Dette ved mer effektiv håndtering av data [6]. Videre hevder de at gevinsten for en organisasjon kan ses ut fra tre synspunkter.

1. **Intern effektivitet.** Dette omhandler alle fordelene internt i organisasjonen. Dette inkluderer blant annet effektivisering av arbeidsprosesser og økt kvalitet, ved eliminering av manuelle steg. Organisasjonene vil også kunne benytte seg av en oversikt av data og arbeidsprosesser sanntid, og kvaliteten på data vil øke betydelig. Dette igjen gjør at man kan bygge opp gode beslutningsverktøy for å hele tiden effektivisere produksjonen. Digitalisering kan også forbedre arbeidsmiljøet ved å kutte ned unødvendige arbeidsoppgaver, og dermed gi mer tid til å utvikle nye givende ferdigheter.
2. **Eksterne muligheter.** Dette er en ekstremt viktig gevinst ved digitalisering. Her forbedrer man kundens interaksjon med organisasjonen, hvor målet er å skape gode kunderelasjoner. Her er det naturlig å se på Kotlers femtrinns modell for kjøpsbeslutningsprosessen til kundene, se figur 4.



Figur 4: Kotlers femtrinns modell for kjøpsbeslutningsprosessen.
[16]

Teknologi, som for eksempel CRM-systemer, har vist seg å forbedre kunderelasjonene massivt og dette fokuserer på disse fem trinnene.

3. **Forstyrrende endringer.** Dette handler om hvordan digitalisering kan foreldre organisasjonens forretningsprosesser. På en annen side kan digitalisering også skape nye forretningsprosesser.

2.1.5 Negativ side ved digitalisering

Digitalisering har sine klare fordeler som diskutert over. Men selv digitalisering kan ha negative virkninger. Teknologistress er noe som fort kan oppstå ved innføring av digital teknologi. I rapporten [17] definerer Torvatn, H. Y., Andersen, K. T. og Kløve, B. teknologistress som: *"arbeidstakeres egenopplevde stress knyttet til bruken av digital teknologi som en del av jobben, innenfor og utenfor normal arbeidstid."*

Her spurte de totalt 2393 personer der de kom frem til at cirka 1/3 opplevde, i varierende grad, negativ eller i alle fall ikke positiv utvikling som en konsekvens av digitalisering på arbeidsplassen. Dette innebærer at de:

...ikke er fornøyde, ikke opplever digital teknologi som nyttig eller produktivitetsøkende, ikke opplever medvirkning eller tilstrekkelig opplæring, føler stress, økt belastning osv.[18].

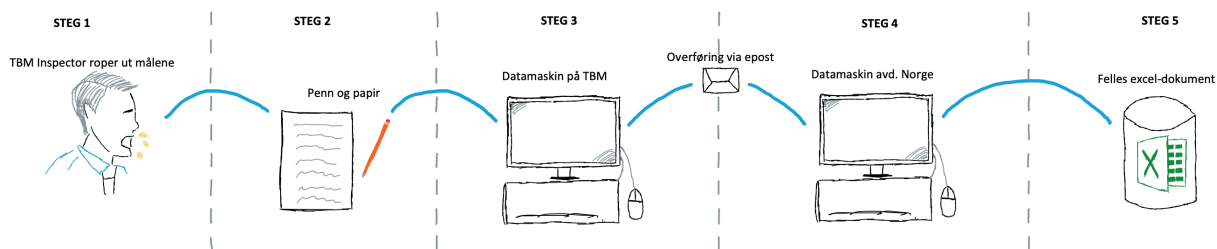
Velger man å se på dårlig digitalisering som et arbeidsmiljøproblem, på grunnlag av stresset det kan gi, er dette et stort problem.

2.1.6 Digitaliseringsprosjektet CutWear

For denne bacheloroppgaven skal det lages en applikasjon for beslutningsstøtte i forbindelse med utskiftning av boreverktøy på tunnelboremaskiner. Dette er et digitaliseringsprosjekt der man skal benytte seg av digital teknologi til å effektivisere utdaterte prosesser. Dette går under det første nivået av digitalisering, hvor man bytter ut penn og papir med digitale løsninger. Se figur 5 og 6 for en visuell oversikt over den eksisterende løsningen mot den nye løsningen.

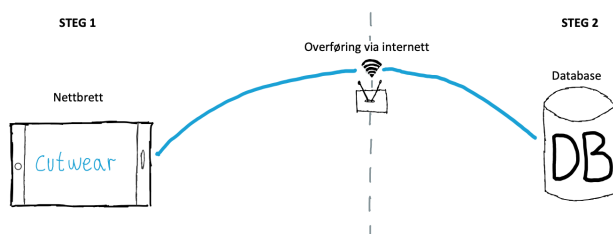
Når digitaliseringsprosjekter omtales er dette oftest prosjekter som budsjetteres i millionklassen. Dette prosjektet er i liten skala i forhold til slike prosjekter, men de fleste prinsipper gjelder fremdeles.

Se kapittel 4.2.1 for hvilke forhåndsregler utviklingsteamet tok for å øke sannsynligheten for å gjennomføre et vellykket digitaliseringsprosjekt.



Figur 5: Eksisterende løsning hos The Robbins Company.

- *Steg 1. En TBM Inspector går inn i borhodet, leser av målingene og roper disse tallene ned til Inspector nr 2.*
- *Steg 2. Inspector nr 2. skriver dette ned på et ark.*
- *Steg 3. Disse notatene blir skrevet inn i et Excel-ark og sendt på mail over til en avdeling i Norge.*
- *Steg 4. Her mottas denne mailen, og dette blir overført til et annet Excel-ark.*
- *Steg 5. Dette Excel-arket fungerer som en felles lagringsplass for inspeksjoner.*



Figur 6: Ny løsning ved bruk av CutWear.

- *Steg 1. En TBM Inspector går inn i borhodet, leser av målingene og skriver dette ned i applikasjonen.*
- *Steg 2. Disse dataene blir automatisk sendt til en felles database.*

2.2 Brukervennlighet

Teknologien blir stadig en mer integrert del av hverdagen til de aller fleste i den vestlige verden. Den gir stadig større muligheter til kommunikasjon og interaksjoner, dette gjør at fagfeltet MMI hele tiden møter nye utfordringer. Dette er alt fra tilfredsstillende av de mer komplekse systemene, til administrering og utnyttelse av den stadig økende tilgjengeligheten av digital informasjon.

En av grunnene til at digitaliseringsprosjekter mislykkes er dårlig interaksjon mellom menneske og teknologi. Om selve brukergrensesnittet er lite intuitivt eller veldig forskjellig fra tidligere programmer vil mennesker fort motarbeide programmet [19]. Med tanke på dette er det svært viktig at man har stort fokus på brukervennlighet.

2.2.1 MMI

MMI omhandler så mye mer enn kun systemutvikling og design av brukergrensesnitt. MMI er et tverrfaglig fagfelt som strekker seg over feltene IKT, psykologi, antropologi, sosiologi og industriell design. Det inneholder også teorier rundt kognitive prosesser, empiriske analyser av brukeres atferd og organisasjonsforståelse [20]. På bakgrunn av dette ser man hvor viktig det er med et tverrfaglig utviklingsteam når man skal lage nye digitale løsninger som skal brukes av mennesker.

2.2.2 Brukbarhet

Brukervennlighet måles ut i fra brukeropplevelse og systemets brukbarhet. Som Nielsen J., påpeker i sin bok 'Usability Engineering' [21] er brukbarhet sammensatt av flere komponenter som alle må være på plass for at systemet skal ha høy brukbarhet. Nielsen definerte brukbarhet ut fra fem komponenter:

1. **Lærbart.** Brukere av systemet skal enkelt kunne sette seg inn i systemet, slik at det hurtig kan bli tatt i bruk.
2. **Effektivitet.** Når brukerne har fått en god forståelse av systemet, skal systemet øke effektiviteten og produktiviteten til brukerne.
3. **Lett gjenkjennelig.** Systemets skal være designet slik at det er lett gjenkjennelig for brukere som har vært fraværende i en lengre periode.
4. **Feilhåndtering.** Frekvensen av feil brukerne kan gjøre i systemet skal holdes til det minimale. Det skal være vanskelig for brukerne å gjøre feil i systemet, og om feil gjøres skal det lett kunne gjenopprettes. Store katastrofale feil skal ikke være mulig å utføre.
5. **Tilfredsstillelse.** Brukerne skal ha et positivt inntrykk av systemet. Systemet bør ikke være en kilde for irritasjon for brukerne.

Dette støttes også av ISO 9241-210 som har definert brukbarhet som:

Extent to which a system, product or service can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use [22]

Når man ser på disse definisjonene ser man hvor stor innflytelse brukerne har for brukbarheten av systemet. I følge Haga og Skjelbek delte også Normans denne oppfatningen, som på 80-tallet kom med begrepet "brukersentrert utvikling". Brukersentrert utvikling er en utviklingsprosess, der brukerne opererer som en viktig interessent som er involvert i utviklingsprosessen.

Denne formen for utviklingsprosess er sentral i dagens systemutvikling og ISO 9241-210 har beskrevet fire brukersentrerte designaktiviteter som bør benyttes [20]:

1. Forstå og spesifisere brukskontekst. Dette gjøres i dialog med brukere.
2. Komme frem til en felles enighet om gode brukerkrav som kan være med å forme designet av systemet.
3. Produsere designløsninger som faktisk møter disse brukerkravene.
4. Evaluere designløsningene gjennom brukertester, og deretter modifisere designløsningene basert på tilbakemeldingene.

Under kapittel 2.1.2 ble det omtalt hva som utgjør suksessfaktorene for digitaliseringsprosjekter. Det er viktig å legge til at det ikke bare er kunden man burde involvere i utvikling, men også brukerne. Dette støttes av Preece J., Rogers Y., og Sharp H., som i boken "Interaction Design: beyond human-computer interaction" snakker om hvordan en brukersentrert utviklingsprosess øker sannsynligheten for at sluttproduktet blir vellykket [23].

Det er essensielt at man finner målgruppen til systemet. Når brukere skal involveres i så stor grad som

beskrevet i brukersentrert utviklingsprosess er det viktig at rett bruker er involvert. Brukere opplever systemer vidt forskjellig, og noe som fungerer godt for en gruppe kan oppleves som dårlig for andre.

2.2.3 Hvordan oppnå høy brukbarhet?

Som diskutert tidligere i kapitlet er det essensielt med en høy brukbarhet i IT-systemer. I følge Haga og Skjelbek kom Normans D., med sju retningslinjer for hvordan man kan oppnå dette [20].

1. Benytte gode etterprøvde metoder fra omverden, samt benytte egne erfaringer. Handler om å lage gode konseptuelle modeller og domenemodeller, samt skrive enkle og oversiktlige brukermanualer.
2. Simplifiser kompleksiteten om mulig. Brukere vil kun huske et begrenset antall handlinger, så det er viktig å organisere strukturen slik at hukommelsen ikke overbelastes.
3. Samsvar mellom utførelse og resultat. Hva som er mulig å utføre for brukeren skal være lett å forstå, samtidig som man skal se effekten av handlingene.
4. Intuitive koblinger mellom handlinger og effekten av dem. Løses visuelt, for eksempel felles grafikk for like knapper.
5. Legge inn begrensninger. Gjøre det enkelt for brukeren å benytte seg av systemet ved at man legger begrensninger på utilgjengelig funksjonalitet. Eksempel på dette kan være hvordan man blokkerer en 'Fortsett' knapp om man mangler å fylle inn nødvendig informasjon.
6. Det skal være vanskelig for brukere å gjøre feil i systemet, og om feil gjøres skal det lett kunne gjenoprettes. Store katastrofale feil skal ikke være mulig å utføre (se punkt 4 i kap 2.2.2).
7. Benytte seg av standardisering der det ikke er mulig med intuitive løsninger. Fordelen med standarder er at de kun må læres én gang, og vil fungere i flere sammenhenger.

2.2.4 Brukeropplevelse

Brukeropplevelsen er det andre punktet som utgjør brukervennligheten. ISO 9241-210 definerte brukeropplevelse som: *“person’s perceptions and responses resulting from the use and/or anticipated use of a product, system or service”* [22].

Videre beskriver de at brukeropplevelsen inkluderer brukerens følelser, oppfatning, preferanser, fysiske og psykologiske responser, oppførsel og prestasjoner som oppstår før, under og etter bruk av systemet, samt tidligere erfaringer av lignende systemer. Til slutt ble det nevnt hvordan brukeropplevelsen er en konsekvens av varemerke, fremstilling, funksjonalitet, systemets ytelse, den interaktive oppførselen og assistanseegenskapene til systemet [22].

Som utvikler og designer må man ha alt dette i bakhodet når nye systemer skal lages. Det er nødvendig at utviklerene har den generelle kunnskapen rundt dette, men samtidig også gjøre undersøkelser på lignende systemer som har gjort det bra. Det kreves også at det blir gjennomført flere iterasjoner av brukertesting før, under og etter utviklingen.

2.2.5 Prototyper

Det finnes flere forskjellige varianter av prototyper. Det er mange faktorer som bestemmer hvilken prototype som egner seg best. Dette kan blant annet bestemmes av hvor i utviklingsprosessen man befinner seg, tiden man har for å levere delprodukt/sluttprodukt, eller ressursene man har tilgjengelig.

Prototyper er relativt lite ressurskrevende, samtidig som de gir en meget god visualisering som kan bedre forståelsen av produktet for de involverte. Prototypene blir også brukt under brukertestene for å få en tidlig og bedre forståelse av brukeropplevelsen, samt brukbarheten av systemet. Dette er viktig for å kutte kostnader og tidsforbruk ved at man slipper å gjøre om store deler av systemet i senere perioder. Det finnes flere typer prototyper, disse er nevnt under [24].

- **Vertikal prototype.** Det at en prototype er vertikal betyr at det er kun har et begrenset antall funksjonaliteter som er tilgjengelig. Om man bruker CutWear som eksempel, kan en vertikal prototype være at man kun logger seg inn og registrere slitasje, mens all annet funksjonalitet er utilgjengelig. Dette benyttes dersom man ønsker å teste ut et begrenset antall funksjonaliteter.
- **Horisontal prototype.** Ser ut som det ferdige produktet, men mangler all funksjonalitet. Dette er mest for å få en oversikt over hvordan brukeropplevelsen av grensesnittet er.
- **Low fidelity prototype.** Denne typen vil ikke inneholde all funksjonalitet, men vil fremdeles gjengi sentrale designideer. Low fidelity prototyper er ofte en kombinasjon av vertikale og horisontale prototyper. Det største kjennetegnet er at de er veldig lite ressurskrevende og benyttes gjerne i starten av utviklingen, eller om man ønsker å teste ut en ny funksjonalitet fort.
- **High fidelity prototype.** Denne typen står i kontrast til low fidelity fordi denne ligner sluttproduktet, men er ofte programmert i et enklere programmeringsspråk. Dette er en mer ressurskrevende prototype og benyttes oftere ved komplekse systemer.

2.2.6 Brukervennlighet i CutWear

For å oppnå en høy brukbarhet og en god brukeropplevelse i CutWear er det laget totalt tre prototyper. Disse er av typen low fidelity, horisontal og vertikal. Disse har blitt flittig brukt i møte med oppdragsgiver for å finne den mest optimale designløsningen.

Utviklingsteamet fikk ikke tilgang til målgruppen for CutWear, da denne gruppen består av et svært begrenset antall mennesker som opererer på byggeplasser rundt om i verden. På tross av dette føler utviklingsteamet at å utføre brukertester vil gi verdi, selv om de må utføres på andre. Brukertestene blir utført på en litt annerledes måte enn om de blir utført direkte mot målgruppen. Målet med testene blir dermed kun å se på om systemet på en generell basis har et intuitivt brukergrensesnitt. Om testene derimot blir utført på rett målgruppe kunne man i større grad finne nye innovative løsninger, da disse personene kjenner prosessene på arbeidsplassen godt.

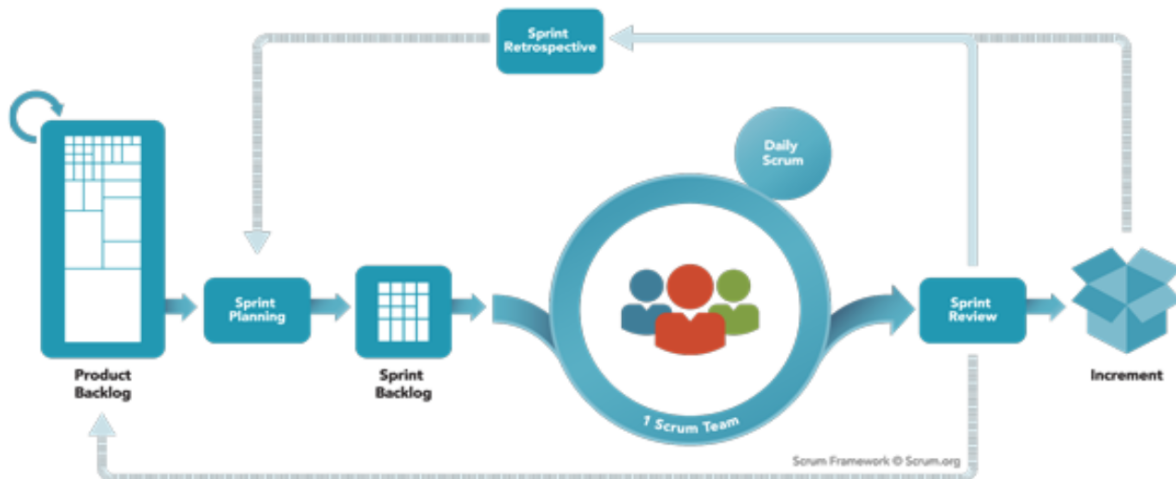
2.3 Metodikk

2.3.1 Agil utviklingsprosess

For denne oppgaven faller valget av utviklingsprosess på en modifikasjon av Scrum. Denne modifikasjonen består av:

- **Product backlog.** Her blir samtlige arbeidsoppgaver som skal gjøres samlet. Denne listen blir utarbeidet av utviklingsteamet i samarbeid med oppdragsgiver.
- **Sprintplaning.** Her planlegger utviklingsteamet hvilke oppgaver som skal gjøres til den spesifikke sprinten. Oppgavene blir hentet fra product backlog.
- **Sprint backlog.** De planlagte oppgavene blir deretter lagt i et trelloboard hvor man kan holde oversikt over hvor langt de har kommet på utviklingen.
- **Sprinter.** Et utviklingsteam som iterativt leverer delprodukter til oppdragsgiver i såkalte sprinter. Dette prosjektet inneholder totalt åtte sprinter.
- **Sprintreview.** Her har utviklingsteamet planer om to forskjellige sprintreview-møter etter hver sprint. Et med veileder og er med oppdragsgiver. Her går man igjennom det som har blitt gjort under sprinten og eventuelle spørsmål. Dette for å sørge for at utviklingsteamet alltid er på rett kurs i følge både veileder og oppdragsgiver.
- **Sprint retrospective.** Dette er interne møter for utviklingsteamet. Her går teamet gjennom hva som er bra, mindre bra og hva som bør gjøres bedre til neste sprint. På denne måten reflekterer alltid teamet over eget arbeid og utvikler seg gjennom perioden.

Les kapittel 3.15 for mer informasjon om hvorfor gruppen valgte Scrum. Vil man lese mer om hva Scrum er og når det egner seg å bruke dette, anbefales det å gå inn på www.scrumguides.org som er den offisielle siden til grunnleggerne av metodikken eller å lese boken til Scwaber K. [26].



Figur 7: Oversikt over Scrum.
Bilde hentet fra Scrum.org [25].

2.3.2 Native kontra Ikke-native applikasjoner

Native applikasjoner er applikasjoner som er skrevet for spesifikke plattformer og som kan kjøres direkte på enheter som bruker plattformen. Begrepene Native og Ikke-native brukes hovedsakelig om mobile applikasjoner. Native applikasjoner er for eksempel applikasjoner man kan finne i App Store for iOS eller Play Store for Android. I bunn og grunn er den store forskjellen at applikasjoner skrives i forskjellige språk og når man tar et valg om språk, tar man også et valg om hvilke plattformer applikasjonen kan kjøres på. Man er da avhengig av at enheten som skal brukes støtter samme plattform. Ikke-native applikasjoner, som også er kjent som hybrid eller web applikasjoner, er som regel applikasjoner som man finner på internett og kan brukes på alle enheter med internett og en nettleser. Dette står i kontrast til native, hvor man må laste ned og installere. Fordelen med native applikasjoner er at man kan skrive kode som er spesifikk for plattformen og man kan deretter i større grad optimalisere applikasjonen med tanke på ytelse og plattform-funksjonalitet. Man får også offline funksjonalitet, noe man ikke får ved en web applikasjon. [27]

2.3.3 Rammeverk

Et rammeverk er en samling programmer som brukes sammen for å utføre en oppgave, dette gjør programmering enklere og mer oversiktlig, i motsetning til rene programmeringsspråk som handler mer om hvordan man kommuniserer med datamaskinen gjennom syntaks og semantikk. Rammeverk er verktøy, hvor andre allerede har gjort grovarbeidet, slik at man kan fokusere på oppgaven som skal løses. Rammeverk kan også tilby løsninger som det originale språket ikke har eller tilby bedre løsninger enn nåværende løsning. Rammeverk kan også ha den fordelen at oppdateringer kommer oftere enn med programmeringsspråk, slik at det ofte kommer forbedrede versjoner, men dette kan komme på bekostning av at det ikke er testet godt nok [28].

2.3.4 Container

Containere tillater utviklere å pakke en applikasjon med alle de forskjellige delene som trengs, for eksempel eksterne biblioteker og andre avhengigheter, og 'shippe' det ut som en pakke. Man kan da være sikker på at applikasjonen kjører som den skal på hvilken som helst annen plattform enn applikasjonen den ble utviklet på. Dette kan minne om en virtuell maskin, men containere kjører som regel nedstrippede versjoner av operativsystemer hvor unødvendig programvare er fjernet, dette gir bedre ytelse. Det finnes verktøy som kan hjelpe til med å styre mange forskjellige containere samtidig, dette gjør at man får et effektivt og oversiktlig produksjonsmiljø [29].

2.3.5 Kontinuerlig Integrasjon

Kontinuerlig integrasjon (CI) er en utviklingspraksis der man sjekker inn kode i for eksempel et versjonskontroll system, og deretter settes det i gang en automatisk prosess, kalt pipeline, hvor koden kompiles/bygges, tester blir kjørt og tilbakemelding blir gitt. Man prøver i stor grad å automatisere prosesser som må utføres ved visse hendelser i utviklingen, slik at man får en kontinuerlig tilbakemelding på statusen og kvaliteten til systemet som blir utviklet [30]. Et eksempel på en CI pipeline er når en utvikler laster opp koden sin til et versjonskontrollsystem, som trigger et annet eksternt program/tjeneste som har ansvar for å sjekke om programmet kompilerer og gi tilbakemeldinger på dette. Hvis programmet kompilerte, kjøres testene og resultatene blir sendt videre til en annen applikasjon som samler statistikk om testene og gir tilbakemelding til utvikleren.

CI sammen med continuous deployment, gjør at man kan automatisere hele prosessen med å produsjonssette en applikasjon, ved å ha en pipeline som verifiserer at applikasjonen er klar for produksjon og levering. Det betyr at brukeren sitter med den nyeste versjonen av applikasjon når pipelinen er fullført. Hvis et ledd i pipelinen mislykkes, stopper den opp og gir tilbakemelding til utvikleren.

3 Valg av teknologi og metode

I dette kapittelet begrunnes valgene av teknologier og metodene benyttet under prosjektet.

3.1 Windows

Gruppen har valgt å utvikle applikasjonen for å kjøre hovedsakelig på Windows. Dette er på grunn av at det ble utført undersøkelser på hvilket operativsystem som brukes på nettbrett som tåler krevende fysiske forhold. Denne undersøkelsen ga inntrykket av at de fleste nettbrett med gode tilbakemeldinger kjørte Windows. Å lage en Windows desktop applikasjon muliggjør også at CutWear kan kjøres på datamaskiner som brukes andre steder i produksjonskjeden av kuttere og andre interessenter. Windows er også godt kjent for utviklerne, også for de fleste brukere rundt om i verden, da Windows har en markedsandel på cirka 40% [31].

3.2 Utviklingsmiljø

For å kunne utvikle denne applikasjonen har gruppen valgt å gå til innkjøp av et nettbrett som kjører Windows 10. Dette muliggjør å teste applikasjonen på et nettbrett som ligner på de nettbrettene som er planlagt å brukes i en TBM. Nettbrettet som har blitt brukt til utviklingen er en Microsoft Surface Go.

3.3 ReactJS

Dette er et JavaScript rammeverk designet for å utvikle brukergrensesnitt, noen av grunnen til at gruppen valgte å utvikle frontend ved hjelp av React er at rammeverket baserer seg på komponenter der det er enkelt å håndtere endringer i applikasjonens tilstand. Bruk av disse komponentene gjør at man enkelt kan gjenbruke kode, noe som effektiviserer utviklingsprosessen. En annen fordel med React er at det er open source. Det er også godt dokumentert, dette er til hjelpelig når man utvikler. Hvis man hadde valgt å bygge brukergrensesnittet med språk som Java eller C++ ville det gått utover brukervennligheten, da GUI bibliotekene til disse, som regel er ment for applikasjoner hvor man har mus og tastatur tilgjengelig, samtidig ville det vært mer utfordrende og tidkrevende å utvikle. React ble valgt over React Native fordi man så at det var mer anvendelig. En annen grunn til dette er at man mente at hvis man etterhvert fikk internett i alle TBMer, kunne CutWear kjøres som en web-applikasjon i nettleser istedenfor å kjøre på iOS eller Android som hadde vært tilfellet med React Native. React er også veldig populært om dagen og derav veldig relevant for arbeidslivet, samtidig som det er godt dokumentert. JavaScript og React har også et rikt utvalg av eksterne biblioteker.

3.4 Electron

På bakgrunn av at man valgte å utvikle applikasjonen i JavaScript, samtidig som man så at det var behov for offline funksjonalitet, sto Electron fram som det beste valget. Hvis man hadde lagd en touch-vennlig web applikasjon, så ville man vært avhengig av internett for å kunne hente ned koden i en nettleser. Det betyr at man ikke ville fått tilgang til CutWear hvis man ikke hadde internett. Med Electron, kan man pakke sammen alle filene som trengs for å gjøre applikasjonen til en native desktop applikasjon. Electron baserer seg på Chromium og koden som kjøres i Electron, kan i stor grad brukes i en vanlig nettleser. Den store forskjellen er hvordan Electron håndterer I/O, da dette må gjøres gjennom en egen Electron prosess.

3.5 Node.js med Express

For lagring og skriving av data til database, ble backenden utformet som et REST-API. Med tanke på at man valgte JavaScript i frontend, valgte man også JavaScript på backend med Node.js og Express. Express er et web rammeverk for Node.js som gjør at man kjapt og enkelt kan komme i gang med APIet og slippe å bruke tid på web konfigurasjon. Teknologiene er også godt dokumentert og har et godt utvalg av eksterne bibliotek. Siden man ikke er avhengig av veldig høy ytelse, da serveren stort sett bare skal skrive og lese data, samtidig som den ikke utsettes for stor belastning døgnet rundt, fungerer Node.js bra til dette bruksområde.

3.6 MySQL

MySQL er et SQL-basert databasesystem. Primært falt valget på MySQL fordi det er mye brukt ute i arbeidslivet, det er open source og utviklingsteamet har mye erfaring med det. Ved bruk av relasjoner opprettholdes dataintegriteten i MySQL. En annen fordel er dokumentasjonen rundt MySQL. Dette gjør det lett for utviklingsteamet, men også for de som eventuelt skal videreutvikle systemet.

3.7 Docker Compose

Siden systemet skal driftes av mennesker som kanskje ikke kan teknologiene eller har tilgang til samme utviklingsmiljø som utviklergruppen, har man valgt å bruke Docker-Compose for å gjøre det enklere å drifte applikasjonen. Ved hjelp av en førstegangs installasjon, kan man starte og stoppe APIet med en kommando. Docker er et verktøy for å pakke og konvertere applikasjoner til Docker images, slik at man kan styre kjøremiljøet uten å vite eksakt hvor det skal kjøres. Docker-compose er et verktøy som styrer flere Docker images, som gjør at man blant annet kan kjøre flere instanser av samme applikasjon for å få bedre ytelse. Docker-compose oppsettet og referanser for CutWearServer finner man i vedlegg F.

3.8 Electron-builder

Electron-builder er et bibliotek for å bygge og pakke Electron applikasjoner. Dette biblioteket ble valgt fordi man kan få applikasjonen en som hvilken som helst filtype, med eller uten en installer. Det vil si at CutWear kan kjøres på Windows, OSX og Linux hvis ønskelig. Installer filen gjør at CutWear kan installeres og avinstalleres på en PC ved hjelp av et klikk. For å slippe mulige feil med pakking, kan man også pakke applikasjonen med Docker [32]

3.9 Github

Github ble valgt som versjonskontroll vert fordi det er et verktøy alle på teamet kjenner til og har brukere på. Github tilbyr gratis private repositories, som gjør at CutWear kan lagres der. CutWear er delt opp i to repositories, server og klient, fordi det virket som det mest oversiktlige.

3.10 Travis CI og Codecov

Travis CI er et verktøy for å bygge og teste kode. Travis CI er en del av kontinuerlig integrasjons (CI) pipelinen til CutWear og har som ansvar å kjøre testene og publisere resultatet til Codecov. codecov.io er et verktøy for å publisere og holde oversikt over kodedekningen til applikasjonen. Codecov er også en del av kontinuerlig integrasjons pipelinen til CutWear. Resultatene av testene fra Travis CI blir publisert her. Codecov tilbyr mer avansert informasjon om kodedekning enn de lokale kodedekningsverktøyene. Travis CI og Codecov ble valgt fordi utviklerteamet har god kjennskap til verktøyene gjennom andre prosjekt, men i løpet av prosjektet fant man ut at Travis CI har en grense for hvor mange 'builds' man kan ha med private repositories, og man oppnådde det i løpet av de siste ukene av prosjektet. Man så deretter etter andre CI tjenester, men kom frem til at man ikke hadde tid til å implementere dette.

3.11 Winston

Winston er et loggeverktøy for Node.js. Winston erstatter 'console.log' for server i dette prosjektet og gjør det mulig å få bedre formatert logger med tidsstempel, man kan i tillegg sortere loggene etter viktighets-grad, og eventuelt skrive de viktigste hendelsene til fil.

3.12 jest

Jest er et teste-bibliotek for JavaScript. Dette ble valgt fordi utviklerteamet har erfaring med dette fra før. Jest brukes til enhetstester.

3.13 jsDoc

For kodedokumentasjon ble jsDoc brukt, dette ble valgt fordi det virket enkelt og oversiktlig.

3.14 Onsen UI

Onsen er et open source bibliotek som brukes til å designe brukergrensesnitt. Det tilbyr ferdige komponenter, generell styling av brukergrensesnitt og en rekke eksempler på hvordan man kan bruke komponentene. Det er derfor til stor hjelp å bruke et slikt bibliotek når man skal utvikle brukergrensesnitt.

3.15 Utviklingsprosess

Valget av utviklingsprosess falt på en modifikasjonen av Scrum som forklart i forrige kapittel. I teorien kom det frem at en av suksessfaktorene ved digitaliseringsprosjekter var å benytte seg av en agil utviklingsprosess. Det er flere ulike prosesser innenfor agil utvikling. Scrum ble valgt på grunnlag av erfaringene utviklingsteamet har med denne metoden. Som nevnt valgte teamet å tilpasse prosessen for å optimalisere utbytte. Siden teamet kun besto av tre personer var det ingen 'Scrum Master', på grunn av dette valgte man å ha direkte kontakt mellom utviklingsteam og interessentene.

Seremonien 'daily Scrum' ble overflødig. Med tre utviklere hadde samtlige full kontroll over hva som ble gjort til en hver tid. Gruppen satt alltid sammen og jobbet, og dette førte til god kommunikasjon innad i teamet [25].

4 Resultater

Dette kapittelet er delt inn i tre deler, 'Vitenskapelige resultater', 'Ingeniørfaglige resultater' og 'Administrative resultater'. Den første delen tar for seg programmet CutWear. Den andre delen beskriver status for hvert delmål i prosjektet. Den siste delen tar for seg hva som var planlagt og hva som faktisk ble gjort i hver sprint.

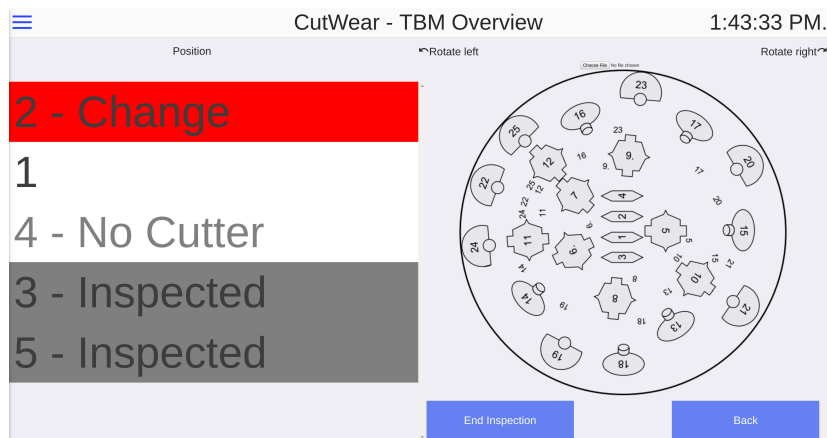
Resultatene blir fremstilt på denne måten, som ble ansett som relevant for den etterfølgende diskusjonen.

4.1 Vitenskapelige resultater

CutWear er en applikasjon med fire forskjellige brukergrupper, av disse får brukergruppene TBM Inspector og Project Supervisor mest funksjonalitet. Under beskrives de forskjellige brukergruppene og hvilken funksjonalitet de har.

TBM Inspector

Dette er den mest sentrale brukergruppen i CutWear og den har dermed mest funksjonalitet, med tilhørende beslutningsstøtte. Oppgaven til en inspektør er å gjennomføre inspeksjoner og lagre informasjon fra disse. Når en inspektør logger inn vil brukeren få mulighet til å starte eller fortsette en inspeksjon på en av TBM'ene i prosjektet brukeren jobber på. Når brukeren går inn på en TBM vil programmet først vise siden der man legger inn inngangsdata for denne inspeksjonen, trykker man seg videre fra denne siden kommer man til hovedsiden for den valgte TBM'en.

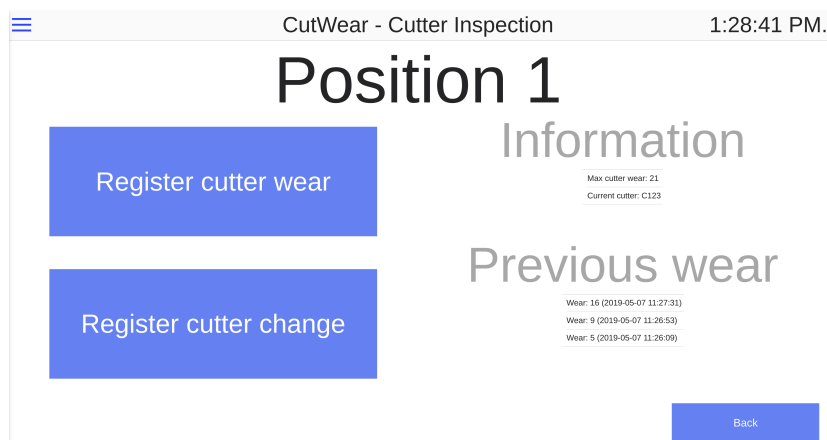


Figur 8: TBM Inspector hovedside.

Når inspektøren er ferdig med å registrere all data til inspeksjonen, trykker brukeren på 'End inspection' som er plassert nede til høyre på hovedsiden for TBM Inspector. Når dette skjer prøver applikasjonen å få kontakt med databasen for å sende inspeksjonsdata, hvis nettbrettet som CutWear kjører på ikke har internett-tilgang ved opplasting, vil inspeksjonsdata lagres lokalt. Hvis nettbrettet er koblet til internett vil informasjonen bli lastet opp og lagret i databasen. Så lenge en bruker er logget inn i applikasjonen vil CutWear forsøke å laste opp inspeksjonsdata som ikke er blitt lastet opp hvert minutt. Hvis inspektøren logger ut eller bytter TBM midt i en inspeksjon må inspeksjonen settes på pause, dette er for at endringene ikke skal gå tapt. Dette gjøres ved å velge 'Log out' eller 'Change TBM' fra toolbar-menyen som vises når man trykker på det blå ikonet helt øverst til venstre på skjermen. I denne menyen får man også muligheten til å få hjelp med applikasjonens funksjoner og slette endringene gjort på en inspeksjon. På hovedsiden for TBM Inspector har brukeren muligheten til å laste opp et bilde av borhodet for å visualisere plasseringen av kutterene, det vil også alltid befinne seg en liste med alle posisjoner i borhodet til venstre på denne siden. Posisjonene i denne listen kan ha følgende koder:

- **Change (rød bakgrunn):** Kutteren i denne posisjonen trenger mest sannsynlig å bli byttet, dette bestemmes enten ved at slitasjen som er registrert er over de grensene satt under registrering av TBM eller ved at brukeren har valgt å markere denne kutteren under inspeksjon.
- **Inspected (grå bakgrunn):** Det er enten registrert slitasje innenfor grensene satt eller registrert et bytte i denne posisjonen under denne inspeksjonen. Posisjoner med denne koden vil bli plassert nederst i listen.
- **No cutter:** Det er ikke registrert en kutter i denne posisjonen.

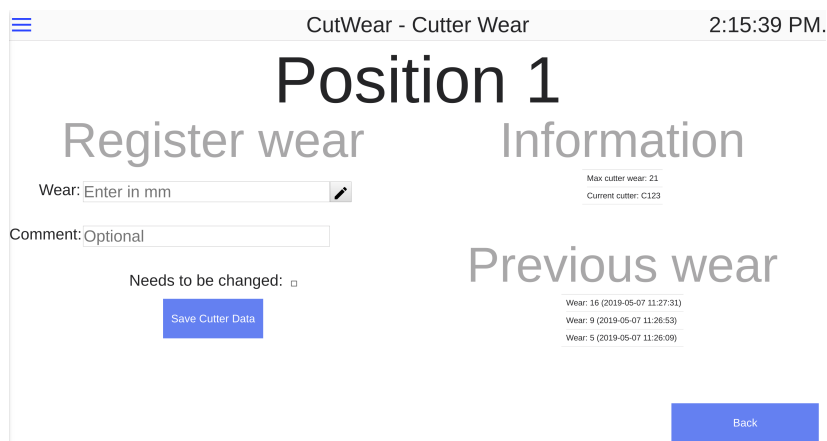
Listen med disse kodene kombinert med et bilde av borhodet gir inspektøren et overblikk over fremdriften i den pågående inspeksjonen. Hvis brukeren trykker på en av posisjonene i denne listen vil han få opp posisjonens hovedside, der vil den høyre delen av siden inneholde informasjon som kan brukes som beslutningsstøtte.



Figur 9: TBM Inspector, hovedside for posisjon.

På den venstre delen av siden får brukeren to valg, når et av disse velges vil venstre del av siden endre seg mens den høyre delen som inneholder data for beslutningsstøtte forblir den samme. De ulike valgene brukeren kan gjøre på denne siden vil føre til følgende endringer:

Register cutter wear: Her kan brukeren registrere slitasje på kutteren i den aktuelle posisjonen, samt markere at denne posisjonen trengs å byttes senere i inspeksjonen.



Figur 10: TBM Inspector, registrer slitasje.

Register cutter change: Her kan brukeren registrere bytte av kutter, det er også mulig å legge inn slitasje her for å lagre mest mulig data om byttet.

CutWear - Cutter Change 2:16:33 PM.

Position 1

Register change

Reason: **abrasive wear**

New Id:

Comment:

Wear:

Information

Max cutter wear: 21
Current cutter: C123

Previous wear

Wear: 16 (2019-05-07 11:27:31)
Wear: 9 (2019-05-07 11:26:53)
Wear: 5 (2019-05-07 11:26:09)

Figur 11: TBM Inspector, registrer bytte.

TBM Supervisor

Denne brukergruppen er personer som har et overordnet ansvar i et prosjekt. Brukergruppen vil ha tilgang til tre funksjoner i applikasjonen, alle funksjonene operer kun innenfor prosjektet brukeren er tilknyttet. Disse funksjonene er å registrere en ny TBM, registrere en ny TBM Inspector og se data registrert i de ulike inspeksjonene. Funksjonene registrere en TBM og TBM Inspector er nødvendig funksjonalitet for at man skal kunne sette opp et prosjekt og starte registrering av data. Funksjonaliteten ved at en supervisor skal kunne se data lagret fra en inspeksjon sørger for at brukeren kan få en oversikt over prosjektets fremgang. Dermed kan dette regnes som beslutningsstøtte til hvordan brukeren skal styre prosjektet. For å se denne informasjonen må brukeren først velge TBM og deretter inspeksjon. Når brukeren har valgt en inspeksjon vises følgende side, der informasjonen fra valgt inspeksjon blir fremvist:

CutWear - View TBM 8:21:25 PM.

Inspection Details

Information

TBM: 40003201
Inspection finished: Tue May 07 2019 14:59:39
Bored meters: 36
Bored hours: 19
Number of cutter changes: 4

Changes

Position 1, reason for change: abrasive wear
Position 4, reason for change: damaged hub
Position 5, reason for change: other
Position 7, reason for change: abrasive wear

Wear

Position 1, wear registered: 19 mm
Position 3, wear registered: 12 mm
Position 4, wear registered: 48 mm
Position 5, wear registered: 23 mm

Figur 12: TBM Supervisor, inspeksjonsdata.

Super User

Dette er en brukergruppe som ikke er bundet til et prosjekt og den eneste funksjonen til denne brukergruppen er å laste ned all tilgjengelig data fra alle inspeksjoner, uavhengig av prosjekt eller TBM. Brukeren kan

deretter bruke den nedlastede informasjonen til analyse eller lignende.

Admin

Admin brukergruppen brukes til å opprette prosjekter slik at man kan starte nye prosjekter kun ved hjelp av applikasjonen. En Admin må også opprette en TBM Supervisor som kan registrere ytterligere informasjon når et prosjekt opprettes.

4.2 Ingeniørfaglige resultater

Målene for dette prosjektet er beskrevet i kapittel 1.1. Her er målene delt inn i tre kategorier, 'forretningsmessige mål', 'resultatmål' og 'prosessmål'.

4.2.1 Forretningsmessige mål

Resultatet av det forretningsmessige målet måles ved å se på graden av suksess ved digitaliseringsprosjektet. Kapittel 2.1.2 beskriver hvilke faktorer som skal til for å øke sannsynligheten for suksess. Under går man igjennom de åtte suksessfaktorene og ser hva som er gjort for å tilfredstille dem.

Resultat	
Suksessfaktor	Hva som er gjort
1. Redusere prosjektstørrelser. Dele opp store komplekse prosjekter inn i flere mindre prosjekter. Dette vil øke innsikten i hver prosjekt og gjøre de mer gjennomførbare. Ulempen med dette kan være at man fort kan miste oversikten. Her kreves det en god programleder som holder orden på samtlige prosjekter.	Ved prosjektstart ønsket både utviklingsteam og oppdragsgiver en kartlegging av hele systemet. Når hele systemet var påtenkt, med prototyper, domene-modeller og ER-diagram så utviklingsteamet at dette ble et alt for komplekst og stort system for hva som var mulig å levere innen tidsfristen. Derfor kom utviklingsteamet frem til hvilke deler av systemet som var realistisk å levere med godkjenning av oppdragsgiver. På denne måten holdt utviklingsteamet prosjektstørrelsen innenfor realistiske rammer.
2. Arbeide agilt, med hyppige leveranser underveis.	Prosjektet fulgte en egendefinert utgave av den agile utviklingsprosessen Scrum.
3. Gjennomgående nyttestyring, fra konseptanalyse, gjennom prosjektet og hos mottager av leveransen.	Det har vært vanskelig å opprettholde dette for prosjektet. De faktiske brukerne har ikke vært en del av prosessen, noe som gjør det vanskelig å måle faktisk nytte av produktet. Det har heller ikke vært en produkteier knyttet til prosjektet som hele veien kan måle nytte mot kostnad knyttet til ulike delmål. For å kompensere for dette, har utviklingsteamet hatt et større fokus mot nettopp dette enn hva som er vanlig for utviklere.
4. Større fokus på bedre forarbeid med evaluering og valg av leverandører.	Dette punktet går mest mot hvordan en oppdragsgiver må gjøre et godt forarbeid for valg av utviklingsteam. Men et grundig forarbeid for utviklingsteam er også viktig for å i det hele tatt vite om man kan klare å levere dette. Utviklingsteamet gjorde meget detaljert og grundig forarbeid før selve utviklingen begynte. Se kapittel 4.3.

5. Om mulig, velge kontraktstyper som ikke gir en fastpris. Slike IKT/digitaliseringsprosjekter er ekstremt vanskelig å estimere, så en fastpris er ikke optimalt.	Dette prosjektet ble gjennomført uten noe form for betaling, men det hadde en streng leveringsfrist, noe som ikke er helt optimalt for et slikt prosjekt. Men siden dette var gratis for oppdragsgiver var det større rom for å endre kravene underveis for å kunne levere et fungerende program innen leveringsfristen.
6. Involvere kunden i stor grad. Kunden sitter med viktig og relevant kompetanse.	Vedlegg A viser at det ble gjennomført hyppige møter med kunden gjennom hele prosjektet. Disse møtene ble holdt som en del av utviklingsprosessen, i noe som kalles sprint review.
7. Bruk av smidig utviklingsprosesser, slik at endringer av krav og målsetninger muliggjøres. Dette er ekstra viktig mot IKT/digitaliseringsprosjekter da verken kunden eller leverandørnøyaktig vet hva som skal lages ved prosjektstart.	Ved prosjektstart viste verken utviklingsteam eller oppdragsgiver nøyaktig hva som skulle lages, men ved flere iterasjoner av kravspesifikasjonen og flere prototyper kom partene til en felles enighet om sluttprodukt. Dette var mulig ved bruken av Scrum.
8. Vektlegging av risiko og usikkerhetsanalyser, slik at involverte aktører blir risikobeviste.	Ved prosjektstart ble det opprettet en risikoanalyse slik at utviklingsteamet var klar over prosjektets risikoer og graden av alvorlighet ved hvert element. Det skal også påpekes at det var på forhånd lav risiko for oppdragsgiver knyttet til prosjektet da dette var en bacheloroppgave, se vedlegg H.

Tabell 1: Forhåndsregler tatt for å oppnå et suksessfullt digitaliseringsprosjekt

4.2.2 Resultatmål

Oversikt over user stories	
user stories som ble fullført	user stories til videreutvikling
1,2,3,4,5,7,8,9,11,17,21,22,23,24,25,26,27,30,31,32,33,34,35,36	6,10,12,13,14,18,19,20

Tabell 2: Oversikt over user stories

Når det gjelder selve applikasjonen måles resultatet ut fra måloppnåelse på kravspesifikasjonen, som er godkjent av oppdragsgiver.

Kravspesifikasjonen er et essensielt dokument, både for utviklerne og oppdragsgiverene. På grunn av sin essensielle rolle i prosjektet ble det laget tilsammen tre versjoner. Revisjonene viser endringene i prosjektet. Første versjon prioriterte alle de gitte rollene like høyt, med mye funksjonalitet for alle rollene.

Andre versjonen ble utarbeidet i starten av sprint tre. Her hadde utviklingsteamet og oppdragsgiver tilegnet seg nok bakgrunnsinformasjon til å vite hva som var realistiske mål. Her ble det flere justeringer av prioriteringer.

Siste versjon ble utarbeidet på slutten av prosjektet etter flere møter med oppdragsgiver, hvor det kom frem nye ønsker. Denne versjonen gjenspeiler applikasjonen ved leveringstidspunkt. Samtlige user stories med prioritering fra og med 7 ble implementert. Se vedlegg C2 for detaljer på hvilke som er endret.

Tabell 3 viser versjon 1 og 3 opp mot hverandre i forhold til hvilke user stories som lå over og under prioritering 7.

Versjon 1 vs. Versjon 3.		
Versjon	User stories med prioritering 7 og over	User stories med prioritering 6 og under
Versjon 1	22 / 28	6/28,
Versjon 3	24 /36	12 / 36

Tabell 3: Versjon 1 vs. Versjon 3, kravspesifikasjon

Under kommer en kategorisering av user storiene og i hvilken grad de ble realisert.

Funksjonalitet i program

Partene kom til enighet om at rollen TBM Inspector var den viktigste rollen for systemet, så kravene knyttet til denne rollen var høyt prioritert, dette inkluderer user stories; 1, 2, 3, 4, 5, 7, 8, 9, 26, 30, 31, 32, 33 og 36. Samtlige av disse ble ferdigstilt, men noen på litt ulike måter enn først antatt. Scenarioene for user storiene er kun eksempler på hvordan man kunne løse det, prototype versjon 3 viser disse user storiene på best måte. Se vedlegg C5.

For at programmet skulle være operativt var det nødvendig med rollene 'Admin' 'Super User' og 'Project Supervisor' med begrenset funksjonalitet. Den begrensede funksjonaliteten besto av user storiene; 8, 11, 17, 21, 22, 23, 24, 25, 27, 28, 34 og 35.

Samtlige av disse kravene ble implementert.

Når det gjelder kravene til disse tre rollene med lav prioritet, user stories; 10, 12, 13, 14, 18, 19 og 20 ble de ikke ferdigstilt. Det var en felles forståelse mellom partene at disse ikke skulle ferdigstilles, men disse ble fortsatt tatt i betraktning som deler av systemet når databasen, domenemodellen, ER-diagrammet og den første prototype ble laget.

Funksjonalitet til videreutvikling

Totalt skal applikasjonen ha seks unike roller. Rollene 'Workshop Supervisor' og 'System Administrator' er ikke en del av applikasjonen ved leveringsfristen. User stories; 15, 16 og 29, tilhører disse brukerne og er derfor ikke fullført. Men databasen, første prototype, domenemodellen og ER-diagrammet tar hensyn til disse brukerne. De tar også hensyn til user storiene som inkluderes i kategorien 'til videreutvikling', som er; 6, 10, 12, 13, 14, 18, 19 og 20.

4.2.3 Prosessmål

På tidspunktet for rapportskrivningen er det for tidlig å si noe om måloppnåelsen for prosessmålene til The Robbins Company og NTNU bygg -og miljøteknikk.

Når det gjelder utviklingsteamet sitter de med en felles opplevelse av sine måloppnåelser. Utviklingen av prosessmetodikken kan man lese mer om under kapittel 4.3 og lese vedlegg I.

Et annet prosessmål for utviklingsteamet var å skrive bedre litterære dokumenter, med det dette innebar, se 1.3. Alle tre deltagerne deltok aktivt i skrivingen av hovedrapporten, samt de ulike vedleggene knyttet til prosjektet. Gjennom denne prosessen fikk deltagerne den ønskede kompetanseoppbyggingen.

Det siste prosessmålet var å få mer erfaring mot arbeidslivet. Her har deltagerne fått erfart hvordan man skal planlegge og gjennomføre større prosjekter ved kundemøter, lage kravspesifikasjon, opprette en felles forståelse av domene og planlegging av arbeidsoppgaver til gitte tidsfrister.

4.2.4 Tester

I perioden 25.03.19 - 05.04.19 ble det utarbeidet en rekke brukbarhetstester. Under disse testene kom det frem en rekke forslag til funksjonalitet for programmet. Resultatet av disse testene ligger under vedlegg B. Det ble ikke gjennomført stresstest og akseptansetest, som planlagt i visjonsdokumentet. Det er skrevet enhetstester for serveren, men ikke for klienten i programmet.

4.3 Administrative resultater

4.3.1 Timeregnskap

Utviklingsteamet førte timelister og sendte de inn til veileder på en ukentlig basis. Se vedlegg J for timelistene. Her følger en liten beskrivelse av hva de enkelte har gjort til enhver tid.

4.3.2 Utviklingsprosess

Utviklingsprosessen for dette prosjektet var Scrum. Prosjektet inneholdt i alt 8 sprinter. Under kommer en oppsummering av hva som var målet for de ulike sprintene, samt hvilke forutsetninger og avhengigheter de hadde.

Sprint 1

Sprint 1 ble gjennomført i perioden 28.01.2019 - 08.02.2019.

Mål

Hovedmålet for sprint 1 var å etablere en fullkommen forståelse av domenet. Det skulle utarbeides en domene-modell. Kravspesifikasjon skulle også stå klar ved slutten av sprinten. Kravspesifikasjonen skulle utredes som user stories hvor de rangeres fra 1-10. Alle tekniske spesifikasjoner rundt utvikling og kjøring av systemet skal være på plass. Utviklingsteamet skulle også begynne på hovedrapporten, samt ferdigstille første versjon av visjonsdokument. Det skulle lages en risikoanalyse slik at utviklingsteamet var klar over mulige risikoer knyttet til prosjektet.

