



Bachelorprosjekt 2019

Gruppe 5





Institutt for Informasjonsteknologi

Postadresse: Postboks 4 St. Olavs plass, 0130 Oslo

Besøksadresse: Holbergs plass, Oslo

PROSJEKT NR.
2019-05

TILGJENGELIGHET
Åpen

BACHELORPROSJEKT

Telefon: 22 45 32 00

HOVEDPROSJEKTETS TITTEL Nysgjerrigper - Prosjektveilederen	DATO 23.05.2019
	ANTALL SIDER / BILAG 264 / 1
PROSJEKTDELTAKERE Thomas Arne Dahll (s170761), Olav Sørli (s309647), Vegard Skullestad Bjørkgård (s236352), Yanislav Todorov Zagorov (s309732)	INTERN VEILEDER Eva Hadler Vihovde

OPPDRAGSGIVER Netcompany Norge	KONTAKTPERSON Kirstine Gyring
---------------------------------------	--------------------------------------

SAMMENDRAG <p>Nysgjerrigper – Prosjektveilederen er et bachelorprosjekt gjennomført ved Oslomet – Storbyuniversitetet i samarbeid med Netcompany og Nysgjerrigperavdelingen hos Forskningsrådet. Målet for prosjektet har vært å bidra til at flere lærere bruker Nysgjerrigpermetoden, som er en forenklet hypotetisk-deduktiv forskningsmetode ment for barn i grunnskolen.</p> <p>For å nå dette målet er det utviklet en nettapplikasjon som veileder lærer og elever gjennom metoden og gjennomføringen av et Nysgjerrigperprosjekt. Den gir mulighet til å fylle inn det man produserer underveis, og genererer et utkast til en prosjektrapport som kan leveres til Nysgjerrigper-konkurransen. Prosjektveilederen er ment å gi god veiledning underveis i arbeidet med metoden, slik at det skal være enkelt å igangsette et prosjekt uten mye forkunnskap. Hensikten er å senke terskelen for at lærere skal ta metoden i bruk i undervisningen.</p> <p>I tillegg til veilederen er det også utviklet et administrasjonsgrensesnitt slik at Nysgjerrigper skal ha god oversikt over bruken av prosjektveilederen slik at de kan benytte denne informasjonen i arbeidet sitt.</p> <p>Prosjektveilederen og administrasjonsgrensesnittet er begge knyttet til en omfattende backend-løsning gjennom et API. Her lagres alt av applikasjonens informasjon sikkert i en database, inkludert alle prosjekter som er påbegynt og all tilhørende informasjon.</p>
--

3 STIKKORD Forskningsmetode
Klient/tjener-nettapplikasjon
Pedagogisk verktøy

INNHALDSFORTEGNELSE

Leserveiledning.....	4
1 PRESENTASJON	6
1.1 Innledning	8
1.2 Bakgrunn for oppgaven	10
1.3 Problemstilling	10
1.4 Konsept.....	11
1.5 Beskrivelse av løsningen	12
1.6 Konklusjon.....	19
2 PROSESSDOKUMENTASJON	20
2.1 Innledning	22
2.2 Prosjektverktøy	22
2.3 Prosjektfaser.....	30
2.4 Samsvar mellom kravspesifikasjon og produkt.....	51
2.5 Potensiale for videre utvikling.....	52
2.6 utfordringer og refleksjoner	55
2.7 Konklusjon til prosessdokumentasjon	56
3 PRODUKTDOKUMENTASJON	57
3.1 Innledning	59
3.2 Funksjonell systembeskrivelse	62
3.3 Programmets oppbygning og virkemåte	79
3.4 Publisering.....	119
3.5 Konklusjon til produktdokumentasjon	122
4 TESTRAPPORT	123
4.1 Introduksjon.....	125
4.2 Enhetstesting	126
4.3 Integrasjonstesting	131
4.4 Systemtesting	137
4.5 Akseptansetesting	138
4.6 Brukertesting.....	139
4.7 Konklusjon til testrapport	141
5 REFERANSELISTE	142
5.1 Referanser	143
6 VEDLEGG	147
6.1 Kravspesifikasjon og brukerhistorier	149
6.2 Attester	164
6.3 Begrepsavklaringer	166
6.4 Risikorapport	169
6.5 Brukerveiledning.....	180
6.6 Brukertester	195
6.7 Akseptansetest	236

Dette dokumentet er den endelige sluttrapporten for vår bacheloroppgave for ingeniørfag – data ved Oslomet – Storbyuniversitetet våren 2019, og beskriver utviklingen og resultatet av vårt arbeid med en prosjektveileder for Nysgjerrigpermetoden. Rapporten er optimalisert for digital lesning, med lenker til relevant tilleggsinformasjon der det er aktuelt, men kan også uproblematisk leses i utskriftsformat om det er ønskelig.

Rapporten består av følgende deler:

1. PRESENTASJON 6-19

Presentasjonsdelen inneholder innledningen og en sammenfatning av hele prosjektet. Det presenterer aktørene i prosjektet; oppdragsgiveren, produkteieren og prosjektgruppen, og har en kortere beskrivelse av produktet og prosjektets utførelse. Det er presentasjonsdelen som beskriver bakgrunnen for prosjektet, problemstillingen prosjektgruppen kom frem til og sluttkonklusjonen for prosjektet.

Dette dokumentet anbefales lest først og i sin helhet.

2. PROSESSDOKUMENTASJON 20-57

Prosessdokumentasjonen er dokumentering som omhandler hvordan vi har jobbet, og hvilke valg vi har tatt underveis. I dette dokumentet har vi loggført arbeidet vårt og delt det inn i den forberedende delen og sprinter, med beskrivelser av arbeidsmetodikk og prosessverktøy.

3. PRODUKTDOKUMENTASJON 58-123

Produktdokumentasjonen er en total fremstilling av applikasjonen. Vi beskriver våre tekniske løsninger i applikasjonen og går i dybden på hvordan løsningen henger sammen.

4. TESTRAPPORT 124-142

Testrapporten beskriver hvordan vi har testet applikasjonen vår, samt hvordan den har blitt sjekket mot kravspesifikasjonen under brukertester og akseptansetest gjort av produkteierne.

5. REFERANSELISTE 143-144

Bibliografist referanseliste

6. VEDLEGG 148-265

Vedlagt finnes; kravspesifikasjon og brukerhistorier, attester, begrepsavklaringer, risikorapport, brukerveiledning, brukertester og akseptansetester.

APPLIKASJONENS NETTADRESSE

Nettapplikasjonen kan nås ved å bruke lenken <http://np-frontend.s3-website.eu-central-1.amazonaws.com/>. Her kan den besøkende opprette en bruker og følge prosjektveilederen for å lage forskningsrapporter.

BEGREPSAVKLARING:

En del begreper som omtales i disse dokumentene vil være engelske ord og begreper det enten ikke finnes fullgode norske ord for eller de har innarbeidede begreper i norske utviklingsmiljøer. Definisjoner for disse og hvordan vi forstår disse begrepene beskrives i vedlegget for [begrepsavklaringer](#).

Utover begrepsavklaringsdokumentet er det noen begreper som er så viktige at vi velger å definere disse her:

Bruker, lærer, administrator og besøkende

En bruker er i vår sammenheng definert som en person som har en brukerkonto i applikasjonen. Både lærere og administratorer er brukere. En person som bruker nettapplikasjonen men som ikke har en brukerkonto har vi valgt å definere som en besøkende.

Frontend

“Frontend er den delen av programvaren som ligger nærmest brukeren. Det er koden som former det du visuelt ser på skjermen, og bestemmer hva som skjer når du interagerer med disse elementene.” (Granevang 2015)¹

Backend

“Backend er den delen av programvaren som ligger nærmest databasen der dataene er lagret. Det er også her de tunge kalkuleringsoperasjonene skjer, som brukeren av systemet ikke nødvendigvis har noe forhold til. Det varierer fra program til program hvor stor backend er i forhold til frontend.” (Granevang 2015)²

¹ Granevang, Merethe. (2015, oktober). Frontend. I Store norske leksikon. Hentet 8. mai 2019 fra <https://snl.no/Frontend>

² Granevang, Merethe. (2015, oktober). Backend. I Store norske leksikon. Hentet 8. mai 2019 fra <https://snl.no/Backend>

1 PRESENTASJON

NYSGJERRIGPER – PROSJEKTVEILEDEREN

BACHELORPROSJEKT 2019

GRUPPE 5

INNHALDSFORTEGNELSE PRESENTASJONSDEL

1.1	INNLEDNING.....	8
1.1.1	Presentasjon av prosjektgruppen	8
1.1.2	Faglige forutsetninger.....	9
1.1.3	Presentasjon av oppdragsgiver.....	9
1.1.4	Presentasjon av kunde	9
1.2	BAKGRUNN FOR OPPGAVEN	10
1.3	PROBLEMSTILLING.....	10
1.4	KONSEPT	11
1.5	BESKRIVELSE AV LØSNINGEN.....	12
1.5.1	Prosjektveileder.....	14
1.5.2	Administrasjonsgrensesnitt	17
1.6	KONKLUSJON	19

1.1 INNLEDNING

Dette er en oppsummering av vår rapport. Hensikten er å presentere produktet, oss selv, oppdragsgiver og kunde. Vi presenterer her problemstillingen til og bakgrunnen for oppgaven, samt et sammendrag der løsningen på hele prosjektet beskrives i grove trekk. Elementer i sammendraget vil gjentas og diskuteres utfyllende i de øvrige delene av prosjektrapporten. Presentasjonen skal gi leseren et godt grunnlag for å lese de etterfølgende dokumentene, slik at dette gir større glede. Den anbefales derfor lest i sin helhet.

Vi i prosjektgruppen ønsker å benytte anledningen til å takke alle parter som har hjulpet oss med å fullføre denne prosjektoppgaven. Vi vil takke vår oppdragsgiver Netcompany, med en spesiell takk til Kirstine Gyiring for god prosjektledelse og Andreas Kemkers for god teknisk støtte. Vi takker kunden, Forskningsrådet, avdeling Nysgjerriger for stor frihet for hvordan vi fikk løse oppgaven og for et godt samarbeid. Takk til veileder Eva H. Vihovde fra OsloMet for god veiledning.

1.1.1 PRESENTASJON AV PROSJEKTGRUPPEN

Vi er fire studenter ved OsloMet som har ulike bakgrunner, erfaringer og personligheter, men med et felles ønske om å bidra til arbeid med god kvalitet som er givende, både på et faglig og personlig plan.

Vegard Bjørkgård har både studert arkitektur og vært restaurantsjef før han begynte på dataingeniørstudiet ved OsloMet. Det er han som har holdt kontroll på Scrumgjennomføringen. Han var hovedansvarlig i frontend og har tatt på seg det dypere tekniske ansvaret for teknologiene der.

Yanislav Zagorov har jobbet som Orakel og studentassistent på OsloMet, hvilket har gjort ham godt rustet til arbeidet med design og utvikling av frontendapplikasjonen. Ettersom han jobber som analytiker i et datasikkerhetsselskap, var det naturlig at han tok på seg hovedansvaret for sikkerhetsaspektene i applikasjonens klient.

Olav Sørli er utdannet avionikker og har jobbet som maskinoperatør. Han er hovedansvarlig i backend og har tatt på seg det dypere tekniske ansvaret for teknologiene der. Olav er i tillegg vår tillitsvalgte, som vi har drøftet våre frustrasjoner med.

Thomas Dahll har fullført hærens befalsskole og er autorisert sykepleier. Arbeider som studentassistent i programutvikling på OsloMet. Thomas er gruppens teamleder. Han jobbet backend, med ansvar for grunnarkitekturen til applikasjonen. Underveis derfor gikk han over til å også fokusere på frontend.

1.1.2 FAGLIGE FORUTSETNINGER

Høsten 2018 valgte tre av gruppemedlemmene faget ITPE3200 – Webapplikasjoner som valgfag, Dette gav en god forståelse av arkitektur og rammeverk som benyttes i moderne webapplikasjoner i dag, med henholdsvis ASP.NET, .Net Core, Entity Framework og et RESTful API konsumert av en enkel [React](#)-basert klient.

Dette i kombinasjon med den kunnskapen de to generelle programmeringsfagene ITPE1400 – Programmering og ITPE1600 – Programutvikling, som begge baserte seg på Java, samt de databasekunnskapene faget DATS1500 Databaser har gitt oss, spå har det skapt gode forutsetninger for vår mulighet til å utvikle et godt produkt til kunden vi samarbeider med.

1.1.3 PRESENTASJON AV OPPDRAGSGIVER

Netcompany Norge er et konsulentbyrå som leverer IT-løsninger til det offentlige og til private kunder. Selskapet har over 2000 ansatte og kontorer i Norge, Danmark, Storbritannia, Polen og Vietnam. De er i sterk vekst, og fokuserer i stor grad på å kultivere langvarige kundeforhold hvor de har tett kontakt med kundene når prosjekter gjennomføres.

Som vår oppdragsgiver har Netcompany stilt med overordnet prosjektledelse, kontakt med kunden og bistått oss med både teknisk støtte og designstøtte ved behov.

1.1.4 PRESENTASJON AV KUNDE

Forskningsrådet, avdeling Nysgjerrigper, er en statlig instans som i all hovedsak har som oppgave å fordele forskningsmidler til ulike forskningsprosjekter på landsbasis. Nysgjerrigper er en underavdeling av Forskningsrådet, og har som mål å drive forskningsformidling rettet mot barn.

Nysgjerrigper har utarbeidet en hypotese-deduktiv forskningsmetode som er noe forenklet i forhold til vanlig forskningsmetode. Denne kaller de Nysgjerrigpermetoden, og de tilbyr denne metoden som et læringsverktøy for barneskoletrinnet.

1.2 BAKGRUNN FOR OPPGAVEN

Hvert år avholder Forskningsrådet og Nysgjerriger en konkurranse hvor elever på 1.-7. trinn i grunnskolen forsker på en selvvalgt problemstilling ved hjelp av Nysgjerrigpermetoden. Her er noen eksempel på problemstillinger fra tidligere konkurranser <https://www.nysgjerrigpermetoden.no/>:

- Er det sant at vi er like lange fra fingertupp til fingertupp som fra hode til føtter? - 2019
Forskere: 2. klasse ved Hedemarken friskole (Løten, Hedmark)
- Mikroplast - finnes det i Mjøsa? - 2018
Forskere: 5.trinn ved Ingeberg skole (Hamar, Hedmark)
- Hunden er menneskets beste venn. Hvordan vil hunder i klasserommet påvirke skolehverdagen og klasse miljøet vårt? -2018
Forskere: 6.trinn ved Storevarden skole (Sola, Rogaland)

Det finnes i dag en nettløsning for veiledning til Nysgjerriger-forskningsrapporter. Denne er tung i bruk og lite intuitiv. Brukeren er i stor grad avhengig av å kjenne Nysgjerrigpermetoden godt og kjenne til brukergrensesnittet fra før, for å kunne gjennomføre et prosjekt effektivt.

Nysgjerriger ønsker en prosjektveileder som kan benyttes til både å planlegge og gjennomføre et forskningsprosjekt med elevene. Dette gjennom å veilede lærerne trinn for trinn gjennom Nysgjerrigpermetoden, og slik senke terskelen for at lærere tar i bruk metoden i undervisningen.

Dette har vært utgangspunktet for vårt prosjekt.

1.3 PROBLEMSTILLING

Problemet som vårt prosjekt søker å løse, er å senke terskelen for at lærere skal velge å ta i bruk Nysgjerrigpermetoden i undervisningen sin.

Denne problemstillingen kommer som følge av Nysgjerrigers mål om å bidra til at flere lærere skal ta i bruk metoden, og slik introdusere forskningsmetodikk for flest mulig elever.

Prosjektets problemstilling ble over en lengre periode i begynnelsen av prosjektperioden videreutviklet i samarbeid med Nysgjerrigeravdelingen hos Forskningsrådet. Det resulterte i at prosjektgruppen produserte en kravspesifikasjon og en liste med brukerhistorier som produkteierne godkjente. Disse ligger [vedlagt](#) og anbefales lest.

Dette arbeidet førte til at prosjektgruppens fokus ble å lage en prosjektveileder for Nysgjerrigpermetoden. Denne skulle gjøre det intuitivt og enkelt å komme i gang med

å gjennomføre et forskningsprosjekt basert på metoden med god veiledning underveis. Det tas sikte på at læreren kan bruke prosjektveilederen sammen med barn på en større skjerm. Dermed var det ønskelig at designet også var rettet mot barn. Utover denne prosjektveilederen, definerte produkteier andre mål for oppgaven:

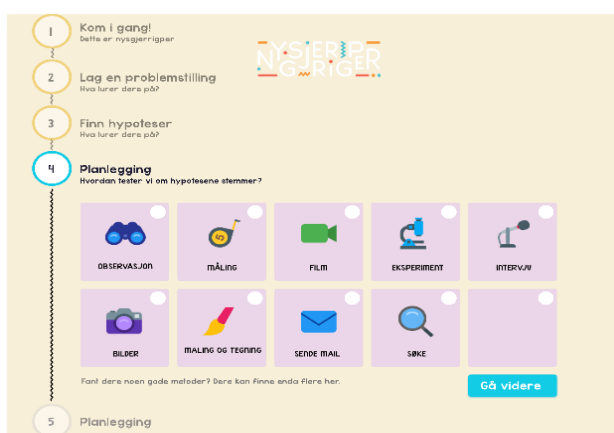
- Opprette et administrasjonsgrensesnitt for applikasjonen.
- Sikre applikasjonen med tofaktor-autentisering.
- Ta i bruk en AWS-skyløsning.

1.4 KONSEPT

Målet og et førsteutkast til konseptet for prosjektet ble i forkant av vår oppstart konkretisert gjennom samarbeid mellom en tjenstedesigner hos Netcompany og Nysgjerrigper-teamet. Det ble da definert at målet for prosjektet var å bidra til at flere lærere bruker Nysgjerrigpermetoden.



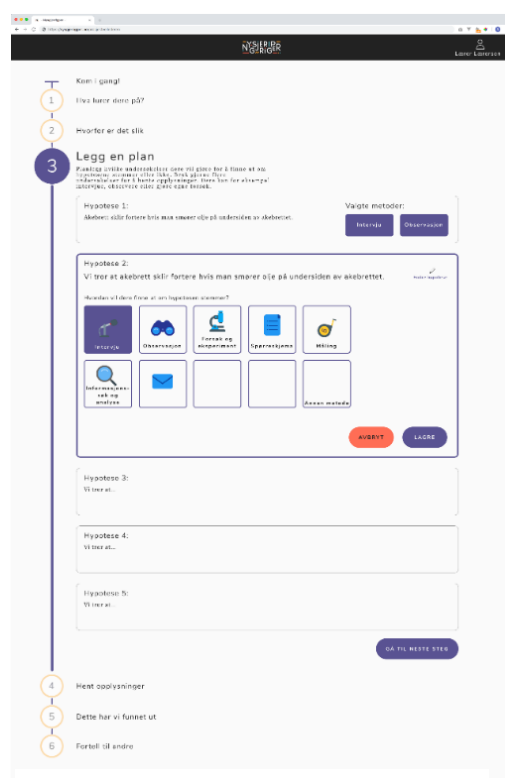
Figur 1 - Mål for prosjektet



Figur 2 - Tidlig utkast til designkonseptet

I denne forberedende fasen ble det utarbeidet flere mulige konsepter der ett ble valgt, som vi tok over ved vår oppstart i januar. Vi videreutviklet dette til et mer konkret konsept.

Konseptet ble å utarbeide en nettapplikasjon som tar læreren stegvis gjennom Nysgjerrigpermetoden. Applikasjonen er ment som en veileder og skal i seg selv være god nok til at metoden blir enkel å ta i bruk, både i oppstartsfasen og i gjennomføringen av en skoleklassens prosjekt.



Figur 3 - Utkast til designkonsept

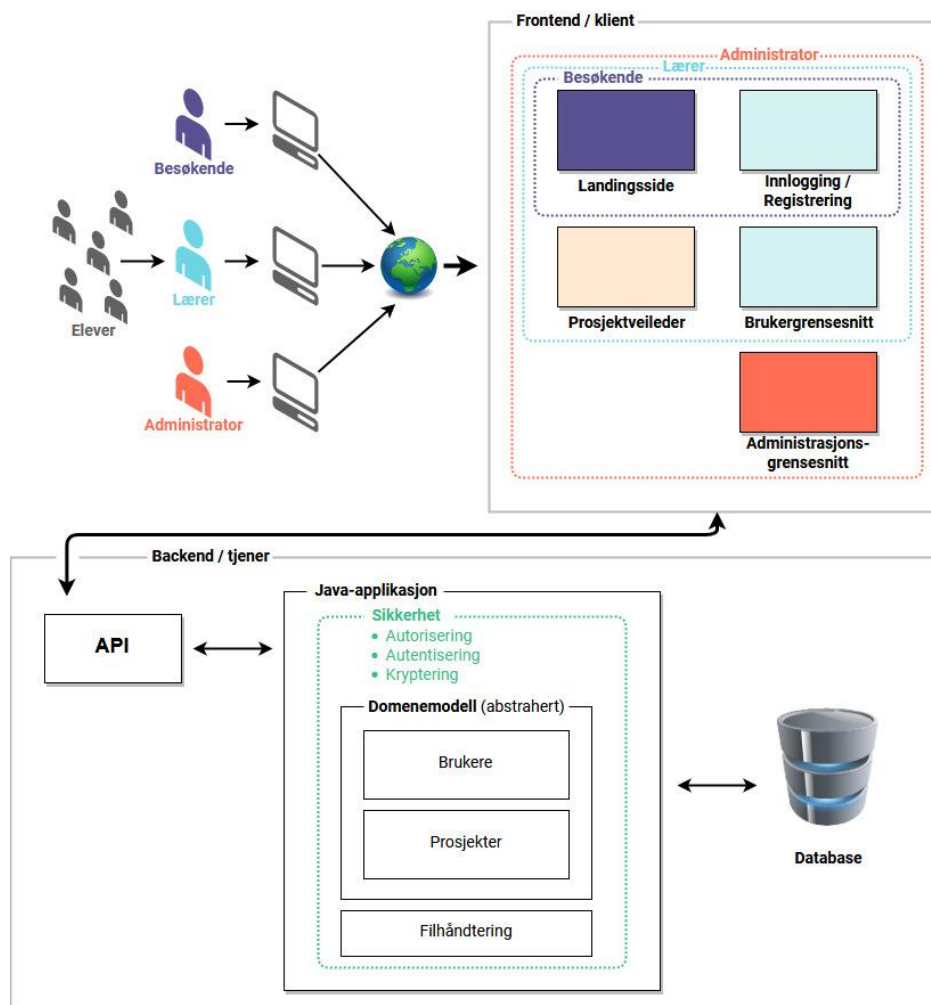
Brukeren skal få tips og veiledning til utførelsen av de forskjellige stegene i metoden, og i tillegg ha mulighet til å fylle inn informasjonen man innhenter underveis.

I designet av konseptet har vi til dels vært bundet til Nysgjerrigers designprofil, men utenom dette ble vi gitt stor frihet til å angripe problemstillingen slik vi selv ønsket. Vi har hatt tilgang til designressurser og innhold vi har kunnet benytte i arbeidet med både designet og den funksjonelle utformingen av konseptet.

1.5 BESKRIVELSE AV LØSNINGEN

Denne beskrivelsen er et sammendrag av beskrivelsen av applikasjonen i produkt-dokumentasjonen og en kort presentasjon av løsningen. For mer utfyllende informasjon anbefales spesielt den [funksjonelle beskrivelsen](#) av løsningen i produktdokumentasjonen.

Den endelige løsningen som vi har levert til kunden, består av to meget omfattende prosjekter: En databaseapplikasjon som tilbyr et API og en nettbasert klient med mange funksjonaliteter, som konsumerer dette API-et.

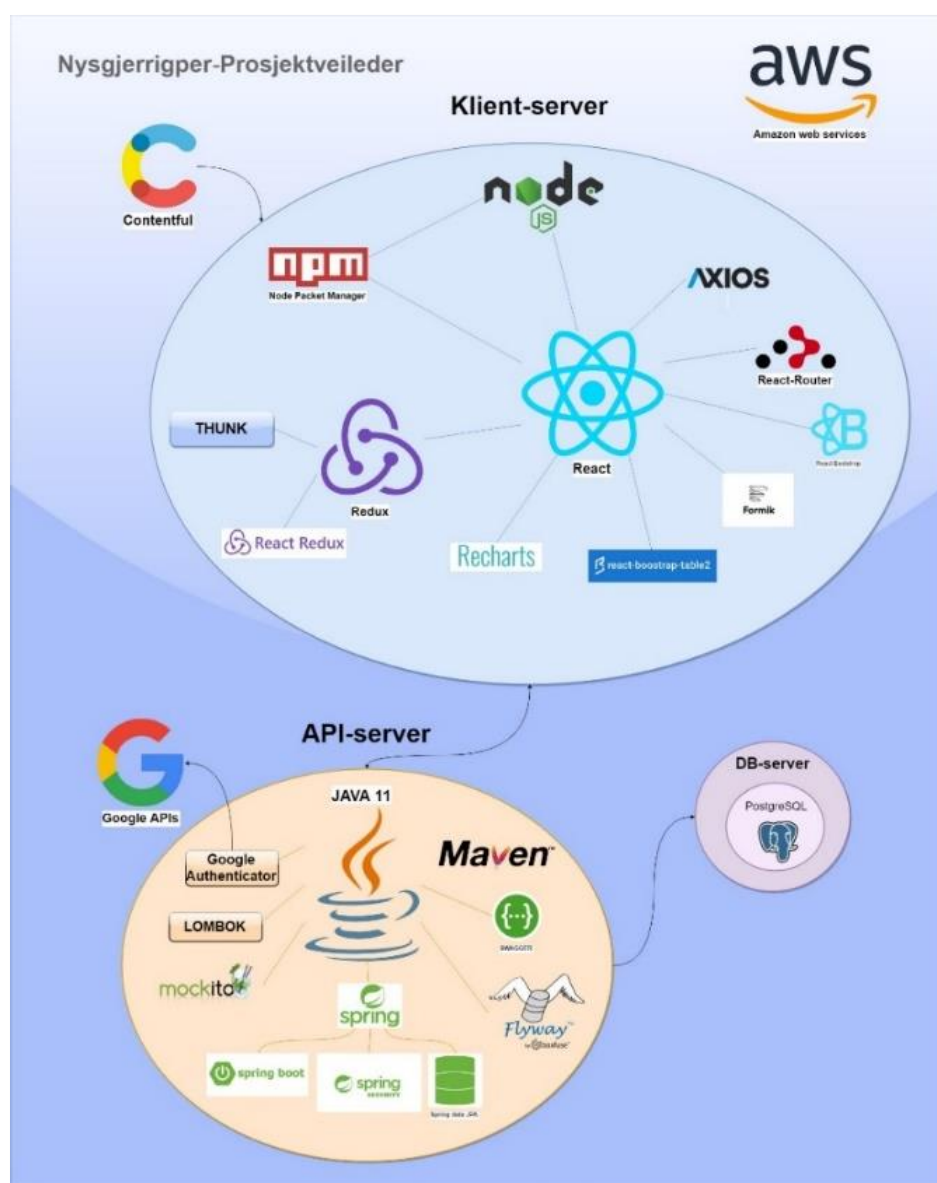


Figur 1 - Abstrahert oversikt over applikasjonens flyt og funksjonalitet.

Klienten er ansvarlig for nettsiden som brukeren ser, inkludert grensesnittet som kunden kommer til å benytte for å administrere alle brukere. Denne klienten er hovedsakelig designet for større skjermer i landskapsformat da den er ment for bruk av en lærer i samhandling med elever.

API-et er koblet til databasen vår og all kode som lar kunden lagre og hente data for bruk i klienten. Denne kommer til å kjøre kontinuerlig i bakgrunnen, og håndterer data om både brukerne og prosjektene deres.

Disse prosjektene er knyttet sammen, og designet for å jobbe i tandem. For å oppnå dette er det benyttet mange forskjellige teknologier, samt tilleggskomponenter til disse.



Figur 2 - Teknologioversikt

Å integrere alle de benyttede teknologiene og å få disse til å fungere godt sammen har vært en utfordring. Men ved å vektlegge dette under utviklingsarbeidet, har dette blitt vellykket. Applikasjonens rammeverk er laget med og tilrettelagt for muligheten til å knytte enda flere teknologier til seg og bli utvidet videre.

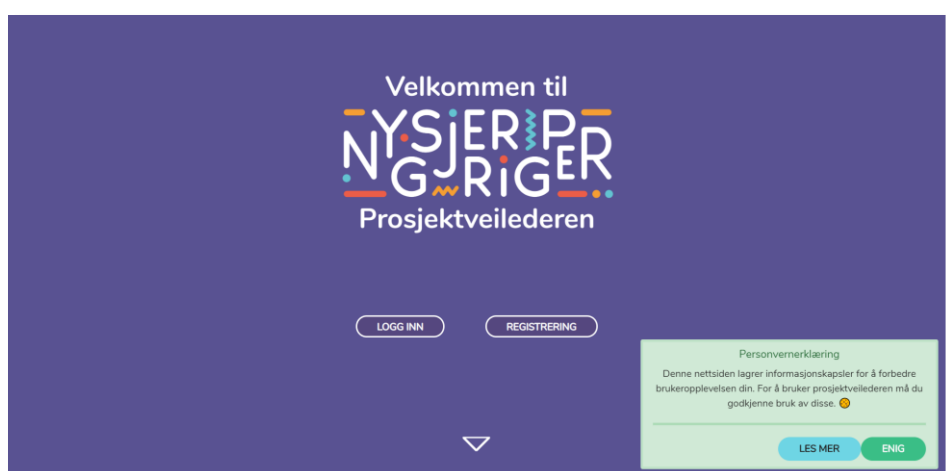
I de neste underkapitlene viser vi litt av prosjektveilederen og administrasjons-grensesnittet Forskningsrådet får tilgang til når prosjektet lanseres.

1.5.1 PROSJEKTVEILEDER

Nysgjerriger – Prosjektveilederen har tre hovedmål:

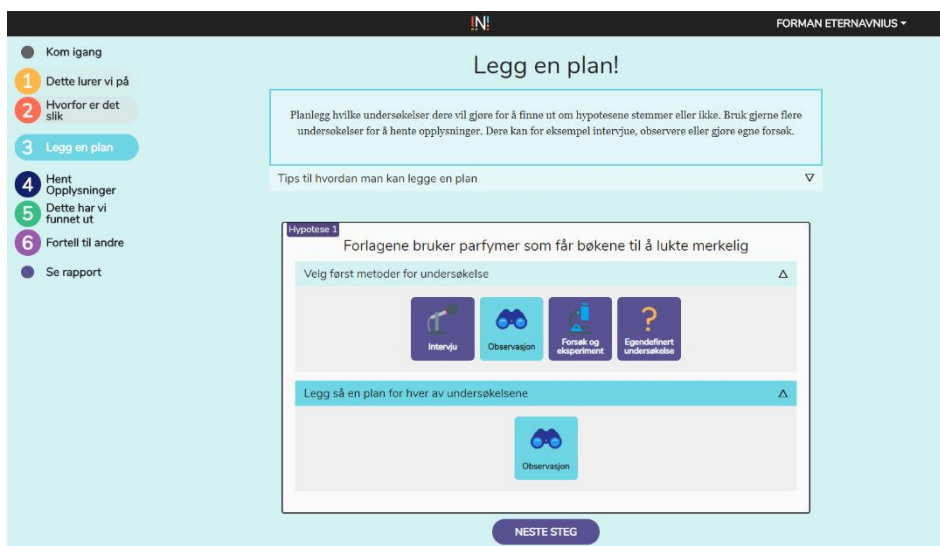
- Å introdusere læreren/klassen til Nysgjerrigpermetoden
- Å veilede læreren/klassen gjennom Nysgjerrigpermetoden
- Å vise læreren/klassen informasjonen de har lagt inn i prosjektveilederen som et dokument som kan leveres inn til Nysgjerrigper-konkurransen.

Det første som møter en besøkende til nettsiden er en enkel og oversiktlig landingsside. Man har her tilgang til en kort innføring i hva Nysgjerrigpermetoden er, og mulighet til å registrere seg med en ny bruker eller logge inn om man allerede har en.



Figur 3 - Landingsiden

Dersom man logger seg inn og ønsker å opprette et nytt prosjekt, kommer man til prosjektveilederen. Prosjektveilederen forteller kort hva de forskjellige stegene i Nysgjerrigpermetoden er, og gir tips til hvordan disse kan utføres av læreren og deres elever. Veilederen er fargerik, og har mange trekk fra den opprinnelige designmalen.



Figur 4 - Prosjektveilederen

Navigasjonen i prosjektveilederen skjer gjennom et panel på siden av applikasjonen. Man kan til enhver tid navigere enten fram eller tilbake i prosjektet for å legge til nye elementer, eller endre informasjon man tidligere har lagt til. Her er det altså greit å feile, enkelt å redigere, og man har kontinuerlig tilgang til rapporten sin. Slik kan læreren (og klassen) se gjennom jobben de har gjort og vurdere hvordan de ønsker å fortsette.

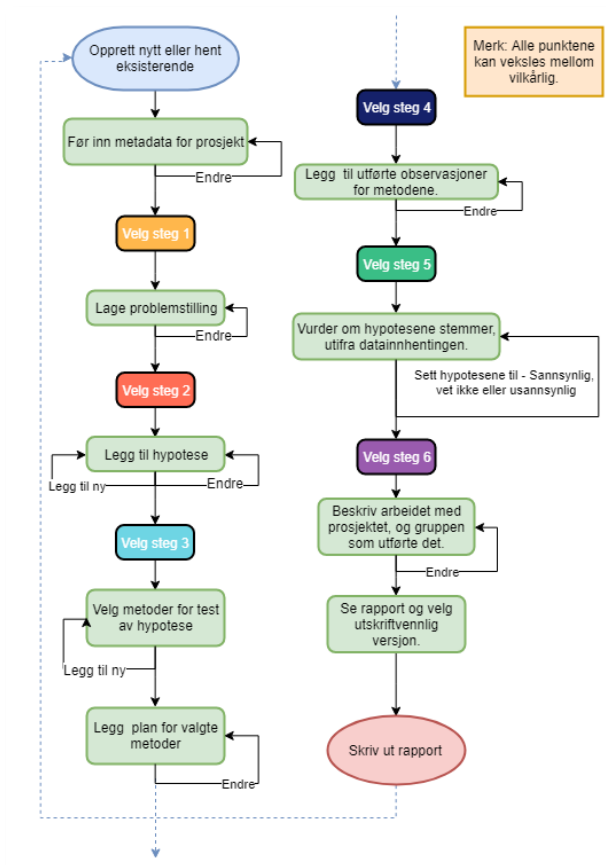
All informasjon fylt inn av brukeren blir lagret og kan, i tillegg til feltet der den fylles inn, vises som et utkast til en prosjektrapport under *Se rapport*. Her kan brukeren til enhver tid se prosjektet sitt, og endringer blir lagt til dynamisk mens elevene jobber. Denne rapportvisningen kan vises i en utskriftsvennlig versjon som kan benyttes som et utkast til videre arbeid, eller lagres og leveres til Nysgjerrigper-konkurransen.



Figur 5 - Navigasjonspanel



Figur 6 – Prosjektframvisningen



Figur 7 - Flowchart for bruken av prosjektveilederen.

1.5.2 ADMINISTRASJONSGRENSESNIITT

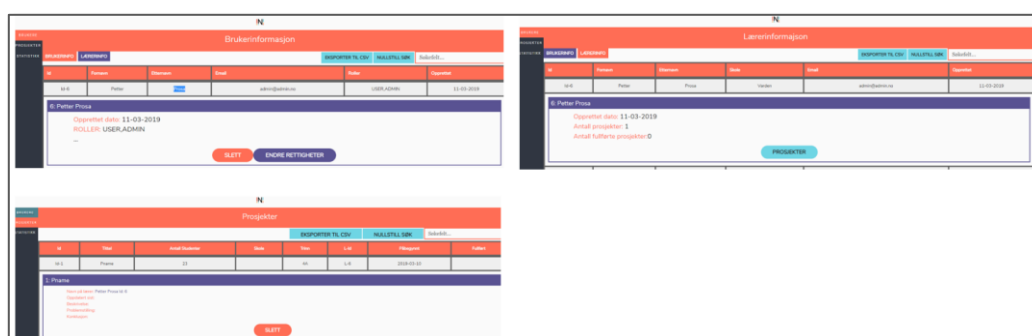
Nysgjerriger ønsket et solid administrasjonsgrensesnitt, for særlig å kunne se aktivitet og bruk av tjenesten i konkurranseperioden.

En bruker med administratorrolle kan gå inn på */admin*, for å aksessere og endre både brukere og prosjekter. Det er ment å være et ryddig oversiktstvindu for en administrator hvor det fås tilgang til informasjon om bruk.

Id	Fornavn	Etternavn	Email	Roller	Opprettet
Id-6	Petter	Prosa	admin@admin.no	USER,ADMIN	11-03-2019
Id-7	Guðrun	Guð	gud@vesterdahls.com	USER	10-03-2019
Id-8	Fritjof	Frisk	frj@het.no	USER	10-03-2019
Id-5	Dummy	Boy	user@user.com	USER	10-03-2019
Id-10	Petter	Prosa	admin@nysgjerriger.no	USER,ADMIN	11-03-2019
Id-2	Yanis	Mussis	forever@young.com	USER	01-03-2019
Id-4	Thomas	Toger	livets@hardskole.no	USER	05-03-2019
Id-1	Olav	Vestlie	olav@miehigh.no	USER	02-03-2019
Id-3	Vegardio	Leivosa	livin@briskeby.no	USER	03-03-2019

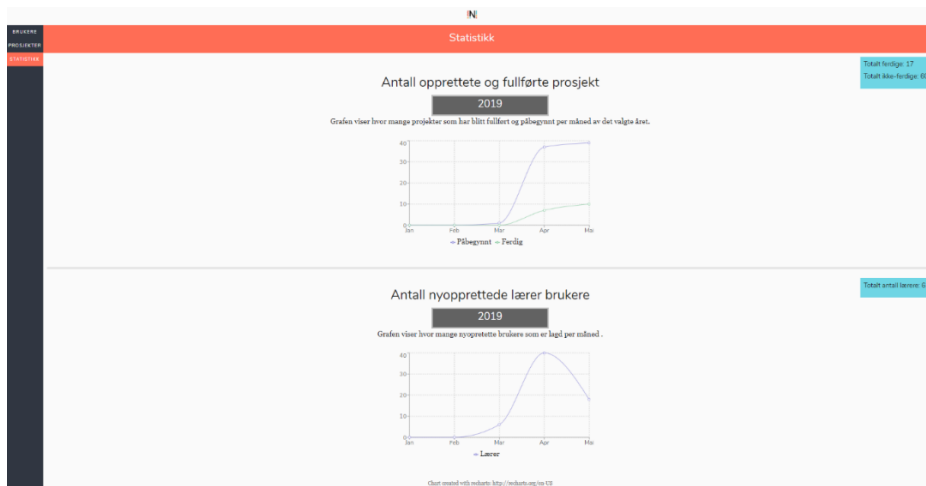
Figur 8 – Administrasjonsgrensesnittets landingsside

På administrasjonsgrensesnittet får systemadministratorer detaljert oversikt over alle brukere og prosjekter. All data kan eksporteres til CSV-filer, søkes i og sorteres. En administrator har mulighet til å tildele eller fjerne rettigheter, samt slette individuelle brukere og prosjekter.



Figur 9 – Funksjonaliteter ved valg av rad.

En administrator har i tillegg tilgang til visuell statistikk. I første omgang kan det vises grafer over antallet påbegynte og fullførte prosjekter, og antall opprettede brukere over tid.



Figur 10: Statistikk for applikasjonsbruk. Hvor det lages grafer som beskriver utvikling i løpet av et år.

1.6 KONKLUSJON

Da vi gjennomførte brukertestene i utviklingsperiodens avsluttende fase opplevde vi at lærerne som var med på gjennomføringen var svært positive til løsningen. Flere av dem hadde ikke brukt Nysgjerrigpermetoden tidligere, da de hadde fått inntrykk av at den var vanskelig å implementere i skolehverdagen og at det opprinnelige verktøyet opplevdes som tungt i bruk og vanskelig å finne frem i. Lærerne ble imponert over den nye løsningen og fikk lyst til å prøve den ut i skolesammenheng.

Forskningsrådet er fornøyd med applikasjonen ettersom den oppfyller problemstillingen, kravspesifikasjonen og de definerte brukerhistoriene, og de er meget imponert over omfanget av det som er levert. Forskningsrådet ser et stort potensial i denne applikasjonen.

Netcompany som oppdragsgiver har hatt utbytte av prosjektet på flere måter. Prosjektgruppen har produsert en applikasjon med et fundament som er solid og har store muligheter for videreutvikling. Den holder en så høy standard og har blitt så godt mottatt av Forskningsrådet, at prosessen for videreutvikling og produksjonssetting allerede er påbegynt.

Applikasjonen er en god demonstrasjon av hvordan en tilsvarende løsning, med behov for bruk av flere teknologier, kan bygges opp fra bunnen av i andre prosjekter. Prosjektdokumentet og kildekoden gir da særlig verdi i forhold til å vise hvordan en løsning med Java-Spring, React-Redux, PostgreSQL, bruk av GoogleAPIs og Contentful, og med en skyløsning i AWS, kan gjennomføres i et høyt utviklingstempo.

Prosjektgruppen hadde en kravspesifikasjon og en liste med brukerhistorier som skulle oppfylles i løpet av prosjektperioden. Disse oppgavene fullførte vi og Nysgjerrigper – Prosjektveilederen ble dermed godkjent under akseptansetesting hos kunden.

Applikasjonen har til hensikt å være et godt verktøy for lærere og skoleklasser slik at terskelen for å ta i bruk Nysgjerrigpermetoden senkes. Tilbakemeldingene fra brukere tyder på at vi treffer dette målet med vår løsning. Både vi og kunden er overbevist om at det vi har produsert kan bli et verdifullt redskap for lærere og elever, og gleder oss til en eventuell lansering i fremtiden. Applikasjonen kan bidra til at elever får undre seg over og forske på sine omgivelser. Vi mener med dette at prosjektet svarer godt på problemstillingen.

Til slutt ønsker vi å nevne at vi selv har hatt et kjempeutbytte av dette bachelorprosjektet. Vi har lært å bruke mange nye teknologier i en profesjonell setting og har fått erfaring med arbeidsmåter som vi vil ha glede av nå som vi skal ut i arbeidslivet. Samtidig har Netcompany fått muligheten til å se prosjektgruppen i arbeid og vi har alle blitt tilbudt jobb hos oppdragsgiver.

2 PROSESSDOKUMENTASJON

NYSGJERRIGPER – PROSJEKTVEILEDEREN

BACHELORPROSJEKT 2019

GRUPPE 5

INNHALDSFORTEGNELSE PROSESSDOKUMENTASJON

2.1	INNLEDNING.....	22
2.1.1	Om kravspesifikasjon og brukerhistorer.....	22
2.2	PROSJEKTVERKTØY.....	22
2.2.1	Generelle verktøy.....	23
2.2.1.1	Realtimeboard (Miro)	23
2.2.1.2	Slack	24
2.2.2	Arbeidsmetodikk - Scrum	24
2.2.2.1	Scrum team og Scrum master.....	24
2.2.2.2	Scrum sprinter	25
2.2.2.3	Daily Standups	25
2.2.2.4	Sprint retrospektiv	25
2.2.2.5	Produkteierens rolle.....	25
2.2.3	Atlassian suite.....	26
2.2.3.1	Confluence	26
2.2.3.2	Jira.....	27
2.2.3.3	Erfaringer med Jira:	28
2.2.3.4	Bitbucket	29
2.2.4	Utviklingsverktøy	29
2.2.4.1	IntelliJ	29
2.2.4.2	Sketch.....	29
2.3	PROSJEKTFASER.....	30
2.3.1	Status ved prosjektstart.....	30
2.3.2	Planleggingsfase.....	31
2.3.2.1	Uke 1 og 2: Introduksjonsuker	31
2.3.2.2	Uke 3: Brukerhistorier og viderutvikling av konsept.....	33
2.3.2.3	Uke 4 og 5: Risikovurderingen og kravspesifikasjon	34
2.3.3	Scrum-sprinter.....	35
2.3.3.1	Sprint 1	35
2.3.3.2	Sprint 2	38
2.3.3.3	Sprint 3	41
2.3.3.4	Sprint 4	43
2.3.3.5	Sprint 5	48
2.3.4	Akseptansetesting of feilretting.....	50
2.3.5	Ferdigstilling	50
2.3.6	Sammendrag.....	50
2.4	SAMSVAR MELLOM KRAVSPESIFIKASJON OG PRODUKT.....	51
2.5	POTENSIALE FOR VIDERE UTVIKLING	52
2.5.1	Forslag til utvidelser Backend	52
2.5.2	Forslag til utvidelser Frontend.....	52
2.5.2.1	Prosjektveilederen	52
2.5.2.2	Administrasjonsgrensesnittet:.....	52
2.5.3	Forslag til videre utvidelser for AWS skyløsningen:	53
2.5.4	Sikkerhet.....	53
2.5.4.1	Anbefalt inkrement til Google Authenticator-Implementasjonen	53
2.6	UTFORDRINGER OG REFLEKSJONER.....	55
2.7	KONKLUSJON TIL PROSESSDOKUMENTASJON.....	56

2.1 INNLEDNING

Prosessdokumentasjonen er en beskrivelse av hvordan vår prosjektgruppe har jobbet med utførelsen av prosjektoppgaven. Beskrivelsen er ment å kunne forstås av lekfolk og bruker relativt enkle forklaringer på tekniske emner. En dypere beskrivelse av teknologiene brukt vil finnes primært i [produktokumentasjonen](#). Dokumentet tar utgangspunkt i logger vi har skrevet underveis i prosessen.

I prosjektgruppen fikk vi muligheten til å utføre et bachelorprosjekt med en oppdragsgiver, *Netcompany*. Oppdraget var en stor utviklingsjobb for Forskningsrådet, som vi forholdt oss til som produkteier og kunde. Prosjektgruppen opplevde at oppdragsgiver og kunde begge ønsket at vi skulle ha en verdifull prosjekttid. De ga oss gode instruksjoner for arbeidet, men samtidig frihet til å utføre arbeidet slik vi så det hensiktsmessig.

Prosessdokumentasjonen starter med en beskrivelse av de verktøyene vi har brukt for å produsere prosjektet vårt. Deretter kommer hovedkapittelet, [prosjektfaser](#), som er en prosessbeskrivelse av hvordan arbeidet med prosjektet har gått fra begynnelsen fram til innlevering.

2.1.1 OM KRAVSPESIFIKASJON OG BRUKERHISTORER

Prosessdokumentasjonen beskriver arbeidet med kravspesifikasjonen og brukerhistoriene. I motsetning til «tradisjonelle» utviklingsprosjekter hvor utviklerne får en kravspesifikasjon, var dette dokumenter som prosjektgruppen selv utviklet for produkteier i begynnelsen av prosjektperioden. Kravspesifikasjonen summerer opp disse definerte brukerhistoriene. Både [kravspesifikasjon og brukerhistoriene](#) er vedlagt, og anbefales lest.

Vi snakker lite om selve løsningen i prosessdokumentasjonen, da dette dokumentet i stedet setter søkelys på selve arbeidsprosessen. Løsningen er beskrevet i [produktokumentasjonen](#). Det anbefales å se på de funksjonelle løsningene der, for å se dem i sammenheng med brukerhistoriene som bli beskrevet i dette prosessdokumentet.

2.2 PROSJEKTVERKTØY

Under prosjektperioden tok vi i bruk flere prosjektverktøy for å løse oppgavene. Dette inkluderte en rekke kreative verktøy. I dette kapitlet presenterer vi de overordnede verktøyene vi brukte under hele utviklingsperioden til kommunikasjon, oppgavefordeling, planlegging og ikke minst under utviklingsarbeidet. Disse er relevante for forståelsen av arbeidsprosessen og forteller mye om hvordan vi har jobbet med prosjektet.

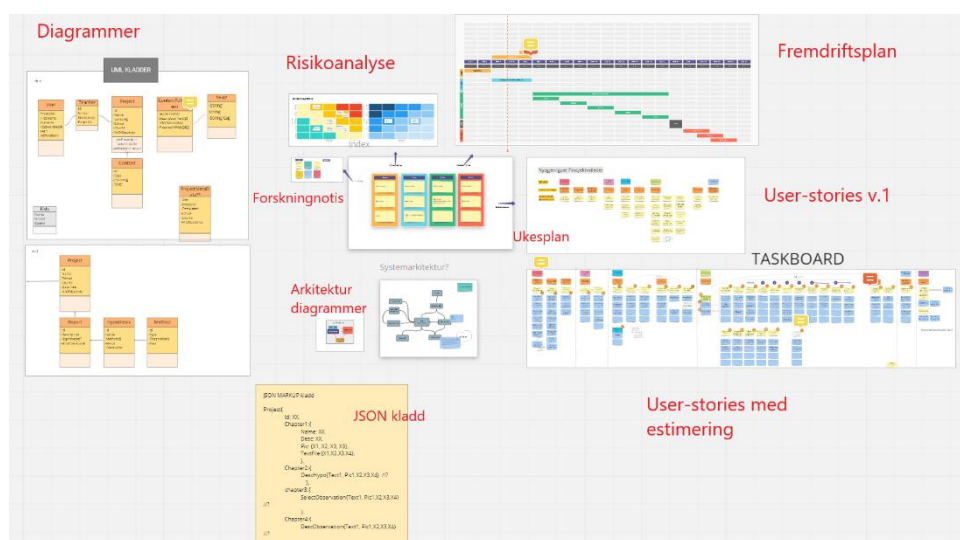
2.2.1 GENERELLE VERKTØY

Vi starter med å beskrive generelle verktøy som vi har brukt til å planlegge og dele ideer innad i gruppen. Disse har vært viktige for hvordan vi har kommunisert og sørget for at vi har hatt en felles forståelse av arbeidet underveis.

2.2.1.1 REALTIMEBOARD (MIRO)

Da vår konsept- og kravspesifikasjonsfase var forholdsvis lang, brukte vi tid på å finne et verktøy som gjorde denne prosessen mest mulig smidig for oss.

Dette fant vi i verktøyet *Realtimeboard*, som er et prosjektverktøy som gjør planlegging, konseptualisering og idémyldring visuelt og håndgripelig. Det er i bunn og grunn en "uendelig" stor tavle hvor flere brukere samtidig kan lage plansjer eller sette opp virtuelle Post-it lapper. Denne relativt enkle Post-it simulatoren har gjort vår planlegging bedre og enklere å forholde seg til.



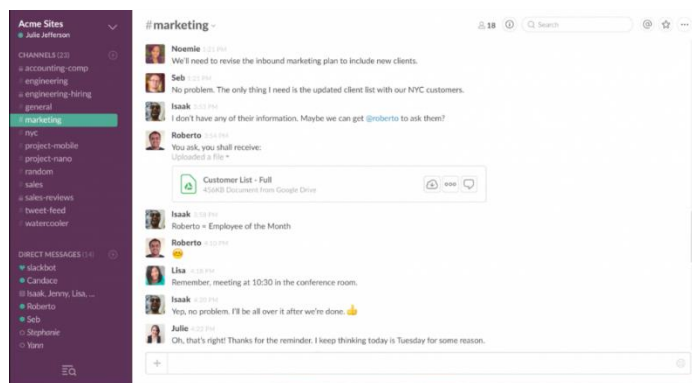
Figur 11 - Illustrasjonsbilde av alle tingene vi bruker Realtimeboard til.

Realtimeboard viste seg å være veldig nyttig til utvikling av brukerhistorier og gjorde det enkelt å lage idéutkast. Det er et godt verktøy for å illustrere tanker og idéer til resten av prosjektgruppen, samarbeidspartnere og kunder. Vi demonstrerte også verktøyet for andre grupper, og mange av gruppene tok det i bruk. Realtimeboard har nå byttet navn til Miro. (Miro, 2019)³.

³ Miro (2019). Team collaboration software & online whiteboard for teams.

2.2.1.2 SLACK

Ved oppstart ble det opprettet kontoer til oss i Slack, som er en kommunikasjonskanal som benyttes av oppdragsgiver. Den er egnet til rask og uhøytidelig korrespondanse innad i en organisasjon. Her opprettet vi en kanal hvor raske arbeidsrelaterte meldinger ble sendt frem og tilbake. Slack var vår hovedkommunikasjonskanal med ressurspersoner hos oppdragsgiver.

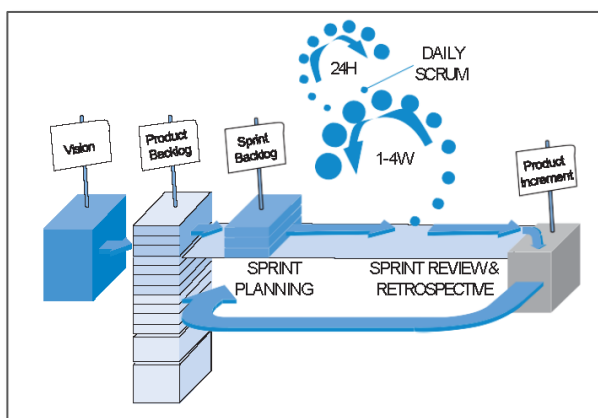


Figur 12 - <https://www.getmailbird.com/best-combine-email-messaging-apps-like-slack/>

2.2.2 ARBEIDSMETODIKK - SCRUM

Under utvikling av applikasjonen brukte vi smidig utviklingsmetodikken. Vi valgte Scrum for å ha en hyppig og inkrementell utvikling. (Schwaber & Sutherland, 2017)⁴

Scrum er et smidig rammeverk for programvareutvikling. Metodikken er godt egnet for inkrementelle endringer. I tillegg hadde produkteier et eksplisitt ønske om at vi benyttet oss av Scrum, for å kunne delta i prosessen som produkteier.



Figur 13 - <https://www.vanharenpublishing.com/wp-content/uploads/sites/28/2016/11/pag-Scrum-framework.png>

Dette kapittelet beskriver de viktigste begrepene i metodikken.

2.2.2.1 SCRUM TEAM OG SCRUM MASTER

I Scrum finnes det ulike roller. Scrum master er hovedansvarlig for utviklings-samspillet mellom medlemmene i gruppen og har det overordnede ansvaret over

⁴ Schwaber, K., & Sutherland, J. (2017). *The Scrum Guide™ - The Definitive Guide to Scrum: The Rules of the Game*. <https://www.scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-US.pdf>

oppgavefordeling og å sørge for jevn progresjon i løpet av sprintene. Scrum master sørger også for at det hver dag blir utført Scrum møter. Vi valgte Vegard Skullestad Bjørkgård som Scrum master for dette prosjektet.

Scrum teamet har vi definert til å være prosjektgruppen.

2.2.2.2 SCRUM SPRINTER

Da vi startet med Scrum-gjennomføringen hadde vi laget en grundig liste med brukerhistorier som var tilknyttet underordnede oppgaver. Disse prioriterte vi etter avhengigheter og viktighetsgrad. Vi estimerte hvor lang tid vi måtte regne med å bruke for å fullføre oppgavene, med verktøyet Planning Poker hvor alle gruppemedlemmene sammen estimerer oppgavene. Før hver sprint ble det holdt et planleggingsmøte hvor et utvalg av brukerhistoriene ble plassert i den kommende utviklingsperioden.

Prosjektgruppen hadde fem sprinter som hver hadde varighet på to uker. Vi satt sammen oppgaver som til sammen ga oss arbeidstimer som tilsvarte ca. 3 ukers utviklingstid, hvorav hver uke tilsvarte 37,5 timer, totalt fordelt på fire utviklere. Dette kan virke som mye tid til utvikling, men det hadde likevel en tendens til å bli jobbet mer med utvikling enn dette. Dette skyldes at vi underestimerte mange oppgaver og ikke alltid så avhengighetene mellom oppgaver.

2.2.2.3 DAILY STANDUPS

Scrum møter holdes hver dag i form av "daily stand-ups". Dette er korte møter som typisk varer under 15 minutter og skal holdes stående. Hvert medlem forteller hva de har gjort i løpet av det siste døgnet, hva de planlegger å gjøre fram til neste dag, og hvilke faktorer som kan hindre at oppgavene de holder på med blir utført. Vi tilpasset dette konseptet til vårt eget prosjekt.

Underveis i utviklingstiden introduserte vi en diskusjon om helhetsbilde til våre daily standups. Her hadde vi en diskusjon om den overordnede planen vår og om hva som skulle til for å få selve produktet levert innenfor tidsrammene. Dette var nødvendig da vi tidvis fikk skylapper for våre egne oppgaver.

2.2.2.4 SPRINT RETROSPEKTIV

Etter hver sprint forsøkte vi å avholde et refleksjonsmøte, hvor vi vurderte hva som gikk bra, eller kunne vært gjort bedre i den avsluttede sprinten. Dette gjorde oss bedre rustet til neste sprint, samt at vi fikk tatt opp eventuelle problemer innad i gruppen.

2.2.2.5 PRODUKTEIERENS ROLLE

Produkteieren er en representant fra kunden, som ved hver sprint er med på å prioritere hvilke oppgaver som er viktigst å gjennomføre. Eieren er også med på planleggingen

av akseptansetestingen. I vårt tilfelle så hadde vi også behov tilbakemelding fra eieren på hvilke oppgaver som kunne prioriteres bort, når scopet måtte reduseres.

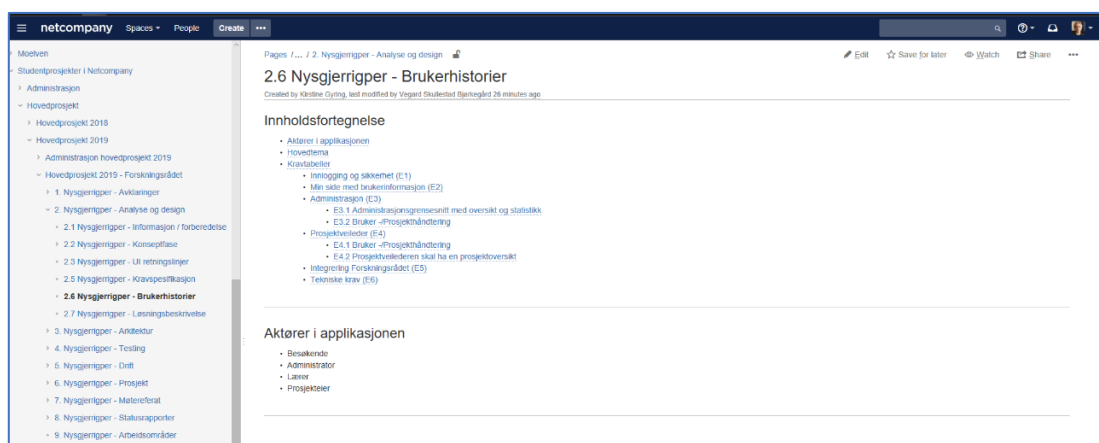
Produkteieren vår hadde en særskilt rolle i implementeringen av [Contentful](#) teknologien, fordi produkteier selv skal administrere denne teknologien videre.

2.2.3 ATLISSIAN SUITE

Atlassian Corporation er et australsk programvareselskap som utvikler produkter for utviklere, prosjektledere og innholdsforvaltere. De har utviklet Jira, Bitbucket og Confluence som alle er verktøy vi benyttet i utviklingsprosessen. (Caitlin Huston, 2015)⁵

2.2.3.1 CONFLUENCE

Confluence er en samarbeidsplattform for organisasjoner, som fungerer tilsvarende Wikipedia, men for intern bedriftskommunikasjon. Tjenesten brukes til å dele dokumentasjon, kommentarer og artikler innad i en bedrift. Det er det offisielle kommunikasjonsverktøyet som både Netcompany og Forskningsrådet benytter. Confluence fungerer som et forum, men med flere muligheter for redigering av innhold. Skriftlige arbeider, som kravspesifisering, brukerhistorier, fremdriftsplan og mer la vi ut her. Det ble satt opp en mappestruktur for prosjektet som fulgte Netcompany sin mal for prosjektstruktur, denne fylte vi opp med dokumenter. Fremgangsmåten var som regel å først levere kladder hos Netcompany som vi fikk tilbakemelding på og endret, før vi kopierte og flyttet disse over til kunden for akseptanse.

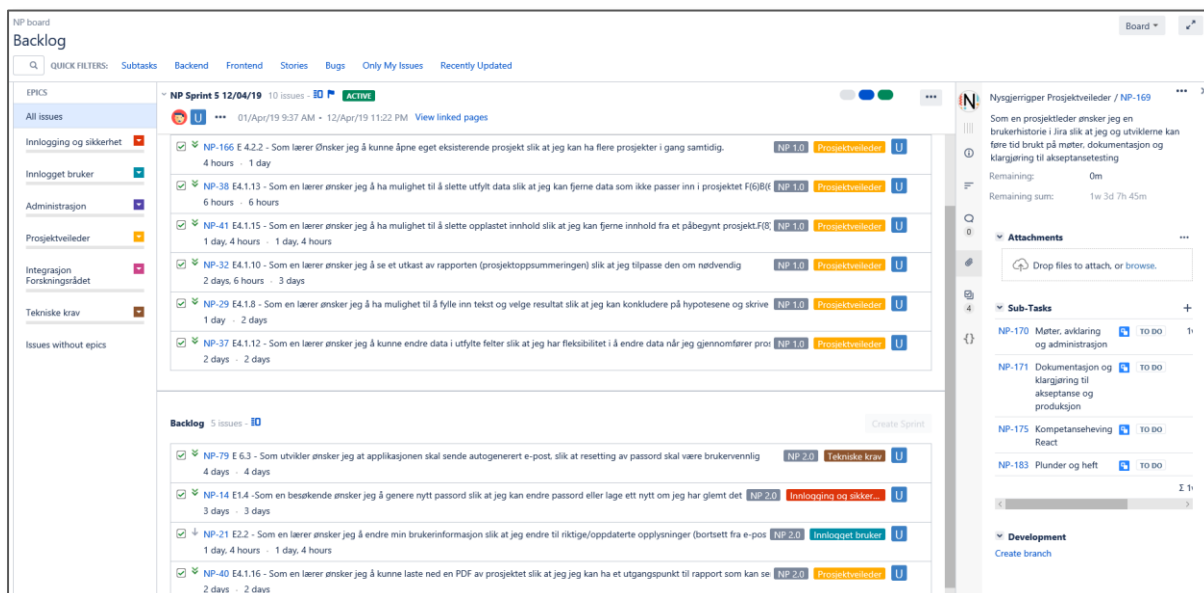


Figur 14 - Utklipp fra egen mappestruktur i Confluence

⁵ Caitlin Huston, C. H. (2015). *6 things to know about Atlassian as it goes public*. <https://www.marketwatch.com/story/6-things-to-know-about-atlassian-before-it-goes-public-2015-12-01>

2.2.3.2 JIRA

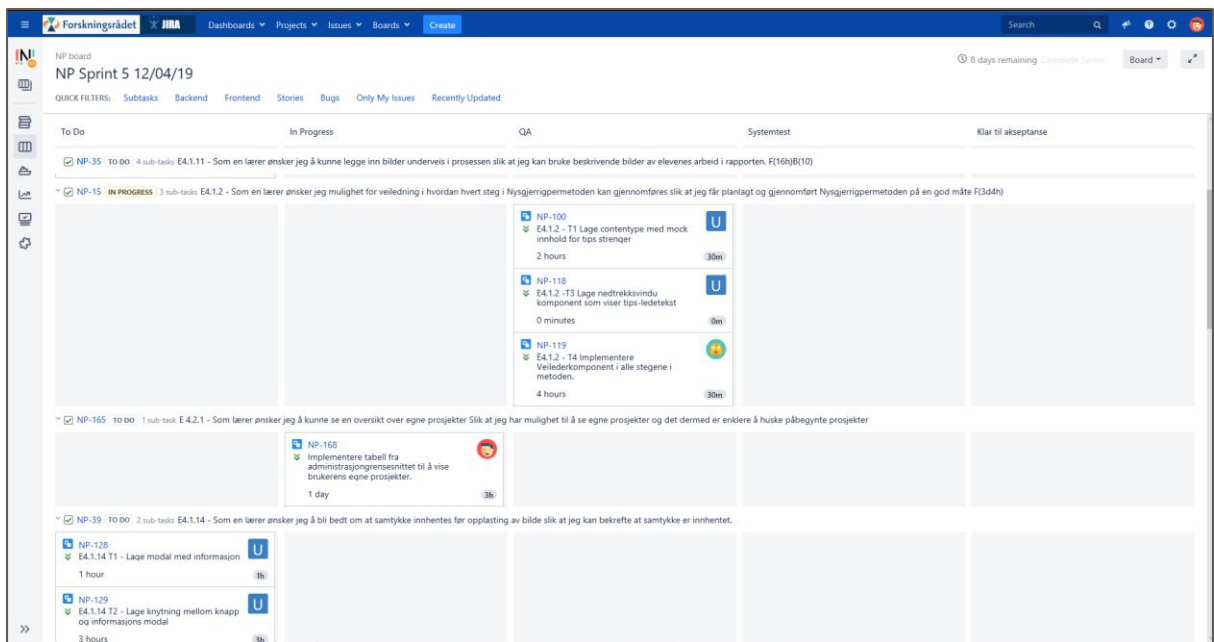
Jira er et Atlassian produkt som er laget for å benyttes til smidig arbeidsmetodikk. Dette var hovedverktøyet vi brukte for å planlegge arbeidet vårt gjennom sprintene. Det er et kraftig verktøy som gir utviklerne, oppdragsgiver og produkteier en god oversikt over hvordan utviklingsløpet forløper seg, med mulighet til blant annet å vise progresjonsstatistikk.



The screenshot shows a Jira backlog board for a project named 'NP board'. The board is titled 'Backlog' and has a search bar and quick filters for 'Subtasks', 'Backend', 'Frontend', 'Stories', 'Bugs', 'Only My Issues', and 'Recently Updated'. The board is divided into two main sections: 'EPICS' and 'Backlog'. The 'EPICS' section lists several categories: 'All issues', 'Innlogging og sikkerhet', 'Innlogget bruker', 'Administrasjon', 'Prosjektveileder', 'Integrasjon Forskningsrådet', and 'Tekniske krav'. The 'Backlog' section shows a list of user stories and tasks, each with a status, priority, and due date. The tasks are: NP-166 E.4.2.2 - Som lærer ønsker jeg å kunne åpne eget eksisterende prosjekt slik at jeg kan ha flere prosjekter i gang samtidig. (NP 1.0, Projektveileder, U, 4 hours - 1 day); NP-38 E4.1.13 - Som en lærer ønsker jeg å ha mulighet til å slette utfyllt data slik at jeg kan fjerne data som ikke passer inn i prosjektet F6(B)(E (NP 1.0, Projektveileder, U, 6 hours - 6 hours); NP-41 E4.1.15 - Som en lærer ønsker jeg å ha mulighet til å slette oppløst innhold slik at jeg kan fjerne innhold fra et påbegynt prosjekt.F6(B)(E (NP 1.0, Projektveileder, U, 1 day, 4 hours - 1 day, 4 hours); NP-32 E4.1.10 - Som en lærer ønsker jeg å se et utkast av rapporten (prosjektoppsummeringen) slik at jeg tilpasse den om nødvendig (NP 1.0, Projektveileder, U, 2 days, 6 hours - 3 days); NP-29 E4.1.8 - Som en lærer ønsker jeg å ha mulighet til å fylle inn tekst og velge resultat slik at jeg kan konkludere på hypotesene og skrive (NP 1.0, Projektveileder, U, 1 day - 2 days); NP-37 E4.1.12 - Som en lærer ønsker jeg å kunne endre data i utfylte felt slik at jeg har fleksibilitet i å endre data når jeg gjennomfører pro (NP 1.0, Projektveileder, U, 2 days - 2 days). The 'Backlog' section also shows a list of tasks: NP-79 E.6.3 - Som utvikler ønsker jeg at applikasjonen skal sende autogenerated e-post, slik at resetting av passord skal være brukervennlig (NP 2.0, Tekniske krav, U, 4 days - 4 days); NP-14 E1.4 - Som en besøkende ønsker jeg å generere nytt passord slik at jeg kan endre passord eller lage ett nytt om jeg har glemt det (NP 2.0, Innlogging og sikker..., U, 3 days - 3 days); NP-21 E2.2 - Som en lærer ønsker jeg å endre min brukerinformasjon slik at jeg endre til riktige/opdaterte opplysninger (bortsett fra e-pos (NP 2.0, Innlogget bruker, U, 1 day, 4 hours - 1 day, 4 hours); NP-40 E4.1.16 - Som en lærer ønsker jeg å kunne laste ned en PDF av prosjektet slik at jeg kan ha et utgangspunkt til rapport som kan se (NP 2.0, Projektveileder, U, 2 days - 2 days). On the right side of the board, there is a sidebar for a selected issue, 'Nysgjerriger Prosjektveileder / NP-169'. The sidebar shows the issue title, a description, remaining time (0m), and a list of attachments and sub-tasks. The sub-tasks include: NP-170 Meter, avklaring og administrasjon (TO DO, 1h); NP-171 Dokumentasjon og klargjøring til akseptanse og produksjon (TO DO); NP-175 Kompetanseheving (TO DO, React); NP-183 Plunder og heft (TO DO). The sidebar also has a 'Development' section with a 'Create branch' button.

Figur 15 – Utklipp av backlog

I figuren vises en liste med brukerhistorier som er med i kravspesifikasjonen for prosjektet. De øverste brukerhistoriene i listen var med i sprint 5 og de nederste er enda ikke plassert i en sprint, men ment for fremtidige sprints. I dette tilfellet er dette oppgaver som har blitt prioritert vekk, etter at det viste seg at vi ikke fikk levert alle brukerhistoriene i løpet av utviklingsperioden. Disse ble da stående igjen til fremtidige inkremer. Til venstre er hovedoppgavene (epics) listet, disse omfavner flere brukerhistorier innenfor ett funksjonsfelt. Til høyre ser vi en valgt brukerhistorie og dens oppgaver som er knyttet direkte til brukerhistoriens akseptansekrav.



Figur 16 – Utklipp av boardet som beskriver sprint 5

Figuren over illustrerer sprint 5 i Jira, og viser hvilke oppgaver som utføres, hvilken utvikler som jobber med dem og hvilke stadier de er på. Oppgavene helt til venstre er oppgaver som venter på at utviklerne begynner på dem.

NP-168 er en oppgave som er satt i gang, og man ser at det er estimert 3 timer igjen på den. Denne tiden synker når man logger brukt tid på den. Kvalitetskontroll (Quality Assurance) kolonnen innebærer at oppgavene er «ferdige» og en eller flere av de andre utviklerne vil sette av tid til å gå gjennom dem, og passe på at kvaliteten samsvarer med de satte rammene. Systemtest, er en kolonne som er dedikert til å teste oppgavene. Til sist når vi er ferdig med alle kolonene, flytter vi oppgavene til akseptanse-kolonnen. Da betyr det at oppgavene og brukerhistoriene de tilhører er klare til å testes av produkteier for å bekrefte at de møter kravspesifikasjonen.

Jira⁶

2.2.3.3 ERFARINGER MED JIRA:

Jira er et godt verktøy for å regulere brukerhistorier. Vi opplevde det allikevel noe begrensende. Blant annet fordi vi ikke kunne dele opp oppgavene underlagt en brukerhistorie. Vi hadde toukers sprinter, noe som kunne medføre at vi i noen tilfeller måtte lage nye midlertidige brukerhistorier, som vi fylte med deler av disse oppgavene, for å fylle en sprint uten å gå over den definerte tiden. Dette førte i noen tilfeller til litt rot.

⁶ Jira | Issue & Project Tracking Software | Atlassian <https://www.atlassian.com/software/jira?>

Vi hadde en diskusjon om vi skulle ha to Jira tavler, da vi jobbet i to *repositorier*. Fordelene med separerte Jira tavler hadde vært at vi hadde fått bedre oversikt over hvor mye tid vi hadde til disposisjon i frontend og backend. Vi valgte å ikke gjøre dette da det var til stor verdi for alle å ha god oversikt over hva andre drev med under utvikling.

2.2.3.4 BITBUCKET

Vi brukte Bitbucket som vårt git versjonshåndteringsverktøy. Bitbucket var koblet opp mot vår Jira tavle og gjorde at vi kunne lage grener tilknyttet våre brukerhistorier og oppgaver.

Når vi var ferdig med en oppgave opprettet utvikleren en pull request hvor en eller flere gruppede medlemmer vurderte koden opp mot overordnet krav til kodenstandard gitt av oppdragsgiver. Den/de som utførte kvalitetskontroll på denne pull requesten ville ved sammenslåing av grenen og utviklergrenen få valget om oppgaven skulle settes som ferdig i Jira.

2.2.4 UTVIKLINGSVERKTØY

Å ha gode utviklingsverktøy er helt sentralt i et profesjonelt utviklingsmiljø. Vi fikk alle utdelt lisenser av oppdragsgiver på utviklerverketøyet IntelliJ. Designansvarlig fikk også utdelt lisens på Sketch.

2.2.4.1 INTELLIJ

Som utvikler har man behov for et godt verktøy for å skrive kode, lage arkitektur, filer, klasser og pakker. Et såkalt Integrated Development Environment, forkortet IDE. Valg av riktig IDE er essensielt for effektiv utvikling og bruk av Version Control Systems (VCS), kodedokumentasjon og testverktøy.

Vi valgte å jobbe med IDE-en IntelliJ fra Netbeans. Dette av praktiske grunner, da oppdragsgiver stort sett benytter denne IDE-en, men også fordi det er en god IDE med alle de funksjonalitetene vi hadde behov for. IntelliJ er et Java-fokusert utviklingsmiljø med mange avanserte funksjoner. Blant annet gode database-, versjonshåndterings- og [Maven](#)-verktøy.

IntelliJ er mest egnet til backendutvikling, men for å ha en felles plattform, så valgte frontend også å bruke det. Dette var en beslutning som vi var forberedt på å kunne endre etterhvert, men det viste seg å ikke være nødvendig.

2.2.4.2 SKETCH

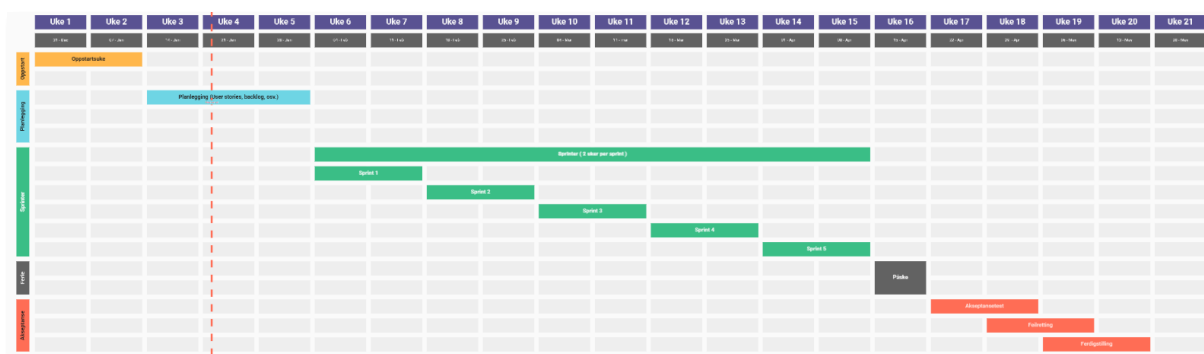
Sketch er et konsept/designverktøy som også kan brukes til å demonstrere koblinger i og utseende på nettsider. Det er svært visuelt og er godt egnet til å utvikle

designløsninger for nett-applikasjoner. Det viser flyten og manøvreringen på en applikasjon på en god måte. (Sketch)⁷

2.3 PROSJEKTFASER

Prosjektet vårt var delt opp i flere faser. Først to introduksjonsuker hos oppdragsgiver, deretter tre uker med konseptualisering, kravspesifisering og planlegging. I løpet av planleggingsfasen ble også kravspesifikasjonens brukerhistorier definert.

Deretter hadde vi fem Scrum-sprinters der hver sprint varte i to uker. Avslutningsvis hadde vi tre uker med akseptansetesting og feilretting. Planen vår passet overens med den planen prosjektlederen vår hadde. I dette kapitlet gjennomgår planleggingsfasen, uke for uke. Deretter følger gjennomgående beskrivelser av hvordan hver sprint utartet.



Figur 17 - Framdriftsplanen visualisert på Realtimeboard

2.3.1 STATUS VED PROSJEKTSTART

Det hadde blitt gjort noe forarbeid før prosjektgruppen startet med utviklingsprosessen. Blant annet hadde en av oppdragsgiverens tjenstedesignere utviklet et konsept som vi sto fritt til å videreutvikle da vi startet opp. Til dette konseptet fulgte det med et designutkast som var såpass bra at vi valgte å benytte og videreutvikle dette videre, og fikk dette godkjent av produkteier.

Vi hadde tilgang til en digital profil laget av Forskningsrådet. Denne profilen inneholdt farger, bilder og retningslinjer til hvordan man skulle bruke overskrifter, farger og ressurser som Nysgjerriger logoen, samt hvordan fargevalg og kontekst skulle påvirke disse. Den store utfordringen ved start var at prosjektgruppen selv måtte utvikle og definere kravspesifikasjonen, da kunden selv ikke hadde spesifikke tanker rundt denne.

⁷ Sketch. The digital design toolkit. <https://www.sketch.com/>

Vi syntes at Nysgjerrigper hadde en veldig god lærerveiledning (2016, Nysgjerrigper)⁸ for bruk av Nysgjerrigpermetoden. Dermed brukte vi mye fra dette heftet som designinspirasjon og som innhold til prosjektveilederen.

2.3.2 PLANLEGGINGSFASE

Forarbeidet omhandler prosjekteringen vi gjorde før vi startet selve utviklingen. Det ble satt av fem uker til planlegging, kompetanseheving og forarbeid. Det var på forhånd avklart med prosjektgruppen, oppdragsgiver og prosjekteier at Yanislav Todorov Zagorov ville mangle under denne fasen og introduseres til prosjektet i starten av utviklingsfasen, da han var på utvekslingsstudier fram til februar. Vi forsøkte å holde ham oppdatert, slik at han hadde god forståelse av prosessen i planleggingsfasen.

2.3.2.1 UKE 1 OG 2: INTRODUKSJONSUKER

Vi var tre bachelorgrupper som skulle gjennomføre hvert vårt prosjekt for Netcompany. To av gruppene var fra OsloMet og en fra Westerdals. Vi var mye sammen under introduksjonsukene.

De to første ukene av prosjektet ble brukt til å sette oss inn i oppdragsgiverens systemer og til introduserende forelesninger om forskjellige emner. Det var viktig for Netcompany at vi skulle sette oss inn i rollen som konsulent og representanter for firmaet. Vi fikk tildelt datamaskiner og satt opp kontoene våre hos Netcompany. Vi hadde diskusjoner rundt teknologivalg, blant annet om vi ville velge å bruke Java med [Spring](#) eller C# med .Net.

⁸ Nysgjerrigper (2016). Lærerveiledning - <https://nysgjerrigper.no/filearchive/nysgjerrigpermetoden2016.pdf>



Figur 18 - Gruppebilde med bachelorgruppene og prosjektlederne

2.3.2.1.1 NYE TEKNOLOGIER

Oppdragsgiveren ga oss en kort introduksjon i [React](#) og [Redux](#), Git og Postman, og Java-rammeverkene [Maven](#), [Spring Boot](#), [Spring Security](#), [Spring Data-JPA](#) og [Lombok](#). Å bruke de ekstra rammeverkene til Java var nytt og spennende, og vi så mange spennende muligheter for hva dette ville medføre. Den store utfordringen teknologimessig så vi for oss at ville være å lære React med Redux. Dette var noe vi kun var nybegynnere på med bare få ukers erfaring fra tidligere. Samtidig så vi også at det ville bli en utfordring å forstå bruken av rammeverkene til Java. Disse har en del automatikk som gjør oppgaver vi har gjort fra bunn av tidligere. Det var veldig mange teknologier vi skulle få til å jobbe sammen, og dette «*grugledet*» vi oss til å ta fatt på.

Samtidig var det en del andre teknologier vi bare hadde hørt om som vi skulle bruke. Viktigste av disse var [PostgreSQL](#), Contentful, og Amazon Web Services.

2.3.2.1.2 INSTALLERING PÅ ARBEIDSPLASSEN.

På slutten av to-ukers perioden installerte vi oss ute hos kunden, Forskningsrådet, på Lysaker. Der fikk vi ett stort kontor til prosjektgruppen. Over gangen for vårt kontor holdt Netcompany konsulenter til, inkludert prosjektlederen, noe som økte tilgjengelighet vi hadde til teknologisk assistanse.

Vi var på et større møte med Nysgjerriger og Forskningsrådet, hvor vi introduserte oss selv og hvordan vi så for oss tiden fremover. Det som var spesielt med vårt prosjekt var at kunden hadde en abstrakt målsetting, uten en konkret bestilling. Det førte til at vi som konsulenter, måtte produsere kravspesifikasjonen selv og få den

godkjent av kunden. Tiden videre ble brukt til brainstorming av hva kunden ville ha samtidig som vi ved ledig tid satt oss inn i React og Redux.

2.3.2.2 UKE 3: BRUKERHISTORIER OG VIDERUTVIKLING AV KONSEPT

Vi videreutviklet konseptet som tjenstedesigneren hadde arbeidet med. I den innledende konseptfasen ble det funnet at målsetningen til kunden, Nysgjerriger, var å bidra til at flere lærere tok i bruk Nysgjerrigermetoden. Ut fra dette konseptet og denne målsetningen brukte vi store deler av planleggingsfasen til å utarbeide brukerhistorier. Disse ble så avgrenset til en konkret applikasjon for eieren.

I uke 3 startet vi å definere brukerhistorier. Vi brukte en del tid på idémyldring og å sette opp brukerhistorier. Disse brukerhistoriene satt vi opp i [Realtimeboard](#) for å på en grafisk og lettlest måte kunne presentere dem for kunden. Vi brukte ledig tid på å sette oss inn i nye teknologier.



Figur 19 - Brukerhistorier, satt opp i Realtimeboard. De små tallene er estimater etter flere runder med planning-poker vi utførte.

Vi gjorde undersøkte forskning på interaksjonsdesign rettet mot barn. Forskningen vi fant fokuserte gjerne på blikkfang og spilldesign. Da vi undersøkte artiklene til *Revelle, Zuckerman, Druin, & Bolas, 2005*⁹, *Khanana & Law, 2013*¹⁰ og *Liu 2018*¹¹ fant vi noen generelle retningslinjer vi kunne følge. Blant annet at vi burde fokusere på store dynamiske elementer, bruke mye farger og spennende blikkfang i applikasjonen. Barn skal ifølge disse artiklene, ikke oppleve dette som støy slik som voksne gjerne gjør. Designet ble et kompromiss mellom disse retningslinjene og brukervennlighet. Applikasjonen er rettet mot lærere som kan bruke det sammen med elevene. Dermed

⁹ Revelle, G., Zuckerman, O., Druin, A., & Bolas, M. (2005). *Tangible user interfaces for children*.

¹⁰ Khanana, K., & Law, E. L.-C. (2013). *Designing children's digital games on nutrition with playability heuristics*. CHI '13 Extended Abstracts on Human Factors in Computing Systems on - CHI EA '13

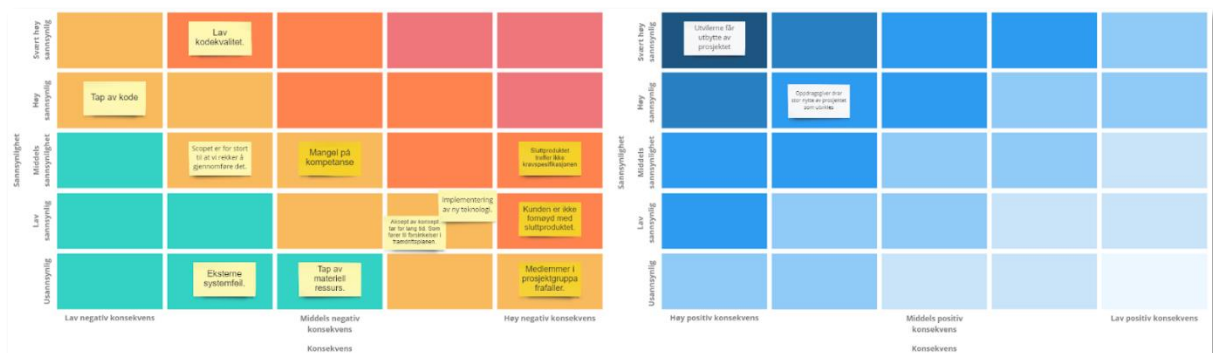
¹¹ Liu (2018). *Design for Kids Based on Their Stage of Physical Development*

må applikasjonen være en middelvei, mellom å kunne treffe lærere og samtidig kunne brukes sammen med elevene.

2.3.2.3 UKE 4 OG 5: RISIKOVURDERINGEN OG KRAVSPESIFIKASJON

Vi utarbeidet en risikovurdering sammen med prosjektlederen. Dette er et viktig dokument for å gjøre kunde, oppdragsgiver og oss selv oppmerksomme på hvilke risikoer som er relevante for prosjektet.

[Risikovurderingen](#), er et viktig og ofte nedprioritert dokument. I dette dokumentet søker man å finne alle mulige faktorer som kan ha innvirkning på prosjektets kvalitet, og hvordan man kan forberede og håndtere disse. Risikovurderingen inneholder en risikomatrix som er et diagram som oppsummerer og beskriver risiko i to dimensjoner, konsekvens og sannsynlighet. Resultatene av en risiko kan enten være positive eller negative. Risikovurderingen ble presentert for kunden og godkjent.



Figur 20 – Risikomatrix som ble utarbeidet for prosjektet

I uke 4 gjennomførte vi en workshop sammen med Forskningsrådet hvor vi gikk gjennom alle brukerhistoriene vi hadde laget. Forskningsrådet fikk mulighet til å komme med innspill til disse. Forskningsrådet var stort sett enige i brukerhistoriene vi hadde satt opp. Workshopen var nødvendig for å få godkjent brukerhistoriene før vi satt i gang med utarbeidelsen av kravspesifikasjonen som tok brukerhistoriene som utgangspunkt.

Da vi hadde fått [brukerhistoriene](#) godkjent laget vi et ferdig utkast til [kravspesifikasjonen](#), før vi gikk gjennom dette sammen med prosjektleder. Vi gjorde nødvendige endringer før kravspesifikasjonen ble presentert for Forskningsrådet. Denne kravspesifikasjonen ble godkjent. Deretter brøt vi ned kravspesifikasjonen og brukerhistoriene til mindre oppgaver, for så å gjøre et forsøk på å estimere omfanget av disse.

Det siste vi gjorde i planleggingsfasen var å legge inn brukerhistorier og oppgavene deres i Jira og bestemme hvilke oppgaver vi skulle ta fatt på i første sprint. Vi gjorde også vurderinger av hva som skulle være med de kommende sprintene deretter.

2.3.3 SCRUM-SPRINTER

Gruppen gjennomførte fem sprinter i løpet av prosjektet, med høy arbeidsmengde under hver av disse. Vi opplevde at intensiteten ble for høy i starten og at vi måtte prioritere bort enkelte brukerhistorier, slik at vi kunne bruke mer tid på andre for å komme i mål og levere kode av ønsket kvalitet. Det viste seg at det var svært vanskelig å estimere hvor mye tid en oppgave ville ta. En av grunnene til dette var at tiden det tok å tilegne seg nødvendige kunnskaper var sterkt varierende.

Vi har fulgt gode Scrum-prinsipper underveis, men vi hadde ikke alltid muligheten til å gjennomføre fullstendige sprint retrospektiv.

Prosjektgruppen delte seg opp i to tydelige definerte teams, *frontend* og *backend*. Backend skulle sette opp [Spring Boot](#) applikasjonen, databasen og serverløsningene. Frontend skulle lage klient-applikasjonen i React og Redux.

Hver sprint forgikk ved at vi hadde en prioritert *backlog*, hvor brukerhistoriene var prioritert i forhold til hverandre. Her tok vi høyde for avhengighetene de hadde til hverandre. Vi valgte ut brukerhistorier som vi til sammen hadde estimert til å ta ca. 120 utviklingstimer. Vi forsøkte å sette disse opp slik at vi kunne produsere iterasjoner av applikasjonen med inkrementelle utvidelser. Videre ønsket vi å sørge for at tidsestimeringen ga lik arbeidsbelastning i frontend og backend, noe som til tider viste seg å være vanskelig.

2.3.3.1 SPRINT 1

Sprint 1 foregikk i uke 6 og 7 av prosjektperioden vår. Det var første gangen vi skulle forsøke oss på Scrum basert arbeidsmetodikk. Vi lærte mye i løpet av denne første sprinten som ikke bare var faglig, men også om hvordan man jobber i team under en sprint.

I starten av sprint 1 kom Yanislav tilbake etter fullført utveksling og ble installert på arbeidsplassen. Dette var problemfritt, og etter kort tid begynte vi med arbeidsoppgavene.

Denne sprinten handlet i hovedsak om å få produsert kjørbare applikasjoner, med oppsett av utviklingsmiljøer og implementering av ulike rammeverk i applikasjonen.

2.3.3.1.1 PLANLEGGING AV FØRSTE SPRINT

I første sprintplanleggingsmøte som vi hadde med produkteier kom vi med et forslag til hva vi syntes var hensiktsmessig å få implementert i første sprint. Forskningsrådet sa seg enig i forslaget vårt, da de mente at vi hadde god forståelse over hva som måtte med i denne sprinten.

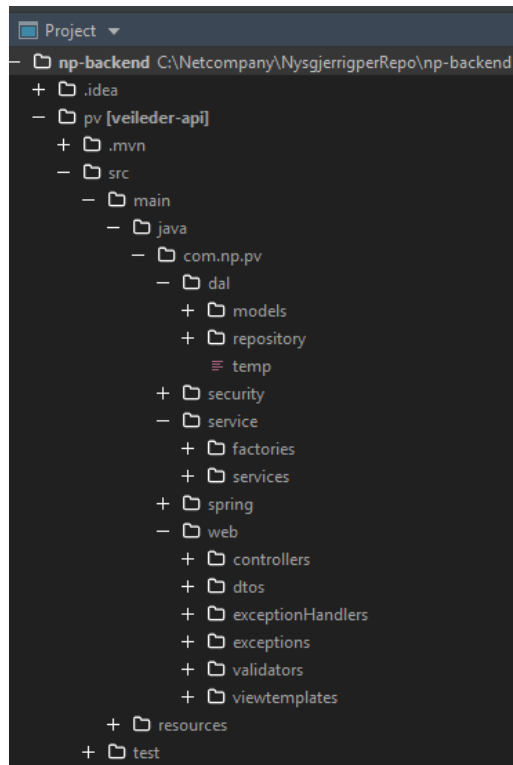
Planlagte brukerhistorier for første sprint:

- Som en besøkende ønsker jeg en landingsside slik at jeg kan få en kort introduksjon til Nysgjerrigpermetoden og hva jeg kan gjøre for å komme i gang.
- Som en prosjekteier ønsker jeg at applikasjonen bruker moderne teknologi tilsvarende lignende løsninger hos Forskningsrådet slik at eksisterende kompetanse kan benyttes, noe som igjen vil kunne føre til at vedlikehold og drift går raskt og er enkelt.
- Som en utvikler ønsker jeg en databasemodell for bruker slik at denne er klar når funksjonalitet som behandler brukere produseres.
- Som utvikler ønsker jeg et funksjonelt skjelett av applikasjonen slik at videre utvikling blir godt strukturert.
- Som utvikler ønsker jeg en databasemodell for Prosjektet, slik at det er klart for videre utvikling av funksjonalitet.

2.3.3.1.2 GJENNOMGANG AV FØRSTE SPRINT

Første sprint startet med at vi gjorde noen feil. I backend startet vi direkte på miljøoppsett i master grenen, noe som gjorde det vanskelig å utføre kvalitetskontroll på oppgaven da det ikke kunne gjøres ryddige pull requests mot denne grenen. Det var sterke avhengigheter i oppgavene som var planlagt for backend i sprinten, særlig at de var avhengig av domenemodellen, før andre oppgaver kunne påbegynnes. Heldigvis var prosjektleder helt klar på at første sprint mest sannsynlig ville være en prøve og feile sprint, noe hun hadde helt rett i. Frontend fokuserte på miljøoppsett og oppretting av struktur for prosjektveileder-seksjonen av nettsiden, samt bygging av et skjelett for en *MVP* (Minimum Viable Product). Ettersom mange av frontend teknologiene var ukjent for oss, brukte vi mye tid på å sette oss inn i dem.

I backend fokuserte vi også på å lage en logisk strukturert applikasjon, med håndtering av brukere og de funksjonelle modellene for applikasjonen. Vi satt opp en database, og jobbet for at applikasjonen skulle fungere konsistent opp mot databasen.



Figur 21 - Mappearkitekturen for java applikasjonen under spint 1. I backend valgte vi å bruke et tre lags prinsipp. Med dal, service og web. Hvor dal modellene og repositoryet håndterer entitetene opp mot JPA-hibernate rammeverket. Service håndterer business logikken som blir kalt på fra kontrollene i web. Mer utfyllende i Produktdokumentasjonen.

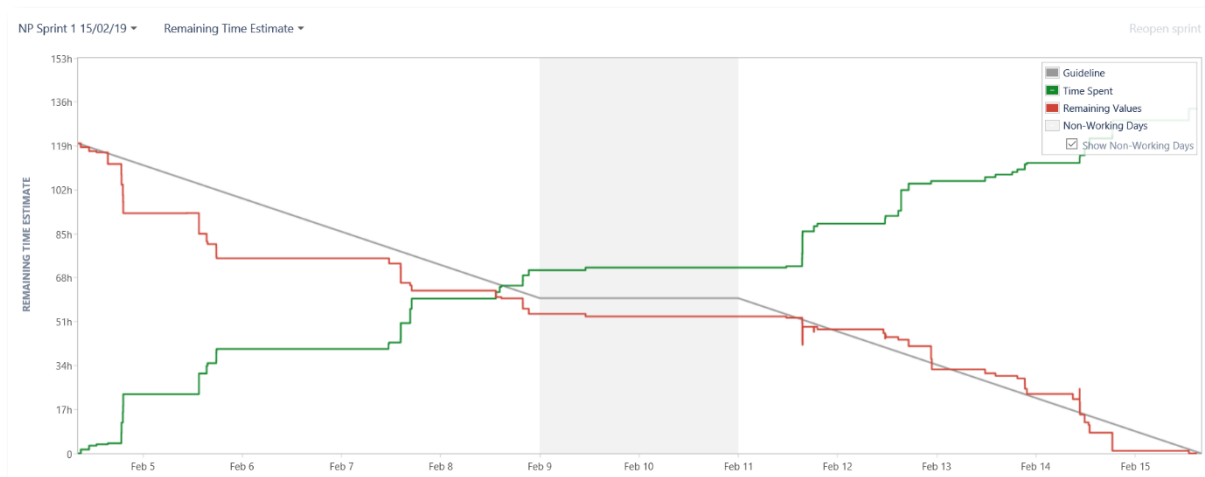
Det ble brukt mye tid på implementering av sikkerhetsløsningen knyttet til bruker. Det omhandlet validering av datafelter i databasen og implementering av JWT-løsning (JSON Web token) for autentisering av brukere. Med en guide på nett og god hjelp fra teknisk støtte hos Netcompany, klarte vi å implementere filtrene som filtrerer ut autorisasjonen fra forespørslene sendt til API-et. (*Protect REST APIs with Spring Security and JWT*)¹² Roller og tildeling av tilgang ble også satt opp.

Vi støtte imidlertid på en utfordring; JWT token-tillegget som guiden på nett foreslo at vi kunne bruke (io.jsonwebtoken) implementerte et Java-integrert klassebibliotek som hadde blitt byttet ut i nyere versjoner etter JDK 8. Vi valgte derfor å bytte ut biblioteket med et annet (org.auth0.jwt).

2.3.3.1.3 SPRINTRETROSPEKTIV

Det som var bra	Det vi kan bli bedre på	Hva holdt oss igjen?
<ul style="list-style-type: none"> Fikk produsert veldig mye innhold Jobbet bra sammen i de adskilte gruppene våre 	<ul style="list-style-type: none"> Lite kommunikasjon mellom frontend og backend Lite kundekontakt 	<ul style="list-style-type: none"> Feilestimert tid for oppbygging av kompetanse Kompatibilitetsproblemer

¹² Protect REST APIs with Spring Security and JWT. (2018, August). <https://medium.com/@hantsy/protect-rest-apis-with-spring-security-and-jwt-5fbc90305cc5>



Figur 22 - Burndown chart av første sprint.

Et burndown chart er en grafisk representasjon av hvor mye arbeid vi har utført i en sprint sett i kontrast med hvor mye tid som har blitt estimert på oppgavene som inngår i sprinten.

- Time spent – tid loggført på oppgaver for sprinten.
- Remaining values – estimert gjenværende tid på oppgavene i sprinten

Grafen ovenfor er hentet fra Jira. Merk at denne beveger seg relativt synkront med hvordan tiden er estimert. Dette vil si at vi traff på estimatene våre denne sprinten og vi leverte dermed det vi hadde planlagt at vi skulle levere. Denne grafen flater ut på onsdager og i helgen, da vi stort sett ikke jobbet med utvikling disse dagene.

2.3.3.2 SPRINT 2

Andre sprint utspant seg over uke 8 og 9 av utviklingen. Det ble fokusert på skyløsning, samt en illustrativ landingsside.

2.3.3.2.1 PLANLEGGING AV ANDRE SPRINT

Rett etter at vi fullførte sprint 1, satt vi i gang planlegging av sprint 2. Tiden ble brukt på å komme litt i forkant på neste sprint, gjennom blant annet å bygge oss opp en forståelse av Contentful.

Planlagte brukerhistorier for andre sprint:

- Som en lærer ønsker jeg en introduksjon til Nysgjerrigpermetoden slik at jeg får kjennskap til prosessen.
- Som en besøkende ønsker jeg å kunne opprette en bruker for applikasjonen slik at jeg kan bruke den i undervisningen.
- Som prosjekteier ønsker jeg at applikasjonen kan ligge i skyen, slik at alle kan nå den fra en nettleser.
- Som en lærer ønsker jeg å kunne opprette ett nytt prosjekt slik at jeg får planlagt og gjennomført Nysgjerrigpermetoden på en god måte.

Prosjektlederen sa seg enig i at brukerhistoriene var hensiktsmessige å gjøre i denne sprinten. Det viste seg dog senere at det tok mer tid enn estimert å få publisert innholdet i skyen. Både fordi det ikke var satt opp en konto for prosjektet på skytjenesten Amazon Web Services (heretter kalt AWS) og fordi det var vanskelig å sette opp riktig.

Prosjektgruppen startet sprinten med et sprintplanleggingsmøte sammen med oppdragsgiver. Oppdragsgiver sa seg enige i hva vi planla å gjøre i denne sprinten. Vi viste dem det vi hadde produsert frontend, og de ga inntrykk av at de var imponert over hvor mye vi hadde fått gjort.

2.3.3.2.2 GJENNOMGANG AV ANDRE SPRINT

Begynnelsen av sprinten ble brukt til å rydde i kodegjeld etter første sprint. Deretter gjorde vi undersøkelser på AWS og serverløs løsningen til Amazon. Vi hadde i oppgave å sette opp AWS-arkitekturen selv for å levere en fullstendig skyløsning. Vi fikk en introduksjon til AWS av teknisk støtte hos oppdragsgiver og diskuterte hva slags arkitektur vi skulle opprette, med eksempler fra andre prosjekter hos Forskningsrådet. Vi endte opp med å benytte Elastic Beanstalk fordi det var mer modent og derfor lett å finne informasjon om det på internett og fordi det var god kompetanse på det hos teknisk støtte. Det var imidlertid litt utfordrende å sette opp AWS-miljøet, blant annet var det problematisk å følge startguidene til AWS uten forbehold. De tok blant annet ikke opp nødvendigheten av en Virtual Private Cloud for å knytte kontakt mellom database (RDS) og API (EC2). Teknisk støtte hjalp oss med å sette opp AWS arkitekturen. Mer om denne og de forskjellige tjenestene vi benyttet i [Produktokumentasjon](#). Vi lærte veldig mye i denne fasen, blant annet hva de forskjellige tjenestene vi benytter gjør og er, hvordan man publiserer applikasjonen i skyen og feilsøking av AWS tjenester.

Vi jobbet med implementering av flere tester og bedre validering av input i backend. Det ble også brukt tid på implementering av [databasemigreringsverktøy](#) for databasen. Noe som gjorde det nye enklere å publisere API-et til skyen ved endringer

i databasen, da alternativet hadde vært å benytte en bakkdør for administrering av databasen. Dette hadde vært mer tidkrevende enn å gjøre endringer ved hjelp av migreringsverktøyet.

Samtidig ble frontend applikasjonen påbegynt med en landingsside med introduksjon til Nysgjerrigpermetoden. Denne siden var lite imponerende visuelt sett, men var vår første introduksjon til kodespråket React. Dette startet diskusjonen om hvordan strukturen til hele webapplikasjonen skulle se ut til slutt og vi endte opp med vår første arkitektur i React.

På grunn av sykdom måtte teamlederen ta over som Scrum-master i uke to av sprinten. Uken var utfordrende, da vi nå lå bak skjema på grunn av sykdom og hadde store teknologiske utfordringer knyttet til AWS oppkoblingen. Dette resulterte i en hard sprint hvor vi måtte ta igjen mye arbeid mot slutten.



Figur 23 - Landingssidens utseende i løpet av sprint 2.

2.3.3.2.3 SPRINTRETROSPEKTIV

Vi holdt ikke en offisiell retrospektiv ved slutten av andre sprint grunnet sykdom, men hadde mye intern diskusjon angående hvordan sprinten gikk ellers.

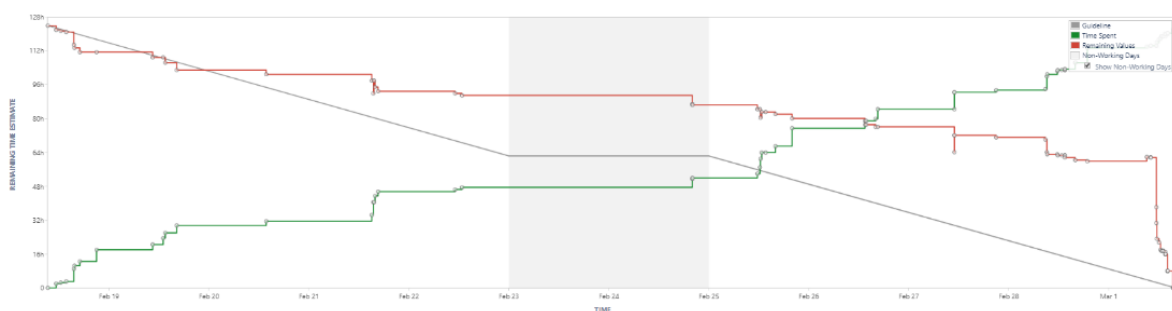
Generelt kan vi si at utviklingen gikk tregt. Både frontend og backend opplevde problemer med infrastrukturen, og sykdom tok det ene gruppemedlemmet bort fra gruppen i en uke. Allikevel klarte vi å hente oss godt inn mot slutten av sprinten, men et par oppgaver måtte tas med over i neste sprint.

Den største milepælen i denne sprinten var å få oppgaven hostet i AWS-skyen. Dette ble den største hindringen vi hadde opplevd til nå.

Det ble kjørt to runder med brukertesting sammen med produkteieren, hvor produkteier fikk mulighet til å komme med tilbakemelding på landingssiden. Vi fikk gode tilbakemeldinger, og implementerte endringene produkteier ønsket før slutten av sprinten.

Vi falt bakpå den første uka fordi det ene gruppemedlemmet var syk, men vi tok opp for den tapte tiden i løpet av sprintens andre uke.

Her ble det synliggjort at arbeidsmengden vi hadde nok var for stor og at vi muligens måtte justere størrelsen på scopet for prosjektet.



Figur 24 - Burndown chart av andre sprint. Grafen viser at vi ligger et stykke unna guidelinen gjennom nesten hele sprinten. Dette kom av sykdom. Men at det plutselig løsner på slutten kommer av at det var en teknisk propp som satt fast i både front end og back end. Da den endelig løsnet, fikk vi utrettet fryktelig mye på kort tid.

2.3.3.3 SPRINT 3

Utspant seg over uke 10 og 11. Sprinten hadde fokus på å sikre brukerautentisering og *-autorisering*. Med implementasjoner av *JWT* og tilganger basert på brukerroller.

Den ene backend utvikleren måtte tilegne seg frontend evner for å også kunne jobbe som frontendutvikler. Dette kom av at vi så at det var mer å gjøre i frontend enn i backend.

2.3.3.3.1 PLANLEGGING AV TREDJE SPRINT

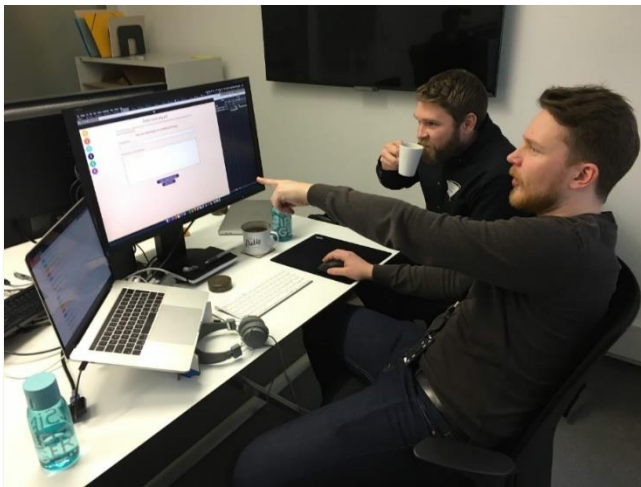
Sprint tre hadde disse brukerhistoriene. Tanken var å forsøke å lage en litt snillere sprint enn i sprint 2. Med ryddigere og mer konkrete oppgaver, som ikke hadde for sammensatte avhengigheter for progresjon.

Planlagte brukerhistorier for tredje sprint:

- Som en besøkende ønsker jeg å kunne logge på min registrerte bruker slik at jeg kan bruke applikasjonen.
- Som en lærer ønsker jeg å få informasjon/tips om hva som kan være gode problemstillinger og mulighet til å fylle inn tekst slik at jeg kan legge til en problemstilling og en beskrivelse av denne.
- Som en lærer ønsker jeg å få informasjon/tips om hva som kan være gode hypoteser, og mulighet til å fylle inn tekst slik at jeg kan legge til hypoteser.
- Som en lærer ønsker jeg å få informasjon/tips om hvordan de ulike metodene kan benyttes, og ha mulighet til å velge metoder slik at jeg får lagt en plan for gjennomføringen av undersøkelse.
- Som en lærer ønsker jeg å ha mulighet til å fylle inn tekst slik at jeg kan beskrive gjennomføringen av undersøkelsene som skal teste hypotesene
- Som en utvikler ønsker jeg at frontend og backend er koblet sammen, slik at applikasjonen fungerer som tiltenkt

Det overordnede målet for sprinten var å få backend og frontend synkronisert. Samtidig å også prøve å ha en jevn progresjon på utviklingsoppgavene.

2.3.3.3.2 GJENNOMFØRING AV TREDJE SPRINT



Figur 25

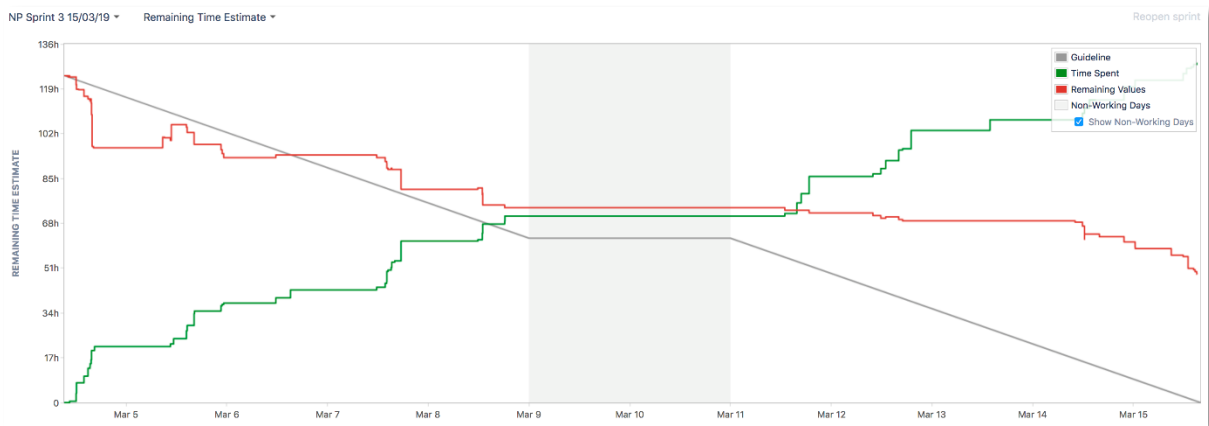
Sprinten startet sterkt ved at vi helt i begynnelsen kuttet ned estimatene svært mye. Tre oppgaver som omhandlet to-faktor autentisering som vi antok at til sammen ville ta over to hele arbeidsdager å gjennomføre, fikk vi gjort på kun en arbeidsdag.

Senere støtet vi på utfordringer ved implementasjon av sikkerhetslaget til [Spring Security](#). Vi brukte tid på å forstå på hvordan vi skulle implementere token autorisering i backend. Spring Security utførte operasjoner på token-header informasjon, og det var vanskelig å teste operasjonene da det var svært mange lag med metoder å undersøke i security-rammeverket.

Etter hvert begynte de forskjelligerammeverkene å fungere stabilt sammen i applikasjonen, noe som gjorde det egnet for at den ene backend utvikleren beveget seg mer over i frontend. Vi satt i gang planlegging av brukertesting, mens vi enda kunne få verdi ut av dem under utviklingsperioden.

Det ble planlagt at det måtte utføres et stort skippertak i løpet av helgen. Da det ikke så ut som det ville være overkommelig å utvikle administrasjonsgrensesnittet innenfor prosjektperioden.

2.3.3.3 SPRINTRETROSPEKTIV



Figur 26 - Problemene som er beskrevet i kapitlet ovenfor kommer til uttrykk i dette chartret. Ser at det særlig i uke to er lite fremgang under utviklingen. Flere oppgaver ble skjøvet til neste sprint.

Det viste seg at det var svært vanskelig å synkronisere hele gruppen og vi fikk en del forsinkelser da vi prøvde å løse dette. Det var særlig implementasjonen av JSON web token-løsningen som var utfordrende og dette forsinket utviklingsflyten. Vi skjønnte dermed at vi burde ha jobbet tettere og planlagt mer samlet på tvers av frontend og backend.

På grunn av tidspresset for å få fullført sprinten, fikk vi heller ikke denne gangen tid til å gjennomføre en retrospektiv.