Avhengigheter og forutsetninger

For å oppnå målene ble det satt opp en rekke møter med oppdragsgiverne. Her ble alle parter enig om hvordan oppgaven skulle løses. Det første som måtte på plass var en felles forståelse av domenet. Etter dette ville utviklingsteamet sammen med oppdragsgiver komme frem til kravspesifikasjonen. For å komme frem til de rette tekniske spesifikasjonen ble det satt opp et møte med Robbins sine egne ingeniører i Seattle, USA. Med dette møte, sammen med eget undersøkelser, skulle alle tekniske spesifikasjoner være satt.

Resultat

Under sprint 1 ble det etablert en fullkommen forståelse av domenet. Det ble også utarbeidet en domene-modell, kravspesifikasjon, første versjon av visjonsdokument og en risikoanalyse. Teamet begynte også på hovedrapporten. Alle tekniske spesifikasjoner rundt utvikling og for kjøring systemet kom ikke på plass under denne sprinten.

Sprint 2

Sprint 2 ble gjennomført i perioden 14.02.2019 - 22.02.2019

Mål

I sprint 2 skulle databasen settes opp. Dette inkluderte design av ER-modellen og konfigurering av en MySQL database hos NTNU. Utviklingsteamet skulle også sette opp konfigurasjoner for kontinuerlig integrasjon og annet tekniske spesifikasjoner for å komme i gang med selve utviklingen. Her skulle teamet spesielt fokusere på rammeverk og biblioteker for å få et enkelt, brukervennlig og effektivt brukergrensesnitt som beskrevet i user story tre, se vedlegg C.

Et annet mål var å begynne med utviklingen. Her skulle front – og backend arbeides på parallelt. Hovedmålet for frontend var å sette opp generelle komponenter som ville bli nyttig videre i utviklingen. Hovedmålet for backend var å få opp en god prosjektstruktur slik at videre utvikling skjer på en ryddig måte. Spesifikt hvilke arbeidsoppgaver knyttet til utviklingen ligger i vedlegget for product backlog, se vedlegg L

Det var også et mål å lage et estimat av planlagte aktiviteter for de kommende sprintene. Grunnen til at dette først er et mål i sprint 2, er at teamet vil ha en bedre bakgrunn til å kunne estimere arbeidet.

Avhengigheter og forutsetninger

Utviklingsteamet var avhengig av å ha tilgang og kommunikasjon mot NTNU sine databaser. For å kunne komme i gang med utviklingen og estimeringen var man avhengig av at kravspesifikasjonene var satt og godkjent av samtlige parter.

Resultat

Det ble utarbeidet en ER-modell og databasen ble opprettet og fungerte som den skulle. Utviklingsteamet kom også godt igang med både front- og backend. Det kom fram at det var hensiktsmessig å benytte seg av biblioteket 'Onsen' for å optimalisere brukergrensesnittet mot tablets. For frontend ble det satt opp flere generelle komponenter for bruk til videre utvikling. For backend ble det etablert en prosjektstruktur basert på MVC, Model, view, controller. Konfigurasjonen for kontinuerlig integrasjon ble påbegynt, men kom ikke helt i mål.

Sprint 3

Sprint 3 ble gjennomført i perioden 25.02.2019 - 01.03.2019

Mål

Hovedmålet for sprint 3 var å få på plass en bedre struktur for utviklingsprosessen. Dette inkluderte forbedring av product backlog i Trello, samt få en bedre struktur for dokumentasjonen. Det var også et mål å lage en ny low fidelity vertikal prototype for rollen TBM Inspector, basert på tilbakemeldingene på første prototype. For backend var målet å fullføre login funksjonalitet, samt automatisere testing. For frontend var målet å ferdigstille toolbar og lage komponenter, blant annet lister, som kan brukes gjentatte ganger gjennom systemet.

Avhengigheter og forutsetninger

For å kunne lage en ny prototype var teamet avhengig av god konstruktiv tilbakemelding fra oppdragsgiver på den første prototypen, med fokus på brukergrensesnittet og annen ønsket funksjonalitet.

Resultat

Sprinten startet med å forandre strukturen og oppsette på både product backlog og dokumentasjonen rundt. Det ble også laget en ny low fidelity vertikal prototype for rollen TBM Inspector. Login funksjonalitet, toolbar og listekomponenten ble ferdigstilt. Automatisering av tester ble ikke helt ferdigstilt.

Sprint 4

Sprint 4 ble gjennomført i perioden 14.03.2019 - 22.03.2019

Mål

Et av målene i sprint 4 var å få opp en minimal versjon av alle sidene tilknyttet en TBM Inspector. Dette skulle testes mot nettbrettet gruppen hadde fått tilgang til, Microsoft surface go. Deretter var det også et mål å få knyttet opp frontend mot backend og kunne liste ut hvilke TBM som lå i databasen. Klienten skulle også gjøres kjørbare ved en exe-fil. Autentiseringen skulle også ferdigstilles. For databasen skulle det legges inn testdata.

Avhengigheter og forutsetninger

Forutsetningene for denne sprinten var at gruppen måtte få tilgang til en tablet.

Resultat

I starten av perioden fikk gruppen klarsignal fra oppdragsgiver til å gå til innkjøp av en ønsket tablet til tidlig testing. Dette gjorde at disse målene ble nådd. Det eneste målet som ikke ble nådd for denne sprinten var koblingen mellom frontend og backend. Dette skyldes at det var mer hensiktsmessig å utvikle mer på frontend når man var inne i det enn å prøve å koble opp mot backend. Dette resulterte i at det ble produsert mer på frontend enn planlagt.

Sprint 5

Sprint 5 ble gjennomført i perioden 25.03.2019 - 05.04.2019

Mål

Hovedmålet for sprint 5 og 6 var å ha et operativt program for TBM Inspector. Det var ønskelig for teamet å ha denne delen av programmet ferdig før påsken, som kom etter sprint 6. Derfor var det naturlig å gi et overordnet mål for begge sprintene. Målet inkluderer at en bruker skal kunne logge seg inn, velge rett TBM, gjøre alle målinger og laste dette opp til databasen samtidig som det lagres lokalt. Det eneste som da skal mangle for denne rollen er finpuss på designet av sidene. Siste design av alle sidene er planlagt til sprint 7, som er siste sprint for selve utviklingen av programmet. For å gjøre dette var det også et mål om å lage en ny prototype som dekket alt som skulle være til rollen TBM Inspector, og utføre brukertester på programmet. Gruppen skulle også begynne på teorikapitlet. Målet var å bli ferdig med et første utkast av teorikapitlet ved slutten av neste sprint.

Avhengigheter og forutsetninger

Teamet var avhengig av en fungerende frontend og backend, slik at man kunne begynne med koblinger imellom. For teorien av det en avhengighet at teamet hadde fått diskutert teamatikken for kapitlet med veilederen.

Resultat

Ved utgangen av sprint 5 har teamet fått en kobling mellom frontend og backend. Dette betyr i praksis at det er satt opp til å få tilgang til alt fra databasen fra brukergrensesnittet. Dette vil si at teamet er i god rute til å nå sitt hovedmål om en fungerende TBM Inspector bruker etter sprint 6.

Teorikapitlets hovedfokus, nemlig digitalisering, ble ferdigstilt under denne sprinten. Dette betyr at gruppen er i god rute til å nå målet om å ha et første utkast før påske som kan sendes til veileder for en tilbakemelding. Det ble også laget en kjøreplan for brukertester og utført tester på seks individer. Resultatet av disse testene forsterket troen på at brukergrensesnittet av programmet er godt, samtidig som utviklingsteamet fikk noen nye gode ideer. Se vedlegget B for resultatet av brukbarhetstestene.

Sprint 6

Sprint 6 ble gjennomført i perioden 08.04.2019 - 12.04.2019

Mål

Sprint 6 deler samme mål som sprint 5. Dette inkluderer en fungerende TBM Inspector bruker og et utkast av teorikapitlet som skal sendes til veileder for tilbakemelding.

Avhengigheter og forutsetninger

For at gruppen skal få denne fungerende TBM Inspector brukeren er man avhengig av at forrige sprint gikk bra og man fikk satt opp denne koblingen mellom frontend og backend. Teamet er også forberedt til å sitte lengre dager denne sprinten for å ferdigstille dette.

Resultat

Utviklingsteamet ble ferdig med samtlige av målene tilknyttet sprint 6. Det ble også gjort mer enn planlagt. Teorikapitlet ble ferdigstilt som en fullstendig del og systemet fikk all funksjonalitet knyttet til en TBM Inspector.

Sprint 7

Sprint 7 ble gjennomført i perioden 23.04.2019 - 03.05.2019

Mål

Planen fra prosjektstart var å bli ferdig med produktet til denne sprinten. Her skal funksjonaliteten til brukere 'Project Supervisor', 'Super User' og 'Admin' lages. Deretter skal designet på hele systemet ferdigstilles. For hovedrapporten er målet å sende inn kapittel 4 og 5 til veileder for tilbakemelding.

Avhengigheter og forutsetninger

For at programmet skal kunne ferdigstilles er utviklingsteamet avhengig av at de tidligere sprintene har vært effektive. For å kunne sende veileder kapitlene er man avhengig av at disse kapitlene har vært i utvikling igjennom hele prosjektet.

Resultat

Programmet ble ferdig med de ønskede rollene. Dette betyr at hele systemet er operativt. Designet på applikasjonen ble også fikset på der det var nødvendig.

Når det gjelder hovedrapporten ble kapittel 4 og 5 ikke sendt inn til veileder.

Sprint 8

Sprint 8 ble gjennomført i perioden 06.05.2019 - 20.05.2019.

Mål

Målet for denne sprinten er å fullføre hovedrapporten og vedleggene som hører til. Det skal også lages en presentasjon av prosjektet og individuelle rapporter om hvordan det er gått. Applikasjonen skal nøye testes og små feil skal fikses.

Avhengigheter og forutsetninger

Her er utviklingsteamet avhengig av at planleggingen på starten av prosjektet, samt justeringene underveis har vært gode, slik at teamet har nok tid til å levere alt som skal leveres.

Resultat

Programmet ble gjennomgått og små feil ble justert. Dette besto kun i små skrivefeil. Alt av dokumentasjon ble ferdig innenfor de gitte tidsfristene og det ble laget en presentasjon og individuelle rapporter.

Det tilhører en rekke vedlegg som dokumenterer det som er snakket om over i dette delkapittelet. Disse vedleggene inkluderer:

- Sprint backlog til hver sprint, se vedlegg L.
- Timeliste med statusrapport, se vedlegg J.
- Retrospektivene til hver sprint, se vedlegg I.
- Trelloboard, se vedlegg L.

5 Diskusjon

Dette kapitlet er delt inn på samme måte som forrige kapittel. Her diskuteres resultatene som er beskrevet i forrige kapittel. Her blir også avvik fra planen og erfaringer beskrevet.

5.1 Drøfting av vitenskapelige resultater

Hovedproblemstilling for denne oppgaven er:

Kan innføring av et digitalt system øke beslutningsstøtten for en eksisterende arbeidsprosess i forbindelse med utskifting av boreverktøy?

Økt beslutningsstøtte måles ut i fra fire punkter som er beskrevet i kapittel 1.4. Programmet CutWear er det vitenskapelige resultatet, programmet forsøker å løse hovedproblemstillingen ved å øke beslutningsstøtten med utgangspunkt i disse punktene. Det første av disse punktene er tiden det tar fra inspektør måler kutterlitasje, til inspeksjonsdata er registrert i en felles database. Dette går betydelig raskere i CutWear enn den tidligere løsningen, når nettbrettet som CutWear kjører på har internett-tilgang vil applikasjonen kunne sende inspeksjonsdata direkte til databasen. Hvis nettbrettet ikke har internett vil inspeksjonsdata for denne inspeksjonen lagres lokalt frem til nettbrettet får internett-tilgang. Dette vil ikke forsinke prosessen i forhold til den eksisterende løsningen fordi her har brukerne ikke mulighet til å sende informasjonen på mail uten internett. I den tidligere løsningen måtte inspeksjonsdata behandles av flere ansatte, dette involverte å skrive data inn i et Excel ark for deretter å sende dette til en ny ansatt som flyttet data videre inn i systemet der de lagrer inspeksjonsdata. Denne prosessen rundt opplastingen av data er betraktelig mer tidkrevende enn løsningen utarbeidet i denne oppgaven.

Punkt to i definisjonen av økt beslutningsstøtte er hvordan fremvisning av lagret data vil hjelpe inspektørene med å ta de rette avgjørelsene under inspeksjonene. Her er det en tydelig forbedring da den eksisterende løsningen tilbyr mindre data, som i tillegg er organisert på en mindre oversiktlig måte. I CutWear vil man få fremvist tidligere registrert litasje og generell informasjon om posisjonen når man velger en posisjon under en inspeksjon. I fremtiden vil det kunne være hensiktsmessig å vise denne informasjonen grafisk til brukeren for å gjøre den lettere å bruke som beslutningsstøtte.

Punkt tre i definisjonen er hvordan applikasjonen hjelper inspektørene med å optimalisere arbeidsprosessen under inspeksjon av borhodet. Organiseringen av listen med posisjoner i en TBM og hvordan denne endrer seg, er der applikasjonen tilbyr funksjonalitet som er med på å løse dette. De ulike kodene en posisjon i listen kan ha vil hjelpe en inspektør med å holde oversikt over hvor langt man er kommet i inspeksjonen, samtidig som listen tilbyr hjelp til å se hvilke kuttere som trengs å byttes. Funksjonen som gir en inspektør muligheten til å markere at kutteren i en posisjon må byttes, kan være med på å optimalisere arbeidsprosessen betydelig. I listen med posisjoner vil inspektøren få en oversikt over hvilke posisjoner som har en kutter som må byttes. Inspektøren vil da kunne vente med å bytte alle kutterene til slutten av inspeksjonen. Å vente med å bytte alle kutterene til slutten av en inspeksjon vil ta mindre tid enn å bytte kutterene sporadisk i løpet av inspeksjonen, dermed bidrar dette til å optimalisere arbeidsprosessen.

Fjerde og siste punkt som definerer økt beslutningsstøtte er hvordan lagring av data vil hjelpe ledelsen med videre analyser og prognoser. Utviklingsteamet velger å se på TBM Supervisor som en del av ledelsen. For denne brukergruppen tilbyr CutWear en enkel og oversiktlig måte å se inspeksjonsdata på, noe som gjør det enklere å analysere fremgangen i prosjektet og utarbeide prognoser for videre fremgang. For viderutvikling av denne brukergruppen kan det være hensiktsmessig å produsere grafer basert på inspeksjonsdata for å gi brukerne en mer visuell presentasjon av informasjonen lagret.

Et problem med fremvisning av inspeksjonsdata er at dette ikke vil være de nyeste målingene hvis det er flere nettbrett på en TBM og det andre nettbrettet ikke har fått registrert data på grunn av manglende internett-tilgang. Utviklingsteamet har forsøkt å løse dette problemet ved å sette tidsstempel på inspeksjonsdata, på denne måten kan brukeren som leser data se når informasjonen ble registrert og ta dette i

betraktning.

Oppgaven adresserer også følgende problemstillingen:

Vil en digitalisert løsning kunne brukes under de vanskelige forholdene for selve registreringen av kutterslitasje?

Det er utfordrende å løse denne problemstillingen fordi CutWear ikke kan bli testet under disse forholdene i perioden prosjektet pågår. Grunnen til dette er at det ikke er tilgjengelige TBM'er innenfor rimelig avstand som det er mulig å teste applikasjonen på. På tross av manglende testmiljø, er applikasjonen utviklet for være så brukervennlig som mulig under de vanskelige forholdene inne i en TBM.

Designet av grensesnittet baserer seg på punktene som fremkommer i kapittel 2.2.2 og 2.2.3, som omhandler hvordan man skal oppnå høy brukbarhet. I tillegg er det laget til sammen tre prototyper som er testet på både oppdragsgivere og andre, se brukbarhetstesten B.

Videre er det designet et intuitivt brukergrensesnitt med store knapper som gir tydelige tilbakemeldinger til brukeren. På grunn av dette bidrar brukergrensesnittet til at applikasjonen er lett å bruke i det krevende miljøet i en TBM. Det er også gjennomført søk på hvilke nettbrett som er best egnet for krevende miljøer med mye varme, støv og fuktighet. De fleste nettbrett som virket passende kjører Windows 10 som operativsystem. Derfor er CutWear utviklet for å kjøre på dette operativsystemet, dette er også hensiktsmessig hvis de mer administrative brukergruppene som Super User og Admin ønsker å kjøre applikasjonen på en datamaskin med Windows.

5.2 Drøfting av ingeniørfaglige resultater

I tabell 1 er det listet opp hvilke forhåndsregler utviklingsteamet har tatt for å møte de åtte suksessfaktorene nevnt i samme tabell.

Siden oppgaven var uklar ved prosjektstart, ble det brukt betydelig mengde tid i starten på å opparbeide seg kunnskap om hva som forventes av produktet. Ved at partene kom frem til en felles forståelse av hva det ferdige systemet kunne bli, opparbeidet oppdragsgiver seg en bedre oversikt over hva de faktisk ønsket seg. Dette legger et godt grunnlag for videre utvikling av produktet og kan gi oppdragsgiver høyere kompetanse på hvordan man kan gjennomføre utviklings- og digitaliseringsprosjekter.

Det ferdige systemet viste seg å være for omfattende til å kunne ferdigstilles innenfor tidsrammene gitt for prosjektet. Hadde partene tidligere kommet frem til at omfanget av oppgaven var for stor til å fullføres innen tidsfristen kunne utviklingen av systemet i henhold til den endelige kravspesifikasjonen starte tidligere. Dette kunne ha resultert i en bedre brukeropplevelse og mer funksjonalitet som tilbyr beslutningsstøtte .

Det kan være flere grunner til dette. Oppdragsgiver selv var ikke klar over hva de spesifikt ønsket seg og utviklingsteamet hadde ikke nok erfaringer med å estimere større prosjekter. Dette gjorde at forventningene ble for høye med tanke på de gitte tidsrammene. Alt dette resulterte i at det kanskje ble brukt for mye tid i starten av prosjektet til planlegging av et større system.

Dette samsvarer med hva studiene om digitaliseringsprosjekter har kommet frem til. Kapittel 2.1.3 viser til hvordan begge partene må ha en høy forståelse av domenet og inneha høy teknisk kompetanse. Samtidig som domenet var noe uklart for utviklingsteamet fra starten av, hadde også ingen av partene god nok erfaring eller teknisk kompetanse til å umiddelbart se at prosjektetstørrelsen var for stor.

Tabell 2 viser hvordan prioriteringene har endret seg fra starten av prosjektet til slutten av prosjektet. Det er en rekke faktorer som muliggjorde disse endringene. Primært var det bruken av en agil utviklingsprosess som fremmet et tett samarbeid mellom partene og en god forståelse av domenet. Ut fra disse nesten ukentlige møtene var det en naturlig kontinuerlig endring gjennom hele prosjektet. Utviklingsteamet hadde også gjort grundig forarbeid som inkluderte en risikoanalyse og undersøkelser av hvordan man skal lykkes med et digitaliseringsprosjekt. Dermed var gruppen forberedt og motiverte for at det kunne komme endringer underveis.

Som nevnt tidligere var det to oppdragsgivere. The Robbins Company sin kontaktperson var utilgjengelig ved flere av sprintmøtene. Ser man på møtereferatet fra sprintreview 5 ser man at det kom viktig informasjon om hvordan housinger og kuttere fungerer sammen og et ønske om å endre noe funksjonalitet i applikasjonen basert på dette. Dette kom for sent ut i prosjektet til at det var mulig å implementere. Dette viser at partene kunne vært enda flinkere med å kommunisere seg imellom. Applikasjonen var funksjonell fremdeles, men kunne blitt ytterligere effektivisert om funksjonaliteten hadde blitt oppdatert på bakgrunn av den nye informasjonen.

De faktiske brukerne av dette systemet har vært utilgjengelig for utviklingsteamet under hele prosjektet. Dette har gjort det vanskelig å måle faktisk nytte av produktet. For å kompensere for dette har produktet aktivt blitt vist frem til kunden slik at de kunne komme med tilbakemeldinger, da det er disse som kjenner den eksisterende arbeidsprosessen best. Det er også gjennomført brukbarhetstester på personer uten kjennskap til prosjektet, se vedlegg B. Alle målene satt for programmet i visjonsdokumentet, se vedlegg E er oppnådd, med unntak av det ikke-funksjonelle kravet om testing. Stresstest ble nedprioritert fordi gruppen ikke hadde tilgang på server før sent ut i perioden og systemet vil sannsynligvis ikke ha stor nok brukerbase til at systemet får problemer på grunn av stor trafikk. Akseptansetest var ikke mulig å gjennomføre fordi prosjektet avsluttes før kunden har hatt tid til å teste produktet i en lengre periode. Det ble også kun utviklet enhetstester for serveren, dette er på grunn av at funksjonene i klienten ikke har et stort behov for å testes da de er en del av React komponenter som håndterer state og fremvisning av komponentene.

Resultatmålene i kapittel 4.2.2 viser at kravene som beskriver kjernefunksjonaliteten ved applikasjonen er prioritert. Dette inkluderer alle krav som må være på plass for å gjennomføre en inspeksjon. Revisjonene er et resultat av hvordan den agile utviklingsprosessen har økt interessentenes forståelse av domenet og det

endelige produktet.

Det var en rekke ulike grunner til at noen krav ble nedprioritert. Den vanligste grunnen til dette var at kravene var for store og tidkrevende til at de kunne realiseres. Andre krav var mindre viktige for funksjonaliteten i programmet, mens noen krav var vanskelige å gjennomføre med de forutsetningene utviklingsteamet hadde. Programmet er utviklet med hensyn på at det skal være skalerbart. Det er fordi det skal være så enkelt som mulig å implementere de resterende kravene senere.

I etterkant så utviklingsteamet at oppgaven kunne blitt løst på en annen måte. Utviklingsteamet brukte mye ressurser på å utvikle et system med tanke på at det skulle kunne gå direkte ut i produksjon. Underveis i prosessen kom det frem at oppdragsgiver ønsker å videreutvikle programmet før det blir tatt i bruk. Hvis dette er tilfellet, kunne det vært mer hensiktsmessig for utviklingsteamet å fokusere kun på å lage et helhetlig brukergrensesnitt med en enklere backend. Dette ville gitt oppdragsgiver mulighet til å få en enda bedre forståelse av hva de ønsket seg. Det ville også gitt en bedre definert oppgave for videreutvikling, som eventuelt kan være en fremtidig bacheloroppgave. I dette tilfellet hadde kommende utviklere hatt mindre tekniske restriksjoner og mindre å sette seg inn i, enn de hadde hatt ved videreutvikling av den nåværende løsningen.

Teamet har samarbeidet godt gjennom prosjektet hvor hver deltager fort fant sin rolle i gruppen, samtidig som kommunikasjonen var god. De ulike rollene grupped medlemmene tok i gruppen var med på å bidra til større fremgang. Dette er fordi teamet kunne jobbe parallelt samtidig som at det alltid var et grupped medlem som hadde mer kunnskap når det kom til de ulike delene av prosjektet. Hvis flere teammedlemmer skulle jobbe på en delen av systemet, kunne den med mest kunnskap hjelpe de andre. Som beskrevet i vedlegg J, oppnådde samtlige av prosjektdeltagerne kravet for arbeidstimer. Gjennom prosjektet jobbet gruppen sammen fra cirka kl 08:00 til kl 16:00 stort sett hver dag, med få unntak ved sykdom eller annet obligatorisk opplegg fra NTNU.

Utviklingsteamet kom fort til enighet om å bruke Scrum. Teamet hadde erfaring med Scrum fra tidligere prosjekter og var kjent med viktigheten av å benytte seg av en agil utviklingsprosess når det kom til utførelsen av digitaliseringsprosjekter. Det som var ulempen med Scrum i dette prosjektet var hvordan det opprinnelig er laget for team på flere enn tre personer. Det er derfor teamet valgte å lage en egendefinert utgave av Scrum, som var bedre tilpasset teamstørrelsen.

Utviklingsteamet har erfart at man må være nøye med å opprettholde regelmessig kontakt med oppdragsgiver, slik at man får den informasjonen og hjelpen man trenger. Dette kan man se spesielt i mangelen på kontakt med aktuelle testpersoner, som vil være brukere av det endelige produktet. Hvis teamet hadde blitt satt i kontakt med brukerne og eventuelt hatt muligheten til å teste applikasjonen i en TBM ville dette ført til gode tester og verdifull informasjon. Dette er noe oppdragsgiver kunne hjulpet teamet med.

5.3 Drøfting av administrative resultater

5.3.1 Timeregnskap

Timer per person ved endt prosjekt:

- **Mats Elias Davidsen:** 488,5 timer.
- **Kristian Nybakken Lund:** 490 timer.
- **Knut Wiig:** 488 timer.

Gruppen arbeidet jevnt gjennom hele prosjektet. Målet var at gruppen skulle møtes hver arbeidsdag og jobbe fra kl 08:00 til kl 16:00. Kravet for arbeidstimer ved prosjektet var mellom 450 og 550 timer per person. For mer detaljer, se vedlegg J.

5.3.2 Utviklingsprosess

I tilknytning til utviklingsprosessen ble det laget en plan for hver sprint, se kapittel 4.3. Den inkluderer mål, avhengigheter og forutsetninger for hver sprint. Se vedlegg L for faktiske oppnådde mål under hver sprint. Under ser man på erfaringer og avviket mellom planlagte mål og faktiske mål for hver sprint.

Sprint 1

Avvik

Det største avviket i denne sprinten var en fullstendig etablering av de tekniske spesifikasjonene. Hovedårsaken til avviket var stadig utsettelse av et møte med det tekniske teamet til The Robbins Company, som befant seg i Seattle, USA. Dette gjorde at valget av nettbrett også ble forskjøvet. I starten av sprint 2 kom klarsignalet om type nettbrett, som ble Windows-basert. Da kunne de siste tekniske spesifikasjonene tas. Denne forsinkelsen hadde liten påvirkning på videre arbeid, da det var gjort detaljerte undersøkelser av de mulige løsningene. De resterende avhengighetene og forutsetningene lå til rette for å nå de estimerte målene.

Erfaring

Teamet erfarte at det kunne være vanskelig å få ferdigstilt en fullstendig kravspesifikasjon for oppgaven. Den største utfordringen var å få en god nok forståelse av alt som skulle gjøres. Siden det var to oppdragsgivere for dette arbeidet, hvor begge hadde sine preferanser ble det mye frem og tilbake. Begge oppdragsgiverne hadde travle hverdager og det ble ikke lett å få alle til å møtes samtidig. Som en følge av denne sprinten ble det satt opp en passende kommunikasjonskanal, hvor Helge- Ivar Frostad var et sentralt bindeledd mellom utviklerne og oppdragsgiverne.

Sprint 2

Avvik

Det eneste avviket for sprint 2 var at ikke samtlige arbeidsoppgaver knyttet til kontinuerlig integrasjon kom på plass. Dette fordi det var vanskelig på forhånd å estimere tiden for dette, og det viste seg og være mer jobb enn først antatt.

Erfaring

Siden sprint 1 gikk ut på å ferdigstille en oppgavebeskrivelse og kravspesifikasjon fikk gruppen ikke kommet ordentlig i gang med å planlegge utviklingsprosess. Dette medførte at det først under sprint 2 ble etablert en struktur for prosessen. Det er derfor først nå retrospectiv ble gjennomført. Av erfaringer fra sprint 1 kom det frem at det må legges ned mer tid til selve planleggingen av arbeidsoppgaver for sprinten. På grunn av at dette var den første sprinten for utvikling, ble det en suboptimal planlegging i forhold til føringer i Trello for backlog.

Sprint 3

Avvik

Avviket ved denne sprinten var at man ikke ble helt ferdig med å automatisere testingen. Dette på grunn av at det var mer å skrive enn først antatt. I etterkant har man erfart at lappene skrevet til denne sprinten var for generelle, dette er også noe av grunnen til at teamet ikke kom i mål med dette punktet. Det burde ha vært brutt noe mer ned.

Erfaring

Under sprint 3 ble det etablert en god struktur for utføring og dokumentasjon av utviklingsprosessen basert på erfaringene fra de forrige sprintene. Blant annet innebærer dette:

- En bedre planleggingsfase. Her ble det gjort endringer på oppføring av arbeidsoppgaver på Trello. Det ble laget en generell product backlog med utgangspunkt i user stories. Deretter brytes de som skal utvikles under denne sprinten igjen ned i mindre arbeidsoppgaver og legges i sprint backlog.
- En strukturert utviklingsprosess. Det ble utformet en struktur for utviklingsprosessen for å sikre at det ble gjort en grundig planlegging, dokumentasjon og retrospective for hver sprint fremover.
- Et bedre retrospective. Det kom frem at teamet måtte gjennomgå retrospectiven på en bedre måte. Dette fordi retrospectiven skulle bli grunnlaget for videre dokumentasjon av utviklingsprosessen. For forrige sprint var en mindre detaljert retrospective ikke et stort problem da man var helt i startfasen av utviklingen, men teamet så fremover og innså at retrospectiven måtte gjennomføres bedre.

Sprint 4

Avvik

Det største avviket fra planen under sprint 4 var hvilke arbeidsoppgaver som ble gjort. Det var planlagt å knytte frontend og backend sammen for å få ut en liste med TBM som lå i databasen. Dette ble forskjøvet fordi gruppen kom frem til at det var mer hensiktsmessig å utvikle mer på frontend først. Dette innebar en mer fullstendig fremstilling av hver side enn først planlagt.

Erfaring

Sprint 4 var en kort, men veldig effektiv sprint for gruppen. Det ble gjort mer enn planlagt og kommunikasjonen var god. En erfaring som gruppen tar videre var hvor viktig det er å hele tiden teste på selve nettbrettet som skal brukes. Man kunne simulere et nettbrett på datamaskinen, men det ble ikke det samme som å benytte seg av et ekte nettbrett.

Sprint 5

Avvik

Det var ingen store avvik denne sprinten.

Erfaring

Kontaktpersonen for The Robbins Company har en travel hverdag og kunne ikke møte opp på hvert sprintreview-møte. Kontaktpersonen stilte opp på møte for sprint 5, men det var da lenge siden forrige oppmøte. Dette gjorde at kontaktpersonen først nå kom med viktige detaljer for prosessen, slik at gruppen måtte gjøre endringer. Her erfarte gruppen at det fort kan komme nye endringer som gjør at tidligere planlegging og arbeid må endres. Dette er en god erfaring å ha med seg inn i arbeidslivet.

Sprint 6

Avvik

Alle planlagte funksjonaliteter for en TBM Inspector ble ferdigstilt. Eneste avviket for denne sprinten var når oppdragsgiver kom med helt ny informasjon om prosessene, noe som gjorde at deler av systemet måtte gjøres om. Men her ble utviklingsteamet enig om at denne informasjonen kom for sent til at de kan legges inn. Dette var ikke veldig viktig da systemet fungerte slik det allerede var, samtidig som det er for liten tid til å legge dette inn.

Erfaring

Den største erfaringen under denne sprinten var hvordan det kan være vanskelig å få full oversikt over alt som skal være med og hvordan prosessene fungerer når det er flere interessenter involvert. Oppdragsgiveren The Robbins Company har ikke anledning til å møte på alle sprint reviews, og dette gir utfordringer knyttet til utvikling av deres system. Essensiell informasjon om kutterposisjoner og kutterbytte kom først under

denne sprinten. Heldigvis var dette ikke informasjon som gjorde det allerede påbegynte systemet ubrukelig, men det kunne optimaliseres.

Sprint 7

Avvik

Avviket for denne sprinten var at et utkast av kapitlene 4 og 5 av hovedrapporten ikke ble sendt inn til veileder. Dette kom av flere grunner. Veileder var syk en periode og hadde ikke anledning til å rette samtidig som utviklingsteamet ikke følte det var nok innhold i kapitlene til å sende inn.

Erfaring

Programmet fungerer godt og all planlagt funksjonalitet som skulle være er med, men det er alltid noe som kan forbedres. Utviklingsteamet har erfart at ved et punkt må man si seg fornøyd på applikasjonen og legge det fra seg.

Sprint 8

Avvik

Det var ingen avvik under denne sprinten. Alt ble fullført innen tidsfristen. Dette innebærer hovedrapporten, annet dokumentasjon og selve programmet.

Erfaring

Under den siste sprinten har utviklingsteamet erfart hvor viktig det er med en overordnet oversikt over alt som skal leveres. Hadde utviklingsteamet ikke planlagt denne sprinten godt, ville teamet fort glemt noe av det som skal inn.

Det er også denne perioden gruppen som en helhet hadde høyeste kompetansebygging på prosessmålet sitt som omhandlet skiving av litterære og vitenskaplige dokumenter.

5.4 Ethiske problemstillinger

Det finnes en bred skala av applikasjoner, alt fra store internasjonale systemer til små interne systemer. Graden av etiske problemstillinger som følger varierer også stort. De som skal utvikle store systemer med store kundedatabaser, som da inneholder sensitiv informasjon, har et mye større ansvar enn de som utvikler små private systemer. De må ha stort fokus på sine profesjonsetiske spørsmål, og se på konsekvensene i et samfunnsmessig-, økonomisk- og miljømessig sammenheng.

5.4.1 Profesjonsetiske spørsmål

Applikasjonen CutWear omfatter kun en relativt liten kundegruppe, hvor det ikke lagres sensitiv informasjon om brukere. Men applikasjonen er laget for aksjeselskapet The Robbins Company, som ikke ønsker å dele sine kvalitetsdata med konkurrentene. Utviklingsteamet må dermed sørge for å beskytte disse dataene på en god måte.

5.4.2 Økonomiske konsekvenser

Det er ikke regnet nøyaktige tall på hva The Robbins Company kan tjene på å få økt kvaliteten og mengden av data, men det er enorme summer i denne bransjen, så potensialet er massivt. Applikasjonen skal skape gevinster ved en rekke ulike poster. Tabellen under viser dette.

Økonomiske konsekvenser ved CutWear	
Poster	Beskrivelse
Bedre forståelse av kuttere	CutWear vil gi en bedre forståelse av hvilke kuttere og kuttereregenskaper som egner seg for de ulike bergartene. Dette ved å se på data om kutterslitasje og bytte og koble dette opp mot kutter. CutWear vil gi en enorm mengde kvalitetsdata over tid.
Bespare tid ved inspeksjoner.	Applikasjonen skal gjøre inspeksjoner hurtigere. Dette skjer primært ved at organiseringen av inspeksjonene skjer på en mer ryddig måte, slik at inspektørene kan gjøre jobben sin mer effektivt.
Beslutningsstøtte for prosjektleder for boreprosjektet	Skal hjelpe prosjektleder med å styre prosjektet sitt. Dette kan føre til mer effektive inspeksjoner ved at de blir monitorert og prosjektleder kan finne ut hvilke metoder som er effektive og lære denne bort til de andre.
Kvalitetsdata til NTNU	NTNU kan bruke disse data til sine egne analyser og prognoser. Dette vil styrke NTNU's stilling innenfor denne bransjen. Denne delingen av data vil også kunne knytte sterkere bånd ut til bransjen og da særlig til The Robbins Company.
Kvalitetsdata til The Robbins Company.	Bedriften kan bruke disse dataen internt til å effektivisere sine metoder, og selge kunnskapene de opparbeider seg videre til sine kunder. Dette kan være i form av kurs om effektivisering av inspeksjoner.
Feilaktig beslutningsstøtte.	Feilaktig beslutningsstøtte i applikasjonen kan føre til at brukerne av systemet tar beslutninger som fører til uventede skader på TBM.

Tabell 4: Økonomiske konsekvenser ved CutWear

5.4.3 Miljømessige konsekvenser

De miljømessige konsekvensene av dette prosjektet varierer med hvilke aspekter man ser på. Papirforbruket vil gå ned, men det var et uvesentlig lite forbruk fra før av, så det vil ikke gjøre store forandringer for miljø. Hvis applikasjonen produsert i dette prosjektet gir økt beslutningsstøtte kan dette være med på å effektivisere tunnelboringen. En mer effektiv tunnelboring vil mest sannsynlig føre til mindre utslipp fordi det tar mindre tid å grave tunnelene. Dermed vil TBM'ene kjøre i kortere tidsperioder og produsere mindre utslipp, dette vil ha positive konsekvenser for miljøet. Dermed vil prosjektets miljømessige konsekvenser i stor grad avhenge av om problemstillingen blir løst.

6 Konklusjon og videre arbeid

Dette kapittelet besvarer problemstillingene definert i kapittel 1 på bakgrunn av resultatet og diskusjonen. Siste del av kapittelet beskriver hvordan det er lagt til rette for videre arbeid, samt forslag til ny funksjonalitet.

6.1 Konklusjon

Utviklingsteamet kom frem til at digitaliseringsprosjektet i stor grad kan anses som vellykket. De mest sentrale grunnene til dette var at gruppen planla godt og hadde regelmessige møter med oppdragsgiver. Dette førte til at gruppen utviklet et produkt som oppdragsgiver var fornøyd med, innenfor prosjektets begrensninger.

Når det gjelder kravene i visjonsdokumentet, se vedlegg E, er den endelige revisjonen av kravspesifikasjonen noe endret. Denne revisjonen har fjernet en brukergruppe og nedprioritert den minst nødvendige funksjonaliteten på de resterende brukergruppene. Denne brukergruppen og mye av funksjonaliteten som ikke er inkludert i den endelige revisjonen er nevnt under videre arbeid. Punktene i den reviderte kravspesifikasjonen er i stor grad realisert, dermed vil også store deler av kravene satt i visjonsdokumentet være nådd. Derav konkluderes det med at CutWear i stor grad oppfyller kravene satt i visjonsdokumentet.

Hovedproblemstillingen var som følger:

Kan innføring av et digitalt system øke beslutningsstøtten for en eksisterende arbeidsprosess i forbindelse med utskifting av boreverktøy?

Ut i fra resultatet og diskusjonen ser man hvilken funksjonalitet applikasjonen har for å øke beslutningsstøtten for de ulike rollene. Det som er viktig å påpeke ved applikasjonen er hvordan den også vil bli brukt i omgivelser med begrenset internett-tilgang. CutWear er laget med utgangspunkt i at applikasjonen som regel har internett-tilgang og at det svært sjeldent vil gå mer enn et par døgn uten. Dette er basert på at oppdragsgiver har informert om at det er tilfeller hvor det ikke er internett-tilgang inne i TBM'ene. Dette har utviklingsgruppen tatt hensyn til ved utviklingen. Applikasjonen er avhengig av å ha internett-tilgang for å kunne tilby brukeren størst mulig beslutningsstøtte. I tilfeller der applikasjonen ikke har internett-tilgang vil en TBM Inspector fremdeles få økt beslutningsstøtte, men denne vil være suboptimal fordi tidligere inspeksjonsdata vil være utilgjengelig. Siden applikasjonen som regel har tilgang til internett, enten i TBM eller i umiddelbar nærhet vil systemet øke beslutningsstøtten. Derav konkluderes det med at CutWear i stor grad løser hovedproblemstillingen, samtidig som applikasjonen bidrar til digitalisering av tunnelbransjen.

Sekundære problemstilling var som følger:

Vil en digitalisert løsning kunne brukes under de vanskelige forholdene for selve registreringen av kutterslitasje?

Ved prosjektstart var det planer om at oppdragsgiver skulle sørge for at utviklingsteamet skulle få besøke en TBM. På bakgrunn av dette ble denne sekundære problemstillingen laget. Her skulle det gjennomføres realistiske tester under de vanskelige forholdene. Som beskrevet under resultat og diskusjon ble disse planene ikke realisert. Derfor er det ikke mulig å komme frem til en endelig konklusjon av denne problemstillingen. På tross av dette har utviklingsteamet opparbeidet seg tilstrekkelig informasjon om de vanskelige forholdene, slik at det til en viss grad er mulig å besvare problemstillingen. Det er tenkt at applikasjonen vil kjøre på nettbrett som tåler de fysiske forholdene inne i borhode. Brukergrensesnittet er også designet med tanke på at man skal kunne bruke verneutstyr som hansker. Størrelsen på nettbrettet er ikke et problem da dagens løsning baserer seg på å bruke A4 ark, som er av tilnærmet lik størrelse. Nettbrettet kan også benytte seg av tilleggsutstyr for å gjøre håndteringen enklere, som for eksempel arm-,kne- og skulderrem. Alt tilsier at applikasjonen skal kunne bli brukt under de krevende forholdene i en TBM uten store problemer.

Når det skal konkluderes er det viktig å ta i betraktning at oppdragsgiver var med på å formulere problemstillingen, spesielt med tanke på hva som definerer økt beslutningsstøtte. Når oppgaven løser problemstillingen i den grad beskrevet over, resulterte dette i at programmet stod til oppdragsgivers forventninger.

Her kommer det noen ord fra oppdragsgiverne som beskriver deres synspunkt på samarbeidet, applikasjonen og prosjektet som en helhet:

Som oppdragsgiver er vi godt fornøyd med prosessen og resultatet av CutWear-oppgaven. Alle møter vi har hatt med gruppen har vært produktive og nyttige for oss og sluttproduktet. Fra vårt ståsted ser det ut som gruppen har løst oppgaven på en god måte med hensyn til planlegging, oppgavefordeling og kommunikasjon med oss
Amund Bruland, Bygg- og miljøteknikk NTNU, se vedlegg D1.

6.2 Videre arbeid

Utviklingsteamet har lagt til rette for videreutvikling av produktet underveis i prosjektperioden. Et av områdene der det i størst grad er lagt til rette for videreutvikling er i databasen, denne er designet for hele domenet som først ble planlagt. På grunn av dette er databasen også strukturert slik at den skal kunne håndtere nødvendig data for brukergruppene som ikke er med i den reviderte kravspesifikasjonen.

Under arbeidet med programmet har utviklingsteamet kommet frem til en rekke punkter som kan være aktuelle for videreutvikling, disse omfatter både eksisterende og planlagte brukergrupper. Disse punktene for videreutvikling av CutWear ligger i vedlegg K.

Referanser

- [1] «Kognitiv psykologi», side: https://snl.no/kognitiv_psykologi.
- [2] «Empiri», side: <https://snl.no/empiri>.
- [3] «Antropologi», side: <https://snl.no/antropologi>.
- [4] «Sosiologi», side: <https://snl.no/sosiologi>.
- [5] «Industriell design», side: <https://www.ntnu.no/studier/mtdesig/om>.
- [6] P. Parviainen, M. Tihinen, J. Kaariainen og S. Teppola, «Tackling the digitalization challenge: how to benefit from digitalization in practice. International journal of information Systems and Project Management», s. 64, 2017. side: <http://www.sciencesphere.org/ijispm/archive/ijispm-050104.pdf>.
- [7] S. Brennen og D. Kreiss, «Digitalization and Digitization», side: <http://culturedigitally.org/2014/09/digitalization-and-digitization/>.
- [8] R. G. Fichman, B. L. Dos Santos og Z. Zheng, «Digital innovation as a fundamental and powerful concept in the information Systems curriculum.», årg. 38, nr. 2, s. 333, 2014.
- [9] «Manufacturing-as-a-Service? Amazon Puts Fast Fashion In The Crosshairs With New Patent», side: <https://www.cbinsights.com/research/amazon-fashion-apparel-manufacturing-patent/>.
- [10] D. Tilson, K. Lyytinen og C. Sørensen, «Research commentary-digital infrastructures: the missing IS research agenda. Information systems research», årg. 21, nr. 4, s. 479, 2010. side: <https://www.uio.no/studier/emner/matnat/ifi/INF5210/h14/pensumliste/articles/tilson-et-al-2010.pdf>.
- [11] P. Nilsen, «Digitalisering: En studie av utfordring og muligheter sett fra fire utvalgte bransjer, master thesis.», s. 5–7, 2017.
- [12] M. Jørgensen, «Suksess og fiasko i offentlige IKTprosjekter: En oppsummering av forskningsbasert kunnskap og evidensbaserte tiltak», side: https://www.regjeringen.no/contentassets/9018344feae44c1f9a2a114e768ebd1b/suksess_fiasko_offentlige_ikt-prosjekter.pdf.
- [13] «Det er også en risiko å ikke igangsette digitaliseringsprosjekter», side: <https://www.digi.no/artikler/det-er-ogsaa-en-risiko-a-ikke-igangsette-digitaliseringsprosjekter/197716>.
- [14] «Kveldsseminar: Nyttestyring og gevinstrealisering; Teori og Praksis. PS: Denne url'en er ikke lenger tilgjengelig, da hitledelsen har anskaffet seg en ny nettside.», side: <http://hitledelse.com/arrangementer/tidligere-avholdte-arrangementer/>.
- [15] A. Neumeier, T. Wolf og S. Oesterle, «The Manifold Fruits of Digitalization- Determining the Literal Value Behind», s. 484–498, 2017. side: <https://www.wi2017.ch/images/wi2017-0164.pdf>.
- [16] P. Kotler og K. L. Keller, «Marketing Management», s. 166, 2012. side: http://socioline.ru/files/5/283/kotler_keller_-_marketing_management_14th_edition.pdf.
- [17] H. Y. Torvatn, B. Kløve og K. T. Andersen, «Teknologiutbredelse og stress i norsk arbeidsliv.», årg. 1, s. 3, 2016. side: <https://www.sintef.no/globalassets/sintef-teknologi-og-samfunn/rapporter-sintef-ts/sintef-rapport-teknologiutbredelse-og-stress-i-norsk-arbeidsliv.pdf>.
- [18] H. Y. Torvatn, B. Kløve og A. D. Landmark, «Ansattes syn på digitalisering. En nasjonal kartlegging av digitale forhold som skaper stress og opplevd produktivitet.», årg. 1, s. 48, 2017. side: <https://sintef.brage.unit.no/sintef-xmlui/bitstream/handle/11250/2490540/Rapportnr%2b2017%2b00681%2bprint%2bversjon.pdf?sequence=1&isAllowed=y>.
- [19] *Kunnskap opparbeidet gjennom faget IBED3002 Prosjektledelse og forretningssystemer, anlagt ved NTNU, Trondheim.*
- [20] K. Haga og L. L. Skjelbek, «Brukersentrert Utvikling for Microsoft Surface, master thesis», s. 15–23, 2011.
- [21] J. Nielsen, *Usability Engineering*. Academic Press, Inc., 1993, s. 26, ISBN: 0-12-518406-9.
- [22] «Ergonomics of human-system interaction — Part 210: Human-centred design for interactive systems», side: <https://www.iso.org/obp/ui/#iso:std:iso:9241:-210:ed-1:v1:en>.

- [23] J. Preece, Y. Rogers og H. Sharp, «Interaction Design: Beyond Human Computer Interaction», s. 279–313, 2002. side: <https://arl.human.cornell.edu/879Readings/Interaction%20Design%20-%20Beyond%20Human-Computer%20Interaction.pdf>.
- [24] *Kunnskap opparbeidet gjennom faget TDAT3019 Systemutvikling 3, anlagt ved NTNU, Trondheim.*
- [25] «Scrum», side: <https://www.scrum.org/resources/what-is-scrum>.
- [26] Schwaber, «Agile Project Management with Scrum», 2004.
- [27] «Native vs. Non-Native Apps», side: <https://www.multimedianaordic.eu/blog/react-native-apps/>.
- [28] «Frameworks, programming languages: Coding from scratch vs. using a framework», side: <https://jaxenter.com/coding-scratch-vs-using-framework-134319.html>.
- [29] . side: <https://opensource.com/resources/what-docker>.
- [30] *What is Continuous Integration?* Side: <https://codeship.com/continuous-integration-essentials>.
- [31] «Operating System Market Share Worldwide», side: <http://gs.statcounter.com/os-market-share>.
- [32] «A complete guide to packaging your Electron app», side: <https://medium.com/how-to-electron/a-complete-guide-to-packaging-your-electron-app-1bdc717d739f>.
- [33] «Hva er Scrum», Beskrivelsen er tatt fra underkapittel 'The Roles of the Scrum Team'. side: <https://www.scrum.org/resources/what-is-scrum>.
- [34] G. Seland og O. A. Alsos, «Papirprototyper og brukbarhetstesting»,

7 Vedlegg

A Møter

A1 Møteinnkalling Bacheloroppgave 29, Oppstartsmøte m/ Alle

Tidspunkt/sted: Torsdag 18.01.2019 kl 12:30 – 14.30, Grupperom 2-048 Institutt for bygg og miljøteknikk 2 etg.

Følgende personer innkalles:

Kristian Nybakken Lund
Knut Wiig
Mats Davidsen
Amund Bruland
Helge Hafting (veileder)
Sindre Log
Helge-Ivar Frostad
Pål Drevland Jakobsen

Agenda

- Sak nr 01/2019 Gjennomgang av praktiske detaljer.
- Sak nr 02/2019 Utviklingsprosess – Scrum.
- Sak nr 03/2019 Gjennomgang av prosjekthåndboka.
- Sak nr 04/2019 Hvordan skal prosjektet dokumenteres.
- Sak nr 05/2019 Avtale faste møtetidspunkter og gå gjennom foreløpig plan.
- Sak nr 06/2019 Finne tidspunkt for møte angående oppgavebeskrivelse.
- Sak nr 07/2019 Eventuelt

Møte planlegges avsluttet ca kl 14.30

Ta kontakt med undertegnede dersom du ikke har anledning til å komme

Med vennlig hilsen

Mats Davidsen

Trondheim 16.01.2019

A1 Referat fra prosjektmøte bacheloroppgave 29

Dato og tid: Torsdag 18.01.2019 kl 12:30 – 14.30

Sted: Grupperom 2-048 Institutt for bygg og miljøteknikk 2 etg

Til stede:

Mats Davidsen
Knut Wiig
Kristian Nybakken Lund
Helge Hafting (veileder)
Amund Bruland
Sindre Log
Helge-Ivar Frostad
Pål Drevland Jakobsen

Frafall: Ingen

Sak nr 01/2019

- Ambisjonsnivå: A
- Prosjektframdriften er studentenes ansvar
- Valg av språk til oppgaven: Engelsk på alt som skal til Robins, norsk på det som skal til NTNU
- Annen praktisk informasjon sender Helge Hafting via mail.

Sak nr 02/2019

- Scrum, her har man Sprinter med sprintreviews hvor man møtes og går gjennom det vi har hittil. Datoer for møter ligger i fremgangsplanen (se vedlegg). Disse møtene holder det at minst 1 fra oppdragsgiver møter.
- Møter med Helge Hafting blir samme dag som Sprintreviews evt noen dager før/etter. Her går vi gjennom prosessen.
- Hver backlog-produkt blir rangert fra 1-10, hvor delprodukter med lav rangering ikke er viktig å utføre, og vil ikke trekke ned karakter om det ikke blir gjort.
- DDD, Domain Driven design, Dette vil si at man finner ut av produktbeskrivelsen sammen med domeneekspertene (Oppdragsgiver i dette tilfellet).
- TDD, Test driven development, Dette er mest for utviklingsteamet, for å sikre at funksjonaliteten blir god.

Sak nr 03/2019

- Timeliste skal føres for all arbeid, dette inkluderer møter, ute hos kunden etc.
- Ukesrapport skal inneholde timelister hvor man skriver et par setninger om hva studenten har gjort denne uken.
- Holde budsjettet, 501.5t (+-10 prosent)
- Karaktervurdering går på:
 - Selve produktet
 - Timelister, dokumentasjon (brukermanual, driftsdokumentasjon, kodedokumentasjon, etc)

- Sluttrapport/vitenskapsdelen
- Presentasjon
- Dokumentasjon kan sendes til Helge for gjennomgang før endelig innlevering.
- Levering til Amund Bruland minst et døgn før siste levering(20. mai) slik at de kan skrive ut sluttrapport.
- Ha med et kapittel i rapporten som heter «videre arbeid» slik at det blir lett for Robbins Company å videreutvikle produktet.

Sak nr 04/2019

- SCRUM, Sekvensdiagram og user stories som en del av kravspesifikasjon
- Visjonsdokument
- Arbeidskontrakt
- Brukermanual
- Product backlog
- Dokumentasjon av kode
- Testing -i TDD
- Hovedrapport

Sak nr 05/2019

- Her har vi laget et fremgangsplan hvor datoer for møter er lagt inn, her er det rom for forandringer om det må til i fremtiden, men for nå er dette planen. Dette dokumentet finner man under vedlegg.
- 15. mars og utover, se på en boremaskin, her vil Sindre finne en passe dato senere.

Sak nr 06/2019

- Fredag 25.01, ettermiddagen, gå gjennom produktbeskrivelse.

Ingen punkter under eventuelt

18.01.2019 Knut Wiig

A1 Møteinnkalling Bacheloroppgave 29 m/ Oppdragsgiver

Tidspunkt/sted: Mandag 28.01.2019 kl 13:00 – 15.00, Grupperom 2-048 Institutt for bygg og miljøteknikk 2 etg.