2.3.3.4 SPRINT 4

Den fjerde sprinten utspant seg over uke 11 og 12.

Mange viktige oppgaver ble gjennomført i løpet av fjerde sprint. Vi fikk blant annet laget en ny side for brukerinstillinger, og laget en bedre integrasjon mellom frontend og Contentful. Denne integrasjonen sørger for at Forskningsrådet enkelt kan endre innhold på siden uten støtte fra oss. I tillegg startet vi å implementere enhetstester i backend for å avdekke eventuelle feil og mangler.

FRONTEND			BACKEND						
Request	Beskrivelse av body(Post eller response)	Beskrivelse av variabel/felt	Hvem	Validert	Utført		Entiteter	Branch	Hvem
Post :/projects	post - ny var	string projectOwner	Vegard		v	lagt til felt i projectOwner i dto		178	Thomas
-/-	post - valgfri	string school, grade	-/-		v			-/-	-/-
-/-	post - valgfri	string course	-/-		v		Project	-/-	-/-
Post/Put :/projects	post - valgfri	string projectOwner	Vegard		v				
NY Post User/MetaData	post - authToken	string username, string email	Yaniz		v	GET /users	ret: userTemplate	178	Thomas
put :/projects	put - andre navn	int? id må hete projectid	Vegard		v	endret dto		-/-	-/-
project	projectDTO	trenger å kunne få tilbake det template'n sender. Den sender description, researchquestion og finalconclusion som null, men måker å ta imot de samme feltene som null tilbake ved put.							
NameValidator	Trenger å ta imot tall		Veggis		v	Text validator skal akseptere null nå		178	Thomas
Resp: hypTemp ny var	int nrHypo		Olav		v			DEV	Olav
IDTO required beskrivelse			Thomas		v				
project dto linjeskift			Olav		v				Olav
Text Validator	apostrof, bindestrekk		Thomas						Olav
post :/projects	post - validering	string school - needs to be valid with "-"							Olav
post/put/get for metode	methodid ny var byttes inn istedet for "type"	int methodid - må være required på post. å sendes med på get ikke nødvendig på put (ending av metodetype skal ikke være mulig)							
get project		Mulighet for å hente en brukers sist brukte prosjekt?							Thomas
/projects PUT -> Template		string aboutUsers			V				
POST/PUT /aboutProject		string aboutProject	Thomas		v	lagt request løkker for /projects put for begge	Project	pull-req -> dev	Thomas
Post :/projects	stengefeltene trenger å kunne ta imot bindestrek		Veggis						
Put :/methods	Trenger en kopi av enumsløsningen fra prosjekt inn i Me		Veggis		V			dev	Thomas
put/post/get (alle) /hypothesis	returntemplate (?)	returnere TRUE/FALSE/UNKNOWN (Det samme som an mottas) i feltet for resultType	Veggis		v	v			
Post/Put/Get :/methods /hypothesis	returntemplate. Går det an at den returnerer en tom string for "" for "plan" og "observation" i stedet for null? Pretty please :D	returnerer nå null for tomme felt, men godtar ikke null som data i en request. Samt tekstfelder og lignende som henter dataene klager og spørsmål/warnings om dataen er null.	Veggis		v				
hypothesis (alle reqs)		resultType. Sender en tom streng tilbake om den ikke er satt, men godtar ikke tom streng i eksempelvis put requests. Må godta tom streng i put				v	Revurdert. Trengs ikke.	Method	dev
									Vegard/Thomas

Figur 27 – Synkroniseringsplan. Da vi i forrige sprint så at vi slet med synkronisering mellom frontend og backend, så vi et behov for en ryddig løsning på samarbeid mellom partene. Slik at frontend kunne komme med bestillinger, eller feilmeldinger, som da backend kunne utbedre ved ledig tid.

2.3.3.4.1 PLANLEGGING AV FJERDE SPRINT

Det ble satt opp mange brukerhistorier for denne sprinten. Likevel var mange av disse tett knyttet til hverandre og relativt små, så totalestimatet over tid var likevel innenfor 120 utviklingstimer. Samtidig var mange av administratoroppgavene nesten ferdige i løpet av helgen under et skippertak, da det viste seg at vi ville få store problemer med å fullføre dem i løpet de resterende sprintene.

Planlagte brukerhistorier for fjerde sprint:

- Som en lærer ønsker jeg å kunne se min brukerinformasjon slik at jeg verifisere at opplysningene er korrekte.
- Som en lærer ønsker jeg å se hvordan jeg skal gå frem for å få slettet brukeren min slik at jeg kan fjerne den ved ønske.
- Som en lærer ønsker jeg å få informasjon/tips om hvordan de ulike metodene kan benyttes, og ha mulighet til å velge metoder slik at jeg får lagt en plan for gjennomføringen av undersøkelser.
- Som lærer ønsker jeg å ha mulighet til å fylle inn tekst slik at jeg kan beskrive gjennomføringen av undersøkelsene som skal teste hypotesene.
- Som en lærer ønsker jeg å ha mulighet til å fylle inn tekst og velge resultat slik at jeg kan konkludere på hypotesene og skrive en endelig konklusjon på problemstillingen for prosjektet.
- Som en lærer ønsker jeg å kunne endre data i utfylte felter slik at jeg har fleksibilitet i å endre data når jeg gjennomfører prosjektet
- Som en lærer ønsker jeg tekstfelter for introduksjon av klassen og beskrivelse av prosessen slik at dette kan legges til i rapporten.

Administrasjonsgrensesnitt

- Som en administrator ønsker jeg et administrasjonsgrensesnitt slik at jeg kan gjøre administrative oppgaver.
- Som en administrator ønsker jeg å kunne se alle opprettede prosjekter på en bestemt bruker slik at jeg har mulighet til å se prosjektene som hører til en bestemt person.
- Som en administrator ønsker jeg å kunne se tall over påbegynte, men ikke ferdigstilte prosjekter slik at jeg har mulighet til å ha en formening om hvor mange prosjekter som blir påbegynt men ikke ferdigstilt.
- Som en administrator ønsker jeg å kunne se tall over ferdigstilte prosjekter slik at jeg har mulighet til å ha en formening om hvor mange prosjekter som blir fullført.
- Som en administrator ønsker jeg å kunne se alle brukere av applikasjonen slik at jeg får en oversikt over brukerne av applikasjonen.
- Som en administrator ønsker jeg å kunne slette en lærer-bruker slik at jeg kan håndtere feil ved brukere, eller ønske om at bruker fjernes.
- Som en administrator ønsker jeg liste over prosjekter med mulighet for sletting slik at jeg kan slette ett prosjekt for en bruker

2.3.3.4.2 GJENNOMFØRING AV FJERDE SPRINT

Vi startet sprinten med å lage brukertestdokumenter (samtykkeskjema og brukertesten) som vi sendte til Netcompany for gjennomgang. Vi fikk gode tilbakemeldinger på disse, og gikk videre med sending av eposter for å få tak i lærere som var villige til å være med på brukertesten. Vi hadde dårlig hell med dette under denne sprinten, men det viste seg å likevel løse seg i sprint 5.

[Cookies](#) ble integrert i frontend-applikasjonen. Dette er informasjonskapsler som lagrer brukerens autorisasjonstoken lokalt. I praksis betyr dette at brukeropplevelsen ble mer stabil. Mange mindre funksjonaliteter ble implementert, slik som panel for

brukerinformasjon og valg og visning av undersøkelsesmetoder i applikasjonen. I tillegg ble en modell for gjenbruk av komponenter i applikasjonen utviklet. Denne demonstreres i figur 28, som viser en komponent som blir gjenbrukt gjennom hele prosjektveilederen.

Hva tror dere kan være svaret på problemstillingen deres?
Skriv ned noen forslag til mulige forklaringer. Slike forklaringer kalles hypoteser.

Tips til hvordan man kan lage gode hypoteser ▽

Planlegg hvilke undersøkelser dere vil gjøre for å finne ut om hypotesene stemmer eller ikke. Bruk gjerne flere undersøkelser for å hente opplysninger. Dere kan for eksempel intervju, observere eller gjøre egne forsøk.

Tips til hvordan man kan legge en plan ▽

Figur 28 - Gjenbruk av komponent for informasjon til brukeren. Komponentene henter tekstene sine fra Contentful.

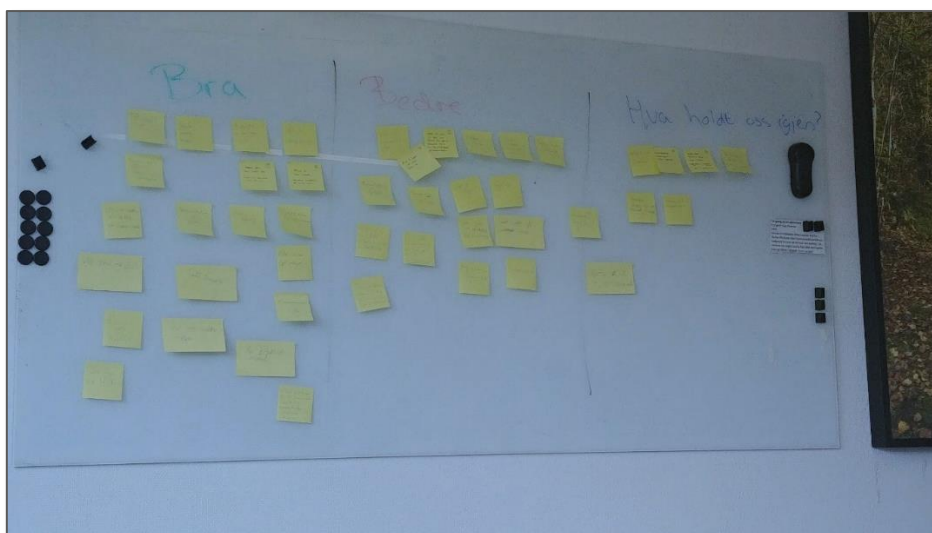
I backend ble det implementert tester for resterende logikk som ikke hadde enhetstester, med et mål om minimum en test per metode i API-et, samt tester for valideringen som skjer i *DTO*-ene. Under denne sprinten gikk også begge backendutviklerne over til å jobbe fullstack, men med prioritet i backend.

Det ble også satt av tid av frontend-utviklerne til å ta administrasjonsgrensesnittet gjennom en lang kvalitetskontroll med revideringer, før denne gikk inn i utviklergrenen.

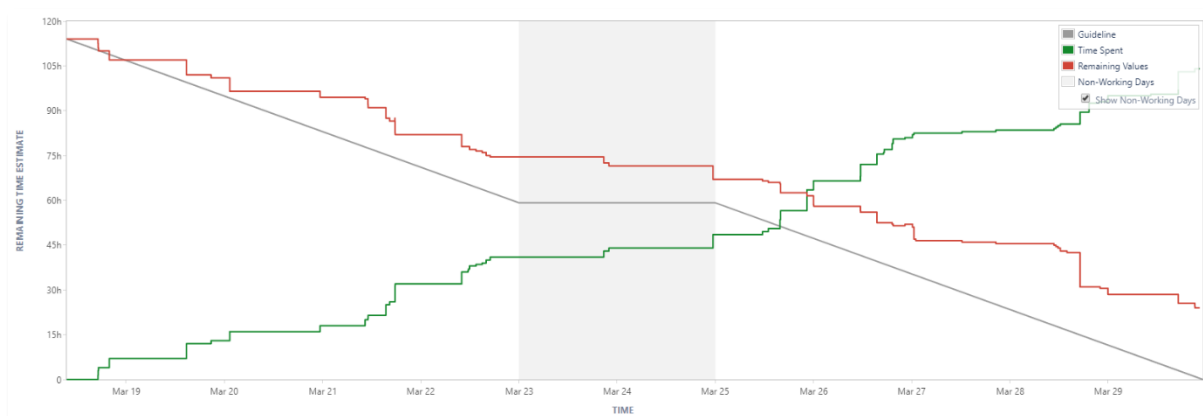
2.3.3.4.3 SPRINTRETROSPEKTIV

Det var vanskelig å sette av tid for alle gruppelemmene, samt inkludert teknisk støtte til å gjennomføre sprint retrospektiven, men vi fikk allikevel frigitt noe tid til dette, da vi så at det var svært viktig. Under retrospektiven ble det blant annet tatt opp at:

Det som var bra	Det vi kan bli bedre på	Hva holdt oss igjen?
<ul style="list-style-type: none"> • Kommunikasjon mellom frontend og backend. • Backend teamet har gått over til å jobbe mer fullstack. • God stemning i teamet. 	<ul style="list-style-type: none"> • Vi må bli flinkere til å føre timer. • Bli flinkere til å ta pauser. • Kan bli bedre på å dokumentere. 	<ul style="list-style-type: none"> • Scopet er for stort. • Har blitt en del kodegjeld. • Oppgaver det er behov for å få gjennomført som vi ikke har tatt høyde for under planleggingsfasen



Figur 29 - Tavlen vi brukte under Sprint Retrospektiv. Under retrospektiven skal man være kritisk, og vi hadde mange nyttige tanker om hva som var bra, hva vi kunne bli bedre på, og hva som eventuelt holdt oss igjen. Det skal sies at det var god stemning under møtet, og vi ser viktigheten av slike møter.



Figur 30 - Burndown chartet viser at vi under denne sprinten hele veien klarte å holde tritt med oppgavene. Dette gir inntrykk av at estimatene våre har vært gode, men vi har ikke brukt den estimerte tiden på dokumentering. Så i realiteten har vi jobbet flere timer, enn det chartet viser.

Det var også to brukerhistorier som ikke fullført. Disse ble overført til sprint 5 og fikk høyest prioritet når sprint 5 satt i gang. Disse var:

- Som en lærer ønsker jeg å ha mulighet til å fylle inn tekst og velge resultat slik at jeg kan konkludere på hypotesene og skrive en endelig konklusjon på problemstillingen for prosjektet.
- Som en lærer ønsker jeg å kunne endre data i utfylte felter slik at jeg har fleksibilitet i å endre data når jeg gjennomfører prosjektet.

2.3.3.5 SPRINT 5

Utspant seg over uke 14 og 15. Den siste sprinten før prosjektet skulle leveres til akseptansetesting. Fokuset var å håndtere bildeopplasting, bildevisning, vise det ferdige prosjektet og få ferdig ryddige komponenter for teksteditering.

2.3.3.5.1 PLANLEGGING AV FEMTE SPRINT

Ved inngangen av sprint fem, dro vi med oss to og en halv dag fra forrige sprint. Dette gjorde at vi måtte prioritere bort noen oppgaver. Dette var den siste sprinten med ny funksjonalitet, og fokuset var å sikte på å produsere en solid MVP, som presenterte en god løsning i forhold til prosjektets målsetning.

I denne sprinten var oppgavene bedre definert, med færre avhengigheter mellom brukerhistoriene. Dermed åpnet dette for at backendutviklerne i større grad kunne utføre frontend oppgaver.

Planlagte brukerhistorier for femte sprint:

- Som en lærer ønsker jeg mulighet for veiledning i hvordan hvert steg i Nysgjerrigpermetoden kan gjennomføres slik at jeg får planlagt og gjennomført Nysgjerrigpermetoden på en god måte
- Som en lærer ønsker jeg å ha mulighet til å fylle inn tekst og velge resultat slik at jeg kan konkludere på hypotesene og skrive en endelig konklusjon på problemstillingen for prosjektet.
- Som en lærer ønsker jeg å se et utkast av rapporten (prosjektoppsummering) slik at jeg kan tilpasse den om nødvendig
- Som en lærer ønsker jeg å kunne legge inn bilder underveis i prosessen slik at jeg kan bruke beskrivende bilder av elevenes arbeid i rapporten.
- Som en lærer ønsker jeg å kunne endre data i utfylte felter slik at jeg har fleksibilitet i å endre data når jeg gjennomfører prosjektet.
- Som en lærer ønsker jeg å bli bedt om at samtykke innhentes før opplasting av bilde slik at jeg kan bekrefte at samtykke er innhentet.
- Som en lærer ønsker jeg å ha mulighet til å slette opplastet innhold fra et påbegynt prosjekt.
- Som lærer ønsker jeg å kunne se oversikt over egne prosjekter slik at jeg har mulighet til å se egne prosjekter og dermed enklere å huske påbegynte prosjekter.
- Som en lærer ønsker jeg å kunne åpne eget eksisterende prosjekt slik at jeg kan ha flere prosjekter i gang samtidig.
- Som en lærer ønsker jeg å ha mulighet til å slette utfylt data slik at jeg kan fjerne data som ikke passer inn i prosjektet

2.3.3.5.2 GJENNOMFØRING AV FEMTE SPRINT

I begynnelsen av denne sprinten var det viktig for oppdragsgiver at applikasjonen ble publisert til skyen, fordi de hadde ønske om å utarbeide et produksjonstilbud til Forskningsrådet basert på produktet vi hadde utviklet. Dette førte til en del uforutsette forsinkelser i begynnelsen av sprinten, fordi det var viktig for oss at applikasjonen var

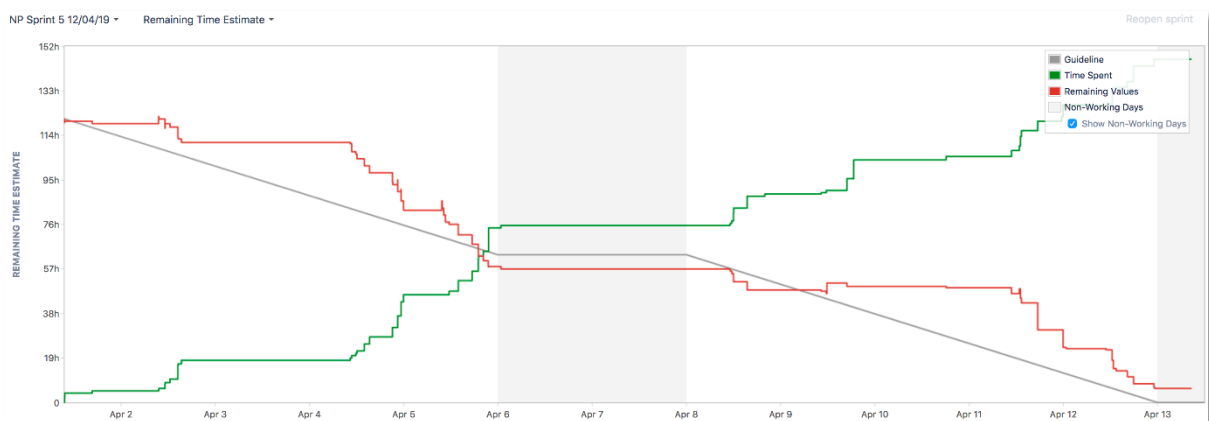
såpass godt utviklet at den kunne brukes. Derfor prioriterte vi å starte sprinten med å implementere utloggingsfunksjonalitet og en dropdown meny for pålogget bruker, samt fiksing av småfeil. Vi klarte likevel å hente oss greit inn igjen, selv om klargjøring til publisering ikke var regnet med under sprint planleggingen.

Prosjekteier hadde veldig tydelige krav om at de virkelig ønsket en solid bildefunksjonalitet. Denne jobbet vi hardt mer å få implementert, da det var en stor jobb både i frontend og backend å få denne til å fungere på en god måte. Med god kommunikasjon mellom frontend og backend, samt litt teknisk støtte fra oppdragsgiver kom vi fram til en løsning som fungerte godt og ikke lagrer bildeinnhold i Redux store, men henter dem når det er behov for dem.

Det ble også implementert funksjonalitet for henting og setting av aktivt prosjekt for bruker slik at en bruker kunne ha flere prosjekter samtidig og arkivere eldre prosjekter.

Under denne sprinten ble det også utført 3 grundige brukertester, noe som også preget progresjonen vår noe, da dette krevde at to av gruppemedlemmene måtte være med under gjennomførelsen av denne.

2.3.3.5.3 SPRINTRETROSPEKTIV



Figur 31 - Sprinten gikk veldig ryddig for seg. Vi kom oss jevnt og trutt gjennom oppgavene og det førte til at chartet ble ganske pent. Det var en liten brukerhistorie som vi dro ut av sprinten, da vi i samarbeid med prosjekteier, fant at den kunne nedprioriteres.

Vi var veldig fornøyd med siste sprinten. Vi rakk derimot ikke å gjennomføre en sprint retrospektiv. Vi hadde god kontroll på småfeil, og hadde en plan for hvordan disse kunne håndteres under akseptansetest. Vi skulle gjerne sett at vi fikk publisert en stabil versjon, men vi måtte håndtere enkelte småfeil i akseptanseperioden først.

Fra denne sprinten var det kun en brukerhistorie som vi ikke fikk tid til å utvikle, og dermed måtte utgå fra prosjektet.

- Som en lærer ønsker jeg å ha mulighet til å slette utfylt data slik at jeg kan fjerne data som ikke passer inn i prosjektet

Dette innebærer her at vi ikke rakk å utvikle en egen slettefunksjon for hypoteser og metoder, de kan dog endres.

2.3.4 AKSEPTANSETESTING OF FEILRETTING

Etter at femte sprint var fullført kunne vi begynne akseptansetesting og feilretting av applikasjonen. Det innebar at produkteieren bruker den utviklede applikasjonen og vurderer om den leverte løsningen innfrir akseptansekriteriene.

I forberedelse til akseptansetestingen lagde vi en oversikt av alle overordnede epics og underliggende brukerhistorier, og skrev veiledning til hvordan funksjonaliteten til enhver brukerhistorie kunne testes av kunde med testskript.

Henviser til [testrapporten](#) for mer om denne fasen.

2.3.5 FERDIGSTILLING

Produktet er klart for å settes i produksjon og er godt strukturert, slik at det burde være svært greit å videreutvikle det.

Det var noen brukerhistorier som ikke ble implementert i løpet av utviklingsperioden, disse vil implementeres ved produksjonssetting og kan leses om i [Samsvar mellom kravspesifikasjon og produkt](#).

Applikasjonen har blitt publisert i skyen og oppfylder de akseptansekriteriene som inngikk i sprintene. Det går raskt å publisere inkrementelle utvidelser til skyen. Beskrivelsen av det ferdigstilte produktet er gjennomgått i [produktokumentasjonen](#).

2.3.6 SAMMENDRAG

Sprintene hadde alle et svært høyt tempo, hvor vi jobbet hardt og strukturert for å rekke å fullføre alle brukerhistoriene. Gruppen klarte å overholde prinsippene om å avholde daglige møter. Dessverre var det heller unntaksvis at prosjektgruppen, prosjektleder og teknisk støtte klarte å sette av tid til sprint retrospektiv.

Det var også en utfordring å finne en balansegang mellom å ha en dedikert frontend og en dedikert backend gruppe. Dette medførte at det ble mer og mer fullstack fokus.

Siden prosjektet var så stort, var det utfordrende å sette av tid til alt som skulle gjøres. Det var mange oppgaver som måtte utføres som det ikke var estimert tid for. Dette resulterte at vi måtte ta enkelte brukerhistorier ut av scopet. Sluttsummen ble allikevel at vi leverte et arbeid som vi alle er svært stolte av.

2.4 SAMSVAR MELLOM KRAVSPESIFIKASJON OG PRODUKT

I henhold til [Kravspesifikasjon](#) så møtte prosjektet vårt 35 av de 39 [brukerhistoriene](#).

Applikasjonen har et administrasjonsgrensesnitt som kravspesifikasjonen etterspør. Her får administrator statistikk over applikasjonen, og oversikt og mulighet til å gjøre endringer på både brukere og prosjekter.

Prosjektet hadde som mål å bidra til at flere lærere tar i bruk Nysgjerrigpermetoden i undervisningen sin og kravspesifikasjonen gjenspeiler dette. Applikasjonen søker å nå dette målet ved å forenkle utførelsen av Nysgjerrigpermetoden. Kapitlet [Funksjonell systembeskrivelse](#) i produktdokumentasjonen, beskriver hvordan mange av funksjonene som kravspesifikasjonen etterspør er levert.

Brukerhistoriene vi ikke fikk tid til å utvikle var disse:

- NP-38 – Som en lærer ønsker jeg å ha mulighet til å slette utfylt data slik at jeg kan fjerne data som ikke passer inn i prosjektet
- NP-21 – Som en lærer ønsker jeg å endre min brukerinformasjon slik at jeg kan endre til riktige/oppdaterte opplysninger (bortsett fra e-postadresse)

Begge de overstående oppgavene er relativt «enkle», men likevel strakk ikke tiden til for å utføre disse.

- NP-79 – Som utvikler ønsker jeg at applikasjonen skal sende autogenerated e-post, slik at resetting av passord skal være brukervennlig

Dette var en oppgave som tidlig ble prioritert bort da vi estimerte at dette ville ta mye mer tid å innføre enn det vi hadde til rådighet.

- NP-14 – Som en besøkende ønsker jeg å generere nytt passord slik at jeg kan endre passord eller lage et nytt om jeg har glemt det jeg har.

Denne var avhengig av NP-79, og ble dermed også prioritert bort.

Det er flere grunner til at det ikke ble tid til å utvikle disse brukerhistoriene. Flere funksjonaliteter og teknologier som ikke var tatt høyde i kravspesifikasjonen for måtte implementeres. Noen av disse var utloggingsfunksjonalitet, implementasjon av lokal lagring i nettleser (altså et cookie verktøy) med lagring av autentisering i denne, og implementasjon av databasemigreringsverktøy. Vi hadde heller ikke estimert nok tid til å lære oss nye teknologier innledningsvis.

Vi kommuniserte tidlig til produkteier at det at noen brukerhistorier ikke ville bli implementert. Produkteier var med på å prioritere brukerhistoriene, og de brukerhistoriene som ikke ble implementert var dermed de med lavest prioritet.

Dermed ble det uproblematisk å legge disse til side og applikasjonen leverer allikevel det kravspesifikasjonen ber om.

2.5 POTENSIALE FOR VIDERE UTVIKLING

Vi har under arbeidet, både i konsept- og prosjekteringsfasen og under utviklingen, hatt ideer og sett nødvendige forbedringer. Av forskjellige grunner har det ikke vært innenfor prosjektets scope å implementere alle disse, men vi har tatt hensyn til de i utviklingen og da gjort oss noen tanker om hvordan de bør implementeres og prioriteres.

2.5.1 FORSLAG TIL UTVIDELSER BACKEND

- Testmiljø for JPA repositories
- Få bedre testdekning slik at man kan drive testdrevet publisering
- Implementere statisk analyseverktøy for å finne potensielle kodetrusler
- Forbedring av testene slik at de ikke låser seg når man kjører dem. Fører til at man ikke får pakket prosjektet til JAR uten å benytte obstruksjonen - DSkipTests.
- Oppsett av automatiske triggere for databasen. Med to hovedfokus.
 - Sletting av ikke-aktiverede brukerkontoer.
 - Oppdatering av datostempling for endringer av entiteter.

2.5.2 FORSLAG TIL UTVIDELSER FRONTEND

- Tilpasse applikasjonen for mobilbruk.
- Heve designkvaliteten på grensesnittene, med dedikerte designere.

2.5.2.1 PROSJEKTVEILEDEREN

- Eksportere dokumentet direkte som en valgt filtype.
- Sende et prosjekt fra applikasjonen, direkte til Nysgjerrigper-konkurransen.
- Utvidelse av bilde funksjonalitet hvor bilder kan settes til vilkårlige område i tekstfeltene.
- Mulighet for å redigere prosjektet direkte i prosjektvisningsvinduet.

2.5.2.2 ADMINISTRASJONGRENSESNIETTET:

- En gjennomgang etter mer brukertesting av bruk av administrasjonsgrensesnittet, som omhandler hva slags informasjon de ønsker skal vises i det.
- Ekstra statistikk, med tanke på hvor ofte et prosjekt oppdateres. Muligens uke til uke.
- Endring av statistikk vinduet til å vise konkurranseperioder fremfor hele år.

- Ta i bruk API-løsningen på hvilket år som skal vises.

2.5.3 FORSLAG TIL VIDERE UTVIDELSER FOR AWS SKYLØSNINGEN:

- Oppsett av en bakdør til RDS serveren i AWS. Slik at man i skyen kan administrere databasen direkte.
- Sette opp løsninger for skaleringen av applikasjonen ved flere samtidige brukere.

2.5.4 SIKKERHET

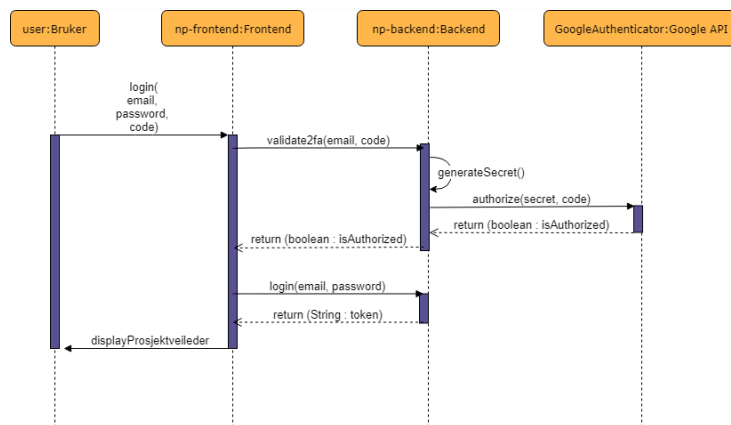
- Et SSL-sertifikat som sikrer trygg kommunikasjon (kryptert kommunikasjon) mellom klient og API, over HTTPS (Hypertext Transfer Protocol Secure)
- Det bør implementeres en «no-reply» e-post løsning for håndtering av autentisering av e-postadresser ved registrering av brukere, og for tilbakestilling av passord.
- Videreutvikling av tofaktor autentisering, for å sette opp kobling via sms.

2.5.4.1 ANBEFALT INKREMENT TIL GOOGLE AUTHENTICATOR-IMPLEMENTASJONEN

Vi lærte mye av å ta i bruk Google Authenticator API-et for brukerautentisering. Både i forhold til hva vi har gjort og hva vi har reflektert over for å ivareta best mulig sikkerhet. Løsningen vi endte opp med, har rom for forbedring som bør implementeres i senere utvikling.

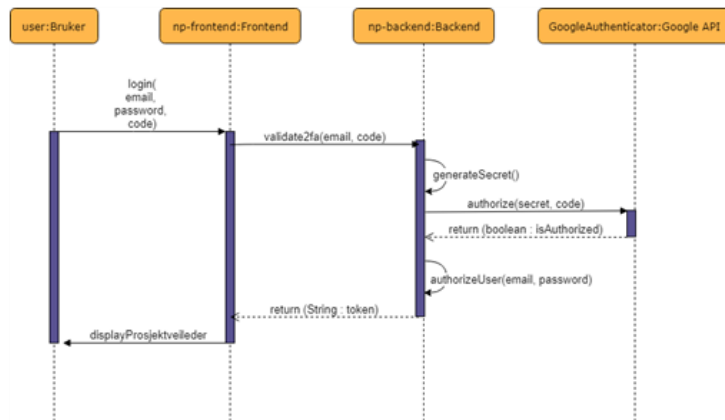
Forbedringen er relatert til en svakhet i implementasjonen som innebærer at man kan omgå tofaktor steget og logge inn ved å benytte API-endepunktet /login med kun brukernavn og passord, uten bruk av tofaktor.

Figuren nedenfor illustrerer den nåværende løsningen.



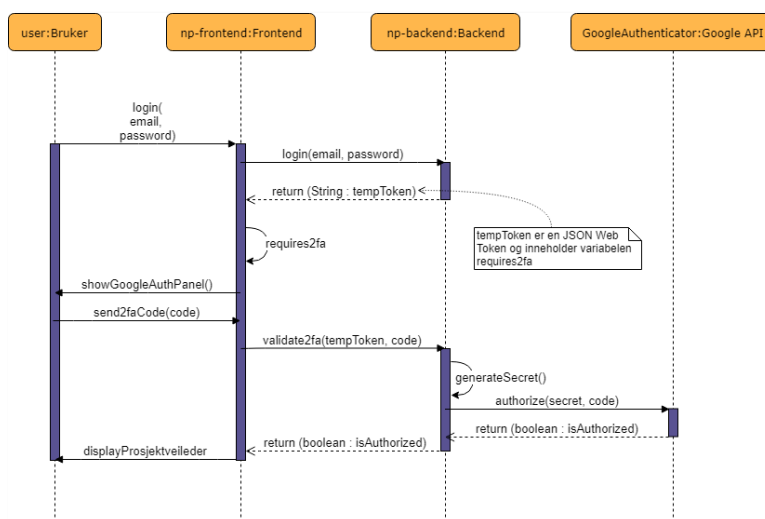
Figur 32 - Sekvensdiagram av nåværende implementasjon

Figur 33 viser et forslag til en enkel løsning for å unngå at API-endepunktet /login kan kalles. Dette innebærer at Google Authenticator-kallet må bakes inn i /login som er en del av Spring Security. Denne løsningen er estimert til å ta et halvt dagsverk å implementere.



Figur 33 - Sekvensdiagram av forslag til ny implementasjon

I tillegg til forslaget over har vi presentert en løsning til kunden der bruk av tofaktor-autentisering kunne være valgfri. Figur 34 viser arkitekturen til denne. Her unngår vi at API-endepunktet /login kan utnyttes ved å kun returnere en midlertidig autorisasjonstoken som deretter kan brukes til validering. Dersom en bruker har tofaktor skrudd på kommer klienten til å be brukeren om en Google Authenticator-kode, og backend vil kreve denne koden. Ellers blir kun det midlertidige tokenet sendt og forespørselen blir godkjent av backend uten at Google Authenticator API-et må kontaktes.



Figur 34 - Sekvensdiagram av forslag 2. Her blir en bruker med tofaktor-autentisering pålogget

2.6 UTFORDRINGER OG REFLEKSJONER

Prosjektet hadde mange og varierende utfordringer. Den første store utfordringen var at kunden bestilte kravspesifikasjonen med brukerhistorier, som vi da måtte produsere og tilpasse til kundens behov. Dette var en utfordrende oppgave som vi prioriterte i planleggingsfasen, og som vi ikke hadde erfaring med fra tidligere. Vi tok utgangspunkt i hovedmålet, å bidra til at flere lærere tar bruker Nysgjerrigpermetoden, og designkonseptet og satt oss godt inn i Nysgjerrigpermetoden. Deretter ga vi kunden et tilbud om en kravspesifikasjon og brukerhistorier utarbeidet utfra disse som vi så for oss at vi kunne levere, med forbehold om endringer.

For å kartlegge faren for endringer og andre risikomomenter, utarbeidet vi en risikorapport. Det store arbeidet med disse viktige dokumentene i den forberedende fasen, gjorde at det var begrenset med tid igjen før utviklingsstart for å planlegge selve utførelsen. Konsekvensen av dette ble at en del oppgaver vi hadde laget endte opp med å være svært avhengige av hverandre, slik at rekkefølgen på utførelsen av brukerhistoriene var lite fleksibel i begynnelsen.

Lærdommen ble at man alltid ønsker at man hadde mer tid, og i dette tilfellet gjaldt det for selve planleggingen av utviklingen. Vi tenker også at vi burde estimert oppgavene på en annen måte. For å få til dette burde vi i større grad fokusert på å få en totaloversikt, samt fokusert på å planlegge balansen mellom arbeidsbelastning i frontend og backend.

Vi kunne med fordel hatt to fullstack utviklere fra begynnelsen av. Dette blir beskrevet i sprint 3 og 4, der vi hadde for tydelige skiller mellom frontend og backend. Dette ga utfordringer knyttet til samarbeidet mellom gruppemedlemmene. Vi tok tak i dette, og vi opparbeidet gode rutiner for samarbeid som sikret tydelig kommunikasjon. For å bedre gruppens felles forståelse bestemte vi oss for å blant annet diskutere totalsituasjonen for prosjektet under de daglige sprintmøtene. Dette gjorde at alle gruppemedlemmene kontinuerlig var oppdatert på hva som skulle til for å komme i mål med de forskjellige oppgavene i løpet av sprinten.

Enkelte av oppgavene var teknisk utfordrende, og vi brukte mer tid enn estimert på å opparbeide oss tilstrekkelig kompetanse for å utføre dem. Dette var spesielt fremtredende under implementasjon av React-Redux, Spring Security og AWS (Amazon Web Services) og førte til at andre oppgaver måtte nedprioriteres under utviklingsperioden. Dette var heldigvis uproblematisk både for Netcompany og for Forskningsrådets del, da vi hele tiden hadde god kommunikasjon om statusen for prosjektet. Slik prioriterte vi bort enkelte brukerhistorier sammen med produkteier. Av dette erfarte vi styrken ved Scrum, nemlig modellens mulighet til å fullføre en leveranse med de fullførte brukerhistorier, og hvordan man kontinuerlig utfører forbedringer i iterasjoner, med mindre uavhengige inkremitter.

2.7 KONKLUSJON TIL PROSESSDOKUMENTASJON

Prosjektperioden har vært preget av høyt tempo og mye lærdom. Det var spesielt utfordrende at vi alle hadde som mål å lære flere nye teknologier som vi hadde lite kjennskap til, men dette var også det vi ble drevet av, da vi ønsket å lære så mye som mulig i løpet av prosjektperioden.

Sluttproduktet ble en stor applikasjon bygd opp med god arkitektur og et fokus på videreutvikling. Både kunden og oppdragsgiver er svært fornøyde med sluttresultatet, og det er vi i prosjektgruppen også, da applikasjonen er solid og adaptiv for videre utvikling.

Målet var at flere lærere skal ta i bruk Nysgjerrigpermetoden, og vi mener at applikasjonen treffer denne målsettingen svært bra. Vår tilnærming var å forsøkte å gjøre applikasjonen "leken" og intuitiv å bruke, og slik senke terskelen for at lærere skal bruke Nysgjerrigpermetoden. Tilbakemeldinger fra sluttbrukere tyder på at dette var en vellykket tilnærming.



Figur 35 - Målsettingen fra kunden i forkant av vår prosjektperiode.

Netcompany og Forskningsrådet er allerede i gang med prosessen rundt produksjonssetting av prosjektet. I denne produksjonssettingen vil brukerhistoriene vi utarbeidet men måtte nedprioritere, fullføres og flere funksjonaliteter legges til. Produksjonssetting vil sannsynligvis skje til allerede i løpet av høsten.

Prosjektet har vært av svært stor verdi for oss i prosjektgruppen. Både i form av økt faglig kompetanse, men også som erfaringer rundt hvordan det er å jobbe med et stort prosjekt i en profesjonell setting. Å bruke gjennomgående smidig utvikling var veldig lærerikt ettersom vi lærte oss å raskt produsere inkremitter, og samtidig differensiere disse oppgavene. Prosjektet har vært veldig givende å jobbe med og vi har blitt godt rustet til å ta fatt på utviklerrollen i arbeidslivet.

3 PRODUKTDOKUMENTASJON

NYSGJERRIGPER – PROSJEKTVEILEDEREN

BACHELORPROSJEKT 2019

GRUPPE 5

INNHALDSFORTEGNELSE PRODUKTDOKUMENTASJON

3.1	INNLEDNING.....	59
3.1.1	Introduksjon til programmet.....	59
3.2	FUNKSJONELL SYSTEMBESKRIVELSE	62
3.2.1	Aktører	62
3.2.2	Brukergrensesnitt.....	63
3.2.2.1	Landingsside.....	63
3.2.2.2	Registrering	64
3.2.2.3	Prosjektveileder	66
3.2.3	Administrasjonsgrensesnitt	74
3.2.4	Use-case diagrammer over funksjonalitet.....	77
3.3	PROGRAMMETS OPPBYGNING OG VIRKEMÅTE.....	79
3.3.1	Backend	80
3.3.1.1	Teknologier	80
3.3.1.2	Arkitektur.....	83
3.3.1.3	Forespørsler mot API-et	94
3.3.1.4	Roller.....	96
3.3.2	Frontend	98
3.3.2.1	Teknologier	98
3.3.2.2	Arkitektur.....	104
3.3.2.3	Implementasjon av valgte teknologier	106
3.3.3	Sikkerhet.....	112
3.3.3.1	Spring Security	113
3.3.4	Skyløsning	116
3.3.4.1	Arkitektur.....	116
3.3.4.2	Tjenester.....	117
3.4	PUBLISERING	119
3.4.1	Publisering av backend til Elastic Beanstalk instans.....	119
3.4.1.1	Publisering av API.....	119
3.4.1.2	Feilsøking av publisert API.....	121
3.4.2	Publisering av frontend til Simple Storage Service	121
3.4.2.1	Publisering klient	121
3.4.2.2	Feilsøking av publisert klient.....	121
3.5	KONKLUSJON TIL PRODUKTDOKUMENTASJON	122

3.1 INNLEDNING

Produktdokumentasjonen har som mål å beskrive det ferdige produktet som leveres til kunden. Her skal de tekniske aspektene og implementasjonene til applikasjonen beskrives.

Det anbefales at leseren har sett over kravspesifikasjonen og brukerhistoriene før dette dokumentet leses.

Hovedkapitler i dokumentet

- [Beskrivelse av programmet](#): Beskrivelse av nettapplikasjonens løsning på kravspesifikasjonen og måloppnåelsen fra [problemstillingen](#).
- [Funksjonell systembeskrivelse](#): Beskrivelse over funksjonalitetene til applikasjonen, med visuell presentasjon av bruker- og administrasjonsgrensesnitt.
- [Programmets oppbygning og virkemåte](#): Hvordan applikasjonen er satt sammen. API-et, databasen, klienten, og hvordan disse virker sammen, samt hvilke forskjellige teknologier som benyttes.
- [Konklusjon og drøfting av sluttprodukt](#): Drøfting av produktkvaliteten på Nysgjerrigper-Prosjektveilederen.

3.1.1 INTRODUKSJON TIL PROGRAMMET



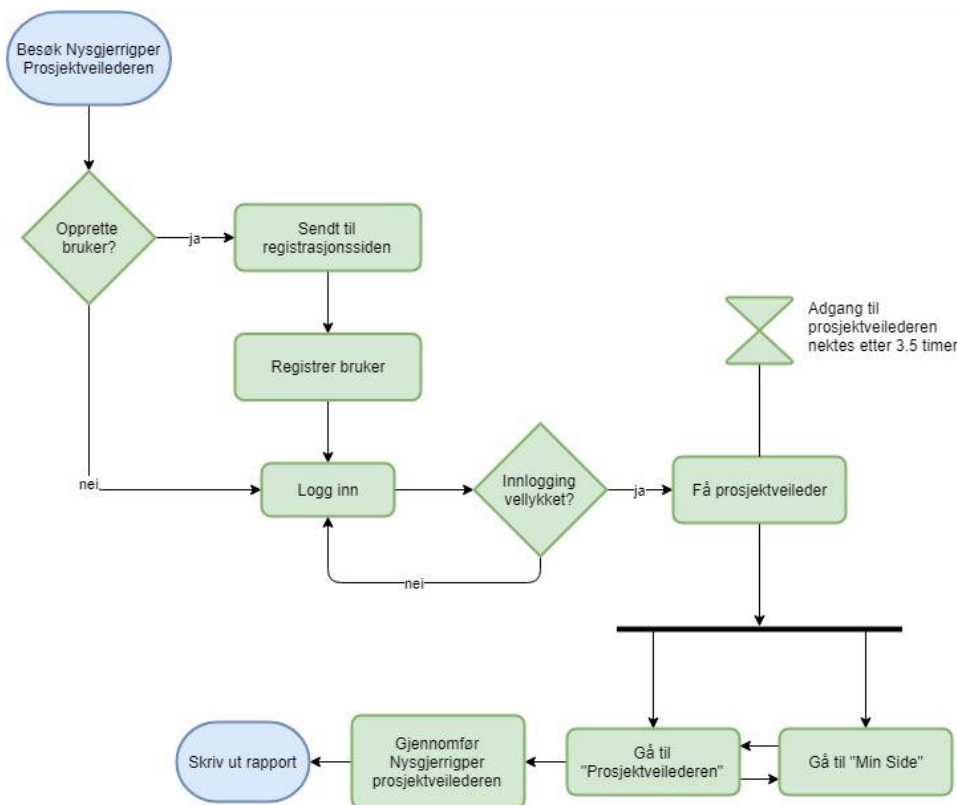
Figur 36

Kjerneprogrammet er et interaktivt verktøy som veileder lærere gjennom Nysgjerrigpermetoden utviklet av Norges Forskningsråd. Nysgjerrigpermetoden har seks primære steg som hver skal presentere et forskningskonsept og tillate elevgrupper å få praktisk erfaring med disse.

En bruker oppretter ett prosjekt med en problemstilling, som det kan legges hypoteser til, som igjen forsøker å svare på problemstillingen. Hver hypotese kan ha flere metoder som vil prøve å motbevise eller bevise hypotesen den er tilknyttet.

Innholdet i prosjektet lagres kontinuerlig, og kan til enhver fase i prosjektets gjennomføring, presenteres som et samlet rapportdokument som brukeren har mulighet til å laste ned.

Applikasjonen er ment for å kunne brukes i fellesskap i et klasserom, hvor læreren går gjennom veilederen med elevene. Programmet har en relativt dynamisk struktur, hvor det skjer en del interaktive reaksjoner ved utfylling av data. Dette innebærer at vi har jobbet for at elementene på siden er store for å kunne fungere bra på store skjermer.



Figur 37 – Flowchart av den generelle flyten i applikasjonen

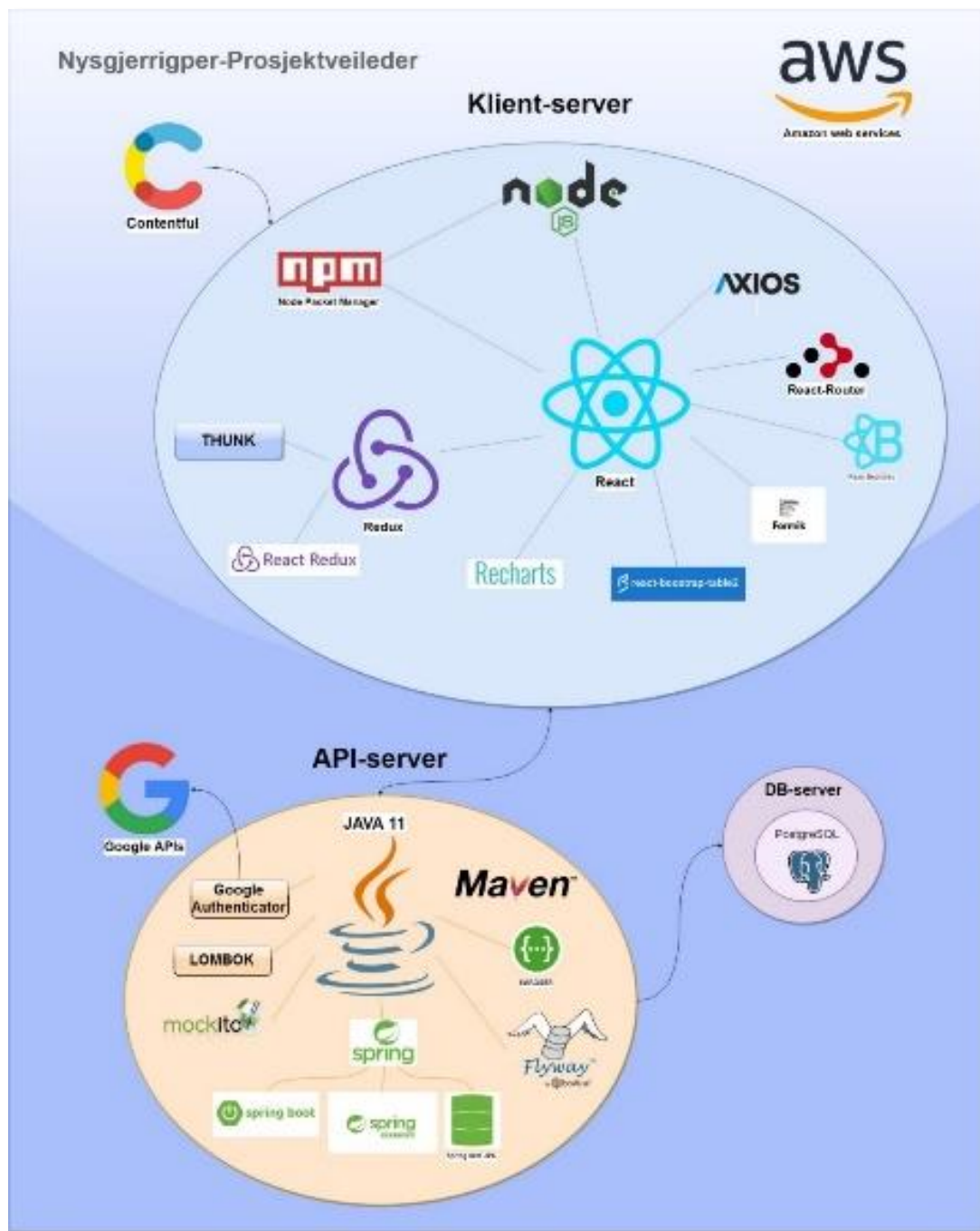
Figuren over viser en abstrahert forklaring av flyten i prosjektveileder-applikasjonen. Det finnes en mer detaljert flowchart til «Gjennomfør Nysgjerriger-Prosjektveilederen» senere i denne produktdokumentasjonen.

Brukerinformasjon						
BRUKERINFO		LÆRERINFO		EKSPORTER TIL CSV NULLSTILL SØK Søkefelt...		
Id	Fornavn	Ettennavn	Email	Rolle	Opprettet	
Id-1	Olav	Vestlie	olav@mliehigh.no	USER	02-03-2019	
Id-2	Yanis	Mussis	forever@young.com	USER	01-03-2019	
Id-3	Vegardo	Leivosa	livin@briskeby.no	USER	03-03-2019	
Id-4	Thomas	Toger	livets@hardeskoie.no	USER	05-03-2019	
Id-5	Dummy	Boy	user@user.com	USER	10-03-2019	
Id-6	Petter	Prosa	admin@admin.no	USER_ADMIN	11-03-2019	
Id-7	Guðrun	Gul	gul@westerdahlis.com	USER	10-03-2019	
Id-8	Fritjof	Frik	fri@net.no	USER	10-03-2019	
Id-9	Gunnar	Gunnis	mail@berprivatsivirgipen.no	USER	10-03-2019	
Id-10	Petter	Prosa	admin@nysgjerriger.no	USER_ADMIN	11-03-2019	

Antall rader per side: 10 25 30 50 Viser fra 1 til 10 av 22 resultat

Figur 38 - Administrasjonsgrensesnitt

Kjerne-applikasjonen har et administrasjonsgrensesnitt. Her har administratorbrukere tilgang til en rekke administrative funksjonaliteter. Dette inkluderer blant annet oversikt og kontroll over alle registrerte brukere og prosjekter som er opprettet i Nysgjerriger - Prosjektveilederen.



Figur 39 – Teknologioversikt over applikasjonen. Applikasjonen bruker mange teknologier og rammeverk. Disse er arbeidet mye med for at de skal integrere vellykket med hverandre.

Applikasjonen tar i bruk mange teknologier for å løse kravspesifikasjonen. Både frontend og backend implementer flere rammeverk. Frontend klientapplikasjonen er i hovedsak utviklet med JavaScript-biblioteket React, og backend API-et er utviklet i Java. Det er implementert flere eksterne rammeverk, blant annet for å ta i bruk API-ene til Contentful og Google APIs. Løsningen benytter en PostgreSQL database hvor alle data lagres. Applikasjonen er publisert i en dedikert skyløsning satt opp i Amazon Web Services.

3.2 FUNKSJONELL SYSTEMBESKRIVELSE

Det samlede systemet som sluttbrukeren benytter består av to overordnede komponenter, prosjektveilederen og administrasjonsgrensesnittet. Systembeskrivelsen er inndelt i de individuelle modulene ettersom dette speiler vår egen struktur.

Målet med den funksjonelle systembeskrivelsen er å bistå i videre utvikling av applikasjonen hos prosjekteieren, samt å demonstrere løsningene.

3.2.1 AKTØRER

Administrator

Administratoren kan endre brukerrettigheter, slette brukere og prosjekter. Administratorer har oversikt over bruken av applikasjonen.

Lærer/bruker

Registrert bruker i Nysgjerrigper - Prosjektveilederen, og kan opprette prosjekter. Gjennomgangen av hvert prosjekt kan gjøres i felleskap med elever.

Produkteier

Forskningsrådet, avdeling Nysgjerrigper. Bestemmer hvem som skal ha administratorrettigheter. Har tilgang til tjenesten Contentful, for å endre innhold på applikasjonen.

Besøkende

En besøkende kan innhente informasjon om Nysgjerrigper på landingssiden og kan opprette en bruker.

3.2.2 BRUKERGRENSESNITT

Brukergrensesnittet er den overordnede betingelsen for Nysgjerriger – Prosjektveilederen og er sammensatt av en landingsside, registrerings skjema, brukerinnstillinger og selve prosjektveileder-applikasjonen. Det er denne siden en vanlig bruker kommer til å få tilgang til og bruke for å fullføre Nysgjerrigermetoden.

3.2.2.1 LANDINGSSIDE



Figur 40 - Landingssiden til Nysgjerriger Prosjektveilederen

Landingssiden er den første siden besøkende havner på når nettsiden besøkes. Den gir den besøkende kort innføring i hva Nysgjerrigermetoden er, gir eksempler og lenker til tidligere utførte prosjekter og viser til diverse andre ressurser fra Forskningsrådet. Her kan besøkende manøvrere for å registrere en ny bruker eller logge på med eksisterende konto.

I henhold til EU-kommisjonens GDPR personvernerpolitikk vises en personvern-erklæring på landingssiden. Denne inneholder en lenke til Forskningsrådets egne ressurser om personvern, og husker om en bruker har gitt samtykke eller ikke. (GDPR, 2018)¹³



Figur 41 - Personvernerklæring som vises på landingssiden

¹³ General Data Protection Regulation (GDPR) (2018, October). <https://gdpr-info.eu/>

3.2.2.2 REGISTRERING

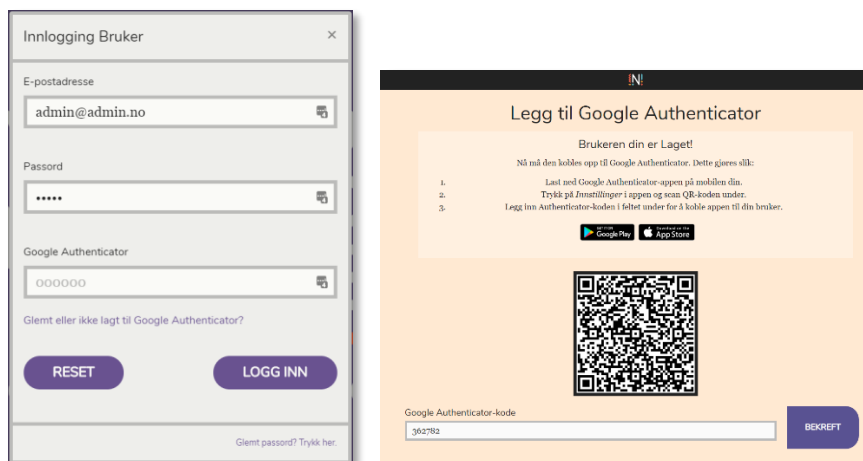
Alle besøkende på applikasjonen har mulighet til å opprette en bruker. Alle feltene i registrerings skjemaet må fylles ut. Det er flere krav, blant annet må epostadressen ha riktig syntaks, den kan heller ikke være registrert i systemet tidligere, og passordene må oppfylle minstekravene til produkteier. Når brukeren er registrert er det påkrevd at de setter opp tofaktor-autentisering gjennom Google Authenticator.

The image contains two screenshots of a web application interface. The left screenshot is titled 'Lag bruker' (Create user) and contains a form with the following fields: 'Fornavn' (First name), 'Etternavn' (Last name), 'E-postadresse' (Email address) with the value 'dahl@admin.no', 'Skole' (School), 'Passord' (Password), and 'Gjenta Passord' (Repeat password). A 'LAG BRUKER' button is at the bottom. The right screenshot is titled 'Legg til Google Authenticator' (Add Google Authenticator) and shows instructions in Norwegian: 'Brukeren din er Lagett!' (Your user is created!), 'Nå må den kobles opp til Google Authenticator. Dette gjøres slik:' (Now it must be connected to Google Authenticator. This is done like this:), followed by a numbered list: 1. Last ned Google Authenticator-appen på mobilen din. (Download the Google Authenticator app to your mobile phone.) 2. Trykk på Innstillinger i appen og scan QR-koden under. (Tap Settings in the app and scan the QR code below.) 3. Legg inn Authenticator-koden i feltet under for å koble appen til din bruker. (Enter the Authenticator code in the field below to connect the app to your user.) Below the instructions are icons for Google Play and the App Store, a QR code, and a text input field for the 'Google Authenticator-kode' (Google Authenticator code) with the value '362782'. A 'BEKREFT' (Confirm) button is at the bottom right.

Figur 42 - Brukeren skanner QR koden med bruk av mobilapplikasjonen Google Authenticator. Fyller deretter inn koden for brukerens epost konto.

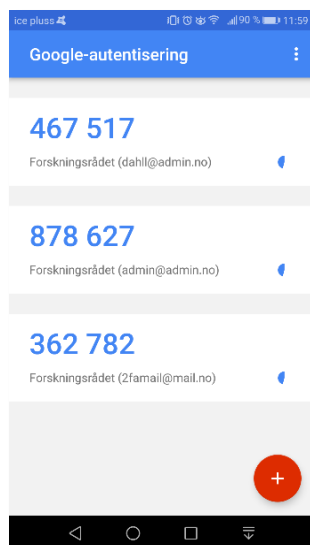
3.2.2.2.1 AUTENTISERING/INNLOGGING

Applikasjonen benytter Google sin tofaktor-løsning Google Authenticator for å autentisere brukeren sikkert ved innlogging. Brukeren er derfor avhengig av å benytte koder generert av Google Authenticator-mobilapplikasjonen for å autentisere seg og få tilgang til prosjektveilederen.



Figur 43 - Brukeren skanner QR koden med mobilens, Google-autentiserings applikasjon. Fyller deretter inn koden for brukerens epost konto.

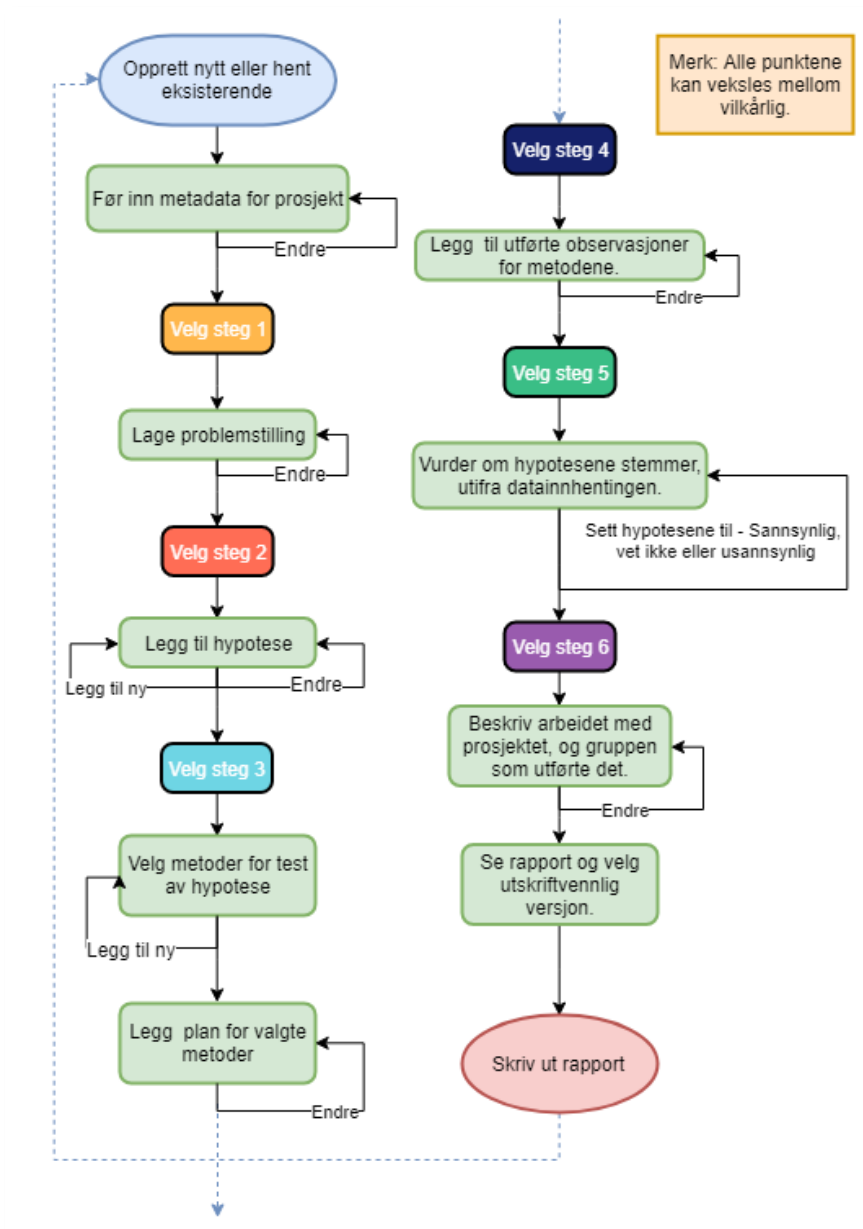
Brukeren kan sette opp Google-autentiseringen på nytt om det skal være nødvendig. I og med at brukeren foreløpig kan sette opp ny QR-kode når brukeren kan sitt eget passord, trenger denne løsningen en videre utvidelse hvor muligheten til å endre QR-kode enten bekreftes via SMS, eventuelt epost. Denne utvidelsen er planlagt for senere utvikling.



Figur 44 - Kodene bytter etter et tids intervall. De er kun synlige på brukerens registrerte mobil.

3.2.2.3 PROSJEKTVEILEDER

Applikasjonen har en prosjektveileder for gjennomføring av et forskningsprosjekt med Nysgjerrigper-metoden. Dette er en stegvis guide, hvor læreren til slutt får satt sammen en prosjektrapport, som skal kunne sendes inn til Nysgjerrigper. Hvert av disse stegene har en rekke komponenter for utfylling og endring av informasjon, samt visning av utfylt informasjon til brukeren.



Figur 45 - Flowchart for bruken av prosjektveilederen

3.2.2.3.1 TIPS OG VEILEDNING

Hvert steg har tips som veileder brukerne i hvordan steget skal gjennomføres. Disse tipsene kan endres på av produkteier via Contentful. På en rekke steder i applikasjonen er det nedtrekksfaner som kan vise disse tipsene til brukeren.



Figur 46 - Element for tips til lærere

3.2.2.3.2 INNTASTINGSFELT

Stegene i applikasjonen har flere inntastingsfelt. Disse fylles med tekst som lagres i databasen. Feltene har valideringer i forhold til hvilke tegn som tillates og lengden på tekstene. Hvert tekstfelt er koblet til sin egen del av prosjektet. Enkelte av feltene har automatisk utfylling av standardtekst, som at prosjektleder i utgangspunktet er lærerens eget navn. Alle inntastingsfeltene er knyttet til skjemaer, som er bygget med [Formik](#) biblioteket. Validering av feltene i klienten er håndtert i Formik-implementasjonen ved bruk av [Yup](#). Feltene valideres også av API-et, og en feilmelding vises til brukeren ved mislykket validering.



Figur 47 - Element for inntastingsfelt

3.2.2.3.3 BILDEOPPLASTNING

Applikasjonen tillater å laste opp bilder i flere av stegene i prosjektveilederen. Brukeren kan bare laste opp et bilde etter at han/hun har huket av at samtykke for publisering er innhentet. En knapp kommer frem i komponentene etter at samtykkebekreftelsen har blitt huket av.



Figur 48 - Element for bildeopplastning

Etter at bildets filtype og størrelse har blitt godkjent og bildet er lastet opp, vises en forhåndsvisning av bildet. Hvor bildet kan slettes om ønskelig. Ved ugyldig bilde vises det en feilmelding til brukeren.



Figur 49 – Miniaturbilde etter at en bruker har lastet opp et bilde

Bildene som er lastet opp blir vist på prosjektvisnings-siden, sammen med de elementene de er tilknyttet. Bildene lagres i databasen, som en Base64 enkodet tekst streng. Bildets størrelse er satt til å kunne være på maksimalt 10 MB, ved at maksimal størrelsen på forespørsler til API-et er begrenset.

3.2.2.3.4 NAVIGASJONSPANEL



Figur 50

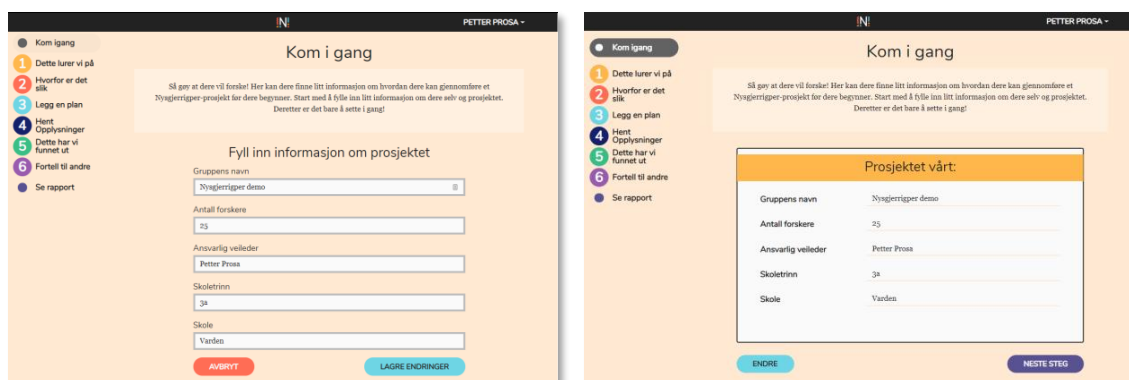
Navigasjonspanelet lar brukeren navigere mellom de forskjellige stegene i prosjektveilederen. Brukeren har alltid tilgang til dette panelet i prosjektveilederen, og kan navigere mellom stegene når enn det måtte ønskes.

3.2.2.3.5 STEG I PROSJEKTVEILEDEREN

Alle stegene i prosjektveilederen inneholder tips til hvordan steget kan utføres, samt et utvidbart panel med mer informasjon.

3.2.2.3.5.1 FORBEREDELSE - KOM I GANG

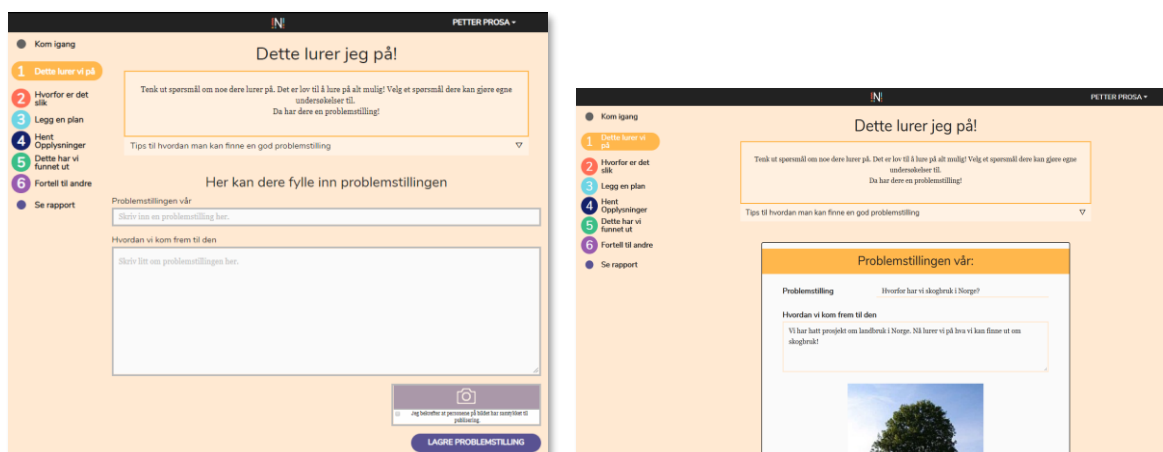
Samler inn grunnleggende informasjon om gruppens oppbygning. Denne informasjonen vises da til brukeren, og kan endres etter ønske.



Figur 51

3.2.2.3.5.2 STEG 1 – DETTE LURER VI PÅ

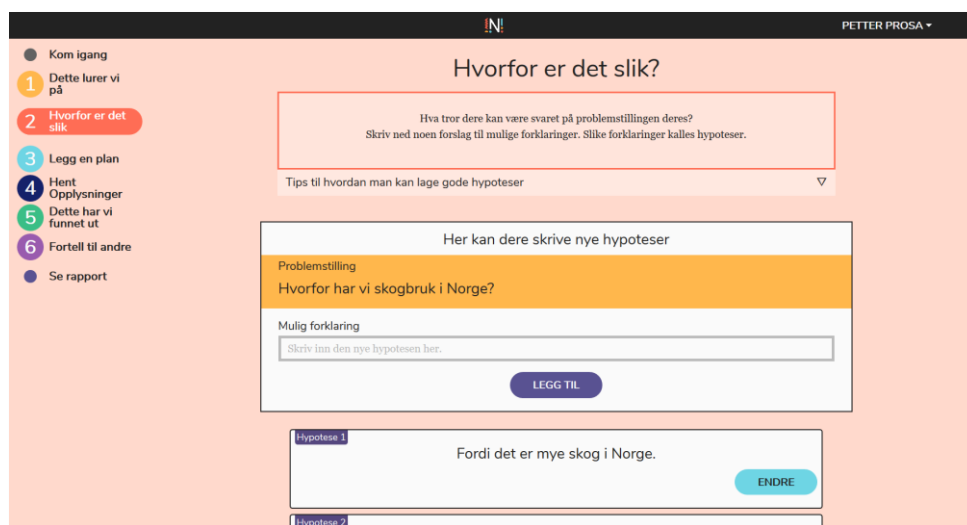
Gir brukeren muligheten til å legge inn problemstilling og beskrivelse av problemstillingen som da blir temaet for prosjektet. Et bilde kan legges til, slettes og vises



Figur 52

3.2.2.3.5.3 STEG 2 – HVORFOR ER DET SLIK

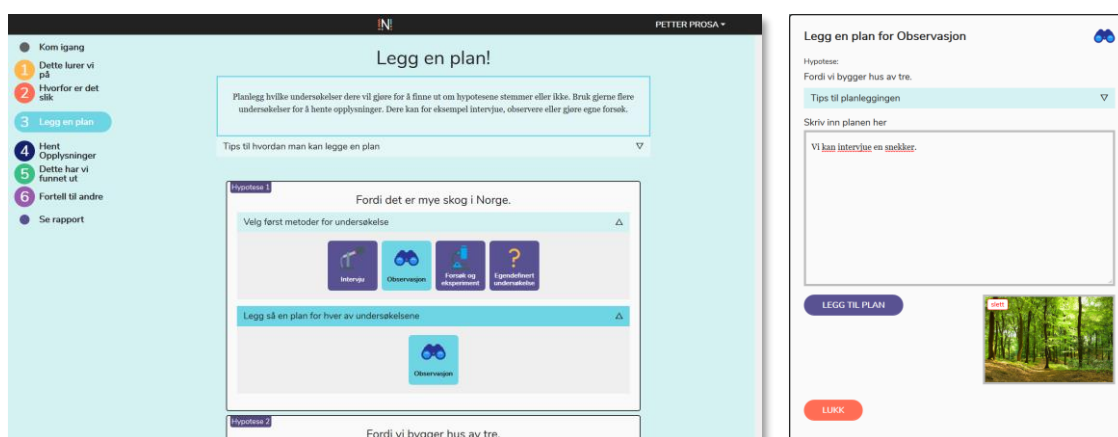
Her kan brukeren legge til hypoteser til den overordnede problemstillingen. Disse hypotesene kan redigeres av brukeren etter ønske.



Figur 53

3.2.2.3.5.4 STEG 3 – LEGG EN PLAN

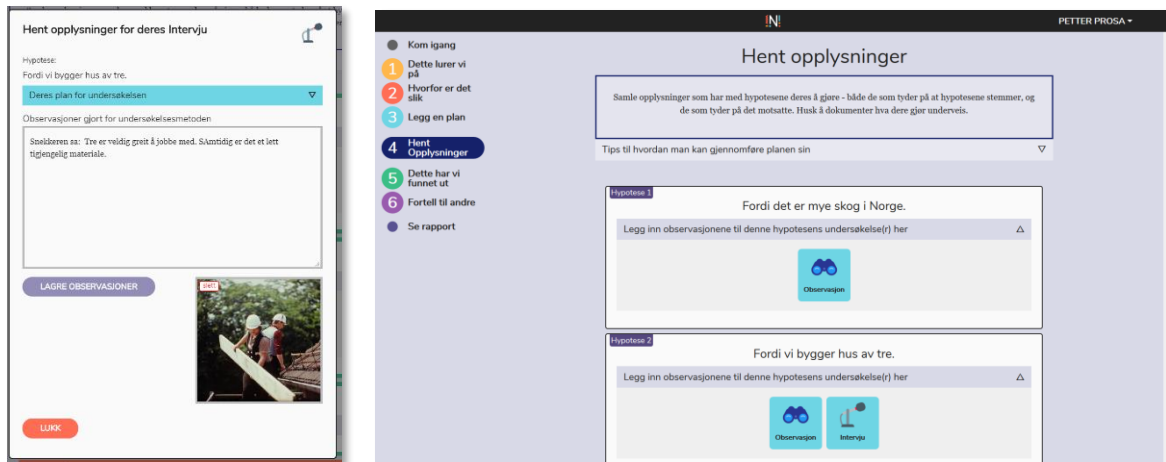
Viser alle hypoteser og tillater at brukeren legger til undersøkelsesmetoder for hver hypotese. Brukeren kan legge en plan for hvordan metoden skal utføres, samt endre denne planen om ønskelig. Hver enkelt metode inneholder tips til hvordan den kan utføres, det tillates at ett bilde lastes opp i planen.



Figur 54

3.2.2.3.5.5 STEG 4 – HENT OPPLYSNINGER

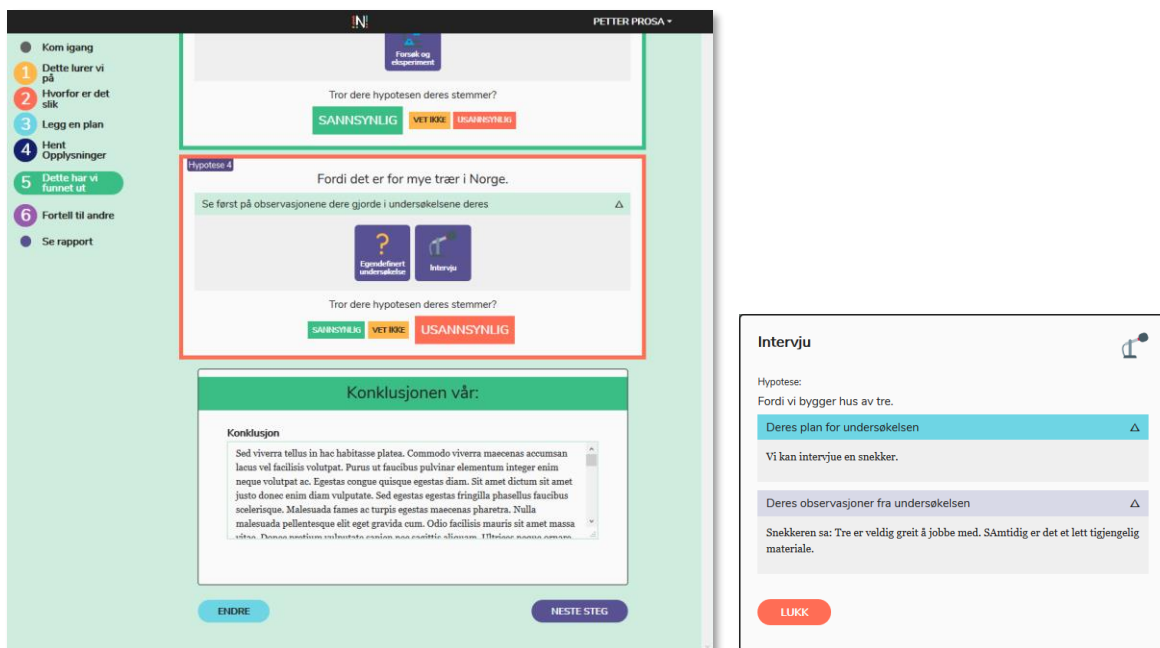
Viser alle hypoteser og tillater at brukeren legger til informasjon på hver enkelt metode. Inneholder tips til utførelse og mulighet til å laste opp ett bilde til hver metode, slik som under steg 3.



Figur 55

3.2.2.3.5.6 STEG 5 – DETTE HAR VI FUNNET UT

Viser alle hypoteser og lar brukeren velge hvor sannsynlig det er at hypotesen er et svar på problemstillingen basert på opplysningene hentet inn. Brukeren kan legge til en avsluttende konklusjon på prosjektet, og laste opp et bilde.

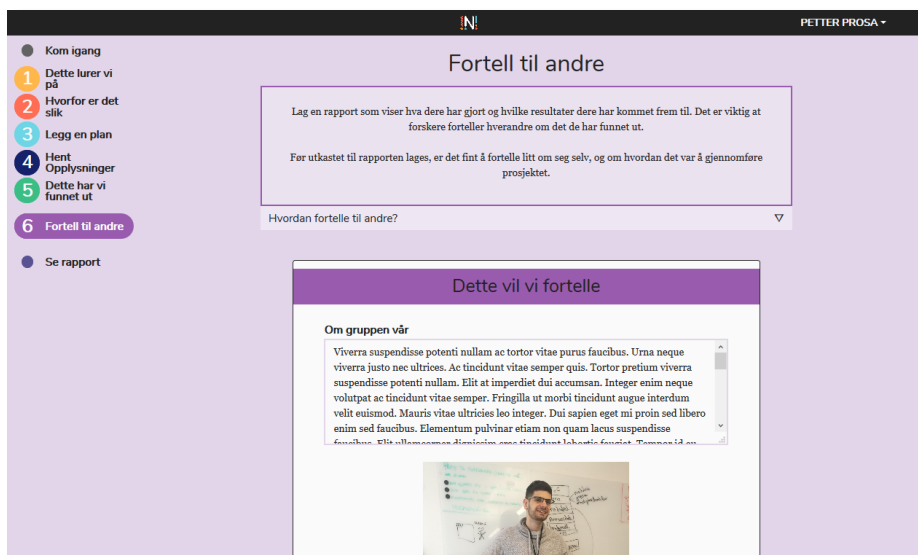


Figur 56

Om et prosjekt er sannsynlig får hypoteseboksen et grønt omriss, usannsynlig gjør omrisset rødt, og usikkert gjør det gult.

3.2.2.3.5.7 STEG 6 – FORTELL TIL ANDRE

Tillater at brukeren kan legge til en beskrivelse av både gruppen og prosjektet, samt legge til ett bilde til hver av disse feltene.



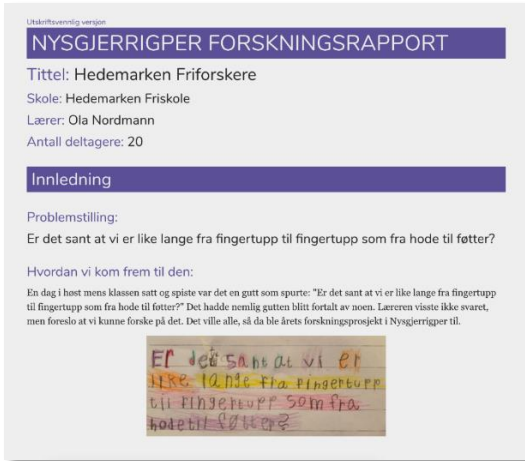
Figur 57

3.2.2.3.5.8 AVSLUTNING - SE RAPPORT

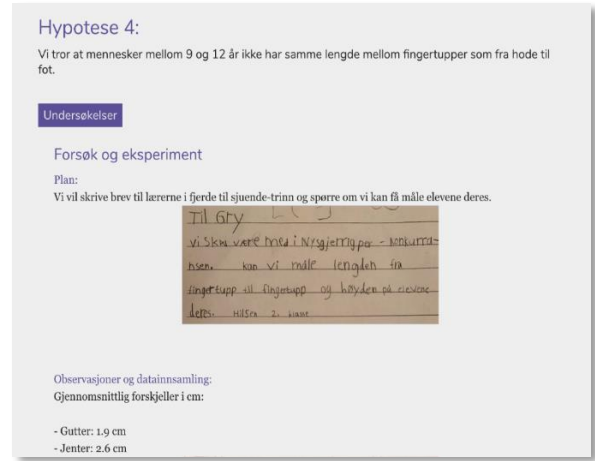
Etterhvert som *modellen* for prosjektet fylles ut med data, hypoteser og metoder, så blir denne informasjonen illustrerbar og kan sees ved at brukeren trykker på *Se rapport*. Brukeren blir sendt til et nytt vindu som viser hele prosjektet med all dataen som er lastet opp til et prosjektdokument. Denne har en lenke til en utskriftsvennlig versjon av dokumentet, som kan lagres på datamaskinen som en PDF-fil.



Figur 58



Figur 59

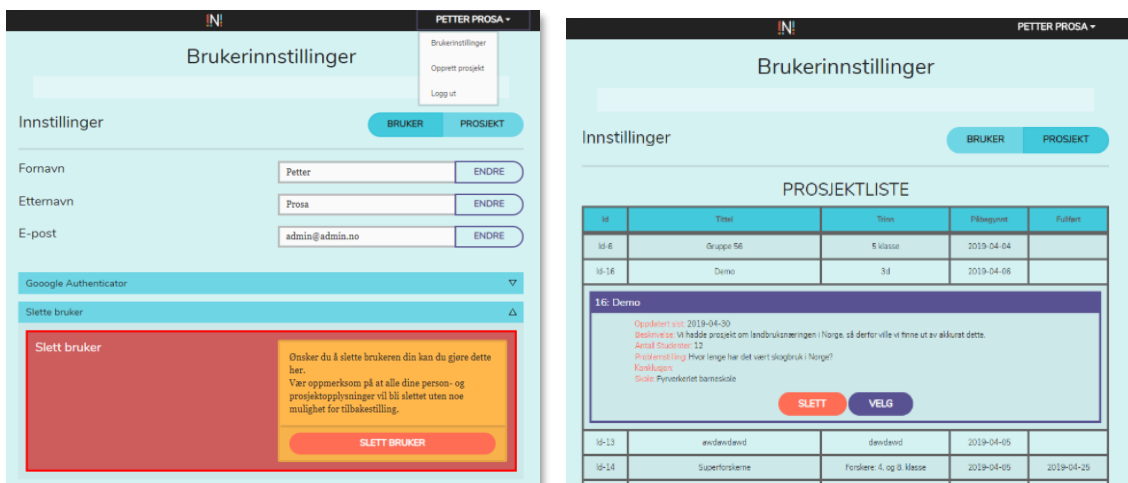


Figur 60

3.2.2.3.6 BRUKERINNSTILLINGER

Brukerinnstillingene er til enhver tid tilgjengelige via et panel øverst på skjermen i applikasjonen ved å trykke på sitt navn, og deretter på «Brukerinnstillinger». Her får brukeren tilgang til to store sett med funksjoner; bruker- og prosjektinnstillinger.

Brukerinnstillinger inneholder brukerinformasjon slik som brukernavn, e-post og brukersletting, mens prosjektinnstillinger har en prosjektliste, hvor brukeren får oversikt over sine prosjekter, kan slette disse eller bestemme hvilket prosjekt som er «aktiv» og lastet inn i prosjektveilederen. Per i dag kan ikke brukeren endre navn, etternavn eller epost, da frontend ikke knyttet til API-et for å endre disse. Dette er dokumentert hos kunde, og klar for implementasjon ved senere utvikling.

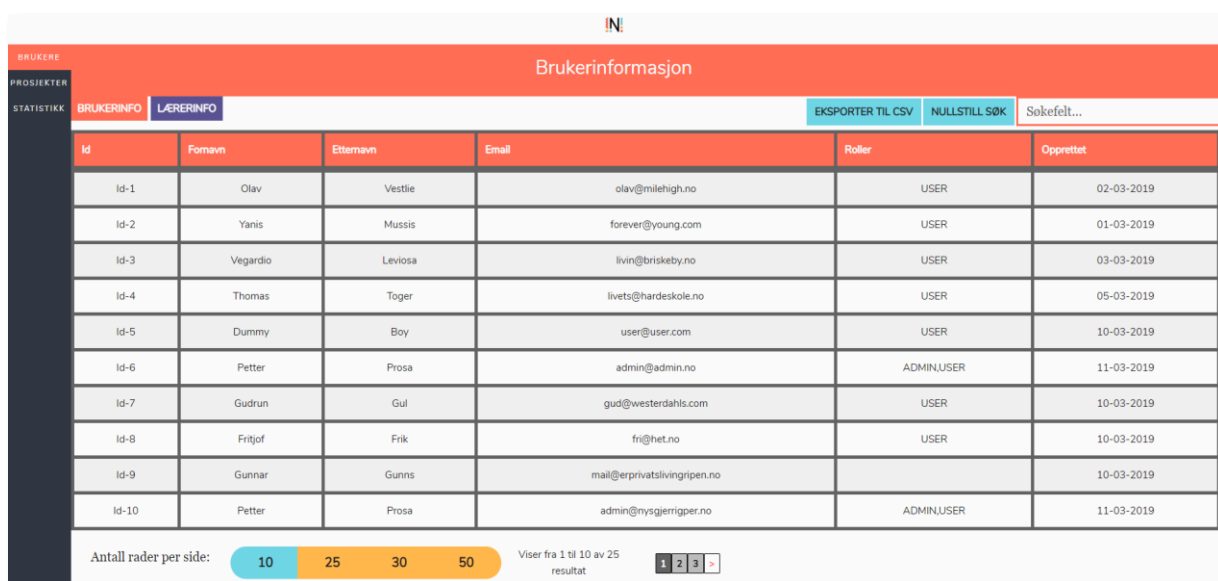


Figur 61

3.2.3 ADMINISTRASJONGRENSESNI TT

Brukere med administrator-rettigheter har tilgang til applikasjonens administrasjonsgrensesnitt. Den kan da nås med adressen <http://np-frontend.s3-website.eu-central-1.amazonaws.com/admin> etter innlogging. Om man ikke har rettighetene blir brukeren sendt til applikasjons-fremsiden.

Den første siden man kommer til når man går inn i administrasjonsgrensesnittet er brukersiden, denne siden viser en tabell med brukerinformasjon, og man kan bytte fane til lærerinformasjon. Disse tabellene samt prosjekttabellene, kan interageres med på mange måter.

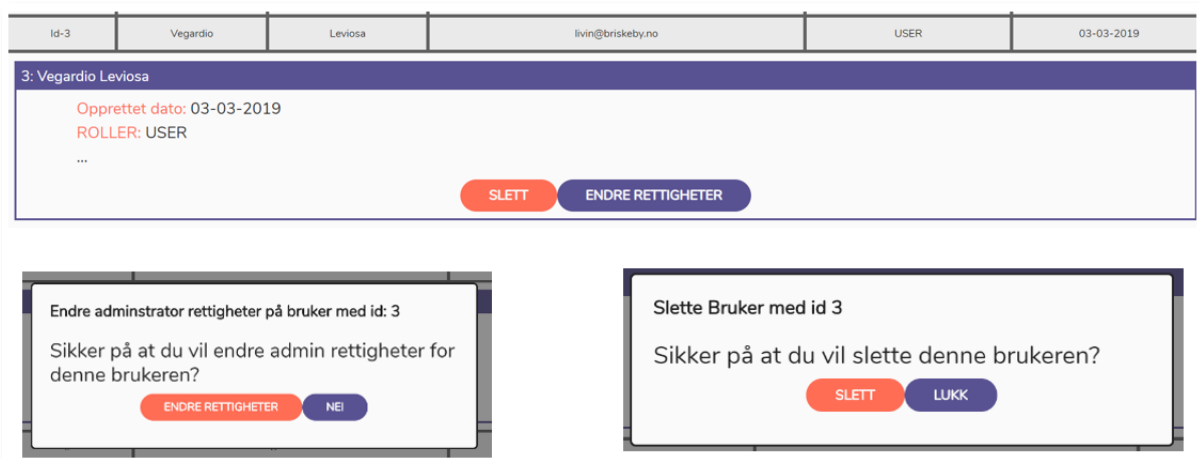


Id	Fornavn	Etternavn	Email	Roller	Opprettet
Id-1	Olav	Vestlie	olav@milehigh.no	USER	02-03-2019
Id-2	Yanis	Mussis	forever@young.com	USER	01-03-2019
Id-3	Vegardio	Leviosa	livin@briskeby.no	USER	03-03-2019
Id-4	Thomas	Toger	livets@hardeskole.no	USER	05-03-2019
Id-5	Dummy	Boy	user@user.com	USER	10-03-2019
Id-6	Petter	Prosa	admin@admin.no	ADMIN.USER	11-03-2019
Id-7	Gudrun	Gul	gud@westerdahls.com	USER	10-03-2019
Id-8	Fritjof	Frik	fri@het.no	USER	10-03-2019
Id-9	Gunnar	Gunns	mail@erprivatslivingripen.no		10-03-2019
Id-10	Petter	Prosa	admin@nysgjerrigper.no	ADMIN.USER	11-03-2019

Figur 62 - Landingsiden i administrasjonsgrensesnittet. Viser Brukerinformasjon.

Tabellfunksjoner

- Brukeren kan sortere ved å trykke på kolonne fanene.
- Kan velge hvor mange oppføringer som skal vises.
- Kan bla mellom sider.
- Kan trykke på en rad for å åpne et *nedtrekksvindu* med mer informasjon, og gir enkelte redigeringsmuligheter for data.
- I søkefeltet kan man søke i tabellene, og filtrere tabellene. Dette søket er da ikke kolonne spesifikt, derfor trenger Id-ene en tag, for at man skal kunne søke opp de. Eks: Å søke på 19, vil gi alle radene, da alle er opprettet i året 2019. Mens Id-19, vil filtrere på brukerId-en.
- Hurtigtast for å nullstille søk.
- Eksportering av hele tabellen til CSV filer, som kan importeres i iblant annet Excel.



Figur 63 – Nedtrekksvindu med slett og endre funksjoner

Bruker tabellens nedtrekks vindu, gir administrator mulighet til å gi eller ta vekk administratorrettigheter for den valgte brukeren, og brukeren kan slettes. Administratoren kan ikke slette seg selv eller endre egne rettigheter.

The screenshot shows a table titled 'Lærertabell' with columns: Id, Fødsel, Etternavn, Fornavn, Skole, Vålerenga, Email, and Opprettet. A dropdown menu is open for user ID 5, 'Dummy Boy', showing 'Opprettet dato: 10-03-2019', 'Antall prosjekter: 4', and 'Antall fullførte prosjekter: 0'. A 'PROSJEKTER' button is visible below the menu.

Figur 64 – Lærertabell

Lærertabellen gir lærerinformasjon om brukerne. Hvor mange prosjekter de har og hvor mange av dem som er fullførte.

Lærerinformasjons-nedtrekksvinduet gir mulighet til å gå direkte til prosjektene til den valgte læreren. Da sendes administrator til adressen `/admin/projects/L-[bruker-id]`, hvor det filtreres på L-[bruker-id], som er id-en til læreren som eier prosjektet.

The screenshot shows a table titled 'Prosjekter' with columns: Id, Tittel, Antall Studenter, Skole, Tittel, L-id, Påbegynnet, and Fullført. The table is filtered by user ID 5. The data rows are:

Id	Tittel	Antall Studenter	Skole	Tittel	L-id	Påbegynnet	Fullført
10-12	Explorer	5	Bjervollta skole	8	L-5	2019-04-05	
10-11	Nybegynner demo	1	Vålerenga	4c	L-5	2019-04-04	
10-30	Trudes test	6	Petsund		L-5	2019-04-24	
10-22	en gruppe id	25	skole	4	L-5	2019-04-25	

At the bottom of the table, there are pagination controls: 'Antall rader per side: 10 25 30 50' and 'Viser fra 1 til 4 av 4 resultat'.

Figur 65 - Prosjekter sortert etter bruker med id 5

Brukere							
Prosjekter							
STATISTIKK							
EKSPORTER TIL CSV							
NULLSTILL SØK							
Søkefelt...							
Id	Tittel	Antall Studenter	Skole	Trinn	L-Id	Påbegynt	Fullført
Id-2	Pname	23		2B	L-7	2019-03-13	
Id-4	Pname	11		9A	L-9	2019-03-09	
Id-16	Demo	12	Fyrverkeriet barneskole	3d	L-6	2019-04-06	
Id-3	Pname	43		1D	L-8	2019-03-12	
Id-12	Explorer	5	Bjørnsletta skole	8	L-5	2019-04-05	
Id-14	Superforskerne	23	Sem skole	Forskere: 4. og 8. klasse	L-6	2019-04-05	2019-04-25

14: Superforskerne

Navn på lærer: Petter Prosa Id: 6
 Oppdatert sist: 2019-04-25
 Beskrivelse: Først ble vi delt i grupper. Så fikk vi mange gule lapper til hver gruppe. Så gikk vi rundt ute og inne og skrev ned alle spørsmålene vi kom på. Læreren valgte ut noen av spørsmålene. Så stemte vi på det vi hadde mest lyst å forske på. Så skulle vi skrive hypoteser til spørsmålet vårt. Hypoteser er forslag til det vi tror er svaret.
 Problemstilling: Hvorfor kaster folk søppel i naturen?
 Konklusjon: fdsafsgdfshgdsfh

SLETT

Figur 66 - Prosjekt-tabell med et åpent nedtrekksvindu

Prosjekttabellen inneholder grunnleggende informasjon over alle prosjekter, som tittelen til prosjektene og når disse ble påbegynt/ferdigstilt. Nedtrekksvinduet gir mer informasjon om prosjektet som er valgt, og administratoren kan slette dette prosjektet og gå direkte til eieren av prosjektet som vises i brukertabellen.

Selve innholdet i disse tabellene er tilpasset for å kunne endres på i videre produksjon, etter behovene eierne og administratorene opplever at de har.



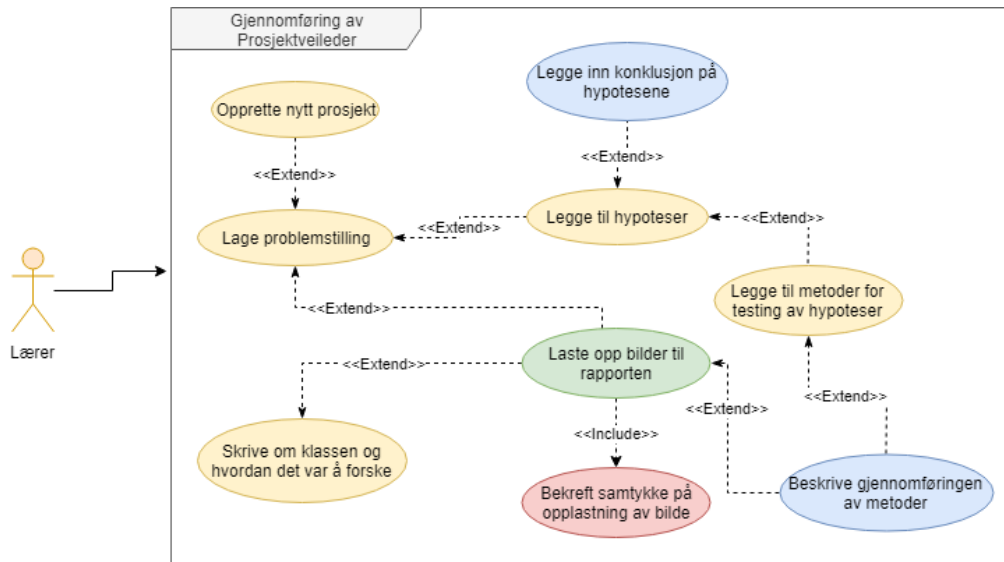
Figur 67 - Statistikk vindu på administrasjon grensesnittet. Med grafer og tallmateriale.

Administratoren kan se statistikk over antall brukere av applikasjonen og hvor mange som blir opprettet per måned. Administrator kan se hvor mange prosjekter som blir påbegynt og ferdige per måned. Dette for å gi administratorene og produkteierne mer oversikt over hvordan applikasjonen brukes, og da særlig i selve konkurranseperiodene. At et prosjekt blir satt til ferdig, trigges ved at feltene konklusjon, om prosjektet og om deltagerne har innhold.

3.2.4 USE-CASE DIAGRAMMER OVER FUNSKJONALITET

Disse brukerdiagrammene tar utgangspunkt i brukerhistoriene for prosjekt, og er alle funksjonaliteter applikasjonen oppfyller.





Figur 68 - Fargekodet Use-Case diagram for «Gjennomføring av Prosjektveileder»-funksjonaliteten
 Fargekoder: **Gul** betyr en standard brukerhistorie. **Blå** betyr en spesifikk utvidelse til en brukerhistorie. **Grønn** betyr en generell brukerhistorie flere kan implementere. **Rød** betyr en essensiell brukerhistorie for å fullføre en annen.
 Extend: Utvidelser til brukerhistorier.
 Include: Påkrevde brukerhistorier for å utføre en brukerhistorie.

3.3 PROGRAMMETS OPPBYGNING OG VIRKEMÅTE

Her vil prosjektets grunnleggende struktur beskrives. Applikasjonen er tydelig delt i et API og en web-klient.

API-et tilbys av en Java-applikasjon hvor Spring Framework benyttes som grunnleggende rammeverk, og [PostgreSQL](#) benyttes som databasesystem. Klienten er en [React](#)-applikasjon som bruker [Redux](#) med [Redux-Thunk](#) for all datahåndtering i web-klienten.

Begge løsningene er publisert i en skyløsning opprettet i Amazon Web Services. Her blir klienten og API-et publisert inkrementelt. API-et kjører kontinuerlig i skyen, mens klienten kjøres lokalt på sluttbrukerens nettleser.

Dette kapitlet er delt opp i fire hoveddeler:

- [Backend \(API\)](#)
- [Frontend \(Klient\)](#)
- [Sikkerhet](#)
- [Amazon Web Services \(Skyløsning\)](#)

3.3.1 BACKEND

Systemarkitekturen for vår applikasjon følger et tre lags model view controller (heretter MVC) prinsipp. Hvor klientens forespørsler blir koblet opp mot funksjoner i kontrollerne til applikasjonen. Applikasjonen bruker Spring som rammeverk, sikkerhets-rammeverket [Spring Security](#) og databasehåndterings-rammeverket [Spring Data-JPA](#).

3.3.1.1 TEKNOLOGIER

Backend omhandler alt innholdet i API-et som igjen benyttes av applikasjonen til å gjøre databasekall på forespørsler fra frontend.

3.3.1.1.1 MAVEN

Målet til Maven er at man raskt skal kunne komme i gang med utviklingsprosessen. En sentral fil i et Maven prosjekt er POM (Project Object Model) filen, Maven prosjektet bygges ved hjelp av denne filen som inneholder et sett med biblioteker og rammeverk kalt plugins. Dette er typiske tillegg som man ellers måtte lagt til manuelt i en tradisjonell Java applikasjon. Maven tillater at disse lastes ned automatisk. Med det kan man si at Maven gjør at man raskt kommer i gang med nye prosjekter, og gjør det enkelt å legge til og fjerne tillegg man ikke har behov for. (*Maven – Introduction, 2019*)¹⁴

```
<dependency>
  <groupId>com.auth0</groupId>
  <artifactId>java-jwt</artifactId>
  <version>3.7.0</version>
</dependency>
```

Figur 69 - Eksempel plugin

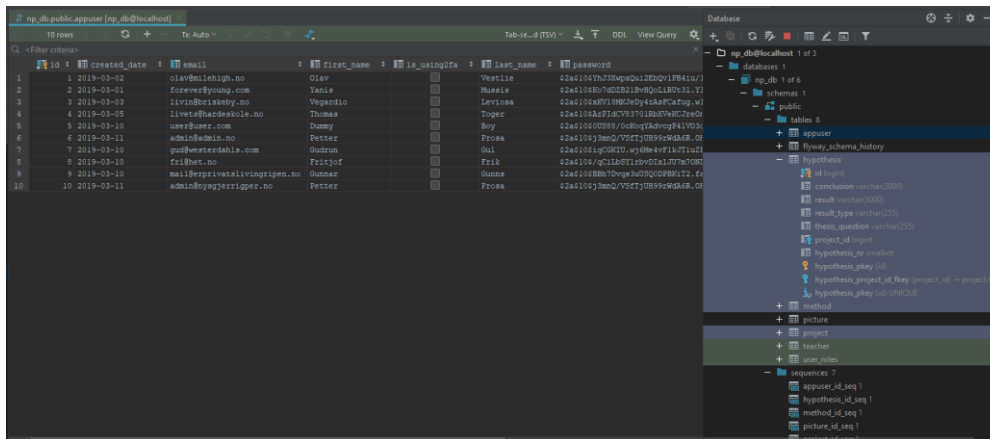
3.3.1.1.2 POSTGRESQL

Er et objektreasjonelt databasesystem med åpen kildekode. En objektreasjonell database er en type database som baserer seg på prinsippene for relasjonsdatabaser, men som har funksjoner som bygger på idéer fra objektorientert programmering. (*What is an Object-Relational Database (ORD)? - Definition from Techopedia*)¹⁵

API-et vårt er koblet opp mot dette databasesystemet ved hjelp av [Spring Data-JPA](#) rammeverket.

¹⁴ Maven – Introduction. <https://maven.apache.org/what-is-maven.html>

¹⁵ What is an Object-Relational Database (ORD)? - Definition from Techopedia. <https://www.techopedia.com/definition/8714/object-relational-database-ord>



Figur 70 - IntelliJ gir en god presentasjon av databasen, samt gir mulighet for å endre den direkte.

3.3.1.1.3 SPRING BOOT

Spring Boot er en applikasjonsplattform i Java som gjør at utviklere kommer raskere i gang med prosjekter, man slipper å bruke mye tid på å sette opp og konfigurere applikasjonen. Ved at man implementerer en main metode som inneholder den statiske SpringApplication klassen som kjører argumentene sendt til main, som igjen tar seg av rutingen av beans videre. (*Spring Boot*)¹⁶

```

@EnableSwagger2
@SpringBootApplication
@EnableJpaRepositories(basePackages = "com.np.pv.dal")
@EntityScan("com.np.pv.dal.models")
public class PvApplication {
    public static void main(String[] args) {
        SpringApplication.run(PvApplication.class, args);
    }

    //https://www.tutorialspoint.com/spring_boot/spring_boot_enabling_swagger2.htm
    @Bean
    public Docket productApi() {
        return new Docket(DocumentationType.SWAGGER_2).select()
            .apis(RequestHandlerSelectors.basePackage("com.np.pv.web.controllers")).build();
    }
}

```

Figur 71

¹⁶ Spring Boot. <https://spring.io/projects/spring-boot>

3.3.1.1.4 SPRING SECURITY

Spring Security er et rammeverk som fokuserer på å muliggjøre autentisering og autorisasjon i Spring applikasjoner. Det er et sterkt og fleksibelt autentiserings og aksess kontroll rammeverk. *(Spring Security)*¹⁷

3.3.1.1.5 SPRING DATA-JPA

Er et Java-rammeverk som håndterer entitets modeller og Repositories. Dette sørger for at man relativt enkelt kan lage spørringer mot databasen.

```
package com.np.pv.dal.repository;

import com.np.pv.dal.models.User;
import org.springframework.data.jpa.repository.JpaRepository;
import org.springframework.stereotype.Repository;

import java.util.Optional;

@Repository
public interface UserRepository extends JpaRepository<User, Long> {
    User findOneByEmail(String email);

    Boolean existsByEmail(String email);

    Optional<User> findByEmail(String email);
}
```

Her er det laget tre egendefinerte spørringer. Utover disse trengs det nesten ingen andre da de ligger innebygd i utvidelsen JpaRepository. @Repository annoteringen forteller spring at interfacet er ett repository. *(Spring Data JPA)*¹⁸

3.3.1.1.6 LOMBOK

Er et rammeverk som produserer standard metoder som getters og setters, og konstruktører for klasser. *(Project Lombok)*¹⁹

3.3.1.1.7 FLYWAY

For å kunne publisere API-et vårt til samme database instans (altså RDS) som tidligere, var det helt sentralt å implementere et database-migreringsverktøy. Migreringsverktøyet Netcompany oppfordret oss til å benytte var Flyway, som implementeres som en avhengighet i Maven.

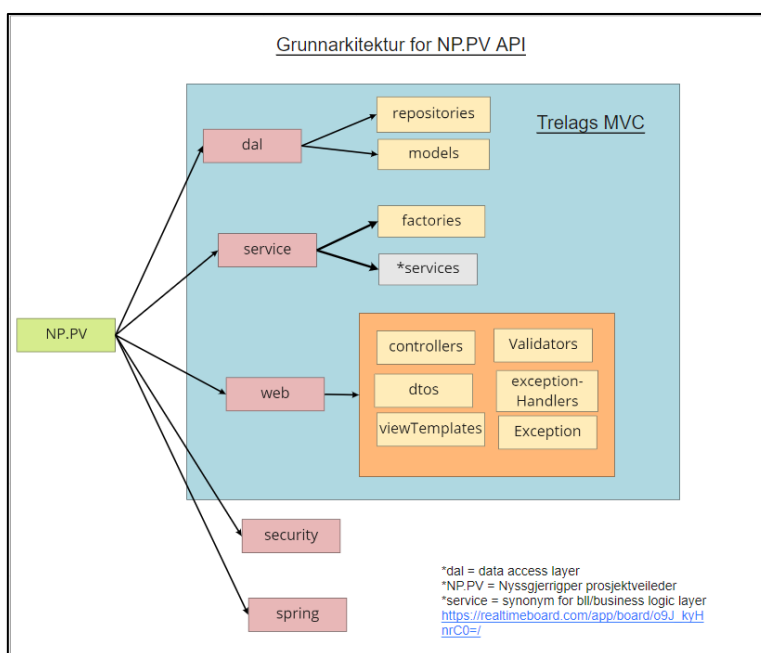
¹⁷ Spring Security. <https://spring.io/projects/spring-security>

¹⁸ Spring Data JPA. <https://spring.io/projects/spring-data-jpa>

¹⁹ Project Lombok. <https://projectlombok.org/features/all>

3.3.1.2 ARKITEKTUR

Dette delkapittelet vil beskrive de ulike klassetypene som finnes i API-et og hvordan de samarbeider for å oppfylle funksjonalitetene til applikasjonen.



Figur 72 - API-ets pakkestruktur. Med mappene som inngår i de tre lagene definert.

For hver pakke, (de rosa rektanglene på figuren over), så vil det bli presentert deres plass i arkitekturen, etterfylt av et UML diagram – Unified Modeling Language, som viser koblingene klassene i hver enkelt pakke har til hverandre. Samt deres verdier og metoder der det er av interesse.

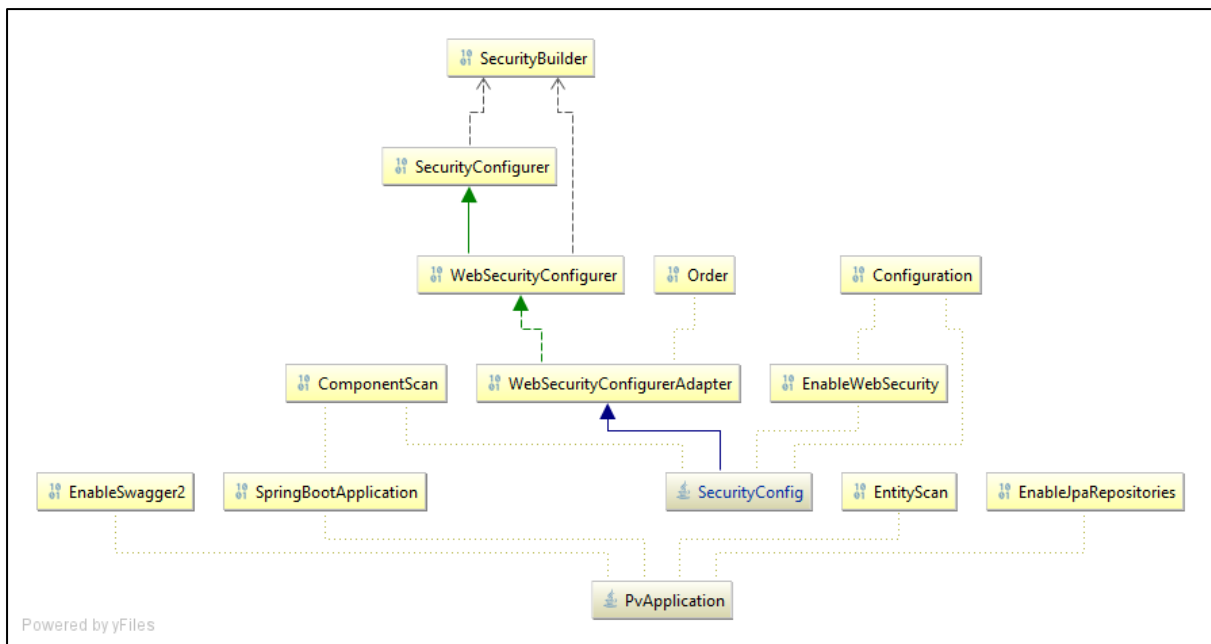
3.3.1.2.1 SPRING

Kjernen i en Spring-basert applikasjon er Spring Boot. Spring Boot er designet for å få applikasjonen opp og kjøre, så raskt som mulig, med minimalt med konfigurering. (spring.io)²⁰

Vi valgte å benytte Spring Security da det dekket behovene vi hadde til sikkerhet, både når det kom til autentisering og autorisasjon.

Spring DataJPA ble benyttet fordi det gjør det enkelt å implementere *Java Persistence API* (JPA) baserte repositories.

²⁰ spring.io. <https://spring.io/>



Figur 73

3.3.1.2.2 ENTITETER OG MODELLER

Applikasjonen har en domenemodell. Denne domenemodellen er konstruert slik at den er beskrivende for bruken av applikasjonen. Domenemodellen består av flere modeller som lagres i databasen og hentes derfra ved hjelp av spørringer generert av Spring DataJPA. Disse modellene samsvarer direkte med entitetene og dataene deres er lagret i databasen.

3.3.1.2.2.1 ENTITETER

I underkapitlene nedenfor beskrives modellene som utgjør domenemodellen.

3.3.1.2.2.1.1 BRUKER:

Blant disse har vi brukeren (User), dette er modellen som inneholder informasjon som gir autentisering og autorisasjon til brukeren av applikasjonen. Da med rolle håndtering som ADMIN og eller USER. Samt inneholder den påloggingsinformasjon, altså epostadresse, hashet passord og en hemmelighet(secret) som benyttes til generering av Google Authenticator koder. Alle brukerkontoene på systemet må å ha attributtene til bruker, derfor var det hensiktsmessig å opprette en egen brukerklasse.