Følgende personer innkalles:

Kristian Nybakken Lund
Knut Wiig
Mats Davidsen
Amund Bruland
Sindre Log
Helge-Ivar Frostad

Agenda

Sak nr 01/2019 Backlog
Sak nr 02/2019 Eventuelt

Møte planlegges avsluttet senest kl 15.00

Ta kontakt med undertegnede dersom du ikke har anledning til å komme

Med vennlig hilsen

Mats Davidsen

Trondheim 22.01.2019

A1 Referat fra prosjektmøte bacheloroppgave 29

Dato og tid: Mandag 28.01.2019 kl 12:30 – 14.30

Sted: Grupperom 2-048 Institutt for bygg og miljøteknikk 2 etg

Til stede:

Mats Davidsen
Knut Wiig
Kristian Nybakken Lund
Helge Hafting (veileder)
Amund Bruland
Sindre Log
Helge-Ivar Frostad

Frafall:

Pål Drevland Jakobsen

Sak nr 01/2019

- Ringen på kutter byttes hver gang, andre komponenter byttes ved behov - dette sjekkes hver gang man bytter ring.
- Hub byttes på grunn av slitasje, aksling gjør også dette, men byttes sjeldnere
- Ikke gå i detalj på dataen som skal lagres
- Registrer når man bytter lager så man vet hvor mange bytter av ring man har hatt siden sist gang man byttet lager.
- Viktig å inkludere Nitin i utviklingsprosessen
- Data som er viktigst å hente ut: Chainage og timestamp
- Bruk separat database for at NTNU skal ha tilgang, fordi det vil være vanskelig å gi NTNU tilgang på datalogger fra Robbins.
- Vi sender altså data til en database fra applikasjonen
- Kategorier på kutter bytte fungerer godt allerede, så bruk dette videre i applikasjonen
- Vil ha muligheten til å gå inn på hver kutter-ID og se historien på hvor denne kutteren har vært.
- Apparatet som brukes angir total slitasje på kutter, IKKE hvor mye som er igjen
- Tallene på slitasje som går nedover i excel-arket vi har fått er grunnet feilaktig føring gjort av operatør
- Ønsker en "Annet" felt ved kutter bytte slik at man kan fylle ut dette feltet når grunnen til bytte ikke faller innenfor noen av de forhåndsbestemte kategoriene
- Legg opp til at det er muligheter for alternative registreringer av kutter-ID i fremtiden
- Få opp cutter changing guidelines for hver posisjon og tegning av kutterhodet i applikasjonen, kan legges inn som pdf
- La supervisor få se alle TBM'er i aktuelt prosjekt
- Vil ha mulighet til å sette opp TBM's som både supervisor og admin
- Ønsker at man skal kunne se hvor kutteren er ved hjelp av kutter ID, Supervisor skal kunne få se hvor kutteren er. Foreløpig er det tilstrekkelig å vite om den er på kutter-hodet eller ikke.

- Admin kan registrere en ny type TBM, supervisor velger så type TBM som brukes til sitt prosjekt ved oppstart
- Passord er ikke så viktig for operatører, kan i utgangspunktet være en 4-sifret kode eller lignende
- Admin skal kunne korrigere/slette data
- Vil gjerne ha: Rulledistanse per omdreining på posisjon =j, kan få til en forecast på når en kutter bør byttes basert på data samlet nylig. Regner ut slitasje av kuttere basert på rullingen
- Kan gjøres lett ved enkel matematikk
- Forskjellige typer kuttere har forskjellige maksverdier på slitasje, har disse konkrete dataene fra før. Kan regne ut hvor mye som er igjen til man må bytte fra dette.
- Hvis industri tablets er vanskelig å få til kan vi skaffe en annen tablet som også kjører Windows i mellomtiden
- Borrhodet er delt inn i deler (trekanter ut fra midten)
- Flere kritiske faktorer som spiller inn når en kutter blir rød, vis hvilken som gjelder her
- Spesifiser at Operatøren er på TBM
- Folk på site:
 - Sjefsoperatør - bestemmer ofte hvilke kuttere
 - Tunnelformann - Styrer inne i tunnelen
 - Prosjektleder
 - Legg inn flere brukergrupper ved videreutvikling
- Skriv inn HUB og lager istedenfor Housing, lager er det som byttes hver 3 gang
- Må lagre etter registrering av hver kutter ved inspeksjon
- Skal legge til en kommentar ved kutter bytte i TILLEGG til kategori
- Vurder å lage en bruker for verksted, som registrerer hva som er gjort på kutteren ved verkstedet
- Bruk til å lage utfyllende historie om kutteren
- Inspiserer kuttere minst hver 12-timer den dag i dag
- illustrer hvor mange byttter man har hatt for hver posisjon, med grunnen til bytte på hver av disse (se side 24 i dokument fra Amund). Dette kan evt være hovedsiden for supervisor
- Bytt til “relevant information”
- Ta også med klokketiden fra maskinen når man registrerer bytte
- Ta med crew-number når man starter en inspeksjon av kutter hodet, men dette er også bestemt ved en tidsplan
- Denne applikasjonen har kvalitet i og med at den samler inn store mengder data, samtidig som å kvalitetssikre denne dataen

Sak nr 02/2019

Ingen punkter under eventuelt

28.01.2019 Knut Wiig

A2 Møteinnkalling Bacheloroppgave 29 Sprint Review 1 m/ Veileder

Tidspunkt/sted: Fredag 08.02.2019 kl 09:00 – 10:00, Grupperom 304 ,Akrinn.

Følgende personer innkalles:

Kristian Nybakken Lund
Knut Wiig
Mats Davidsen
Helge Hafting (veileder)

Agenda

Sak nr 01/2019 Gjennomgang av user stories.
Sak nr 02/2019 Se på arbeidskontrakt.

Møte planlegges avsluttet ca kl 10.00

Ta kontakt med undertegnede dersom du ikke har anledning til å komme

Med vennlig hilsen

Mats Davidsen

Trondheim 16.01.2019

A2 Referat fra prosjektmøte bacheloroppgave, sprint review 1 m/ Veileder

Dato og tid: Fredag 08.02.2019 kl 09:00 – 09:30

Sted: Grupperom G304 Akkrinn

Til stede:

Mats Davidsen

Knut Wiig

Kristian Nybakken Lund

Helge Hafting (veileder)

Frafall: ingen.

Sak nr 01/2019

- Greit å prøve å automatisk laste opp filene på nytt når man får nett
- Kan gi forslag for hvilken TBM man skal koble seg til som TBM Inspector basert på MAC-adresse
- Pass på at man kan registrere unntakstilstander og ikke blir låst til kategoriene
- På bildet av kutterhodet burde man kunne dreie bildet
- Applikasjonen bør ikke være avhengig av størrelsen på displayet

Sak nr 02/2019

- Kontrakt for avtale mellom parter ble signert av Helge

08.02.2019 Knut Wiig

A2 Møteinnkalling Bacheloroppgave 29 Sprint Review 1 m/ Oppdragsgiver

Tidspunkt/sted: Fredag 08.02.2019 kl 13:00 – 15:00, Gløshaugen Byggteknisk 2-048.

Følgende personer innkalles:

Kristian Nybakken Lund

Knut Wiig

Mats Davidsen

Sindre Log

Amund Bruland

Helge-Ivar Frostad

Agenda

Sak nr 01/2019 Gjennomgang av prototype.

Sak nr 02/2019 Se på arbeidskontrakt.

Møte planlegges avsluttet ca kl 15.00

Ta kontakt med undertegnede dersom du ikke har anledning til å komme

Med vennlig hilsen

Knut Wiig

Trondheim 05.02.2019

A2 Referat fra prosjektmøte bacheloroppgave, sprint review 1 m/ Oppdragsgiver

Dato og tid: Fredag 08.02.2019 kl 12:30 – 14.30

Sted: Grupperom 2-048 Institutt for bygg og miljøteknikk 2 etg

Til stede:

Mats Davidsen
Knut Wiig
Kristian Nybakken Lund
Helge Hafting (veileder)

Sindre Log
Helge-Ivar Frostad

Frafall:

Pål Drevland Jakobsen
Amund Bruland

Sak nr 01/2019

- INSPEKTØR:
 - Serienummer på TBM:
 - * To bokstaver som beskriver type
 - * Maskinnummer
 - * Tall som indikerer størrelse på TBM
 - * Tall som indikerer hvilke prosjekt det er på
 - Inspektør ser på slitasje på ring og lager.
 - Hvorfor vi sjekker er hovedsakelig på grunn av slitasje.
 - Varierer hvor mange kuttere som inspiserer.
 - En posisjon kan ikke stå tom.
 - Tegning indikerer toleranse(maks slitasje) på ulike posisjoner
 - Kommentar på slitasje inspeksjon og sist inspeksjon.
 - I en inspeksjon kunne gå inn på en allerede inspisert og bytte den ut.
 - Oversiktsbilde over slitasje på hver posisjon.
 - Mere funksjoner på inspeksjonsdashboard.
- PROJECT SUPERVISOR:
 - Regging av ny TBM: ikke knytt opp mot modeller fra Robbins.
 - Samme registrering funksjon av TBM på både ADMIN og Robbins.
- SUPER USER:
 - Mulighet til å hente ut data.
 - Mer statistikk?
- WORKSHOP SUPERVISOR:
 - Vil ha masse cutterdata, typ som superuser.

- Legg inn ringdelnummer. Hver ring har egen produksjonsID og type. Hva slags ringtype man setter på, kan være en input fra bruker (trenger ikke å ha en fast liste). Ønsker å se på forskjellige ringer opp mot kuttere. Se på hvilken ring som er knyttet til hvilken kutter. Kan bruke ringer som ikke er produsert av Robbins.
- Ha med hvilken posisjon kutteren sto i under en inspeksjon.
- Hvor mange ringer har du byttet siden sist du byttet LAGER?
- Mer dokumentasjonsmuligheter (moment). ID/type på kategoriene (kategoriene kan endres etterhvert). Legge til Seals.

Sak nr 02/2019

- Housing kan ikke stå tom.
- Workshop Supervisor må kunne registrere nye kuttere, men noen siter kan ha egne cutterid.
- Må kunne koble robbinserienummer opp mot siteserienummer på kuttere.
- Hvis posisjon er tom, får man beskjed om at MAN MÅ SJEKKE INN SERIENUMMER eller annen løsning. TENKE PÅ DET.

Ingen punkter under eventuelt

08.02.2019 Mats Davidsen

A3 Møteinnkalling Bacheloroppgave 29 Sprint Review 2 m/ Veileder

Tidspunkt/sted: Fredag 22.02.2019 kl 09:00 – 09:30, Grupperom 304 ,Akrinn.

Følgende personer innkalles:

Kristian Nybakken Lund
Knut Wiig
Mats Davidsen
Helge Hafting (veileder)

Agenda

Sak nr 01/2019 Visjonsdokumentet
Sak nr 02/2019 Problemstilling
Sak nr 03/2019 Hovedrapport

Møte planlegges avsluttet ca kl 09:30

Ta kontakt med undertegnede dersom du ikke har anledning til å komme

Med vennlig hilsen

Knut Wiig

Trondheim 05.02.2019

A3 Referat fra prosjektmøte bacheloroppgave, sprint review 2 m/ Veileder

Dato og tid: Fredag 22.02.2019 kl 09:00 – 09:30

Sted: Grupperom G304 Akkrinn

Til stede:

Mats Davidsen
Knut Wiig
Kristian Nybakken Lund
Helge Hafting (veileder)

Frafall: ingen.

Sak nr 01/2019

- Tilbakemeldinger på visjonsdokument fra Helge Hafting:
 - Fjern rammene som oppstår rundt klikkbare funksjoner i pdf
 - Bruk bold i tabelloverskrifter
 - Skriv mer utfyllende om nåværende løsning som vi skal erstatte
 - Ikke referer til User Story i visjonsdokumentet
 - Flytt kryptring fra ikke-funksjonelle krav til funksjonelle krav

Sak nr 02/2019

- Bra definert problemstilling siden vi spesifiserer godt
- Spør The Robbins Company om målinger vi kan bruke i problemstillingen

Sak nr 03/2019

- Det er interessant å se om kravspesifikasjonen ble oppfylt
- Forskjellige kapitler kan bruke forskjellig skrivemåte, men må være den samme skrivemåten innad i hvert kapittel
- For å vise frem Low-Fidelity bør vi ikke bruke video, vi kan heller bruke en kombinasjon av en overordnet plansje og lineære fremvisninger av skjermbilder
- Video kan brukes som en del av brukermanualen for å beskrive ferdig produkt

22.02.2019 Mats Davidsen

A3 Møteinnkalling Bacheloroppgave 29 Sprint Review 2 m/ Oppdragsgiver

Tidspunkt/sted: Fredag 22.02.2019 kl 13:30 – 15:00, Gløshaugen Byggteknisk 2, 2-048.

Følgende personer innkalles:

Kristian Nybakken Lund
Knut Wiig
Mats Davidsen
Sindre Log
Amund Bruland
Helge-Ivar Frostad
Pål Drevland Jakobsen

Agenda

Sak nr 01/2019 Problemstilling, veiledning
Sak nr 02/2019 Fremvisning av prototype
Sak nr 03/2019 Eventuelt

Møte planlegges avsluttet ca kl 15:00

Ta kontakt med undertegnede dersom du ikke har anledning til å komme

Med vennlig hilsen

Knut Wiig

Trondheim 05.02.2019

A3 Referat fra prosjektmøte bacheloroppgave, sprint review 2 m/ Oppdragsgiver

Dato og tid: Fredag 22.02.2019 kl 13:30 – 15.00

Sted: Grupperom 2-048 Institutt for bygg og miljøteknikk 2 etg

Til stede:

Mats Davidsen
Knut Wiig
Kristian Nybakken Lund
Helge Hafting (veileder)
Helge-Ivar Frostad
Amund Bruland

Frafall:

Pål Drevland Jakobsen
Sindre Log

Sak nr 01/2019

- Kan si “Redusere ressursbruk og øke kvaliteten”
- Det vi vet er at det tar ca 1 time i snitt per bytte av kutter, dette er den fysiske prosessen ved å bytte kutter og vi kan bruke dette som et mål på hvor lang tid tar.
- Vi kan legge litt trykk på at vårt program hjelper med å ta beslutninger for de som inspiserer, som deretter vil senke tiden det tar per kutter
- Selve registreringen kan vi anta at er lik for vår løsning og nåværende, det er det som kommer etterpå vi sparer tid
- Legg inn at det også kan hjelpe de på verkstedene å ta beslutninger
- “Dreiemodellen” kan hjelpe til å øke effektiviteten

Sak nr 02/2019

- Inspektør er “operator” på engelsk
- Navnet på applikasjonen: Prøv å få med cutter, inspection, wear og change
- Hopp over valg av TBM når det bare er en TBM på prosjekt
- Kan hende at et bytte skjer over flere skift, fordi det tar ca 1 time per kuttere. Skift kan ofte være på rundt 10 timer
 - Kan evt ha med “Continue inspection”
 - Skal kunne kvittere seg ut hvis man ikke er helt ferdig med inspeksjon
- Kan ha en oversikt over transaksjoner til databasen
- Få opp ID som står i på dette tidspunktet for å sjekke
- Sjeldent at man bytter alle kuttere, så må ha en ny løsning for å avslutte inspeksjoner
 - F.eks Pause”
- Legg inn data som chainage osv ved oppstart av et prosjekt, dette er nødvendigvis ikke null. Supervisor kan evt legge inn dette når man starter boring/registrerer TBM for prosjekt

- Vedlikehold: Vil ha muligheten til å se hvor lenge kutteren har vært i TBM siden sist
- Ikke legg vekt på Data som skal hentes ut, legg heller ved dette som eksempler på hvordan dette kan gjøres i fremtiden.

Sak nr 03/2019

- Sindre Log sa til Amund at foreslått touchpad var fra 2014 og vil at vi skal sjekke om det finnes en nyere modell
- Vi kan kjøpe en enkel pad i byen og få godtgjørelse fra NTNU, husk å send forslag før vi kjøper

22.02.2019 Mats Davidsen.

A4 Møteinnkalling Bacheloroppgave 29 Sprint Review 3 m/ Veileder

Tidspunkt/sted: Fredag 01.03.2019 kl 09:00 – 09:30, G104 Akrinn

Følgende personer innkalles:

Kristian Nybakken Lund
Knut Wiig
Mats Davidsen
Helge Hafting (veileder)

Agenda

Sak nr 01/2019 Justering av user stories
Sak nr 02/2019 Spørsmål knyttet til hovedrapporten
Sak nr 03/2019 Dokumentasjon av koden
Sak nr 04/2019 Eventuelt

Møte planlegges avsluttet ca kl 09:30

Ta kontakt med undertegnede dersom du ikke har anledning til å komme

Med vennlig hilsen

Knut Wiig

Trondheim 28.02.2019

A4 Referat fra prosjektmøte bacheloroppgave, sprint review 3 m/ Veileder

Dato og tid: Fredag 01.03.2019 kl 09:00 – 09:30

Sted: Grupperom G304 Akrinn

Til stede:

Mats Davidsen

Knut Wiig

Kristian Nybakken Lund

Frafall:

Helge Hafting (veileder)

Sak nr 01/2019

På grunn av frafallet ble ikke møte gjennomført.

01.03.2019 Knut Wiig

A4 Møteinnkalling Bacheloroppgave 29 Sprint Review 3 m/ Oppdragsgiver

Tidspunkt/sted: Fredag 01.03.2019 kl 13:30 – 15:00, Gløshaugen Byggeteknisk 2, 2-048.

Følgende personer innkalles:

Kristian Nybakken Lund
Knut Wiig
Mats Davidsen
Sindre Log
Amund Bruland
Helge-Ivar Frostad
Pål Drevland Jakobsen

Agenda

Sak nr 01/2019 Justering av user stories
Sak nr 02/2019 Fremvisning av prototype versjon2
Sak nr 03/2019 Spørsmål knyttet til arbeidsprosess
Sak nr 04/2019 Eventuelt

Møte planlegges avsluttet ca kl 15:00

Ta kontakt med undertegnede dersom du ikke har anledning til å komme

Med vennlig hilsen

Knut Wiig

Trondheim 28.02.2019

A4 Referat fra prosjektmøte bacheloroppgave, sprint review 3 m/ Oppdagsgiver

Dato og tid: Fredag 08.02.2019 kl 09:00 – 09:30

Sted: Grupperom G304 Akkrinn

Til stede:

Mats Davidsen

Knut Wiig

Kristian Nybakken Lund

Helge-Ivar Frostad

Frafall:

Sindre Log

Amund Bruland

Pål Drevland Jakobsen

Sak nr 01/2019

Siden Helge- Ivar Frostad ikke var noe direkte kunde fikk vi ikke svar på spørsmålene eller godkjenning av user stories.

01.03.2019 Knut Wiig

A5 Møteinnkalling Bacheloroppgave 29 Sprint Review 4 m/ Veileder

Tidspunkt/sted: Fredag 22.03.2019 kl 09:00 – 09:30, G304 Akrinn

Følgende personer innkalles:

Kristian Nybakken Lund
Knut Wiig
Mats Davidsen
Helge Hafting (veileder)

Agenda

Sak nr 01/2019 Justering av user stories
Sak nr 02/2019 Spørsmål knyttet til hovedrapporten
Sak nr 03/2019 Dokumentasjon av koden
Sak nr 04/2019 Eventuelt

Møte planlegges avsluttet ca kl 09:30

Ta kontakt med undertegnede dersom du ikke har anledning til å komme

Med vennlig hilsen

Knut Wiig

Trondheim 20.03.2019

A5 Referat fra prosjektmøte bacheloroppgave, møte med veileder nr 4

Dato og tid: Fredag 22.03.2019 kl 09:00 – 09.30

Sted: Grupperom 304 Akrinn

Til stede:

Mats Davidsen

Knut Wiig

Helge Hafting (veileder)

Frafall:

Kristian Nybakken Lund

Sak nr 01/2019

- Bra at dere informerer, det er naturlig at planene justeres underveis

Sak nr 02/2019

- Møtereferat skal printes sammen med resten, kan ha de forskjellige dokumentene i forskjellige deler av oppgaven.
- Å gå litt over 40 sider på hovedrapporten er ikke noe problem, bedre å lage en fin rapport på 45 sider enn en som er trykket sammen for å holde seg under 40.
- Forside bachelor: Den skal være akkurat slik malen sier, siden den skal arkiveres
- Kan hende at et bytte skjer over flere skift, fordi det tar ca 1 time per kuttere. Skift kan ofte være på rundt 10 timer
- Teori: Ser veldig bra ut slik vi har tenkt

Sak nr 03/2019

- Referanser til bibliotek brukt i kode: Dokumenter det i dokumentasjonen til koden
 - Pass på lisenser, sjekk om det er gratis for kommersiell bruk

Sak nr 04/2019

- Vis kutterslitasje ved forrige inspeksjon på hovedsiden for en posisjon
- Legg inn litt flere sjekker når man skal slette en inspeksjon - farlig å bare ha en knapp for så å slette.

22.03.2019 Mats Davidsen.

A5 Møteinnkalling Bacheloroppgave 29 Sprint Review 4 m/ Oppdragsgiver

Tidspunkt/sted: Fredag 22.03.2019 kl 13:30 – 15:00, Gløshaugen Byggteknisk 2, 2-048.

Følgende personer innkalles:

Kristian Nybakken Lund
Knut Wiig
Mats Davidsen
Sindre Log
Amund Bruland
Helge-Ivar Frostad
Pål Drevland Jakobsen

Agenda

Sak nr 01/2019 Fremvisning av systemet
Sak nr 02/2019 Spørsmål
Sak nr 03/2019 Eventuelt

Møte planlegges avsluttet ca kl 15:00

Ta kontakt med undertegnede dersom du ikke har anledning til å komme

Med vennlig hilsen

Knut Wiig

Trondheim 20.03.2019

A5 Referat fra prosjektmøte bacheloroppgave, møte med oppdragsgiver nr 4

Dato og tid: Fredag 22.03.2019 kl 13:30 – 15.00

Sted: Grupperom 2-048 Institutt for bygg og miljøteknikk 2 etg

Til stede:

Mats Davidsen
Knut Wiig
Kristian Nybakken Lund
Helge Hafting (veileder)
Helge-Ivar Frostad
Amund Bruland
Pål Drevland Jakobsen

Frafall:

Kristian Nybakken Lund
Sindre Log

Sak nr 01/2019

- Sorter cutterene alfabetisk
- Det vi vet er at det tar ca 1 time i snitt per bytte av kutter, dette er den fysiske prosessen ved å bytte kutter og vi kan bruke dette som et mål på hvor lang tid tar.
- Cutter change: Vis hvilken kutter som står i posisjonen før byttet
- Chainage osv registreres i starten av inspeksjonen, men kan også registreres senere men da må brukeren tvinges til det

Sak nr 02/2019

- Bra at dere justerer user storiene underveis, da dette er en naturlig del av denne prosessen.

Sak nr 03/2019

- Ingen kommentarer

22.03.2019 Mats Davidsen.

A6 Møteinnkalling Bacheloroppgave 29 Sprint Review 5 m/ Veileder

Tidspunkt/sted: Fredag 05.04.2019 kl 09:00 – 09:45, G204 Akrinn

Følgende personer innkalles:

Kristian Nybakken Lund
Knut Wiig
Mats Davidsen
Helge Hafting (veileder)

Agenda

Sak nr 01/2019 fremvisning av applikasjon
Sak nr 02/2019 Evt spørsmål til applikasjon
Sak nr 03/2019 Gjennomgang og spørsmål til teori
Sak nr 04/2019 Eventuelt

Møte planlegges avsluttet ca kl 09:45

Ta kontakt med undertegnede dersom du ikke har anledning til å komme

Med vennlig hilsen

Knut Wiig

Trondheim 03.04.2019

A6 Referat fra prosjektmøte bacheloroppgave, møte med veileder nr 5

Dato og tid: Fredag 05.04.2019 kl 09:00 – 09.45

Sted: Grupperom 204 Akrinn

Til stede:

Mats Davidsen
Knut Wiig
Helge Hafting (veileder)
Kristian Nybakken Lund

Frafall:

Sak nr 01/2019

- Fremvisning gikk som det skulle

Sak nr 02/2019

- Ingen ytterlige spørsmål

Sak nr 03/2019

- Metodikk i teorikapittel: Overordnet om teknologien og metoden (har brukt database osv)
- Valg av teknologi og metode: Her kan du ha detaljene om teknologien som er beskrevet i metodikken med mer
- Beskrive detaljer rundt SCRUM?
 - Hvis det kan oppsummeres veldig kort, gjør det
 - Begrepsforklaring/ordbok
 - ikke ta med alt om SCRUM, heller henvis til annen litteratur
- Ha et avsnitt om hvordan strukturen i dokumentet for en bacheloroppgave er bygd opp, for å forklare for lesere som ikke er kjent med formatet. Særlig ting som ikke er opplagt ut i fra overskriftene i innholdsfortegnelsen. Tilhører kap 1.
- Kilder:
 - Kilder som referer til andre artikler: KUN referer til ting man har brukt/lest, ikke referansene som finnes i det du referer til.
 - Figurer: Hvis man tar bilder fra andre skal det gjøre HELT klart at dette er tatt fra et annet sted, akkurat som med tekst
 - Ikke nødvendig å referere til hvilken side i et dokument ting er tatt fra, men det skader ikke å ta med dette
 - Informasjon lært via studiet:
 - * Ikke gjengi ordrett, gjengi med egne ord
 - * ref til lærebok om det er brukt
 - * IKKE legg ved forelesningsnotater som vedlegg
 - * LØSNING: ref til hvilke fag det er hentet fra
 - kan sitere verker som ikke er tilgjengelig for alle

- Digitalisering:
 - hva er digitalt? digitalt eller elektronisk
 - lignende prosjekt kan være prosjekter som ligger på samme nivå som oss. (nivå 1)
- Bra at man tar tak i brukervennelighet MMI. Få med at det er touch app.
- Teori:
 - Bør handle om å presentere teori om de forskjellige temaene, slik at leseren får kunnskapen som trengs når man leser resten
 - ikke ta med egne tanker og løsninger her
 - bare beskriv generelt og etablerte ting
 - Ikke bli for ivrig og dra inn ting som ikke hører hjemme under teori
 - For vårt digitaliseringsprosjekt bør vi flytte den delen der vi begynner å snakke hvilke forhåndsregler vi gjorde for å øke sannsynligheten for et vellykket prosjekt. Mens den generelle delen over fortsatt kan ligge der.
- Hafting kan lese gjennom et kap og gi tilbakemelding på dette. Kan ikke gi tilbakemelding på samme kap flere ganger
- timelister: Harfting har motatt alle timelistene som er sendt.

05.04.2019 Mats Davidsen.

Tidspunkt/sted: Fredag 05.04.2019 kl 13:30 – 15:00, Gløshaugen Byggteknisk 2, 2-048.

A6 Møteinnkalling Bacheloroppgave 29 Sprint Review 5 m/ Oppdragsgiver

Tidspunkt/sted: Fredag 05.04.2019 kl 13:30 – 15:00, Gløshaugen Byggteknisk 2, 2-048.

Følgende personer innkalles:

Kristian Nybakken Lund
Knut Wiig
Mats Davidsen
Sindre Log
Amund Bruland
Helge-Ivar Frostad
Pål Drevland Jakobsen

Agenda

Sak nr 01/2019 Fremvisning av applikasjon
Sak nr 02/2019 Spørsmål
Sak nr 03/2019 Eventuelt

Møte planlegges avsluttet ca kl 15:00

Ta kontakt med undertegnede dersom du ikke har anledning til å komme

Med vennlig hilsen
Knut Wiig

Trondheim 03.04.2019

A6 Referat fra prosjektmøte bacheloroppgave, møte med oppdragsgiver nr 5

Dato og tid: Fredag 05.04.2019 kl 13:30 – 15.00

Sted: Grupperom 2-0138 Institutt for bygg og miljøteknikk 2 etg

Til stede:

Mats Davidsen
Knut Wiig
Kristian Nybakken Lund
Helge Hafting (veileder)
Helge-Ivar Frostad
Amund Bruland
Kristian Nybakken Lund
Sindre Log

Frafall:

Pål Drevland Jakobsen

Sak nr 01/2019

Fremvisning av applikasjon.

- Id av TBM: to bokstaver-ett tall(diameter)-modellnr-antall f.eks RE-5-543-43
- omformulere teksten for ongoing insp pagen
- ha med forrige bored hours og bored meter
- chainage og machine hours skal ikke skrives inn for hver insp, heller en supervisor som setter inn dette når man oppretter en TBM
- Må ikke tvinge inspektører til å måtte skrive inn chainage og hours, men huske å måtte gjøre det før insp slutt.
- Evt hente ut bilde fra DB, men først hente ut bilde lokalt.
- Beslutningsstøtte, hvilke chainage den ble bytte på sist. Dato for sist bytte.
- Reason for change, default blank.
- Pos 1 og 3 deler samme kutter, 2 og 4, 5,6, ofte opp til pos 8-. Bytter du ut en av de bytter du ut begge. Så når man bytter i pos 1 må pos 3 automatisk også bli sjekket.
- Når man reg ny TBM, må man vite om hvilke posisjoner som deler kutterID.
- Kutter må kunne ha to posisjoner.
- Om en pos blir blank for en insp, forblir den lik den forrige.
- Når man skal velge bytte, ofte abressive wear, der må man også kunne skrive inn kutterslitasje.
- varsel på maks kutterslitasje, varsel på maks forskjellen mellom nabokutter. Naturlig at dette settes inn når man registrerer en TBM. fra 1-8, 9-xx etc.
- Insp report når man trykker på end insp. f.eks Hvilke pos som har blitt bytta. Få den opp før det varselet om at du er i ferd med å avslutte insp. Få ut hvilke pos som har blitt bytt, grunnen, slitasje målt, maksslitasje. Bruke samme lista som på TBM overview. Vite hvor mange som har blitt bytt. Litt ekstra info vil bli diskutert i noen senere sprintreviews. Også litt generell informasjon(chainage, etc).

- Kunne hake av hvilke posisjoner som skal byttes. Evt gi fargekoder på pos som er planlagt til å bytte. Dette inne på register wear.
- Kommentar på slitasje også. Her burde forrige kommentar komme opp.

Sak nr 02/2019

- Server: Virtuell server på NTNU Bygg, vet ikke om vi får en egen virtuell maskin.
- får en ubuntu server fra NTNU.

Sak nr 03/2019

Eventuelt.

- Ingen ytterligere spørsmål.

05.04.2019 Knut Wiig.

A7 Møteinnkalling Bacheloroppgave 29 Sprint Review 6 m/ Veileder

Tidspunkt/sted: Fredag 12.04.2019 kl 09:00 – 09:45, G301 Akrinn

Følgende personer innkalles:

Kristian Nybakken Lund
Knut Wiig
Mats Davidsen
Helge Hafting (veileder)

Agenda

Sak nr 01/2019 Gjennomgang av hva vi vil ha tilbakemelding på angående teorikap.
Sak nr 02/2019 Gjennomgang av sikkerheten i systemet.
Sak nr 03/2019 Eventuelt

Møte planlegges avsluttet ca kl 09:45

Ta kontakt med undertegnede dersom du ikke har anledning til å komme

Med vennlig hilsen

Knut Wiig

Trondheim 10.04.2019

A7 Referat fra prosjektmøte bacheloroppgave, møte med veileder nr 6

Dato og tid: Fredag 12.04.2019 kl 09:00 – 09.15

Sted: Grupperom 301 Akrinn

Til stede:

Mats Davidsen

Knut Wiig

Helge Hafting (veileder)

Kristian Nybakken Lund

Frafall:

Sak nr 01/2019

- Sikkerheten ble diskutert og partene kom til en enighet.

Sak nr 02/2019

- Helge ble informert om hvordan utviklingsteamet ville ha tilbakemelding på teorien.
- Helge gir tilbakemelding på mail.

Sak nr 03/2019

-
- Foreslår 22 Mai klokken 14:00, utviklingsgruppen skal sjekke med oppdragsgiver og vil gi tilbakemelding til Helge.

12.04.2019 Kristian Nybakken Lund.

A8 Møteinnkalling Bacheloroppgave 29 Sprint Review 7 m/ Veileder

Tidspunkt/sted: Fredag 03.05.2019 kl 09:00 – 09:45, G302 Akrinn

Følgende personer innkalles:

Kristian Nybakken Lund
Knut Wiig
Mats Davidsen
Helge Hafting (veileder)

Agenda

Sak nr 01/2019 Gjennomgang av ferdigstilt programvare.
Sak nr 02/2019 Hovedrapport-spørsmål.
Sak nr 03/2019 Eventuelt.

Møte planlegges avsluttet ca kl 09:45

Ta kontakt med undertegnede dersom du ikke har anledning til å komme

Med vennlig hilsen

Knut Wiig

Trondheim 26.04.2019

A8 Referat fra prosjektmøte bacheloroppgave, møte med veileder nr 7

Dato og tid: Fredag 03.05.2019 kl 09:00 – 09.30

Sted: Grupperom 302 Akrinn

Til stede:

Mats Davidsen

Knut Wiig

Helge Hafting (veileder)

Kristian Nybakken Lund

Frafall:

Sak nr 01/2019

- Fremvisning av produkt.
- Helge nevner at vi bør ta høyde for glemte passord.

Sak nr 02/2019

- Trenger ikke å endre visjonsdokumentet.
- Helge har glemt å sjekke de spørsmålene vi hadde angående visjonsdokument, men mener vi har løst det på en bra måte.

Sak nr 03/2019

- Under vitenskapelige resultater, kan vi skrive produktet ble sånn her”. Kan skrive hva vi har gjort for å svare på problemstillingen.
- Kan spørre de om produktet vårt er bedre enn forrige løsning. Svare at vi har lagt til rette for bruk i produksjon, men har ikke fått testet ordentlig.

Sak nr 04/2019

- Helge har ikke fått lest teori. Tar det til uka.

03.05.2019 Kristian Nybakken Lund.

A8 Møteinnkalling Bacheloroppgave 29 Sprint Review 7 m/ Oppdragsgiver

Tidspunkt/sted: Fredag 03.05.2019 kl 13:30 – 15:00, Gløshaugen Byggteknisk 2, 2-048.

Følgende personer innkalles:

Kristian Nybakken Lund
Knut Wiig
Mats Davidsen
Sindre Log
Amund Bruland
Helge-Ivar Frostad
Pål Drevland Jakobsen

Agenda

Sak nr 01/2019 Fremvisning av ferdigstilt applikasjon.
Sak nr 02/2019 Eventuelt

Møte planlegges avsluttet ca kl 15:00

Ta kontakt med undertegnede dersom du ikke har anledning til å komme

Med vennlig hilsen

Knut Wiig

Trondheim 25.04.2019

A8 Referat fra prosjektmøte bacheloroppgave, møte med oppdragsgiver nr 7

Dato og tid: Fredag 03.05.2019 kl 13:30 – 15.00

Sted: Grupperom 2-048 Institutt for bygg og miljøteknikk 2 etg

Til stede:

Mats Davidsen
Knut Wiig
Kristian Nybakken Lund
Amund Bruland
Kristian Nybakken Lund

Frafall:

Pål Drevland Jakobsen
Sindre Log
Helge-Ivar Frostad

Sak nr 01/2019

Fremvisning av applikasjon.

- Fint at man kan laste ned all informasjon(Super User) og laste det opp i Excel ved hjelp av guidelines.
- Vise prosjektnavnet på supervisor Main, på toppen.
- Register TBM, greit med navnet number of cutters
- Tydliggjøre at machine hours og chainage er startverdier.
- Vise prosjektnavn på Insp sidene også, iallefall på change TBM siden.
- skrive opp mm på data overalt der det trengs.
- Bored meters ikke bored meter
- 'Inspected' som står i grått etter den har blitt inspisert eller byttet må forklares under guidelines at den også inkluderer posisjoner som har blitt byttet.
- videreutvikling, en aktiv handling at man fysisk velger å ikke bytte de posisjonene som er huket av til bytte. Dette skal kunne hjelp til å bevise ovenfor interessentene om man har overbrukt kuttere etc, veldig relevant for kutterleverandørene og entreprenørene. En form for forretningslogging.
- max cutterwear i to ord
- små skrivefeil, liten C i Current Cutter.
- Videreutvikling, få en kvittering på at inspeksjonen blir sendt til server og om det ble gjort suksessfullt.
- videreutvikling, få en graf som viser de tre siste målingene istedet for kun tre tall som det er nå.
-

Sak nr 02/2019

Eventuelt.

- Ingen ytterligere spørsmål.

03.05.2019 Knut Wiig.

A9 Møteinnkalling Bacheloroppgave 29 Sprint Review 8 m/ Veileder

Tidspunkt/sted: Fredag 16.05.2019 kl 09:00 – 09:45, G303 Akrinn

Følgende personer innkalles:

Kristian Nybakken Lund
Knut Wiig
Mats Davidsen
Helge Hafting (veileder)

Agenda

Sak nr 01/2019 Spørsmål angående rapport.
Sak nr 02/2019 Eventuelt.

Møte planlegges avsluttet ca kl 09:45

Ta kontakt med undertegnede dersom du ikke har anledning til å komme

Med vennlig hilsen

Knut Wiig

Trondheim 15.05.2019

Referat fra prosjektmøte bacheloroppgave, møte med veileder nr 8

Dato og tid: Fredag 16.05.2019 kl 11:00 – 11.30

Sted: Grupperom LYU1.064 Lysholmbygget

Til stede:

Mats Elias Davidsen
Knut Wiig
Helge Hafting (veileder)
Kristian Nybakken Lund

Frafall:

Sak nr 01/2019

- Antall sider: Går greit om det går et par sider over 40.
- Dokumentere bibliotek: Bra i systemdokumentasjon, greit å ha dobbelt opp i kode
- Levere kildekode: Kan linke til Git også sende Helge påloggingsinformasjon på mail. Linken kan legges til som vedlegg.
- Presentasjonen av produktet blir et vedlegg.
- Ugyddig URL i referanser: Spør de som hadde lagt den ut om den fortsatt ligger ute et annet sted. Hvis vi ikke får tak i de må vi bare skrive i referansene at vi er klar over at denne linken ikke er tilgjengelig lenger og at vi har prøvd å kontakte kilden. Hvis noen er veldig interessert kan de selv ta kontakt med de som la den ut for å få tak i den.
- Tittel: “CutWear - digitalisering av tunnelbransjen”: Bra tittel.
 - Tittelen skal fylles inn i Inspira

- Sammendrag skal også inn i Inspira. Ingen krise hvis dette blir dobbelt opp, ved at det fortsatt står i rapporten.
- Blanke sider i starten dokumentet skal ikke legges inn, Latex fikser dette selv hvis du sier at dette er et dobbeltsidig dokument.
- Oppgavetekst kapittel: Skrivefeilen i quote skal generelt ikke rettes.
- Vedlegg med egen innholdsfortegnelse: Referansene her kan stå i egne referanselister. Hvis vedlegget følger nummereringen i rapporten kan referansene godt være i referanselisten for hovedoppgaven.
- Diskusjon, etiske problemstillinger: Kan si hvilke problemene feil i programmet vil føre til.
- Miljømessige: Bør ha med dette, gjerne fyll ut mer. Mindre drivstoff på grunn av mer effektivt arbeid

16.05.2019 Mats Elias Davidsen.

B Brukbarhetstest

Disse brukbarhetstestene ble utført i perioden 25.03.19- 05.04.19. Siden samtlige tester kun var avhengig av å ha et nettbrett ble de utført der det var best for testpersonene. Til sammen ble testen gjennomført på seks testpersoner. Under testene fulgte testgruppen de ti prinsippene utarbeidet av K. Gommoll og A. Nicole [34]. Gjennom samtale med oppdragsgiver kom det frem at den gjennomsnittlige TBM Inspektoren har middels forståelse og erfaring med IKT og de er voksne(30 år og opp), så testpersonene var hovedsakelig individer med tilsvarende erfaringer. Testgruppen ville også teste programmet på noen individer med litt høyere erfaring med IKT.

Bakgrunn

Før selve testen begynte fikk individene en grunnleggende opplæring av hvordan prosessene fungerer og nødvendig informasjon til å kunne forstå systemet. Som forklart i kapittel 2.2.6, kunne utviklingsgruppen ikke teste opp mot reelle brukere og derfor måtte disse testpersonen få en grundigere opplæring.

Oppgave

Oppgaven gikk ut på å:

1. Registrere bored meters og bored hours som var gitt på forhånd av testgruppen.
2. Registrere slitasje på 4 mm mer enn forrige måling på posisjon 14.
3. Sette inspeksjonen på pause, for så å fortsette den igjen.
4. Registrere bytte av kutter på posisjon 8 på grunn av at den har hatt en oljelekkasje.
5. Registrere bytte av kutter på posisjon 12 på grunn av for høy slitasje.
6. Lokalisere hvor man sletter en inspeksjon, for deretter å kansellere slettingen.
7. Sende opp og avslutte inspeksjonen.

Resultater

Person A

Alder: 45 år.

Kjønn: Mann.

Yrke: Barneskolelærer.

Erfaring med IKT: Gjennomsnittlig.

Sentrale tilbakemeldinger:

- Når man registrerer en slitasje som er høyere enn hva den maks kan være bør den markeres på en eller annen måte slik at man ser at den bør byttes.
- Litt usikker på hva fargekodene hjelper med på dette tidspunktet.

Person B

Alder: 27 år.

Kjønn: Kvinne.

Yrke: Sykepleier.

Erfaring med IKT: Gjennomsnittlig.

Sentrale tilbakemelding:

- Burde kanskje kunne fylle inn bored meters og bored hours når som helst i inspeksjonen og ikke bare på starten.

- Alt annet var intuitivt når man først hadde fått en innføring av hva som er hva.
- Når man registrerer en slitasje som er høyere enn hva som er tillatt må brukeren få en varseling

Person C

Alder: 33 år.

Kjønn: Mann.

Yrke: Signalmontør.

Erfaring med IKT: Gjennomsnittlig

Sentrale tilbakemeldinger:

- Gjøre forskjellen på pause inspection og end inspection litt større. Var ikke 100 % forståelig hva som var forskjellen på de første gang.
- Legge inn back knapper overalt, slik at man alltid kan avbryte en handling

Person D

Alder: 23 år.

Kjønn: Mann.

Yrke: Student.

Erfaring med IKT: Litt over gjennomsnittlig

Sentrale tilbakemeldinger:

- Kunne registrere slitasje på siden ved kutterbytte, siden det kan tenke seg at man vil registrere slitasje når man bytter.
- Få opp en slags tilbakemelding på hva som er gjort når man har trykket på end inspection.
- Kunne avslutte en inspeksjon uten å lagre endingene

Person E

Alder: 31 år.

Kjønn: Kvinne.

Yrke: Sosionom.

Erfaring med IKT: Litt under gjennomsnittlig.

Sentrale tilbakemeldinger:

- Få tydeligere beskjed om at den posisjonene man har vært inne på og gjort endringer blir lagt nederst.
- Kunne velge å begynne en ny inspeksjon istedenfor å fortsette den gamle når man velger TBM
- Back knappen kunne vært en bakoverpil isteden, som i nettleser

Person F

Alder: 22 år.

Kjønn: Mann.

Yrke: Student.

Erfaring med IKT: Litt over gjennomsnittlig.

Sentrale tilbakemeldinger:

- Kunne slippe å velge TBM hver gang, applikasjonen burde kunne vite hvilke TBM denne brukeren er knyttet til.
- Legge 'bytte TBM' opp i toolbaren.

C Kravdokumentasjon

Kravdokumentasjon CutWear v1.0 - Gruppe 29.

Daidsen, Mats

Lund, Kristian Nybakken

Wiig, Knut

20.05.2019

C1 Introduksjon

Dette vedlegget beskriver de funksjonelle og ikke funksjonelle kravene til applikasjonen CutWear v1.0. Disse kravene kommer i formen user stories og er utarbeidet av utviklingsteamet, med godkjenning av oppdragsgiver. I løpet av prosjektet er det laget tre versjoner av kravspesifikasjonen. Alle tre er lagt til i dette dokumentet. Disse er skrevet på Engelsk slik at utviklere i The Robbins Company kan lese disse.

Her finner man også domenemodellen som er laget for systemet, samt ER-diagrammene. Det ble også utarbeidet tre prototyper i løpet av prosjektet. Første prototype er av typen som dekker samtlige funksjonaliteter og roller. Versjon to og tre er kun for TBM Inspector. En grundigere beskrivelse er knyttet til hver prototype.

C2 User Stories

C2.1 Version 1

The priorities are written in parentheses behind each user story. The priorities range from 1 to 10, where 1 is least significant.

roles:

User - everyone.

TBM Inspector - the person who is doing the inspection.

Administrator - a person with administrative access to the system.

Project Supervisor - a person on site with viewing access to all TBMs in a project.

Super user - a person with reading access to all TBMs in the database.

System Administrator - a person who administers and monitors the system.

Workshop Supervisor - person who register what type of changes done to the Cutters in the workshop.

User story 1. (10)

As an TBM Inspector

I want to Register the cutter wear of the selected cutter into a database

So that we can use it for further inspections

Scenario: Register cutter wear

Given that I have chosen the right cutter by position-ID

When I enter the right measurements into the inputfield

And pressed the "Submit button"

then the new data should be uploaded to the database.

User story 2. (9)

As an TBM Inspector

I want to get an indication of critical cutter wear

So that I can easily change cutters

Scenario: Graphical interface.

Given that I am going to inspect the TBM

When I open the cutter interface

Then I will get a graphic view over the cutters

Scenario: Color labeled.

Given that I am going to inspect the TBM

When I open the cutter interface

Then the cutters will be color labeled based on their status.

Scenario: Bearing change

Given that I am going to inspect the TBM

And one or more cutters are on its third shift without the bearing change

When I open the cutter interface
Then I will be notified that this cutters need housing change.

User story 3. (10)

As an TBM Inspector

I want to have a simple, touch-friendly and efficient user interface

So that I can do the cutter operations quickly with no hassle

Scenario: TBM inspection

Given that I am going to inspect the TBM

And I am using working gloves

And my ability to interact with the application is affected by the environment

When I open the cutter interface

Then interface will be arranged so that it is easy to use

User story 4. (10)

As an TBM Inspector

I want to be able to send data to a database

So that the information is correctly stored

Scenario: Temporary storage offline

Given that the internet connection is unavailable

When I try to submit the data

Then I want to get notified that the submission failed

And that the data is temporary stored in local a csv-file.

Scenario: Feedback on the submission

Given that I have registered data in the forms

When I press to submit

Then I want a feedback if it was successfully submitted to the DB.

Scenario: Timestamp on the submission

Given that I have submitted data

When the data was successfully uploaded

Then I want a timestamp for submit.

User story 5. (9)

As an TBM Inspector

I want to submit previously failed uploads

So that the data in the DB is up to date.

Scenario: Attempt to submit previously failed uploads with internet connection.

Given that there is previously failed data stored locally

When I press the “Failed uploads” button

Then I get a list of the timestamps connected to the failed uploads.

And when I press the “Retry” button, the application will upload the data.

And I will get notified that the upload was successful

Scenario: Attempt to submit previously failed uploads without internet connection

Given that there is previously failed data stored locally

When I press the “Failed uploads” button

Then I get a list of the timestamps connected to the failed uploads.
And when I press the “Retry All” button, the application will fail to upload the data
And I will get notified that the upload failed

User story 6.(1)

As an TBM Inspector
I want to I want to scan the serial number on the cutter
So that I can use minimal time to inspect cutters

User story 7. (10)

As an TBM Inspector
I want to register a change of a cutter
So that the cutter positions is up to date

Scenario: Category

Given that the cutter needs a change due to the cutter becoming unfunctional
When I change the cutter
Then I assign a specific category to the change
And I get the option of adding a comment.

Scenario: Relocate cutter

Given that the cutter is relocated
When I change the cutter
Then I assign a specific the category “Relocated” to this change
And I get the option of adding a comment.

Scenario: Remark do not fall under any pre-set category

Given that the cutter needs a change
And the remark do not fall under any pre-set category
When I change the cutter
Then I can comment on what the remark is.

Scenario: Connecting Cutter- ID to position

Given that the cutter needs a change
When I replace the cutter
Then I need to register the new cutter-ID to this position

Scenario: No cutter in housing

Given that a position has no cutter
When I view the overview
Then the overview gives me feedback that the position is empty

User story 8.(7)

As a TBM Inspector
I want to register Chainage, Machine Hours, Bored Meter, Bored Hours
So that this information is stored in the database to its specific intervention.

Scenario: Register essential information.

Given that I am logged on as a TBM Inspector
When I a have pressed the “new Inspection” button
Then a display with input fields to register the data for Chainage, Machine Hours, Bored Meter, Bored

Hours will show.
And then I fill in data and press “next”.

User story 9. (8)

As a TBM Inspector

I want to choose which TBM in this project I am going to inspect

So that the data i register is connected to the correct TBM

Scenario: Choose TBM

Given that I am logged on as a TBM Inspector

And the application is displaying a list of available TBM's

When I have chosen one of the TBM's

Then I should be able to start the inspection of this TBM

User story 10. (10)

As a Project Supervisor

I want to view the data from all TBM's connected to this project

So that I can review and process the data gathered

Scenario: Number of changes

Given that I have selected a TBM

When the button for “Number of changes”

And chosen a period of time

Then I can see how many changes was made in this period of time.

Scenario: Graph displaying total number of changes per position

Given that I have selected a TBM

When I press the button for “Consumption overview”

Then the application will show the graph displaying consumption per position

Scenario: Graph displaying number of changes in a specific period, per position

Given that I have selected a TBM

When I press the button for “Consumption overview”

And choose a period of time

Then the application will show the graph displaying consumption per position in this period of time

User story 11.(8)

As a Project Supervisor

I want to register a TBM for a project

So that I can use it for the project

Scenario: Selecting TBM type

Given that I get a list of all available types of TBM's

When I select one of these types

Then a TBM of this type will be added to the project

User story 12. (5)

As a Project Supervisor

I want to register a shifts for a project

So that the TBM Inspectors can log their inspections correctly

User story 13. (10)

As a Super User

I want to view all history of an given project

So that I can use this information for further analysis.

Scenario: Selecting project

Given that I get a list of all projects

When I select a project

Then I get multiple options on how to view data from this project

User story 14. (6)

As a Super User, Project Supervisor

I want to view the history of a given cutter

So that I can use this information later.

Scenario: Cutter position

Given that I am logged in as a Super User / Project- or Workshop Supervisor

When I have pressed “Cutter History”

Then the application should display “Cutter History Page”

When I put in a Cutter-ID into the Cutter-ID input field

And press the button “select Cutter”

Then the history of this cutter will be displayed.

User story 15. (10)

As a Workshop Supervisor

I want to register the change done to the cutters in the workshop

So that the data can be used in later prognoses for cutter changes.

Scenario: Register the change done to a cutter

Given that I am logged on as a Workshop Supervisor

When I a have submitted the Cutter-ID into the Cutter-ID input field

And pressed “Register maintenance”

Then a checkboxlist containing the different types of maintenance (steelring, bearing, oil,hub) will appear

And then I will select the maintenance which I have performed on this cutter

And then press “Finish”.

User story 16. (3)

As a Workshop Supervisor

I want to get history of a given cutter

So that i can use this information to determine what maintenance is required.

Scenario: View cutter history.

Given that I am logged on as a Workshop Supervisor

When I a have submitted the Cutter-ID into the Cutter-ID input field

And pressed “View History”

Then a display with this following information: current location, maintenance history, change history (category), etc...

User story 17. (10)

As a Admin, Workshop, Project Supervisor, TBM Inspector

I want to register a new cutter

So that it gets stored in the database

Scenario: add brand new cutter-ID.

Given that I am logged on as a TBM Inspector

And I am on the “Register cutter change” page

When I have inputted the brand new cutter ID into the “New cutter ID” field

And pressed “Submit”

Then the brand new cutter will be stored into the database

User story 17. (10)

As a Admin

I want to register a new project

So that the customer will have a project registered in our system and a system administrator that belongs to this project

User story 18. (8)

As a Admin

I want to register a new type of TBM

So that the users in this software can select this type of TBM

Scenario: Register TBM type

Given that I get an interface allowing me to easily set up the specs of a TBM

When I click the button “Register new TBM”

Then a TBM of this type will be added to the list of available TBM’s

User story 19. (5)

As a System Admin

I want to have a database with user accounts

So that I can handle access to the system

User story 20. (9)

As an Admin

I want to have an view of all TBM’s who have been managed with this system

So that I can manage the different TBM’s

Scenario: View all TBM’s

Given that I have logged in as an Admin

When I have pressed the “View all TBM’s” button

Then the application should display all the TBM’s

And When I chose one of the TBM’s

Then the application opens this TBM’s main page

Scenario: View all TBM’s

Given that I have logged in as an Admin

When I have pressed the “View all TBM’s” button

Then the application should display all the TBM’s

And When I chose one of the TBM’s

Then the application opens this TBM's main page

User story 21. (5)

As a Super User / admin

I want to query the REST-API

So that I can get the data in a JSON file.