3.3.1.2.2.1.2 LÆRER:

Arver av User og har dermed alle dens attributter. Læreren er tiltenkt å være den brukeren av applikasjonen som lager prosjekter. Derfor har en lærer en-til-mange kobling mot prosjekt, da læreren kan ha mange prosjekter.

3.3.1.2.2.1.3 PROSJEKT:

Er selve grunnstenen for applikasjonens funksjonalitet. Et prosjekt starter med en problemstilling som brukeren/læreren har definert. Sammen med litt metadata om prosjektet. Som hvilken skole, hvor mange elever som deltar og hvilket alderstrinn de er. Hvert prosjekt har en en-til-mange kobling med til hypoteser.

3.3.1.2.2.1.4 HYPOTESE:

For å svare på problemstillingen beskrives flere hypoteser som elevene skal undersøke om svarer på problemstillingen. Disse hypotesene har hver en beskrivelse av et mulig svar på problemstillingen.

3.3.1.2.2.1.5 METODE:

Hver hypotese har da en eller flere metoder som innebærer at man gjennomfører en eller flere forskningsmetoder hvor man innhenter informasjon. Metoden har en type id, denne samsvarer med iden der metodetypen er definert i [Contentful](#). Metodene er typisk Intervju, Observasjon, Måling osv. Samtidig er det mulig for brukeren å lage egendefinerte metoder.

Til slutt i hver enkelt hypotese fyller man ut en konklusjon på basert på informasjon man har hentet inn ved hjelp av metodene.

Deretter avslutter man prosjektet ved å fylle ut en sluttkonklusjon basert på hva man har funnet ut i hver hypotese og man kan fylle inn mer informasjon om hvordan det var å jobbe med prosjektet.

3.3.1.2.2.1.6 BILDE:

Samtlige bilder som brukeren laster opp til prosjektet lagres i databasen. For å unngå at bildene er spredt ut i de forskjellige tabellene de tilhører har vi implementert en egen bildetabell. Bildetabellen er bygd opp en måte som gjør at den støtter lagring av bilder fra de entitetene man ønsker skal ha bilder i databasen.

Dette gjøres ved at det blir lagret en forelder id (`parent_id`) som representerer id-en til entiteten den tilhører. Videre har tabellen en streng med navn forelder type (`parent_type`) som representerer hvor i prosjektveilederen bildet skal plasseres. Forelder typen blir håndtert ved hjelp av en enum i backend. Et bilde lagres ved hjelp av bruker id, forelder id, forelder typen og selve bildet som lagres som en base64 enkodet streng. Typen blir sendt inn som en `@PathVariable*`, noe som gjør at den vil kunne tolkes som en vanlig URL av

klienten, denne kontrolleres opp mot enumen, dersom den matcher vil bildet lagres. (“PathVariable (Spring Framework 5.1.7.RELEASE API),” 2019)²¹

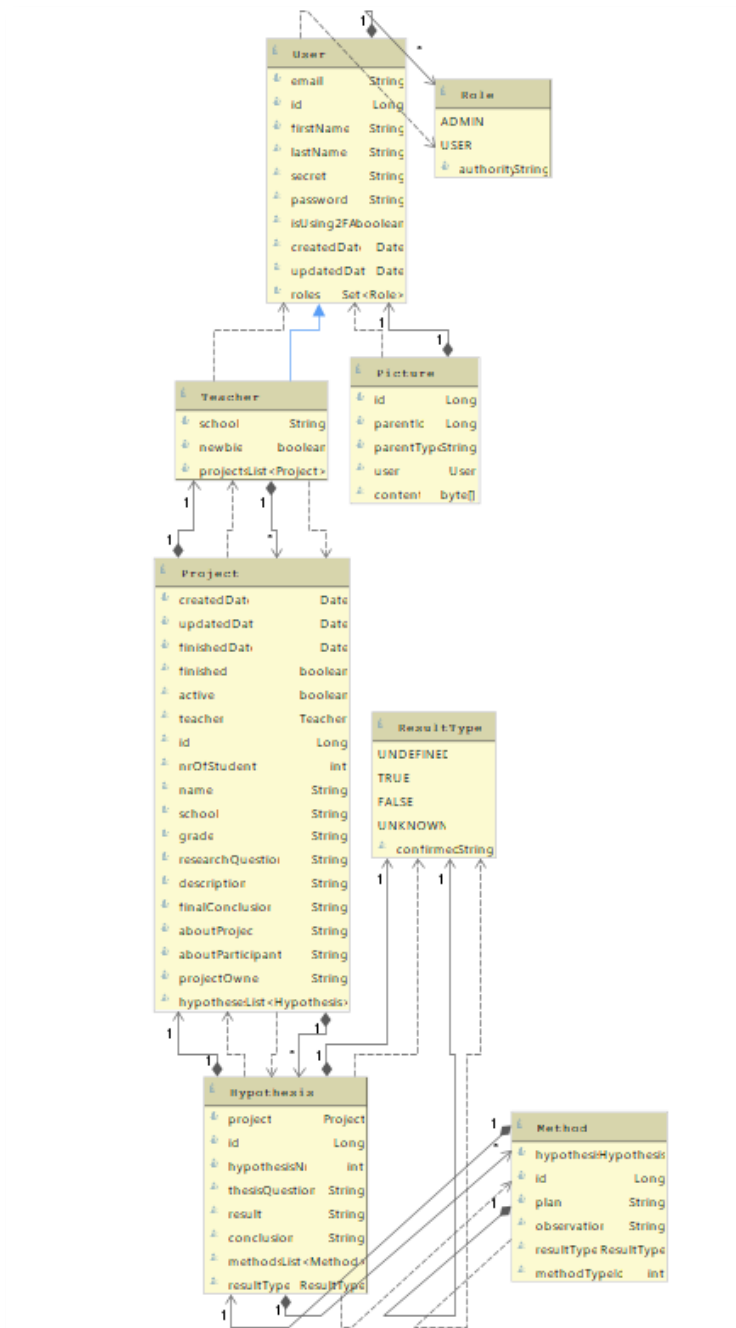
Bilde id-en til entiteten som skal kunne ha bilder blir sendt med i ViewTemplaten til den korresponderende entiteten.

Klienten vil så ha mulighet til å hente bildene ved hjelp av id-en til bildet når det er behov for dem.

²¹ PathVariable (Spring Framework 5.1.7.RELEASE API). (2019, mai)
<https://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/web/bind/annotation/PathVariable.html>

3.3.1.2.2.2 DOMENEMODELLEN

En lærer kan ha flere prosjekter, hvert prosjekt kan ha flere hypoteser som skal svare på problemstillingen, hver hypotese kan ha flere metoder som forsøker å svare på hypotesen. Når alt dette er samlet sammen og dataene er fylt inn blir prosjektet automatisk satt til ferdig (finished settes til sann). Det gjør at produkteier vil kunne se at prosjektet er ferdig i administratorgrensesnittet.



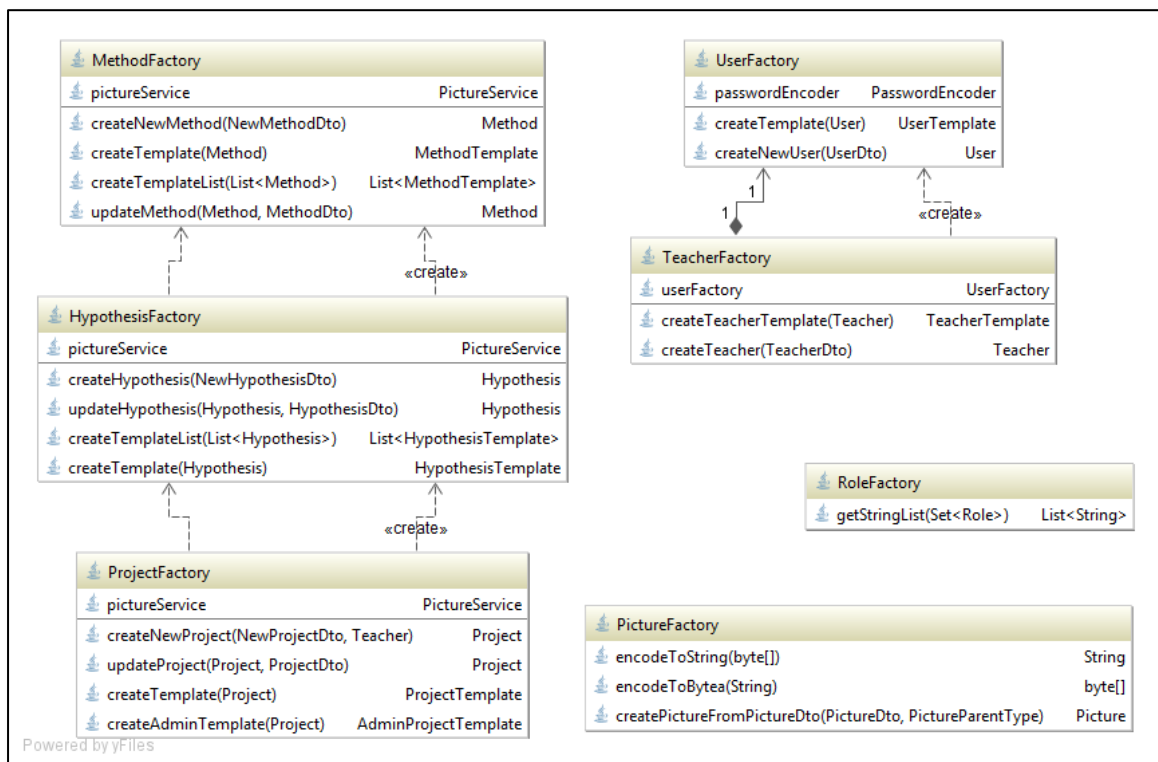
Figur 74 - UML for Bruker og Lærer samt koblingen til ->Prosjekt->Hypotese->Metode

3.3.1.2.3 FABRIKKER

Fabrikkene har til hensikt å konvertere objekter til og fra DTO-er, modeller og ViewTemplates. Her brukes de til å lage modell-objekter fra DTO-objekter. Det har blitt opprettet funksjoner for endring av objekter. Endring av gamle objekter skjer ved hjelp av DTO-objekter med ny informasjon.

```
public class ProjectFactory {  
  
    public static Project createNewProject(@NonNull NewProjectDto dto, @NonNull Teacher teacher) {  
        return new Project(teacher, dto.getNrOfStudents(), dto.getSchool(),  
            dto.getGrade(), dto.getName(), dto.getProjectOwner());  
    }  
}
```

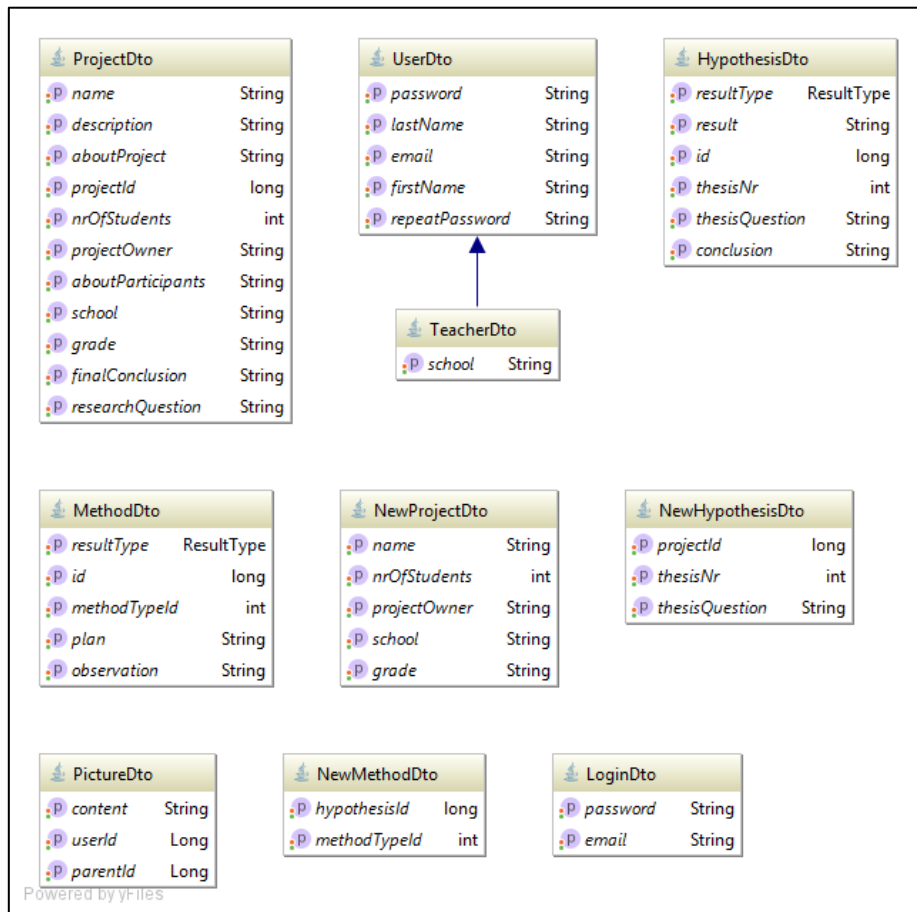
Figur 75 - Funksjon som lager ett nytt Prosjekt utfra DTO og en Teacher



Figur 76 - UML av fabrikkene til applikasjonen.

3.3.1.2.4 DTO – DATA TRANSFER OBJECT

Er klasser som representerer objektene som sendes fra klienten til API-et. Disse sendes til fabrikkene som enten lager et nytt modell-objekt, eller endrer et eksisterende modell-objekt, før de lagres i databasen. Forskjellen mellom newDto og Dto er at de som er new i navnet ikke har en id, da denne genereres automatisk første gang objektet lagres i databasen. Mens der man har Id, kan man benytte denne til å hente ut de gamle objektene fra databasen, før de endres, for så å lagres.



Figur 77

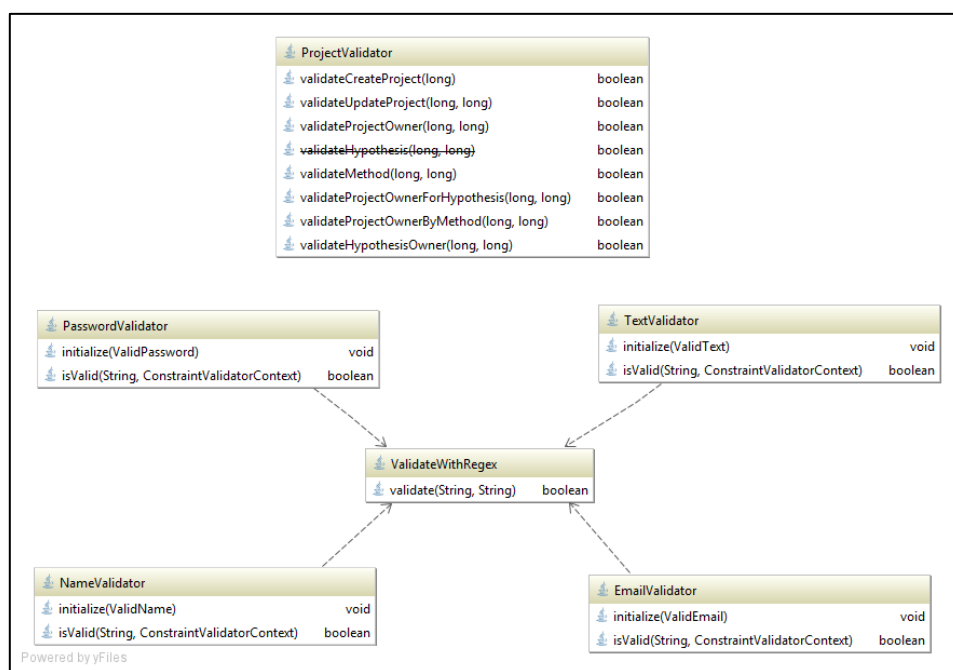
3.3.1.2.5 VALIDATORER:

Hensikten med validatorene er å kontrollere at dataene sendt mot API-et er korrekte i forhold til krav satt av produkteieren. Derfor blir attributtene som settes av sluttbruker i DTOen validert før DTOen konverteres, for så å lagres, eventuelt forkastes. Eksempelvis kan dette være at produkteier ikke ønsker navn med tegn og symboler. Eller en sjekk på at en epostadresse faktisk er en epostadresse.

Eller i prosjekt valideringen (projectValidator), så sjekkes det at en bruker eier de ressursene de forsøker å endre.

SQL injisering

Da vi ikke lar brukerne av applikasjonen lage spørringer selv, men vi selv har definert spørringene mot databasen, og spørringene kun benytter posisjonsbaserte parameter er databasen vår relativt trygg i forhold til SQL injisering. (Punjal, 2017)²² Dermed slipper vi å validere data for å forhindre SQL injisering.



Figur 78

²² Punjal, R. P. (2017 april). Custom Annotation in Java for SQL Injection Safe Parameters <https://dzone.com/articles/custom-annotation-in-java-for-sql-injection-safe-p>

3.3.1.2.6 KONTROLLER KLASSER

Kontroller klassene er API-knutepunktet som klienten sender forespørsler mot. Headeren i forespørselen filtreres først av Spring Security, hvor autorisasjons-tokenet hentes ut. Tokenet benyttes til å sjekke om brukeren har rettighetene til å aksessere metoden forespørselen ble gjort mot. Avhengig av kallet vil parameterne som mottas fra klienten valideres, se [Validatorer](#). Deretter kan kontrollerne gjøre et direkte repository-kall om forespørslene er enkle og uten behov for særlig logikk, men om de trenger mer logikk så kjøres de gjennom service klasser.

Dersom forespørslene feiler, så kastes et unntak med informasjon om feilen. Applikasjonen har flere kontrollere, de er inndelt etter hvilke behov forespørselen møter, samtidig har inndelingen blitt gjort med hensyn til tilgangskontroll.

Kontrollerne er av typen Spring RestController, noe som gjør at de automatisk vil kunne «mappes» til forskjellige adressekall. Dersom metodene har annoteringen `@ResponseBody`, konverteres returverdien automatisk til JSON eller XML respons. (Pankaj, 2018)²³

```
@RestController
@RequestMapping(value = "/projects")
public class ProjectController {
```

Figur 79 – Grunn-pathen til forespørslene ligger her på [API URL]/projects, de som har tilgang til denne sektoren vil kunne benytte den, se [Sikkerhet](#). Defineres her at det er en RestController

```
@RequestMapping(method = RequestMethod.POST)
public ResponseEntity<ProjectTemplate> createProject(@RequestBody @Valid NewProjectDto projectDto) {
    long userId = sessionService.getUserId();
    if (projectValidator.validateCreateProject(userId) {
        Project project = projectService.createProject(projectDto, userId);
        return ResponseEntity.ok(projectService.createTemplateFromProject(project));
    }

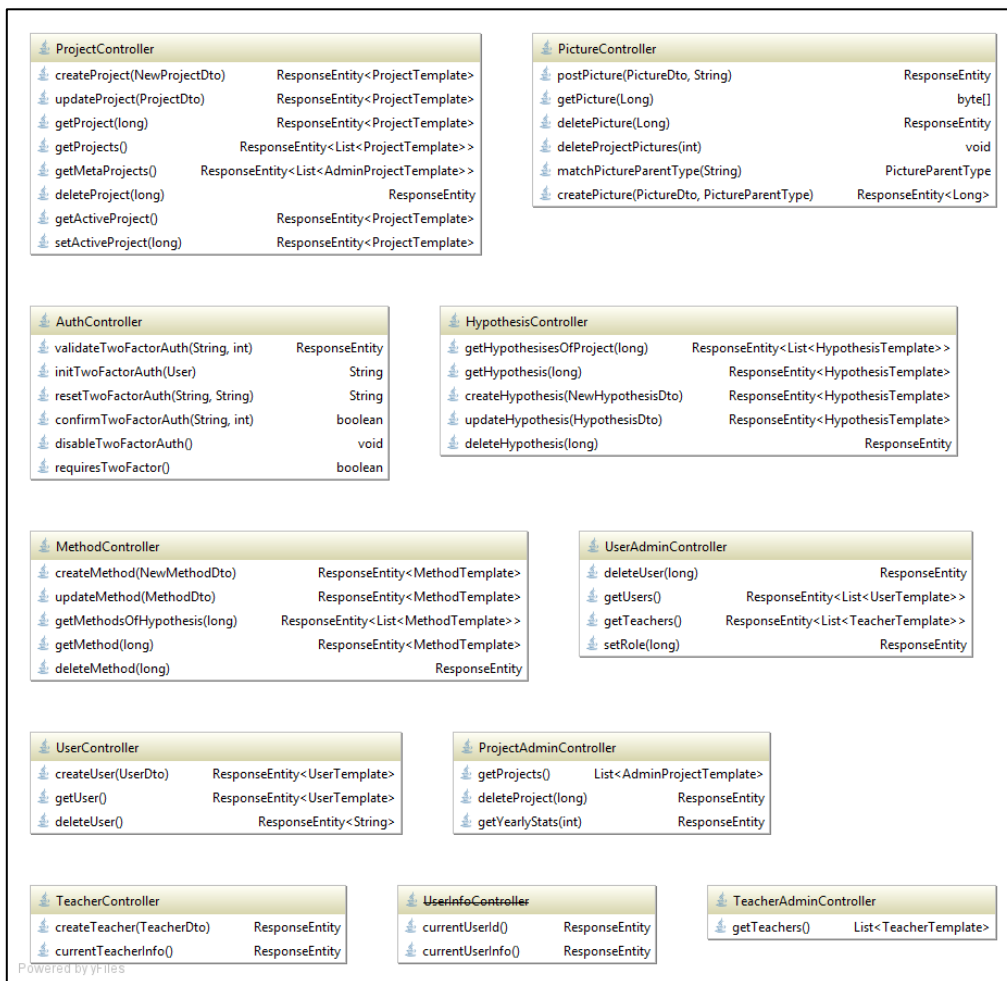
    throw new ProjectNotValidatedException("Project data not validated.");
}
```

Figur 80 - Typisk Post forespørsel.

Figur 80 viser en typisk Post forespørsel. Da metoden ikke har sin egen path, er dette kallet mappet til /projects, hvor det defineres at det skal være en POST forespørsel. Denne tar imot NewProjectDto, sjekker om DTOen valideres riktig med `@Valid`. Deretter sjekkes det om brukeren som er pålogget eier ressursen. Dersom dette stemmer lages en ny modell av typen Prosjekt som lagres i databasen ved hjelp av et

²³ Pankaj, P. (2018, Juni). Spring RestController. <https://www.journaldev.com/21536/spring-restcontroller>

kall til service, videre lages en ViewTemplate som returneres tilbake wrappet i en ResponseEntity. Om dette feiler vil det kastes et unntak.



Figur 81

3.3.1.2.7 VIEWTEMPLATES:

Er objektene som returneres fra API-et til klienten. Fabrikkene gjør om modeller til ViewTemplates. ViewTemplatene inneholder informasjon som er av verdi for klienten.

Class	Field	Type
ProjectTemplate	description	String
	conclusionPic	Long
	aboutUsPic	Long
	nrOfStudents	int
	projectOwner	String
	aboutParticipants	String
	hypothesisTemplate	List<HypothesisTemplate>
	aboutProjectPic	Long
	descriptionPic	Long
	grade	String
	finalConclusion	String
	researchQuestion	String
	projectId	Long
	name	String
	aboutProject	String
	school	String
	teacherId	Long
HypothesisTemplate	result	String
	hypothesisPic	Long
	id	Long
	resultType	String
	methodTemplateList	List<MethodTemplate>
	thesisNr	int
	thesisQuestion	String
conclusion	String	
StatTemplate	countFinishedMonth	List<Integer>
	totalTeachers	int
	notFinished	int
	countCreatedTeacherMonth	List<Integer>
	finished	int
	countCreatedMonth	List<Integer>
TeacherTemplate	lastName	String
	id	long
	created	String
	numberOfProjects	int
	numberOfCompletedProjects	int
	EMail	String
	school	String
firstName	String	
MethodTemplate	id	Long
	resultType	String
	methodTypeId	int
	planPic	Long
	observationPic	Long
	plan	String
observation	String	
UserTemplate	id	long
	lastName	String
	created	String
	email	String
	firstName	String
roles	Set<Role>	

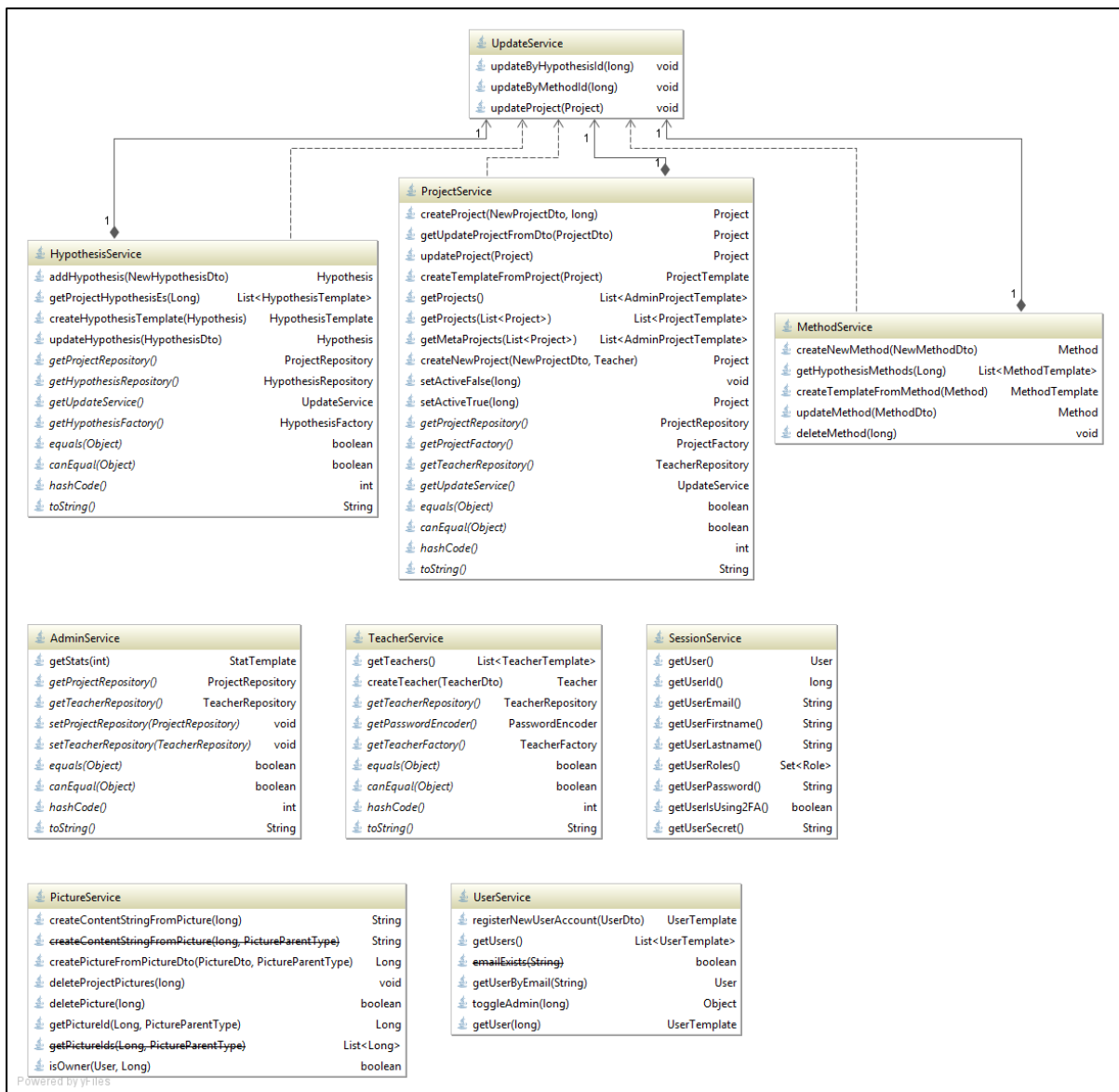
Figur 82

3.3.1.2.8 BUSINESS LAYER SERVICES:

Klassene i service pakken er klasser som utfører avanserte forespørsler mot databasen, samt bruker fabrikker til å endre DTOer til modeller. Eller modeller til ViewTemplates. Servicene gjør kall på repositoriene for å endre, lagre nye eller hente objekter fra databasen, før de returneres til kontrollerne som ViewTemplates. Hensikten med servicene er også å forhindre at kontrollerne utfører for mye logikk.

```
public List<AdminProjectTemplate> getProjects() {  
    return projectRepository.findAll() List<Project>  
        .stream() Stream<Project>  
        .map(ProjectFactory::createAdminTemplate) Stream<AdminProjectTemplate>  
        .collect(Collectors.toList());  
}
```

Figur 83 - Funksjon i ProjectService. Denne sender alle prosjektene i projectRepository gjennom Collector. For å på samtlige gjennomføre en operasjon hvor de omgjøres til AdminProjectTemplate. De returneres da som en liste med slike template.



Figur 84

3.3.1.3 FORESPØRSLER MOT API-ET

API-et tar imot forespørsler fra klienten, autorisering av brukeren gjøres ved hjelp av rettighetene brukeren har og eierskapet til ressursene. Forespørslene går gjennom løkker i API-et. Kontrollerne sender forespørslene gjennom en validering, for så å be Service-laget om å utføre logikk på dem.

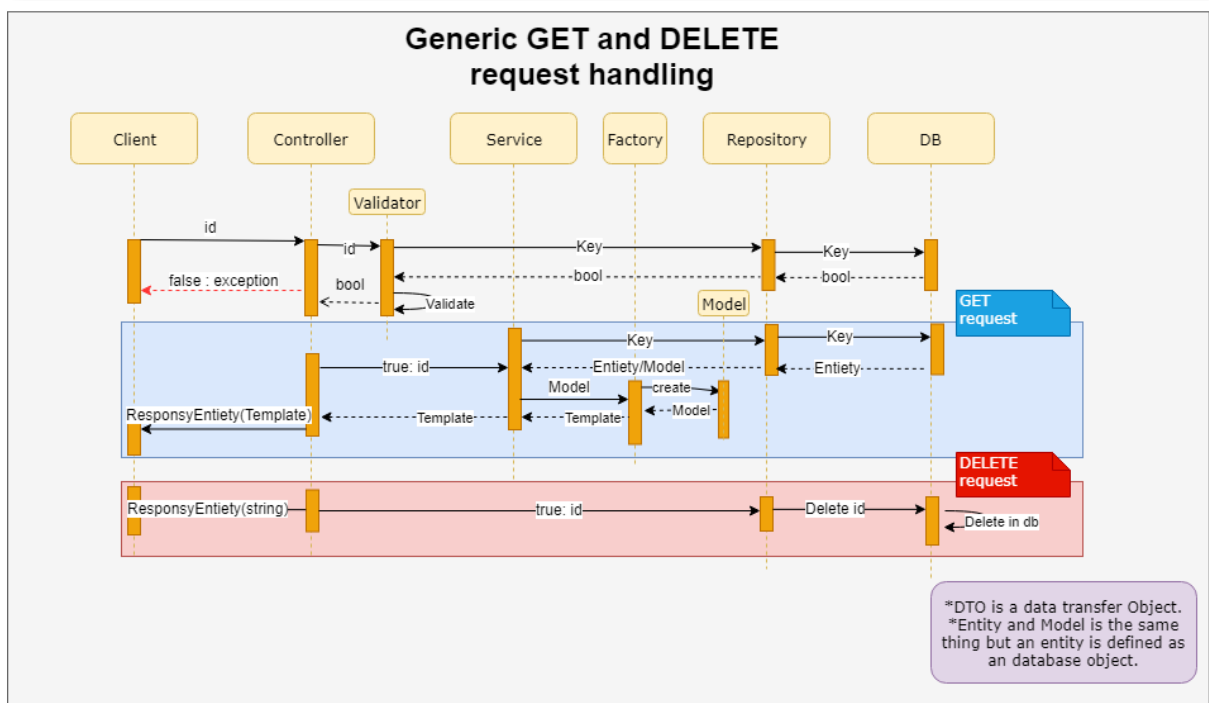
Service-laget benytter Sping DataJPA rammeverket til repositoriene for å håndtere databasekallene, som omhandler henting, endring og sletting av entiteter.

Ved en GET forespørsel, vil en eller flere entiteter hentes ut fra databasen, deretter endres det eller disse til ViewTemplate-objekt(er) ved hjelp av Factories. GET, POST og PUT vil alle returnere et objekt av denne typen, hvor objektet som håndteres i databasen blir returnert i sin oppdaterte form.

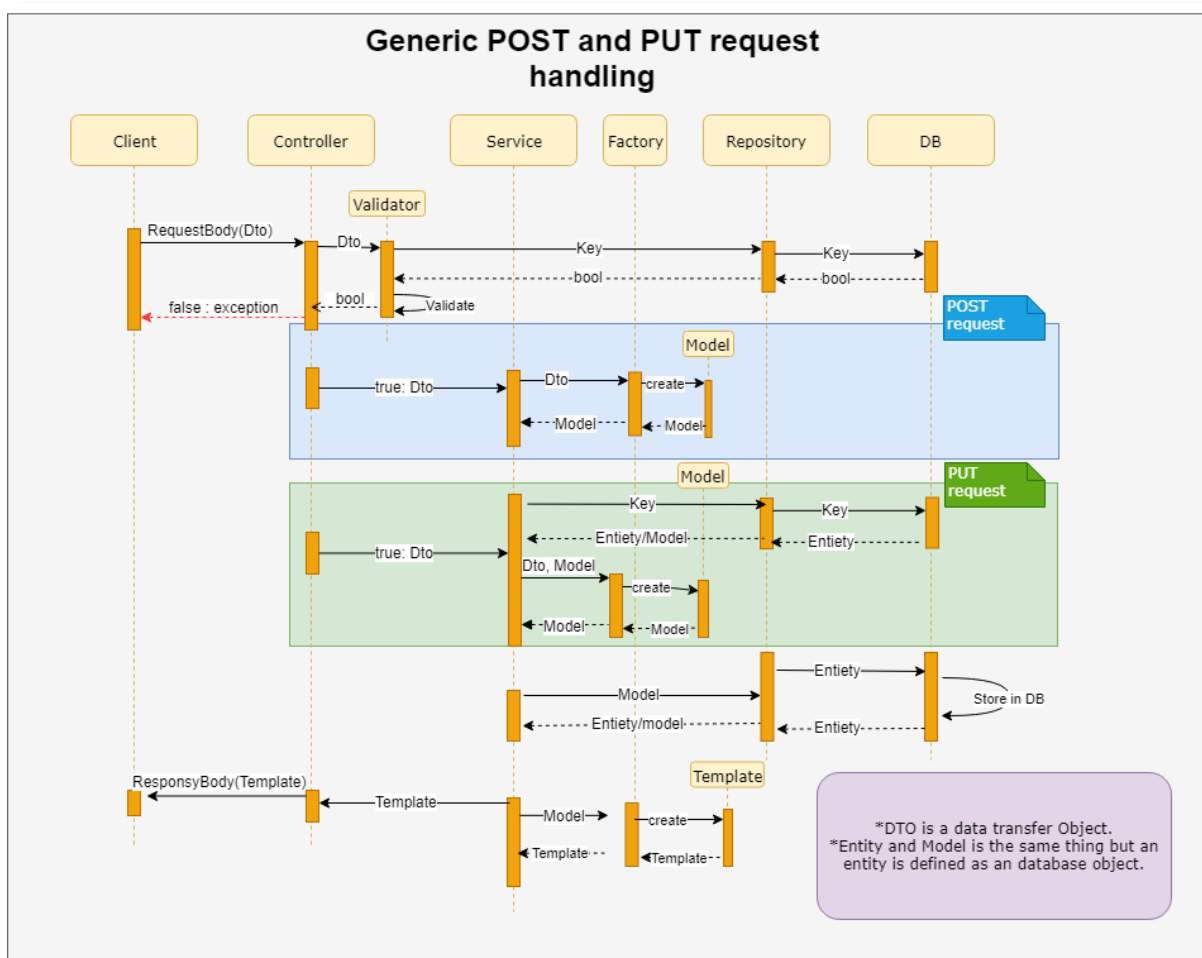
Ved en POST forespørsel, vil det i de fleste tilfeller bli lagret en ny entitet i databasen. Et DTO objekt gjøres om ved hjelp av en fabrikk til et objekt som kan lagres i databasen, før det lagres. Unntaksvis benyttes POST forespørsler også til brukerpålogging, selv om det ikke er noe nytt som lagres i databasen.

Ved en PUT forespørsel, vil en entitet hentes ut fra databasen, og deretter vil ny oppdatert informasjon overskrive gammel informasjon. Et DTO-objekt som inneholder den nye informasjonen mottas fra klient. Den oppdaterte entiteten lagres så til databasen igjen.

Ved en DELETE forespørsel, vil man slette en entitet fra databasen, en id vil mottas i forespørselen. Id-en benyttes til å slå opp i databasetabellen det gjelder, for så å slette oppføringen.



Figur 85 - Generisk visning av hvordan GET og DELETE forespørsler ser ut i applikasjonen.

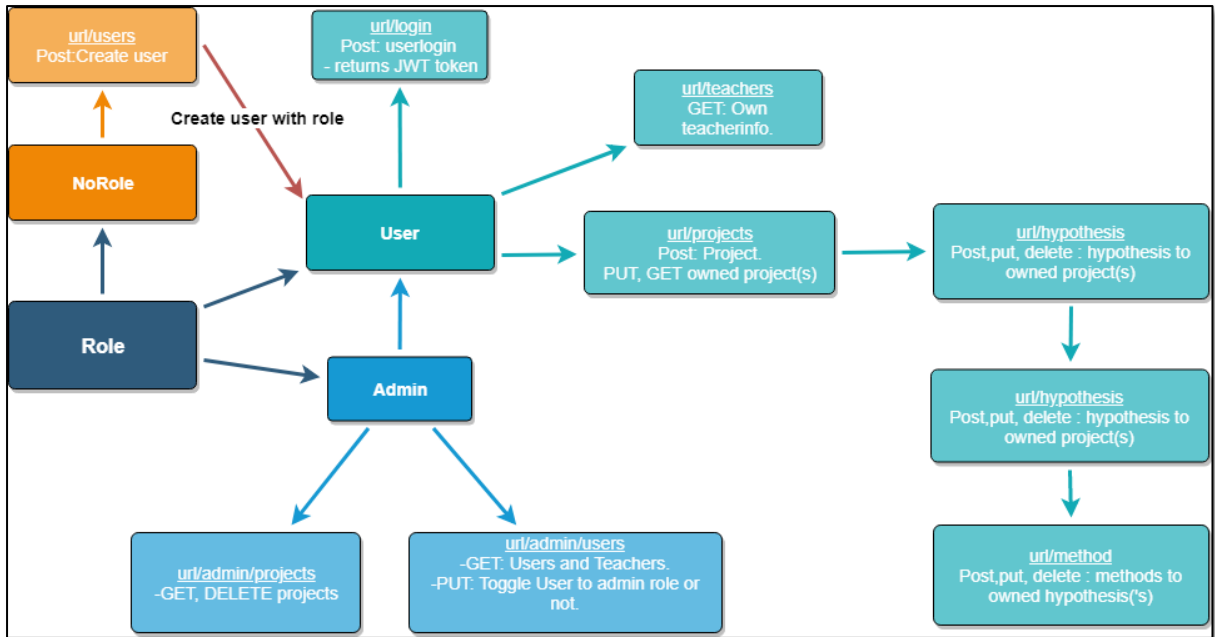


Figur 86 - Generisk visning av hvordan POST og PUT forespørsler ser ut i applikasjonen.

3.3.1.4 ROLLER

Applikasjonen har to ulike type brukere, som gir tilgang til ulike API-adresser. Besøkende på applikasjonen som ikke har opprettet en bruker, eller ikke har logget inn, har ingen rolle; disse har kun tilgang til forespørslene for brukeroppsettelse og pålogging. Ved pålogging får en vanlig bruker et autorisasjonstoken (JWT), som gir tilgang til API-adressene som en vanlig bruker skal ha tilgang til. Dette innebærer at brukeren, som i de fleste tilfeller har en lærerfunksjon, kan opprette nye prosjekter og endre på de som de selv eier, samt se brukerinformasjon.

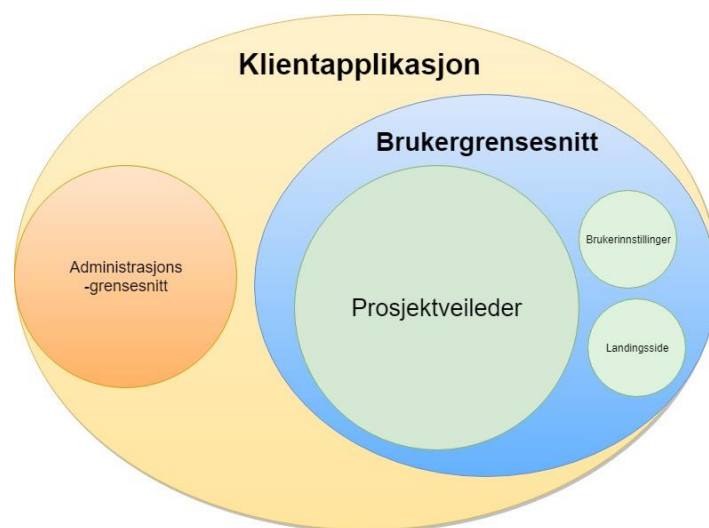
En bruker som i de fleste tilfeller vil ha USER-rollen, kan også settes til å ha en administratorrolle (ADMIN) av en som allerede er administrator. Som administrator får man full oversikt over alle brukerne av applikasjonen, med mulighet til å slette brukere, og prosjekter.



Figur 87 – Illustrasjon over rolletilganger på API-et.

3.3.2 FRONTEND

Frontend er presentasjonslaget til applikasjonen. Denne håndterer brukergrensesnittet, samt de aller fleste visuelle aspektene som sluttbrukeren kommer til å interagere med. Det samlede systemet består av prosjektveilederen og administrasjonsgrensesnittet.



Figur 88 - Hoveddelene av klienten. Størrelsene viser hvor stor del av applikasjonen de forskjellige delene er.

3.3.2.1 TEKNOLOGIER

Etter krav fra oppdragsgiver er nettapplikasjonens klient utviklet i et node.js miljø og med React. Vi har benyttet oss av create-react-app for å sette opp prosjektet og NPM for å legge til avhengigheter og pakker vi har benyttet oss av.

3.3.2.1.1 NODE.JS

Node.js er et «open-source» java-script servermiljø. Kan kjøre på de fleste plattformer og er blitt en industristandard. En av de viktigste grunnene for å benytte seg av Node.js er at det kan arbeide asynkront. Node.js kan også brukes for å sette opp en lokal webserver for klienten ved utvikling. *(Nodejs)²⁴*

3.3.2.1.2 NPM – NODE PACKAGE MANAGER

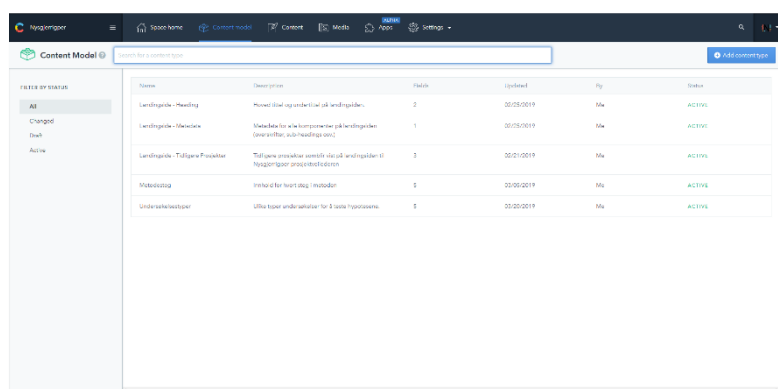
NPM er en «package manager» som via en kommandolinjeklient håndterer installasjon og oppdatering av pakker med tilleggsmoduler for «node.js». NPM er koblet opp til et offentlig tilgjengelig register av «open-source» pakker for alt av utvikling i Javascript. NPM ble startet som et «open-source» prosjekt i 2009 for å gjøre

²⁴ About Node.js <https://nodejs.org/en/about/>

deling av moduler til javascriptutvikling lettere, og blir i dag brukt i de aller fleste web-utviklingsprosjekter. (npm)²⁵

3.3.2.1.3 CONTENTFUL

Eieren hadde som krav at applikasjonen skulle benytte Contentful for å ta seg av innholds håndtering. Contentful er et «Content Management System» hvor man kan definere modeller for applikasjonen og i disse legge til innhold som da kan oppdateres ett sentralt sted, uten å måtte gjøre endringer direkte i applikasjonens kodebase. Dette innebærer at administratorer kan endre innholdet i applikasjonen ved å gå inn på Contentful sin nettapplikasjon, som i praksis betyr at de kan bytte bilder, lenker og tekst i applikasjonen ved å kun bruke et Contentful administratorpanel.



Figur 89 - Contentful administratorpanel

Innholdet i Contentful blir hentet gjennom API kall mot tjenesten, og lagres lokalt på brukerens maskin da de først besøker nettsiden. Dette introduserer en mulig svakhet i applikasjonen der vi er avhengige av en tredjepart for store deler av innholdet til vår nettside, men gjør endring av innhold lett tilgjengelig til ikke teknologer gjennom et enkelt grensesnitt vist i figuren ovenfor. (Contentful)²⁶

3.3.2.1.4 CREATE REACT APP

Verktøy for raskt oppsett av React web-applikasjoner. Dette brukte vi for å skape et tomt React prosjekt som vi da kunne bygge på videre. (Facebook Open Source)²⁷

3.3.2.1.5 REACT

React.js er et JavaScript-bibliotek som er mye brukt i moderne nettapplikasjoner. React utvider det nettbaserte språket Javascript, med tilleggsfunksjonalitet der man eksempelvis kan skrive JSX kode som er veldig likt HTML kode i React-komponenter.

²⁵ Npm. About npm. <https://docs.npmjs.com/about-npm/>

²⁶ Contentful. Contentful developer documentation. <https://www.contentful.com/developers/docs/>

²⁷ Facebook Open Source. <https://reactjs.org/docs/create-a-new-react-app.html>

React brukes hovedsakelig til å utvikle brukergrensesnitt på nett og er spesielt godt egnet til å håndtere dynamisk data, som f.eks. tekst som redigeres av bruker.

React har økt i popularitet de siste årene, noe som i stor grad er på grunn av bibliotekets implementasjon av komponenter. Dette er de sentrale byggeklossene bak enhver React-applikasjon, og gjør det relativt lett å skrive, oppdatere og implementere funksjonalitet i et program. På en side med brukerinformasjon kan React for eksempel ha adskilte komponenter for profilbildet til brukeren, alle knappene på siden og informasjonen om brukeren. Disse komponentene flettes til slutt sammen til kjørbare kode som kan vises av en tradisjonell nettleser.

Kunden ba eksplisitt om at React skulle brukes til utvikling av web-klienten, da dette skulle gjøre videre utvikling lettere og koden mer forståelig. (Facebook Open Source)²⁸

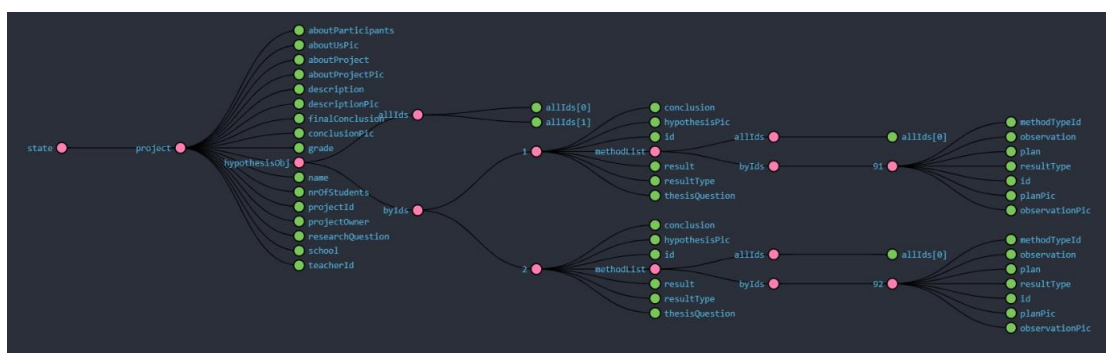
3.3.2.1.6 PAKKER/MODULER BRUKT SOM TILLEGG TIL REACT:

Det finnes et utall javascript-biblioteker som er skrevet som tillegg til React og som kan legges til i en React-applikasjon som pakker/moduler. Disse kan utvide mulighetene man har under utvikling eller forenkle en del oppgaver.

3.3.2.1.6.1 REDUX

React isolerer attributtene til komponentene sine fra hverandre. Dette betyr at det ofte er vanskelig å få tilgang til variabler og objekter man har lagret i en annen komponent. I større React-applikasjoner er det helt essensielt å ha en form for sentral tilstandshåndtering. (Abramov)²⁹

React med Redux er i dag den mest brukte av denne typen løsning og den passer vår applikasjon godt. Redux ble derfor valgt for å håndtere klientens state eller tilstand.



Figur 90 - Redux state som oppbevarer prosjekt-data

²⁸ Facebook Open Source. <https://reactjs.org/docs/getting-started.html>

²⁹ Abramov, D. Redux Core Concepts. <https://redux.js.org/introduction/core-concepts>

3.3.2.1.6.2 REACT-REDUX

Kobling mellom React og Redux. Kobler React-komponenter til Redux sin [store](#) ved å koble deler av Redux sin [store](#) til komponentene.

3.3.2.1.6.3 JS-COOKIE

Js-cookie er et enkelt lettvekts Javascript-bibliotek laget for å håndtere informasjonskapsler. (*js-cookie*)³⁰

3.3.2.1.6.4 REDUX-THUNK

Såkalt *middleware*-utvidelse til Redux som gjør at en komponent, i stedet for å kun kunne sende en enkel [action](#), kan sende en funksjon til [Redux-storens](#) dispatcher. Denne funksjonen, eller thunken, plukkes så opp av Redux-thunk og sender en eller flere actions til reduceren asynkront i forhold til resten av applikasjonen.

3.3.2.1.6.5 REDUCE-REDUCERS

Gjør at man kan dele opp koden til Redux sin [reducer](#) i flere filer for å gjøre kodebasen lettere å lese og utvikle. Reduce-Reducers setter så sammen disse filene til en reducer. Redux har en mulighet som standard for å gjøre dette, gjennom en funksjon som kalles «ComposeReducers», men til forskjell fra Redux sin standardløsning gjør Reduce-Reducers at alle reduserer kan behandle hele [state](#) der composeReducers gir hver reducer hver sin del av redux sin state. (*Redux-Utilities*)³¹

3.3.2.1.6.6 FORMIK

Et Java Script bibliotek og rammeverk for håndtering av innfyllingsskjemaer og innsending av disse. Lite av størrelse og ressursbruk, men med mange muligheter og mye funksjonalitet. (*Palmer*)³²

Nysgjerriger - Prosjektveilederen benytter Formik til alle innfyllingsskjemaer.

3.3.2.1.6.7 YUP

Rammeverk for håndtering av validering. Man setter opp valideringsskjemaer, og kan så sammenligne det som skal valideres opp mot disse skjemaene. Inneholder mange standarder og standard tilbakemeldinger ved feil. (*jquense/yup*)³³

3.3.2.1.6.8 AXIOS

Axios er et bibliotek som gjør det enklere og mer oversiktlig å sende forespørsler til og fra REST API-et gjennom XML-Httpforespørsler. Disse tillater oss å sende vanlige

³⁰ JavaScript Cookie. <https://github.com/js-cookie/js-cookie>

³¹ Redux-Utilities. Reduce-Reducers. <https://github.com/redux-utilities/reduce-reducers>

³² Palmer, J. Formik Docs. <https://github.com/jaredpalmer/formik>

³³ jquense/yup. <https://github.com/jquense/yup>

internettrafikk-pakker til API-et, som igjen kan hente og skrive data til databasen, og er verktøyet som benyttes til all kommunikasjon fra klient til server. (*axios*)³⁴

3.3.2.1.6.9 PROP-TYPES

Prop-types gir kontroll av parametertyper i kjøretid under utvikling.

React applikasjonen kan bruke prop-types for å definere hvilke typer attributter som kan sendes til en komponent. React sjekker at parameterne passer typene som er definert i komponenten. (*npm: prop-types*)³⁵

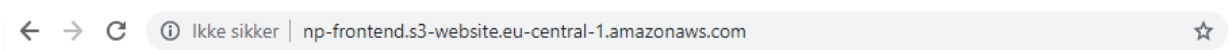
3.3.2.1.6.10 REACT-BOOTSTRAP

React-Bootstrap er et rammeverk for rask implementasjon av frontend-komponenter. Det hjelper blant annet med plassering og tilpassing av elementer. (*react-bootstrap*)³⁶

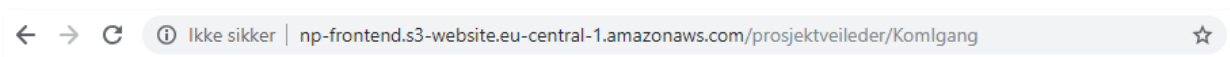
3.3.2.1.6.11 REACT-ROUTER

En Single-Page Application (forkortet til SPA) er en webapplikasjon som kan bli lastet ned av brukerens nettleser, og deretter kjørt uavhengig, uten å måtte hente inn noe annet enn backend-data. React er bygd for SPA-er.

En Single-Page Applikasjon er enhver webapplikasjon der alt frontend-innhold blir lastet ned av brukerens nettleser ved oppstart, og deretter kjørt uavhengig uten å måtte hente inn noe annet enn backend-data fra serverne. Et problem dette skaper er at navigasjon på nettsiden blir noe utfordrende. React Router er et tilleggsbibliotek som gjør det mulig å simulere tradisjonell navigasjon på en nettside, og emulerer dette med blant annet brukerens URL, ved hjelp av routing og nettleserens historie. (*ReactTraining/react-router*)³⁷



Figur 91 - URLen til landingssiden til applikasjonen



Figur 92 – URL-en til "Kom i gang"-steget i prosjektveilederen

³⁴ *axios*. (n.d.). <https://github.com/axios/axios>

³⁵ *npm: prop-types*. (2019, February 13). <https://www.npmjs.com/package/prop-types>

³⁶ *react-bootstrap*. (n.d.). Retrieved May 23, 2019, from <https://react-bootstrap.github.io/>

³⁷ *ReactTraining/react-router*. <https://github.com/ReactTraining/react-router>

```
<Router history={history}>
  <div className="App">
    <Route path="/" exact component={LandingPage}/>
    <Route path="/prosjektveileder" component={GuidePage}/>
    <Route path="/lagBruker" component={CreateUser}/>
    <Route path="/admin" component={ AdminApp } />
    <Route path="/prosjektutskrift" render = { () => ( <Proje
    <LoginModal show={this.props.showLoginModal}/>
  </div>
</Router>
```

Figur 93 - Routing tabellen i App.js

3.3.2.1.6.12 REDUX DEVTOOLS EXTENTION

Utviklerverktoy-panel som brukes til feilsøking og oversikt over Redux-tjenester som state og store i en nettleser. (*redux-devtools-extension*)³⁸

³⁸redux-devtools-extension. <https://github.com/zalmoxisus/redux-devtools-extension>

3.3.2.2 ARKITEKTUR

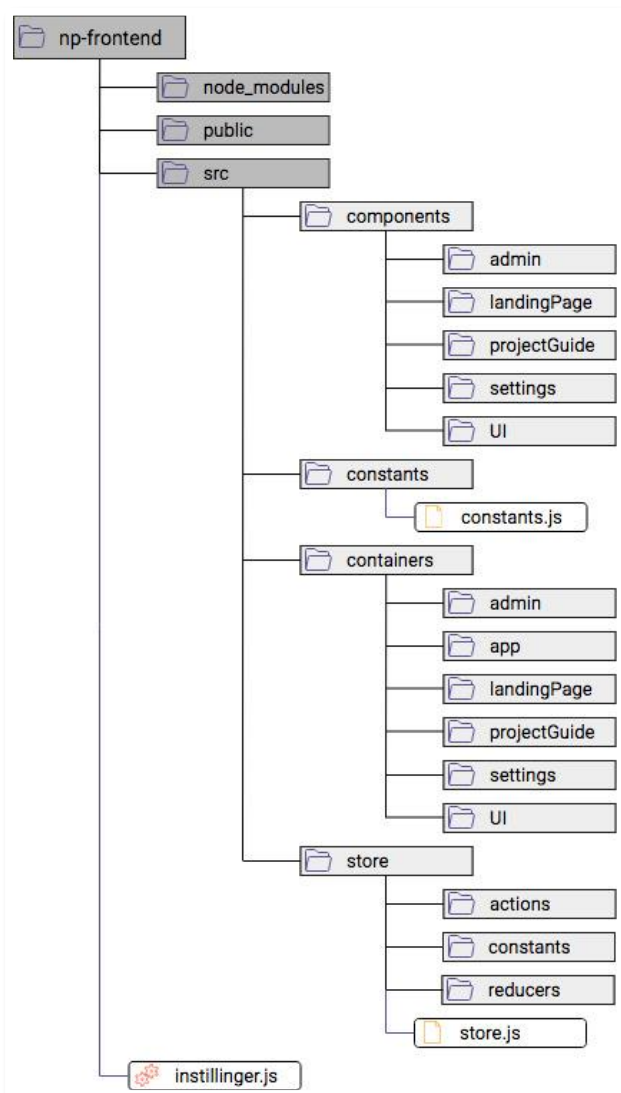
Følgende delkapittel sikter på å beskrive den underliggende strukturen i klienten og detaljene bak vår implementasjon av React og alle verktøy som har blitt lagt til for å lage produktet.

3.3.2.2.1 MAPPESTRUKTUR, FILER OG NAVNGIVNING

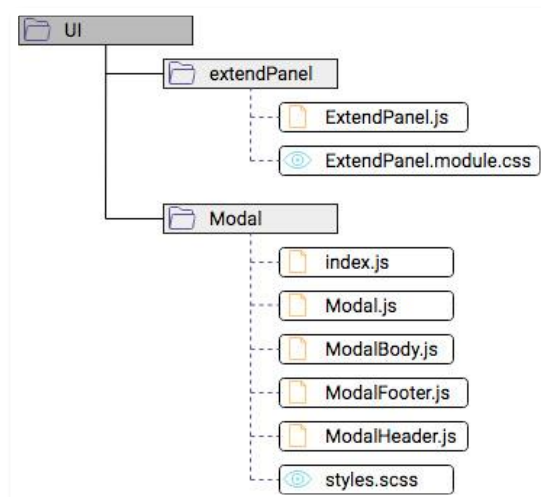
Mappestrukturen og navngivning av filer er strukturert etter et «design-pattern» som passer til prosjektets størrelse, og applikasjonens oppbygging.

3.3.2.2.1.1 MAPPESTRUKTUR

Applikasjonens kildekode ligger i «src» mappen. Det er her valgt å dele koden i fire undermapper, components, constants, containers og store. Store inneholder all kode knyttet til applikasjonens [Redux-store](#) som blant annet holder på og vedlikeholder klientens [state](#).



Figur 95 - Mappestruktur



Figur 94 - Navngivningskonvensjoner

3.3.2.2.1.2 NAVNGIVNINGSKONVENSJONER

Vi har valgt en navngivningsmetode som skal være gjennomgående for prosjektet som skal gjøre importering av komponenter, funksjoner og [actions](#) lettere. Vi har definert at mapper i hovedsak skal ha camel-case (med liten forbokstav) bortsett ifra i tilfeller der komponentmoduler sammensatt av flere komponenter kan importeres direkte fra mappen (via en index.js fil). I disse tilfellene skal mappen ha stor forbokstav.

3.3.2.2.1.3 STYLING

Det er så langt det er hensiktsmessig definert at komponenter skal ha sin egen styling knyttet til seg i egne CSS-filer for å øke komponentenes gjenbrukbarhet. Disse skal ligge i komponentens mappe. Dette gjøres i all hovedsak gjennom Reacts module løsning. Til komponentmoduler, satt sammen av flere mindre komponenter som er ment å alltid brukes sammen, er stylingen gjort for hele modulen ved bruk av SASS/SCSS.

Det er i tillegg en større CSS-fil som definerer den generelle stylingen for hele klienten.

3.3.2.2.2 KONTEINERE

For applikasjonen er konteinere definert som de React-komponentene som bygger en side som skal vises ved å sette sammen andre komponenter. Disse ligger alle i «containers» mappen i koden.

Kontainerne inneholder i hovedsak lite funksjonalitet i seg selv, men knytter seg ofte opp til klientens tilstand, eller state, for å tilgjengeliggjøre verdier derfra for komponentene.

3.3.2.2.3 KOMPONENTER

I prosjektets components-mappe ligger alle komponenter som er ment som byggesteiner for brukergrensesnittet. Disse inneholder funksjonaliteten som trengs for å bygge sammen klienten og de er tilstrebet å i hovedsak være godt oppdelt for å øke mulighet for gjenbruk.

3.3.2.2.4 KONSTANTER

Det er i flere tilfeller benyttet konstanter på tvers av komponenter og konteinere. Generelle konstanter er ment å ligge i constants-mappen, mens konstanter knyttet til Redux-[actions](#) ligger i store-mappen.

3.3.2.2.5 STORE

Den delen av koden som definerer applikasjonens [store](#), som håndterer alt av endringer av klientens tilstand eller [state](#), ligger i prosjektets store-mappe. Her er alle Redux-[actions](#) som komponentene benytter for å hente eller lagre verdier i state. Disse

er ofte bygget opp som thunks, som er funksjoner som har mulighet til å sende actions asynkront.

Redux-[reduceren](#) som fanger opp disse action-ene, og gjør endringer i state i henhold til hva de inneholder, er bygget sammen av mange mindre reducere. Disse er knyttet sammen med reduce-reducers biblioteket.

I tillegg til Redux-actions og -reducerene, er alle konstanter som er koblingspunktet mellom disse, våre action-types, definert i denne delen av prosjektet.

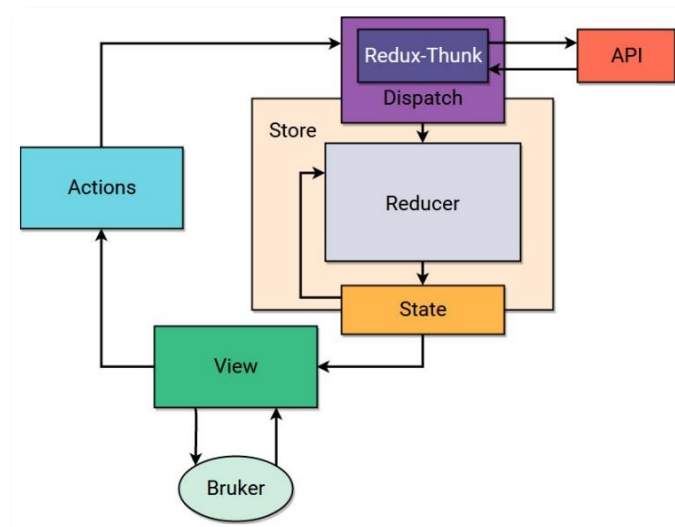
3.3.2.3 IMPLEMENTASJON AV VALGTE TEKNOLOGIER

Klienten balanserer effektivitet og et barnevennlig utseende for å appellere til både lærere og elever. Den er meget omfattende, men effektivt bygget sammen av tidligere nevnte konteinere og komponenter. For å gjøre denne koden, og utviklingen av den, så effektiv som mulig har vi benyttet mange tilleggsmoduler til React. I følgende avsnitt vil vår implementasjon av de viktigste av disse beskrives.

3.3.2.3.1 REDUX

Redux gjør at vår applikasjons state til enhver tid blir behandlet i en sentral [store](#) i stedet for å måtte være lagret i komponentene og måtte sendes fra komponent til komponent. For å gjøre endringer i state må det sendes [actions](#) via Redux sin dispatcher. Denne videresender disse til Redux sin reducer som på bakgrunn av innholdet i en action gjør endringer i state. Endringene blir så sendt ut til eventuelle komponenter som abonnerer på verdiene som er blitt endret, og endrer da dynamisk hva som vises til brukeren av applikasjonen på bakgrunn av brukerens handlinger.

Vi har utvidet Redux sin funksjonalitet med både React-Redux og React-Thunk beskrevet nedenfor.



Figur 96 - Flyten i vår implementasjon av Redux

3.3.2.3.2 REACT-REDUX

For å koble våre komponenter opp til Redux storen har vi benyttet React-Redux sin connect metode. Dette gjør at vi kan sende verdier fra state og actions, som kan sendes til dispatcheren, til en komponent som parametere.

```
125  const mapStateToProps = state => ({
126    formIsSubmitting: state.formIsSubmitting,
127    firstName: state.teacher.firstName,
128    lastName: state.teacher.lastName,
129    school: state.teacher.school
130  });
131
132  const mapDispatchToProps = dispatch => {
133    return {
134      createProject: (values) => dispatch(createProject(values))
135    }
136  };
137
138  export default connect(mapStateToProps, mapDispatchToProps)(CreateProjectForm);
```

Figur 97 - Typisk kobling av state og dispatch til en komponent

3.3.2.3.3 REDUX-THUNK

I vår applikasjon er Redux-Thunk brukt aktivt, spesielt ved forespørsler til API-et. Det gir komponentene våre mulighet til å sende en funksjon, kalt en thunk, som igjen sender actions avhengig av hva API-kallet får i retur. I et typisk tilfelle vil denne funksjonen først sende en action som gir beskjed om at et API-kall har startet. Dette kan plukkes opp i reduceren som for eksempel setter en variabel i state som forteller applikasjonen at en forespørsel pågår. Avhengig av om kallet feiler eller er vellykket vil thunken så typisk sende en action som forteller at den feilet eller en action med returverdiene om den er vellykket.

Bruken av thunks gjør at sendingen av API-kall og handlingene gjort på bakgrunn av resultatet av kallet fungerer asynkront fra resten av applikasjonens handlinger. Klienten slipper å vente på API-kallet da dette kan foregå «i bakgrunnen».

```

9   export const createProject = values => {
10
11   return ( dispatch, getState ) => {
12     dispatch(createProjectStarted());
13
14     const dataToSend = transformValues(values);
15
16     axios({method: 'post'...})
25     .then( response => {
26       console.log("Det jeg fikk tilbake: ", response.data);
27       dispatch(createProjectSuccess(transformResponseData(response.data)));
28       history.push('/prosjektveileder/KomIgang');
29     })
30     .catch( error => {
31       console.log("Noe gikk galt: ", error);
32       dispatch(createProjectFailure(error.message));
33     });
34   }
35 };
36
37 const createProjectStarted = () => ({
38   type: CREATE_PROJECT_STARTED
39 });
40
41 const createProjectSuccess = (data) => ({
42   type: CREATE_PROJECT_SUCCESS,
43   payload: {
44     ...data
45   }
46 });
47
48 const createProjectFailure = error => ({
49   type: CREATE_PROJECT_FAILURE,
50   payload: {
51     error: error
52   }
53 });

```

Figur 98 - Typisk thunk i klient-applikasjonen

3.3.2.3.4 API-KALL MED AXIOS

REST er et designmønster som skaper en kobling mellom datamaskiner og internett, og som fungerer som et abstraksjonslag mellom klienten og serveren. Gjennom såkalte RESTfulle API-kall kan vi fra React applikasjonen sende forespørsler om ressurser, som blant annet kan være autentisering av brukere, henting av prosjektinformasjon og bildeopplastning til vår backend server.

All kommunikasjon mellom klienten og serveren vår skjer via REST API kall, og disse brukes hyppig i bakgrunnen, ved hjelp av thunks, mens en bruker benytter prosjektveilederen.

I alle våre API-kall har vi benyttet [Axios](#) for å gjøre arbeidet med disse forespørslene enklere for oss å holde oversiktlige og konsistente.

```

16     axios({
17       method: 'post',
18       url: apiUrl.NP_API + '/projects',
19       data: { ...dataToSend },
20       headers: {
21         Authorization: 'Bearer ' + getState().userSession.authToken,
22         'Content-type': 'application/json'
23       }
24     })

```

Figur 99 - Typisk axios forespørsel

3.3.2.3.5 BRUK AV REACT-BOOTSTRAP

Vi har brukt [React-Bootstrap](#) for å raskt kunne implementere våre egne komponenter i applikasjonen tidlig i utviklingsprosessen. I senere sprints bygde vi videre på biblioteket ved å skreddersy *bootstrap.css*-filen som implementerte de visuelle elementene som har blitt beskrevet i Nysgjerrigpers digitale profil.

I mange av våre komponenter, som blant annet komponentene for bildeopplastning og Formik-feltene, er React-Bootstrap elementer brukt for å gjøre designet av dem raske å utvikle for å passe klientens design. I de fleste tilfeller er det React-Bootstraps container, row og column som er brukt for å forenkle arbeidet med komposisjon. I tillegg er en modifisert versjon av React-Bootstraps button hyppig brukt.

3.3.2.3.6 INFORMASJONSKAPSLER

Informasjonskapsler, også kalt for *cookies*, er midlertidige filer som lagres lokalt i brukerens nettleser. Disse gjør det mulig for en nettside å oppbevare informasjon som f.eks. brukerinnstillinger og reklamepreferanser og gjenbruke disse i en bestemt periode.

Vår applikasjon benytter en cookie for å oppbevare autorisasjons-tokenet under brukerens økt. Dette gjør det mulig for en bruker å oppdatere siden gjennom «refresh»-knappen, eller midlertidig lukke fanen, for å så kunne komme tilbake til prosjektveilederen uten å måtte re-autentiseres. Denne varer i tre og en halv timer etter at brukeren blir pålogget. Det brukes i tillegg en cookie som passer på om en bruker tidligere har samtykket til personvernerklæringen på applikasjonens landingsside, slik at den ikke kommer opp hver gang. (Nettvett, 2019)³⁹

For å håndtere disse informasjonskapslene har vi brukt jscookie.

Name	Value	Domain	Path	Expires / Max-Age	Size
auth	eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJzdWUiOiJ5YW5pc2xhdkZhZG1...	np-frontend.s3-website...	/	2019-05-22T01:13:04.000Z	215
cookieBanner	true	np-frontend.s3-website...	/	N/A	16


Figur 100 - Informasjonskapslene til en innlogget bruker vist i Chrome

³⁹ Nettvett (2019, januar). Slik administrer du informasjonskapsler <https://nettvett.no/slik-administrer-du-informasjonskapsler/>

3.3.2.3.7 VALIDERING

I vår applikasjon brukes Yup ved validering av brukerinput i skjemaer. Eksempelvis for å sjekke at antall medlemmer i et nytt prosjekt er et tall og ikke en streng. Dette vises i figuren nedenfor.

```
29 // Validation schema:
30 const createProjectSchema = yup.object().shape({
31   gruppenavn: yup
32     .string()
33     .required("Det må fylles inn et gruppenavn."),
34   antallMedlemmer: yup
35     .number()
36     .typeError("Antallet må være et tall")
37     .positive("Antallet må være et positivt tall.")
38     .integer("Antallet må være et heltall.")
39     .required("Det må fylles inn antall medlemmer i prosjektgruppen.")
40 });
```



Antall forskere

Figur 101 -Validering av innfillingskjema med Yup.

3.3.2.3.8 ADMINISTRASJONGRENSESNIETT

Administrasjonsgrensesnittet er aksesserbart for brukere med rolle attributtet, ADMIN. Kan nås ved å aksessere [API URL]/admin, når brukeren er pålogget i applikasjonen. Grensesnittet bruker løsninger fra modulen React-bootstrap-table2 for visning av tabeller med data fra applikasjonen, samt modulen Recharts for å vise statistikk som grafer (*Recharts*)⁴⁰.

Administrasjonsgrensesnittet er ikke en egen applikasjon, den bruker isteden den samme løsningen som brukergrensesnittet. App.js - se figur 102 router til AdminApp komponenten. Den har ikke en interaktiv lenke som kan trykkes på. I stedet må administrator bruke URL-en url/admin som redirectes til og åpner en administrator landingsside med url /admin/users/0. Grunnen til /0 er at denne variabelen brukes som en parameter for søkeboksene i tabellene. Slik at man kan lenke fra en lærer, til lærerens prosjekter og motsatt.

⁴⁰ Recharts. <http://recharts.org/en-US>

```

<div className={styles.wrapper}>
  <Route path={ADMIN_PATH_ADMIN} exact render={() => (
    <Redirect to={ADMIN_PATH_USERS+'/'}/>
  )}/>
  <Route path={ADMIN_PATH_USERS+'/:handle'} exact component={Users}/>
  <Route path={ADMIN_PATH_PROJECTS+'/:handle'} exact component={Projects}/>
  <Route path={ADMIN_PATH_STATISTICS} exact component={Statistics}/>
</div>

```

Figur 102 - Routing tabell i AContent.js

3.3.2.3.8.1.1 TABELLER

Tabellene samler en del komponenter for å vise tabellene. De bruker Toolkitprovidere *react-bootstrap-table2*. Her sendes dataene man vil lese inn i data parametere, kolonnene, blir definert, samt om tabellen skal kunne søkes på. BootstrapTable tar imot flere felter for hvordan tabellen skal fungere. Da at den skal ha et definert nedtrekksvindu. Pagination er løsningen for hvordan arkene i tabellen kan blas mellom.

Dette er et temmelig utfordrende bibliotek å bruke, da det ikke har eksistert lenge og fordi det er vanskelig å spesifisere hvordan man vil implementere det. (*React-Bootstrap-Table*)⁴¹

```

<ToolkitProvider
  keyField="id"
  data={this.getArray()}
  columns={this.getColumn()}
  search = { {
    defaultSearch: this.getSearch(),
    searchFormatted: true
  } }
  >
  {
    props => (
      <div className={styles.buttonWrapper}>
        <ExportCSVButton className = {styles.ExportButton} { ...props.csvProps }>Eksporter til CSV</ExportCSVButton>
        <ClearSearchButton { ...props.searchProps }
          text="Nullser tall søk" className = {styles.clearButton}/>
        <SearchBar
          className={styles.searchBar}
          { ...props.searchProps }
          delay={500}
          placeholder="Søk i felt..."
        />
      </div>
      <BootstrapTable
        expandRow={this.getExpand()}
        striped
        hover
        pagination={ paginationFactory(options) }
        headerClasses={styles.tableHeader}
        { ...props.baseProps }
      />
    </div>
  )
}
</ToolkitProvider>

```

```

export const ProjectColumns = [
  {
    dataField: 'id',
    text: 'Id',
    sort: true,
    formatter: cell => `Id-${cell}`
  },
  {
    dataField: 'name',
    text: 'Tittel',
    sort: true
  },
  {
    dataField: 'nrOfStudents',
    text: 'Antall studenter',
    sort: true
  },
  {
    dataField: 'school',
    text: 'Skole',
    sort: true
  },
  {
    dataField: 'grade',
    text: 'Trinn',
    sort: true
  },
  {
    dataField: 'teacherId',
    text: 'L-Id',
    sort: true,
    formatter: cell => `L-${cell}`
  },
  {
    dataField: 'begun',
    text: 'Påbegynnet',
    sort: true
  },
  {
    dataField: 'finished',
    text: 'Fullført',
    sort: true
  }
];

```

Figur 103 - t.v. Utdrag fra ProjectList.js. t.h. definering av kolonner fra ProjectTable.js

⁴¹ *React-Bootstrap-Table*. (2019, April). <https://github.com/react-bootstrap-table/react-bootstrap-table2>. Storybook. <https://react-bootstrap-table.github.io/react-bootstrap-table2/storybook/index.html?selectedKind=Welcome>

3.3.2.3.8.1.2 STATISTIKK

For å i statistikk vinduet illustrere dataene, brukes et react komponent kalt Linechart fra Recharts. Dette er en ryddig komponent, med god brukerdokumentasjon. I utgangspunktet fyller man en data modell med lister av tallverdier.

```
getData = props =>{
  let size = (props.statistics.countFinishedMonth.length);
  let data = [];
  for(let i = 0; i < size; i++){
    data.push({name: getMonth(i), Påbegynnt: props.statistics.countCreatedMonth[i],
              Ferdig: this.props.statistics.countFinishedMonth[i]})
  }
  return data;
};

render() {
  return (
    <LineChart width={600} height={300} data={
      this.getData(this.props)
    }
      margin={{top: 5, right: 30, left: 20, bottom: 5}}>
      <XAxis dataKey="name"/>
      <YAxis/>
      <CartesianGrid strokeDasharray="3 3"/>
      <Tooltip/>
      <Legend/>
      <Line type="monotone" dataKey="Påbegynnt" stroke="#8884d8" activeDot={{r: 8}}/>
      <Line type="monotone" dataKey="Ferdig" stroke="#82ca9d"/>
    </LineChart>
  )
}
```

Figur 104 - Visning av bruken av LineChart fra Recharts. I ProjectChart.js

3.3.3 SIKKERHET

Sikkerhetsaspektene i vår webapplikasjon er viktige da vi lagrer personopplysninger og passord i databasen vår. Det er viktig for oss at konfidensialiteten ivaretas der det er viktig at informasjon ikke skal være kjent for andre enn brukeren selv. Det er viktig at integriteten til informasjonen som lagres i databasen vår ikke kan endres av andre enn administrator og brukeren selv. Vi har derfor valgt å benytte oss av Spring Securitys funksjonaliteter, som dekker behovene våre både ved autorisasjon og autentisering.

Autorisasjonsmodellen vi benytter er rolle basert. Noe som vil si at aksess til forskjellige funksjonaliteter i applikasjonen tildeles etter hvilke roller brukeren har. I skrivende stund har applikasjonen to roller, en brukerrolle som samtlige brukere på systemet har samt en administratorrolle som kun administratorene på systemet har. Autorisasjonen gjøres gjennom filtrering av forespørsler hvor header filens autorisasjonsfelt hentes ut. Feltet inneholder en JSON Web Token som inneholder autorisasjonen til brukeren.

Autentiseringen av brukeren gjøres gjennom tofaktorer, hvorav den ene er et passord som brukeren kan og den andre er en kode generert av Google Authenticator, altså en kode generert av noe brukeren har.

3.3.3.1 SPRING SECURITY

Spring Security er et rammeverk som i bunn og grunn inneholder alle basisfunksjonalitetene man behøver for å sikre en webapplikasjon mot uautorisert bruk. Spring Security inneholder både autentiserings- og autoriseringsfunksjonaliteter som er tilstrekkelige for vår applikasjon.

3.3.3.1.1 BCRYPT

BCrypt er en passord-hashfunksjon som ble introdusert i 1999. Den baserer seg på en Blowfish shiffer som er en symmetrisk block shiffer. Den ble utviklet av Niels Procos og David Mazières. *(Dan Arias, 2018)*⁴²

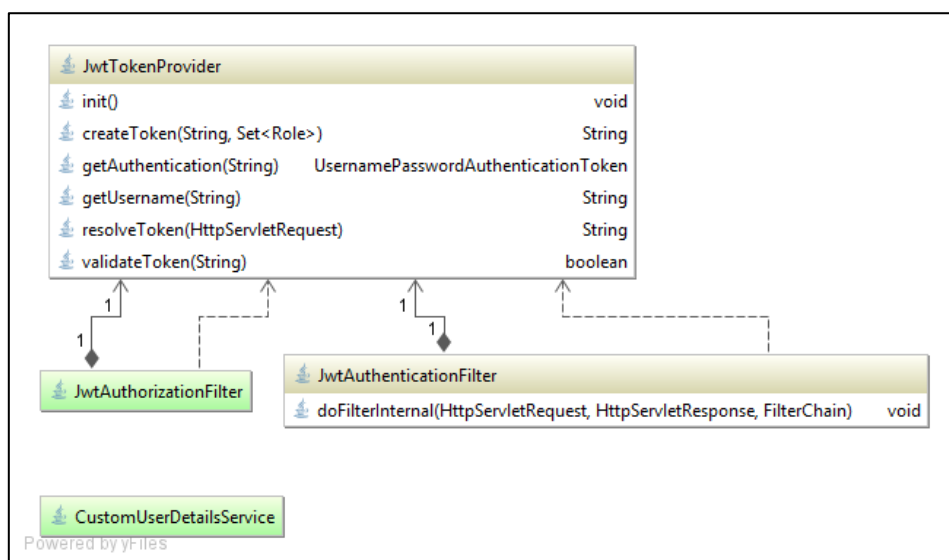
BCrypt er innebygd i Spring Security og er enkel i bruk. Når oppretter en passordhash så lages det både en passordhash og SALT. Et SALT er en tilfeldig generert verdi som forsikrer at to passord som er like ikke vil se like ut når de ligger lagret i databasen. Når man skal kontrollere om passordhashen og klartekstpassordet er like så vil man sende passordhashen og SALTET som parameter inn i algoritmen som vil dekryptere passordhashen til et klartekstpassord. Så sammenlignes det dekrypterte passordet med klartekstpassordet som brukeren tastet inn, dersom de er likeså, blir passordet validert. I BCrypt lagres både passordhashen og SALT-et i samme streng, man slipper dermed å lagre disse i hver sin kolonne i databasen.

⁴² Dan Arias, D. A. (2018, May 31). Hashing in Action: Understanding bcrypt. <https://auth0.com/blog/hashing-in-action-understanding-bcrypt/>

3.3.3.1.2 AUTENTISERINGSFILTER

Spring Security tilbyr mange forskjellige filter som i all hovedsak utfører handlinger basert på forespørsler fra klient. Vi har implementert disse klassene i vår sikkerhetsløsning med tilpasninger som støtter våre behov.

- *JwtAuthenticationFilter* (extends *UsernamePasswordAuthenticationFilter*) – Gjennomfører autentisering gjennom et utfylt skjema sendt fra klient. Skjemaet inneholder da brukernavn og passord, brukernavn i vårt tilfelle er epostadressen til brukeren. Dette er informasjon som brukeren typisk taster inn i brukergrensesnittet frontend. Passordet er et klartekstpassord som blir testet mot passordhashen lagret i databasen.
- *JwtAuthorizationFilter* (extends *BasicAuthenticationFilter*) – Dette filteret tar seg av prosessering av forespørsler. Det tar imot JWT tokenet gjennom «Authorization» headeren på HTTP forespørsler, og nekter/tilbyr adgang basert på det dekodete innholdet. JWT tokenet er kryptert med HMAC256.



Figur 105

```
<dependency>
  <groupId>com.auth0</groupId>
  <artifactId>java-jwt</artifactId>
  <version>3.7.0</version>
</dependency>
```

Figur 106 - Vi benytter dette biblioteket til å generere JWT, som opprettes backend og sendes til frontend i det brukeren autentiserer seg til applikasjonen. (JAVA-JWT.)⁴³

⁴³ JAVA-JWT. <https://github.com/auth0/java-jwt>

3.3.3.1.3 GOOGLE AUTHENTICATOR

Google Authenticator er en tofaktor autentiseringsløsning som leveres av Google. Tjenesten er gratis i bruk og det finnes mange forskjellige biblioteker som er laget for det. Da vi har Maven i backend var det enkelt å implementere et bibliotek for dette. Google Authenticator funksjonaliteten i webapplikasjonen genererer en QR kode basert på en sikkerhetskode som blir generert av Google Authenticator biblioteket. Sikkerhetskoden blir lagret i entiteten til brukeren i databasen. QR-koden leses aktiveres i Google Authenticator applikasjonen til brukeren, men det vil gå an å benytte andre authenticator applikasjoner for dette, for eksempel LastPass Authenticator. Etter at QR-koden er lest inn og brukeren har valgt å gå til neste steg i webapplikasjonen vil han/hun bli bedt om å skrive inn en kode for å verifisere at QR-koden har blitt lest inn på brukerens autentiserings applikasjon. Når dette steget er gjort vil brukeren kunne logge seg på kontoen sin ved å skrive inn både passord og tofaktor autentiseringskode som blir generert av Google Authenticator.

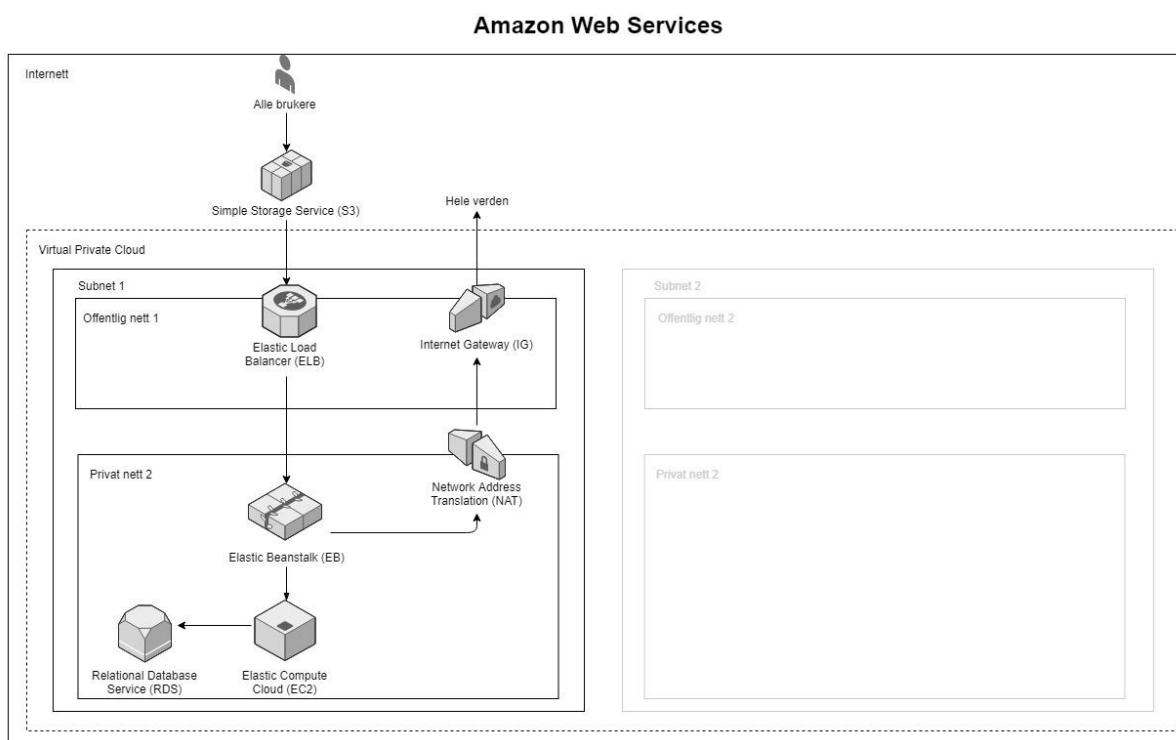
```
<dependency>
  <groupId>com.warrenstrange</groupId>
  <artifactId>googleauth</artifactId>
  <version>1.2.0</version>
</dependency>
```

Figur 107 - Dette biblioteket benyttes til å generere innledende sikkerhetskoder, og til å autentisere brukeren ved hjelp av denne sikkerhetskoden og koden gitt av brukerens autentiseringsapplikasjon. -

3.3.4 SKYLØSNING

Applikasjonen vår er publisert i Amazon Web Services (AWS). AWS er et datterselskap av Amazon som tilbyr en skytjeneste plattform som er priset etter hvor mye tjenestene brukes. Tjenestene de leverer dekker alt det en skulle ha behov for i en tradisjonell webapplikasjon. Oppdragsgiver har høy kompetanse på bruk av tjenestene, derfor var det lett å velge AWS framfor andre tjenesteleverandører. (*About AWS*)⁴⁴

3.3.4.1 ARKITEKTUR



Figur 108 - Illustrasjon av AWS oppsett. IG - nternet Gateway, et routing table som tillater sending av informasjon til og fra subnettene.

API-et vårt ligger på et [Elastic Compute Cloud \(EC2\)](#) instans. Vi benytter et [Elastic Beanstalk](#) for å administrere EC2 instanser, dette gir oss et enkelt grensesnitt for å håndtere miljøet og publisering av API-et. Gjennom Elastic Beanstalk administreres en [Elastic Loadbalancer \(ELB\)](#) som fordeler forespørsler mellom en til flere kjørende applikasjonsserverere. En ELB sitter foran Java-nodene, når den gjør det fungerer den som et endepunkt for brukere av nettsiden, ELB kan dermed delegere brukere mellom de ulike Java-nodene. ELB kan ta ned noder, og erstatte dem med nye dersom de oppfører seg unormalt. Per nå er det ikke satt opp noen form for automatisk skalering basert på last, men dette er noe som kan gjøres ved behov.

⁴⁴ About AWS. <https://aws.amazon.com/about-aws/>

Databasen vår er satt opp på en Relational Database Service. Denne instansen kommuniserer direkte med EC2 instansen som API-et vårt ligger på.

Hele klienten vår ligger i en Simple Storage Service som er satt opp som en statisk hosting node.

3.3.4.2 TJENESTER

Dette delkapittelet introduserer AWS tjenestene vi benyttet oss av.

3.3.4.2.1 AMAZON VIRTUAL PRIVATE CLOUD (VPC)

Er en virtuell sky som tilbys av Amazon som består av et eller flere virtuelle nettverk hvor man kan sette opp alle mulige typer AWS tjenester. Når man setter opp en VPC så blir det automatisk opprettet en Network Address Translation (NAT) som står mellom det private og det offentlige nettverket. Det blir opprettet en Internet gateway (IG) som står mellom det offentlige nettverket og internett. Denne tar seg av rutingen av pakker til og fra det offentlige nettverket. *(Amazon Virtual Private Cloud (VPC))*⁴⁵

3.3.4.2.2 AMAZON ELASTIC BEANSTALK (EB)

Er en tjeneste som tilbys av AWS som benyttes til å samkjøre forskjellige AWS tjenester, blant annet Elastic Compute Cloud(EC2), [CloudWatch](#)(CW), og Elastic Loadbalancer(ELB). *(AWS Elastic Beanstalk Documentation)*⁴⁶

3.3.4.2.3 AMAZON SIMPLE STORAGE SERVICE (S3 BUCKET)

Benyttes til å lagre objekter og filer, og kan settes opp som en statisk hosting node. Klienten vår ligger i en S3 bucket. *(Introduction to Amazon S3 - Amazon Simple Storage Service)*⁴⁷

3.3.4.2.4 AMAZON CLOUDWATCH (CW)

Benyttes til å monitorere og vedlikeholde AWS tjenester. Laget for at utviklere, og systemoperatører skal kunne monitorere og vedlikeholde AWS tjenester. Det er et svært hendig verktøy når man skal feilsøke en AWS-tjeneste, da den muliggjør nedlasting av loggfiler som blant annet kan inneholde logger fra servermotor, applikasjon, og trafikk. Det opprettes automatisk en slik tjeneste på alle instanser man setter opp i AWS. Dette var et svært nyttig verktøy å benytte når vi publiserte API-et vårt. *(Amazon CloudWatch - Application and Infrastructure Monitoring.)*⁴⁸

⁴⁵ Amazon Virtual Private Cloud (VPC). <https://aws.amazon.com/vpc/>

⁴⁶ AWS Elastic Beanstalk Documentation. <https://docs.aws.amazon.com/elastic-beanstalk/index.html>

⁴⁷ Introduction to Amazon S3 - Amazon Simple Storage Service. <https://docs.aws.amazon.com/AmazonS3/latest/dev/Introduction.html>

⁴⁸ Amazon CloudWatch - Application and Infrastructure Monitoring. https://aws.amazon.com/cloudwatch/?sc_channel=PS

3.3.4.2.5 AMAZON ELASTIC LOADBALANCER (ELB)

Kan benyttes til å sende innkommende trafikk til forskjellige tjenester, som for eksempel [Elastic Compute Cloud](#) instanser, eller containere. Vår instans fungerer som en brannmur mot API-et vårt, dvs. at vi med denne har mulighet til å begrense tilgangen til API-et basert på porter og IP-adresser. (*Elastic Load Balancing - Amazon Web Services*)⁴⁹

3.3.4.2.6 AMAZON ELASTIC COMPUTE CLOUD (EC2)

Er en webtjeneste som tilbyr sikker, skalerbar datakraft i skyen. Støtter blant annet kjøring av JAR filer, derfor har vi benyttet et slik instans til API-et vårt. (*Amazon EC2*)⁵⁰

3.3.4.2.7 AMAZON RELATIONAL DATABASE SERVICE (RDS)

Er en distribuert databasetjeneste levert av AWS. Støtter svært mange forskjellige databaser. I vårt tilfelle har vi benyttet en instans som er satt opp med [PostgreSQL](#). (*Amazon Relational Database Service (RDS) – AWS*)⁵¹

⁴⁹ Elastic Load Balancing - Amazon Web Services. <https://aws.amazon.com/elasticloadbalancing/>

⁵⁰ Amazon EC2. <https://aws.amazon.com/ec2/>

⁵¹ Amazon Relational Database Service (RDS) – AWS. <https://aws.amazon.com/rds/>

3.4 PUBLISERING

I dette kapittelet er en instruks på hvordan man publiserer applikasjonen i skyen. Publisering krever at man har påloggingsinformasjonen til Forskningsrådets AWS konto.

3.4.1 PUBLISERING AV BACKEND TIL ELASTIC BEANSTALK INSTANS

Vi laget en instruks til Forskningsrådet med fremgangsmåte for publisering av API-et til Amazon Elastic Beanstalk:

3.4.1.1 PUBLISERING AV API

- Oppdater felter i application.properties fra å benytte lokal database til å benytte RDS. Bildet nedenfor viser filinnholdet med lokalt oppsett (linje 4-6 og 9-12 skal ut, mens 15-17 og 20-23 skal inn)

```
3 #Local
4 spring.datasource.url=jdbc:postgresql://localhost:5432/np_db
5 spring.datasource.username=postgres
6 spring.datasource.password=NetAdmin
7
8 #Migration: Local
9 spring.flyway.baseline-on-migrate=true
10 spring.flyway.url=jdbc:postgresql://localhost:5432/np_db
11 spring.flyway.user=postgres
12 spring.flyway.password=NetAdmin
13
14 #Fields provided by the environmental variable from the server:
15 #spring.datasource.url=jdbc:postgresql://${RDS_HOSTNAME}:${RDS_PORT}/${RDS_DB}
16 #spring.datasource.username=${RDS_USER}
17 #spring.datasource.password=${RDS_PASSWORD}
18
19 #Migration: Fields provided by the environmental variable from the server:
20 #spring.flyway.baseline-on-migrate=true
21 #spring.flyway.url=jdbc:postgresql://${RDS_HOSTNAME}:${RDS_PORT}/${RDS_DB}
22 #spring.flyway.user=${RDS_USER}
23 #spring.flyway.password=${RDS_PASSWORD}
```

Figur 109

```

3 #Local
4 spring.datasource.url=jdbc:postgresql://localhost:5432/np_db
5 spring.datasource.username=postgres
6 spring.datasource.password=NetAdmin
7
8 #Migration: Local
9 spring.flyway.baseline-on-migrate=true
10 spring.flyway.url=jdbc:postgresql://localhost:5432/np_db
11 spring.flyway.user=postgres
12 spring.flyway.password=NetAdmin
13
14 #Fields provided by the environmental variable from the server:
15 #spring.datasource.url=jdbc:postgresql://${RDS_HOSTNAME}:${RDS_PORT}/${RDS_DB}
16 #spring.datasource.username=${RDS_USER}
17 #spring.datasource.password=${RDS_PASSWORD}
18
19 #Migration: Fields provided by the environmental variable from the server:
20 #spring.flyway.baseline-on-migrate=true
21 #spring.flyway.url=jdbc:postgresql://${RDS_HOSTNAME}:${RDS_PORT}/${RDS_DB}
22 #spring.flyway.user=${RDS_USER}
23 #spring.flyway.password=${RDS_PASSWORD}

```

Figur 110

- Kjør kommandoen:

mvn clean package spring-boot:repackage -DskipTests

i [Maven](#) fanen i IntelliJ ved å trykke på knappen:



- Prosjektet vil nå pakkes og lagres til target mappen i prosjektet.
- Logg på
- Last opp "veileder-api-X.X.X-SNAPSHOT" til Elastic Beanstalk via "Dashboard" fanen i konsollen på AWS. Bildet nedenfor viser knappen:

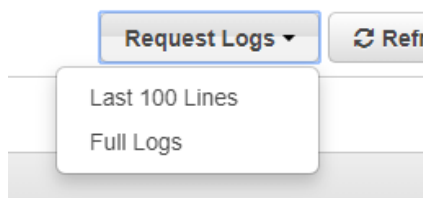
Running Version

veileder-api-0.0.5-SNAPSHOT

Upload and Deploy

3.4.1.2 FEILSØKING AV PUBLISERT API

- Elastic Beanstalk instansen er satt opp med et Cloud Watch instans. Noe som tillater at man kan generere og laste ned log fra serveren. For å benytte seg av denne naviger til Log i konsoll vinduet til Elastic Beanstalk instansen. Klikk så på:



Som regel er de 100 siste linjene tilstrekkelig. Last ned rapporten ved å trykke på "Download", API-ets log kan leses under headeren:

```
-----  
/var/log/web-1.log  
-----
```

3.4.2 PUBLISERING AV FRONTEND TIL SIMPLE STORAGE SERVICE

Vi laget en instruks til Forskningsrådet med fremgangsmåte for publisering av klienten til Amazon S3 Bucket:

3.4.2.1 PUBLISERING KLIENT

1. Last ned AWS CLI og installer.
 - a. Kjør **aws configure** i cmd/Terminal. Legg til AWS ID og Secret til NYS AWS-brukeren.
2. Pass på at np-frontend/np-frontend/src/store/constants/apiUrl.js er satt til aws-API-et og ikke localhost.
3. Kjør **npm run build** for å opprette nyeste den nyeste versjon.
4. Kjør **npm run deploy** for å laste opp på AWS.

3.4.2.2 FEILSØKING AV PUBLISERT KLIENT

- Pass på at react-scripts er på ^2.1.5
- Sjekk at "aws s3 sync build/ s3://np-frontend" ligger under "scripts" i package.json
- Pass på at du ligger på DEPLOYMENT

3.5 KONKLUSJON TIL PRODUKTDOKUMENTASJON

Det har vært et mål å lage ryddig kodearkitektur av høy kvalitet, slik at applikasjonen skal være tilpasset videreutvikling. Dette mener vi at har blitt oppnådd. Det var forholdsvis uproblematisk å utvikle en god backend-arkitektur, da vi brukte kjente designmønstre som vi hadde god kjennskap til, som *model view controller*, DAL og BLL. I frontend var utfordringen derimot langt større, ettersom vi både hadde mindre erfaring med komponentbasert utvikling i tillegg til at det var færre kjente designprinsipper. Dermed ble en del av arkitekturprosessen å finne ut av designmønstre selv, noe som var en verdifull erfaring for utviklerne.

For både kunde og oppdragsgiver vil det dermed være av høy verdi å kunne lettere videreutvikle applikasjonen.

En av hovedutfordringene for applikasjonen var å integrere all teknologien, både å få dem til å funke hver for seg, og med hverandre. Selv om et så høyt antall teknologier kunne vært kaotisk, samhandler de nå godt, og vi klarte å ta dem i bruk på ryddige måter. Dette er verdifullt ikke bare for å bedre selve applikasjonens funksjonalitet, men også som en demonstrasjon for andre om hvordan man kan ta i bruk de forskjellige teknologiene.

Sluttproduktet er stort, og har klart å omfavne et svært stor skop med mange funksjonaliteter.

Den funksjonelle systembeskrivelsen viser at applikasjonen treffer godt på kravspesifikasjonen, som blir diskutert i Prosessdokumentasjonen.

4 TESTRAPPORT

NYSGJERRIGPER – PROSJEKTVEILEDEREN

BACHELORPROSJEKT 2019

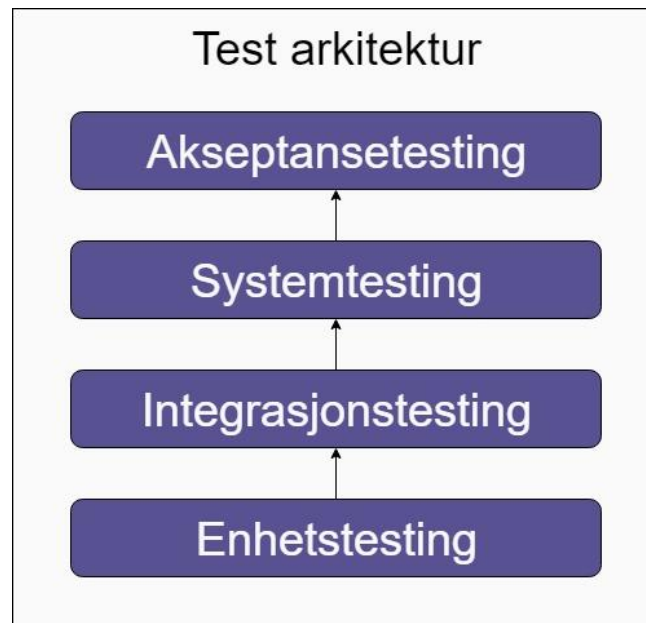
GRUPPE 5

INNHALDSFORTEGNELSE TESTRAPPORT

4.1	INTRODUKSJON.....	125
4.2	ENHETSTESTING	126
4.2.1	Om enhetstesting.....	126
4.2.2	Backend	126
4.2.2.1	Dependency.....	126
4.2.2.2	Testklasseoppsett med Mockito og JUnit	127
4.2.2.3	Beskrivelse av Mockito og JUnit.....	129
	Testdekning	130
4.2.3	Oppsummering.....	131
4.3	INTEGRASJONSTESTING	131
4.3.1	Om integrasjonstesting	131
4.3.2	Integrasjonstesting av backend	132
4.3.2.1	Integrasjonstesting av hele applikasjonen.....	136
4.3.3	Oppsummering.....	136
4.4	SYSTEMTESTING	137
4.4.1	Om systemtesting.....	137
4.4.2	Oppsummering.....	137
4.5	AKSEPTANSETESTING	138
4.5.1	Om akseptansetesting.....	138
4.5.2	Oppsummering.....	138
4.6	BRUKERTESTING	139
4.6.1	Samtykkeskjema	139
4.6.2	Brukertestdokumentet	139
4.6.3	Om brukertestdokumentet	139
4.6.4	Om gjennomførelsen	140
4.6.5	Brukertestene	140
4.6.6	Oppsummering.....	141
4.7	KONKLUSJON TIL TESTRAPPORT.....	141

4.1 INTRODUKSJON

Testing er en sentral del av utviklingsfasen, og målet med testing er å få konkret tilbakemelding på om applikasjonen, dens metoder og funksjoner faktisk gjør det som er tenkt at den skal gjøre. Tester gjør også gjerne at feil synliggjøres, og dermed kan rettes, eller at man har mulighet til å søke dypere ned i feilen ved å utføre mer testing for så å til slutt finne opphavet og rette opp i feilen.



Figur 111 - Illustrerer rekkefølgen de forskjellige test typene normalt blir utført i, fra bunn til topp. ("What is System Testing?" 2019)⁵²

⁵² What is System Testing? Types & Definition with Example (2019). <https://www.guru99.com/system-testing.html>

4.2 ENHETSTESTING

Under enhetstesting tester man programvaren på laveste nivå. Dette innebærer at man tester hver enkelt metode og funksjon i programvaren individuelt. Det er vanlig å teste grenseverdiene som metodene og funksjonene skal kunne håndtere. Ofte vil man prøve å få metodene og funksjonene til å feile for å sjekke at systemet kaster unntak på riktig måte når ulovlige parameter verdier sendes inn. (*“What is Unit Testing?”* 2018) 53

4.2.1 OM ENHETSTESTINGEN

På grunn av arbeidsmengden under dette bachelorprosjektet var det vanskelig å sette av tid til enhetstesting. Dette gjaldt spesielt for frontend, da det var for stor arbeidsmengde for å rekke å implementere enhetstester for klienten. Isteden ble klienten integrasjonstestet mye under hele utviklingsfasen.

Derimot fikk vi tid til å implementere enhetstester for hele API-et. Det kunne dog vært implementert enda flere tester hvor grenseverdiene til metodene ble testet enda grundigere. Men tiden satt begrensninger for hvor godt vi fikk enhetstestet.

4.2.2 BACKEND

Enhetstestene på API-et ble laget ved hjelp av Mockito og JUnit.

Mockito er et open source testrammeverk for Java. Med rammeverket har man mulighet til å mocke objekter og lage automatiserte enhetstester som har hensikt å følge prinsippene rundt testdrevet og adferdsdrevet utvikling. (*Adzic, 2009*)⁵⁴

JUnit er ett enhetstest rammeverk for Java. JUnit har spilt en sentral rolle under utviklingen av test-drevet utvikling, og er ett av mange testrammeverk i samlingen xUnit, hele denne samlingen startet med JUnit. (*JUnit Tutorial*)⁵⁵

4.2.2.1 DEPENDENCY

Mockito og JUnit er i vårt tilfelle integrert i spring-boot-starter-test som er en dependency som er lagt inn i pom.xml filen vår:

⁵³ *What is Unit Testing?* (2018, February 15). <https://www.agilealliance.org/glossary/unit-test>

⁵⁴ Adzic, G. A. (2009, oktober). *Mockito in six easy examples*.

<https://gojko.net/2009/10/23/mockito-in-six-easy-examples/>

⁵⁵ *JUnit Tutorial*. <https://www.tutorialspoint.com/junit/>

```

<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-test</artifactId>
  <version>${spring-boot.version}</version>
  <scope>test</scope>
</dependency>

```

Figur 112 - Dependency som inneholder testrammeverkene vi behøver

4.2.2.2 TESTKLASSEOPPSETT MED MOCKITO OG JUNIT

Vi benytter prosjekt kontroller testklassen som eksempel. Vi måtte annotere klassen med `@RunWith(MockitoJUnitRunner.class)`. JUnit vil med denne annoteringen kjøre testen med klassen som det refereres til istedenfor å kjøre testen med klassen bygd inn i JUnit. Videre viser figuren nedenfor en oversikt over de importerte klassene og bibliotekene som har blitt benyttet i denne testklassen. (*RunWith (JUnit API)*)⁵⁶

```

package com.np.pv.web.controllers;

import com.np.pv.dal.models.Project;
import com.np.pv.dal.repository.ProjectRepository;
import com.np.pv.service.ProjectService;
import com.np.pv.service.SessionService;
import com.np.pv.web.dtos.NewProjectDto;
import com.np.pv.web.dtos.ProjectDto;
import com.np.pv.web.exceptions.ObjectNotDeletedException;
import com.np.pv.web.exceptions.ProjectNotValidatedException;
import com.np.pv.web.validators.ProjectValidator;
import com.np.pv.web.viewtemplates.ProjectTemplate;

import org.junit.Before;
import org.junit.Test;
import org.junit.runner.RunWith;
import org.mockito.InjectMocks;
import org.mockito.Mock;
import org.mockito.junit.MockitoJUnitRunner;
import org.springframework.http.ResponseEntity;

import static org.junit.Assert.*;
import static org.mockito.ArgumentMatchers.*;
import static org.mockito.Mockito.verify;
import static org.mockito.Mockito.when;

@RunWith(MockitoJUnitRunner.class)
public class ProjectControllerTest {

```

Figur 113 - Øvre del av prosjekt kontroller testklassen

Alle instansene som blir injisert i kontrolleren oppretter vi instanser av i klassen og annoterer dem med `@Mock`. Instansen disse mockene skal injiseres i, annoteres med

⁵⁶ *RunWith (JUnit API)*. <http://junit.sourceforge.net/javadoc/org/junit/runner/RunWith.html>

@InjectMocks. Også alle objektene som skal kunne opprettes, returneres osv. oppretter vi instanser av, for å teste nettopp returverdier, grenseverdier osv. Videre har vi laget en metode som kjøres før hver enkelt test eksekveres, nemlig setUp(), denne annoterer vi med @Before.

```
//region SETUP
//-----
@InjectMocks
private ProjectController controller;

@Mock
ProjectService projectService;

@Mock
ProjectRepository projectRepository;

@Mock
SessionService sessionService;

@Mock
ProjectValidator validator;

private final long userId = 4;
private final long projectId = 5;
//Projects
private NewProjectDto newProjectDto;
private Project project;
private ProjectDto projectDto;
private ProjectTemplate projectTemplate;

@Before
public void setUp() {
    newProjectDto = NewProjectDto.builder().build();
    project = new Project();
    project.setId(projectId);
    projectDto = ProjectDto.builder().projectId(5L).build();
    projectTemplate = ProjectTemplate.builder().projectId(6L).build();
}
```

Figur 114 - Attributter og mocker i prosjekt kontroller testklassen

Hver test annoteres med @Test. Vi kommer tilbake til metodene til Mockito og JUnit i neste underkapittel.


```

@Test
public void getActiveProjectShouldReturnNotFoundIfNoActiveProject() {
    when(projectRepository.findActiveProject(anyLong())).thenReturn(null);
    ResponseEntity<ProjectTemplate> response = controller.getActiveProject();

    assertEquals( expected: 404, response.getStatusCode().value());
}

```

Figur 115 - En test i prosjekt kontroller testklassen

4.2.2.3 BESKRIVELSE AV MOCKITO OG JUNIT

When

Metoden when benyttes når man vil at et mocket instans skal returnere en spesifikk verdi når et spesifikt metodekall gjøres, altså når man vil «stubbe» metoder. Hvor en «stubb» er en simulering av oppførselen til en metode.

```
when(projectRepository.findActiveProject(anyLong())).thenReturn(null);
```

Figur 116 – Eksempel på when, tilhører Mockito

Verify

Med verify verifiserer man at et metodekall har blitt kjørt.

```
verify(sessionService).getUserId();
```

Figur 117 – Eksempel på verify, tilhører Mockito

Dersom et metodekall blir kjørt flere ganger kan man sende med ett times parameter i metodekallet.

```
verify(service, times( wantedNumberOfInvocations: 2)).updateByHypothesisId(hypothesis.getId());
```

Figur 118 – Eksempel på verify med times, tilhører Mockito

Spy

Av og til ønsker man å ivareta det meste av originalobjektets egenskaper, og mocke kun en del av det. Mockitos spy funksjonalitet lar oss gjøre det. (Mokkapaty, 2019)⁵⁷

⁵⁷ Mokkapaty, R. M. (2019, April). Mockito Spy Example | Examples Java Code Geeks - 2019. <https://examples.javacodegeeks.com/core-java/mockito/mockito-spy-example/>

```

@Test
public void updateByHypothesisId() {
    Hypothesis hypoSpy = spy(hypothesis);
    when(hypothesisRepository.findOne(hypothesis.getId())) .thenReturn(hypoSpy);
    when(hypoSpy.getProject()) .thenReturn(project);
    service.updateByHypothesisId(hypothesis.getId());
    verify(service).updateProject(project);
}

```

Figur 119 - Eksempel på bruk av spy

Assert

Assert finnes i en hel del varianter, og vi har stort sett benyttet oss av assertEquals, assertTrue, assertFalse. Det disse metodene gjør er å sjekke om det stemmer, at to variabler er like (assertEquals), eller om en returverdi er sann (assertTrue) eller ikke (assertFalse)

```
assertEquals( expected: 404, response.getStatusCode().value() );
```

Figur 120 - Eksempel på assertEquals, tilhører JUnit

TESTDEKNING

Dekningsgrad forklarer hvor mye av koden som er testet. Det vil si at en IDE undersøker om alle utfallene av alle funksjoner testes. Det betyr at hver if eller switch statement skal ha sin egen test for alle mulige utfall. Ved 100% dekningsgrad av en fil, så betyr det at alle metodenes *synlige* utfall har fått en designert test.