User story 22. (9)

As a User

I want to get the information- and instruction page for the given TBM

So that the information is available for me.

User story 23. (10)

As a User

I want to view the data from the database

So that I can review and process the data gathered.

User story 24. (10)

As a User

I want to read the user manual to be able to install and use the application

So that I can register, view and process cutterdata

User story 25 (10)

As a User

I want to logon

So that I can do my job

Scenario: TBM Inspector

Given that I have entered a username connected to a TBM Inspector account

And entered the correct password

When I press "Login" button

Then the application should display the TBM Inspector home page

Scenario: Project Supervisor

Given that I have entered a username connected to a Project Supervisor account

And entered the correct password

When I press "Login" button

Then the application should display the Project Supervisor home page

Scenario: Workshop Supervisor

Given that I have entered a username connected to a Workshop Supervisor account

And entered the correct password

When I press "Login" button

Then the application should display the Workshop Supervisor home page

Scenario: Admin

Given that I have entered a username connected to a Admin account

And entered the correct password

When I press "Login" button

Then the application should display the Admin home page

Scenario: Super User

Given that I have entered a username connected to a Super User account

And entered the correct password

When I press “Login” button

Then the application should display the Super User home page

Scenario: System Admin

Given that I have entered a username connected to a System Admin account

And entered the correct password

When I press “Login” button

Then the application should display the System Admin home page

User story 26. (10)

As a User

I want my data to be handled correctly and securely

So that that all stored data is correct.

User story 27. (8)

As a User

I want the application to be responsive under heavy load

So that the user experience is smooth

User story 28. (10)

As a System Administrator

I want to be able to start new projects

So that we can do our job

User story 29. (6)

As a TBM Inspector

I want to make a comment on an inspection

So that if something of importance happened during the inspection, I can log it.

C2.2 Version 2

date: 26.feb 2019.

The priorities are written in parentheses behind each user story. The priorities range from 1 to 10, where 1 is least significant.

roles:

User - everyone.

TBM Inspector - the person who is doing the inspection.

Administrator - a person with administrative access to the system.

Project Supervisor - a person on site with viewing access to all TBMs in a project.

Super user - a person with reading access to all TBMs in the database.

System Administrator - a person who administers and monitors the system.

Workshop Supervisor - person who register what type of changes done to the Cutters in the workshop.

Changes Version 2 has the same user stories as version 1 with one addition user story, but the priorities has changed.

User story 1. (10)

As an TBM Inspector

I want to Register the cutter wear of the selected cutter into a database

So that we can use it for further inspections

Scenario: Register cutter wear

Given that I have chosen the right cutter by position-ID

When I enter the right measurements into the inputfield

And pressed the “Submit button”

then the new data should be uploaded to the database.

User story 2. (7)

As an TBM Inspector

I want to get an indication of critical cutter wear

So that I can easily change cutters

Scenario: Graphical interface.

Given that I am going to inspect the TBM

When I open the cutter interface

Then I will get a graphic view over the cutters

Scenario: Color labeled.

Given that I am going to inspect the TBM

When I open the cutter interface

Then the cutters will be color labeled based on their status.

Scenario: Bearing change

Given that I am going to inspect the TBM

And one or more cutters are on its third shift without the bearing change

When I open the cutter interface

Then I will be notified that this cutters need housing change.

User story 3. (10)

As an TBM Inspector

I want to have a simple, touch-friendly and efficient user interface

So that I can do the cutter operations quickly with no hassle

Scenario: TBM inspection

Given that I am going to inspect the TBM

And I am using working gloves

And my ability to interact with the application is affected by the environment

When I open the cutter interface

Then interface will be arranged so that it is easy to use

User story 4. (10)

As an TBM Inspector

I want to be able to send data to a database

So that the information is correctly stored

Scenario: Temporary storage offline

Given that the internet connection is unavailable
When I try to submit the data
Then I want to get notified that the submission failed
And that the data is temporary stored in local a csv-file.

Scenario: Feedback on the submission

Given that I have registered data in the forms
When I press to submit
Then I want a feedback if it was successfully submitted to the DB.

Scenario: Timestamp on the submission

Given that I have submitted data
When the data was successfully uploaded
Then I want a timestamp for submit.

User story 5. (10)

As an TBM Inspector
I want to submit previously failed uploads
So that the data in the DB is up to date.

Scenario: Attempt to submit previously failed uploads with internet connection.

Given that there is previously failed data stored locally
When I press the “Failed uploads” button
Then I get a list of the timestamps connected to the failed uploads.
And when I press the “Retry” button, the application will upload the data.
And I will get notified that the upload was successful

Scenario: Attempt to submit previously failed uploads without internet connection

Given that there is previously failed data stored locally
When I press the “Failed uploads” button
Then I get a list of the timestamps connected to the failed uploads.
And when I press the “Retry All” button, the application will fail to upload the data
And I will get notified that the upload failed

User story 6.(1)

As an TBM Inspector
I want to I want to scan the serial number on the cutter
So that I can use minimal time to inspect cutters

User story 7. (10)

As an TBM Inspector
I want to register a change of a cutter
So that the cutter positions is up to date

Scenario: Category

Given that the cutter needs a change due to the cutter becoming unfunctional
When I change the cutter
Then I assign a specific category to the change
And I get the option of adding a comment.

Scenario: Relocate cutter

Given that the cutter is relocated
When I change the cutter
Then I assign a specific the category “Relocated” to this change
And I get the option of adding a comment.

Scenario: Remark do not fall under any pre-set category

Given that the cutter needs a change
And the remark do not fall under any pre-set category
When I change the cutter
Then I can comment on what the remark is.

Scenario: Connecting Cutter- ID to position

Given that the cutter needs a change
When I replace the cutter
Then I need to register the new cutter-ID to this position

Scenario: No cutter in housing

Given that a position has no cutter
When I view the overview
Then the overview gives me feedback that the position is empty

User story 8.(7)

As a TBM Inspector
I want to register Chainage, Machine Hours, Bored Meter, Bored Hours
So that this information is stored in the database to its specific intervention.

Scenario: Register essential information.

Given that I am logged on as a TBM Inspector
When I a have pressed the “new Inspection” button
Then a display with input fields to register the data for Chainage, Machine Hours, Bored Meter, Bored Hours will show.
And then I fill in data and press “next”.

User story 9. (8)

As a TBM Inspector
I want to choose which TBM in this project I am going to inspect
So that the data i register is connected to the correct TBM

Scenario: Choose TBM

Given that I am logged on as a TBM Inspector
And the application is displaying a list of available TBM’s
When I have chosen one of the TBM’s
Then I should be able to start the inspection of this TBM

User story 10. (6)

As a Project Supervisor
I want to view the data from all TBM’s connected to this project
So that I can review and process the data gathered

Scenario: Number of changes

Given that I have selected a TBM

When the button for “Number of changes”

And chosen a period of time

Then I can see how many changes was made in this period of time.

Scenario: Graph displaying total number of changes per position

Given that I have selected a TBM

When I press the button for “Consumption overview”

Then the application will show the graph displaying consumption per position

Scenario: Graph displaying number of changes in a specific period, per position

Given that I have selected a TBM

When I press the button for “Consumption overview”

And choose a period of time

Then the application will show the graph displaying consumption per position in this period of time

User story 11.(8)

As a Project Supervisor

I want to register a TBM for a project

So that I can use it for the project

Scenario: Selecting TBM type

Given that I get a list of all available types of TBM's

When I select one of these types

Then a TBM of this type will be added to the project

User story 12. (5)

As a Project Supervisor

I want to register a shifts for a project

So that the TBM Inspectors can log their inspections correctly

User story 13. (6)

As a Super User

I want to view all history of an given project

So that I can use this information for further analysis.

Scenario: Selecting project

Given that I get a list of all projects

When I select a project

Then I get multiple options on how to view data from this project

User story 14. (5)

As a Super User, Project Supervisor

I want to view the history of a given cutter

So that I can use this information later.

Scenario: Cutter position

Given that I am logged in as a Super User / Project- or Workshop Supervisor

When I have pressed “Cutter History”

Then the application should display “Cutter History Page”
When I put in a Cutter-ID into the Cutter-ID input field
And press the button “select Cutter”
Then the history of this cutter will be displayed.

User story 15. (5)

As a Workshop Supervisor

I want to register the change done to the cutters in the workshop

So that the data can be used in later prognoses for cutter changes.

Scenario: Register the change done to a cutter

Given that I am logged on as a Workshop Supervisor

When I have submitted the Cutter-ID into the Cutter-ID input field

And pressed “Register maintenance”

Then a checkboxlist containing the different types of maintenance (steelring, bearing, oil,hub) will appear

And then I will select the maintenance which I have performed on this cutter

And then press “Finish”.

User story 16. (3)

As a Workshop Supervisor

I want to get history of a given cutter

So that i can use this information to determine what maintenance is required.

Scenario: View cutter history.

Given that I am logged on as a Workshop Supervisor

When I have submitted the Cutter-ID into the Cutter-ID input field

And pressed “View History”

Then a display with this following information: current location, maintenance history, change history (category), etc...

User story 17. (10)

As a Admin, Workshop, Project Supervisor, TBM Inspector

I want to register a new cutter

So that it gets stored in the database

Scenario: add brand new cutter-ID.

Given that I am logged on as a TBM Inspector

And I am on the “Register cutter change” page

When I have inputted the brand new cutter ID into the “New cutter ID” field

And pressed “Submit”

Then the brand new cutter will be stored into the database

User story 17. (10)

As a Admin

I want to register a new project

So that the customer will have a project registered in our system and a system administrator that belongs to this project

User story 18. (5)

As a Admin
I want to register a new type of TBM
So that the users in this software can select this type of TBM

Scenario: Register TBM type

Given that I get an interface allowing me to easily set up the specs of a TBM
When I click the button “Register new TBM”
Then a TBM of this type will be added to the list of available TBM’s

User story 19. (3)

As a System Admin
I want to have a database with user accounts
So that I can handle access to the system

User story 20. (5)

As an Admin
I want to have an view of all TBM’s who have been managed with this system
So that I can manage the different TBM’s

Scenario: View all TBM’s

Given that I have logged in as an Admin
When I have pressed the “View all TBM’s” button
Then the application should display all the TBM’s
And When I chose one of the TBM’s
Then the application opens this TBM’s main page

Scenario: View all TBM’s

Given that I have logged in as an Admin
When I have pressed the “View all TBM’s” button
Then the application should display all the TBM’s
And When I chose one of the TBM’s
Then the application opens this TBM’s main page

User story 21. (5)

As a Super User / admin
I want to query the REST-API
So that I can get the data in a JSON file.

User story 22. (9)

As a User
I want to get the information- and instruction page for the given TBM
So that the information is available for me.

User story 23. (10)

As a User
I want to view the data from the database
So that I can review and process the data gathered.

User story 24. (10)

As a User

I want to read the user manual to be able to install and use the application

So that I can register, view and process cutterdata

User story 25 (10)

As a User

I want to logon

So that I can do my job

Scenario: TBM Inspector

Given that I have entered a username connected to a TBM Inspector account

And entered the correct password

When I press “Login” button

Then the application should display the TBM Inspector home page

Scenario: Project Supervisor

Given that I have entered a username connected to a Project Supervisor account

And entered the correct password

When I press “Login” button

Then the application should display the Project Supervisor home page

Scenario: Workshop Supervisor

Given that I have entered a username connected to a Workshop Supervisor account

And entered the correct password

When I press “Login” button

Then the application should display the Workshop Supervisor home page

Scenario: Admin

Given that I have entered a username connected to a Admin account

And entered the correct password

When I press “Login” button

Then the application should display the Admin home page

Scenario: Super User

Given that I have entered a username connected to a Super User account

And entered the correct password

When I press “Login” button

Then the application should display the Super User home page

Scenario: System Admin

Given that I have entered a username connected to a System Admin account

And entered the correct password

When I press “Login” button

Then the application should display the System Admin home page

User story 26. (10)

As a TBM Inspector

I want to register a new cutter

So that it gets stored in the database

Scenario: add brand new cutter-ID

Given that I am logged on as a TBM Inspector
And I am on the “Register cutter change” page
When I have inputted the brand new cutter ID into the “New cutter ID” field
And pressed “Submit”
Then the brand new cutter will be stored into the database

User story 27. (10)

As a User
I want my data to be handled correctly and securely
So that that all stored data is correct.

User story 28. (8)

As a User
I want the application to be responsive under heavy load
So that the user experience is smooth

User story 29. (10)

As a System Administrator
I want to be able to start new projects
So that we can do our job

User story 30. (6)

As a TBM Inspector
I want to make a comment on an inspection
So that if something of importance happened during the inspection, I can log it.

C2.3 Version 3

date: 3.mai 2019.

The priorities are written in parentheses behind each user story. The priorities range from 1 to 10, where 1 is least significant.

roles:

User - everyone.

TBM Inspector - the person who is doing the inspection.

Administrator - a person with administrative access to the system.

Project Supervisor - a person on site with viewing access to all TBMs in a project.

Super user - a person with reading access to all TBMs in the database.

System Administrator - a person who administers and monitors the system.

Workshop Supervisor - person who register what type of changes done to the Cutters in the workshop.

Changes Version 3 has the same user stories and priorities as version 2, but this version has an additional 6 user stories.

User story 1. (10)

As an TBM Inspector
I want to Register the cutter wear of the selected cutter into a database
So that we can use it for further inspections

Scenario: Register cutter wear

Given that I have chosen the right cutter by position-ID
When I enter the right measurements into the inputfield
And pressed the “Submit button”
then the new data should be uploaded to the database.

User story 2. (7)

As an TBM Inspector
I want to get an indication of critical cutter wear
So that I can easily change cutters

Scenario: Graphical interface.

Given that I am going to inspect the TBM
When I open the cutter interface
Then I will get a graphic view over the cutters

Scenario: Color labeled.

Given that I am going to inspect the TBM
When I open the cutter interface
Then the cutters will be color labeled based on their status.

Scenario: Bearing change

Given that I am going to inspect the TBM
And one or more cutters are on its third shift without the bearing change
When I open the cutter interface
Then I will be notified that this cutters need housing change.

User story 3. (10)

As an TBM Inspector
I want to have a simple, touch-friendly and efficient user interface
So that I can do the cutter operations quickly with no hassle

Scenario: TBM inspection

Given that I am going to inspect the TBM
And I am using working gloves
And my ability to interact with the application is affected by the environment
When I open the cutter interface
Then interface will be arranged so that it is easy to use

User story 4. (10)

As an TBM Inspector
I want to be able to send data to a database
So that the information is correctly stored

Scenario: Temporary storage offline

Given that the internet connection is unavailable
When I try to submit the data
Then I want to get notified that the submission failed
And that the data is temporary stored in local a csv-file.

Scenario: Feedback on the submission

Given that I have registered data in the forms
When I press to submit
Then I want a feedback if it was successfully submitted to the DB.

Scenario: Timestamp on the submission

Given that I have submitted data
When the data was successfully uploaded
Then I want a timestamp for submit.

User story 5. (10)

As an TBM Inspector
I want to submit previously failed uploads
So that the data in the DB is up to date.

Scenario: Attempt to submit previously failed uploads with internet connection.

Given that there is previously failed data stored locally
When I press the “Failed uploads” button
Then I get a list of the timestamps connected to the failed uploads.
And when I press the “Retry” button, the application will upload the data.
And I will get notified that the upload was successful

Scenario: Attempt to submit previously failed uploads without internet connection

Given that there is previously failed data stored locally
When I press the “Failed uploads” button
Then I get a list of the timestamps connected to the failed uploads.
And when I press the “Retry All” button, the application will fail to upload the data
And I will get notified that the upload failed

User story 6.(1)

As an TBM Inspector
I want to I want to scan the serial number on the cutter
So that I can use minimal time to inspect cutters

User story 7. (10)

As an TBM Inspector
I want to register a change of a cutter
So that the cutter positions is up to date

Scenario: Category

Given that the cutter needs a change due to the cutter becoming unfunctional
When I change the cutter
Then I assign a specific category to the change
And I get the option of adding a comment.

Scenario: Relocate cutter

Given that the cutter is relocated
When I change the cutter
Then I assign a specific the category “Relocated” to this change
And I get the option of adding a comment.

Scenario: Remark do not fall under any pre-set category

Given that the cutter needs a change
And the remark do not fall under any pre-set category
When I change the cutter
Then I can comment on what the remark is.

Scenario: Connecting Cutter- ID to position

Given that the cutter needs a change
When I replace the cutter
Then I need to register the new cutter-ID to this position

Scenario: No cutter in housing

Given that a position has no cutter
When I view the overview
Then the overview gives me feedback that the position is empty

User story 8.(7)

As a TBM Inspector
I want to register Chainage, Machine Hours, Bored Meter, Bored Hours
So that this information is stored in the database to its specific intervention.

Scenario: Register essential information.

Given that I am logged on as a TBM Inspector
When I a have pressed the “new Inspection” button
Then a display with input fields to register the data for Chainage, Machine Hours, Bored Meter, Bored Hours will show.
And then I fill in data and press “next”.

User story 9. (8)

As a TBM Inspector
I want to choose which TBM in this project I am going to inspect
So that the data i register is connected to the correct TBM

Scenario: Choose TBM

Given that I am logged on as a TBM Inspector
And the application is displaying a list of available TBM's
When I have chosen one of the TBM's
Then I should be able to start the inspection of this TBM

User story 10. (6)

As a Project Supervisor
I want to view the data from all TBM's connected to this project
So that I can review and process the data gathered

Scenario: Number of changes

Given that I have selected a TBM
When the button for “Number of changes”
And chosen a period of time
Then I can see how many changes was made in this period of time.

Scenario: Graph displaying total number of changes per position

Given that I have selected a TBM

When I press the button for “Consumption overview”
Then the application will show the graph displaying consumption per position

Scenario: Graph displaying number of changes in a specific period, per position

Given that I have selected a TBM
When I press the button for “Consumption overview”
And choose a period of time
Then the application will show the graph displaying consumption per position in this period of time

User story 11. (8)

As a Project Supervisor
I want to register a TBM for a project
So that I can use it for the project

Scenario: Selecting TBM type

Given that I get a list of all available types of TBM’s
When I select one of these types
Then a TBM of this type will be added to the project

User story 12. (5)

As a Project Supervisor
I want to register a shifts for a project
So that the TBM Inspectors can log their inspections correctly

User story 13. (6)

As a Super User
I want to view all history of an given project
So that I can use this information for further analysis.

Scenario: Selecting project

Given that I get a list of all projects
When I select a project
Then I get multiple options on how to view data from this project

User story 14. (5)

As a Super User, Project Supervisor
I want to view the history of a given cutter
So that I can use this information later.

Scenario: Cutter position

Given that I am logged in as a Super User / Project- or Workshop Supervisor
When I have pressed “Cutter History”
Then the application should display “Cutter History Page”
When I put in a Cutter-ID into the Cutter-ID input field
And press the button “select Cutter”
Then the history of this cutter will be displayed.

User story 15. (5)

As a Workshop Supervisor

I want to register the change done to the cutters in the workshop
So that the data can be used in later prognoses for cutter changes.

Scenario: Register the change done to a cutter

Given that I am logged on as a Workshop Supervisor

When I have submitted the Cutter-ID into the Cutter-ID input field

And pressed “Register maintenance”

Then a checkboxlist containing the different types of maintenance (steelring, bearing, oil,hub) will appear

And then I will select the maintenance which I have performed on this cutter

And then press “Finish”.

User story 16. (3)

As a Workshop Supervisor

I want to get history of a given cutter

So that i can use this information to determine what maintenance is required.

Scenario: View cutter history.

Given that I am logged on as a Workshop Supervisor

When I have submitted the Cutter-ID into the Cutter-ID input field

And pressed “View History”

Then a display with this following information: current location, maintenance history, change history (category), etc...

User story 17. (10)

As a Admin, Workshop, Project Supervisor, TBM Inspector

I want to register a new cutter

So that it gets stored in the database

Scenario: add brand new cutter-ID.

Given that I am logged on as a TBM Inspector

And I am on the “Register cutter change” page

When I have inputted the brand new cutter ID into the “New cutter ID” field

And pressed “Submit”

Then the brand new cutter will be stored into the database

User story 17. (10)

As a Admin

I want to register a new project

So that the customer will have a project registered in our system and a Project Supervisor that belongs to this project

User story 18. (5)

As a Admin

I want to register a new type of TBM

So that the users in this software can select this type of TBM

Scenario: Register TBM type

Given that I get an interface allowing me to easily set up the specs of a TBM

When I click the button “Register new TBM”

Then a TBM of this type will be added to the list of available TBM’s

User story 19. (3)

As a System Admin

I want to have a database with user accounts

So that I can handle access to the system

User story 20. (5)

As an Admin

I want to have an view of all TBM's who have been managed with this system

So that I can manage the different TBM's

Scenario: View all TBM's

Given that I have logged in as an Admin

When I have pressed the "View all TBM's" button

Then the application should display all the TBM's

And When I chose one of the TBM's

Then the application opens this TBM's main page

Scenario: View all TBM's

Given that I have logged in as an Admin

When I have pressed the "View all TBM's" button

Then the application should display all the TBM's

And When I chose one of the TBM's

Then the application opens this TBM's main page

User story 21. (5)

As a Super User / admin

I want to query the REST-API

So that I can get the data in a JSON file.

User story 22. (9)

As a User

I want to get the information- and instruction page for the given TBM

So that the information is available for me.

User story 23. (10)

As a User

I want to view the data from the database

So that I can review and process the data gathered.

User story 24. (10)

As a User

I want to read the user manual to be able to install and use the application

So that I can register, view and process cutterdata

User story 25 (10)

As a User

I want to logon
So that I can do my job

Scenario: TBM Inspector

Given that I have entered a username connected to a TBM Inspector account
And entered the correct password
When I press “Login” button
Then the application should display the TBM Inspector home page

Scenario: Project Supervisor

Given that I have entered a username connected to a Project Supervisor account
And entered the correct password
When I press “Login” button
Then the application should display the Project Supervisor home page

Scenario: Workshop Supervisor

Given that I have entered a username connected to a Workshop Supervisor account
And entered the correct password
When I press “Login” button
Then the application should display the Workshop Supervisor home page

Scenario: Admin

Given that I have entered a username connected to a Admin account
And entered the correct password
When I press “Login” button
Then the application should display the Admin home page

Scenario: Super User

Given that I have entered a username connected to a Super User account
And entered the correct password
When I press “Login” button
Then the application should display the Super User home page

Scenario: System Admin

Given that I have entered a username connected to a System Admin account
And entered the correct password
When I press “Login” button
Then the application should display the System Admin home page

User story 26. (10)

As a TBM Inspector
I want to register a new cutter
So that it gets stored in the database

Scenario: add brand new cutter-ID

Given that I am logged on as a TBM Inspector
And I am on the “Register cutter change” page
When I have inputted the brand new cutter ID into the “New cutter ID” field
And pressed “Submit”
Then the brand new cutter will be stored into the database

User story 27. (10)

As a User
I want my data to be handled correctly and securely
So that that all stored data is correct.

User story 28. (8)

As a User
I want the application to be responsive under heavy load
So that the user experience is smooth

User story 29. (10)

As a System Administrator
I want to be able to start new projects
So that we can do our job

User story 30. (6)

As a TBM Inspector
I want to make a comment on an inspection
So that if something of importance happened during the inspection, I can log it.

User story 31. (8)

As an TBM Inspector
I want to Pause the inspection
So that I can continue the inspection later.

User story 32. (8)

As a TBM Inspector
I want to mark a position for cutter change
So that i know which positions to change later on.

User story 33. (9)

As a TBM Inspector
I want to get the previous data on a specific position
So that i can use this data when i make decisions.

User story 34. (10)

As a Project Supervisor
I want to create an TBM Inspector user for a project.
So that the project has a TBM Inspector.

User story 35. (10)

As a Admin
I want to create an Project Supervisor user for a project.
So that the project has a Project Supervisor.

User story 36. (8)

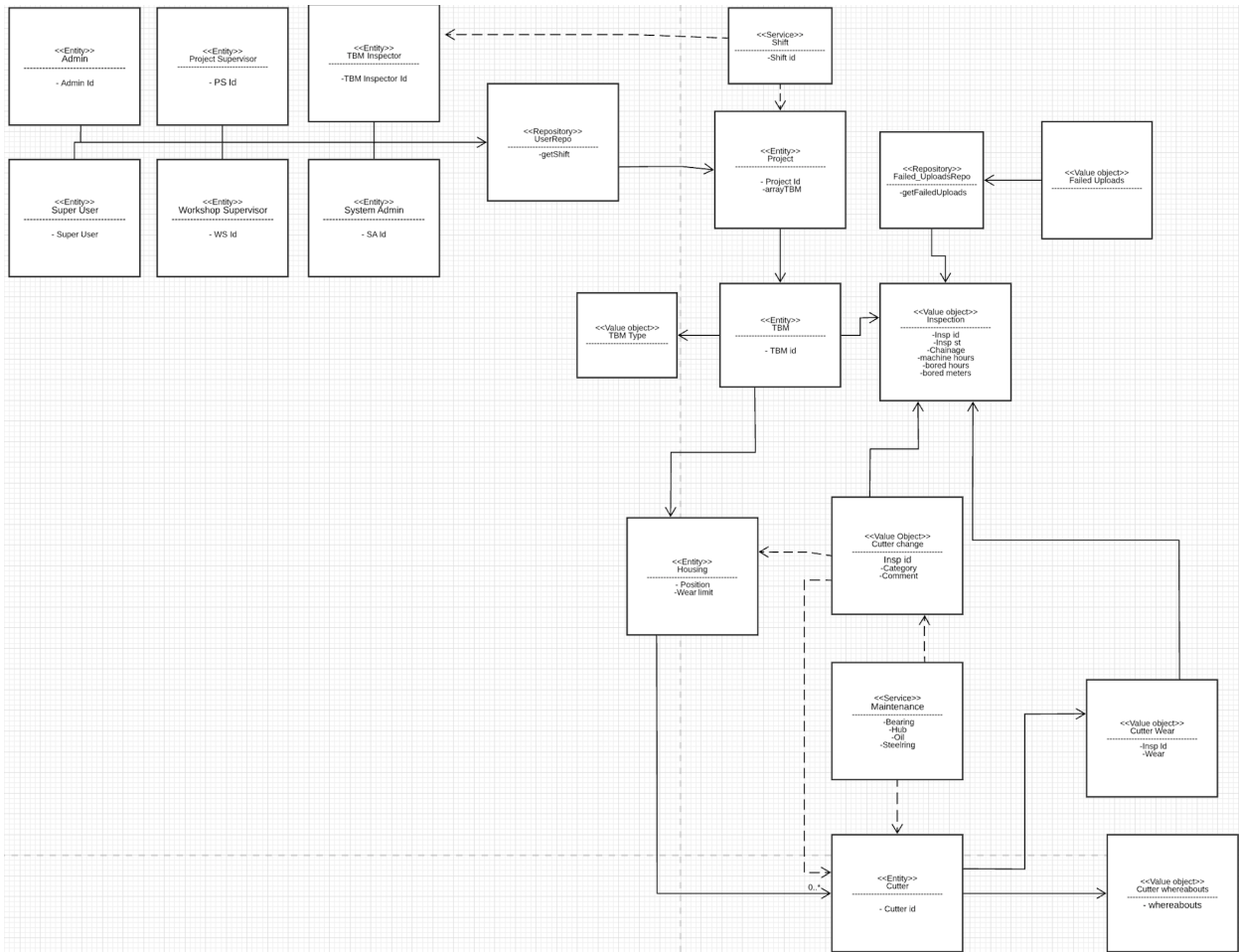
As a TBM Inspector

I want to be able to delete a ongoing inspection.

So that i can use the application as training or deleting a started inspection.

C3 Domenemodell

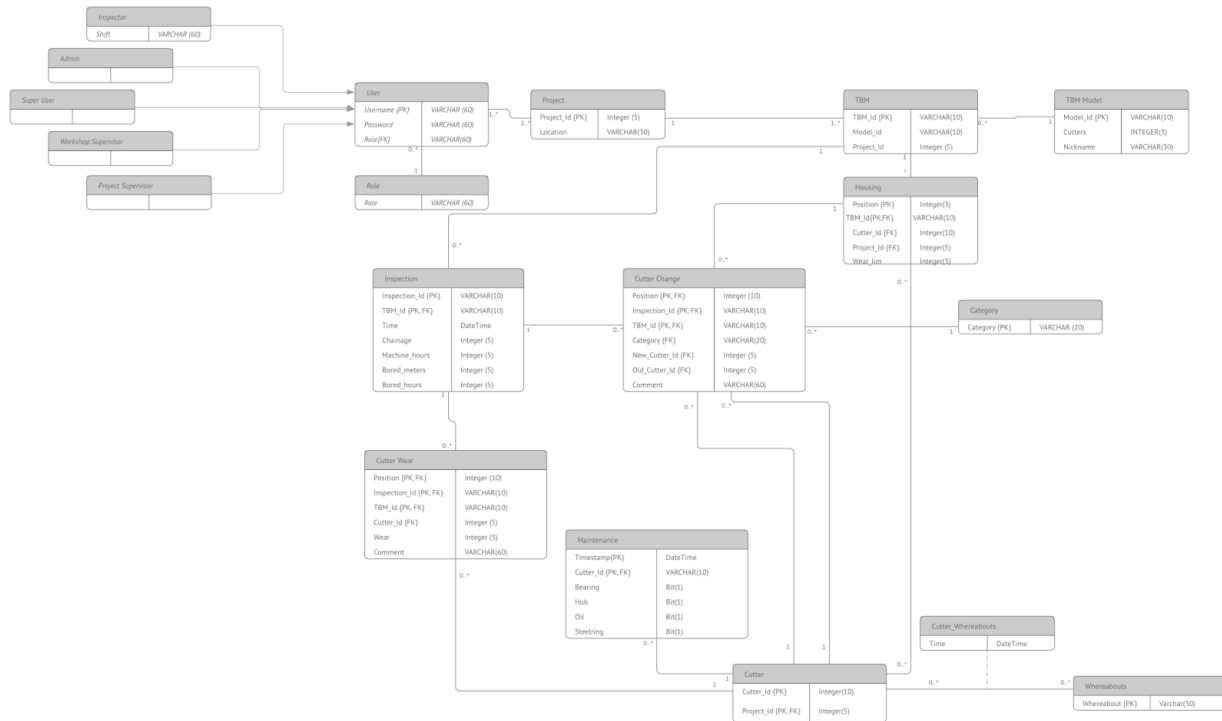
Domenemodellen ble utarbeidet av utviklingsteamet under sprint 1 i samarbeid med oppdragsgiver.



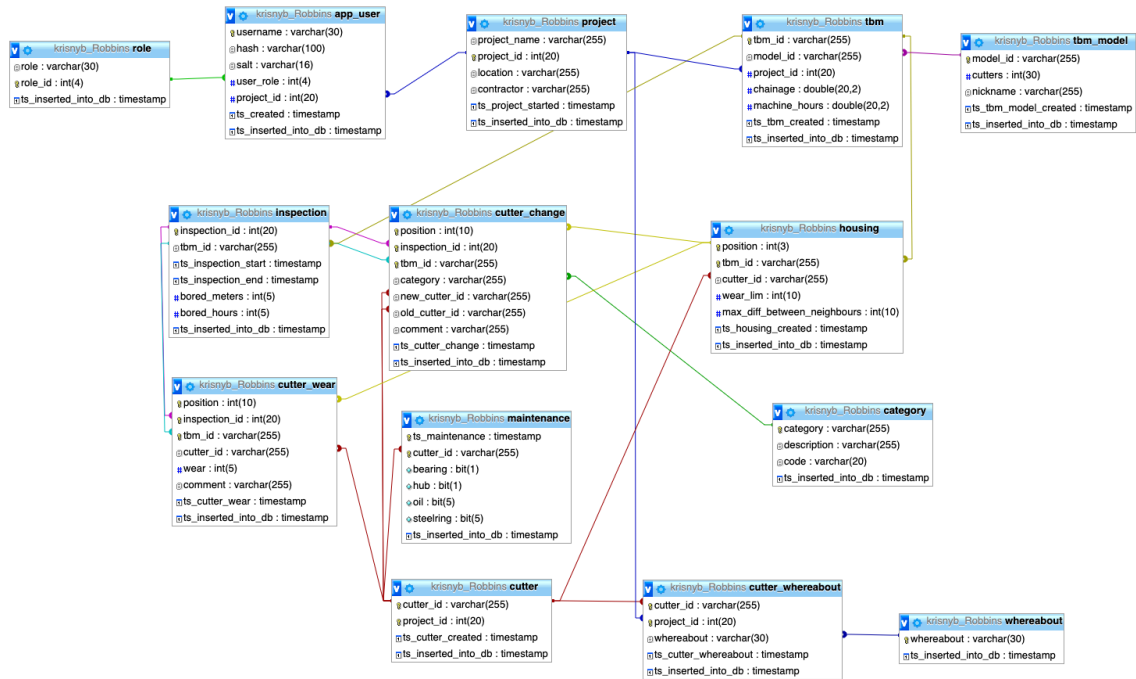
C4 ER-diagram

Det ble utarbeidet to ER-diagrammer i løpet av prosjektet. Versjon 1 ble utarbeidet under sprint 2 og versjon 2 ble utarbeidet i sprint 8 etter at ny data var lagt til underveis i prosjektet. Versjon 2 representerer databasen ved leveringsfristen.

C4.1 ER-diagram versjon 1.



C4.2 ER-diagram versjon 2.



C5 Prototyper

C5.1 Prototype 1

Beskrivelse

Denne prototypen ble utarbeidet i perioden 30.01.2019 – 01.02.2019. Den ble laget basert på første versjon av user stories som ble laget av utviklingsteam og godkjent av oppdragsgiver. Her var det kun user stories med prioritering 6 eller høyere som ble med.

Prototypen skulle gi utviklingsteamet en bedre forståelse av systemet. Ut i fra prototypen fant utviklingsteamet flere mangler ved designet/systemet. Her fant utviklingsteamet ut at det måtte legges til en ny bruker, «Workshop Supervisor» med alle funksjoner knyttet til denne brukeren. Prototype versjon 1 ble også brukt som mal for selve utviklingen av frontend.

Brukere

TBM Inspector – Person som gjør selve inspeksjonen.

Administrator – Person med administrativ tilgang til systemet.

Project Supervisor – Person ute hos kunden som har lesetilgang til all data knyttet til sitt prosjekt og begrenset skrivetilgang.

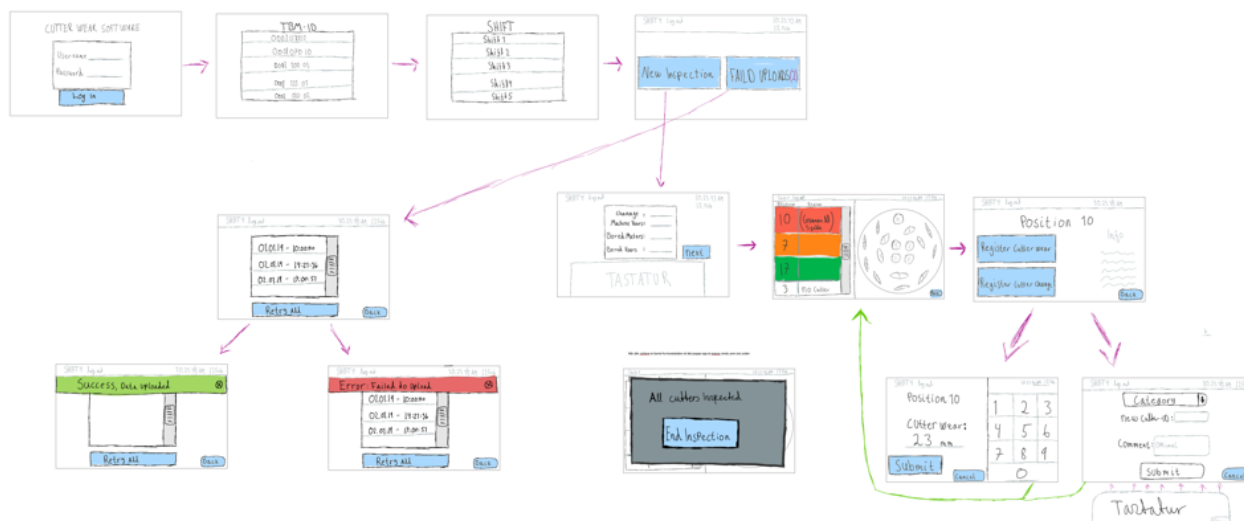
Super User – Person som har lesetilgang til alle prosjekter i systemet.

System Administrator – Person som administrerer og overvåker systemet.

Workshop Supervisor – Person som registrerer hvilke bytter som er gjort på en kutter.

TBM Inspector

Oversikt



individuelle sider

1.0

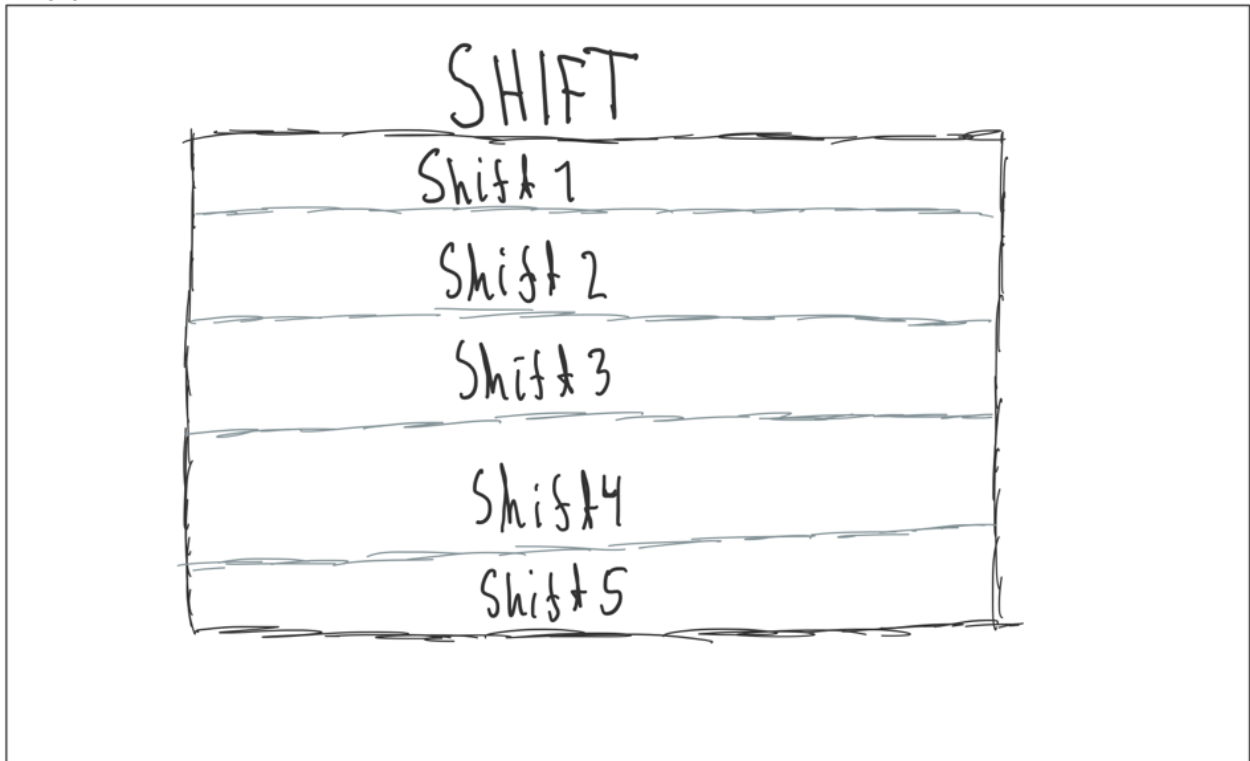
CUTTER WEAR SOFTWARE

Username _____

Password _____

Log In

TBM-ID		
0001	0700	1
0001	070	10
0001	100	03
0001	111	01
0001	020	02



4.0

SHIFT 4 Log out

10.23.43 AM
12. Feb

New Inspection

FAILED UPLOADS (2)

4.1

SHIFT 4 Logout

10.23.43 AM
12. Feb

Change :	_____
Machine Hours:	_____
Bored Meters:	_____
Bored Hours :	_____

next

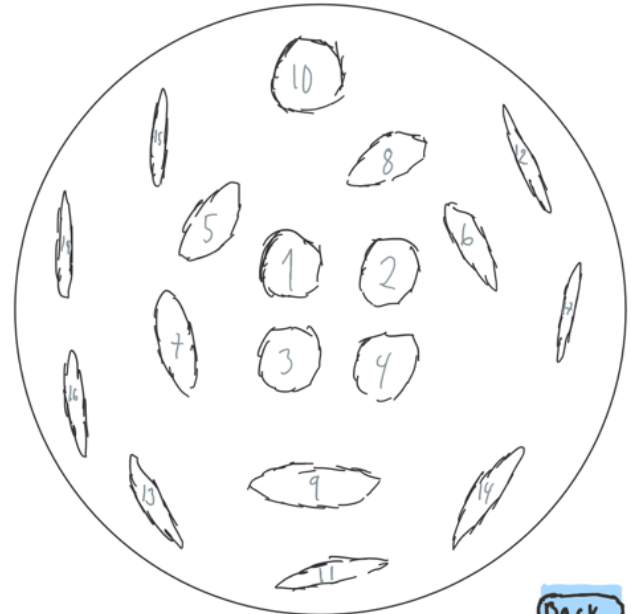
TASTATOR

4.1.1

Shift 4 log call

10.23.46AM 12.Feb

Position	Reason
10	(Grannen bil) Sjekk
7	
17	
3	No cutter



Back

Position 10

Register Cutter wear

Register Cutter Change

Info

~~~~~  
~~~~~  
~~~~~  
~~~~~  
~~~~~

Back

4.1.3

SHIFT 4 log out

10.23.46AM 12.Feb

Position 10

Cutter wear:  
23 mm

|   |   |   |
|---|---|---|
| 1 | 2 | 3 |
| 4 | 5 | 6 |
| 7 | 8 | 9 |
| 0 |   |   |

SHIFT 4 log out 10.23.43 AM 12 Feb

Category

New cutter-ID:

Comment:

↑ ↑ ↑ ↑ ↑ ↑ ↑

Tartarur

SHIFT 4 log out

10.23.43 AM 12 Feb

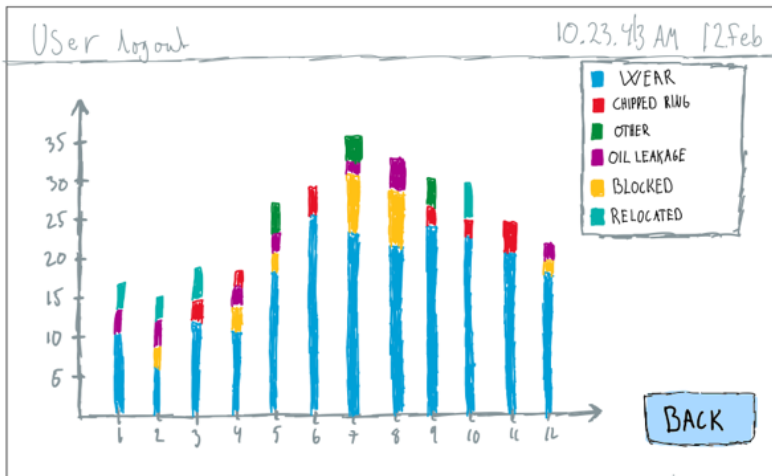
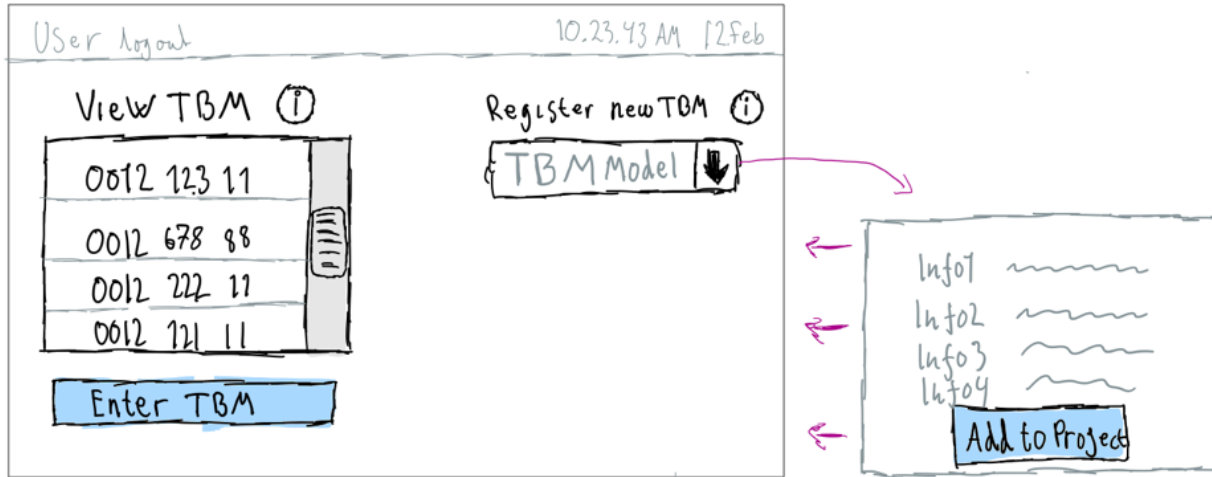
|                     |  |
|---------------------|--|
| 01.01.19 - 10:00:00 |  |
| 02.01.19 - 14:21:36 |  |
| 02.07.19 - 17:00:57 |  |
|                     |  |

Retry All

Back

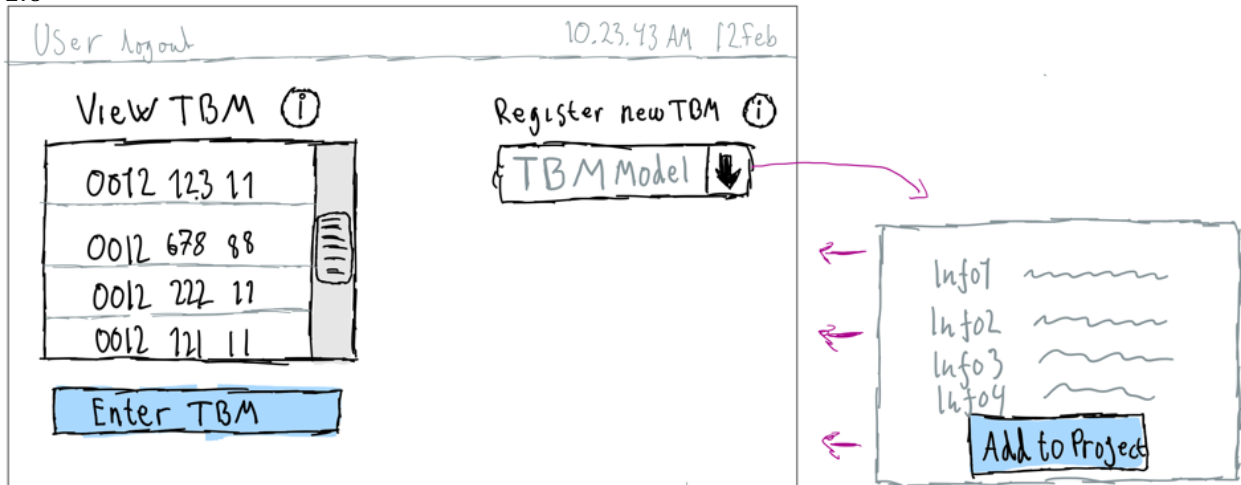
# Project Supervisor

## Oversikt

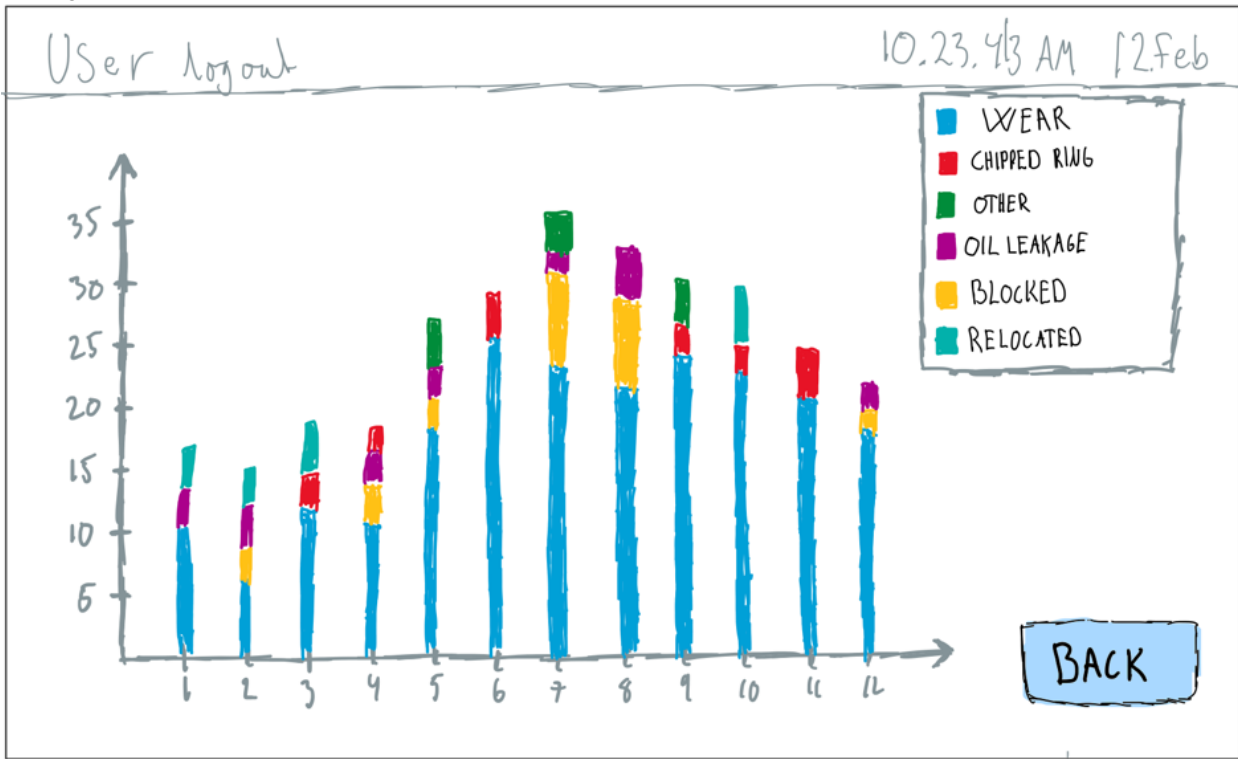


Individuelle sider

1.0







# Super User

## Oversikt

SuperUser App v1.0 10.23.19 AM 12Feb

Select Project (1)

- Rome
- Bergen
- Barcelona
- Leeds

Enter Project

View Cutter History (1)

Cutter-10:

Select cutter

SuperUser App v1.0 10.23.19 AM 12Feb

Rome

View TBM (1)

|      |     |    |
|------|-----|----|
| 0012 | 123 | 11 |
| 0012 | 478 | 88 |
| 0012 | 222 | 11 |
| 0012 | 111 | 11 |

Enter TBM

Project Info:

BACK

SuperUser App v1.0 10.23.19 AM 12Feb

Cutter 10

View by category:

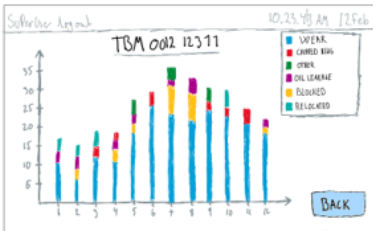
| Date    | Position | Wear | Change   | Maintenance          | Comment |
|---------|----------|------|----------|----------------------|---------|
| 01Jan19 | 32       | 21mm | Wear     |                      |         |
| 04Jan19 | 2        | 8 mm | None     |                      |         |
| 07Jan19 | 2        | 14mm | Replaced |                      |         |
| 09Jan19 | 15       | 23mm | Other    | Bearing needs change |         |

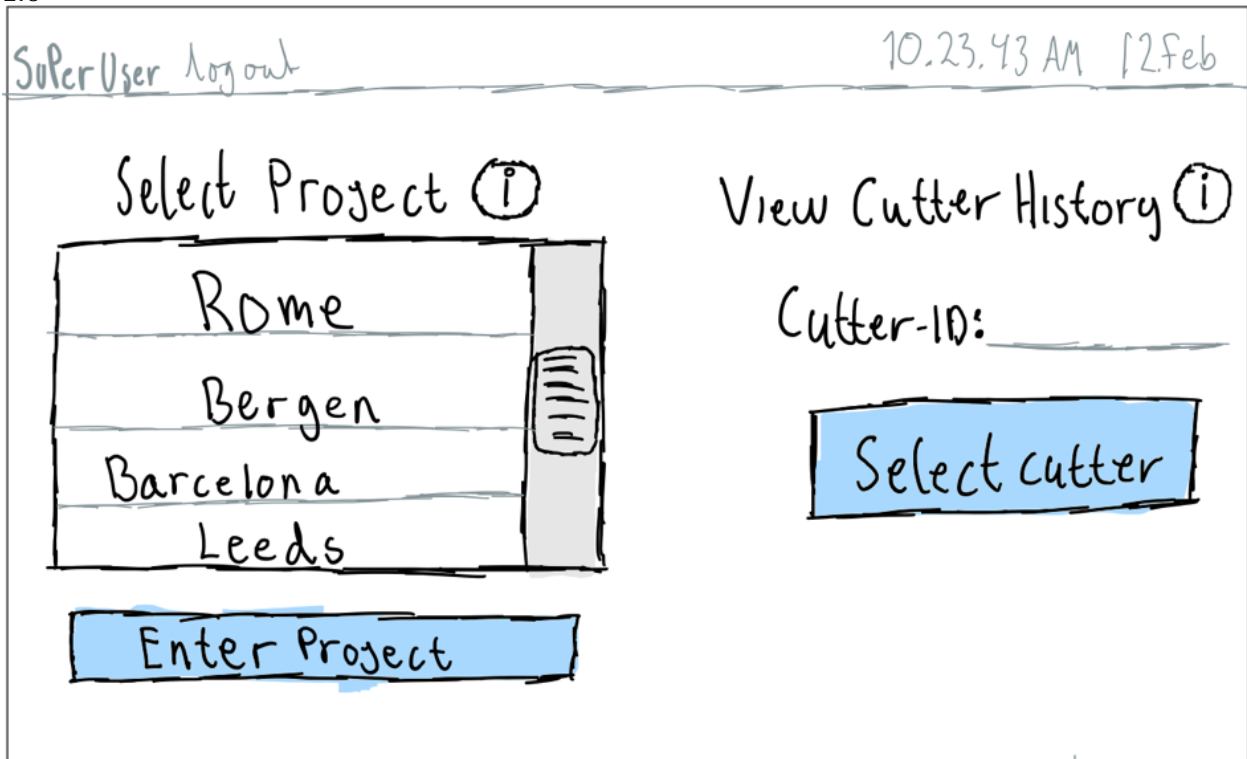
BACK

Cutter 10

View by category:

| Date    | Maintenance |         |     |     |         |
|---------|-------------|---------|-----|-----|---------|
|         | Steering    | Bearing | oil | hub | connect |
| 01Jan19 | X           |         |     |     |         |
| 04Jan19 | X           | X       | X   |     |         |
| 15Jan19 | X           |         |     | X   |         |
| 09Jan19 | X           | X       | X   | X   |         |





Super User Logout

10.23.43 AM 12 Feb

# Rome

View TBM ⓘ

|      |     |    |
|------|-----|----|
| 0012 | 123 | 11 |
| 0012 | 678 | 88 |
| 0012 | 222 | 11 |
| 0012 | 121 | 11 |

Enter TBM

Project Info:

~~~~~  
~~~~~  
~~~~~  
~~~~~  
~~~~~

A

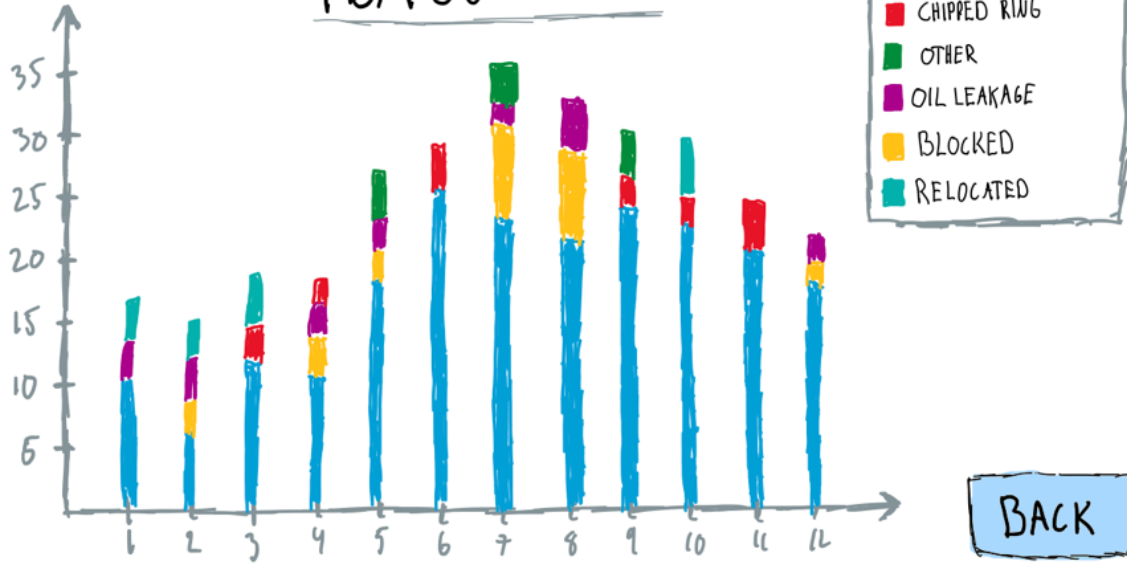
BACK

1.1.1

SuperUser Logout

10.23.43 AM 12.Feb

TBM 0012 12311



BACK

1.2

Super User Logout 10.23.43 AM 12 Feb

Cutter 10

View by category: **Inspection** Maintenance

Date	Position	Wear	Change	Comment
03.01.14	32	21mm	Wear	
06.01.14	2	8 mm	none	
07.01.14	2	16 mm	Relocated	
08.01.14	15	23mm	Other	Bearing needs change

BACK



Cutter 10

View by category: Inspection **Maintenance**

Date	Steering	Bearing	oil	hub	Comment
03.01.14	X				
23.01.14	X	X	X		
15.02.14	X			X	
07.03.14	X	X	X	X	

Administrator

Oversikt

Admin logout 10.23.43 AM 12Feb

Register new Project

Register new TBM-Model



Admin logout 10.23.43 AM 12Feb

New Project

Project name : _____

Location : _____

Create Project

BACK

Admin logout 10.23.43 AM 12Feb

New TBM-model

TBM-ID : _____

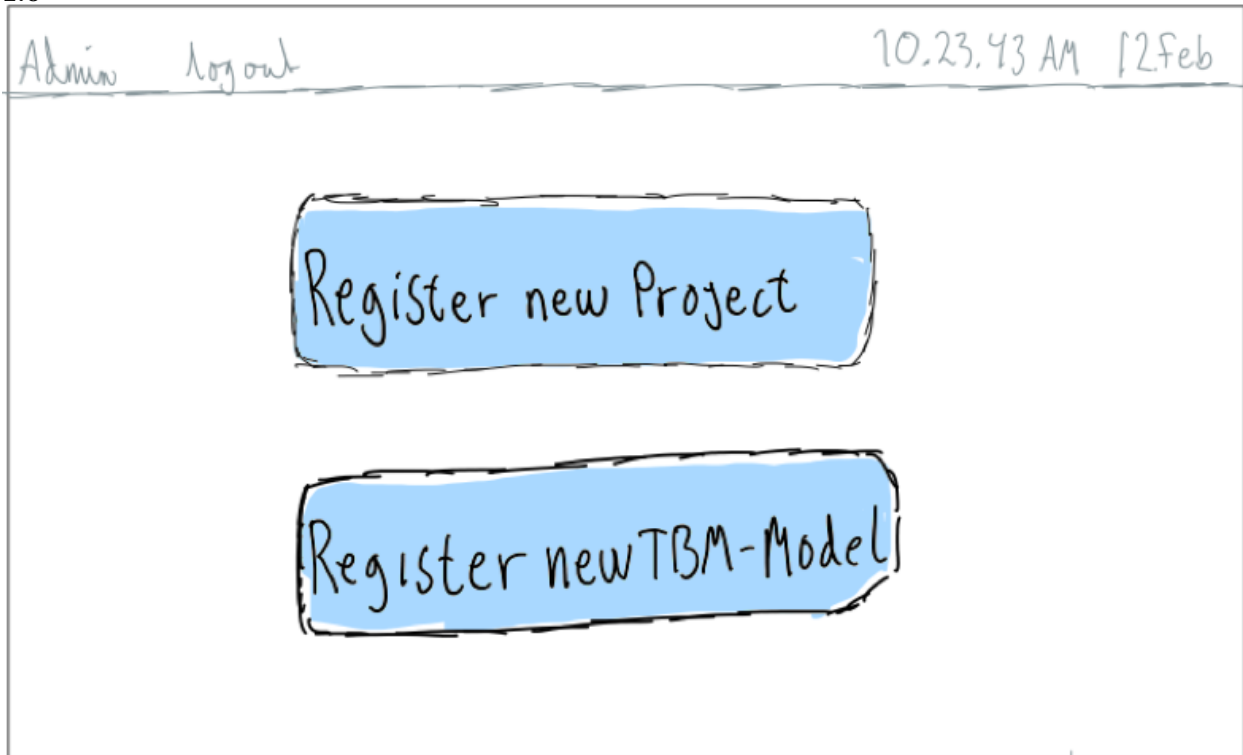
Nr. of cutters : _____

Add Model

BACK

Individuelle sider

1.0



Admin logout 10.23.43 AM 12Feb

New Project

Project name : _____

Location : _____

Create Project

BACK

Admin [logout](#)

10.23.43 AM 12Feb

New TBM-model

TBM-ID : _____

Nr. of cutters : _____

Add Model

BACK

Workshop Supervisor

Oversikt

Log out 10.23.43 AM
12. Feb

Kommer opp tastatur når du trykker på inputfeltet

Cutter ID:

Register change View Cutter

Log out 10.23.43 AM
12. Feb

Cutter 10

① Changes done:

Steering	<input checked="" type="checkbox"/>
Bearing	<input type="checkbox"/>
oil	<input checked="" type="checkbox"/>
hub	<input type="checkbox"/>

② Add comment:
Optional

Submit

Log out 10.23.43 AM
12. Feb

Cutter 10

View by category: Inspection Maintenance

Date	Steering	Bearing	oil	hub	Comment
03.01.14	X				
28.01.14	X	X	X		
15.02.14	X			X	
07.03.14	X	X	X	X	

Back

Individuelle sider

1.0

A hand-drawn wireframe of a web page. At the top left, it says "Log out". At the top right, it shows the time "10.23.43 AM" and the date "12. Feb". Below a horizontal line, the text "Cutter ID:" is followed by an underlined input field containing "InPut here". At the bottom, there are two blue rectangular buttons: "Register Change" on the left and "View Cutter" on the right.

Log out

10.23.43 AM
12. Feb

Cutter 10

① Changes done:

Steering	<input checked="" type="checkbox"/>
Bearing	<input type="checkbox"/>
oil	<input checked="" type="checkbox"/>
hub	<input type="checkbox"/>

① Add Comment:

Optional

Submit

Log out 10.23.43 AM
12. Feb

Cutter 10

View by category: Inspection Maintenance

Date	Steering	Bearing	oil	hub	Comment
03.01.14	X				
23.01.14	X	X	X		
15.02.14	X			X	
07.03.14	X	X	X	X	

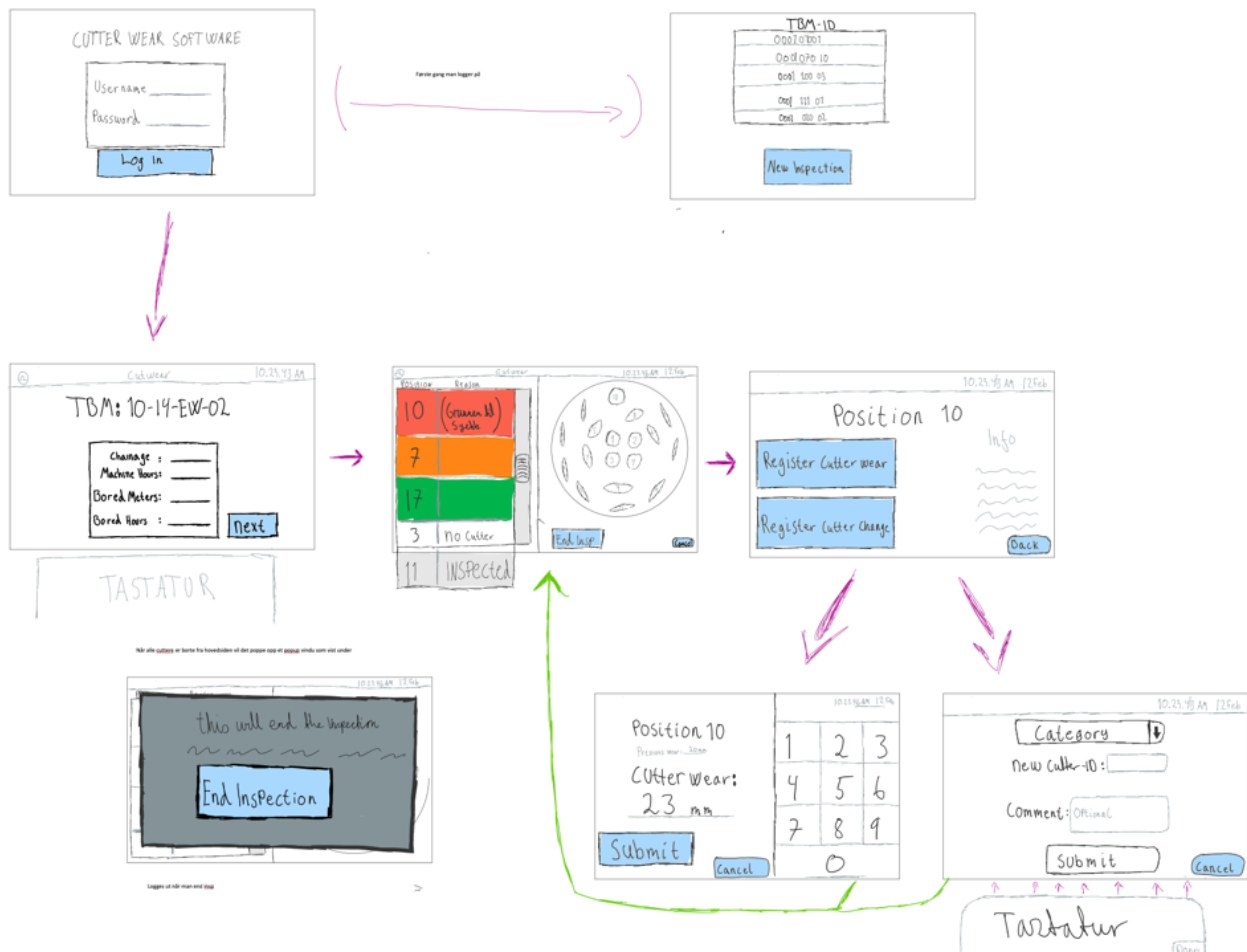
Back

C5.2 Prototype 2

Beskrivelse

Denne prototypen ble utarbeidet 26.02.2019. Den ble laget basert på tilbakemeldingene fra første prototype. Dette er en low fidelity vertikal prototype. Her går man kun inn i detaljer for rollen TBM Inspector. Dette på grunnlag av at TBM Inspector er den viktigste rollen og oppdragsgiver ønsket at teamet fokuserte først og fremst på å få til funksjonalitet knyttet denne brukeren først.

Oversikt



Individuelle sider

1.0

CUTTER WEAR SOFTWARE

Username _____

Password _____

Log In

Når man nå logger inn vil man automatisk komme inn på den TBM'en denne brukeren er registrert på. Første gang man logger inn får man opp valget av TBM'er, se punkt 6.0. Om en bruker må bytte TBM gjøres dette i toolbar, se punkt 7.0. En inspeksjon vil starte når noen man logger seg på.



TBM: 10-14-EW-02

Change :	_____
Machine Hours:	_____
Bored Meters:	_____
Bored Hours :	_____

next

TASTATOR

3.0

Cutwear 10.23.46AM 12.Feb

Position	Reason
10	(Grannen bil) Sjekk
7	
17	
3	no Cutter
11	INSPECTED

End Insp Cancel

Når man har inspisert en posisjon vil denne ikke bli borte som i prototype versjon 1, men heller havne nederst på listen. Her må man da trykke på end inspections for å avslutte inspeksjonen. Se punkt 5.0.

Position 10

Register Cutter wear

Register Cutter change

Info


~~~~~  
~~~~~  
~~~~~  
~~~~~  
~~~~~

Back

4.1

|                                       |  |                   |   |
|---------------------------------------|--|-------------------|---|
| Position 10                           |  | 10.23.46AM 12.Feb |   |
| Previous wear: <u>20mm</u>            |  | 1                 | 2 |
| Cutter wear:                          |  | 4                 | 5 |
| <u>23 mm</u>                          |  | 7                 | 8 |
| <input type="button" value="Submit"/> |  | 0                 |   |
| <input type="button" value="Cancel"/> |  |                   |   |

10.23.4/3 AM 12Feb

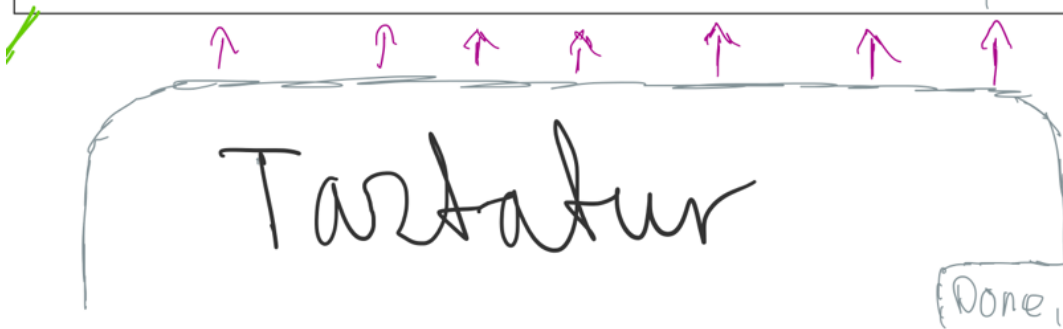
Category 

new cutter-ID:

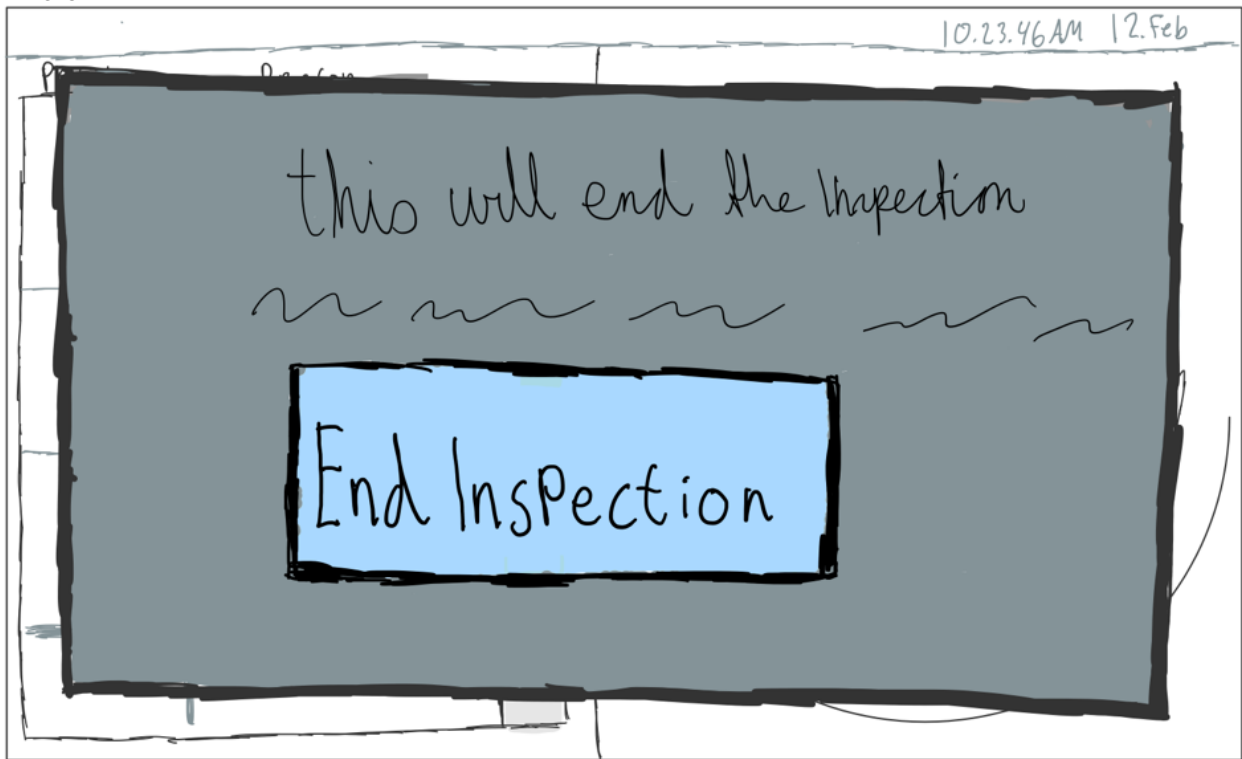
Comment:

Submit

Cancel



5.0



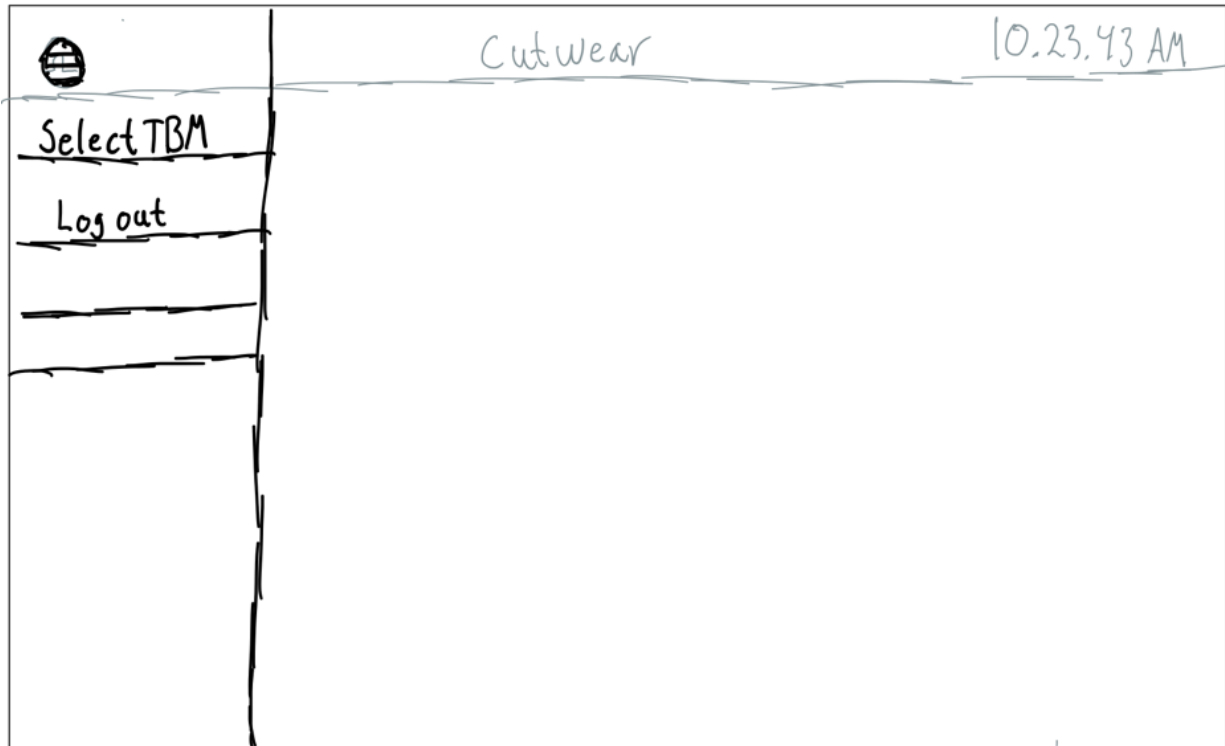
Når man er ferdig blir man automatisk logget av.

# TBM-10

|           |
|-----------|
| 000107001 |
| 000107010 |
| 000110003 |
| 00011101  |
| 000102002 |

New Inspection

7.0



Dette er valgene under toolbaren for en TBM Inspector.

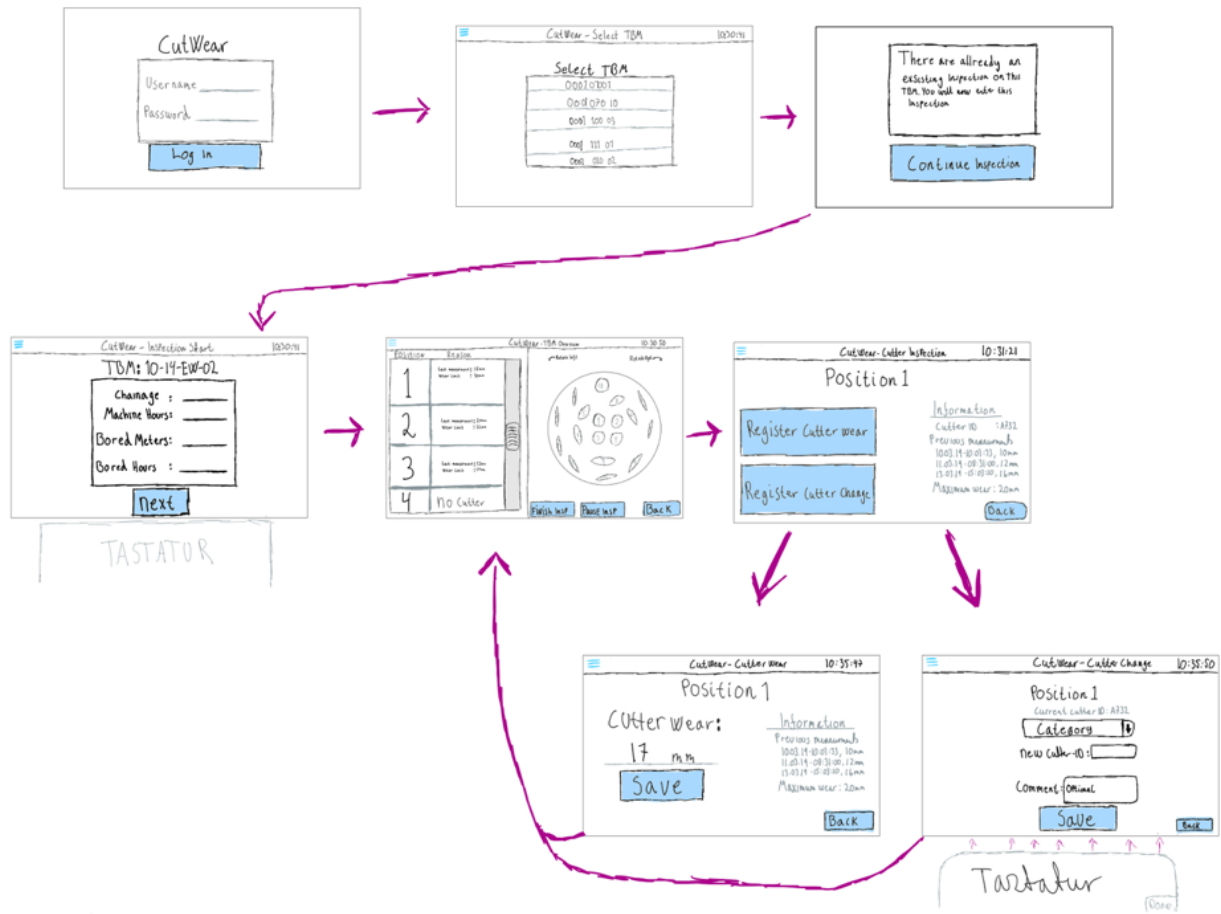


### C5.3 Prototype 3

#### Beskrivelse

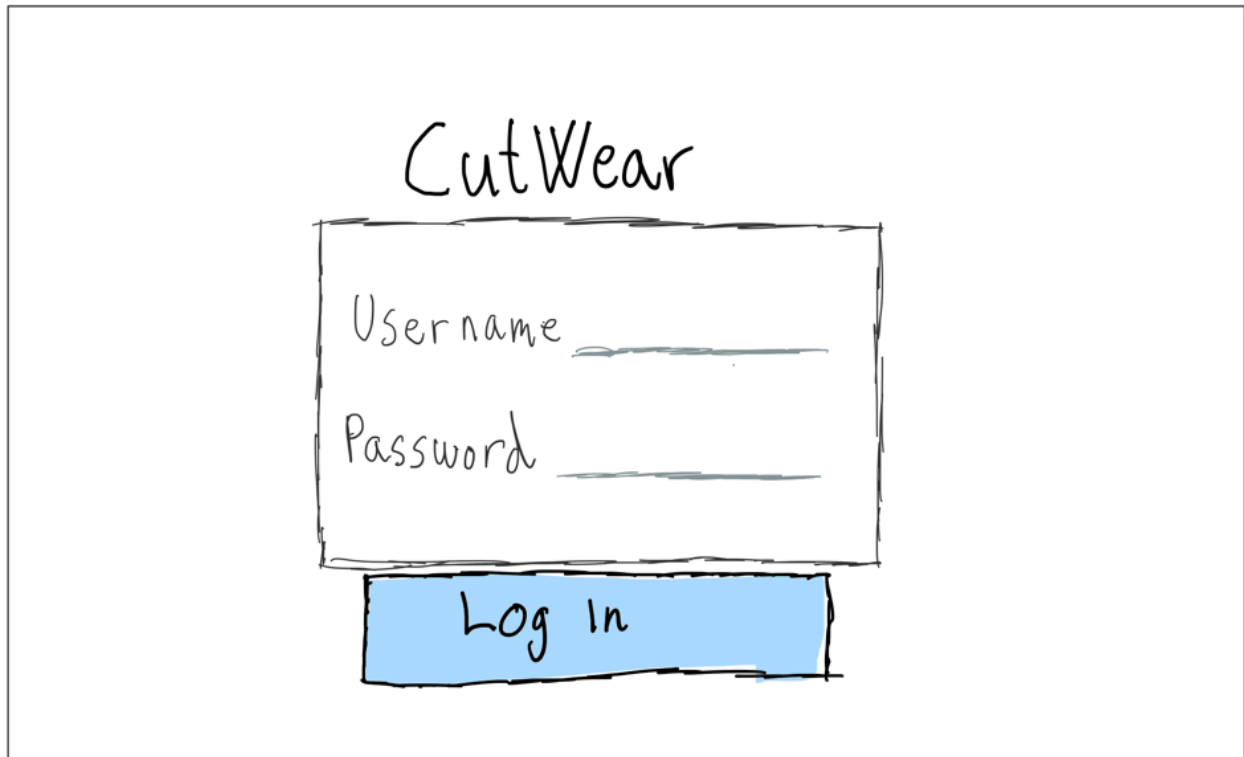
Denne prototypen ble utarbeidet i perioden 25.03.2019-27.03.2019. Den ble laget basert på tilbakemeldingene fra møte «sprintreview 4, m/ Oppdragsgiver», samt diskusjon innad i utviklingsteamet. Prototypen går i dybden på rollen TBM Inspector, som er hovedbrukeren for systemet.

#### Oversikt



## Individuelle sider

1.0



A hand-drawn sketch of a login form. At the top, the text "CutWear" is written in a large, casual font. Below this, there is a rectangular box containing two input fields. The first field is labeled "Username" and the second is labeled "Password", both followed by horizontal lines representing text input. Below the input fields is a blue rectangular button with the text "Log In" written on it.

Her har vi gått tilbake til at når man logger seg på må man først velge en ønsket TBM. Dette er gjort fordi vi har lagt til funksjonalitet for å kunne gå inn i en allerede påbegynt inspeksjon. For dette er det mest hensiktsmessig at man må velge TBM på starten av inspeksjoner.



## Select TBM

|             |
|-------------|
| 000107001   |
| 00:01070 10 |
| 0001 100 03 |
| 0001 111 01 |
| 0001 020 02 |



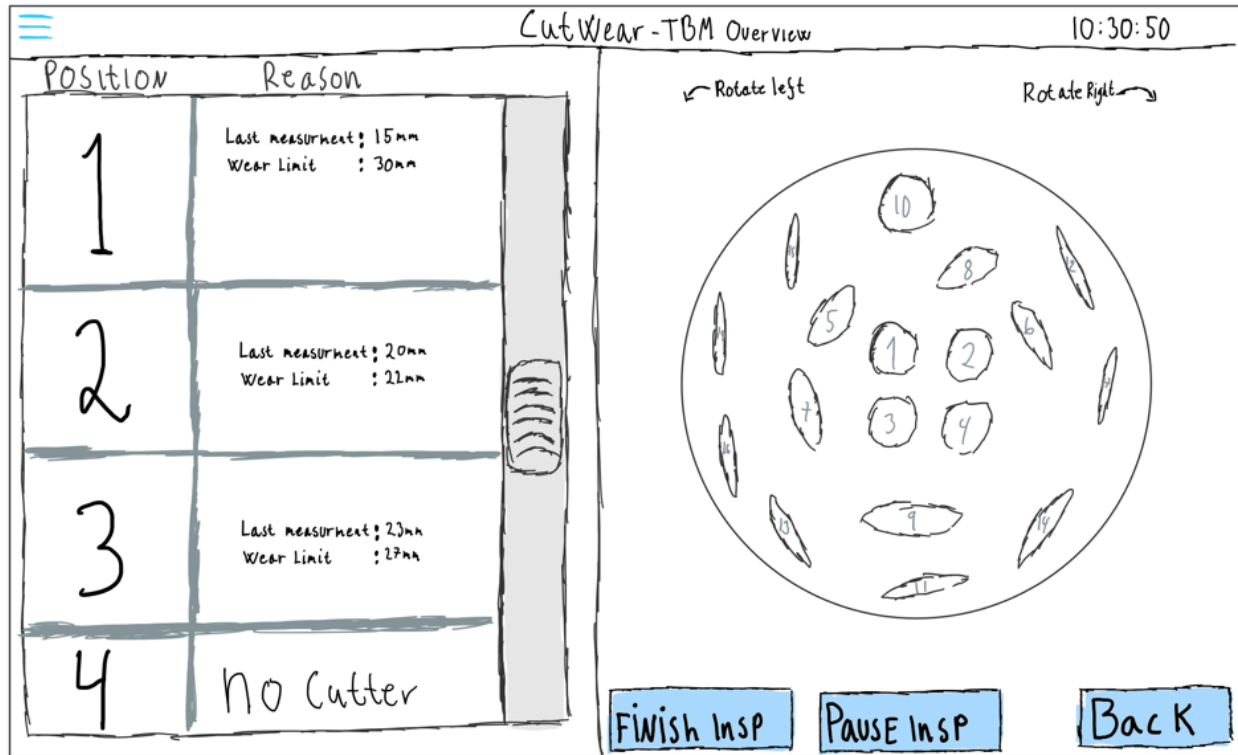
Om man går inn på en TBM som allerede har en påbegynt inspeksjon vil man komme til denne siden. Denne siden har som funksjon å gjøre TBM Inspektoren bevist på at man går inn i en allerede påbegynt inspeksjon. Om den TBM man velger derimot ikke har noe påbegynt inspeksjon vil man direkte komme til punkt 4.0.



TBM: 10-14-EW-02

|                |       |
|----------------|-------|
| Chamage :      | _____ |
| Machine Hours: | _____ |
| Bored Meters:  | _____ |
| Bored Hours :  | _____ |

next



Her trykker man seg inn på den posisjonen man ønsker å registrere kutterslitasje eller bytte. Her har gruppen gått bort i fra fargekode for nå, da det ikke virker hensiktsmessig da alle posisjoner uansett sjekkes. Derimot er det satt inn sist måling og maks slitasje-grense for posisjonen.

Finish insp knappen vil avslutte inspeksjonen og logge brukeren ut. Her vil det komme opp en alert for sikkerhetsskyld.

Pause insp knappen vil sette inspeksjonen på pause slik at den kan tas opp igjen når noen logger seg på og velger denne TBM'en. Det vil komme opp en alert for sikkerhetsskyld.

På høyre side kan man laste opp en figur, her kan man alltid gå tilbake og laste opp et annet bilde om det er ønskelig. Figuren kan også roteres ved hjelp av knappene øverst til høyre.



# Position 1

Register Cutter wear

Register Cutter change

## Information

Cutter ID : A732

### Previous measurements

10.03.19-10:01:33, 10mm

11.03.19-08:31:00, 12mm

13.03.19-15:03:10, 16mm

Maximum wear: 20mm

Back

☰

CutWear - Cutter Wear 10:35:47

## Position 1

Cutter wear:

17 mm

[Save](#)

Information

Previous measurements

- 10.03.19-10:01:33, 10mm
- 11.03.19-08:31:00, 12mm
- 13.03.19-15:03:10, 16mm

Maximum wear: 20mm

[Back](#)



☰

CutWear - Cutter Change 10:35:50

Position 1

Current cutter ID: A732

Category

new cutter-ID:

Comment:

## Toolbar

|                   | CutWear - Guidelines <span style="float: right;">10:30:41</span> |
|-------------------|------------------------------------------------------------------|
| Change TBM        |                                                                  |
| Guidelines        |                                                                  |
| Log out           |                                                                  |
| Delete Inspection |                                                                  |
|                   |                                                                  |

Dette er toolbaren for applikasjonen. Her finner man Change TBM, Guildelines, Log out og delete inspection. Change TBM sender brukeren ut av inspeksjonen og tilbake til side 2.0. Inspeksjonen settes på pause. Guildelines er en hjelpeside, denne kommer under. Log out, her logges brukeren ut av applikasjonen. Her vil inspeksjonen settes på pause. Delete inspection, dette er den eneste måten å slette en inspeksjon på.

## Guidelines

Denne siden er scrollbar, så de tre figurene som følger er en og samme side.

☰

### CutWear - Guidelines

10:30:41

---

1. Codes:

Abrasive wear : Cutter changed, due to maximum wear. Cutter moved to workshop.

Relocate : If the cutter is taken out to wait for a suitable position to be placed in.

Blocked : Blocked. Cutter moved to workshop

Chipping : Change cutter due to chipping. Cutter moved to workshop.

Mushrooming : EXCESSIVE mushrooming. Cutter moved to workshop.

Damaged hub : The hub is damaged. Cutter moved to workshop.

oil leakage : oil leakage. Cutter moved to workshop.

Ring crack : crack in the ring. cutter moved to workshop.

Other : Use the comment field to comment the remark. Cutter moved to workshop. ↓↓

☰

### CutWear - Guidelines

10:30:41

---

2. Instruction

Her legges delene i kjøppålitens bruksmanual

↓↓



### 3. Cutter Changing guidelines

PDF



# D Korrespondanse og kontrakter

## D1 Korrespondanse, ord fra oppdragsgiver



Amund Bruland  
fr. 10.05.2019 12:47



Knut Wiig: [redacted]

Et forsøk:

«Som oppdragsgiver er vi godt fornøyd med prosessen og resultatet av CutWear-oppgaven. Alle møter vi har hatt med gruppen har vært produktive og nyttige for oss og sluttproduktet. Fra vårt ståsted ser det ut som gruppen har løst oppgaven på en god måte med hensyn til planlegging, oppgavefordeling og kommunikasjon med oss.»

Mvh Amund

...

---

**From:** Knut Wiig <knutwii@stud.ntnu.no>

**Sent:** Friday, May 10, 2019 9:54 AM

**To:** Amund Bruland <amund.bruland@ntnu.no>

**Subject:** Tilbakemelding bacheloroppgaven.

Hei!

Vi trenger en liten tilbakemelding som vi skal sette inn i hovedrapporten vår om hvordan dere føler samarbeidet, applikasjonen og prosjektet som en helhet har vært. Fint om dere alle tre kunne skrevet et par setninger om dette. Vi setter pris på både konstruktive tilbakemeldinger og eventuell skryt:)

Hadde vært fint med tilbakemeldingene ila uken.

Takk på forhånd.

Med vennlig hilsen

Knut Wiig  
tlf. 97890632

Med vennlig hilsen

Knut Wiig  
tlf. 97890632

## Avtale mellom partene

Bedrift/virksomhet: NTNU Bygg og Miljø og The Robtins company

Student(ene): Mats Davidsen, Knut Wiig og Kristian N Lund

og

NTNU, IDI AIT

Studentprosjekt 29, Applikasjon for tunnelboring i hardt berg

### Gjennomføring

Studenten skal gjennomføre et studentprosjekt i samarbeid med bedriften/virksomheten. IDI AIT veileder arbeidet faglig. Det er utarbeidet retningslinjer for gjennomføring av studentprosjekt som beskriver oppgavefordeling og hvordan studentprosjekter gjennomføres. Retningslinjene tar også opp ansvarsfraskrivelse, opphavsrettigheter og tilgjengelighet med muligheter for individuelle avtaler.

### Ansvarsfraskrivelse

Instituttet er ikke ansvarlig for eventuelle ødeleggelse som studenten måtte påføre oppgavestillers utstyr direkte eller som følge av programvare studenten lager og/eller bruker, eller som studenten på annen måte medvirker til.

### Opphavsrett og tilgjengelighet

Når ikke annet er avtalt, eier studenter selv den IPR (immaterielle rettigheter) de skaper som en del av studier/studieopphold ved IDI AIT. Alle resultater er åpent tilgjengelig. Opphavsretten reguleres av Åndsverksloven. Avtaler som inngås mellom IDI AIT og studenter skal som minimum sikre instituttet rett til å bruke generert IPR til utdannings- og forskningsformål. IDI AIT skal også motta en vurderingskopi av arbeidet inkludert eventuell kildekode.

Marker med kryss det som gjelder denne oppgaven:

- Normalsituasjonen: Studentene har selv alle rettigheter knyttet til resultatet fra bacheloroppgaven, med de unntak som er beskrevet over.
- Oppdragsgiveren har rettighetene og kan utnytte produktet kommersielt og videreutvikle produktet/metoden. Instituttet vil ikke utnytte produktet kommersielt, men vil kunne arbeide videre med den grunnlagskompetansen som er vunnet gjennom prosjektet, som beskrevet over.
- Resultatene fra arbeidet legges ut som OpenSource iht lisens \_\_\_\_\_ (Se <http://creativecommons.no/lisenser>).
- Bacheloroppgaven (det skriftlige arbeidet) skal være undergitt utsatt offentliggjøring i \_\_\_\_ (maks 3) år.


Oppdragsgiver er selv ansvarlig for å avtale håndtering av eventuelle konfidensielle opplysninger med veileder/sensor og studenten(e).

Denne avtalen er underskrevet i 3 – tre - eksemplarer hvor partene skal ha hver sin.

08.02.2018  
(dato, sted)

  
Bedrift/virksomhet

  
Veileder ved IDI AIT

  
Student(ene)

## Kontrakt – levering av kildekode.

Sted: NTNU Kalvskinnet, Trondheim.

Utviklingsteam: Mats Davidsen, Kristian Nybakken Lund og Knut Wiig.

Veileder: Helge Hafting.

Partene (utviklingsteam og veileder) er enige om at kildekoden skal levers separat fra hovedoppgaven, i form av en lenke til «Repository». Veileder vil få tilgang til en GitHub-bruker som har tilgang til kildekoden til CutWear i GitHub. Brukernavnet, passordet og lenke til kildekoden vil bli sendt til veileder(via epost) før tidsfristen (20.mai 2019 kl.23:59) for innlevering av bacheloroppgaven.

16.05.2019

(dato)

Helge Hafting

(Veileder, Helge Hafting)

Knut Wiig, Mats Davidsen, K Lund

(Utviklingsteam)

## D4 Korrespondanse, Innlogging til Github

Levering av kildekode, bacheloroppgave 29



Mats Elias Davidsen

ma. 20.05.2019 09:32

Til: Helge Hafting

Kopi: Knut Wiig; Kristian Nybakken Lund; Mats Elias Davidsen ↕

Kildekoden til vår bacheloroppgave ligger i to repositories, et for server og et for klient. Begge disse har en release v1.0. For å få tilgang til kildekoden logg inn på <https://github.com/> med følgende bruker:

Brukernavn: [redacted]

Passord: [redacted]

Repositoryene heter cutWearServer og cutWearClient, disse kan finnes ved å følge lenkene:

Server: <https://github.com/cutwear/cutWearServer>

Klient: <https://github.com/cutwear/cutWearClient>

Send oss en mail hvis du har spørsmål angående hvor du finner koden.





## E Visjonsdokument

# Visjonsdokument Bachelor - Gruppe 29.

Daidsen, Mats Elias

Lund, Kristian Nybakken

Wiig, Knut

20. mai 2019

# Innhold

|                                                             |           |
|-------------------------------------------------------------|-----------|
| Figur og tabelliste                                         | ii        |
| <b>1 Innledning</b>                                         | <b>1</b>  |
| <b>2 Sammendrag problem og produkt</b>                      | <b>1</b>  |
| 2.1 Problemsammendrag                                       | 1         |
| 2.2 Produktsammendrag                                       | 1         |
| <b>3 Overordnet beskrivelse av interresenter og brukere</b> | <b>2</b>  |
| 3.1 Oppsummering interessenter                              | 2         |
| 3.2 Oppsummering brukere                                    | 3         |
| 3.3 Brukermiljø                                             | 4         |
| 3.4 Sammendrag av brukeres behov                            | 4         |
| 3.5 Alternativer til vårt produkt                           | 6         |
| 3.5.1 Eksisterende produkt                                  | 6         |
| 3.5.2 Office 365                                            | 6         |
| 3.5.3 Wufoo                                                 | 6         |
| <b>4 Produktoversikt</b>                                    | <b>7</b>  |
| 4.1 Produktets rolle i brukermiljø                          | 7         |
| 4.2 Forutsetninger og avhengigheter                         | 7         |
| <b>5 Produktets funksjonelle egenskaper</b>                 | <b>9</b>  |
| 5.1 User Stories                                            | 9         |
| 5.2 Sikker overføring av data                               | 12        |
| <b>6 Ikke-funksjonelle egenskaper og andre krav</b>         | <b>13</b> |
| 6.1 Hosting av kildekode                                    | 13        |
| 6.2 Database                                                | 13        |
| 6.3 Testing                                                 | 13        |
| 6.4 Dokumentasjon                                           | 13        |
| 6.5 Ytelse                                                  | 13        |
| 6.6 Logging                                                 | 13        |
| 6.7 Skalerbarhet                                            | 13        |
| 6.8 Språk                                                   | 13        |
| 6.9 Installering og kjøremiljø                              | 13        |
| 6.10 Hardware                                               | 14        |
| <b>Referanser</b>                                           | <b>15</b> |

## Figur og tabelliste

### Figurer

|   |                               |   |
|---|-------------------------------|---|
| 1 | Bilde av brukermiljø. . . . . | 7 |
|---|-------------------------------|---|

### Tabeller

|   |                                          |   |
|---|------------------------------------------|---|
| 1 | Oppsummering av interessentene . . . . . | 2 |
| 2 | Oppsummering av brukere . . . . .        | 4 |
| 3 | Sammendrag av brukeres behov . . . . .   | 6 |

# 1 Innledning

Hensikten med dette visjonsdokumentet er å kartlegge nåværende situasjon ved registrering av kutterslitasje i tunnelboringsmaskiner, og få en oversikt over interessenter og problemdomenet som kan brukes til å utvikle en bedre og mer effektiv løsning.

## 2 Sammendrag problem og produkt

### 2.1 Problemsammendrag

Det er flere **Problem med** dagens system. For det første skjer arbeidsprosessen manuelt, ved bruk av penn og papir. Her går to inspektører inn i borrehode hvor en måler slitasjene mens den andre noteter på et ark. Deretter blir denne informasjonen senere skrevet inn på en datamaskin for så og bli sendt videre til en tredje person. Denne vil skrive informasjonen inn på et excel-ark som fungerer som en database. Et annet problem med dagens løsning er at inspektørene ikke får noe form for beslutningstøtte fra tidligere data. Det tredje og siste problemet er at det er uoversiktlig og vanskelig for ledelsen, enten på prosjektet eller for Robbins, å bruke disse data til videre analyser og prognoser.

Dette **berører** inspektørene, prosjektledere, produktansvarlige og tredjeparts interessenter som vil ha tilgang til data, som for eksempel NTNU bygg- og miljøteknikk.

**Som resultatet av dette** er registreringen ineffektiv, da det kreves to mann for å utføre en kutterinspeksjon og det tar lang tid fra inspeksjonen utføres til dataen ligger i databasen. Disse mengdene data som blir lagret brukes ikke optimalt. Det ligger en enorm verdi i å benytte seg maksimalt av den data man opparbeider seg.

**En vellykket løsning vil** minimere inspeksjonstiden (fra registreringa av slitasje og bytte, til lagring i databasen), samt gi bedre oversikt over behov og mangler, og gi ledelsen mulighet til å benytte seg av data til å hele tiden forbedre prosesser.

### 2.2 Produktsammendrag

**For** Robbins Company

**som** har behov for å digitalisere og effektivisere prosessen ved registrering av kutterslitasje for sine tunnelborremaskiner.

**Produktet navngitt** er en applikasjon som kjøres på et industri-nettbrett inne i selve TBM'en

**som** først og fremst skal effektivisere hele prosessen rundt registrering av slitasje og evalueringen av innsamlet data.

**I motsetning til** dagens system hvor alt skjer ved penn og papir,

**har vårt produkt** realtime oppdatering av behandlet data.

### 3 Overordnet beskrivelse av interresenter og brukere

#### 3.1 Oppsummering interessenter

| Interessenter     |                                                                                                                                                                            |                                                                                                                                                                                                                                                                                                   |
|-------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Navn              | Utdypende beskrivelse                                                                                                                                                      | Rolle under utvikling                                                                                                                                                                                                                                                                             |
| Eier              | Dette er Robbins Company og NTNU bygg- og miljøteknikk. De har rettigheter og kan utnytte produktet kommersielt og videreutvikle systemet.                                 | Ansvarlig for budsjettering                                                                                                                                                                                                                                                                       |
| Kunderepresentant | Sindre Log, Product Manager for cutters and tools, representerer Robbins Company i dette prosjektet.                                                                       | Sindre Log står ansvarlig for produktbeskrivelsen og har det siste ordet.                                                                                                                                                                                                                         |
| Oppdragsgiver     | Oppdragsgiveren for denne Bacheloroppgaven er Instituttet for bygg- og miljøteknikk NTNU, representert ved Amund Bruland.                                                  | Tett kontakt med eier, skal også stille med veiledning til utviklingsteamet når det trengs.                                                                                                                                                                                                       |
| Utviklingsteam    | Utviklingsteamet for denne Bacheloroppgaven er Knut Wiig, Kristian Nybakken Lund og Mats Elias Davidsen.                                                                   | Utviklingsteamet har ansvaret for å levere et produkt i henhold til produktbeskrivelsen alle partene har kommet til enighet om. De har også selv ansvaret for framdriften av prosjektet. I tillegg til dette skal de levere bestemte dokumenter til NTNU ved ferdigstillelse av Bacheloroppgaven. |
| Teknisk support   | Robbins Company har et utviklingsteam i Seattle, USA, som via Skype kan veilede utviklingsteamet. Utviklingsteamet som sitter i Seattle er representert ved Nitin Malhotra | Nitin Malhotra skal komme med nødvendig teknisk veiledning angående den interne IT-policyen til Robbins Company, samt hvordan deres PLC- system fungerer.                                                                                                                                         |
| Veileder          | Representert ved Helge Hafting, som er veileder fra NTNU.                                                                                                                  | Veileder skal kontrollere timelister, veilede med skriving av hovedrapport og gi karakter på prosjektet.                                                                                                                                                                                          |
| Domeneekspert     | Ekspert på domenet som oppgaven baserer seg på. Domeneekspert i dette prosjektet er representert av oppdragsgiver og kunde.                                                | Være tilgjengelig for spørsmål og fungerer som fagfor utviklingsteamet.                                                                                                                                                                                                                           |
| Testansvarlig     | Testansvarlig er Knut Wiig, Kristian Nybakken Lund, Mats Elias Davidsen                                                                                                    | Det skal gjøres en rekke forskjellige tester gjennom hele prosjektet. Her står alle tre studentene ansvarlig for at dette blir utført.                                                                                                                                                            |
| Testgruppe        | Testgruppen for dette prosjektet blir interressenter som har mye kunnskap om prosessen.                                                                                    | Teste low- og highfidelity prototyper fortløpende gjennom utviklingen, samt teste det ferdige produktet i slutten av prosjektet.                                                                                                                                                                  |

Tabell 1: Oppsummering av interessentene

### 3.2 Oppsummering brukere

| Brukere              |                                                                                                                                                                                                                                                                                                                                                                 |                                                                                                                                                                                      |                                                                                     |
|----------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------|
| Navn                 | Utdypende Beskrivelser                                                                                                                                                                                                                                                                                                                                          | Rolle under utviklingen                                                                                                                                                              | Representert av                                                                     |
| Inspektør            | Inspektøren er ansvarlig for å registrere data ved hjelp av nettbrett på arbeidsplassen der TBM benyttes.                                                                                                                                                                                                                                                       | Minimal da disse er vanskelig å få tak med tanke på at de er spredt over hele verden og ingen i umiddelbar nærhet per dags dato, men håper å få snakket med noen under besøk av TBM. | N/A                                                                                 |
| Administrator        | Overordnet bruker med tilgang til å redigere det meste, skal også få en oversikt over hele systemet.                                                                                                                                                                                                                                                            | Fungerer som domeneekspert og kunde.                                                                                                                                                 | Sindre Log fra The Robbins Company                                                  |
| Workshop-Supervisor  | Dette er en bruker som driver med vedlikehold av kuttere når de kommer til verkstedet, brukeren vil ha tilgang til samtlige TBM'er som hører til prosjekt tilknyttet dette verkstedet. Workshop-Supervisor har skriverettigheter tilknyttet vedlikehold av kuttere, brukeren skal også kunne se all data som kan brukes som grunnlag for vedlikehold av kuttere | Får mest sannsynlig ikke mulighet til å innvolvere denne brukergruppen i utviklingen                                                                                                 | Ledelsen på verkstedene som utfører vedlikehold av kuttere.                         |
| Project-Supervisor   | Denne brukeren vil være tilstede på arbeidsplassen der et prosjekt utføres. Brukeren vil ha tilgang til data fra samtlige TBM'er som er knyttet til aktuelt prosjekt, brukeren har også noen skriverettigheter som han bruker til å administrere prosjektet.                                                                                                    | Får mest sannsynlig ikke mulighet til å innvolvere denne brukergruppen i utviklingen                                                                                                 | Ledelsen ved de ulike tunnelprosjektene.                                            |
| Super User           | Forskjellen mellom en Super User og en Project-Supervisor er at Super User har lesetilgang til samtlige prosjekter, mens en Project Supervisor kun har tilgang til sitt prosjekt. En Super User har kun lesetilgang og kan dermed ikke skrive data til databasen.                                                                                               | Brukergruppen vil ha tett kontakt med utviklingsteamet gjennom hele prosjektet i form av veiledning.                                                                                 | Ledelsen i Robbins Company og utvalgte ansatte på Institutt for bygg og miljø NTNU. |
| System-Administrator | Administrerer og drifter systemet.                                                                                                                                                                                                                                                                                                                              | Bidrar til dokumentasjon og hvordan man distribuerer systemet.                                                                                                                       | Utviklere av prosjektet og serveransvarlig på hostingtjenesten.                     |

Tabell 2: Oppsummering av brukere

### 3.3 Brukermiljø

Systemets hovedbruksområde er inspeksjon av TBM'er og skal brukes i harde og lysfattige miljøer, nærmere bestemt under konstruksjon av tunneler. Miljøet er preget av mye støv, møkk og høy varme under inspeksjoner. Systemet brukes mellom borer for å registrere slitasje på kuttere i borehodet. Under inspeksjoner brukes det diverse verneutstyr som for eksempel hansker, dette er ikke særlig touchvennlig og må tas hensyn til under utviklingen.

De andre brukermiljøene kan være kontor (on- og offsite), dette kan beskrives som normale brukermiljøer, derfor trenger vi ikke ta samme hensyn ved utvikling av UI.

### 3.4 Sammendrag av brukeres behov

| Brukerens behov                                     |           |                                   |                                                                                                                                                                                                                                                |                                                                                                                                                        |
|-----------------------------------------------------|-----------|-----------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------|
| Behov                                               | Prioritet | Påvirker                          | Dagens løsning                                                                                                                                                                                                                                 | Foreslått løsning                                                                                                                                      |
| Touchvennlig grensesnitt                            | Høy       | Alle, men i hovedsak inspektørene | Tungvindt løsning med bruk av ark og mengder med excel-ark                                                                                                                                                                                     | Samle alle prosessene under en applikasjon, hvor både grensesnittet og tablet'en er tilrettelagt krevende miljø.                                       |
| Registrere kutterslitasje                           | Høy       | Inspektør                         | Dette er en tomanns jobb hvor en inspektør måler slitasje og en annen skriver ned målingene på et vanlig ark. Deretter skriver de inn målingene på datamaskinen, sender dette til Sindre Log, som plotter dette inn i et overordnet Excel ark. | Bruk av en tablet skal redusere dette til en enmanns jobb. Her skal personen skrive målingene rett inn i systemet som laster data opp til en database. |
| Få en indikasjon på kritiske kuttere                | Medium    | Inspektør                         | Ingen.                                                                                                                                                                                                                                         | Inspektøren vil få opp et grafisk grensesnitt som viser hvilke kuttere som har status 'Kritisk' under selve inspeksjonen.                              |
| Dele data internt i bedriften                       | Høy       | Alle                              | Ingen god løsning på dette.                                                                                                                                                                                                                    | En felles database hvor all data blir lagret.                                                                                                          |
| Kunne gjøre inspeksjoner uten tilgang til internett | Høy       | Inspektør                         | Trenger ikke internett for å gjøre skrive ned målingene.                                                                                                                                                                                       | Data vil bli lagret lokalt på tablet'en om internett ikke er tilgjengelig, for så å lastes opp når man får tilbake internett.                          |



|                                                                                             |        |                                |                                                                                            |                                                                                                                                         |
|---------------------------------------------------------------------------------------------|--------|--------------------------------|--------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------|
| Registrere tidspunkt for inspeksjoner                                                       | Høy    | Inspektør                      | Skriver ned dato på ark.                                                                   | Automatisk timestamp.                                                                                                                   |
| Registrere grunnen for kutterbytte                                                          | Høy    | Inspektør                      | Skriver ned på ark, bruker gitte kategorier.                                               | Velger mellom gitte kategorier i systemet, samt en mulighet for å legge ved en kommentar.                                               |
| Registrere reallokering av kuttere                                                          | Medium | Inspektør                      | Skriver ned den nye posisjonen til kutteren i samme felt hvor man vanligvis har slitasjen. | Velger kategorien 'Relocated' når man registrerer grunnen for kutterbytte. Hver posisjon vil vise hvilke kutter-ID som er i posisjonen. |
| Oversikt over hvilke kutter som befinner seg i hvilke posisjon til en hver tid              | Medium | Inspektør                      | Ingen                                                                                      | Systemet registrerer hvilke kutter som sitter i hvilke posisjon                                                                         |
| Registrere 'Chainage', 'Machine Hours', 'Bored Meters' og 'Bored Hours' ved hver inspeksjon | Medium | Inspektør                      | Leser av data fra PLC systemet og skriver dette ned på ark                                 | Lese av data fra PLC systemt og skrive dette inn i systemet.                                                                            |
| Registrere hvilke skift som gjør inspeksjonen                                               | Lav    | Project Supervisor, Super User | Ingen.                                                                                     | Når en inspektør logger inn velger personen skift.                                                                                      |
| Prosjektleder skal kunne se all data tilgjengelig for dem                                   | Høy    | Project Supervisor             | Lese av dokumentene de har laget ut i fra arkene.                                          | Velger en TBM og deretter få en grafisk oversikt over data                                                                              |
| Super Users skal kunne se all data tilgjengelig for dem                                     | Høy    | Super User                     | Ser på Excel-filen som Sindre Log administrerer.                                           | Logger seg inn som Super user, velger ønsket prosjekt og deretter TBM og får en grafisk oversikt over data.                             |
| En prosjektleder skal kunne sette opp en ny TBM for sitt prosjekt                           | Høy    | Project Supervisor             | Lager bare ny Excel-fil / ark                                                              | Registrerer ny TBM via applikasjonen.                                                                                                   |
| En prosjektleder skal kunne sette opp nye skift for sitt prosjekt                           | Lav    | Project Supervisor             | Gjøres ikke.                                                                               | Registrerer nytt skift via applikasjonen.                                                                                               |
| En Administrator skal kunne registrere en ny TBM modell for bruk i hele systemet            | Høy    | Admin                          | Trengs ikke, da alt av informasjon skrives ned på ark.                                     | Admin registrerer en ny TBM modell via systemet som Project Supervisor senere kan velge som en mulig TBM.                               |

|                                                                               |        |                                 |                                                                 |                                                                                                                                |
|-------------------------------------------------------------------------------|--------|---------------------------------|-----------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------|
| Se historien til kuttere                                                      | Høy    | Workshop Supervisor, Super User | Se Excel-ark, begrenset med informasjon.                        | Systemet skal lagre mer omfattende informasjon om hver kutter, som blir lett tilgjengelig.                                     |
| Se om en kutter er på en TBM, transport, lager eller verksted til en hver tid | Lav    | Project Supervisor.             | Ingen.                                                          | Lagre status for hvor en kutter befinner seg og sende denne informasjonen til databasen.                                       |
| Registrere vedlikehold som ble gjort på kuttere på verkstedet                 | Medium | Inspektør                       | Dette gjøres kun internt på verksted                            | En Workshop Supervisor går inn å sier hva som ble gjort på de forskjellige kutterne, dette blir så brukt til videre prognoser. |
| Registrere nye kuttere                                                        | Høy    | Inspektør, Project Supervisor   | Trengs ikke da det bare er å skrive inn nytt kutter- ID på ark. | TBM Inspector og Project Supervisor skal kunne registrere nye kuttere til databasen                                            |
| Registrere nytt prosjekt                                                      | Høy    | Project Supervisor              | Ikke nødvendig, da de bruker ark.                               | Registrere nytt prosjekt via applikasjonen.                                                                                    |
| Lese manualer for å få nødvendig informasjon om prosessene                    | Høy    | Alle                            | Ser på tilgjengelig dokumentasjon                               | Egen siden for instruksjoner knyttet til hvilke bruker som er logget inn.                                                      |

Tabell 3: Sammendrag av brukeres behov

### 3.5 Alternativer til vårt produkt

#### 3.5.1 Eksisterende produkt

Et alternativ er å beholde nåværende system. Dette vil bli den minst ressurskrevende løsningen. En stor utfordring ved implementering av et nytt produkt vil være endringsledelse mot interessentene som blir påvirket. Beholde det eksisterende produktet vil unngå dette. Se kapittel 2 for et sammendrag av nåværende produkt.

#### 3.5.2 Office 365

En annen alternativ løsning er å benytte seg av Office 365. Her har man samskriving og lagring via skyen. Denne løsningen kan gi problemer ved offlineaktivitet over lang tid. Office 365 vil ikke kunne tilby skreddersydd funksjonalitet som Robbins Company trenger. Dette er en lite ressurskrevende løsningen, men vil ikke skape mye merverdi. [1]

#### 3.5.3 Wufoo

Denne løsningen ligner på Office 365, hvor det man skriver blir lagret i skyen. Her kan man også lage egne skjemaer og distribuere dem, men applikasjon har en abonnementskostnad. [2]

## 4 Produktoversikt

### 4.1 Produktets rolle i brukermiljø



Figur 1: Bilde av brukermiljø.

### 4.2 Forutsetninger og avhengigheter

Det er flere viktige forutsetninger for at vi skal kunne gjennomføre dette prosjektet, en av disse er at vi har tilgang til NTNU sine servere for å kjøre databasen. Det er viktig at vi har tilgang til disse serverene når vi skal begynne å jobbe med produktet etter fullført planleggingsfase, dette er for å kunne teste funksjonalitet i løpet av utviklingsfasen.

Siden vi har bestemt oss for å lage en applikasjon til nettbrett som bruker Windows 10 som operativsystem vil det være en forutsetning at oppgavestiller ikke bytter til en nettbrett som kjører et annet operativsystem. Hvis oppgavestiller bytter operativsystem vil dette kunne føre til at vår applikasjon, som er beregnet for Windows 10 ikke vil kunne kjøre.

Det er også nødvendig at vi har et miljø der vi kan teste ut applikasjonen, det er ideelt at dette er en nettbrett av lik modell som den som skal brukes når man inspiserer en TBM. Det er også en mulighet å kjøre dette ved hjelp av andre alternativer som for eksempel datamaskiner med touch-funksjonalitet eller andre touchpadder som kjører Windows 10, da dette er operativsystemet som den modellen Robbins Company planlegger å kjøpe inn bruker.

Muligheten til å ta kontakt med domeneeksperter er også en viktig forutsetning for gjennomførelsen av denne oppgaven, fordi vi som datastudenter ikke har tilstrekkelig kunnskap om TBM eller arbeidsprosessen rundt bytting av kuttere.

## 5 Produktets funksjonelle egenskaper

Liste over funksjonelle egenskaper som produktet må ha for å løse problemene beskrevet i 3.4 er definert ved bruk av user stories.

### 5.1 User Stories

#### *User story 1.*

As an TBM Inspector

**I want to** Register the cutter wear of the selected cutter into a database

**So that** we can use it for further inspections

#### *User story 2.*

As an TBM Inspector

**I want to** get an indication of critical cutter wear

**So that** I can easily change cutters

#### *User story 3.*

As an TBM Inspector

**I want to** have a simple, touch-friendly and efficient user interface

**So that** I can do the cutter operations quickly with no hassle

#### *User story 4.*

As an TBM Inspector

**I want to** be able to send data to a database

**So that** the information is correctly stored

#### *User story 5.*

As an TBM Inspector

**I want to** submit previously failed uploads

**So that** the data in the DB is up to date.

#### *User story 6.*

As an TBM Inspector

**I want to** I want to scan the serial number on the cutter

**So that** I can use minimal time to inspect cutters

#### *User story 7.*

As an TBM Inspector

**I want to** register a change of a cutter

**So that** the cutter positions is up to date

#### *User story 8.*

As a TBM Inspector

**I want to** register Chainage, Machine Hours, Bored Meter, Bored Hours

**So that** this information is stored in the database to its specific intervention.

#### *User story 9.*

As a TBM Inspector

**I want to** choose which TBM in this project I am going to inspect

**So that** the data i register is connected to the correct TBM

***User story 10.***

As a Project Supervisor

**I want to** view the data from all TBM's connected to this project

**So that** I can review and process the data gathered

***User story 11.***

As a Project Supervisor

**I want to** register a TBM for a project

**So that** I can use it for the project

***User story 12.***

As a Project Supervisor

**I want to** register a shifts for a project

**So that** the TBM Inspectors can log their inspections correctly

***User story 13.***

As a Super User

**I want to** view all history of an given project

**So that** I can use this information for further analysis.

***User story 14.***

As a Super User, Project Supervisor

**I want to** view the history of a given cutter

**So that** I can use this information later.

***User story 15.***

As a Workshop Supervisor

**I want to** register the change done to the cutters in the workshop

**So that** the data can be used in later prognoses for cutter changes.

***User story 16.***

As a Workshop Supervisor

**I want to** get history of a given cutter

**So that** i can use this information to determine what maintenance is required.

***User story 17.***

As a Admin, Workshop, Project Supervisor, TBM Inspector

**I want to** register a new cutter

**So that** it gets stored in the database

***User story 17.***

As a Admin

**I want to** register a new project

**So that** the customer will have a project registered in our system and a system administrator that belongs to this project

***User story 18.***

As a Admin

**I want to** register a new type of TBM

**So that** the users in this software can select this type of TBM

***User story 19.***

**As a System Admin**  
**I want to** have a database with user accounts  
**So that** I can handle access to the system

*User story 20.*

**As an Admin**  
**I want to** have an view of all TBM's who have been managed with this system  
**So that** I can manage the different TBM's

*User story 21.*

**As a Super User / admin**  
**I want to** query the REST-API  
**So that** I can get the data in a JSON file.

*User story 22.*

**As a User**  
**I want to** get the information- and instruction page for the given TBM  
**So that** the information is available for me.

*User story 23.*

**As a User**  
**I want to** view the data from the database  
**So that** I can review and process the data gathered.

*User story 24.*

**As a User**  
**I want to** read the user manual to be able to install and use the application  
**So that** I can register, view and process cutterdata

*User story 25*

**As a User**  
**I want to** logon  
**So that** I can do my job

*User story 26.*

**As a User**  
**I want** my data to be handled correctly and securely  
**So that** that all stored data is correct.

*User story 27.*

**As a User**  
**I want** the application to be responsive under heavy load  
**So that** the user experience is smooth

*User story 28.*

**As a System Administrator**  
**I want to** be able to start new projects

**So that** we can do our job

*User story 29.*

**As a** TBM Inspector

**I want to** make a comment on an inspection

**So that** if something of importance happened during the inspection, I can log it.

## 5.2 Sikker overføring av data

TLS/SSL - Kommunikasjon mellom klient og server skal være ende-til-ende kryptert med TLS 3.0.

Hashing av passord - Passord og annen sensitiv informasjon skal krypteres/hasjes før det lagres i databasen.

Ingen passord skal håndteres i klartekst.



## 6 Ikke-funksjonelle egenskaper og andre krav

### 6.1 Hosting av kildekode

Kildekoden skal hostes på Github.com ved hjelp av et eller flere private repositories. Disse repositoryene skal dokumenteres i systemdokumentasjonen.

### 6.2 Database

Transaksjoner - For å opprettholde dataintegriteten.

Databasepool - Brukes for å få best mulig databaseytelse. SQL-Injection - Det skal ikke være mulig å utføre SQL-injections mot databasen.

### 6.3 Testing

Enhetstesting - Alle funksjoner/metoder skal enhetstestes. Ønsket kodedekning er 80%.

Systemet skal stresstestes, regresjonstestes, akseptansetestes og testes for usability.

### 6.4 Dokumentasjon

Kravdokumentasjon skal være tilgjengelig for kunden etter overlevering av produkt. Systemdokumentasjon skal dokumentere alle krav for installering og kjøring av server, denne dokumentasjonen skal medfølge produktet. Brukermanual skal følge applikasjonen og skal vise hvordan man bruker og installerer applikasjonen.

### 6.5 Ytelse

Systemet har et krav på 99% oppetid i året.

### 6.6 Logging

Systemet skal logge alle brukeroprasjoner. Systemet skal logge alle feilhendelser. Systemet kan ha mulighet til å kjøres med verbose logging hvis ønsket.

### 6.7 Skalerbarhet

Systemet skal designes slik at det er skalerbart, både med tanke på ytelse og ved valg av datastrukturer med tilhørende navn.

### 6.8 Språk

Applikasjon og medfølgende dokumentasjon skal skrives på engelsk. Hovedrapport, visjonsdokument og kravdokumentasjon skal skrives på norsk.

### 6.9 Installering og kjøremiljø

Klientapplikasjonen skal kjøres på et Windows 10 nettbrett og skal leveres med instruksjoner på hvordan man installerer klientapplikasjonen.

Valg av operativsystemer for server er opp til utvikler. Det er ønskelig å bruke Docker for å muliggjøre enkel distribusjon og installering. Instruksjoner for installering vil være tilgjengelig i [github.com/cutterWear/README.md](https://github.com/cutterWear/README.md)

## 6.10 Hardware

Nettbrett for prototyping er av typen Panasonic TOUGHPAD FZ-M1, og dets tilhørende spesifikasjoner vil være tilnærmet optimale for klientapplikasjon. Se produktside for oversikt her.

## Referanser

1. Office, M. Office 365. URL: <https://www.office.com/>.
2. Wufoo. Wufoo. URL: <https://www.wufoo.com/features/>.

## F Systemdokumentasjon

# System documentation CutWear v1.0 - Group 29

Daidsen, Mats Elias

Lund, Kristian Nybakken

Wiig, Knut

20th of May 2019

# Innhold

|          |                                                                |           |
|----------|----------------------------------------------------------------|-----------|
| <b>1</b> | <b>Introduction</b>                                            | <b>4</b>  |
| <b>2</b> | <b>Architecture and Design Patterns</b>                        | <b>4</b>  |
| 2.1      | Technology Stack . . . . .                                     | 4         |
| 2.2      | Architecture . . . . .                                         | 5         |
| 2.3      | Model Route Controller . . . . .                               | 5         |
| 2.4      | Docker-Compose . . . . .                                       | 5         |
| 2.5      | Client . . . . .                                               | 7         |
| <b>3</b> | <b>Projectstructure</b>                                        | <b>8</b>  |
| 3.1      | Server . . . . .                                               | 8         |
| 3.1.1    | npm scripts . . . . .                                          | 8         |
| 3.1.2    | Docker commands . . . . .                                      | 9         |
| 3.1.3    | Docker-Compose commands . . . . .                              | 9         |
| 3.2      | Client . . . . .                                               | 10        |
| 3.2.1    | npm scripts . . . . .                                          | 12        |
| 3.2.2    | Docker scripts . . . . .                                       | 12        |
| <b>4</b> | <b>Database Model</b>                                          | <b>13</b> |
| <b>5</b> | <b>REST Services</b>                                           | <b>14</b> |
| <b>6</b> | <b>Security</b>                                                | <b>23</b> |
| 6.1      | Server . . . . .                                               | 23        |
| 6.1.1    | Hashing and storing passwords . . . . .                        | 23        |
| 6.1.2    | JSON Web Tokens . . . . .                                      | 23        |
| 6.1.3    | Let's Encrypt . . . . .                                        | 23        |
| 6.1.4    | SQL security . . . . .                                         | 23        |
| 6.1.5    | Protecting routes with middleware . . . . .                    | 24        |
| 6.1.6    | Heavy load scenarios . . . . .                                 | 25        |
| 6.2      | Client . . . . .                                               | 25        |
| 6.2.1    | Cross Site Scripting(XSS) . . . . .                            | 25        |
| 6.2.2    | Client-security . . . . .                                      | 25        |
| 6.2.3    | Timestamps . . . . .                                           | 25        |
| <b>7</b> | <b>Installation and running the application</b>                | <b>25</b> |
| 7.1      | Server . . . . .                                               | 25        |
| 7.1.1    | First time installation . . . . .                              | 25        |
| 7.1.2    | Running server outside Docker . . . . .                        | 26        |
| 7.1.3    | Running server with Docker . . . . .                           | 26        |
| 7.1.4    | Running server in production with Docker-Compose [1] . . . . . | 26        |
| 7.1.5    | Handling updates when in production . . . . .                  | 27        |
| 7.1.6    | Libraries . . . . .                                            | 27        |
| 7.2      | Client . . . . .                                               | 30        |
| 7.2.1    | Creating a installer without Docker . . . . .                  | 30        |
| 7.2.2    | Creating a installer with Docker . . . . .                     | 30        |
| 7.2.3    | Installing on Windows x64 PC . . . . .                         | 31        |
| 7.2.4    | Uninstalling the desktop application . . . . .                 | 31        |
| 7.2.5    | Development environment . . . . .                              | 31        |
| 7.2.6    | Libraries . . . . .                                            | 31        |
| 7.3      | Documentation of source code . . . . .                         | 34        |
| 7.4      | Continous Integration and testing . . . . .                    | 35        |
| 7.4.1    | Continous Integration . . . . .                                | 35        |

7.4.2 Tests . . . . . 35

# 1 Introduction

*Last updated 20th May 2019*

This document is an attachment to a bachelor's thesis at NTNU IDI, and functions as system documentation for the CutWear-system (version 1), which consists of a client and a server (API). The purpose of this document is to document all demands and dependencies to be able to install and maintain the system. The document contains all relevant information for maintaining, and further developing the system, by documenting how the system works and how to use it. For documentation of why certain choices, regarding technology, were made, please see the chapter 4 in the bachelor's thesis. This document is written for people with some knowledge of running applications and has some knowledge of programming with databases. If a tutorial or article has been used for developing some of the solutions, a reference is provided. One does not need knowledge of Node.js or npm to run the application for production, but some knowledge of Docker and Docker-compose is needed to start or stop the application.

All diagrams and models were made using draw.io.

The client and server is separated in two private Github repositories:

Server: Github - cutWearServer

Client: Github - CutWearClient

Each repository contains a README.md file, where the latest information will be available. If there is a reference to one of the README files in this document, the files are located on the repository root, see 3. For access to repositories, please refer to the customer in the thesis.

For future reference, any mentions of 'server' or 'client' in this document, will be referencing CutWearServer or CutWearClient, respectively.

The server in which the API is hosted (at the time this documentation is being written) and tested on, is a virtual machine running Ubuntu 18.04, 127 GB storage, 2 GB RAM and dual-core CPU and is available at <https://cutware.ibm.ntnu.no> (yes, there is a typo). See chapter 5 for more information about how to use the API.

## 2 Architecture and Design Patterns

### 2.1 Technology Stack

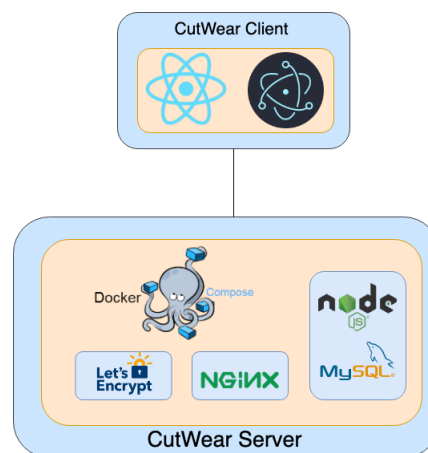


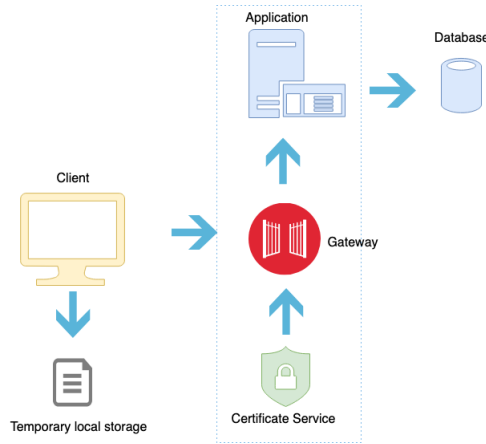
Figure 1: Overview of the technology stack of CutWear

Figure 1 shows a simple overview of the CutWear system. The client uses the JavaScript framework



ReactJS v16.8.1 and Electron v4.1.0 as platform. The server stack is bit more complicated, and uses Node.js v10.15.1 and Express v4.16.4. CutWear uses Docker-compose to build, start and stop server using multiple Dockerimages. The Docker images (represented by blue rectangles inside the golden rectangle) consists of one Docker image to obtain and renew Let’s Encrypt (TLS) certificates, the second Docker image (nginx) works as a load balancer/proxy (in case you want to run more instances of CutWear) and functions as the gateway to the CutWear API, by receiving a (HTTPS) request and passing it through to CutWear. [1] The third Docker image is the server application, which uses NodeJS with Express framework and connects to a external MySQL-database. [2]

## 2.2 Architecture



Figur 2: Architecture of CutWear. (not shown in image, the response flow)

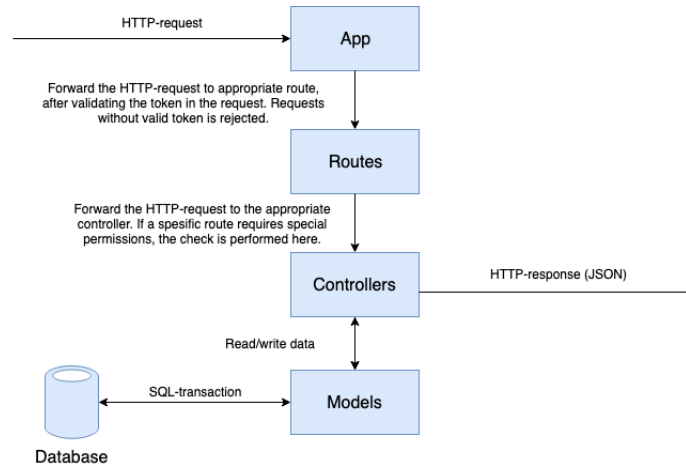
Figure 2 shows the architecture of CutWear, which is a REST-architecture. See 5 for REST-documentation. The client sends a HTTPS requests to the gateway, and the gateway processes the HTTPS the request and forwards it to the application server as a HTTP-request, where the request will be processed. If the request requires data, the application server makes a request to the database. The certificate service provides the gateway with valid certificates from Let’s Encrypt. Upon renewal of certificates the certificate service restarts the gateway. [1] If the client is offline, it will store changes in a temporary local file until reconnected to internet, where it will try to re-upload.

## 2.3 Model Route Controller

CutWear server uses a Model View Controller(MVC) design pattern, but since the View is only accessible through the desktop application, it has transformed into a Model Route Controller (MRC) pattern, where the Route has been included for easier understanding. Figure 3 shows the how a request is handled by the server and how different components interact with eachother. The response is always in JSON format. [3]

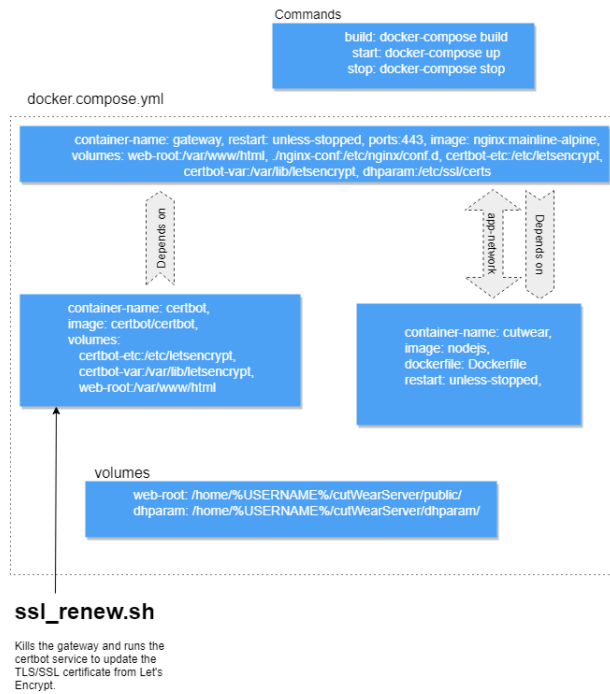
## 2.4 Docker-Compose

Figure 4 shows the Docker-compose setup for CutWear. The gateway is running the latest nginx-image from Dockerhub, and handles all incoming requests and passes them to cutwear. The gateway depends on cutwear running for performing its tasks. The cutwear service uses the Dockerfile in the server repository. Gateway and Cutwear uses a app-network, which is a virtual network, to communicate between each other. Cutwear and gateway can only be stopped by using the ‘stop’ command, since ‘restart’ is configured to ‘unless-stopped’, meaning the services will be restarted only if you use the stop command. Certbot runs the certbot image from Let’s Encrypt’s Dockerhub. Certbot sends a request to Let’s Encrypt to get or renew



Figur 3: Design pattern of the CutWear server application. Shows the flow of request and responses.

TLS/SSL certificates. Certbot should exit upon finishing its tasks. The `ssl_renew.sh` is a executable and kills the gateway and starts the certbot image and should be run in intervals using crontab. See chapter 7 for more information. [2] [1]



Figur 4: Docker-Compose setup for CutWear

## 2.5 Client

The client for this project was bootstrapped with Create React App, this gave the development team a starting point for developing the frontend. This is also explained in the README file of the project. If you do not have access to the README file, the documentation for Create React App can be found by following this link: [Create React App Documentation](#)

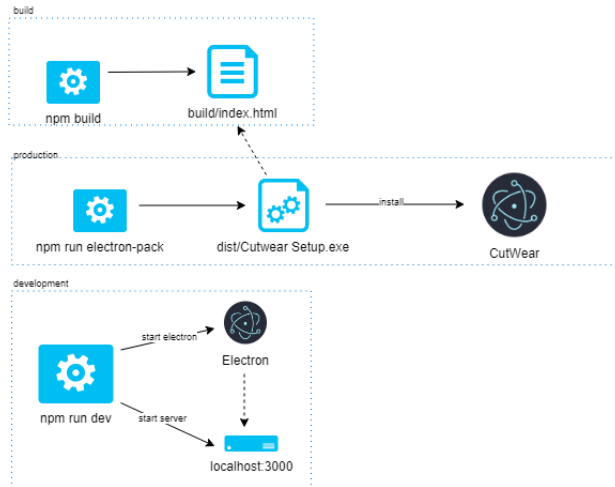


Figure 5: Overview of client in different environments

Figure 5 shows what files are being used by the client when running in different environments. To generate the build files, the node package manager (npm) calls the build script in the package.json, and then creates the build files which is located in the build directory. When packaging the application for production (`NODE_ENV=production`), using the dependency (electron-builder) and the command 'npm run electron-pack', it uses the build files to package and generate a installer for the application. [4] When running in a development environment (`NODE_ENV=development`), using 'npm run dev', it will start a local server, and start electron which points to localhost:3000 and opens the developer tools. When running electron with localhost instead of build files, you enable hot-reloading which means the application updates upon changes to code.

## 3 Projectstructure

### 3.1 Server