Class	Methods	Lines
factories	100%	95%
HypothesisFactory	80%	89%
MethodFactory	100%	100%
PictureFactory	66%	50%
ProjectFactory	100%	100%
RoleFactory	100%	100%
TeacherFactory	75%	96%
UserFactory	100%	100%

Figur 121 - Dekningsgraden av tester for Java filene i fabrikkmappen.

De grønne feltene på venstre side viser at det angitte området i metoden er testet, mens det røde ikke er testet.

```

65 public HypothesisTemplate createTemplate(Hypothesis hypothesis) {
66     String type = null;
67     if (hypothesis.getResultType() != null) {
68         type = hypothesis.getResultType().getConfirmed();
69     }
70     List<MethodTemplate> methodTemplates = null;
71     if (hypothesis.getMethods() != null) {
72         MethodFactory methodFactory = new MethodFactory(pictureService);
73         methodTemplates = methodFactory.createTemplateList(hypothesis.getMethods());
74     }
75
76     return new HypothesisTemplate(
77         hypothesis.getId(),
78         pictureService.getPictureId(hypothesis.getId(), PictureParentType.HYPOTHESIS),
79         hypothesis.getHypothesisNr(),
80         hypothesis.getThesisQuestion(),
81         hypothesis.getResult(),
82         hypothesis.getConclusion(),
83         type,
84         methodTemplates
85     );
86 }
87 }

```

Figur 122 - Utsnitt fra HypothesisFactory

4.2.3 OPPSUMMERING

Vi klarte å avdekke en hel del feil i backend når vi gjennomførte enhetstesting, og dermed kunne vi utbedre disse feilene tidligere enn hva vi ellers hadde klart uten enhetstesting. Dersom applikasjonen hadde vært i en produksjonssetting, kunne i verste fall disse feilene blitt med i en versjonslansering. Det som gjør enhetstesting ekstra attraktivt, er at man tester enkeltmetoder og funksjoner i applikasjonen. Dette vil i mange tilfeller gjøre feilsøking mye lettere, siden det ikke er like mye kode som omhandler den spesifikke funksjonaliteten som testes, i forhold til om man for eksempel oppdaget en feil mens man systemtestet applikasjonen.

4.3 INTEGRASJONSTESTING

Integrasjonstesting handler om å teste at alle de integrerte modulene fungerer sammen som det systemet de til sammen skal utgjøre. Integrasjonstesting som vi utførte ble gjennomført lokalt på hver enkelt maskin.

4.3.1 OM INTEGRASJONSTESTINGEN

Vi startet integrasjonstesting av backend ved hjelp av Swagger, noe som tillatte oss å sende inn data til de forskjellige API-endepunktene våre.

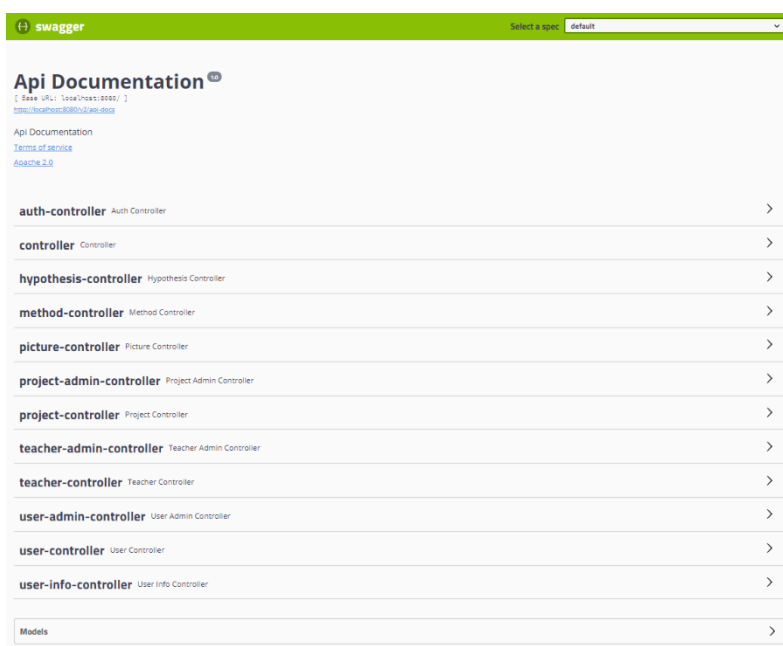
4.3.2 INTEGRASJONSTESTING AV BACKEND

Swagger

Swagger var nyttig for oss tidlig i utviklingsfasen. Swagger trålet alle kontrollerne våre og fant samtlige API-enderpunkter. Den genererte også et brukergrensesnitt med alle endepunktene i, som kunne nås på localhost:8080/swagger-ui.html.

Om Swagger

I denne delen gis en kort forklaring av Swagger. Skjermbildet nedenfor er tatt av brukergrensesnittet vårt i Swagger, som representerer alle endepunktene i API-et vårt.



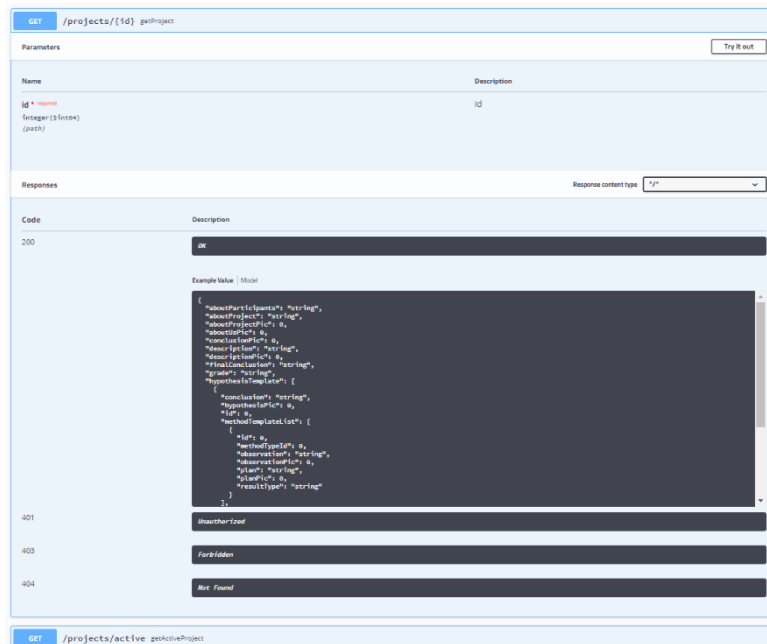
Figur 123 - Brukergrensesnitt Swagger

Navnet på panelet samsvarer med navnet på kontrollerne. Åpner man et panel så vil alle endepunktene til denne kontrolleren synliggjøres.



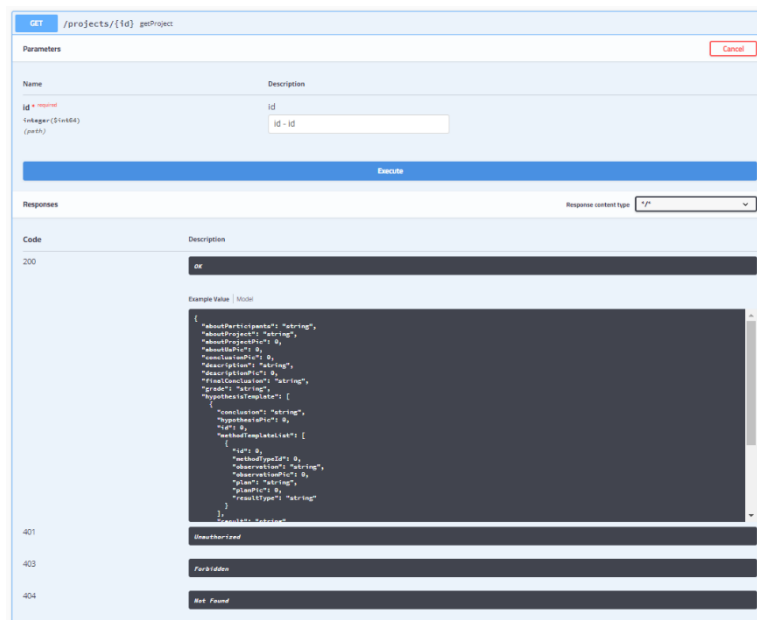
Figur 124 - API-enderpunktene i prosjekt kontrolleren.

Man kan så åpne panelet til endepunktet man vil teste. Endepunktet vil kunne testes ved å trykke på «Try it out».



Figur 125 - API-endepunktet som henter et prosjekt.

Dette endepunktet som henter et prosjektobjekt krever at man sender med en prosjekt-id. Ved å skrive inn en prosjekt-id og trykke på «Execute» vil man få en respons med en status kode lik som en av de vist nedenfor.



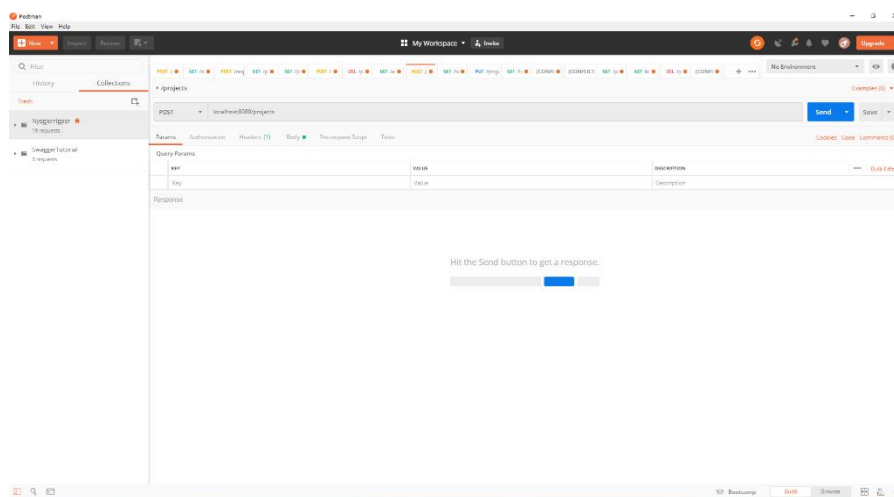
Figur 126 - Testing av kall.

Postman

Da vi hadde implementert autoriseringsløsningen med Json Web Token strakk ikke Swagger helt til, på grunn at vi nå hadde behov for å legge tokenet i autorisasjonsfeltet i headeren på forespørselen for de fleste kall mot API-et etter dette. Derfor begynte vi å benytte Postman, som hadde alle de funksjonalitetene vi hadde behov for under resten av utviklingsfasen. Swagger var likevel kjekt å ha i tillegg, fordi vi kunne bruke parameterlisten som var påkrevd til API-endepunktet fra Swagger grensesnittet, og benytte disse i Postman.

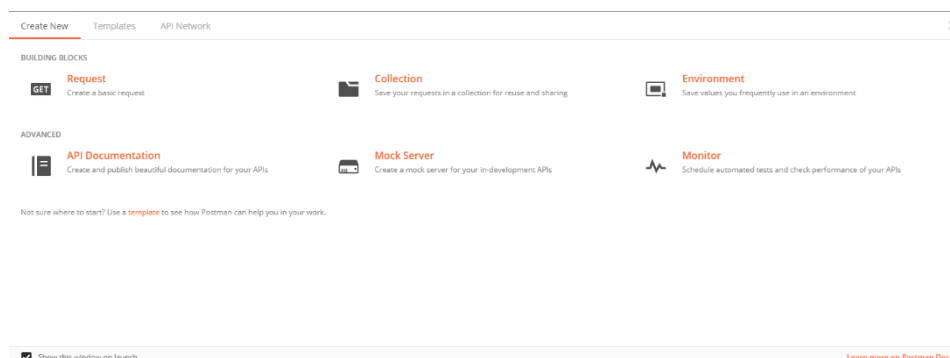
Om Postman

I Postman kan man opprette mapper (collections) med API-kall, og disse mappene kan inneholde enda flere mapper slik at man her kan opprette en struktur tilsvarende som i Swagger.



Figur 127 - Postman

Trykker man på «New» oppe i venstre hjørne får man opp valgene i figuren nedenfor



Figur 128 - Ved å trykke på New i venstre hjørne kommer dette vinduet opp

Ved å trykke på «Request» (forespørsel) kan man lage en ny API-forespørsel. Man får så opp bildet nedenfor, her skriver man inn navnet på forespørselen, en eventuell beskrivelse og man kan velge hvor denne forespørselen skal lagres.

SAVE REQUEST

Requests in Postman are saved in collections (a group of requests).
[Learn more about creating collections](#)

Request name

Request description (Optional)
Adding a description makes your docs better

Descriptions support Markdown

Select a collection or folder to save to:

Search for a collection or folder

All Collections [+ Create Collection](#)

Nysgjerrigper

SwaggerTutorial

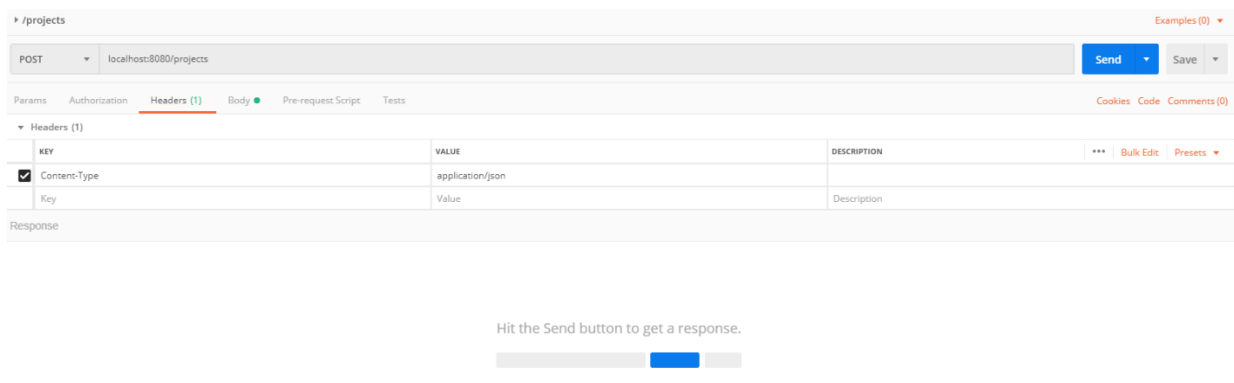
Cancel Save

Figur 129 - Request opprettelsesvinduet

Når forespørselen er opprettet, skriver man inn URL-en til API-endepunktet som skal testes og velge typen forespørsel (POST, GET, PUT, osv.). Videre kan man legge inn et autorisasjonstoken under fanen Authorization, samt alle de nødvendige dataene som er nødvendige for testing av API-endepunktet. Typisk brukte vi Params, Headers og Body fanene mye, i tillegg til Authorization som ble nødvendig for nesten alle forespørslene mot API-et.

Forklaring på de forskjellige fanene vi benyttet i Postman:

- Authorization – er en «bearer token», dvs. et autoriseringstoken generert av backend gjennom autentisering, i vårt tilfelle var dette et JSON Web Token. Enkelt forklart benyttes denne for å få tilgang til API-endepunktene i backend.
- Params – benyttes der parameterverdier sendes mot API-endepunktet gjennom URL-en.
- Headers – er typisk informasjon som forteller endepunktet hva slags format dataene som sendes er i og hva slags format som forventes tilbake. Autorisering blir en del av headeren, likevel har Postman dette som en egen fane.
- Body – kan blant annet være skjema data eller JSON objekter. Generelt er dette dataoverføringsobjekter (DTO-er), som er informasjon som skal prosesseres av backend og/eller lagres i databasen.



Figur 130 - Postman forespørsel

4.3.2.1 INTEGRASJONSTESTING AV HELE APPLIKASJONEN

Integrasjonstesting av hele applikasjonen gjorde vi helt fra vi begynte å koble sammen frontend og backend under sprint 3, og fram til aksept ble gitt av kunden i det akseptansetestperioden var over. Med tanke på at vi gjennomførte såpass mye integrasjonstesting, var det lettere å forsvare at frontend applikasjonen ikke ble enhetstestet grunnet tidsmangel.

Vi la inn en [PostgreSQL](#)-klient og server på hver enkelt maskin som tillatte at vi fikk kjørt en database lokalt hos hver enkelt. Vi klonet og kjørte begge prosjektene, både frontend og backend, lokalt og gjennomførte testing via nettleseren. Dette er mulig i moderne velutviklede nettlesere som Chrome og Firefox. Disse kommer med utviklerverktoy som er integrert i nettleserne. I tillegg til dette lastet vi ned ekstra tillegg til nettleserne som blant annet Redux tillegg som gir utvikleren tilgang til [Store](#) som er lagret på klienten, og et React tillegg som gir utvikleren mulighet til å lese ut parametere (props) og tilstand (state) fra komponentene.

4.3.3 OPPSUMMERING

Integrasjonstesting ga oss mulighet til å finne feil i integrasjonen mellom frontend og backend, samt koblingen mellom backend og den lokale databasen satt opp på maskinene våre. Det fine med integrasjonstesting er at man har hele applikasjonen kjørende lokalt på maskinen, og dermed har utvikleren enkel tilgang til loggfiler som inneholder eventuelle feilmeldinger fra applikasjonen. Dette gjør integrasjonstesting effektivt i forhold til systemtesting som gjerne krever at man kobler seg opp til en ekstern server gjennom en klient eller en nettleser, for så å hente ned loggfilene som inneholder feilmeldingene.

4.4 SYSTEMTESTING

Systemtesting går innunder det man kaller Black Box testing. Ved Black Box tester man applikasjonen og systemet uten at man behøver å ha kjennskap til kildekoden til applikasjonen.

Systemtesting tester den komplette og integrerte programvaren. Systemtesting ble gjennomført med jevne mellomrom i utviklings- og akseptansefasen ved at vi publiserte applikasjonene til skyen og testet de ulike funksjonalitetene fra klienten vår.

(“What is System Testing? Types & Definition with Example”, 2019)⁵⁸

4.4.1 OM SYSTEMTESTINGEN

Vi gjennomførte relativt lite systemtesting før akseptansetestperioden. Da gruppen var liten og det var mye annet å gjøre, hadde systemtesting lett for å bli nedprioritert. Så etter at vi publiserte applikasjonen til skyen for første gang, og alt tilsynelatende fungerte, ble det ikke publisert til skyen på lenge. Det dukket ikke opp problemer senere med tanke på publisering av applikasjonen til skyen, noe som det ville gjort, dersom vi ikke hadde kommet på at et [databasemigreringsverktøy](#) måtte implementeres.

Under akseptansetestperioden ble det gjennomført mye systemtesting. Da fant vi både feil som måtte rettes i kildekoden og en begrensning i selve skyløsningen vår. Begrensningen i skyen var at *nginx* som er webserveren som backend delen vår kjører på, ikke var konfigurert til å akseptere så store forespørsler som bildeopplastning krevde at den taklet. Vi fikset dette ved å konfigurere *nginx* ved hjelp av et skript som vi lastet opp sammen med backend applikasjonen.

4.4.2 OPPSUMMERING

Det er viktig å teste systemet for å sørge for at applikasjonen klarer å kjøre på serveren(e) det er meningen at den skal kjøre på. Man er avhengig av at kompilatorer, eventuelt virtuelle maskiner som applikasjonen skal kjøre på, er compatible. I tillegg må maskinene konfigureres riktig slik at man oppfyller kravspesifikasjonen gitt av prosjekteier.

⁵⁸ *What is System Testing? Types & Definition with Example. (2019, Mars). <https://www.guru99.com/system-testing.html>*

4.5 AKSEPTANSETESTING

For at vi skulle danne oss et realistisk bilde på hvordan det er å jobbe som konsulent, ble det satt opp en akseptansetestperiode for applikasjonen. I akseptansetesten er det sentralt å få testet alle akseptanskriteriene som partene ble enige om at skulle utvikles. Akseptanskriteriene er de kriteriene som settes for at produkteier godkjenner om de enkelte brukerhistoriene er løst. Ofte kan akseptanskriterier endres eller dukker opp underveis, særlig i tilfeller hvor produkteier ikke helt vet hva det er de har behov for. Dette var ikke et unntak for oss, derfor tok vi høyde for også de endringene som var blitt gjort i akseptanskriteriene underveis.

4.5.1 OM AKSEPTANSETESTINGEN

I testperioden gikk produkteier gjennom alle akseptanskriteriene og ga tilbakemelding på feil og eventuelle ønsker om endringer. Siden vi hadde såpass dårlig tid under denne perioden valgte prosjektleder å gi beskjed til produkteier om at feil og ønskede endringer skulle prioriteres. 1. og 2. prioritet ville bli utbedret under testperioden, mens 3. prioritet ville bli utbedret før i en eventuell produksjonssetting, henholdsvis kalt A, B og C feil i tekstdokumentet, hvorav A er 1. prioritet.

Akseptansetesten ligger vedlagt: [Akseptansetest](#)

4.5.2 OPPSUMMERING

Akseptansetestperioden er særlig viktig når man jobber som konsulent. Den gjør det mulig for produkteier å komme med sine ønsker om endringer og eventuelle feil eller mangler ved applikasjonen. I mange tilfeller kommer produkteier med små ønsker om endringer, særlig dersom man har diskutert oppbygningen godt sammen med kunden under utviklingsløpet. I slike tilfeller er akseptansetestperioden en liten pris å betale for å gjøre produkteier fornøyd. Dette var tilfelle for oss. Det ble avholdt demoer av prosjektet for produkteier etter hver sprint. Under disse møtene fikk kunden komme med innspill og ønsker, samtidig som det ble diskutert mye med produkteier underveis i utviklingsfasen.

4.6 BRUKERTESTING

Da applikasjonen vi utviklet, ble utviklet med hensyn til at lærere i barneskolen skal benytte den, var det hensiktsmessig å gjennomføre brukertestene våre på lærere i 1.-7.-trinn. Ideelt sett burde vi gjennomført fem brukertester, da forskning viser at man omtrent vil få like stort utbytte av hver enkelt av brukertestene når man gjennomfører opptil fem brukertester. Med andre ord vil alt over fem brukertester gi lavere utbytte enn de første fem. Likevel ble det gjennomført tre brukertester, på grunn av vanskeligheter med å finne lærere som var villige til å være med på brukertestene, noe som sannsynligvis skyldes mangel på tid. (Nielsen, 1993)⁵⁹

4.6.1 SAMTYKKESKJEMA

Det er hensiktsmessig å signere et samtykkeskjema før en brukertest starter. Et slikt samtykkeskjema inneholder typisk litt informasjon om bakgrunnen til prosjektet, hvordan brukertesten vil foregå, og hvor lang tid den vil ta. Testeren bør selv kunne velge om han/hun ønsker å være anonym, og derfor inneholder samtykkeskjemaet sjekkbokser som gir testeren mulighet til å velge dette før dokumentet signeres.

I rammen nedenfor ser du samtykkeskjemaet som vi laget. Vi fikk en del hjelp fra oppdragsgiver til å lage dette dokumentet, da vi fikk bruke samtykkeskjemaene som de hadde liggende på Confluence som mal.

Samtykkeskjemaet ligger vedlagt: [Informasjon og samtykke](#)

4.6.2 BRUKERTESTDOKUMENTET

Et typisk brukertestdokument inneholder en kort prosjektbeskrivelse, litt informasjon om bakgrunnen til prosjektet, hva målet med applikasjonen er, samt bilder og illustrasjoner som gir en visuell presentasjon av applikasjonen, eventuelt også endringer i forhold til tidligere løsning. Videre finner man informasjon om testleder, og observatører, samt spørsmålene og oppfølgingsspørsmål lenger ned i tekstdokumentet.

4.6.3 OM BRUKERTESTDOKUMENTET

Spørsmålene utarbeidet vi ved hjelp av akseptanskriteriene som vi produserte sammen med oppdragsgiver før vi startet utviklingsprosessen. Det var her viktig å skrive spørsmålene på en slik måte at de ikke veiledet testeren for mye under gjennomføringen. Det var viktig å gjøre de ønskede resultatene målbare der det var

⁵⁹ Nielsen, N. J. (1993). *A mathematical model of the finding of usability problems.*
<https://dl.acm.org/citation.cfm?id=169166>

hensiktsmessig (for eksempel «Testeren bruker under 30 sekunder på å finne brukerinformasjonsiden»), slik at man hadde noe å trekke slutninger etter.

Siden vi hadde tilgang til så god kompetanse hos Netcompany hadde vi mulighet til å sende tekstdokumentet til gjennomgang hos en profesjonell brukertester. Dette er vi svært glade for at vi gjorde, fordi vi fikk en del rettleiding i hvordan spørsmålene burde skrives.

Da applikasjonen vår var på et MVP-stadie valgte vi å gjennomføre brukertesting i en litt mer diskusjonsorientert retning enn normalt. Istedenfor å forholde oss kun til spørsmålene i brukertestdokumentet, søkte vi forslag til forbedringer under hele brukertesten.

Brukertestdokumentet kan leses [her](#).

4.6.4 OM GJENNOMFØRELSEN

Brukertesting ble gjennomført på besøkssiden og prosjektveilederen. Administrasjonsgrensesnittet ble testet av produkteier under akseptansetestperioden, se [Akseptansetesting](#).

Under gjennomførelsen av brukertestene var det viktig at vi ikke rettleidet testerne for mye i riktig retning. Dette for å simulere et mer realistisk scenario. Da testerne sannsynligvis ikke ville hatt en person bak seg som fortalte nøyaktig hva det var han/hun skulle gjøre ved sin egen kontorpult i et realistisk brukstilfelle. (Morrison, 2017)⁶⁰

4.6.5 BRUKERTESTENE

NB! Bildene av det nye brukergrensesnittet som befinner seg i brukertestdokumentet er hentet fra designforslaget som ble gitt til oss av oppdragsgiver ved oppstart.

Fordi brukertestdokumentene er ekvivalente før spørsmålene kommer linker vi her direkte til de interessante delene med tilbakemeldingene.

Brukertestene anbefales å lese og de ligger vedlagt:

[Brukertest – Ragnhild](#)

[Brukertest – Katrine](#)

[Brukertest - Anonym](#)

⁶⁰ Morrison, M. C. (2017, Mars). 10 Simple Steps to Start User Testing Tomorrow. <https://usabilityhour.com/start-user-testing/>

4.6.6 OPPSUMMERING

Lærerne som var med på brukertesting gav oss mange gode innspill på hvordan vi kunne forbedre brukeropplevelsen i prosjektveilederen. Flere av de samme forslagene gikk igjen under brukertesting. Noen av forslagene de kom med ble implementert, men tiden strakk ikke helt til under prosjektet, så vi fikk ikke tid til å drøfte og implementere alle forslagene de kom med. Det var likevel viktig for produkteier at brukertesting ble utført, da resultatene kan brukes ved produksjonssetting.

4.7 KONKLUSJON TIL TESTRAPPORT

Med enhetstestene oppdaget vi feil som var vanskelig å oppdage med det blotte øyet. Disse feilene ville kanskje ellers tatt lenger tid å oppdage, eller i verste fall hadde de ikke blitt oppdaget før produkt lansering. Dette kunne ført til at sluttbrukeren hadde vært den som oppdaget feilen først. Integrasjonstesting var helt sentral under utviklingen. Det er en rask og effektiv måte å teste ut nye funksjonaliteter på, fordi man har tilgang til loggfiler og kildekoden lokalt. Brukertestene hadde spesielt stor verdi for applikasjonen. Vi har fått mange gode forbedringsforslag og vi fikk inntrykk av at dette var en applikasjon som sluttbrukerne hadde svært lyst til å ta i bruk. Dette tjener Forskningsrådet på, siden de har fått et bedre beslutningsgrunnlag for avgjørelsen om de vil satse videre på applikasjonen. Akseptansetesten var en liten investering i forhold til hvor fornøyd produkteier ble med resultatet. Utbedringene vi ble bedt om å implementere var små og enkle.

Selv om man ikke kan teste alle mulige bruksscenarioer som kan forekomme i en applikasjon, er det viktig å gjennomføre testing som utfordrer de kravene som er satt for applikasjonen. Testingsaktivitetene må balanseres opp mot utviklingsarbeidet som er gjort. Det er viktig å huske på at testing skal være til fordel for utviklingen, og ikke må bli en belastning for utviklingsarbeidet. Denne balansen opplevde vi at vi fikk til og vi har derfor samlet oss gode erfaringer med testing som vi vil ta med oss videre.

5 REFERANSELISTE

NYSGJERRIGPER – PROSJEKTVEILEDEREN

BACHELORPROSJEKT 2019

GRUPPE 5

5.1 REFERANSER

- About AWS. Hentet 1. april 2019, fra <https://aws.amazon.com/about-aws/>
- About Node.js Hentet 1. mai 2019, fra <https://nodejs.org/en/about/>
- Abramov, D. Redux Core Concepts. Hentet 3. april 2019, fra <https://redux.js.org/introduction/core-concepts>
- Adzic, G. A. (2009, oktober). Mockito in six easy examples. Hentet 9. mai 2019, fra <https://gojko.net/2009/10/23/mockito-in-six-easy-examples/>
- Amazon CloudWatch - Application and Infrastructure Monitoring. Hentet 8. april 2019, fra https://aws.amazon.com/cloudwatch/?sc_channel=PS
- Amazon EC2. Hentet 1. april 2019, fra <https://aws.amazon.com/ec2/>
- Amazon Relational Database Service (RDS) – AWS. Hentet 5. april 2019, fra <https://aws.amazon.com/rds/>
- Amazon Virtual Private Cloud (VPC). Hentet 1. april 2019, fra <https://aws.amazon.com/vpc/>
- AWS Elastic Beanstalk Documentation. Hentet 1. april 2019, fra <https://docs.aws.amazon.com/elastic-beanstalk/index.html>
- axios. Hentet 23. mai 2019, fra <https://github.com/axios/axios>
- Caitlin Huston, C. H. (2015, 1. desember). 6 things to know about Atlassian as it goes public. Hentet 1. mai 2019, fra <https://www.marketwatch.com/story/6-things-to-know-about-atlassian-before-it-goes-public-2015-12-01>
- Contentful. Contentful developer documentation. Hentet 1. mai 2019, fra <https://www.contentful.com/developers/docs/>
- Dan Arias, D. A. (2018, 31. mai). Hashing in Action: Understanding bcrypt. Hentet 13. mai 2019, fra <https://auth0.com/blog/hashing-in-action-understanding-bcrypt/>
- Design for Kids Based on Their Stage of Physical Development. Hentet 15. februar 2019, fra <https://www.nngroup.com/articles/children-ux-physical-development/>
- Elastic Load Balancing - Amazon Web Services. Hentet 1. april 2019, fra <https://aws.amazon.com/elasticloadbalancing/>
- Facebook Open Source. Create React App. Hentet 22. mai 2019, fra <https://reactjs.org/docs/create-a-new-react-app.html>
- Facebook Open Source. ReactJs Docs. Hentet 22. mai 2019, fra <https://reactjs.org/docs/getting-started.html>

- General Data Protection Regulation (GDPR) (2018, oktober). Hentet 1. mai 2019, fra <https://gdpr-info.eu/>
- Introduction to Amazon S3 - Amazon Simple Storage Service. Hentet 5. april 2019, fra <https://docs.aws.amazon.com/AmazonS3/latest/dev/Introduction.html>
- JAVA-JWT. Hentet 10. april 2019, fra <https://github.com/auth0/java-jwt>
- Jira | Issue & Project Tracking Software | Atlassian. (n.d.-b). Hentet 25 januar 2019, fra <https://www.atlassian.com/software/jira>
- jquense/yup. Hentet 23. mai 2019, from <https://github.com/jquense/yup>
- JUnit Tutorial. Hentet 11. mai 2019, fra <https://www.tutorialspoint.com/junit/>
- Khanana, K., & Law, E. L.-C. (2013). Designing children's digital games on nutrition with playability heuristics. CHI '13 Extended Abstracts on Human Factors in Computing Systems on - CHI EA '13. <https://doi.org/10.1145/2468356.2468548>
- Maven – Introduction. Hentet 19. mars 2019, fra <https://maven.apache.org/what-is-maven.html>
- Modelling – UML Aktivitetsdiagram
Applying UML and Patterns, An Introduction to Object-Oriented Analysis and Design and Iterative Development Third Edition, Craig Larman - Kapittel 28 UML Activity Diagrams and Modeling (s.477-484)
- Mokkaapaty, R. M. (2019, april). Mockito Spy Example | Examples Java Code Geeks - 2019. Hentet 11. mai 2019, fra <https://examples.javacodegeeks.com/core-java/mockito/mockito-spy-example/>
- Morrison, M. C. (2017, mars). 10 Simple Steps to Start User Testing Tomorrow. Hentet 9. mai 2019, fra <https://usabilityhour.com/start-user-testing/>
- Nielsen, N. J. (1993). A mathematical model of the finding of usability problems. Hentet 28. april 2019, fra <https://dl.acm.org/citation.cfm?id=169166>
- Npm. About npm. Hentet 1. mai 2019, fra <https://docs.npmjs.com/about-npm/>
- npm: prop-types. (2019, Februar). Hentet 23. mai 2019, fra <https://www.npmjs.com/package/prop-types>
- Palmer, J. Formik Docs. Hentet 22. mai 2019, fra <https://github.com/jaredpalmer/formik>
- Pankaj, P. (2018, juni). Spring RestController. Hentet 12. mai 2019, fra <https://www.journaldev.com/21536/spring-restcontroller>

- PathVariable (Spring Framework 5.1.7.RELEASE API). (2019, May). Hentet 13. mai 2019, fra <https://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/web/bind/annotation/PathVariable.html>
- Project Lombok. Hentet 1. februar 2019, fra <https://projectlombok.org/features/all>
- Protect REST APIs with Spring Security and JWT. (2018, August 26). Hentet 23. mars 2019, fra <https://medium.com/@hantsy/protect-rest-apis-with-spring-security-and-jwt-5fbc90305cc5>
- Punjal, R. P. (2017, 13. april). Custom Annotation in Java for SQL Injection Safe Parameters. Hentet 10. april 2019, fra <https://dzone.com/articles/custom-annotation-in-java-for-sql-injection-safe-p>
- react-bootstrap. Hentet 23. mai 2019, fra <https://react-bootstrap.github.io/>
- React-Bootstrap-Table. (2019, 15. april). React-bootstrap-table/react-bootstrap-table2. Hentet 3. mars 2019, fra <https://github.com/react-bootstrap-table/react-bootstrap-table2>
- ReactTraining/react-router. Hentet 23. mai 2019, fra <https://github.com/ReactTraining/react-router>
- Recharts. Hentet 7. mars 2019, fra <http://recharts.org/en-US>
- redux-devtools-extension. (n.d.). Retrieved May 23, 2019, from <https://github.com/zalmoxisus/redux-devtools-extension>
- Revelle, G., Zuckerman, O., Druin, A., & Bolas, M. (2005). Tangible user interfaces for children. CHI '05 Extended Abstracts on Human Factors in Computing Systems - CHI '05. <https://doi.org/10.1145/1056808.1057095>
- RunWith (JUnit API). (n.d.). Hentet 18. mai 2019, fra <http://junit.sourceforge.net/javadoc/org/junit/runner/RunWith.html>
- Schwaber, K., & Sutherland, J. (2017). The Scrum Guide™ - The Definitive Guide to Scrum: The Rules of the Game. <https://www.scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-US.pdf>
- Scrum Guide | Scrum Guides. (2017, November). Hentet 13 februar 2019, fra <https://www.scrumguides.org/scrum-guide.html>
- Slik administrer du informasjonskapsler - Nettvett.no. (2019, Januar). Hentet 5. mars 2019, fra <https://nettrett.no/slik-administrer-du-informasjonskapsler/>
- Spring Boot. Hentet 15. mars 2019, fra <https://spring.io/projects/spring-boot>

- Spring Data JPA. Hentet 23. mars 2019, fra <https://spring.io/projects/spring-data-jpa>
- Spring Security. Hentet 30. mars 2019, fra <https://spring.io/projects/spring-security>
- spring.io. Hentet 11. mai 2019, fra <https://spring.io/>
- Storybook. Hentet 3. mars 2019, fra <https://react-bootstrap-table.github.io/react-bootstrap-table2/storybook/index.html?selectedKind=Welcome>
- Team collaboration software & online whiteboard for teams | Miro. Hentet 1. mai 2019, fra <https://miro.com/>
- The digital design toolkit. Hentet 1. mai 2019, fra <https://www.sketch.com/>
- What is an Object-Relational Database (ORD)? - Definition from Techopedia. Hentet 18. mai 2019, fra <https://www.techopedia.com/definition/8714/object-relational-database-ord>
- What is System Testing? Types & Definition with Example. (2019, Mars). Hentet 9. mai 2019, fra <https://www.guru99.com/system-testing.html>
- What is Unit Testing? (2018, 15. februar). Hentet 9. mai 2019, fra <https://www.agilealliance.org/glossary/unit-test>

6 VEDLEGG

NYSGJERRIGPER – PROSJEKTVEILEDEREN
BACHELORPROSJEKT 2019
GRUPPE 5

INNHOLDSFORTEGNELSE VEDLEGG

6.1 KRAVSPESIFIKASJON OG BRUKERHISTORIER.....	149
6.2 ATTESTER.....	164
6.3 BEGREPSAVKLARINGER.....	166
6.4 RISIKORAPPORT	169
6.5 BRUKERVEILEDNING	180
6.6 BRUKERTESTER	195
6.7 AKSEPTANSETEST	236

6.1 KRAVSPESIFIKASJON OG BRUKERHISTORIER

KRAVSPESIFIKASJON

Kravspesifikasjonen har til hensikt å gi leverandør og kunden en oversikt over hva produktet som leveres skal inneholde. Spesifikasjonen er utviklet i samarbeid mellom Netcompany og Forskningsrådet, hvor det er forsøkt å definere og prioritere kundens behov, for så å sette dem sammen til brukerhistorier og systembeskrivelser. Kravspesifikasjonen har utgangspunkt fra konseptfasen for prosjektet.

Skisser utarbeidet under konseptfasen: [Invision](#)

Prosjektets målsetning

Bidra til at flere lærere tar i bruk Nysgjerrigpermetoden.

Applikasjonen har som målsetning å senke terskelen for bruk av Nysgjerrigpermetoden, ved å gi lærere veiledning i bruk av denne, slik at flere barn får lære om forskning og forskningsmetodikk.

Ikke-funksjonelle mål

- Det skal være lett og intuitivt å bruke prosjektveilederen for lærere.
- Det skal oppleves brukervennlig når data blir plottet inn
- Statistikk skal presenteres på en brukervennlig måte for ansatte i Forskningsrådet
- Gi mulighet for diskusjon og argumentasjon i plenum i klasserommet.
- Dokumentasjon underveis for læreren

Funksjonelle behov

Prosjektveileder

Applikasjonen skal ha en prosjektveileder for å utføre et prosjekt med Nysgjerrigpermetoden. Dette er en stegvis guide, hvor læreren, enten alene eller sammen med elever, får hjelp til hvert trinn i Nysgjerrigpermetoden og til å strukturere forskningsprosessen. Resultatet er en kort rapport eller oppsummering. Denne kan brukes som utgangspunkt for å lage en rapport til Nysgjerrigperkonkurransen.

Gå gjennom Nysgjerrigpermetoden:

- Dette lurer jeg på: Skrive inn problemstilling
- Hvorfor er det slik: Skrive inn hypoteser
- Legg en plan: Velge metoder (f.eks. intervju, observasjon) for å teste hypotesene, samt skrive hvordan man skal utføre undersøkelsen.
- Hent opplysninger: Skrive inn informasjon og erfaringer som man får mens man bruker forskningsmetodene. Her bør det være mulig å laste opp bilder
- Dette har jeg funnet ut: Skrive en konklusjon på hver hypotese, og skrive en samlet konklusjon for problemstillingen
- Fortell til andre: Skrive om hvordan det var å gjennomføre prosjektet, samt mulighet for å skrive en introduksjon om klassen. Her bør det være mulig å laste opp bilder.

Administrasjon

Applikasjonen skal ha et administrasjonsgrensesnitt, hvor eier kan administrere brukere og se statistikk. Her er det ønskelig å:

- Slette brukere
- Slette prosjekter
- Se alle prosjekter som er opprettet
- Se tall på hvor mange prosjekter som er ferdigstilt
- Se tall på hvor mange prosjekter som er påbegynt
- Se tall på hvor langt i Nysgjerrigpermetoden et eller alle prosjekt har kommet

Tekniske krav

- Web-applikasjon, slik at man ikke binder seg til en plattform
- Ønske om å tilrettelegge for Smartboard
- Tofaktor for pålogging (minstekrav)
- Enkel oppdatering av tekster og annet innhold ved hjelp av Contentful
- Enkel vedlikehold og drift av tjenesten i produksjon/test- miljø ved hjelp av skytjenester.

BRUKERHISTORIER

6.1.1 AKTØRER I APPLIKASJONEN

- Besøkende
- Administrator
- Lærer
- Prosjekteier

6.1.2 HOVEDTEMA

OBS! Est. er lagt inn i dager. Nytt est. står oppført med dager og timer

Nr	Tema	Som en ...	Ønsker jeg ...	Slik at	Esti mat	Priorite ring
E1	Innlogging og sikkerhet	besøkende	et brukergrensesnitt	jeg kan lage meg en bruker, logge inn sikkert og få tilbakestillt passord om det blir glemt	14 d	PRI 3
E2	Innlogget bruker	lærer/administrator	tilgang på min informasjon	jeg kan se all min brukerinformasjon og endre på den (ikke e-postadresse)	2 d	PRI 2
E3	Administrasjon	administrator	et administratorgrensesnitt med oversikt, statistikk og bruker- og prosjekthåndtering	jeg kan se oversikt over og statistikk for alle prosjekter og har mulighet til å slette prosjekter samt at jeg kan se og administrere brukere	5 d	PRI 1 (E3.1 viktiger enn E3.2)
E4	Prosjektveileder	lærer	en prosjektveileder for Nysgjerrigpermetoden	jeg kan bli veiledet gjennom gjennomføring en av et forskningsprosjekt i henhold til	43 d	PRI 4

				Nysgjerrigperm etoden		
E5	Integrasjon Forsknings rådet	administrator	at applikasjonen integrerer mot tredjepartssyste mer som Forskningsrådet allerede benytter	vedlikehold og oppdatering av for eksempel innhold kan gjøres på en enkel måte.	4 d	
E6	Tekniske krav	prosjekteier	at applikasjonen bruker moderne teknologi	applikasjonen er portabel og drift og vedlikehold er raskt og enkelt	11 d	PRI 3
Su m					79 d = 632 h	

6.1.2.1 LISTE OVER NEDPRIORTERTE BRUKERHISTORIER: (PRIORITERT ETTER REKKEFØLGE DE BØR IMPLEMENTERES)

1. Bruker skal kunne se liste over sine egne prosjekter og laste dem inn etter behov (Spesielt prioritert). **E2***
2. Se egen brukerinformasjon og endre den.
3. Administrator skal kunne se en oversikt over brukere og justere på dem.
4. Bruker skal kunne velge å tilbakestille passord og få tilsendt autogenerated epost.

Kravtabeller

Røde felter er brukerhistorier som måtte prioriteres vekk fra utviklingen

E1 Innlogging og sikkerhet

Nr	Tema	Brukerhistorie	Akseptanskriterie	Nytt estimat
E 1.1	Landingsside	<p>Som besøkende</p> <p>Ønsker jeg en landingsside</p> <p>Slik at jeg kan få en kort introduksjon til Nysgjerrigpermetoden og hva jeg kan gjøre for å komme igang.</p>	<p>Gitt at jeg ikke er innlogget eller ikke har en bruker</p> <p>Når jeg velger å navigere til applikasjonens nettsadresse</p> <p>Så skal applikasjonen vise meg en landingsside med informasjon og mulighet til å komme videre.</p>	1 d
E 1.2	Opprette bruker	<p>Som besøkende</p> <p>Ønsker jeg å kunne opprette en bruker for applikasjonen</p> <p>Slik at jeg kan bruke den i undervisningen.</p>	<p>Gitt at jeg ønsker å opprette bruker</p> <p>Når jeg velger opprett bruker funksjonen</p> <p>Så skal applikasjonen opprette en bruker basert på innfylte data fra brukeren</p>	4 d
E 1.3	Logge inn	<p>Som besøkende</p> <p>Ønsker jeg å kunne logge på min registrerte bruker</p> <p>Slik at jeg kan bruke applikasjonen</p>	<p>Gitt at jeg har en registrert bruker</p> <p>Når jeg velger logg-inn funksjonen</p> <p>Så skal applikasjonen logges på med tofaktor og brukeren får sine definerte autorisasjoner.</p>	6 d
E 1.4	Tilbakestille passord	<p>Som besøkende</p> <p>Ønsker jeg å genere nytt passord</p> <p>Slik at jeg kan endre passord eller lage ett nytt</p>	<p>Gitt at jeg har en bruker og ønsker å få nytt passord</p> <p>Når jeg velger resett-passord funksjonen</p>	3 d

		om jeg har glemt det jeg har.	Så skal applikasjonen sende en epost til brukerregistrert epostadresse med en lenke til vindu for å lage nytt passord	
E 1.5	Bruker	Som en utvikler Ønsker jeg en databasemodell for en bruker Slik at denne er klar når funksjonalitet som behandler brukere produseres	Gitt at jeg er en utvikler Når jeg skal lage funksjonalitet som behandler brukere Så skal databasemodellen for brukere allerede være tilgjengelig	3 d
Sum				14 d

E2 Innlogget bruker

Nr	Tema	Brukerhistorie	Akseptanskriterie	Estimat
E 2.1	Se egen brukerinformasjon	Som lærer Ønsker jeg å kunne se min brukerinformasjon Slik at jeg verifisere at opplysningene er korrekte.	Gitt at jeg har en bruker, Når jeg velger min side funksjonen Så skal applikasjonen vise min brukerinformasjon	2 d
E 2.2	Oppdatere egen brukerinformasjon	Som lærer Ønsker jeg å endre min brukerinformasjon Slik at jeg endre til riktige/oppdaterte opplysninger (bortsett fra e-postadresse)	Gitt at jeg har en registrert bruker og er inne på min side Når jeg velger å endre brukerinformasjon Så skal applikasjonen vise et vindu hvor brukeren kan endre sin brukerinformasjon.	1 d
E 2.3	Slette bruker	Som lærer	Gitt at jeg har en registrert bruker og er inne på min side	0.5 d

		<p>Ønsker jeg å se hvordan jeg skal gå frem for å få slettet brukeren min</p> <p>Slik at jeg kan fjerne den ved ønske</p>	<p>Når jeg ser informasjon om min egen bruker</p> <p>Så skal jeg kunne se hvordan jeg går frem for å få slettet brukeren min</p>	
Sum				2.5 d

E3 Administrasjon

E3.1 Administrasjonsgrensesnitt med oversikt og statistikk

Nr	Tema	Brukerhistorie	Akseptanskriterie	Estimat
E 3.1.1	Administrasjonsgrensesnitt	<p>Som administrator</p> <p>Ønsker jeg et administrasjonsgrensesnitt</p> <p>Slik at jeg kan gjøre administrative oppgaver</p>	<p>Gitt at jeg har administratorrettigheter</p> <p>Når jeg logger inn</p> <p>Så skal applikasjonen vise meg ett administrasjonsgrensesnitt</p>	3 d
E 3.1.2	Se alle opprettede prosjekter (knyttet til bruker)	<p>Som administrator</p> <p>Ønsker jeg å kunne se alle opprettede prosjekter på en bestemt bruker</p> <p>Slik at jeg har mulighet til å se prosjektene som hører til en bestemt person.</p>	<p>Gitt at jeg har administratorrettigheter</p> <p>Når jeg velger brukeren fra administrasjonsgrensesnittet</p> <p>Så skal applikasjonen vise alle prosjektene som er knyttet til denne brukeren.</p>	2 d
E 3.1.3	Se tall over påbegynte, men ikke ferdigstilte prosjekter	<p>Som administrator</p> <p>Ønsker jeg å kunne se tall over påbegynte, men ikke ferdigstilte prosjekter</p>	<p>Gitt at jeg har administratorrettigheter</p> <p>Når jeg velger administratorforsiden</p>	2 d

		Slik at jeg har mulighet til å ha en formening om hvor mange prosjekter som blir påbegynt men ikke ferdigstilt	Så skal applikasjonen vise prosjekter som er påbegynt men ikke ferdigstilt	
E 3.1.4	Se tall over ferdige prosjekter	Som administrator Ønsker jeg å kunne se tall over ferdigstilte prosjekter Slik at jeg har mulighet til å ha en formening om hvor mange prosjekter som blir fullført	Gitt at jeg har administratorrettigheter Når jeg velger administratorforsiden Så skal applikasjonen vise prosjekter som er ferdigstilt	1 d
Sum				8 d

E3.2 Bruker -/Prosjekthåndtering

Nr	Tema	Brukerhistorie	Akseptanskriterie	Estimat
E 3.2.1	Se alle brukere i applikasjonen	Som administrator Ønsker jeg å kunne se alle brukere av applikasjonen Slik at jeg får en oversikt over brukere av applikasjonen	Gitt at jeg er administrator av applikasjonen Når jeg velger vis brukere Så skal applikasjonen vise en liste med alle brukerne av applikasjonen	2 d
E 3.2.2	Slette lærer-brukere (sletter kobling)	Som administrator Ønsker jeg å kunne slette en lærer-bruker Slik at jeg kan håndtere feil med brukere, eller ønske om å fjerne bruker	Gitt at jeg er inne i administratorgrensesnittet og ser alle brukere av applikasjonen. Når jeg velger slett bruker metoden for en valgt bruker	0.5d

			Så skal applikasjonen slette denne brukeren og informasjon om denne fra applikasjonen.	
E 3.2.3	Slett prosjekt for bruker	Som administrator Ønsker jeg liste over prosjekter med mulighet for sletting Slik at jeg kan slette ett prosjekt for en bruker.	Gitt at jeg er i liste over alle påbegynte prosjekter. (3.1.1) Når jeg velger slett prosjekt funksjonen Så skal applikasjonen slette dette prosjektets og dets koblinger fra applikasjonen.	1 d
Sum				3.5 d

E4 Prosjektveileder

E4.1 Bruker -/Prosjekthåndtering

Nr	Tema	Brukerhistorie	Akseptansekriterie	Estimat
E 4.1.1	Opprette nytt prosjekt	Som lærer Ønsker jeg å kunne opprette ett nytt prosjekt Slik at jeg får planlagt og gjennomført Nysgjerrigpermetoden på en god måte.	Gitt at jeg er lærer Når jeg velger opprett nytt prosjekt Så skal applikasjonen starte en ny prosjektveileder og opprette et prosjekt.	3 d
E 4.1.2	Se tips for hvert steg i prosjektverktøyet	Som lærer Ønsker jeg mulighet for veiledning i hvordan hvert steg i Nysgjerrigpermetoden kan gjennomføres Slik at jeg får planlagt og gjennomført Nysgjerrigpermetoden på en god måte.	Gitt at jeg er inne i ett av mine prosjekt i veilederen Når jeg er ved et steg i metoden og velger å vise veiledning for steget Så skal applikasjonen vise tips og veiledning for dette steget.	3.5 d

E 4.1.3	Få introduksjon (kom i gang)	<p>Som lærer</p> <p>Ønsker jeg en introduksjon til Nysgjerrigpermetoden</p> <p>Slik at jeg får kjennskap til prosessen.</p>	<p>Gitt at jeg er lærer</p> <p>Når jeg oppretter ett nytt prosjekt</p> <p>Så skal prosjektveilederen vise en introduksjon til Nysgjerrigpermetoden «kom i gang».</p>	2.5 d
E 4.1.4	Legge til problemstilling (Dette lurer vi på)	<p>Som lærer</p> <p>Ønsker jeg å få informasjon/tips om hva som kan være gode problemstillinger og mulighet til å fylle inn tekst</p> <p>Slik at jeg kan legge til en problemstilling og en beskrivelse av denne.</p>	<p>Gitt at jeg er i prosjektveilederen</p> <p>Når jeg velger steg 1, «Dette lurer vi på» i prosjektveilederen</p> <p>Så skal jeg kunne få veiledning til steget og mulighet til å legge inn problemstilling</p>	0.5 d
E 4.1.5	Legge til hypoteser (Hvorfor er det slik)	<p>Som lærer</p> <p>Ønsker jeg å få informasjon/tips om hva som kan være gode hypoteser, og mulighet til å fylle inn tekst</p> <p>Slik at jeg kan legge til hypoteser</p>	<p>Gitt at jeg er i prosjektveilederen</p> <p>Når jeg velger steg 2, «Hvorfor er det slik?»</p> <p>Så skal jeg kunne få veiledning til steget og mulighet til å føre inn en eller flere hypoteser.</p>	1 d
E 4.1.6	Legge inn metoder for testing av hypoteser (Legg en plan)	<p>Som lærer</p> <p>Ønsker jeg å få informasjon/tips om hvordan de ulike metodene kan benyttes, og ha mulighet til å velge metoder</p> <p>Slik at jeg får lagt en plan for gjennomføringen av undersøkelser</p>	<p>Gitt at jeg er i prosjektveilederen</p> <p>Når jeg velger steg 3, «Legg en plan»</p> <p>Så skal jeg kunne få veiledning til steget og mulighet til å velge en eller flere metoder for testing av hypotesene valgt i steg 2.</p>	5 d
E 4.1.7	Notere gjennomføringer av metodene (Hent opplysninger)	<p>Som lærer</p> <p>Ønsker jeg å ha mulighet til å fylle inn tekst</p> <p>Slik at jeg kan beskrive gjennomføringen av</p>	<p>Gitt at jeg er i prosjektveilederen</p> <p>Når jeg velger steg 4, «Hent opplysninger»</p> <p>Så skal jeg kunne få veiledning til steget og</p>	2 d

		undersøkelsene som skal teste hypotesene.	mulighet til å beskrive gjennomføringen av planen lagt i punkt 3.	
E 4.1.8	Å legge inn konklusjon på om hypotesene stemmer (Dette har vi funnet ut)	<p>Som lærer</p> <p>Ønsker jeg å ha mulighet til å fylle inn tekst og velge resultat</p> <p>Slik at jeg kan konkludere på hypotesene og skrive en endelig konklusjon på problemstillingen for prosjektet.</p>	<p>Gitt at jeg er i prosjektveilederen</p> <p>Når jeg velger steg 5, «Dette har vi funnet ut»</p> <p>Så skal jeg kunne få veiledning til steget og mulighet til å konkludere per hypotese, samt konkludere på problemstillingen på bakgrunn av arbeidet gjort med hypotesene.</p>	2 d
E 4.1.9	Skrive introduksjon om klassen i rapporten og hvordan det var å forske (Fortell til andre)	<p>Som lærer</p> <p>Ønsker jeg tekstfelter for introduksjon av klassen og beskrivelse av prosessen</p> <p>Slik at dette kan legges til i rapporten.</p>	<p>Gitt at jeg er i prosjektveilederen</p> <p>Når jeg velger steg 6, «Fortell til andre»</p> <p>Så skal jeg kunne få veiledning til steget og mulighet til å skrive en introduksjon og en beskrivelse av vår gjennomføring av prosjektet.</p>	0.5 d
E 4.1.10	Se utkast av forskerrapport	<p>Som lærer</p> <p>Ønsker jeg å se et utkast av rapporten (prosjektoppsummeringen)</p> <p>Slik at jeg tilpasse den om nødvendig</p>	<p>Gitt at jeg som lærer har fullført stegene</p> <p>Når jeg velger å se utkast funksjonen</p> <p>Så skal prosjektveilederen vise en presentasjon av prosjektet.</p>	4 d
E 4.1.11	Laste opp bilde	<p>Som lærer</p> <p>Ønsker jeg å kunne legge inn bilder underveis i prosessen</p> <p>Slik at jeg kan bruke beskrivende bilder av elevenes arbeid i rapporten.</p>	<p>Gitt at jeg som lærer har et bilde som jeg vil laste opp</p> <p>Når jeg velger å laste opp bilde til et steg i metoden</p> <p>Så skal det legges inn ett bilde knyttet til</p>	4 d

			stedet jeg velger å laste det opp.	
E 4.1.12	Redigere for utfylt data	<p>Som lærer</p> <p>Ønsker jeg å kunne endre data i utfylte felter</p> <p>Slik at jeg har fleksibilitet i å endre data når jeg gjennomfører prosjektet</p>	<p>Gitt at jeg er i prosjektveilederen</p> <p>Når jeg velger et datafelt som er utfylt</p> <p>Så skal jeg ha mulighet til å redigere teksten som står i feltet og rapporten oppdateres deretter</p>	2 d
E 4.1.13	Slette utfylt data	<p>Som lærer</p> <p>Ønsker jeg å ha mulighet til å slette utfylt data</p> <p>Slik at jeg kan fjerne data som ikke passer inn i prosjektet</p>	<p>Gitt at jeg er i prosjektveilederen</p> <p>Når jeg sletter data fra utfylte felter</p> <p>Så skal rapporten oppdateres deretter</p>	1.5 d
E 4.1.14	Bekreftet innhentet samtykke på bilde	<p>Som lærer</p> <p>Ønsker jeg å bli bedt om at samtykke innhentes før opplasting av bilde</p> <p>Slik at jeg kan bekrefte at samtykke er innhentet.</p>	<p>Gitt at jeg er innlogget</p> <p>Når jeg er i Prosjektveilederen</p> <p>Så skal jeg ha mulighet til å legge til bilder mot at samtykke er innhentet</p>	1 d
E 4.1.15	Slette opplastet innhold	<p>Som lærer</p> <p>Ønsker jeg å ha mulighet til å slette opplastet innhold</p> <p>Slik at jeg kan fjerne innhold fra et påbegynt prosjekt</p>	<p>Gitt at jeg er i prosjektveilederen</p> <p>Når jeg underveis i et prosjekt</p> <p>Så skal jeg ha mulighet til å slette innhold fra et påbegynt prosjekt og rapporten oppdateres deretter</p>	1.5 d
E 4.1.16	Laste ned generert rapport som PDF (Prioritert kan krav)	<p>Som lærer</p> <p>Ønsker jeg å kunne laste ned en PDF av prosjektet</p> <p>Slik at jeg jeg kan ha et utgangspunkt til rapport som kan sendes inn til konkurransen og/eller</p>	<p>Gitt at jeg er i prosjektveilederen</p> <p>Når jeg har kommet til "Fortell til andre"</p> <p>Så skal jeg ha mulighet til å laste ned prosjektet som PDF</p>	2 d

		benytte prosjektet utenfor prosjektverktøyet		
E 4.1.17	Utgangspunkt for applikasjonen	<p>Som en utvikler</p> <p>Ønsker jeg et funksjonelt skjelett av applikasjonen</p> <p>Slik at videre utvikling blir godt strukturert</p>	<p>Gitt at jeg er en utvikler</p> <p>Når jeg skal lage ønsket funksjonalitet for applikasjonen</p> <p>Så skal stedet i applikasjonen der det skal inn allerede være opprettet.</p>	5 d
E 4.1.18	Databasemodell for prosjekt	<p>Som en utvikler</p> <p>Ønsker jeg en databasemodell for et prosjekt</p> <p>Slik at det er klart for videre utvikling av funksjonalitet rundt prosjektene</p>	<p>Gitt at jeg er en utvikler</p> <p>Når jeg skal lage funksjonalitet som omhandler prosjekter</p> <p>Så skal databasemodellen allerede være tilgjengelig</p>	1 d
Sum				40 d

E4.2 Prosjektveilederen skal ha en prosjektoversikt

Nr	Tema	Brukerhistorie	Akseptansekriterie	Estimat
E 4.2.1	Se egne eksisterende prosjekter	<p>Som lærer</p> <p>Ønsker jeg å kunne se en oversikt over egne prosjekter</p> <p>Slik at jeg har mulighet til å se egne prosjekter og det dermed er enklere å huske påbegynte prosjekter</p>	<p>Gitt at jeg har bruker</p> <p>Når jeg velger egne prosjekter</p> <p>Så skal applikasjonen vise alle prosjektene som er knyttet til min bruker</p>	2 d
E 4.2.2	Åpne eget eksisterende prosjekt	<p>Som lærer</p> <p>Ønsker jeg å kunne åpne eget eksisterende prosjekt</p> <p>Slik at jeg kan ha flere prosjekter i gang samtidig</p>	<p>Gitt at jeg har bruker</p> <p>Når jeg velger egne prosjekter</p> <p>Så skal applikasjonen åpne det valgte prosjektet</p>	1 d

Sum				3 d
------------	--	--	--	------------

E5 Integrering Forskningsrådet

Nr	Tema	Brukerhistorie	Akseptansekristerie	Estimat
E 5.1	Oppdatere ledetekst	<p>Som administrator</p> <p>Ønsker jeg at prosjektveilederen implementerer Contentful</p> <p>Slik at jeg har mulighet til å endre innhold uten å måtte endre tekst direkte i kildekoden.</p>	<p>Gitt at jeg har tilgang til Contentful</p> <p>Når jeg går inn på Nysgjerrigper spacet i Contentful</p> <p>Så skal jeg har mulighet til å endre ledeteksten i Prosjektveilederen.</p>	4 d
Sum				4 d

E6 Tekniske krav

Nr	Tema	Brukerhistorie	Akseptansekristerie	Estimat
E 6.1	Benyttet teknologi	<p>Som prosjekteier</p> <p>Ønsker jeg at applikasjonen bruker moderne teknologi tilsvarende lignende løsninger hos Forskningsrådet</p> <p>Slik at eksisterene kompetanse kan benyttes og vedlikehold og drift går raskt og er enkelt.</p>	<p>Gitt at jeg er prosjekteier</p> <p>Når jeg vil ha utført vedlikehold og drift av applikasjonen</p> <p>Så skal det gjøres med moderne og kjente verktøy.</p>	5 d
E 6.2	Hosting	<p>Som prosjekteier</p> <p>Ønsker jeg at applikasjonen kan ligge i skyen</p> <p>Slik at alle kan nå den fra en nettleser.</p>	<p>Gitt at jeg er prosjekteier</p> <p>Når jeg vil få tilgang til applikasjonen</p> <p>Så skal det være mulig å ha tilgang til hele applikasjonen gjennom en nettleser.</p>	6 d

E 6.3	Autogenerert e-post	Som utvikler Ønsker jeg at applikasjonen skal sende autogenerert e-post Slik at resetting av passord skal være brukervennlig	Gitt at jeg er en lærer med en forhånds-eksisterende konto Når jeg velger å få mitt passord tilbakestilt Så skal jeg få et automatisk e-post som lar meg gjøre dette.	4 d
Sum				11 d

6.2 ATTESTER

netcompany

Netcompany Norway AS Henrik Ibsens gate 20 NO-0255 Oslo
Telefon: +47 24131730 kontakt@netcompany.com www.netcompany.com

Til den det måtte angå

Oslo, 22.05.2019

Attest

Vegard Skullestad Bjørkegård, Thomas Dahl, Olav Sørli og Yanislav Zagorov har våren 2019 gjennomført et prosjekt i samarbeid med Netcompany og Nysgjerriger ved Norges Forskningsråd.