```
├── Dockerfile
├── README.md
├── coverage
│   ├── clover.xml
│   ├── coverage-final.json
│   └── lcov.info
├── docker-compose.yml
├── nginx-conf
│   └── nginx.conf
├── package-lock.json
├── package.json
├── public
│   └── index.html
├── resources
│   ├── config
│   │   └── winston.js
│   ├── db
│   │   ├── database.sql
│   │   ├── db.json
│   │   ├── droptables.sql
│   │   └── insertFile.sql
│   ├── jwt_token
│   │   └── jwt_token.json
│   ├── logs
│   │   └── 2019-05-06.cutwear.log
│   └── templates
│       └── RESTdoctemplate.md
├── src
│   ├── app.js
│   ├── controller
│   │   ├── checkToken.controller.js
│   │   ├── cutter.controller.js
│   │   ├── inspection.controller.js
│   │   ├── project.controller.js
│   │   ├── tbm.controller.js
│   │   └── user.controller.js
│   ├── model
│   │   ├── cutter.model.js
│   │   ├── inspection.model.js
│   │   ├── model.js
│   │   ├── project.model.js
│   │   ├── tbm.model.js
│   │   └── user.model.js
│   └── routes
│       ├── auth.route.js
│       ├── cutter.route.js
│       ├── inspection.route.js
│       ├── project.route.js
│       ├── routes.js
│       ├── tbm.route.js
│       └── user.route.js
└── ssl_renew.sh
```

Figur 6: File structure of the server (generated using tree command)

Figure 6 shows the file structure of the server. The source code is located under `src/` folder, with `app.js` being the entry file. The main route file is `route.js` which then passes the request to the correct route. Source files are usually build up with `*.role.js`. The source code is organized in MCR, as shown in 2.3.

The resources directory contains configuration files, templates and logs. The `Dockerfile`, `docker.compose.yml` and `ssl_renew.sh` contains the scripts for building and running the server. See 2.4 for more information.

Coverage folder contains result of tests. See 7.4.2 for more details. The `public` folder only contains `index.html`, which works as a API homepage. `Package.json` and `package-lock.json` contains information about the server, such as dependencies, versions, name, and npm scripts.

Not shown in 6 is the tests (7.4.2) folder, `out`, `pack_with_docker.sh` (7.3) folder and `node_modules`.

#### 3.1.1 npm scripts

##### npm install

Installs all the dependencies. Needs to be run first time and upon changes to dependencies.

##### npm run test

Script for running the tests in the server. See 7.4.2 for more information.

### **npm start**

Starts the server outside of Docker. The script points the './src/app.js' file and starts it using 'node'. Usually used in development environments.

### **npm run doc**

Generates the code documentation. See 7.3 for more details.

## **3.1.2 Docker commands**

Commands might require 'sudo'.

### **docker build -t cutwear node .**

Builds the image described in Dockerfile. The '-t' is added to give a name to the container, in this case 'cutwear'. The '.' describes the context of the build.

### **docker run -p 80:5000 -d cutwear**

Starts the image 'cutwear'. '-p' forwards port 80 to port 5000 in the image. Use '-d' to run image in the background. Port is hardcoded in app.js in server, so if you wish to use another port update the command and port in app.js or set the PORT environment variable. The application is now available at <http://localhost:5000>

### **docker stop cutwear**

Stops the image cutwear

### **docker stop**

Stops all images running

### **docker ps**

List of all running images.

## **3.1.3 Docker-Compose commands**

### **docker-compose build**

Builds all the images described in the docker.compose.yml file.

### **docker-compose build cutwear**

Builds the image named cutwear.

### **docker-compose up -d**

Start all the images. '-d' for running in the background.

### **docker-compose stop**

Stops all the images. Only valid way to stop gateway and cutwear image.

### **docker-compose stop cutwear**

Stops one image. In this case, 'cutwear' is the image to stop.

**docker-compose up -d --force-recreate --no-deps nodejs**

If you have stopped one image, use this command to start the image you stopped without affecting other images running.

**docker-compose logs**

Gets all the logs from the running images.

**docker-compose logs cutwear**

Gets the logs from cutwear.

## 3.2 Client

The Client has the following simplified structure, shown in figure 7. In addition to this there is also a node modules folder, but there is not relevant to go into detail about this folder.

```
├── icons
│   └── icon.png
├── package.json
├── package-lock.json
├── Procfile
├── public
│   ├── electron-starter.js
│   ├── favicon111.ico
│   ├── favicon.png
│   ├── index.html
│   └── manifest.json
├── README.md
└── src
    ├── Apps
    ├── Components
    ├── electron-wait-react.js
    ├── index.css
    ├── index.js
    ├── Pictures
    ├── Services
    └── serviceWorker.js
```

Figur 7: File structure of the client (generated using tree command)

The source code is located under src/ folder, with index.js being the entry file and index.css styling most of the app. Each user group has a main file located in the Apps/ folder that handles the state and saves the fundamental data throughout the session.

```
Apps
├── AdminApp.js
├── InspectorApp.js
├── SuperUserApp.js
└── SupervisorApp.js
```

Figur 8: File structure of the Apps/ folder (generated using tree command)

Since the client is written in React the application is composed of Components, these components is located in the Components/ folder. In the Components/ folder the components belonging to one specific user group is located in a folder with the same name as the user group.

```
Components
├── Admin
│   ├── AdminHelp.js
│   ├── AdminMain.js
│   ├── AdminTC.js
│   └── RegisterProject.js
├── Clock.js
├── Guidelines.js
├── ImageUpload.js
├── Login.js
├── PasswordReset.js
├── Project_Supervisor
│   ├── RegisterInspector.js
│   ├── RegisterTBM.js
│   ├── SupervisorHelp.js
│   ├── SupervisorMain.js
│   ├── SupervisorTC.js
│   └── ViewTBM.js
├── StartupList.js
├── Super_User
│   ├── InfoDownload.js
│   ├── SuperUserHelp.js
│   └── SuperUserTC.js
├── TBM_Inspector
│   ├── CutterChange.js
│   ├── CutterMain.js
│   ├── CutterWear.js
│   ├── ExistingInsp.js
│   ├── InspectorMain.js
│   ├── InspectorTC.js
│   └── InspInitialize.js
├── TBMSelect.js
└── Toolbar.js
```

Figur 9: File structure of the Components/ folder (generated using tree command)

The last folder that contains source code is the Services/ folder, this contains files that handle the communication with the server.

```
Services
├── Inspection.js
├── Project.js
└── User.js
```

Figur 10: File structure of the Services/ folder (generated using tree command)

### 3.2.1 npm scripts

#### **npm install**

Installs all the dependencies. Needs to be run first time and upon changes to dependencies.

#### **npm run dev**

Starts the client in development mode. If running app on Windows, remove 'BROWSER=none' from package.json. NODE\_ENV must be set to 'development'.

#### **npm run electron-pack**

Builds and packages the application to .exe. NODE\_ENV must be set to 'production'. Could take some time and consume some memory.

#### **npm test**

Calls the test script from create-react-app.

#### **npm run electron**

Starts electron without running server.

#### **npm run preelectron-pack**

Calls the build script before electron-pack.

#### **npm run doc**

Generates the code documentation. See 7.3 for more details.

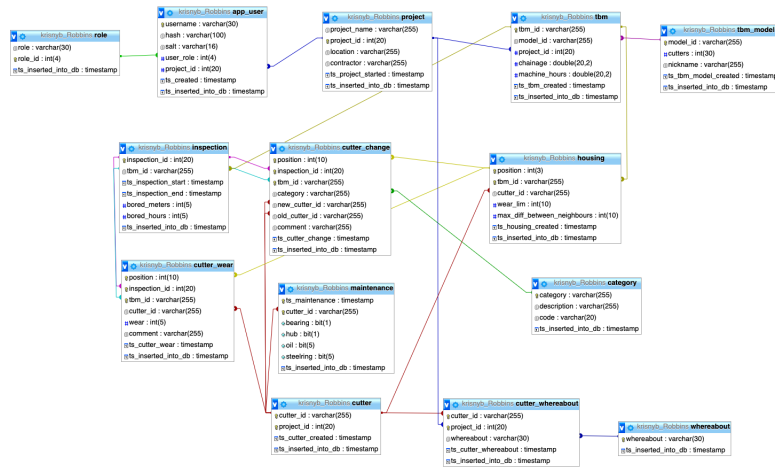
### 3.2.2 Docker scripts

#### **pack\_with\_docker.sh**

Starts up a Dockerimage and generates a Installer for platforms described in package.json. Results are stored in dist folder.



## 4 Database Model



Figur 11: Database model of CutWear, foreign keys are included

## 5 REST Services

This section has been written alongside development in markdown and is meant for the README in the repository. The file is therefore converted from markdown to pdf, which explains the inconsistent format of this section.



Auth required: YES  
Permissions required: Administrator or a project supervisor in the project which the new user is to added.  
Data Constraints:

```
{
  "username": "yourusernamehere",
  "password": "yourpasswordhere",
  "user_role": "role of user (integer)",
  "project_id": "id of project(integer)"
}
```

### Success Responses

Condition: User is created.  
Code: 200 OK  
Content Example:

```
{
  "status": true,
  "message": "User created successfully"
}
```

### Error Response

Condition: The user trying to register a new user does not have permission to create users.  
Code: 400 BAD REQUEST  
Content Example:

```
{
  "status": false,
  "message": "Permission Denied"
}
```

OR

Condition: Missing or incorrect parameters.  
Code: 422 Unprocessable Entity  
Content Example:

```
{
  "status": false,
  "message": "Missing or incorrect parameters"
}
```

OR

Condition: Something went wrong in registering user.  
Code: Ex: 500 INTERNAL SERVER ERROR  
Content Example: What the response content is to look like. Example:

```
{
  "status": false,
  "message": "Internal Server Error, user not created"
}
```

## Get a User

GET for getting an user.  
GET api/v1/users/:name  
URL parameters: Username is provided in the :name param.  
Auth required: YES  
Permissions required: NO

### Success Responses

Condition: A user object is provided in the response.  
Code: 200 OK  
Content Example:

```
{
  "status": true,
  "message": "success",
  "user": {
    "username": "UsyNamy",
    "user_role": 1,
    "project_id": 3,
    "ts_created": "2019-03-21T09:14:49.000Z"
  }
}
```

### Error Response

Condition: User is not found or internal server error.  
Code: 500 Internal Server Error  
Content Example:

16  
200

```
{
  "status": false,

```

```
"message": "Internal Server Error"
}
```

## Create a project

Creates a new project.

POST api/v1/project/create

URL parameters: NONE

Auth required: YES

Permissions required: User has to be admin.

Data Constraints:

```
{
  "name": "name of project",
  "location": "location of the project",
  "contractor": "contractor of the project",
  "ts_project_created": "Unix Timestamp of when the project was created"
}
```

Other constraints: The timestamp should be a unix timestamp.

### Success Responses

Condition: Project is created and a project id is returned.

Code: 200 OK

Content Example:

```
{
  "status": true,
  "message": "Project Created",
  "project_id": 1034
}
```

### Error Response

Condition: User is not admin

Code: 400 BAD REQUEST

Content Example:

```
{
  "status": false,
  "message": "Could not create project"
}
```

OR

Condition: Something went wrong server side.

Code: 500 Internal Server Error

Content Example:

```
{
  "status": false,
  "message": "Internal Server Error"
}
```

## Get a project

Gets a project with its information

GET api/v1/project/:projectId

URL parameters: YES, project\_id is provided in :projectId param.

Auth required: YES

Permissions required: User has to be member of the project, or be an admin or super user.

### Success Responses

Condition: Project is fetched from database and returned.

Code: 200 OK

Content Example:

```
{
  "status": true,
  "message": "Returning project 98",
  "project": [
    {
      "project_name": "ProjectsWithTBM",
      "project_id": 98,
      "location": "Korea",
      "contractor": "Arnes Entreprenarurrurur",
      "ts_project_started": "2019-03-20T09:12:01.000Z",
      "ts_inserted_into_db": "2019-03-20T14:46:51.000Z"
    }
  ]
}
```

### Error Response

Condition: Could not find project

Code: 400 Internal Server Error

Content Example:

```
{
  "status": false,
  "message": "Could not find project"
}
```

## Get all projects

Gets all the projects in database.

GET api/v1/project

URL parameters: NONE

Auth required: YES

Permissions required: User has to be admin, workshop supervisor or super user.

### Success Responses

Condition: A list of projects.

Code: 200 OK

Content Example:

```
{
  "status": true,
  "message": "Returning projects ",
  "projects": [
    {
      "project_id": 1,
      "name": "Arnes Prosjekt",
      "location": "Valhall",
      "contractor": "LNS",
      "project_started": "2019-03-20T09:15:01.000Z"
    },
    {
      "project_id": 2,
      "name": "KÅshagen",
      "location": "Ytre Enebakkt",
      "contractor": "Arne",
      "project_started": "2019-03-20T04:15:01.000Z"
    }
  ]
}
```

### Error Response

Condition: No projects

Code: 400 BAD REQUEST

Content Example:

```
{
  "status": false,
  "message": "Could not find projects"
}
```

### OR

Condition: User does not have permission to view this data

Code: 401 UNAUTHORIZED

Content Example:

```
{
  "status": false,
  "message": "Permission Denied"
}
```

### OR

Condition: Internal Server Error

Code: 500 Internal Server Error

Content Example:

```
{
  "status": false,
  "message": "Internal server error"
}
```

## Get all TBMs in project

Gets all the TBMs connected to a project.

GET api/v1/project/:projectId/tbm/all

URL parameters: :projectId

Auth required: YES

Permissions required: User has to be admin,superuser or member of project

### Success Responses

Condition: List of TBMs is returned in body.

Code: 200 OK

Content Example:

```
{
  "success": true,
  "message": "Retrieving TBMs successful",
}
```

```
"tbn": "tbms[]"
}
```

### Error Response

Condition: Project doesnt exist or project has no tbms.

Code: 500 Internal Server Error

Content Example:

```
{
  "status": false,
  "message": "Fetching TBM unsuccessfull"
}
```

### Create a new TBM in project

Creates a new TBM in a project

POST api/v1/project/:projectId/tbm/create

URL parameters: :projectId

Auth required: YES

Permissions required: User has to be admin or project supervisor

Data Constraints:

```
{
  "tbmId": "string",
  "ts_tbm_created": "unix timestamp",
  "housing": [
    {
      "position": "int",
      "wear": "int",
      "max_between_neighbours": "int"
    }
  ]
}
```

### Success Responses

Condition: TBM is created

Code: 200 OK

Content Example: What the response content is to look like. Example:

```
{
  "success": true,
  "message": "TBM tbm203 created"
}
```

### Error Response

Condition: Missing parameters in request or user does not have permission to create tbm

Code: 400 BAD REQUEST

Content Example:

```
{
  "status": false,
  "message": "One or more missing attributes" || "user does not have permission"
}
```

OR

Condition: Error in checking user permission or error in creating the tbm.

Code: 500 Internal Server Error

Content Example:

```
{
  "status": false,
  "message": "Internal Server Error"
}
```

### Get cutters in a TBM

Fetches all the cutters in a TBM

GET api/v1/project/:projectId/tbm/:tbmId/cutters/

URL parameters: projectId and tbmId

Auth required: YES

Permissions required: User has to be admin, super user or member of project.

### Success Responses

Condition: Found cutters

Code: 200 OK

Content Example:

```
{
  "success": true,
  "message": "Cutterdata from tbm tbm203",
  "data": "cutters[]"
}
```

## Error Response

Condition: Did not find any cutters

Code: 404 NOT FOUND

Content Example:

```
{
  "status": false,
  "message": "Error in fetching cutterdata for tbn tbn203"
}
```

## All cutters in project

Gets all the cutters in a project

GET api/v1/project/:projectId/cutters/all

URL parameters: :projectId

Auth required: YES

Permissions required: User has to be admin,super user or member of project

### Success Responses

Condition: Found cutters

Code: 200 OK

Content Example:

```
{
  "success": true,
  "message": "Cutterdata from project 3",
  "data": "cutters[]"
}
```

## Error Response

Condition: Did not find any cutters

Code: 404 NOT FOUND

Content Example:

```
{
  "status": false,
  "message": "Error in fetching cutterdata for project 3"
}
```

## Get a cutter

Fetch a cutter from a project

GET api/v1/project/:projectId/cutter/:cutterId

URL parameters: :projectId, :cutterId

Auth required: YES

Permissions required: User has to be admin,super user or member of project

### Success Responses

Condition: Found cutter

Code: 200 OK

Content Example:

```
{
  "status": true,
  "message": "returning cutter 234",
  "data": "{cutter}"
}
```

## Error Response

Condition: Could not find cutter

Code: 404 NOT FOUND

Content Example:

```
{
  "status": false,
  "message": "No data or wrong project"
}
```

OR

Condition: Error in fetching cutters

Code: 500 Internal Server Error

Content Example:

```
{
  "status": false,
  "message": "Internal Server Error"
}
```

20

204

## Cutter categories



Gets the cutter categories for changing cutters.

GET /api/v1/project/cutters/categories

URL parameters: NONE

Auth required: YES

Permissions required: NONE

### Success Responses

Condition: Categories are found

Code: 200 OK

Content Example: What the response content is to look like. Example:

```
{
  "success": true,
  "message": "found categories",
  "categories": "categories[]"
}
```

### Error Response

Condition: Error in fetching categories

Code: EX: 500 Internal Server Error

Content Example:

```
{
  "status": false,
  "message": "Internal Server Error"
}
```

### Post Inspection

Creates a new inspection

POST api/v1/project/:projectId/tbm/:tbmId/inspection/post

URL parameters: :projectId, tbmId

Auth required: YES

Permissions required: User has to admin, superuser or member of project

Data Constraints:

```
{
  "ts_start": "unix timestamp",
  "ts_end": "unix timestamp",
  "boredhours": "double",
  "boredmeters": "double",
  "newCutters": [
    {
      "cutterId": "string",
      "ts_cutter_created": "unix timestamp",
      "position": "int"
    }
  ],
  "wear": [
    {
      "position": "int",
      "cutter_id": "string",
      "wear": "int",
      "comment": "string",
      "ts_cutter_wear": "unix timestamp",
    }
  ],
  "changes": [
    {
      "position": "int",
      "comment": "string",
      "category": "string",
      "new_cutter_id": "string",
      "old_cutter_id": "string",
      "ts_cutter_change": "unix timestamp"
    }
  ]
}
```

### Success Responses

Condition: Inspection is created

Code: 200 OK

Content Example:

```
{
  "status": true,
  "message": "inspection successful"
}
```

### Error Response

Condition: Missing parameters

Code: 400 BAD REQUEST

Content Example:

```
{
  "status": false,
  "message": "Missing one or more inputs"
}
```

OR

Condition: Error in creating inspection

Code: 500 Internal Server Error

Content Example:

```
{
  "status": false,
  "message": "Error in creating a inspection"
}
```

## Get inspections on tbm

Gets all the inspections done on a tbm

GET api/v1/project/:projectId/tbm/:tbmId/inspection/all

URL parameters: :projectId, :tbmId

Auth required: YES

Permissions required: User has to be admin, super user or member of project

### Success Responses

Condition: Inspections found

Code: 200 OK

Content Example: What the response content is to look like. Example:

```
{
  "status": true,
  "message": "inspectiondata from tbm tbm203",
  "data": "inspections[]"
}
```

### Error Response

Condition: Could not find inspections

Code: 404 NOT FOUND

Content Example:

```
{
  "status": false,
  "message": "error in fetching inspectiondata for tbm203"
}
```

## Get 3 latest inspections on tbm

Gets the three latest inspections done on a tbm

GET api/v1/project/:projectId/tbm/:tbmId/inspection

URL parameters: :projectId, :tbmId

Auth required: YES

Permissions required: User has to be admin, super user or member of project

### Success Responses

Condition: Inspections found

Code: 200 OK

Content Example: What the response content is to look like. Example:

```
{
  "status": true,
  "message": "inspectiondata from tbm tbm203",
  "data": "inspections[]"
}
```

### Error Response

Condition: Could not find inspections

Code: 404 NOT FOUND

Content Example:

```
{
  "status": false,
  "message": "error in fetching inspectiondata for tbm203"
}
```

## 6 Security

### 6.1 Server

#### 6.1.1 Hashing and storing passwords

The server uses bcrypt as hashing algorithm and all passwords are stored as hashes with the same length, alongside a salt in the database.[5]

There are only two ways to create or change a hash using the API, and that is through the registerUser and updatePassword methods in user.controller, which only a administrator or a project supervisor (of the same project as the user to be created/updated) has permission to do.

| username | hash                                                     | salt             |
|----------|----------------------------------------------------------|------------------|
| test2    | \$2b\$10\$PRdKd8P7oIE/g2kpBsBK/ B7VrMROmL5 UTX5Tvbwic... | b2cdc860bb1913c9 |

Figur 12: A users credentials stored in the database

#### 6.1.2 JSON Web Tokens

JSON Web Token (JWT) is a digital signed token. For more information about creating a secret and setting expiry date for CutWear server, see 7

CutWear uses jwt for access to the system. Without a token in the request, the request will be denied. To receive a token, see 5

To use the token in a request, put the token as a Bearer token in the request header under x-access-token or authorization.

Example:

```
authorization : Bearer myToken1234
x - access - token : Bearer myToken1234
```

For more details about JWT: [jwt.io](http://jwt.io) or [6]

#### 6.1.3 Let's Encrypt

Communication between client and server uses HTTPS. Let's Encrypt(LE) provides the CutWear server application with free TLS/SSL certificates to ensure secure connections between client and server.[7] The certificate is fetched using the certbot Docker-image. LE certificates expire after 90 days, but with a simple script, the renewal process can be automated. Add the `ssl.renew.sh` to crontab or an equivalent program to renew the certificates before expiry date. For more detailed information on how to setup auto renewal, see chapter 7

For more details about Let's Encrypt: [letsencrypt.org](http://letsencrypt.org)

#### 6.1.4 SQL security

For connecting to the database, the node.js mysql driver is used. Mysql provides a transaction method, which is used for all queries. To prevent SQL-injections, all queries are escaped using parameters. For easier use for developers, model.js has a method "query", which takes in a sql statement and parameters. "query" is transactional and rolls back the changes if the statement fails [8]. All \*.model classes should inherit the Model class, and by using the super method "query", making sure that all queries are executed correctly [9]. Example of usage from `src/controllers/CutterModel.js`:

```

/**
 * Model for reading and writing user data.
 * @extends Model
 */
class UserModel extends Model {
  /**
   * Takes a username and returns the matching row in app_user
   * @param {string} name
   */
  getUser(name) {
    let sql = 'SELECT * FROM app_user WHERE username=?';
    let params = [name];
    return new Promise((resolve, reject) => {
      super.query(sql,params)
        .then(rows => {
          resolve(rows);
        })
        .catch(err => {
          reject(err);
        });
    });
  }
}

```

Figur 13: Example of SQL-seurity from CutterModel.js

### 6.1.5 Protecting routes with middleware

Express provides simple ways of integrating middleware in the code, by using the ‘use’ and ‘param’ method, one can run security checks before executing the target code [10]. The ‘use’ method excecutes when a request route matches the route in the ”use”method. This makes it possible to check everytime someone requests data which requires special permissions [11]. Example from the route ‘/api/v1/users/:name’ which requires a valid token to be able to request. Figure 14 shows the middleware used for checking tokens. If the token is

```

/**
 * Checks is the token in the request is valid.
 * If valid, send to next middleware,
 * if not valid, return http-400.
 * @param (*) req Request
 * @param (*) res Response
 * @param (*) next Next, function to be called next.
 */
let authUser = (req,res,next) => {
  check.checkToken(req, res)
  // if token is valid, go to next function.
  .then(() => next())
  .catch(err => {
    // if token is invalid or not available, return error message.
    res.status(400).json({
      status: false,
      message: err.message
    });
  });
}

```

Figur 14: The middleware for checking valid token

invalid, it will return a 400, but if the token is valid, it calls the next method, which tells the ‘use’ method to call the next method in the chain. If the next method is called in authUser, 15 shows that addPool (another

```

// baseroute, auth user and adds pool, then next()
app.use(urlRoot,authUser,addPool,routes);

```

Figur 15: Middleware for all requests starting with urlRoot(”api/v1”)

middleware) is run, and the request is then sent through to the next handler, which in this case is the route.js, which then again will forward the request to the user.route and then the user.controller. The param”method is excecuted everytime a certain parameter is present in a route.

### 6.1.6 Heavy load scenarios

All requests are initially received by the gateway (see 2.4), and therefore works as a load balancer if you wish to run more instances of the server and configure it correctly. If the gateway crashes under heavy load, the cutwear server will not be affected other than it will not receive the requests meant for it.

## 6.2 Client

### 6.2.1 Cross Site Scripting(XSS)

ReactJS handles Cross Site Scripting for you by escaping all user input [12].

### 6.2.2 Client-security

To have a minimal amount of time where the users password is stored as clear text (in variables etc), the password is also hashed (and salted) in the client, before removed from variables. When the client sends a login request it uses the fore mentioned hash to authenticate itself. The hashing adds no further security other than that the password is not in clear text for others to see (backend or frontend).

### 6.2.3 Timestamps

Since the client has the possibility of being used offline, there are some problems with knowing what is the correct and newest data. This is handled by timestamping every event on the frontend at the time of the event, and is then sent with the post-request to the database and stored. Then you can have uncommitted events offline on a device, and still not have any conflicts when it is committed. To prevent other errors with using timestamps over different timezones, the timestamp is a UNIX timestamp which is in milliseconds since 1 jan 1970 UTC [13].

## 7 Installation and running the application

### 7.1 Server

To install and run the application, you need this software installed:

- node.js (needed for development and production)
- Docker (not necessary for development)
- Docker-compose (needed for production)
- mysql (needed for development and production, if you do not have external database)
- git (useful for easy version control)
- if the documentation is highlighted by %, then it means to replace the text with your own information.

#### 7.1.1 First time installation

1. Clone or pull the latest version from Github
2. If you do not have a pre-configured cutwear database, you need to install the database.
  - Create a database in mysql.
  - Run the `./resources/db/insertFile.sql` to setup the database structure.
3. Create the file `./resources/db/db.json`.
4. Add the database settings to `db.json`, like figure 16
5. Create the file `./resources/jwt_token/jwt_token.json`
6. Add the jwt secret and expiry date to `jwt_token.json`, like figure 17

```

{
  "connectionLimit":100,
  "host": "hostname",
  "database":"databasename",
  "user":"user",
  "password":"password of db",
  "debug": false,
  "multipleStatements":true
}

```

Figur 16: Example of database connections settings

```

{
  "token":"Secret token, make your own",
  "expiresIn": "when the token expires, can be written like '12h'"
}

```

Figur 17: JWT token settings

### 7.1.2 Running server outside Docker

1. run 'npm install'
2. run 'npm start'
3. Server is now available on http://localhost:5000

### 7.1.3 Running server with Docker

1. Build the image
2. Start the image
3. application is now available on http://localhost/

### 7.1.4 Running server in production with Docker-Compose [1]

1. This configuration is tested on Ubuntu 18.04.
2. Make sure the project is cloned to the '/home/%USERNAME%/' directory, otherwise you need to update the directories in the nginx-conf/nginx.conf file and docker-compose.yml.
3. Update the nginx-conf/nginx.conf with the correct domains. Replace %DOMAIN% with the actual domain.

**Example:**

```

server_name cutware.ibm.ntnu.no;
ssl_certificate /etc/letsencrypt/live/cutware.ibm.ntnu.no/fullchain.pem;
ssl_certificate_key /etc/letsencrypt/live/cutware.ibm.ntnu.no/privkey.pem;
To add more domains: server_name cutware.ibm.ntnu.no www.cutware.ntnu.no

```

4. Update docker-compose.yml. Replace the %EMAIL% with an email and replace %DOMAIN% with the same domain from nginx.conf. If you have more than one domain, append -d other.domain.com at the end of command.

**Example:**

```

command: certonly --webroot --webroot-path=/var/www/html --email krisnyb@stud.ntnu.no --agree-tos
--no-eff-email --force-renewal -d cutware.ibm.ntnu.no -d www.cutware.ibm.ntnu.no

```

5. Creating the keys.

On the project root: 'mkdir dhparam'.

Run 'sudo openssl dhparam -out /home/%USERNAME%/cutWearServer/dhparam/dhparam-2048.pem 2048'

6. Setting up auto-renewal of Let's Encrypt certificate. Update the `ssl_renew.sh` with the correct directory (update `%USERNAME%`).
7. Open crontab using `'sudo crontab -e'`. Add:  
`'0 12 * * * /home/%USERNAME%/cutWearServer/ssl_renew.sh >> /var/log/cron.log 2>&1'`  
to the bottom of the file. This will try to renew every 12th hour. A certificate lasts 90 days, so the interval could be bigger.
8. To see logs for certification renewal: `'tail -f /var/log/cron.log'`.
9. Build the images with `'docker-compose build'`
10. Starting the servers. Make sure Docker-Compose is installed. Run `'docker-compose up -d'`. The `'-d'` tells it to run in the background. Check status of servers: `'docker-compose ps'`. Certbot should have exited with 0, and cutwear and gateway should have state=UP.

NOTE: Even though server is running, it might not be possible to log on using tools like curl or postman with the same user and password one uses to log in to the client, since the passwords are hashed on client-side. The production environment setup(with Let's Encrypt, Docker-compose) was created using this.

### 7.1.5 Handling updates when in production

1. Clone or pull the latest version of the image.
2. Stop the server in question:  
`sudo docker-compose stop %IMAGENAME%`
3. Rebuild the image:  
`sudo docker-compose build %IMAGENAME%`
4. Restart the server:  
`sudo docker-compose up -d --force-recreate --no-deps %IMAGENAME%`.
5. Check the server status (state should be UP):  
`sudo docker-compose ps`

### 7.1.6 Libraries

This section will contain libraries used in the server with where they are used and what version. Links or other resources to the libraries will be documented here. If the resource is a guide or tutorial, then it has been used in the project as guidance or been used as a learning resource.

#### **bcrypt**

- Version 3.0.4
- Usage
  - `user.controller.js` - used to hash, and to compare two hashes
- Documentation
  - `bcrypt-npm`

## **crypto**

- Version 1.0.1
- Usage
  - user.controller.js - used to create a random salt of 16 bytes
- Documentation
  - The site containing the algorithm to create salt
  - Built-in to NodeJS

## **winston**

- Version 3.2.1
- Usage
  - ./resources/config/winston.js - configuration file
  - Replaces console.log, and writes to file and console.
- Documentation
  - Github
  - winston-npm
  - Tutorial used for setting winston up for server

## **jsonwebtoken**

- Version 8.5.0
- Usage
  - user.controller.js - Generates a token
  - checkToken.controller.js - Verifies a token
- Documentation
  - Guide for adding JWT to application
  - jsonwebtoken-npm

## **morgan**

- Version 1.9.1
- Usage
  - ./src/app.js - Logs all incoming requests and responses
- Documentation
  - Tutorial used for setting winston up for server
  - morgan-npm



## **express**

- Version 4.16.4
- Usage
  - Web-framework for NodeJS
  - ./src/app.js - Instanced to create a HTTP server
  - ./src/routes/ - Forwards all routes from app.js to the correct routes in route folder.
- Documentation
  - express-npm
  - Express Tutorial Part 4: Routes and controllers

## **jsDoc**

- Version 24.1.0
- Usage
  - npm run doc - used for generating code documentation.
- Documentation
  - jsDoc Dev Docs
  - jsdoc-npm
  - jsDoc cheatsheet

## **jest**

- Version 24.1.0
- Usage
  - npm test - used for running unittests
- Documentation
  - Jest Docs

## **moment**

- Version 2.24.0
- Usage
  - Used to convert unix timestamps (milliseconds since 1 jan 1970) to YYYY-MM-DD H:mm:ss (mysql format)
- Documentation
  - MomentJS

## mysql

- Version 2.16.0
- Usage
  - model.js - Used to connect to mysql database using a pool.
  - Uses transactions and roles back on error.
- Documentation
  - mysql-npm
  - Example of creating the model from TDAT3019
  - Creating transactions in mysql-npm

## 7.2 Client

### 7.2.1 Creating a installer without Docker

1. Clone or pull the latest version of the client from Github.
2. Set 'NODE\_ENV=development'
3. 'electron-builder' has dependencies who rely on python27. Easiest way to avoid the problem is to download and install build essentials.  
For Windows:
  - (a) Open CMD as administrator.
  - (b) run 'npm install -global -production windows-build-tools'For OSX: Python27 should come with OSX.  
For Linux:
  - (a) run 'sudo apt-get update'
  - (b) run 'sudo apt-get install build-essential'
4. run 'npm install' to install dependencies.
5. Set 'NODE\_ENV=production'
6. run 'npm run electron-pack'. This might take a while.
7. The result of the build is located in the 'dist' directory. Run the CutWear Setup version.exe to install on client-pc.

NOTE: if error under npm run electron-pack caused by fsevents, try downgrading the node version to 8.15. This has shown to work if node 10 or 11 errors out. npm i --save-dev node@8.15.0. If error with .sh not finding the build script, set 'NODE\_ENV=development' to make sure it downloads the electron-builder dependency, as it is a devDependency in package.json. To see if that worked, see if './node\_modules/.bin/build' exists

For further development, it would be wise to make a Docker image who builds and packages the application, to prevent any error caused by different dependencies, and also make it a part of the continous integration pipeline.

### 7.2.2 Creating a installer with Docker

1. Make sure Docker is installed, and bash is installed (check if able to run bash scripts)
2. Run './pack\_with\_docker.sh'. Run with sudo if no permission.
3. The result of the build is located in the 'dist' directory. Run the CutWear Setup version.exe to install on client-pc.

### 7.2.3 Installing on Windows x64 PC

1. Download or get the installer.
2. Double click the installer and it will install automatically.
3. If there is warning about installing the program, press ‘More info’ or equivalent, and press ‘Run/install anyway’.
4. The application opens upon finishing the installation.
5. A shortcut will be saved on the desktop, called ‘CutWear’.

### 7.2.4 Uninstalling the desktop application

1. Navigate to ‘C:/users/%username%/appdata/local/programs/cutwear’
2. Double click the uninstaller and it will uninstall automatically.

### 7.2.5 Development environment

1. Clone or pull the latest version of the client from Github.
2. run ‘npm install’
3. Set ‘NODE\_ENV=development’.
4. if developer machine is a Windows machine, remove ‘BROWSER=none’ from ‘package.json’.
5. Run ‘npm run dev’. A local server and electron starts up.
6. Ctrl+C to exit server and electron.

NOTE: If ‘npm run dev’ fails because of too many connections error, try setting ‘NODE\_ENV=production’. This will make electron point to the build files instead of the server. This disables hot-reloading and ‘npm run dev’ has to be run again. The error message is inconsistent, and does not show up everytime, so it might work after a reboot.

References for all tutorials, guides or other resources used to help with electron and react: [4], [14], [15], [16], [17] and [18].

### 7.2.6 Libraries

This section will contain libraries used in the client with where they are used and what version. Links or other resources to the libraries will be documented here. If the resource is a guide or tutorial, then it has been used in the project as guidance or been used as a learning resource.

#### onsen UI

- Version 2.10.6
- Usage
  - Used for styling the app and providing components.
- Documentation
  - <https://onsen.io/v2/guide/react/>
- Parts of the following examples have been used in the application
  - Toolbar: <https://onsen.io/playground/?framework=react&category=reference&module=splitter>
  - Login: <https://onsen.io/playground/?framework=react&category=reference&module=input>

## **react-router**

- Version 4.3.1
- Usage
  - Used for routing between Login and the different Apps
- Documentation
  - <https://www.npmjs.com/package/react-router>
  - Guide: <https://reacttraining.com/react-router/>

## **react-numericinput**

- Version 4.1.2
- Usage
  - Used for displaying a numericinput for input in the application
- Documentation
  - <https://www.npmjs.com/package/react-numericinput>

## **create-react-app**

- 

## **bcrypt**

- Version 3.0.5
- Usage
  - src/Services/User.js - used to hash, and to compare two hashes
- Documentation
  - [bcrypt-npm](#)

## **electron**

- Version 4.1.0
- Usage
  - Cross-platform desktop app to be used with JavaScript, HTML and CSS.
  - Client is running in a electron application
- Documentation
  - [Homepage of electron](#)

## electron-builder

- Version 20.39.0
- Usage
  - Node library for building and packaging electron apps.
  - Creates a .exe installer for the client.
- Documentation
  - Github of electron-builder
  - Electron-builder documentation
  - How to build an Electron app using create-react-app. No webpack configuration or ‘ejecting’ necessary.
  - Using Electron with React: The Basics
  - Turn The Famous React Boilerplate Into Electron Desktop Application
  - Build Electron App using Docker on a Local Machine

### 7.3 Documentation of source code

For code documentation, jsCode has been used. To generate the docs, first 'npm install', then 'npm run doc'. This will generate a directory called './out', and in the directory, open 'index.html' in a browser of your own choosing.

## 7.4 Continuous Integration and testing

### 7.4.1 Continuous Integration

The continuous integration (CI) of CutWear uses TravisCI for running the tests in an Linux Ubuntu Trusty environment and Codecov.io for publishing test results.

The server's continuous integration pipeline works as following:

Push to Github -> TravisCI: Run tests -> Codecov.io: Test results and statistics on code coverage is published on codecov.

The configuration is described in 'travis.yml'. The configuration file is a boilerplate travis-node\_js configuration [19] with added functionality to push test results to codecov.io after success. [20]

NOTE: as of 15th of May, the project has reached it's maximum builds for private Github repositories in TravisCI. The replacement for TravisCI could be CircleCi, but is not implemented at this time. Running the test locally (see under) will produce the same results.

### 7.4.2 Tests

The server uses jest for unit testing. To run('npm install' first) the tests, use the script 'npm test' or 'npm test file.js' to test a specific file. The results of the testing, can be seen in the file './coverage/lcov-report/index.html'. Here one can see the code coverage, which is also available on codecov.io.

The controller methods have been tested for logic, using stubs and mocks for all I/O from the models. The models are not unit tested since they mainly only returns the result from the sql queries, but the main query from Model class has been tested (used by all child models to query database). The controllers have a code coverage of about 97%.

Each SQL-query is tested manually using NTNU Php Myadmin. For testing each REST-endpoint (routes+controllers+models), Postman has been used, but is a manual job, and hard to document in a reusable way.

## Referanser

- [1] *How To Secure a Containerized Node.js Application with Nginx, Let's Encrypt, and Docker Compose.* side: <https://www.digitalocean.com/community/tutorials/how-to-secure-a-containerized-node-js-application-with-nginx-let-s-encrypt-and-docker-compose>.
- [2] «How To Build a Node.js Application with Docker», side: <https://www.digitalocean.com/community/tutorials/how-to-build-a-node-js-application-with-docker>.
- [3] «Express Tutorial Part 4: Routes and controllers», side: [https://developer.mozilla.org/en-US/docs/Learn/Server-side/Express\\_Nodejs/routes](https://developer.mozilla.org/en-US/docs/Learn/Server-side/Express_Nodejs/routes).
- [4] «Using Electron with React: The Basics», side: <https://medium.com/@brockhoff/using-electron-with-react-the-basics-e93f9761f86f>.
- [5] «Hashing in Action: Understanding bcrypt», side: <https://auth0.com/blog/ hashing-in-action-understanding-bcrypt/>.
- [6] «A guide for adding JWT token-based authentication to your single page Node.js applications», side: <https://medium.com/dev-bits/a-guide-for-adding-jwt-token-based-authentication-to-your-single-page-nodejs-applications-c403f7cf04f4>.
- [7] «How to Use SSL/TLS with Node.js», side: <https://www.sitepoint.com/how-to-use-ssl-tls-with-node-js/>.
- [8] «MySQL transactions in NodeJS», side: <https://www.codediesel.com/nodejs/mysql-transactions-in-nodejs/>.
- [9] «TDAT3019», side: <https://gitlab.stud.iie.ntnu.no/nilstesd/DatabaseTest>.
- [10] «A Simple Explanation Of Express Middleware», side: <https://medium.com/@agoiabeladeyemi/a-simple-explanation-of-express-middleware-c68ea839f498>.
- [11] «Learn how to handle authentication with Node using Passport.js», side: <https://medium.freecodecamp.org/learn-how-to-handle-authentication-with-node-using-passport-js-4a56ed18e81e>.
- [12] «JSX Prevents Injection Attacks», side: <https://reactjs.org/docs/introducing-jsx.html#jsx-prevents-injection-attacks>.
- [13] «What is the unix time stamp?», side: <https://www.unixtimestamp.com/>.
- [14] «How to build an Electron app using create-react-app. No webpack configuration or “ejecting” necessary.», side: <https://medium.freecodecamp.org/building-an-electron-application-with-create-react-app-97945861647c>.
- [15] «Build a File Metadata App in Electron», side: <https://codeburst.io/build-a-file-metadata-app-in-electron-a0fe8d32410e>.
- [16] «How to make an electron app using Create-React-App and Electron with Electron-Builder.», side: <https://gist.github.com/matthewjberger/6f42452cb1a2253667942d333ff53404>.
- [17] «Turn The Famous React Boilerplate Into Electron Desktop Application», side: <https://medium.com/@mjangir70/turn-the-famous-react-boilerplate-into-electron-desktop-application-68d91dce8d3a>.
- [18] «A complete guide to packaging your Electron app», side: <https://medium.com/how-to-electron/a-complete-guide-to-packaging-your-electron-app-1bdc717d739f>.
- [19] «Building a JavaScript and Node.js project», side: <https://docs.travis-ci.com/user/languages/javascript-with-nodejs/>.
- [20] «Build Environment Overview», side: <https://docs.travis-ci.com/user/reference/overview/>.



## G Ordinær oppgavebeskrivelse

**Arbeidstittel:**

Beslutningsstøtteverktøy for tunnelboring i hardt berg

**Hensikten med oppgaven:**

Utvikling av programvare for beslutningsstøtte i forbindelse med utskifting av borverktøy på tunnelboremaskiner (TBMer) som borer tunneler i sterk og slitende bergmasse.

Fagområdet Anleggsteknikk driver forskning på TBM, dvs. fullprofilboring eller mekanisk brytning av berg uten bruk av sprengstoff. Brytning skjer ved hjelp av borverktøy (kuttere) som er utformet som en ring av spesialstål, som er det mekaniske verktøyet som bryter ut bergmassen basert på aksiell kraft og rotasjon. Se aktuell link f.eks.: <https://www.youtube.com/watch?v=J0YqpZCoYEQ>. I typisk norske bergmasser (som forøvrig finnes over hele verden), er slitasje og dermed bytte av borverktøy en kritisk faktor for suksess.

Basert på eksisterende NTNU prognosemodell for estimering av tids- og kostnadsforbruk, ble det av to studenter fra Institutt for datateknologi og informatikk i 2017 utviklet en programmodul (dataverktøy) for bruk av denne prognosemodellen.

Det ønskelig med ny programutvikling innen fullprofilboring av tunneler. I samarbeid med The Robbins Company i Seattle, USA, har vi identifisert et behov for digitale verktøy som støtte i produksjonsprosessen, nærmere bestemt for overvåking av slitasjetilstand og optimal strategi for bytte av borverktøy (kuttere).

Overordnede hensikter med oppgaven er å bidra til digitalisering av tunnelbransjen, samt å utvikle dataingeniører med kunnskap om tunnelbransjen.

**Kort beskrivelse av oppgaveforslag:**

1. Programmet skal brukes til å registrere tilstand på alle borverktøy på borhodet ved periodisk inspeksjon (1 eller flere ganger i døgnet). Flere innparametere.
2. Programmet skal brukes til å registrere bytter av borverktøy. Flere innparametere.
3. Innsamlede data skal overføres til en egnet database (nettbasert?).
4. Registrerte data skal analyseres for å gi real-time tilbakemeldinger og beslutningsstøtte til operatør som foretar inspeksjon og/eller bytte av borverktøy.
5. Registrerte data skal også brukes til mer omfattende analyser av f.eks. borverktøylevetid og TBM-design.
6. Programmet skal tilpasses bruk i krevende tunnelmiljø ved inspeksjon og bytte av borverktøy. Programmet må kunne brukes på industristandard nettbrett, f.eks. IP67-kvalitet.
7. Det er ønskelig at brukergrensesnittet fullt ut er basert på touch-screen.
8. Programmet utvikles i samarbeid med Institutt for bygg- og miljøteknikk ved NTNU og The Robbins Company ved deres representant i Norge/trondheim Sindre Log.
9. Det må påregnes besøk til tunnelanlegg i Norge eller Europa som introduksjon til teknologien for fullprofilboring av tunneler i sterk og slitende bergmasse.
10. Studentene vil få grundig og regelmessig veiledning og oppfølging ved gjennomføring av oppgaven.

**Oppgaven passer for (kryss av de(t) som passer og skriv evt. en kommentar til oss):**

- Bacheloroppgave  
- Masteroppgave

**Kan oppgavestiller stille arbeidsplass med nødvendig utstyr og programvare:** Bærbar PC, nettbrett, nødvendig programvare, ja.

**Begrensninger i tilgjengelighet av opplysninger o.l.:**

Deler av designinformasjon for TBMer fra The Robbins Company vil

være fortrolig.

**Oppgaven passer best for, antall studenter:**

- 2
- 3
- 4

#### Opplysninger om oppgavestiller

**Navn på bedrift eller organisasjon:** Institutt for bygg- og miljøteknikk, NTNU og The Robbins Company, Seattle/Solon, USA

**Adresse** Høgskoleringen 7A

**Postnummer** 7491

**Poststed** Trondheim

**Navn på kontaktperson/veileder:** Amund Bruland

**Telefon:** 97093253

**Epost:** amund.bruland@ntnu.no

**Navn på kontaktperson 2/veileder 2:** Sindre Log

**Telefon kontaktperson 2/veileder 2:** 48298792

**Epost kontaktperson 2/veileder 2:** logsindre@robbinstbm.com

#### Utfyllende kommentarer til hva oppgaven gjelder:

Ta gjerne kontakt i forkant av oppgavevalg for utfyllende informasjon.

## H Risikoanalyse

Risikoen blir regnet ut i fra to faktorer. Den ene er hvor sannsynlig det er at hendelsen inntreffer. Under ser man at det er fire nivåer av denne sannsynligheten. Den andre faktoren er konsekvensen av at hendelsen inntreffer. Ut ifra disse faktorene kommer man frem til et risikonivå som man ser under. Her er det fire ulike nivåer av denne.

Den generelle formelen for risiko er: Sannsynlighet x Konsekvens = risiko.

| Sannsynlighet |
|---------------|
| Svært høy     |
| Høy           |
| Moderat       |
| Lav           |

| Risikonivå |
|------------|
|            |
|            |
|            |
|            |

| Risikoanalyse |                                                            |               |                                                                                                                                                                    |        |                                                                                                                                                                                 |
|---------------|------------------------------------------------------------|---------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Nr:           | Beskrivelse                                                | Sannsynlighet | Konsekvens                                                                                                                                                         | Risiko | Anbefalt tiltak                                                                                                                                                                 |
| 1             | Dårlig forståelse av domenet                               | Moderat       | Utviklere forstår ikke hvordan prosessene fungerer og utvikler en dårlig løsning                                                                                   |        | Nøye gjennomgang av domenet tidlig i prosjektet.                                                                                                                                |
| 2             | Dårlig kravspesifikasjon                                   | Moderat       | Systemet vil ikke bli slik kunden ønsker                                                                                                                           |        | Mange møter i starten av prosjektet mellom utviklere og oppdragsgiver og møter underveis gjennom prosjektet for å sikre en felles forståelse av kravene.                        |
| 3             | Dårlig oppmøte på sprintreview-møtene av utviklingsteamet. | Lav           | Oppdragsgiver og veileder får en dårlig forståelse av hvor langt utviklingsteamet har kommet. Kan ikke komme med gode tilbakemeldinger for videre arbeid fremover. |        | Utviklingsteamet må være motiverte når det kommer til utviklingen og må ha et ønske om å vise frem produktet og få tilbakemeldinger. Et godt miljø mellom partene må etableres. |

|   |                                                                                                                 |     |                                                                                                         |  |                                                                                                                                                                                                                                                                                           |
|---|-----------------------------------------------------------------------------------------------------------------|-----|---------------------------------------------------------------------------------------------------------|--|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 4 | Dårlig oppmøte på sprintreview-møtene av oppdragsgiver.                                                         | Høy | Utviklingsteamet får ikke den kritiske tilbakemeldingen de trenger for å sikre kvaliteten på produktet. |  | Et godt miljø mellom partene må etableres. Om oppdragsgiver får et godt inntrykk av utviklingsteamet slik at de ser fortjenesten i å møte opp slik at produktet kan bli så godt som mulig. Møtene gjennom prosjektet bør også planlegges på starten slik at de kan sette av disse timene. |
| 5 | Dårlig oppmøte på sprintreview-møtene av veileder.                                                              | Lav | Utviklingsteamet får ikke svar på spørsmålene de sitter med, samt hjelp til det de lurer på.            |  | Et godt miljø må etableres mellom partene og møtene må planlegges i starten av perioden slik at veileder kan sette av disse timene.                                                                                                                                                       |
| 6 | Manglende kunnskaper til å klare å utvikle den ønskede løsningen.                                               | Lav | Systemet blir ikke fullført / optimalt.                                                                 |  | Benytte seg av et tverrfaglig utviklingsteam som har den nødvendige kunnskapen.                                                                                                                                                                                                           |
| 7 | Dårlig ressursallokering. Utviklingsteamet jobber kun med deler av løsningen og glemmer andre vesentlige deler. | Lav | Deler av prosjektet blir oversett og ikke fullført.                                                     |  | Utviklingsteamet må hele veien ha en god oversikt over helheten av prosjektet for å sikre at alt blir gjort og at ikke kjedelige/vanskelige viktige deler blir nedprioritert hele veien.                                                                                                  |

|    |                                                                                              |         |                                                                                                                                                                                                                                                                                         |  |                                                                                                                                                                                                                                                                                                                                                       |
|----|----------------------------------------------------------------------------------------------|---------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 8  | Utviklingsteamet tar på seg for mye og prosjektet blir uoversiktlig og umulig å gjennomføre. | Moderat | Prosjektet vil aldri bli funksjonelt, det vil være mye løse tråder i systemet og det vil ikke henge sammen. Utviklingsteamet vil miste moralen etter hvert som de ser at de ikke har kontroll og oppdragsgiver vil bli misfornøyd og det blir en dårlig stemning mellom interessentene. |  | Først må man skaffe seg en helhetlig oversikt over hva som skal lages, deretter må interessentene komme til en enighet om hvilke deler som skal lages og hvilke som skal lages ved en senere anledning. Det er essensielt å bryte ned produktet i små håndterlige oppgaver som man forstår og vet cirka hvor lang tid tar. Disse må også prioriteres. |
| 9  | Oppdragsgiver kommer med ny og kritisk informasjon for sent ut i prosjektet.                 | Høy     | Utviklingsteamet vil få for dårlig tid til å ta hensyn til denne informasjonen og systemet vil ikke bli optimalt. I værste fall kan slik informasjon gjøre det påbegynte systemet ubrukelig og teamet må begynne på nytt.                                                               |  | Etablere et godt miljø i starten av perioden og holde hyppige møter i starten slik at alle parter sitter med den nødvendige informasjonen. Oppdragsgiver bør møte opp så ofte som mulig slik at de kan komme med ny informasjon så fort som mulig om det dukker opp nye ting.                                                                         |
| 10 | Utviklingsteamet benytter seg ikke maksimalt av tiden gitt til prosjektet.                   | Lav     | Systemet blir ikke fullført eller alt for dårlig for oppdragsgiver.                                                                                                                                                                                                                     |  | Etablere en god arbeidstruktur tidlig å prosjektet og holde denne strukturen gjennom hele prosjektet. Et godt arbeidsmiljø må etableres innad i utviklingsteamet slik at alle vil dra på jobb.                                                                                                                                                        |

|    |                                                                                                        |         |                                                                                                                                                                                                                                       |  |                                                                                                                                                                         |
|----|--------------------------------------------------------------------------------------------------------|---------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 11 | Valg av teknologi var ikke optimalt for denne typen løsning.                                           | Moderat | Systemet blir sub-optimalt. Dette kan bety at det fort blir utdatert eller at de ikke bruker den mest optimale teknologien for at systemet skal kjøre godt og effektivt.                                                              |  | Utviklingsteamet må gjøre grundig undersøkelser på hvilke teknologier som er tilgjengelig og hvem som egner seg best til dette systemet. Her bør veilederen inkluderes. |
| 12 | Dårlig kodedokumentasjon som gjør det vanskelig for The Robbins Company å videreutvikle applikasjonen. | Lav     | The Robbins Company må bruke mer ressurser på å videreutvikle produktet og totale kostnaden vil øke mye.                                                                                                                              |  | Utviklingsteamet etablerer en god struktur for kodedokumentasjon fra starten.                                                                                           |
| 13 | Utviklingsteam holder ikke oppdragsgiver og veileder oppdaterte med sitt arbeid.                       | Lav     | Oppdragsgiver vil ikke kunne gi den ønskede tilbakemeldingene og vet ikke hvordan programmet sitt blir. Veileder vil få et dårlig inntrykk og vil ikke kunne hjelpe.                                                                  |  | Utviklingsteamet må gjøre et god forarbeid før prosjektstart og etablere en god struktur for å holde interessentene oppdaterte. Dette gjennom møter og samtaler utenom. |
| 14 | Utviklingsteam følger ikke den valgte utviklingsprosessen Scrum og unngår å gjøre gitte seremonier.    | Moderat | Det er en god grunn til valget av Scrum, så om dette ikke følges blir sluttproduktet fort ikke slikt kunden ønsker. Unngår teamet å gjøre spesifikke seremonier kan dette gå utover forbedringsevnen og teamet stagnerer i utvikling. |  | Utviklingsteamet må sette gode rutiner på de ulike seremoniene og sørge for å opprettholde de og gjennomføre de på en ordentlig måte.                                   |
| 15 | Interessentene er dårlig forberedt til møtene.                                                         | Lav     | Ingen av interessentene får utbytte av møtene som er veldig kritiske for å oppnå et suksessfullt prosjekt for alle parter.                                                                                                            |  | Utviklingsteamet må gjøre gode møteforberedelser og sende ut møteinnkalling i god tid, slik at mottagerne vet hva de skal forberede seg til.                            |

Risikoanalyse

## Risikokart

Nummereringen er hentet fra risikoanalysetabellen.

|                      |               |          |     |        |       |
|----------------------|---------------|----------|-----|--------|-------|
| <b>Sannsynlighet</b> | 4             |          |     |        |       |
|                      | 3             |          | 4   |        | 9     |
|                      | 2             |          |     | 11,14  | 1,2,8 |
|                      | 1             | 10,13,15 | 3,5 | 6,7,12 |       |
|                      |               |          |     |        |       |
|                      |               | 1        | 2   | 3      | 4     |
|                      | <b>Risiko</b> |          |     |        |       |

Ut ifra dette kan utviklingsteamet få en god oversikt over hvilke risikoer de står ovenfor og hvilke som må fokuseres på. Om tiltakene følges skal prosjektet kunne fullføres uten store problemer. Risikokartet viser at utviklingsteamet må passe ekstra på risikoelement 9,1,2 og 8.



# I Retrospective

## Retrospective Sprint 1

Som nevnt i hovedrapporten ble det ikke gjennomført noe retrospective for sprint 1.

## Retrospective Sprint 2

Prosjektdeltakere tilstede: Mats Elias Davidsen, Kristian Nybakken Lund og Knut Wiig.

Prosjektdeltakere frafall: Ingen.

Dato: 22.02.19

### Hva gikk bra under sprinten?

- Det var en god fordeling av roller under denne sprinten. Her kom vi igang med backend, frontend og selve hovedrapporten.

### Hva kunne vært bedre?

- Bedre planlegging av sprint-backlog, vi endte opp med å ha flere kort igjen på "To do" når sprinten var fullført.
- Spesifisere kortene som brukes i sprint-backloggen bedre, slik at disse gjenspeiler arbeidsoppgavene som skal gjøres på en bedre måte.

### Hva vil vi gjøre annerledes til neste sprint?

- Her vil vi starte sprint 3 med en nøye gjennomgang av hva vi skal gjøre i denne sprinten og få det ned på detaljerte kort for sprint-backloggen.
- Få en mer helhetlig oversikt over de kommende sprintene.

## Retrospective Sprint 3

Prosjektdeltakere tilstede: Kristian Nybakken Lund og Knut Wiig.

Prosjektdeltakere frafall: Mats Elias Davidsen

Dato: 01.03.19

### Hva gikk bra under sprinten?

- Planleggingen av selve sprinten
- Oppsett av sprint backlog på trello
- Arbeidsfordelingen er bra, effektiv jobbing
- En felles enighet om veien videre for designet av produktet og prototype versjon 2

### Hva kunne vært bedre?

- Sprinten var litt for kort, men dette er noe vi ikke fikk gjort så mye med på grunn av eksamen i TDAT 3002 Systemtenkning med økonomi.
- Kunne samtidig vært mer effektive til å utvikle produktet samtidig som vi utformet en god arbeidsstruktur for de kommende sprintene.

### **Hva vil vi gjøre annerledes til neste sprint?**

- Ha større fokus mot selve produktet og ta med oss arbeidsstrukturen som ble dannet under denne sprinten. Om vår nye arbeidsstruktur fungerer vil vi forhåpentligvis unngå mye usikkerheter rundt dokumentasjon av hva man gjør og når ulike aktiviteter skal gjøres.

### **Retrospective Sprint 4**

Prosjektdeltakere tilstede: Mats Elias Davidsen, Kristian Nybakken Lund og Knut Wiig.

Prosjektdeltakere frafall: ingen.

Dato: 26.03.19

### **Hva gikk bra under sprinten?**

- Vi var veldig effektive når vi jobbet med applikasjonen, som førte til at vi fikk gjort nesten alt vi planlagte å gjøre i denne sprinten.
- Vi hadde en god kommunikasjon innad i teamet.
- planlegging av sprinten, bedre lapper på trello.
- bruk av branches.
- Fikk kjøpt testtablett.
- Vi fikk gjort mer enn det vi planlagte, dette inkluderer blant annet skriving til fil.

### **Hva kunne vært bedre?**

- Vi kan bli enda bedre på å planlegge neste sprint, selv om planleggingen var bedre i sprint 4 enn i sprint 3.
- Kjøre mer tester av brukbarhet på tab.
- Tidligere ble det nevnt at vi fikk gjort mer enn planlagt. Dette burde vi ha klart å forutsatt og planlagt i starten av sprinten. Det kom naturlig å gjennomføre disse oppgavene her.

### **Hva vil vi gjøre annerledes til neste sprint?**

- Teste mer på tab
- Planlegge sprinten bedre, med tanke på oppgavene som kan gjøres. Splitte trellokortene i mindre oppgaver.

### **Retrospective Sprint 5**

Prosjektdeltakere tilstede: Mats Elias Davidsen, Kristian Nybakken Lund og Knut Wiig.

Prosjektdeltakere frafall: ingen.

Dato: 05.04.19

### **Hva gikk bra under sprinten?**

- Teamet hadde en god sprintplanlegging. God nedbrytning av arbeidsoppgaver til trello-boardet.
- Teamet jobbet effektivt med sine respektive arbeidsoppgaver.
- God forberedt til sprintreview-møtene med oppdragsgiver og veileder, som resulterte i mye god feedback.
- Denne sprinten varte i 2 uker, dette syntes teamet var en god lengde, da man kom ordentlig godt inn i sine oppgaver.

### **Hva kunne vært bedre?**

- Bli flinkere til å estimere tidsforbruk til hver arbeidsoppgave.
- Være mer oppmerksom på å sette lappene på "Doing" når vi jobber med denne oppgaven, for å vise de andre på gruppa hva vi gjør

### **Hva vil vi gjøre annerledes til neste sprint?**

- Bruke trelloboardet mer aktivt.

## **Retrospective Sprint 6**

Prosjektdeltakere tilstede: Mats Elias Davidsen, Kristian Nybakken Lund og Knut Wiig.

Prosjektdeltakere frafall: ingen.

Dato: 12.04.19

### **Hva gikk bra under sprinten?**

- Estimering av hver arbeidsoppgave.
- Teamet jobbet effektivt med sine respektive arbeidsoppgaver, noe som resulterte i at målet med å bli ferdig med TBM Inspector ble nådd.
- benyttet trelloboardet mer aktivt, satt lappene på de riktige plassene hele veien.

### **Hva kunne vært bedre?**

- Bli flinkere til at alle møter opp kl 08:00 og komme hurtig i gang med oppgavene ved starten av dagen.

### **Hva vil vi gjøre annerledes til neste sprint?**

- bli mer effektive på starten av dagen.

## **Retrospective Sprint 7**

Prosjektdeltakere tilstede: Mats Elias Davidsen, Kristian Nybakken Lund og Knut Wiig.

Prosjektdeltakere frafall: ingen.

Dato: 03.05.19

### **Hva gikk bra under sprinten?**

- Programmet er nå endelig operativt og ferdig. Det er kun igjen små puss.
- Hovedrapporten begynner å bli ferdig. Alle kap har mye innhold og en god struktur.
- Fikk en god oversikt over alt som skal leveres og gjøres til siste sprint. Dette var viktig at det var veldig mye forskjellig som skal inn og det er viktig at alt blir levert.

### **Hva kunne vært bedre?**

- Under denne sprinten fant vi ikke noe spesifikt vi burde ha gjort bedre.

### **Hva vil vi gjøre annerledes til neste sprint?**

- Fortsette slik vi gjorde denne sprinten.

## **Retrospective Sprint 8**

### **Hva gikk bra under sprinten?**

- Teamet hadde svært god arbeidsmoral og satt flere sene kvelder og jobbet med prosjektet.
- Programmet, både frontend og backend har blitt ferdigstilt
- Kom i mål med alt som skulle leveres i god tid før innlevering, slik at vi kunne bruke tid på å lese over oppgaven.

### **Hva kunne vært bedre?**

- Ingenting konkret som kunne forbedret, alt i alt en svært god sprint.

### **Hva vil vi gjøre annerledes til neste sprint?**

- Dette var siste sprint, så dette vil ikke være relevant å se på. Hvertfall med tanke på at dette også var en svært godt gjennomført sprint.

# J Timelister

Det er totalt 16 perioder for dette prosjektet. Disse periodene inneholder hvilke arbeidsoppgaver hvert enkelt gruppemedlem har gjort. Her står det når de spesifikke arbeidsoppgavene ble gjort, og hvor lang tid det tok. Totalt antall timer:

- Mats Elias Davidsen: 488,5 timer.
- Kristian Nybakken Lund: 490 timer.
- Knut Wiig: 488 timer.

## Periode 1

| Periode 1                |       |                        |                      |       |                                              |                      |       |                                              |  |  |  |
|--------------------------|-------|------------------------|----------------------|-------|----------------------------------------------|----------------------|-------|----------------------------------------------|--|--|--|
| Timer Kristian Sum: 26,5 |       |                        | Timer Knut Sum: 26,5 |       |                                              | Timer Mats Sum: 16,5 |       |                                              |  |  |  |
| Dato                     | Timer | Kommentar              | Dato                 | Timer | Kommentar                                    | Dato                 | Timer | Kommentar                                    |  |  |  |
| 16.01.2019               | 2     | Planlegging oppstart   | 16.01.2019           | 2     | Planlegging for oppstartsmøte                | 16.01.2019           | 2     | Planlegging for oppstartsmøte                |  |  |  |
| 19.01.2019               | 2     | Oppstartsmøte          | 19.01.2019           | 2     | Oppstartsmøte                                | 19.01.2019           | 2     | Oppstartsmøte                                |  |  |  |
| 22.01.2019               | 1     | Se på tabs             | 22.01.2019           | 1     | Research for hardware                        | 22.01.2019           | 1     | Research av hardware                         |  |  |  |
| 23.01.2019               | 3,5   | Jobbe med Backlog      | 23.01.2019           | 3,5   | Utarbeidning av backlog                      | 23.01.2019           | 3,5   | Utarbeide backlogg                           |  |  |  |
| 23.01.2019               | 0,5   | Forb. til skypermøte.  | 23.01.2019           | 0,5   | forberedelse til møte med ingeniør i Seattle | 23.01.2019           | 0,5   | Forberedelse til møte med ingeniør i Seattle |  |  |  |
| 24.01.2019               | 8,5   | Jobbe med Backlog      | 24.01.2019           | 8,5   | Utarbeidet backlog og userstories            | 24.01.2019           | 7,5   | Utarbeide backlog og user stories            |  |  |  |
| 25.01.2019               | 2,5   | Møte med kunde         | 25.01.2019           | 2,5   | møte med oppdragsgiver                       |                      |       |                                              |  |  |  |
| 25.01.2019               | 4,5   | Visjonsdokument        | 25.01.2019           | 4,5   | Visjonsdokument                              |                      |       |                                              |  |  |  |
| 25.01.2019               | 1,5   | Møteforberedelser      | 25.01.2019           | 1,5   | Møteforberedelser + research                 |                      |       |                                              |  |  |  |
| 25.01.2019               | 0,5   | Skype-møte med Seattle | 25.01.2019           | 0,5   | Skypemøte med utvikler i Seattle             |                      |       |                                              |  |  |  |

## Periode 2

| PERIODE 2              |       |                           |                      |       |                                                                    |                    |       |                                                                   |  |  |  |
|------------------------|-------|---------------------------|----------------------|-------|--------------------------------------------------------------------|--------------------|-------|-------------------------------------------------------------------|--|--|--|
| Timer Kristian Sum: 12 |       |                           | Timer Knut Sum: 29,5 |       |                                                                    | Timer Mats Sum: 25 |       |                                                                   |  |  |  |
| Dato                   | Timer | Kommentar                 | Dato                 | Timer | Kommentar                                                          | Dato               | Timer | Kommentar                                                         |  |  |  |
| 31.01.2019             | 7,5   | User stories og DDD.      | 28.01.2019           | 2,5   | Skype-møte med oppdragsgiver, Sindre Log, angående videre framgang | 28.01.2019         | 2,5   | Skype-møte med oppdragsgiver, Sindre Log angående videre framgang |  |  |  |
| 01.02.2019             | 4,5   | Jobbe med visjonsdokument | 29.01.2019           | 1,5   | Møte med oppdragsgiver Amund Bruland                               | 29.01.2019         | 1,5   | Møte med oppdragsgiver, Amund Bruland                             |  |  |  |
|                        |       |                           | 29.01.2019           | 6     | Videre utarbeidning med user stories og scenarier                  | 29.01.2019         | 6     | Videre arbeid med user stories og scenarier                       |  |  |  |
|                        |       |                           | 30.01.2019           | 7,5   | Low fidelity prototyping og videreutvikling av user stories        | 30.01.2019         | 7,5   | Low fidelity prototyping og videreutvikling av user stories       |  |  |  |
|                        |       |                           | 31.01.2019           | 7,5   | Utvikling av domenemodell                                          | 31.01.2019         | 7,5   | Low fidelity prototyping og utvikling av domenemodell             |  |  |  |
|                        |       |                           | 01.02.2019           | 4,5   | Low fidelity prototyping og visjonsdokument                        |                    |       |                                                                   |  |  |  |

## Periode 3

| PERIODE 3                |       |                                                         |                    |       |                                         |                      |       |                                                                    |  |  |  |
|--------------------------|-------|---------------------------------------------------------|--------------------|-------|-----------------------------------------|----------------------|-------|--------------------------------------------------------------------|--|--|--|
| Timer Kristian Sum: 24,5 |       |                                                         | Timer Knut Sum: 11 |       |                                         | Timer Mats Sum: 24,5 |       |                                                                    |  |  |  |
| Dato                     | Timer | Kommentar                                               | Dato               | Timer | Kommentar                               | Dato                 | Timer | Kommentar                                                          |  |  |  |
| 05.02.2019               | 2     | problemløsningsworkshop med skolen                      | 05.02.2019         | 2     | Problemløsningsmøte med skolen          | 05.02.2019           | 2     | Problemløsningsmøte med skolen                                     |  |  |  |
| 05.02.2019               | 4,5   | visjonsdokument + oppsett av latex for rapport.         | 05.02.2019         | 4     | visjonsdokument + div oppsett           | 05.02.2019           | 4,5   | Jobbet med visjonsdokument og oppsett av diverse hjelpemidler      |  |  |  |
| 06.02.2019               | 2,5   | Gjorde ferdig akkefunksjonellekrav i visjonsdokumentet. | 08.02.2019         | 1     | Møte med Helge (veiledermøte)           | 06.02.2019           | 2,5   | Gjorde ferdig forutsetninger og avhengigheter i visjonsdokumentet. |  |  |  |
| 06.02.2019               | 5     | Testet ut electron og react/native                      | 08.02.2019         | 2,5   | Møte innkalling og referat mail i LaTeX | 06.02.2019           | 5     | Testet ut electron og React/native                                 |  |  |  |
| 07.02.2019               | 5,5   | Leking med React og oppsett LaTeX hovedrapport.         | 08.02.2019         | 1,5   | Møte med Kunde og Helgeivar.            | 07.06.2019           | 5,5   | Skaffet mer erfaring innen React                                   |  |  |  |
| 08.02.2019               | 1     | Møte med Helge (veiledermøte)                           |                    |       |                                         | 08.02.2019           | 1     | Møte med Helge (veiledermøte)                                      |  |  |  |
| 08.02.2019               | 2,5   | Møte innkalling og referat mail i LaTeX                 |                    |       |                                         | 08.02.2019           | 2,5   | Møte innkalling og referat mail i LaTeX                            |  |  |  |
| 08.02.2019               | 1,5   | Møte med Kunde og Helgeivar.                            |                    |       |                                         | 08.02.2019           | 1,5   | Møte med Kunde og Helgeivar.                                       |  |  |  |

## Periode 4

| PERIODE 4             |       |                             |                   |       |                          |                   |       |                          |  |  |  |
|-----------------------|-------|-----------------------------|-------------------|-------|--------------------------|-------------------|-------|--------------------------|--|--|--|
| Timer Kristian Sum: 6 |       |                             | Timer Knut Sum: 6 |       |                          | Timer Mats Sum: 6 |       |                          |  |  |  |
| Dato                  | Timer | Kommentar                   | Dato              | Timer | Kommentar                | Dato              | Timer | Kommentar                |  |  |  |
| 15.02.2019            | 6     | Oppsett av gitrepo og react | 15.02.2019        | 3     | database-oppsett         | 15.02.2019        | 3     | database-oppsett         |  |  |  |
|                       |       |                             | 15.02.2019        | 3     | oppsett gitrepo og react | 15.02.2019        | 3     | oppsett gitrepo og react |  |  |  |

Denne perioden var det obligatorisk grundercamp.

## Periode 5

| PERIODE 5                |       |                                                      |                      |       |                                                    |                      |       |                                                                  |  |  |  |
|--------------------------|-------|------------------------------------------------------|----------------------|-------|----------------------------------------------------|----------------------|-------|------------------------------------------------------------------|--|--|--|
| Timer Kristian Sum: 37,5 |       |                                                      | Timer Knut Sum: 37,5 |       |                                                    | Timer Mats Sum: 37,5 |       |                                                                  |  |  |  |
| Dato                     | Timer | Kommentar                                            | Dato                 | Timer | Kommentar                                          | Dato                 | Timer | Kommentar                                                        |  |  |  |
| 18.02.2019               | 7,5   | Electron og react. Backendstruktur.                  | 18.02.2019           | 1     | ER modell                                          | 18.02.2019           | 1     | Jobbet med ER model                                              |  |  |  |
| 19.02.2019               | 7,5   | CI med Travis og CodeCov. KOblet backend opp mot db. | 18.02.2019           | 3     | Dokumentasjon, estimering av prosjekt              | 18.02.2019           | 6,5   | Utforsket "Onsen UI" framework for bruk i vår ReactJS frontend   |  |  |  |
| 20.02.2019               | 7,5   | Backendstruktur, Automatisk enhetstesting med CI.    | 18.02.2019           | 3,5   | Oppsett av database + div                          | 19.02.2019           | 7,5   | Laget utkast til innloggingside og side der man velger TBM       |  |  |  |
| 21.02.2019               | 7,5   | backend, autentisering med bcrypt.                   | 19.02.2019           | 7,5   | Jobbing med hovedrapport og møteforberedelser      | 20.02.2019           | 2,5   | Lærte meg React Router for å bruke dette i frontend              |  |  |  |
| 22.02.2019               | 1     | sprint review med veileder                           | 20.02.2019           | 0,5   | møteforberedelser                                  | 20.02.2019           | 5     | Satt opp koblinger mellom sidene i appikasjonen med React Router |  |  |  |
| 22.02.2019               | 1,5   | sprint review med kunde                              | 20.02.2019           | 1     | hvello sprint backlog oppsett                      | 21.02.2019           | 3     | Redigerte Visjonsdokument i LaTeX                                |  |  |  |
| 22.02.2019               | 5     | retrospective etter sprintreview                     | 20.02.2019           | 6     | hovedrapport                                       | 21.02.2019           | 4,5   | Lage komponenten "Toolbar" til bruk i programvaren               |  |  |  |
|                          |       |                                                      | 21.02.2019           | 7,5   | Hovedrapport, møteforberedelser og database        | 22.02.2019           | 1     | Møte med veileder                                                |  |  |  |
|                          |       |                                                      | 22.02.2019           | 1     | sprintreview med veileder                          | 22.02.2019           | 1,5   | Møte med oppgavestiler                                           |  |  |  |
|                          |       |                                                      | 22.02.2019           | 1,5   | sprintreview med oppdragsgiver                     | 22.02.2019           | 5     | Arbeid med retrospektiv. For- og etter arbeid                    |  |  |  |
|                          |       |                                                      | 22.02.2019           | 5     | forarbeid og etterarbeid med møte og retrospektive |                      |       |                                                                  |  |  |  |



## Periode 12

| PERIODE 12, påske |      |  |   |            |     |  |   |            |     |               |
|-------------------|------|--|---|------------|-----|--|---|------------|-----|---------------|
| Timer Kristia     | Sum: |  | 0 | Timer Knut | Sum |  | 0 | Timer Mats | Sum | 8             |
|                   |      |  |   |            |     |  |   | 15.04.2019 | 4   | Dokumentasjon |
|                   |      |  |   |            |     |  |   | 16.04.2019 | 3   | Dokumentasjon |
|                   |      |  |   |            |     |  |   | 21.04.2019 | 1   | Dokumentasjon |

Denne perioden var det påske.

## Periode 13

| PERIODE 13    |      |                                                                                            |    |            |     |                                                                      |    |            |     |                                                            |
|---------------|------|--------------------------------------------------------------------------------------------|----|------------|-----|----------------------------------------------------------------------|----|------------|-----|------------------------------------------------------------|
| Timer Kristia | Sum: |                                                                                            | 32 | Timer Knut | Sum |                                                                      | 32 | Timer Mats | Sum | 32                                                         |
| 23.04.2019    | 7,5  | Oppretting av prosjekter                                                                   |    | 23.04.2019 | 7,5 | hovedrapport, resultat, diskusjon, sprintplanlegging.                |    | 23.04.2019 | 7,5 | Utvikling av frontend for ibm supervisor                   |
| 24.04.2019    | 4    | Docker og docker-compose                                                                   |    | 24.04.2019 | 7,5 | dokumentasjon hovedrapport                                           |    | 24.04.2019 | 7,5 | Kolling av frontend og backend for ibm supervisor          |
| 24.04.2019    | 3,5  | Oppsett av server -> cutware.ibm.rtnu.no                                                   |    | 25.04.2019 | 8,5 | dokumentasjon hovedrapport og administrasjon                         |    | 25.04.2019 | 8,5 | Utvikling av frontend for Admin og koble dette mot backend |
| 25.04.2019    | 8,5  | Kjører server med HTTPS og docker-compose, må begynne å dokumentere hvordan man setter opp |    | 26.04.2019 | 8,5 | dokumentasjon resultat + finne ut av alt som skal gjøres i sprint 8. |    | 26.04.2019 | 8,5 | Utvikling av funksjonalitet for SuperUser brukerguppen     |
| 26.04.2019    | 8,5  | packing av app, må gjøres med docker, får feilmeldt av en dependencie                      |    |            |     |                                                                      |    |            |     |                                                            |

## Periode 14

| PERIODE 14    |      |                                                |      |            |     |                                               |      |            |     |                                                                            |
|---------------|------|------------------------------------------------|------|------------|-----|-----------------------------------------------|------|------------|-----|----------------------------------------------------------------------------|
| Timer Kristia | Sum: |                                                | 42,5 | Timer Knut | Sum |                                               | 42,5 | Timer Mats | Sum | 43,5                                                                       |
| 29.04.2019    | 8,5  | Merge packager med master, pakket og bygd app. |      | 29.04.2019 | 8,5 | hovedrapport og kravdokumentasjonsrapport.    |      | 29.04.2019 | 8,5 | Fikse mindre bugs og jobbe med at supervisor skal se inspeksjoner          |
| 29.04.2019    | 1    | Electron-builder funker med windows.           |      | 30.04.2019 | 3   | administrativt                                |      | 30.04.2019 | 8,5 | Fikse alt med registrering av brukere, slik at brukeren får tilbakemelding |
| 30.04.2019    | 7,5  | Jobbe med CI                                   |      | 30.04.2019 | 5,5 | hovedrapport                                  |      | 01.05.2019 | 8,5 | Teste applikasjonen for å finne bugs, for så å fikse disse                 |
| 01.05.2019    | 9,5  | dokumentasjon og ferdigstille server           |      | 01.05.2019 | 8,5 | kravspekkdokumentasjon                        |      | 02.05.2019 | 9,5 | Fikse på siste rest av UI i applikasjonen                                  |
| 02.05.2019    | 7,5  | Tester og refactor controller til moduler      |      | 02.05.2019 | 8,5 | hovedrapport                                  |      | 03.05.2019 | 1   | Møte med veileder                                                          |
| 02.05.2019    | 1    | se på bug i packager                           |      | 03.05.2019 | 1   | møte veileder                                 |      | 03.05.2019 | 1,5 | Møte med oppgavestiler                                                     |
| 03.05.2019    | 0,5  | møte med veileder                              |      | 03.05.2019 | 1,5 | møte oppdragsgiver                            |      | 03.05.2019 | 6   | Fjerne unødvendig kode og gå gjennom alle filer tilhørende applikasjonen   |
| 03.05.2019    | 1    | møte med kunde - levering                      |      | 03.05.2019 | 6   | forberedelse til møter, bearbeiding av møtene |      |            |     |                                                                            |
| 03.05.2019    | 6    | tester                                         |      |            |     |                                               |      |            |     |                                                                            |

## Periode 15

| PERIODE 15    |      |                                                 |      |            |     |                                                                                                               |      |            |     |                                                                   |
|---------------|------|-------------------------------------------------|------|------------|-----|---------------------------------------------------------------------------------------------------------------|------|------------|-----|-------------------------------------------------------------------|
| Timer Kristia | Sum: |                                                 | 52,5 | Timer Knut | Sum |                                                                                                               | 43,5 | Timer Mats | Sum | 44                                                                |
| 06.05.2019    | 8,5  | Småttik og er-domene modell.                    |      | 06.05.2019 | 8,5 | ferdigstille kravspekken, reviderte domenenmodell og ER-modell.                                               |      | 06.05.2019 | 4   | Gå over kode for frontend og legge inn gode kommentarer osv.      |
| 07.05.2019    | 8,5  | Resultater og diskusjon. Begynte med systemdok. |      | 07.05.2019 | 8,5 | Gjorde ferdig ingenting og administrative resultat, satt inn retrospectives og planta diskusjon og konklusjon |      | 06.05.2019 | 4,5 | Revidere domenenmodell og ER-modell                               |
| 07.05.2019    | 1,5  | Rise noen db tester                             |      | 08.05.2019 | 8,5 | hovedrapport diskusjon                                                                                        |      | 07.05.2019 | 8,5 | Skrive på vitenskapelig resultat i resultatdelen til hovedoppaven |
| 08.05.2019    | 8,5  | systemdokumentasjon                             |      | 09.05.2019 | 8,5 | hovedrapport diskusjon                                                                                        |      | 08.05.2019 | 8,5 | Jobbe med diskusjon                                               |
| 09.05.2019    | 9    | diskusjon                                       |      | 10.05.2019 | 8,5 | hovedrapport konklusjon                                                                                       |      | 09.05.2019 | 8,5 | Jobbe med diskusjon                                               |
| 10.05.2019    | 8,5  | systemdokumentasjon og konklusjon               |      | 10.05.2019 | 1   | retting av teori.                                                                                             |      | 10.05.2019 | 8,5 | Jobbe med konklusjon og slåplukk i applikasjonen                  |
| 12.05.2019    | 8    | kode dokumentasjon + sysdok                     |      |            |     |                                                                                                               |      | 12.05.2019 | 1,5 | Kodedokumentasjon og bug fixes                                    |

## Periode 16

| PERIODE 16    |      |                                                    |    |            |      |                                                                                  |    |            |      |                                                                   |
|---------------|------|----------------------------------------------------|----|------------|------|----------------------------------------------------------------------------------|----|------------|------|-------------------------------------------------------------------|
| Timer Kristia | Sum: |                                                    | 57 | Timer Knut | Sum  |                                                                                  | 60 | Timer Mats | Sum  | 58                                                                |
| 13.05.2019    | 7,5  | sysdok                                             |    | 13.05.2019 | 8,5  | teori                                                                            |    | 13.05.2019 | 8,5  | Brukermanual                                                      |
| 14.05.2019    | 12   | Systemdokumentasjon, valg av teknologi og metoder. |    | 14.05.2019 | 12   | Visjonsdokument, kap.valg av metode og teknologi, forord, strukturert rapporten. |    | 14.05.2019 | 12   | Brukermanual, fikse småting i kode og valg av teknologi og metode |
| 15.05.2019    | 8,5  | Systemdokumentasjon, valg av teknologi og metoder. |    | 15.05.2019 | 8,5  | Visjonsdokument, valg av teknologi, sammendrag.                                  |    | 15.05.2019 | 8,5  | Skriv sammendrag og lese over andre deler av hovedrapporten       |
| 16.05.2019    | 0,5  | møte med veileder                                  |    | 16.05.2019 | 10,5 | jaløring på hovedrapport.                                                        |    | 16.05.2019 | 10,5 | Jobbe med hovedrapport                                            |
| 16.05.2019    | 10,5 | sysdok, valg av metode, tester, teori, mye annet   |    | 16.05.2019 | 0,5  | møte med veileder                                                                |    | 16.05.2019 | 0,5  | Møte med veileder                                                 |
| 18.05.2019    | 6    | ferdigstille hovedrapporten.                       |    | 18.05.2019 | 2    | presentasjon                                                                     |    | 18.05.2019 | 6    | ferdigstille rapport.                                             |
| 20.05.2019    | 12   | Ferdigstille, lese gjennom og levere bachelor      |    | 19.05.2019 | 6    | ferdigstille rapport                                                             |    | 20.05.2019 | 12   | Ferdigstille rapporten, lese gjennom og levere.                   |
|               |      |                                                    |    | 20.05.2019 | 12   | lese gjennom, ferdigstille og levere bacheloroppaven                             |    |            |      |                                                                   |

## K Further work

During the development of the application the team had ideas about the further development of CutWear. These ideas are listed below:

- Register the shift that recorded the inspection data, this can be determined by the time the inspection took place.
- Determine the TBM that is going to be inspected based on what internet the application is connected to.
- Make graphs for displaying the inspection data, both for supervisors and inspectors.
- Add an overview of the consumption of cutters for the supervisor user. The user should be able to limit the time period for this overview.
- Implement scanning of a chip in the cutter to register the cutter id.
- For offline log in, store which role and project a user belongs to in a better way. This is currently stored in clear text in the users.json file.
- The image in the 'Guidelines' page for an inspector is currently the same for every TBM. This should be loaded when registering a TBM to have the correct image.
- The image of the TBM in the overview page should be uploaded when registering a TBM, in the current solution you need to upload this again every time you inspect a TBM.
- Choose to not enter a new cutter when registering a cutter change, in case the inspector wish to not have a cutter in this position.
- When the application has been without internet connection for a long time, it should include the inspections that has not been uploaded in the decision support.
- With no internet connection the application should use the local files for presenting the previous inspection data.
- Add a Workshop Supervisor user group, this user group is going to be the workers in the workshops that fix cutters so that these can be reused. This includes functionality such as:
  - Register what maintenance that has been performed on a cutter.
  - Check the history of a cutter. This includes maintenance, changes and wear.
- There is no way of changing passwords for the Admin and Super User users that are added in the database by default. This should be a feature in the next version of the program.

## L Trello board

Under utviklingen av applikasjonen benyttet gruppen seg av trello som er et digitalt scrum-board. Dette er et system som hjelper utviklere med å holde orden på arbeidsoppgaver. Her var det en kolonne med 'product backlog' som innehold alle arbeidsoppgaver som gruppen viste om. Til hver sprint ble kortene som man planla å gjøre under den sprinten lagt i kolonne 'sprint backlog'. Deretter ble de flyttet til 'doing' og 'review'. Ved sprints slutt ble kortene på 'review' gjennomgått og de som var ferdige ble satt i 'kolonnen 'sprint done (2-7)'. første bilde viser dette oppsette og bilde nr to og tre viser hvilke kort som ble ferdig til de ulike sprintene.



Bachelor gruppe 29 ☆ Personlig 🗄 Privat **KW MD K 3** Inviter

### Product backlog

- Lage et testcase for å sjekke om alt som blir registrert er riktig.
- US10. Vis frem data samlet fra en TBM
- US12. Registrere et nytt skift.
- US6. Scanning av serienummer på kutter
- US13. Se prosjekthistorie for et gitt prosjekt som en superuser
- US20. Oversikt over alle TBMer som Admin
- US14. Se på historikken til en gitt kutter
- US15. registrere vedlikehold hos verksteder
- US16. oversikt over kutter for å se hva slags vedlikehold som må gjøres
- US24. Manualer for å bruke applikasjon og installere server
- beslutningsstøtte, hvilken chainage, cutter ble byttet på sist og dato for bytte

### Sprint backlog

+ Legg til et kort

### Doing

+ Legg til et kort

### Review

+ Legg til et kort

### done

+ Legg til et kort

#### done sprint 7

- US3. Teste brukervennligheten til Ulet.
- Backend - automatisere testing
- Gå over og se på utseende til appen + fjern unødvendig kode
- Admin skal lage prosjekt med tilhørende supervisor bruker
- Superuser skal få hentet ut informasjon skrevet til fil
- Få informasjon om en TBM i et prosjekt - Supervisor
- Finn ut hvordan du skal håndtere roller og prosjekt ved offline pålogging
- US5. Automatisere reupload av offline inspeksjoner
- US2. Gi informasjon om tidligere kutterslitasje og maks slitasje i en posisjon

+ Legg til enda et kort

#### done sprint 6

- US11, regge ny tbn til prosjekt, backend
- Kommentar på regging av slitasje (ferdig frontend).
- Få inn data fra de 3 siste målingene inne på "Kutter Slitasje"
- Få inn korrekt informasjon på "Kutter inspeksjon" siden
- Oppplattning av en inspeksjon til backend
- Ha med forrige boredmeters, boredhours
- Hent ut inspeksjoner på en tbn
- Endre tekst på ongoing inspection siden
- Bare be om boredhours og boredmeters ved ny inspeksjon.

+ Legg til enda et kort

#### done sprint 5

- Endre popup som kommer når man forsøker å logge ut eller endre TBM
- US21. REST-queries.
- Flytt knappen som gjør at man sletter en inspeksjon, så denne blir litt mer skjult
- Endre fra Abrasive Wear med checklist til at det bare er en kategori til som heter Relocate
- US19 - Ved utlogging fra app, fjern all sensitiv data knyttet til den utloggede brukeren (Token).
- Få inn TBM fra database til "Select TBM" siden
- Få inn korrekt kuttær Id knyttet til en posisjon
- Få inn data på "Inspection Start" siden hvis denne informasjonen allerede finnes
- Når man registrer slitasje/bytte på

+ Legg til enda et kort

#### done sprint 4

- gjør klienten kjørbær (.exe/dmg)
- US1. Frontend - Lag side for registrering av kutter slitasje for en bestemt posisjon
- Legge inn testdata i db
- US8. Frontend - Lage side som tar inn informasjon ved start av inspeksjon
- US7. Frontend - Lag side for registrering av kutterbytte
- Få lagret tilstanden til en inspeksjon på fil ved midlertidig stopp av inspeksjon
- Backend - autentisering
- US1. Frontend - Lage side for en gitt posisjon, der man velger mellom registre slitasje eller bytte

+ Legg til enda et kort

#### done sprint 3

- Få dropdown menyen på toolbaren til å fungere
- US1. Lag frontend for å velge foretrukket TBM for aktuell inspektor
- Backend - CI

+ Legg til enda et kort

#### done sprint 2

- Backend - Prosjektstruktur
- Sprint 2 retrospective
- Lag et utgangspunkt for frontend til innlogginnside
- Møteinnkalling for sprintreview 2
- Hovedrapport, starte på resultat
- Sette opp databasen
- Sette opp React Routing
- Lag utgangspunkt for frontend til side 2 for TBM Inspector
- Lag utkast til toolbar
- Hovedrapport, starte på introduksjon og relevans

+ Legg til enda et kort

| done sprint 7                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | done sprint 6                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           | done sprint 5                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  | done sprint 4                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                | done sprint 3                                                                                                                                                                      | done sprint 2                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>Sette opp server</p> <p>Supervisor skal registrere ny TBM</p> <p>Post REq for nytt prosjekt</p> <p>Registrere inspeksjon</p> <p>Backend - Docker</p> <p>Legg til Machine Hours og Chainage ved registrering av TBM</p> <p>US1. Backend - hente ut all kutterdata for en gitt tbm</p> <p>Merge packager og legge til fsevents i optionaldependencies i klienten.</p> <p>På admin siden bør man ha hentet ut en liste med brukernavn som man kan sjekke opp mot før man registrerer et prosjekt</p> | <p>Endre tekst på ongoing inspection siden</p> <p>Bare be om boredhours og boredmeters ved ny inspeksjon. machinehours og chainage kan regnes ut</p> <p>Varsel på hvilke kuttere som bør byttes etter en wear-registrering.</p> <p>Register new cutters in the project when inserted into tbm</p> <p>sjekke at man har fylt inn boredmeters og boredhours før avslutning av inspeksjon.</p> <p>Fiks så man kan markere at en kutter skal byttes når man registrerer slitasje</p> <p>Registrere max forskjell mellom nabokuttere.</p> <p>Endre chainage, machine hours, boredmeters etc, til double siden det brukes double i excel?</p> | <p>Når man registrer slitasje/bytte på en allerede inspesert posisjon må dette skrive over den gamle informasjonen</p> <p>Få login til å fungere offline</p> <p>Få inn alerts på trykking på alle knapper som resulterer i avslutning av inspeksjon</p> <p>Når det ikke står en kutter i en posisjon - få en egen side der man registrer cutterid tilknyttet posisjonen</p> <p>Last inn state fra fil når man går inn på en TBM med en pauset inspeksjon</p> <p>Legg til en ny alert som kommer når man forsøker å slette en inspeksjon</p> <p>Få koblingen mellom frontend og backend, for å hente ut en liste med TBM'er</p> | <p>US1. Frontend - Lag side for registrering av kutter slitasje for en bestemt posisjon</p> <p>Legge inn testdata i db</p> <p>US8. Frontend - Lage side som tar inn informasjon ved start av inspeksjon</p> <p>US7. Frontend - Lag side for registrering av kutterbytte</p> <p>Få lagret tilstanden til en inspeksjon på fil ved midlertidig stopp av inspeksjon</p> <p>Backend - autentisering</p> <p>US1. Frontend - Lage side for en gitt posisjon, der man velger mellom registre slitasje eller bytte</p> <p>laste opp bilde og få det til å rotere</p> | <p>Få dropdown menyen på toolbaren til å fungere</p> <p>US1. Lag frontend for å velge foretrukket TBM for aktuell inspekter</p> <p>Backend - CI</p> <p>+ Legg til enda et kort</p> | <p>Lag utgangspunkt for frontend til side 1 for TBM Inspector</p> <p>Hovedrapport, starte på oppgavetekst</p> <p>Planlegging av sprintreview 2</p> <p>Finn passende framework til å bruke i React for dette prosjektet</p> <p>Planlegging av sprintaktiviteter</p> <p>Møtereferat for sprintreview 2</p> <p>utarbeide databasemodell (ER-modell)</p> <p>Skaffe informasjon om hvordan skrive hovedrapport</p> <p>Backend - koble opp mot db</p> |
| + Legg til enda et kort                                                                                                                                                                                                                                                                                                                                                                                                                                                                              | + Legg til enda et kort                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 | + Legg til enda et kort                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | + Legg til enda et kort                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |                                                                                                                                                                                    | + Legg til enda et kort                                                                                                                                                                                                                                                                                                                                                                                                                         |

## M User manual

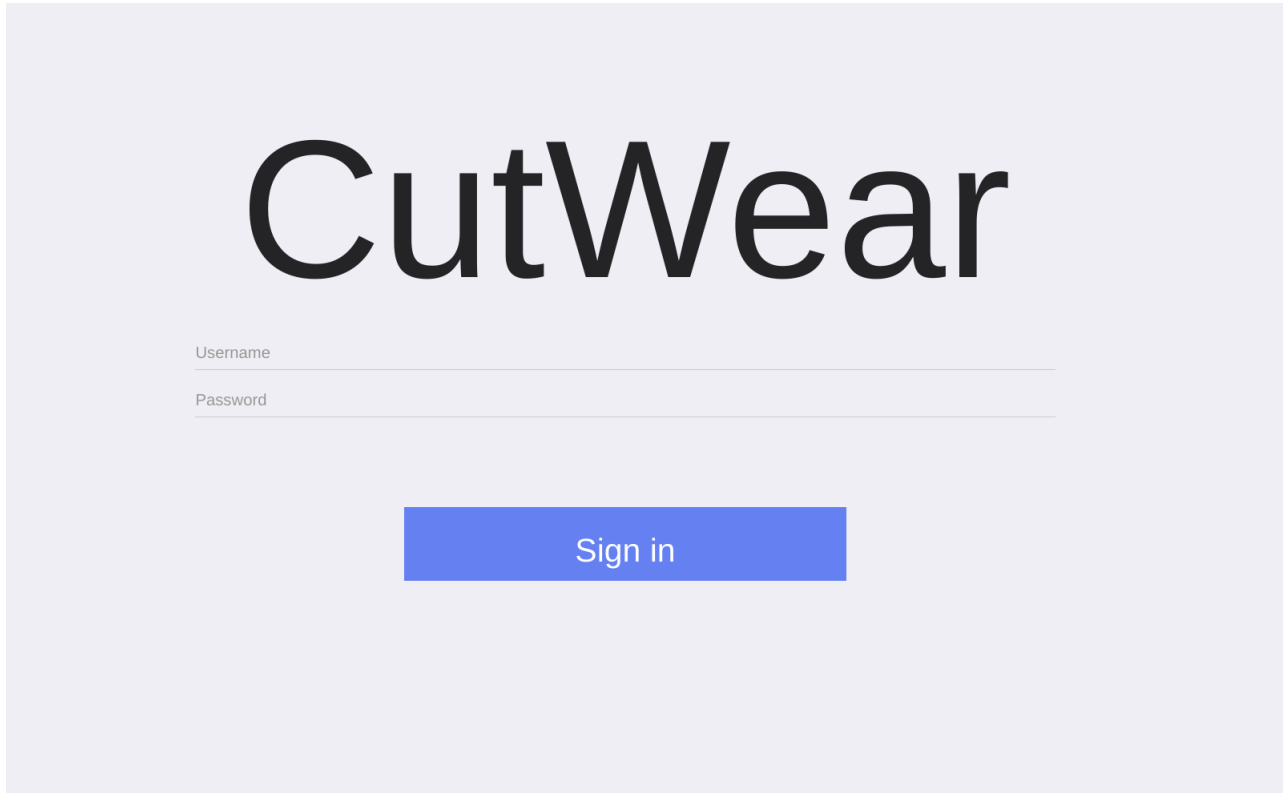
# User Manual - CutWear v1.0

Explaining all features within the CutWear application

20th of May 2019

## Logging in

When you start CutWear you will enter the log in screen where you will have to enter a valid username and password to get access to the application. The user you sign in with will belong to one of the four user groups, the user group your user belongs to will determine the content of the application. CutWear has four different user groups, these are TBM Inspector, Project Supervisor, Super User and Admin. The following chapters will explain the different features of the application within these user groups.



*Figur 1: Login screen*

If you are working on a project and need a user for the application, please contact the following based on what kind of user you need:

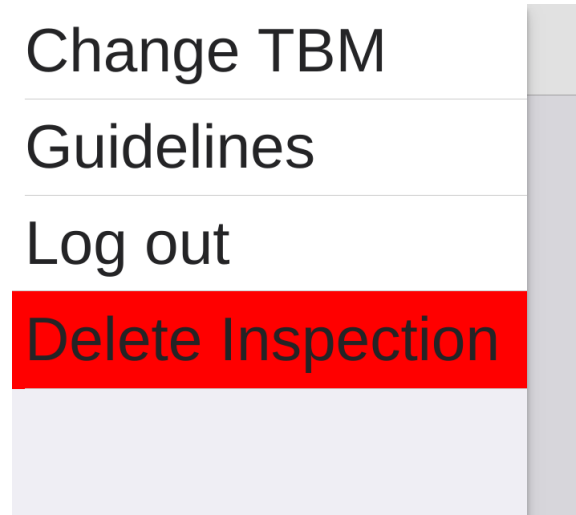
- **TBM Inspector:** Contact a Project Supervisor on your project
- **Project Supervisor:** Contact an Admin

## TBM Inspector

The functionality within this user group is focused on registering data from inspections performed on a TBM. This user manual assume that the users logged in as TBM Inspectors has basic knowledge of TBM's and conduct inspections.

### The toolbar menu

The toolbar of the application displays the name of the site you are on and the current time, in addition to this it has a menu which can be opened by clicking the blue icon in the top left corner of the screen.



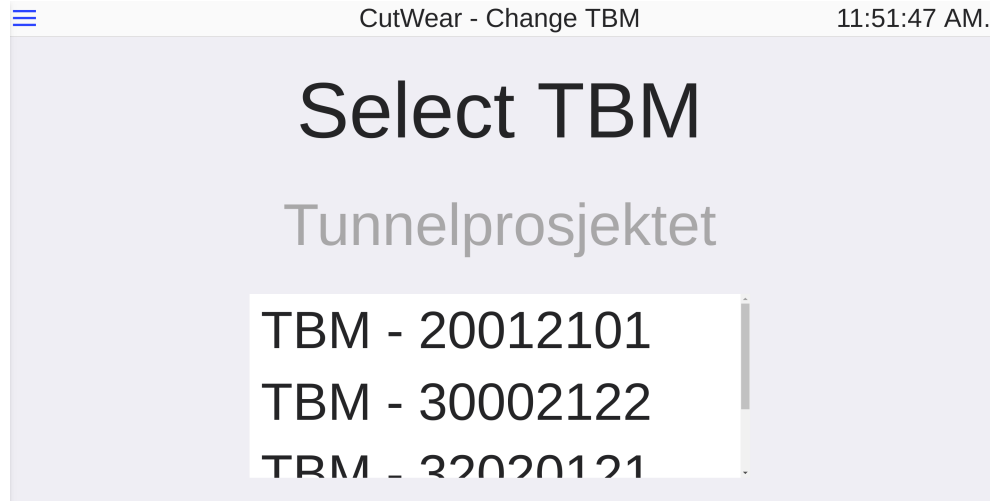
*Figur 2: TBM Inspector Toolbar Menu*

This menu has four four options, choosing the different options will result in:

- **Change TBM:** If you have started an inspection you will have to pause the inspection before continuing, this will save your progress. When paused, you will be redirected to the page where you can select which TBM to inspect.
- **Guidelines:** You will be redirected to a page in the application containing guidelines on how to use the application.
- **Log out:** If you have started an inspection you will have to pause the inspection before continuing, this will save your progress. When paused, you will be logged out of the application and returned to the login page.
- **Delete Inspection:** This option is used when you wish to exit the current inspection and delete all progress, you will then be redirected to the page where you select which TBM to inspect.

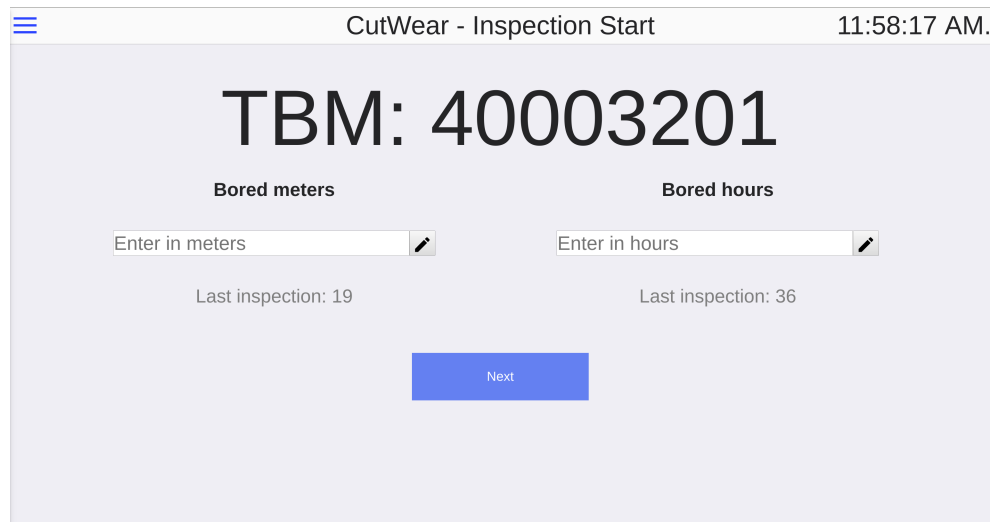
## Start an inspection

When you first log in with a TBM Inspector user, you will enter the page where you need to select which TBM to inspect from a list of all the TBM's in your project. On this page the name of the project your user is connected to is displayed with a grey text above the list of TBM's.



*Figur 3: The Select TBM page*

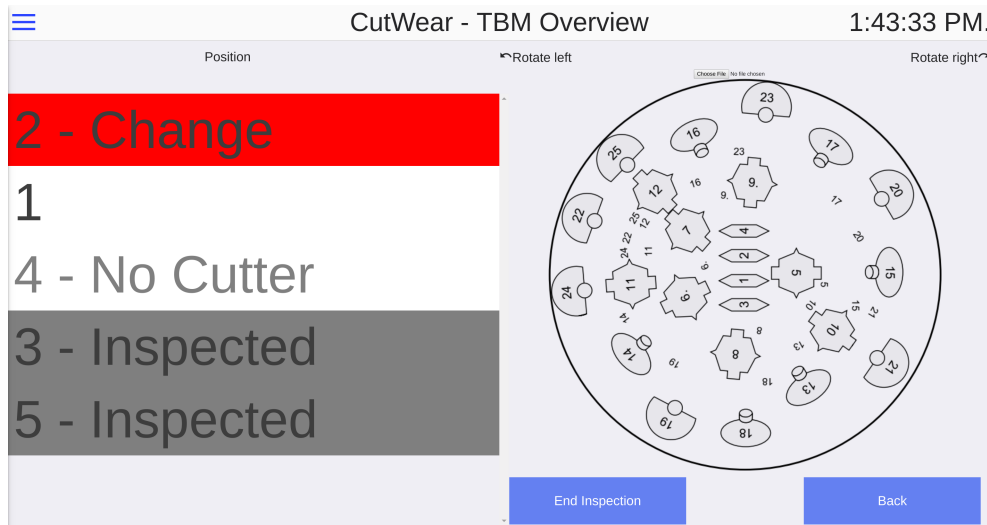
If there is a paused inspection registered on this TBM you will be notified of this before continuing. When you continue from select a TBM, you will be redirected to a page where you can register bored meters and bored hours for this inspection. The name of the selected TBM is displayed on this page, you will also get the values of bored meter and bored hours from the previous inspection.



*Figur 4: The Inspection Start page*

## The TBM Overview page

When you have pressed next from the inspection start page, you will enter the TBM overview page. On the right side of this page you have the option to upload a picture of the different positions in the TBM, this can be used to help the inspector find the positions. Press the Choose File button to upload a photo, when uploaded you can rotate picture using the buttons above it.



Figur 5: The TBM Overview page, with picture uploaded

On the left side of the page there will be a list off all the positions in the selected TBM, the positions can have the following codes:

- **Change (red background):** It is recommended to change the cutter in this position, this is because there is registered wear the exceeds the maximum wear limit or the user has marked the cutter to be changed during the inspection. Positions with this code will be placed at the top of the list.
- **Inspected (grey background):** It is either registered cutter wear within the wear limit or registered a cutter change in this position during the inspection. Positions with this code will be placed at the bottom of the list.
- **No cutter:** There is no cutter registered in this position

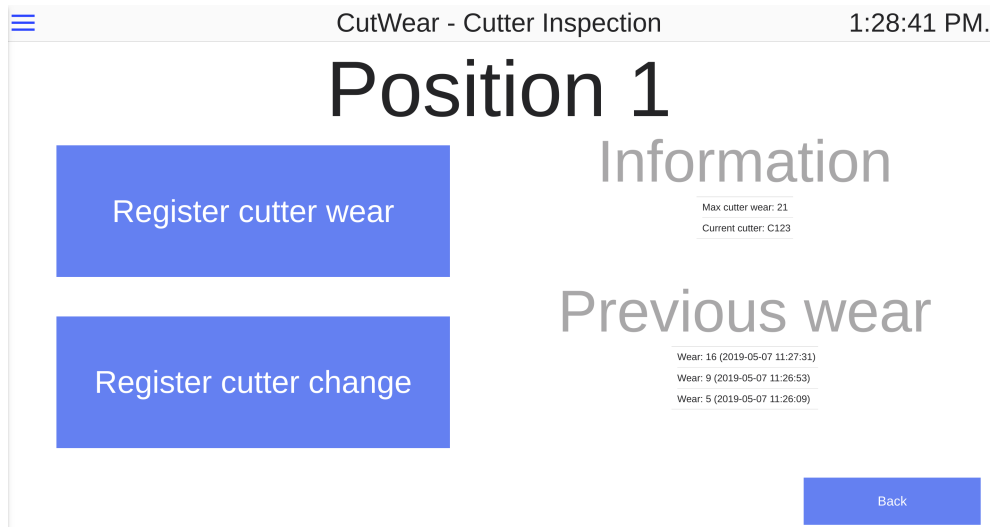
To view or register data in one of the positions, select the position by pressing the number in the list. When doing this you will be redirected to the cutter inspection page for the selected position.

### Register a cutter in position

If you select a position from the list in the TBM overview page and there is not registered a cutter in a position, you will be redirected to a page where you have to register the cutter located in this position before you can inspect the position/cutter. When you do this the position will no longer be marked with No Cutter in the list and you can enter the position to register inspection data.

## Inspecting a cutter/position

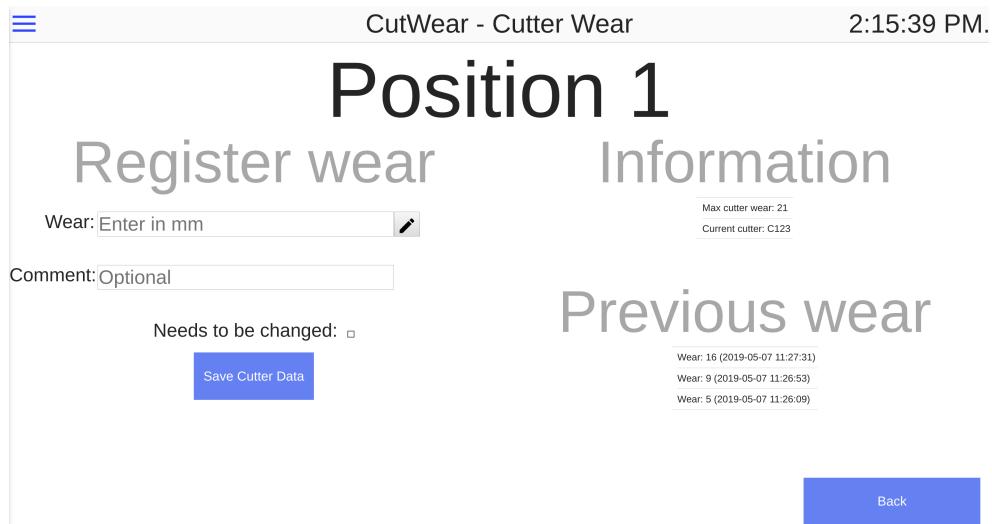
When you have selected a position from the position list in the TBM overview page, you will be redirected to the cutter inspection page for this position. The selected position number will be displayed on the top of the page. On the right side of this page there will be information about the position selected as well as the registered cutter wear from the three previous inspections.



Figur 6: The Cutter Inspection page, position 1

When inspecting a position you will get two options on the left side of the page., choosing one of these will change the left side of the page and leave the right side unchanged.

**Register cutter wear:** When you want to register the wear of this cutter you press this button, you will then be redirected to the cutter wear page where you can register wear. In addition to this you can add a comment as well as mark the position for a change, marking the position will give the position the Change code in the list of positions. When finished press Save Cutter Data.



Figur 7: The Cutter Wear page, position 1



**Register a cutter change:** When you select "Register cutter change" you will be redirected to the cutter change page. On this page you can register a cutter change, the reason for the change is selected from a dropdown menu containing a list of predetermined categories. You then enter the ID of the new cutter located in the position, if you wish you can also add a comment or register the wear of the cutter that was changed. When finished press Save Cutter Data.

CutWear - Cutter Change 2:16:33 PM.

# Position 1

## Register change

Reason: **abrasive wear**

New Id:

Comment:

Wear:

[Save Cutter Data](#)

## Information

Max cutter wear: 21  
Current cutter: C123

## Previous wear

Wear: 16 (2019-05-07 11:27:31)  
Wear: 9 (2019-05-07 11:26:53)  
Wear: 5 (2019-05-07 11:26:09)

[Back](#)

*Figur 8: The Cutter Change page, position 1*

## Ending an inspection

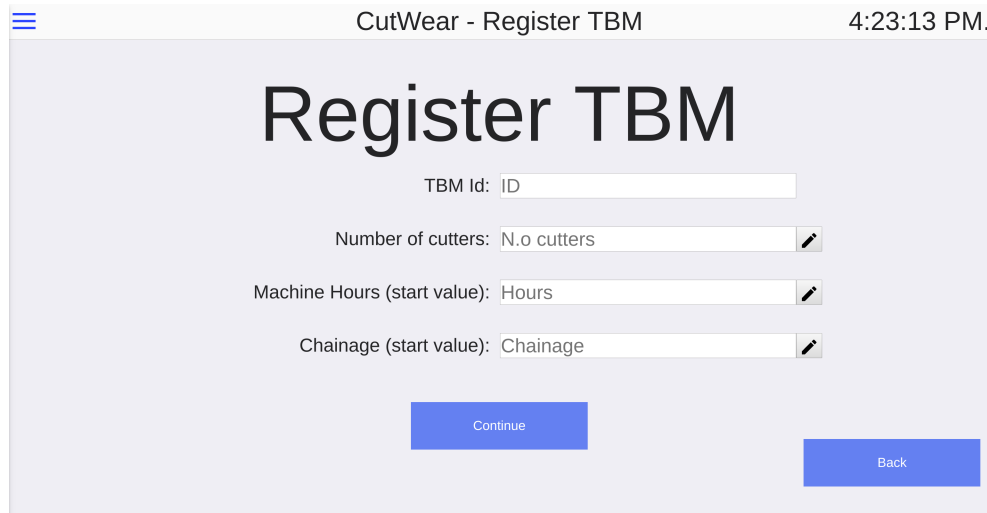
Pressing the End Inspection button in the TBM overview page will end the inspection and try to upload the inspection data to the database. If there is no internet connection the inspection data will be saved locally, as soon as the device is connected to the internet the data is uploaded. After doing this you will not be able to edit the data in this inspection. After ending the inspection the user will be logget out and you will be returned to the login page. If you do not wish to upload the data registered or pause the inspection, please follow the instructions in chapter 2.1 The toolbar menu

## Project Supervisor

The functionality available in this user group is connected to managing a project and viewing the data from this project. From the main menu of the Project Supervisor page you will have four options, these will be presented below.

### Register a TBM

To start registering a new TBM for the project the user you are logged in with is connected to, fill the input fields on the first page of the Register TBM pages. Then press the Continue button to go to the next page.



The screenshot shows the 'Register TBM' form in the CutWear application. The page title is 'CutWear - Register TBM' and the time is 4:23:13 PM. The main heading is 'Register TBM'. Below the heading are four input fields: 'TBM Id: ID', 'Number of cutters: N.o cutters', 'Machine Hours (start value): Hours', and 'Chainage (start value): Chainage'. Each input field has a small edit icon to its right. At the bottom of the form are two blue buttons: 'Continue' and 'Back'.

*Figur 9: Register TBM, part 1*

When you have finished the first step you will be redirected to the second page where you will have to register max wear and max wear for the neighbouring positions in each position. The list will contain as many positions as you entered in page 1, if you wish to change this press Back. Note that going back will delete your progress on registering max wear. When you have entered all of the data and are ready to finish the registration, press the Finish button.

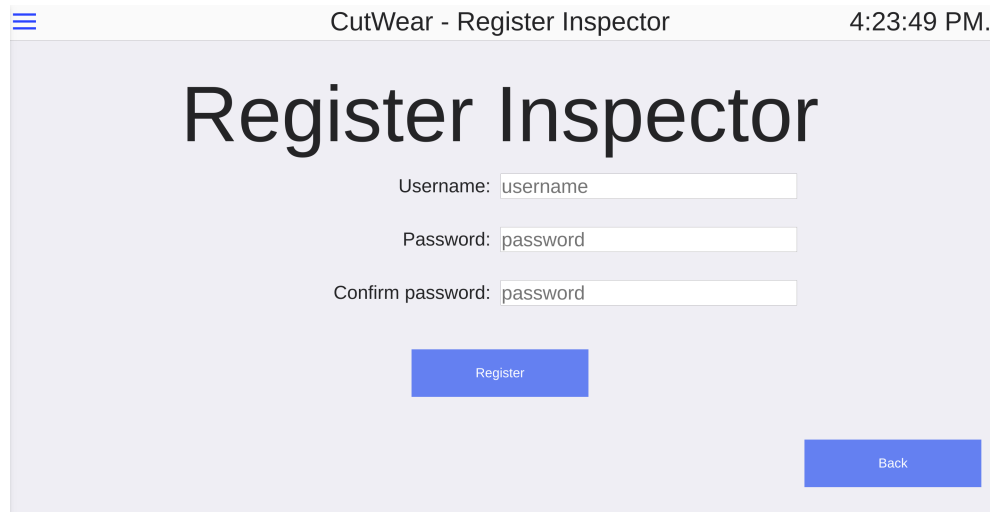


The screenshot shows the 'Register Position Details' form in the CutWear application. The page title is 'CutWear - Register TBM' and the time is 4:31:14 PM. The main heading is 'Register Position Details'. Below the heading is a table with three columns: 'Position', 'Max wear', and 'Neighbour max'. The table has four rows, numbered 1 to 4. Each row has two input fields: 'Max' and 'Neighbour max'. Each input field has a small edit icon to its right. At the bottom of the form are two blue buttons: 'Finish' and 'Back'.

*Figur 10: Register TBM, part 2*

## Register a TBM Inspector

When a Project Supervisor need to register a new TBM Inspector user you click the Register Inspector button on the main page, you will then need to enter a valid username that is not in use and a password. Press the Register button to register the user, this user will belong to the same project as the logged in Project Supervisor.



CutWear - Register Inspector 4:23:49 PM.

# Register Inspector

Username:

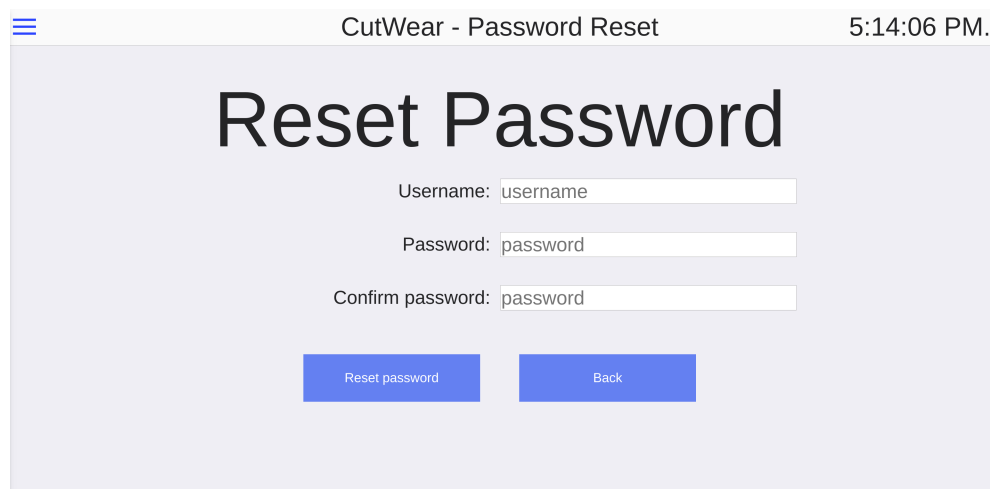
Password:

Confirm password:

*Figur 11: Register TBM Inspector*

## Reset Inspector password

If a TBM Inspector has forgotten the password, a Project Supervisor can give him a new password. The supervisor can do this by selecting Reset Inspector Password from the main page, the user will then be redirected to a page where you need to enter the username of the TBM Inspector and the new password.



CutWear - Password Reset 5:14:06 PM.

# Reset Password

Username:

Password:

Confirm password:

*Figur 12: Reset Inspector Password*

## View TBM

The View TBM feature is used to read inspeksjon data from the TBM's in the project. When you click the View TBM button from the main page you will first be asked to select a TBM from a list of all TBM's in the project, then you will have to choose which inspection connected to the selected TBM you wish to view. When you have selected an inspection you will be redirected to a page that display the data collected from this inspection.

The screenshot shows a web application interface titled "CutWear - View TBM" with a timestamp of "8:21:25 PM.". The main heading is "Inspection Details". The interface is split into two columns: "Information" on the left and "Changes" on the right. The "Information" section contains the following data:

|                                               |
|-----------------------------------------------|
| TBM: 40003201                                 |
| Inspection finished: Tue May 07 2019 14:59:39 |
| Bored meters: 36                              |
| Bored hours: 19                               |
| Number of cutter changes: 4                   |

The "Changes" section is titled "Changes" and "Wear". It contains two scrollable lists:

- Changes:**
  - Position 1, reason for change: abrasive wear
  - Position 4, reason for change: damaged hub
  - Position 5, reason for change: other
  - Position 7, reason for change: abrasive wear
- Wear:**
  - Position 1, wear registered: 19 mm
  - Position 3, wear registered: 12 mm
  - Position 4, wear registered: 48 mm
  - Position 5, wear registered: 23 mm

A blue "Back" button is located at the bottom right of the page.

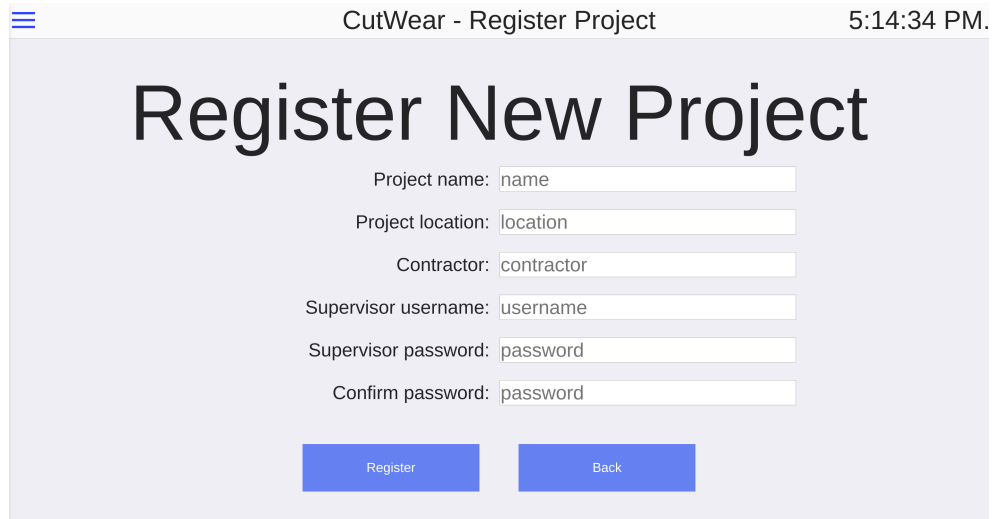
*Figur 13: View TBM, Inspection Information*

## Admin

The Admin user group has two options which will be presented below, Registering Project and Change Supervisor Password.

### Register Project

To register a project fill the input fields, this includes registering a Project Supervisor to manage the project. When you are ready to register the project with the data provided, press Register.

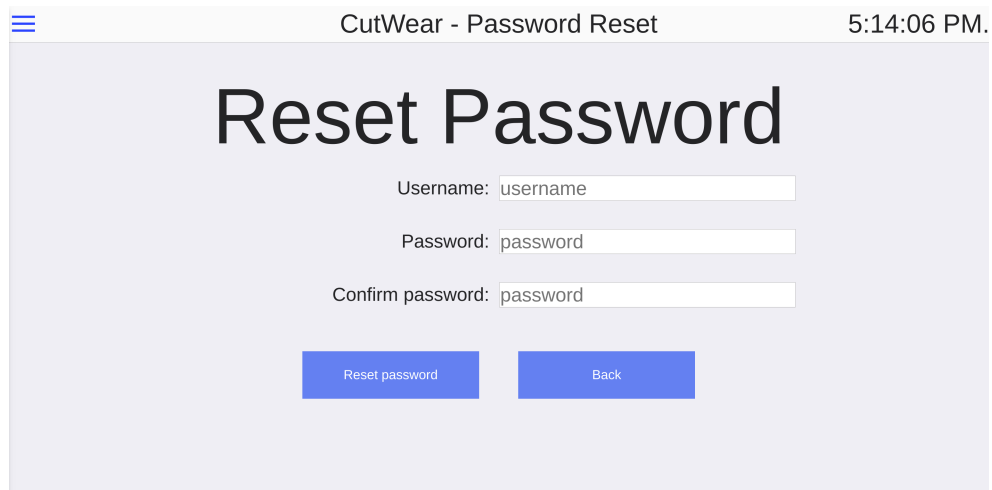


The screenshot shows a web browser window titled "CutWear - Register Project" with a timestamp of 5:14:34 PM. The main heading is "Register New Project". Below the heading are six input fields: "Project name: name", "Project location: location", "Contractor: contractor", "Supervisor username: username", "Supervisor password: password", and "Confirm password: password". At the bottom of the form are two blue buttons: "Register" and "Back".

*Figur 14: Register Project*

### Reset Supervisor password

If a Project Supervisor has forgotten the password, an Admin can give him a new password. The Admin can do this by selecting Reset Supervisor Password from the main page, the user will then be redirected to a page where you need to enter the username of the Project Supervisor and the new password.

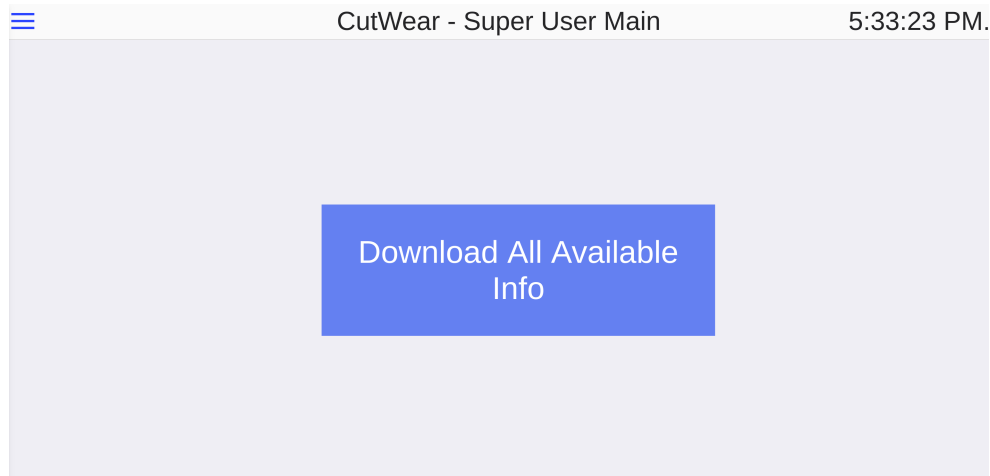


The screenshot shows a web browser window titled "CutWear - Password Reset" with a timestamp of 5:14:06 PM. The main heading is "Reset Password". Below the heading are three input fields: "Username: username", "Password: password", and "Confirm password: password". At the bottom of the form are two blue buttons: "Reset password" and "Back".

*Figur 15: Reset Inspector Password*

# Super User

This user group has only one function and that is to download all the inspection data in all projects from the database, these information will be stored in the info.json file located in the directory where the program is installed. This is a JavaScript Object Notation file and can be opened in programs like Excel.



Figur 16: Super User Page

To open the info.json file in Excel, follow this tutorial by Microsoft (Note: this is for Windows only): <https://support.office.com/en-ie/article/connect-to-a-json-file-f65207ab-d957-4bf0-bec3-a08bb53cd4c0> If you follow this tutorial you will open **Power Query**, then follow this instructions:

1. Press the **Convert to table** button.
2. Then you can extend the table to view the desired data by clicking this icon on the top of the column.



Figur 17: Convert to table button

3. Then you have to press **Extend to more rows**



Figur 18: Extend table button

4. Then click the same icon as in the second step.
5. Then you can choose the values you to add to the table.
6. Repeat the steps above until you are satisfied with the data in the table. We recommend that you extract changes and wear in two separate tables.
7. At last press the **Close and Load** to import it into Excel. You are now done.



Figur 19: Close and load button