Netcompany er et internasjonalt it-konsulentselskap med over 2000 ansatte. Vi leverer morgendagens løsninger i det digitale landskapet, og vår grunntanke er å jobbe sammen for å nå våre mål.

Nysgjerriger er Forskningsrådets tilbud til barn og unge, og spesielt for elever og lærere i barneskolen. Fremtidig forskning er avhengig av at barna som vokser opp i dag er nysgjerrige og ønsker å utforske ting. I den forbindelse har Forskningsrådet og Nysgjerriger-teamet ønsket å utforske hvilke muligheter man har ved et studentprosjekt.

Prosjektet startet med en konseptualiseringsperiode hvor en designer fra Netcompany kartla muligheter og konsepter. Studentene har selv vært med på å forme hvilken retning projektet skulle gå, samt scope omfanget. Resultatet av prosjektet er "Nysgjerriger prosjektveileder"; en webapplikasjon som gir en praktisk gjennomgang av hvordan man skal utføre et forskningsprosjekt med Nysgjerrigermetoden. Prosjektveilederen er i første omgang rettet mot lærere for å senke terskelen slik at det blir enklere å gå i gang med Nysgjerrigerprosjekt, samt skape mer glede og interesse rundt forskning og undring for barn fra 6 - 13 år.

Jeg har hatt gleden av å være prosjektleder for studentene og samarbeide med de fra oppstarten i januar. Forventningene ved oppstart var at studentene skulle opptre profesjonelt ute hos kunde, utøve høy faglig integritet og evne å be om hjelp for å sikre fremdrift. Det var også forventet at de skulle bruke de ressursene vi hadde avsatt til prosjektet på en effektiv og ryddig måte. Studentene leverte på disse forventningene på en svært god måte, og de har vist stor grad av selvstendighet og lærevilje i prosjektet. De har levert en løsning som både Netcompany og Norges Forskningsråd er meget godt fornøyd med. Jeg vil benytte anledningen til å rose samtlige deltakere i prosjektet i skyene og gir de varmeste anbefalinger videre.

Med vennlig hilsen

Netcompany

Kirstine Gyving

Prosjektleder

Tlf: +47 468 51 173, e-post: kirstg@netcompany.com



Til den det måtte angå

Vår saksbehandler/tlf.

Vår ref.

Oslo,
21.05.2019

Deres ref.

Tilbakemelding etter studentprosjekt for Nysgjerriger og Forskningsrådet

Vi har hatt gleden av å samarbeide med Thomas Dahll, Olav Sørлие, Vegard Skullestad Bjørkegård og Yanislav Zagorov om en ny digital prosjektveileder for Nysgjerrigermetoden – en vitenskapelig arbeidsmetode for barneskolen. Studentprosjektet ble opprettet via Netcompany.

Det har vært spennende å samarbeide med studentene om det nye verktøyet for Nysgjerrigermetoden. Vi opplever at de har tilført nye perspektiver til vårt arbeid, og dette har vært lærerikt for alle parter. Studentene har også jobbet selvstendig, satt seg inn i Nysgjerrigers tilbud og løst bestillingen på en svært god måte. Vi har hatt full tillit til at de har løst det tekniske på en ansvarlig måte, og i tillegg til at de har møtt våre brukere på en god måte.

Fra et kundeperspektiv har vi også opplevd samarbeidet som profesjonelt og proft gjennomført. Leveranser har kommet som planlagt og etter avtalt omfang.

I tillegg har det vært både hyggelig og positivt å omgås studentgruppen og vi ønsker dem lykke til videre.

Med vennlig hilsen
Norges forskningsråd

Marit Møllhausen
Rådgiver
Område for kommunikasjon

Norges forskningsråd/
The Research Council of Norway
Drammensveien 288
Postboks 564
NO-1327 Lysaker

Telefon +47 22 03 70 00
post@forskningsradet.no
www.forskningsradet.no
Org.nr. 970141669

All post og e-post som inngår i saksbehandlingen, bes adressert til Norges forskningsråd og ikke til enkeltpersoner.

Kindly address all mail and e-mail to the Research Council of Norway, not to individual staff.

6.3 BEGREPSAVKLARINGER

Dette tilleggsdokumentet skal være en referansetabell for flere ord, uttrykk, konsepter og forkortelser som har blitt brukt i prosess- og produktdokumentasjonen.

@PathVariable	Annotering i Spring Framework indikerer at ett metodeparameter blir sendt inn som en URI variabel. (definisjon docs.spring.io)
Action	En action er et objekt som representerer en intensjon om å endre applikasjonens tilstand/state. En action må ha en type, kalt en action-type, og kan ellers ha det innholdet man ønsker. I de fleste tilfeller er dette innholdet variablene man ønsker å endre, og verdiene man ønsker å endre de til.
API	Application Programming Interface, er et sett med klart definerte metoder for kommunikasjon mellom forskjellige komponenter (definisjon Wikipedia). API-et vi snakker om i oppgaven vår er backend grensesnittet.
Autentisering	Handler om å bekrefte en brukers identitet.
Autorisering	Handler om å hva man gir brukeren tilgang til av ressurser.
Backend	Er den delen av programvaren som ligger nærmest databasen der dataene er lagret. Det er også her de tunge kalkuleringsoperasjonene skjer, som brukeren av systemet ikke nødvendigvis har noe forhold til. Det varierer fra program til program hvor stor backend er i forhold til frontend. <i>Granevang, Merethe. (2015, 19. oktober). Backend. I Store norske leksikon. Hentet 8. mai 2019 fra https://snl.no/Backend</i>
Backlog	En liste med Oppgaver/Tasks og brukerhistorier i Jira programmet. Hvor disse er beskrevet og estimert i forhold til tid. Hver gang en sprint startes, plukkes et utvalg fra denne liste ut som oppgaver som skal utføres.
Branch - Gren	Å lage en forgreining utfra en grein i et repository. For eksempel: når vi skal sette i gang en brukerhistorie i sprinten, så lager vi en utgreining fra en development grein og lager en ny grein for å jobbe med bruker historien. Denne flettes inn «merges*» med development når oppgaven er ferdig.
Build	En samling av brukerhistorier som resulterer en samlet versjon av applikasjonen.
Database-migrering	Databasemigrering må gjøres på en database når det er nødvendig å oppdatere eller reversere databaseskjema til en ny eller eldre versjon. Hvor databaseskjema er definisjonen av en relasjonsdatabase.

Design-pattern	Designmønster
DTO	Data Transfer Object, er et object som frakter data mellom prosesser (definisjon Wikipedia). Vi benytter slike objekter når vi sender data fra klient til API, dataene som sendes mot API-et valideres, slik at vi er kan forsikre oss om at verdiene som blir mottatt er innenfor de kriteriene vi setter.
Frontend	Begrepet frontend er som oftest brukt i forhold til webprogrammering, men er et paraplybegrep for all teknologi som blir direkte brukt av sluttbrukeren, da disse typisk er brukergrensesnitt. JavaScript, JSON og XML er eksempler på tradisjonell frontend.
Full stack	En full stack webutvikler er en person som kan utvikle både klient og server software (definisjon w3schools.com).
IDE	Fra engelsk «Integrated Development Environment»
JPA	En Javaspesifikasjon som beskriver håndteringen av relasjonsdata i Java applikasjoner. Det er bindeleddet mellom programkoden og JDBC (SQL) laget.
JSON	JavaScript Object Notation (forkortet til JSON) er en filarkitektur for organisering av informasjon i et lettleselig og relativt kompakt filformat. JSON-filer kan leses av
JWT	JSON Web Token, er en åpen JSON-basert standard RFC 7519 som benyttes til å opprette aksess tokens som inneholder et sett med påstander som kan bekrefte en bruker, eller en maskins identitet.
Load balancer	Hensikten med en loadbalancer er å distribuere arbeidslasten på flere maskiner bedre. Brukt i serverarkitektur hvor man har mer enn en server som håndterer forespørsler. Den fungerer som et knytestpunkt mellom serverne og omverdenen, alle forespørsler går via denne og fordeles til serverne basert på en algoritme.
Middelware	Rammeverk som gjør oppgaver enn hver applikasjon ville ha utført. Slik at man slipper å kode de samme delene igjen og igjen. Eksempler getters og setters kan autogeneres i Lombok. Det gjør at utviklere kan fokusere på den spesifikke jobben de vil at applikasjonen skal gjøre.
MVP	Fra engelsk « <i>Minimum Viable Product</i> », det grunnleggende produktet som kan løse oppgavebeskrivelsen uten at man har med all opsjonell funksjonalitet. Målet med en MVP er å teste ut konsepter og samle informasjon fra f.eks. produkteieren.
Open source	Programvare som tillater fri bruk og modifikasjon av sin kodebase kan klassifiseres som open source. Disse programvarene driftes som regel av frivillige bidragsyttere.

Plugin	Utvidelse til applikasjoner. For eksempel, har IntelliJ en rekke plugins for bedre integrasjon med databaser eller versjonshåndteringsverktøy.
Pull Request	
Reducer	Reduceren i Redux sin store er en funksjon som har som oppgave å kalkulere en ny state på bakgrunn av den foregående og en action . En Redux-store skal kun ha en reducer, men denne er typisk satt sammen av mange mindre reducere. Reducerene i vår applikasjon er typisk «switch-statements» over action-types.
Repository	Del av versjonshåndtering. Er en datastruktur som lagrer metadata. Datastrukturen befinner seg på en server og blir klonet til brukeren sin pc. Repositoryet inneholder endringslogg for filer. https://git-scm.com/book/en/v2/Getting-Started-About-Version-Control
Scope	Brukes om omfanget av prosjektet. Begrep som beskriver hvor mye eller lite funksjonalitet man skal produsere. Ofte sier man at man må holdet seg til scopet for ikke å bruke tid på unødvendige oppgaver.
State	Beskriver samlingen av alle variabler og verdiene deres, som applikasjonens tilstand, som Redux til enhver tid har lagret i sin store .
Store	Redux sin store er et objekt som holder på applikasjonens state . En applikasjon bør bare ha en store. Denne har en dispatch funksjon som brukes for å sende actions til storens reducer .
VCS	Fra engelsk « <i>Version Control System</i> », altså programvare brukt for versjonshåndtering. Den mest kjente og brukte formen for VCS i dag er git, som benyttes av flere industristandarder verktøy som GitHub og BitBucket.
View template	Er et presentasjonsobjekt som inneholder informasjon som skal vises til sluttbrukeren. Fra API til klient benyttes disse, som for så vidt gjør det samme som en DTO, men denne veien gjør vi ingen validering på dataene som sendes.

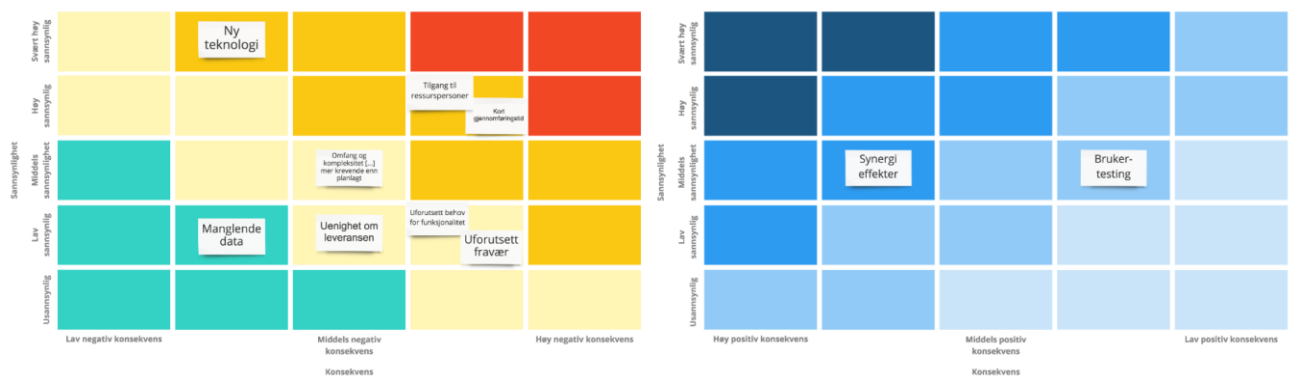
6.4 RISIKORAPPORT

Risikorapport

Sannsynlighet (S): Aktuell bedømmelse av sannsynligheten for at hendelsen inntreffer dersom planlagte tiltak ikke gjennomføres

Konsekvenser (K): Bedømmelse av konsekvensene om hendelsen inntreffer

Kode(S)	Sannsynligheter	Kode(K)	Konsekvenser
5 – Svært høy sannsynlighet	Sikker hendelse (mer enn 90 % sannsynlighet)	5 – Kritisk konsekvens	Hele prosjektet er i fare, målene ikke oppnås
4 – Høy sannsynlighet	Høy sannsynlighet (60 – 90 %)	4 – Høy konsekvens	Hele prosjektets planer må gjøres om, usikkert om målene nås
3 – Middels sannsynlighet	Middels sannsynlighet (30 - 60 %)	3 – Middels konsekvens	Hele prosjektets planer påvirkes, men totalrammen for prosjektet kan holdes
2 – Lav sannsynlighet	Lav sannsynlighet (10 – 30 %)	2 – Mindre farlig konsekvens	Delplaner for prosjektet påvirkes, men totalrammen holdes
1 – Usannsynlig	Usannsynlig (mindre enn 10 %)	1 – Ufarlig konsekvens	Begrenset innvirkning, kan innhentes



Risikofaktorer

- Risikofaktorer er kartlagt og vil gjennomgås ved første statusmøte.

Aktuelle risikofaktorer

Nr.	Beskrivelse	Sannsynlighet	Konsekvens	Tiltak	Ansvarlig	Status per 03/03
RF 01	<p>Tilgang til ressurspersoner</p> <p>For å sikre at løsningen som utvikles får tilfredsstillende funksjonalitet og kvalitet betinges det god deltakelse fra fagpersoner innen systemområdet. Mangelfull tilgang til ressurspersonell, manglende eller forsinkede avklaringer i forbindelse med spesifikasjons- og utviklingsarbeid, vil kunne forsinke fremdriften i prosjektet.</p>	4 – Høy sannsynlighet	4 – Høy konsekvens	<p>Tidlig identifisere og avsette faktisk tid for ressurspersonell</p> <p>Hvem kan svare på hva?</p> <p>Hvem er tilgjengelig når?</p> <p>Sette opp kommunikasjonskanaler for hurtig kommunikasjon</p> <p>Microsoft Teams</p> <p>Slack</p> <p>Lage belegningsplan for å få oversikt over tilgjengelige ressurser og forventet bruk av tid</p>	Leverandør Kunde	Ok.
RF 02	<p>Kort gjennomførings tid</p> <p>Det ønskes mye funksjonalitet på relativt kort kalendertid. Små forsinkelser pga. uforutsett merarbeid eller annet får direkte konsekvenser fordi prosjektet</p>	4 – Høy sannsynlighet	4 – Høy konsekvens	<p>Bevissthet om å lage det som er avtalt i fastpris</p> <p>Sette av tilstrekkelig tid til avklaringer og presiseringer i løsningsbeskrivelsesfasen</p> <p>Revidere prosjektplanene jevnlig mot forbrukt tid og levert funksjonalitet</p>	Leverandør Kunde	Inntruffet. Vi ser at det har vært nødvendig å omprioritere brukerhistoriene. Dette er gjennomgått med Forsknings

	<p>har kort kalendertid og man vil få utfordringer med å hente seg inn igjen. Det anbefales ikke å sette på flere utviklere i samme kalendertid som følge av avtalt øvre kostnadsramme og effektivitet per time.</p>			<p>Kommunisere godt slik at forventningene er realistiske</p> <p>Jobbe kontinuerlig med å prioritere funksjonalitet slik at man kan fokusere arbeidet på det som er viktigst for å få en fungerende løsning.</p> <p>Det er Netcompany sin oppgave å kontinuerlig rapportere om fremdrift og status. Det må avklares med Kunden hva som inngår i en endelig leveranse.</p>		<p>rådet i sprint 2.</p>
RF 05	<p>Uforutsett fravær</p> <p>Sykdom over lengre perioder kombinert med knappe tidsfrister kan hemme fremdrift av prosjektet. Prosjektet er spesielt sårbart da det er et studentprosjekt.</p>	2 – Lav sannsynlighet	4 – Høy konsekvens	<p>Studentgruppen er ansvarlig for å kommunisere med prosjektleder i forhold til mangel av arbeidskraft. Ved sykdom og uforutsett fravær for allokerte ressurser, skal Netcompany eksalere dette internt. Det vil bli undersøkt internt om det finnes ressurser som eventuelt kan avlaste prosjektet, samt andre løsninger.</p>	Leverandør	<p>Inntruffet. Det har vært en del sykdom i sprint 2.</p>
RF 07	<p>Ny teknologi</p>	5 – Svært høy	2 – Mindre farlig	<p>Sette av god tid til oppsett av</p>	Leverandør	<p>Ny teknologi gjør at ting</p>

	Det benyttes ny teknologi og plattform for de fleste i utviklingsteamet, som kan påvirke fremgangen til teamet. Det er satt av tid til å lære seg ny teknologi, men det vil være en risiko for at det vil ta lenger tid enn antatt.	sannsynlighet	konsekvens	utviklingsmiljø og plattform Kunden har godkjent den tekniske plattformen i eget arkitekturforum Utviklere har fått kursing og gjennomgang av nøkkelresurser hos leverandøren. Det er avsatt nøkkelpersonell hos leverandøren. I startfasen vil det settes ekstra ressurser på opplæring og oppfølging som sikrer kvalitet og fremdrift i arbeidet.		tar tid. Oppsett av AWS har vært krevende, men vi har tilgang til ressurspersoner som har hjulpet med dette.
--	---	---------------	------------	---	--	---

Alle risikofaktorer

Nr.	Beskrivelse	Sannsynlighet	Konsekvens	Tiltak	Ansvarlig	Status per 03/03
1	Tilgang til ressurspersoner For å sikre at løsningen som utvikles får tilfredsstillende funksjonalitet og kvalitet betinges det god deltakelse fra fagpersoner innen systemområdet. Mangelfull tilgang til ressurspersonell, manglende	4 – Høy sannsynlighet	4 – Høy konsekvens	Tidlig identifisere og avsette faktisk tid for ressurspersonell Hvem kan svare på hva? Hvem er tilgjengelig når? Sette opp kommunikasjonskanaler for hurtig kommunikasjon Microsoft Teams Slack Lage belegningsplan for å få oversikt over tilgjengelige ressurser og	Leverandør Kunde	Ok.

	eller forsinkede avklaringer i forbindelse med spesifikasjons- og utviklingsarbeid, vil kunne forsinke fremdriften i prosjektet.			forventet bruk av tid		
2	<p>Kort gjennomføringstid</p> <p>Det ønskes mye funksjonalitet på relativt kort kalendertid. Små forsinkelser pga. uforutsett merarbeid eller annet får direkte konsekvenser fordi prosjektet har kort kalendertid og man vil få utfordringer med å hente seg inn igjen. Det anbefales ikke å sette på flere utviklere i samme kalendertid som følge av avtalt øvre kostnadsramme og effektivitet per time.</p>	4 – Høy sannsynlighet	4 – Høy konsekvens	<p>Bevissthet om å lage det som er avtalt i fastpris</p> <p>Sette av tilstrekkelig tid til avklaringer og presiseringer i løsningsbeskrivelsesfasen</p> <p>Revidere prosjektplanene jevnlig mot forbrukt tid og levert funksjonalitet</p> <p>Kommunisere godt slik at forventningene er realistiske</p> <p>Jobbe kontinuerlig med å prioritere funksjonalitet slik at man kan fokusere arbeidet på det som er viktigst for å få en fungerende løsning.</p> <p>Det er Netcompany sin oppgave å kontinuerlig rapportere om fremdrift og status. Det må avklares med Kunden hva som inngår i en endelig leveranse.</p>	Leverandør Kunde	Inntruffet. Vi ser at det har vært nødvendig å omprioritere brukerhistoriene. Dette er gjennomgått med Forskningsrådet i sprint 2.

3	<p>Uforutsett behov for funksjonalitet</p> <p>Det åpnes for en viss grad av fleksibilitet mht. endringer og tillegg underveis i prosessen. Nye erkjennelser etter hvert som prosjektet skrider fram kan føre til større endringer og på den måten forsinke prosjektet. Det samme kan gjelde uforutsett funksjonalitet som ender i en <i>endringsordre</i>.</p>	2 – Lav sannsynlighet	4 – Høy konsekvens	<p>Koordinering og informasjonsflyt mellom leverandør og kundens fagperson(er)</p> <p>Det er Netcompany sin oppgave å kontinuerlig rapportere om fremdrift og status. Det må avklares med Kunden hva som inngår i en endelig leveranse.</p> <p>Utarbeide løsningsbeskrivelse/ designdokument som vil fungere som rettesnor for prosjektet.</p>	Leverandør Kunde	Ok
4	<p>Uenighet om leveransen</p> <p>Etter siste leveranse er ferdigstilt kan det være uenighet om innholdet i leveransen.</p> <p>Uten testcaser til brukerhistoriene og en prosess rundt dette kan det oppstå situasjoner hvor Kunde og Leverandør ikke har samme oppfattelse av</p>	2 – Lav sannsynlighet	3 – Middels konsekvens	<p>Allerede fra dag én opparbeide og dokumentere avgjørelse slik at man har et tydelig utgangspunkt for overlevering ved endt prosjekt.</p> <p>Definere akseptansekriterier</p> <p>Testing vil bli gjennomført av leverandør og kunde</p> <p>Godt kommunisert kravspesifikasjon</p>	Leverandør Kunde	Ok

	hva en fullført brukerhistorie er definert som.					
5	<p>Uforutsett fravær</p> <p>Sykdom over lengre perioder kombinert med knappe tidsfrister kan hemme fremdrift av prosjektet. Prosjektet er spesielt sårbart da det er et studentprosjekt.</p>	2 – Lav sannsynlighet	4 – Høy konsekvens	Studentgruppen er ansvarlig for å kommunisere med prosjektleder i forhold til mangel av arbeidskraft. Ved sykdom og uforutsett fravær for allokerte ressurser, skal Netcompany eksalere dette internt. Det vil bli undersøkt internt om det finnes ressurser som eventuelt kan avlaste prosjektet, samt andre løsninger.	Leverandør	Inntruffet. Det har vært en del sykdom i sprint 2.
6	<p>Manglende data</p> <p>Kunden skal fremskaffe eksisterende grafiske elementer og data (tekster) som skal benyttes i produktet.</p>	2 – Lav sannsynlighet	2 – Mindre farlig konsekvens	Leverandør og Kunde må koordinere og utarbeide det som må til for å eventuelt populere produktet med data. Ved mangel på data er det Netcompany sin oppgave å melde ifra til Kunde. Kunde er ansvarlig for å vedlikeholde tekster og data i Contentful	Leverandør Kunde	Ok
7	<p>Ny teknologi</p> <p>Det benyttes ny teknologi og</p>	5 – Svært høy sannsynlighet	2 – Mindre farlig	Sette av god tid til oppsett av utviklingsmiljø og plattform	Leverandør	Ny teknologi gjør at ting tar tid.

	<p>plattform for de fleste i utviklingsteamet, som kan påvirke fremgangen til teamet. Det er satt av tid til å lære seg ny teknologi, men det vil være en risiko for at det vil ta lenger tid enn antatt.</p>		<p>konsekvens</p>	<p>Kunden har godkjent den tekniske plattformen i eget arkitekturforum. Utviklere har fått kursing og gjennomgang av nøkkelresurser hos leverandøren. Det er avsatt nøkkelpersonell hos leverandøren. I startfasen vil det settes ekstra ressurser på opplæring og oppfølging som sikrer kvalitet og fremdrift i arbeidet.</p>		<p>Oppsett av AWS har vært krevende, men vi har tilgang til ressurspersoner som har hjulpet med dette.</p>
8	<p>Omfang og kompleksitet i utvikling av løsningen er mer krevende enn planlagt</p> <p>Utviklingen av de planlagte leveransene er mer komplekse enn antatt/forutsatt, og det kreves mer tid pr brukerhistorie enn forutsatt. Prosjektet greier ikke levere planlagte brukerhistorier til avtalt tid. Risiko for at prosjektet forsinkes.</p>	3 – Middels sannsynlighet	3 – Middels konsekvens	<p>Planlegge "pakker" med funksjonalitet. Oppdatere planverk etter det opparbeides erfaringstall. Bryte brukerhistorier ned slik at det blir enklere å treffe med estimatene.</p>	Leverandør	Ok

9	<p>Synergi effekter</p> <p>En god løsning vil inneha god teknisk kvalitet og brukeropplevelse. Av den grunn er det å foretrekke at kompetanse fra både kunde og leverandør blir utnyttet best mulig.</p>	3 – Middels sannsynlighet	4 – Høy konsekvens	Samarbeid mellom Netcompanys UX ressurs, prosjektgruppen og nøkkelpersonell hos kunden, både i konseptfasen og prosjektgjennomføringen	Leverandør Kunde	Ok
10	<p>Brukertestning</p> <p>For å sikre god brukeropplevelse vil det være nødvendig å gjøre et grundig arbeid mot målgruppen. Det er allerede utført arbeide med dette i forkant av prosjektstart for utvikling av konsept, men det vil være nødvendig å teste ut konseptet og implementasjon på målgruppen.</p> <p>Resultatene av brukertestningen vil være med på å påvirke utviklingen av produktet, samt Forskningsrådet får tilbakemelding</p>	3 – Middels sannsynlighet	2 – Mindre farlig konsekvens	Invitere testkandidater til brukertestning i god tid Netcompany har UX-ressurs tilgjengelig for samarbeid underveis i prosjektet	Leverandør Kunde	Ok

	r som kan være nyttig for utvikling av Nysgjerrigper(metoden).					
--	---	--	--	--	--	--

Alle positive risikofaktorer

Nr.	Beskrivelse	Sannsynlighet	Konsekvens	Tiltak	Ansvarlig	Status per 03/03
9	<p>Synergi effekter</p> <p>En god løsning vil inneha god teknisk kvalitet og brukeropplevelse. Av den grunn er det å foretrekke at kompetanse fra både kunde og leverandør blir utnyttet best mulig.</p>	3 – Middels sannsynlighet	4 – Høy konsekvens	Samarbeid mellom Netcompanys UX ressurs, prosjektgruppen og nøkkelpersonell hos kunden, både i konseptfasen og prosjektgjennomføringen	Leverandør Kunde	Ok
10	<p>Brukertestning</p> <p>For å sikre god brukeropplevelse vil det være nødvendig å gjøre et grundig arbeid mot målgruppen. Det er allerede utført arbeide med dette i forkant av prosjektstart for utvikling av konsept, men det vil være nødvendig å teste ut konseptet og</p>	3 – Middels sannsynlighet	2 – Mindre farlig konsekvens	Invitere testkandidater til brukertestning i god tid Netcompany har UX ressurs tilgjengelig for samarbeid underveis i prosjektet	Leverandør Kunde	Ok

	<p>implementasjon på målgruppen.</p> <p>Resultatene av brukertesting vil være med på å påvirke utviklingen av produktet, samt Forskningsrådet får tilbakemeldinger som kan være nyttig for utvikling av Nysgjerrigper(metoden).</p>					
--	--	--	--	--	--	--

6.5 BRUKERVEILEDNING

FOR NYSGJERRIGPER – PROSJEKTVEILEDER OG ADMINISTRASJONSPANEL

6.5.1 PROSJEKTVEILEDEREN

6.5.1.1 AKSESSERING

Nysgjerrigper – Prosjektveilederen ligger åpent på nett og kan nås på:

<http://np-frontend.s3-website.eu-central-1.amazonaws.com/>

6.5.1.2 LANDINGSSIDE

På landingssiden får besøkende en kort introduksjon til Nysgjerrigpermetoden og hvordan denne utføres i prosjektveilederen.

Herifra har man også tilgang til innloggingspanelet og siden for registrering av nye brukere gjennom knappene på toppen av siden.

6.5.1.3 REGISTRERING

For å bruke prosjektveilederen må besøkende ha en profil. Denne profilen har ikke tilknytning til andre tjenester hos Forskningsrådet, og kan dermed ikke benyttes til andre formål enn gjennomføring av Nysgjerrigper Prosjektveilederen.

Fra landingssiden vil man kunne komme deg til brukerregistreringen ved å trykke på knappen "Registrering".



Figur 131 - Landingssiden, "Registrering"-knappen er markert med rødt

Etter å ha trykt på linken kommer man til «Lag bruker» siden, i dette skjemaet er alle feltene påkrevd.

Epostadressen må ikke ha blitt brukt tidligere, passordet må inneholde en stor, og en liten bokstav, to tall og et spesialtegn.

Figur 132 Registreringsskjema, alle feltene er påkrevd

For å sikre kontoen din, samt informasjonen du har lagret i applikasjonen, er grensesnittet sikret med 2-faktor autentisering gjennom Google Authenticator, som er obligatorisk for applikasjonen.

Google Authenticator appen kan man laste ned gjennom lenkene under.

- Google Authenticator for Android: <https://play.google.com/store/apps/details?id=com.google.android.apps.authenticator2&hl=en>
- Google Authenticator for iOS: <https://itunes.apple.com/no/app/google-authenticator/id388497605?mt=8>

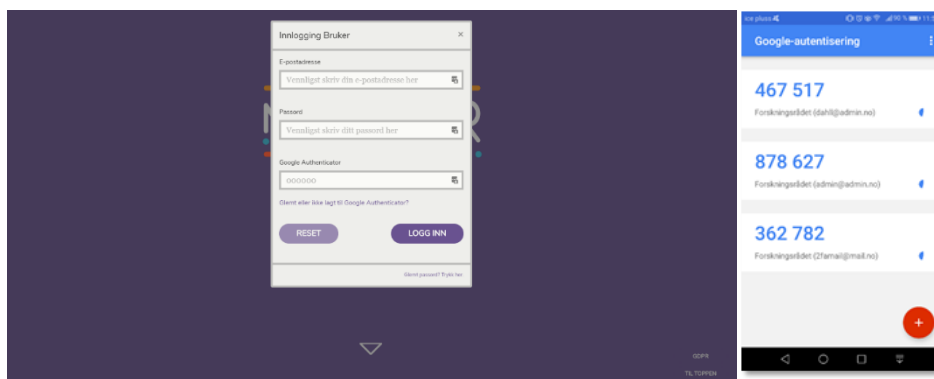
Figur 133 Skjema foroppsettelse av to-faktor autentisering

6.5.1.4 INNLOGGING

Fra landingssiden kan du åpne innloggingspanelet ved å trykke på knappen "Logg inn". Her skriver du inn påloggingsinformasjonen, altså registrert e-postadresse og passord, samt kode generert av Google Authenticator-appen. Dersom du har glemt eller mistet Google Authenticator, er det mulig å sette opp Google Authenticator på nytt etter registrering via lenken på innloggingspanelet.



Figur 134 Landingsiden



Figur 135 - Innloggingspanel og Google Authenticator applikasjon demonstrert på en Android mobiltelefon

6.5.1.5 OPPRETTELSE AV NYTT PROSJEKT

Brukeren har mulighet til å opprette et nytt prosjekt til en hver tid, selv om brukeren skulle ha et påbegynt prosjekt som ikke er ferdigstilt aktivt. Nytt prosjekt kan opprettes i menyen øverst i høyre hjørne i Prosjektveilederen.



Figur 136 Opprett prosjekt, befinner seg i menyen øverst til høyre i Prosjektveilederen

6.5.1.6 INNSTILLINGER

Brukeren vil ha tilgang til all informasjon lagret i Nysgjerriger - Prosjektveilederen ved å navigere til bruker- og prosjektinnstillinger. Her kan brukeren bla gjennom alle sine

tidligere prosjekter og enten slette eller velge dem. Man kan bare ha et aktivt prosjekt i Prosjektveilederen til enhver tid.



Figur 137 Innstillinger, disse finner i menyen øverst til høyre i Prosjektveilederen



Figur 138 - Side for bruker- og prosjektinnstillinger

6.5.1.7 UTLOGGING

For å logge ut av Nysgjerrigper Prosjektveilederen kan man trykke på eget navn øverst i høyre hjørne og deretter trykke på "Logg ut".



Figur 139 Logg ut, befinner seg i menyen øverst til høyre i Prosjektveilederen

6.5.1.8 NAVIGASJON PÅ SIDEN

Man har til enhver tid tilgang til alle stegene i nysgjerrigpermetoden, visse steg er dog avhengig av hverandre og krever at du har gjennomført dem. Man kan for eksempel ikke legge til hypoteser uten å ha en problemstilling fra før.

Merk at avsluttende steg "Se rapport" alltid er tilgjengelig, og oppdateres konstant med informasjon som brukeren har lagret på Prosjektveilederen.



Figur 140 - Navigasjonspanelet til Nysgjerrigper Prosjektveilederen

6.5.1.8.1 STEG I PROSJEKTVEILEDEREN

Alle stegene i prosjektveilederen inneholder tips til hvordan steget kan utføres, samt et utvidbart panel med mer informasjon.

6.5.1.8.1.1 FORBEREDELSE - KOM I GANG

Samler inn grunnleggende informasjon om gruppen. Denne informasjonen vises da til brukeren, og kan endres etter ønske.

Kom i gang

Så gøy at dere vil forske! Her kan dere finne litt informasjon om hvordan dere kan gjennomføre et Nysgjerrigper-prosjekt for dere begynner. Start med å fylle inn litt informasjon om dere selv og prosjektet. Deretter er det bare å sette i gang!

Fyll inn informasjon om prosjektet

Gruppens navn: Nysgjerrigper demo

Antall forskere: 23

Ansvarlig veileder: Petter Prosa

Skoletrinn: 3A

Skole: Varden

AVBRYT LAGRE ENDRINGER

Figur 141

Kom i gang

Så gøy at dere vil forske! Her kan dere finne litt informasjon om hvordan dere kan gjennomføre et Nysgjerrigper-prosjekt for dere begynner. Start med å fylle inn litt informasjon om dere selv og prosjektet. Deretter er det bare å sette i gang!

Prosjektet vårt:

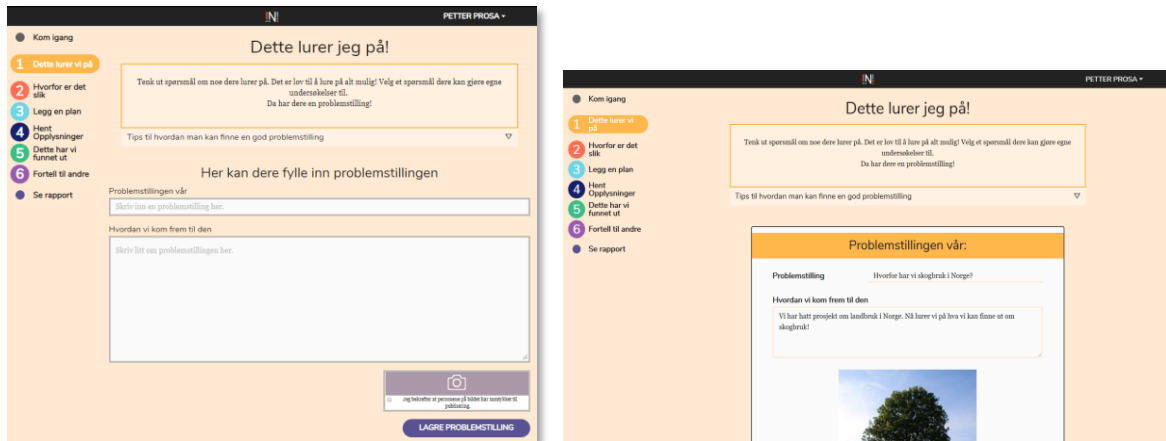
Gruppens navn	Nysgjerrigper demo
Antall forskere	23
Ansvarlig veileder	Petter Prosa
Skoletrinn	3A
Skole	Varden

ENDRE NESTE STEG

Figur 142

6.5.1.8.1.2 STEG 1 – DETTE LURER VI PÅ

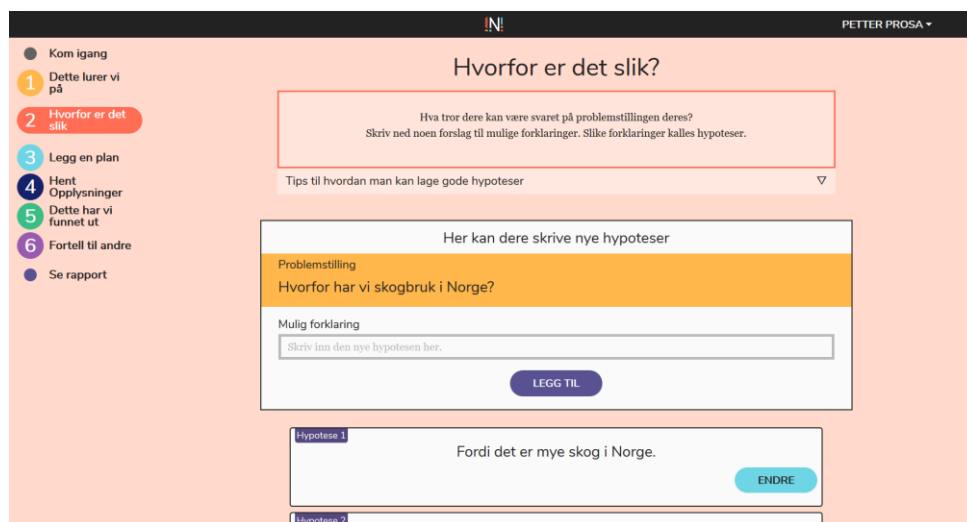
Gir brukeren muligheten til å legge inn problemstilling og beskrivelse av problemstillingen som da blir temaet for prosjektet. Her kan man også legge til et bilde om ønskelig.



Figur 13

6.5.1.8.1.3 STEG 2 – HVORFOR ER DET SLIK

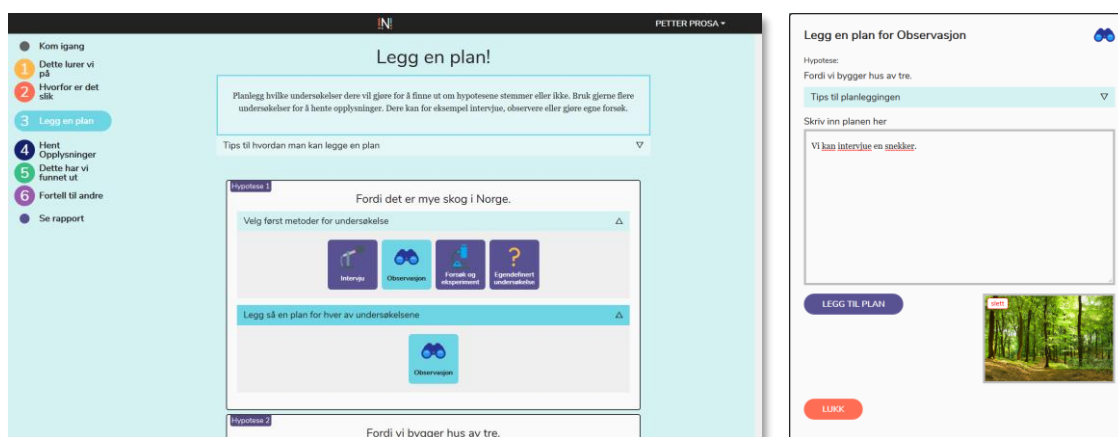
Her kan brukeren legge til hypoteser til den overordnede problemstillingen. Disse hypotesene kan redigeres av brukeren etter ønske.



Figur 143

6.5.1.8.1.4 STEG 3 – LEGG EN PLAN

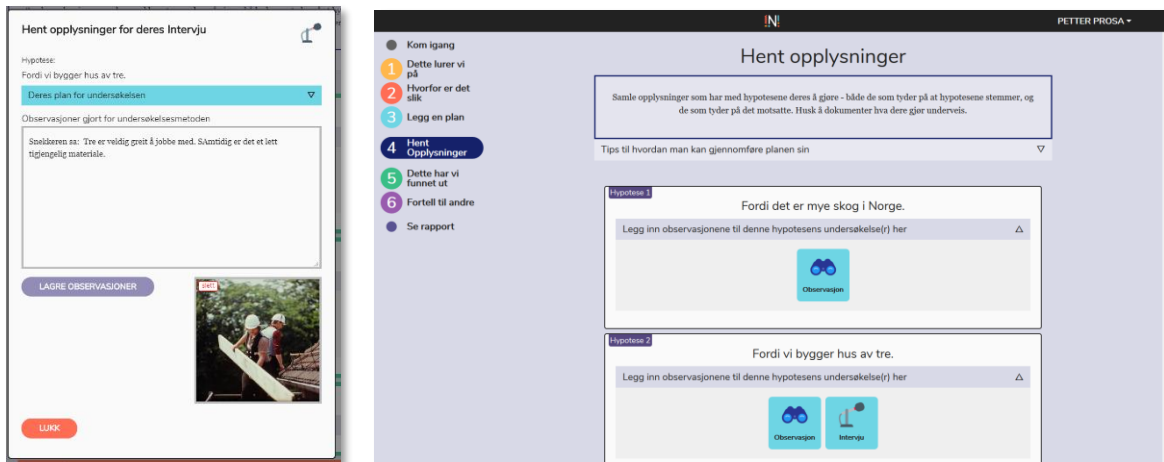
Viser alle hypoteser og tillater at brukeren legger til undersøkelsesmetoder for hver hypotese. Brukeren kan legge en plan for hvordan metoden skal utføres, samt endre denne planen om ønskelig. Hver enkelt metode inneholder tips til hvordan den kan utføres, det tillates at ett bilde lastes opp i planen.



Figur 144

6.5.1.8.1.5 STEG 4 – HENT OPPLYSNINGER

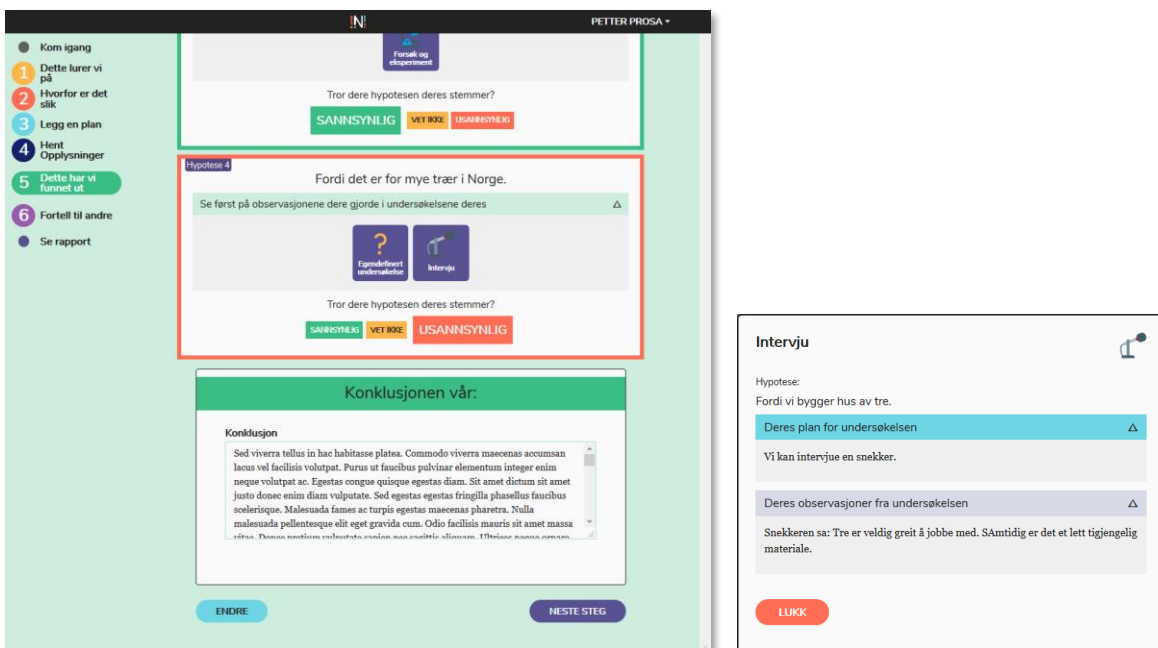
Viser alle hypoteser og tillater at brukeren legger til informasjon på hver enkelt metode. Brukeren har også her mulighet til å laste opp ett bilde til hver metode, slik som i steg 3.



Figur 145

6.5.1.8.1.6 STEG 5 – DETTE HAR VI FUNNET UT

Viser alle hypoteser og lar brukeren velge hvor sannsynlig det er at hypotesen er et svar på problemstillingen basert på opplysningene hentet inn. Brukeren kan også legge til en avsluttende konklusjon på prosjektet, og laste opp et bilde.

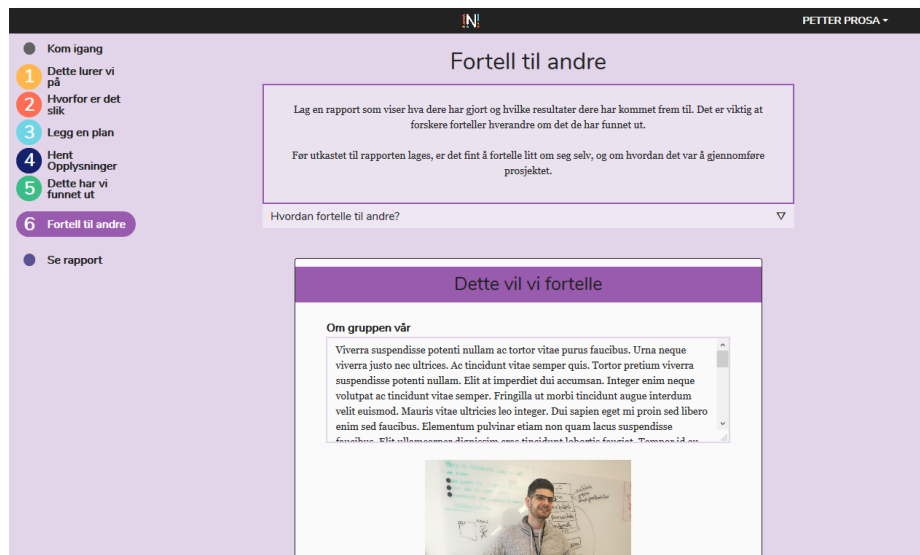


Figur 146

Om et prosjekt er sannsynlig får hypoteseboksen et grønt omriss, usannsynlig gjør omrisset rødt, og usikkert gjør det gult.

6.5.1.8.1.7 STEG 6 – FORTELL TIL ANDRE

Tillater at brukeren kan legge til en beskrivelse av både gruppen og prosjektet, samt legge til ett bilde til hver av disse feltene.



Figur 147

6.5.1.8.1.8 AVSLUTNING - SE RAPPORT

Etterhvert som *modellen* for prosjektet fylles ut med data, hypoteser og metoder, så blir denne informasjonen illustrerbar og kan sees ved at brukeren trykker på *Se rapport*. Brukeren blir sendt til et nytt vindu som viser hele prosjektet med all dataen som er lastet opp til et prosjektdokument. Denne har en lenke til en utskriftsvennlig versjon av dokumentet, som kan lagres på datamaskinen som en PDF-fil.



Figur 148

Utaktilsynlig versjon

NYSGJERRIGPER FORSKNINGSRAPPORT

Tittel: Hedemarken Friforskere
 Skole: Hedemarken Friskole
 Lærer: Ola Nordmann
 Antall deltagere: 20

Innledning

Problemstilling:
 Er det sant at vi er like lange fra fingertupp til fingertupp som fra hode til føtter?

Hvordan vi kom frem til den:

En dag i høst mens klassen satt og spiste var det en gutt som spurte: "Er det sant at vi er like lange fra fingertupp til fingertupp som fra hode til føtter?" Det hadde nemlig gutten blitt fortalt av noen. Læreren visste ikke svaret, men foreslo at vi kunne forske på det. Det ville alle, så da ble årets forskningsprosjekt i Nysgjerrigper til.



Figur 149

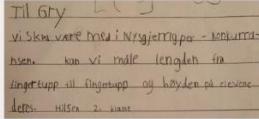
Hypotese 4:

Vi tror at mennesker mellom 9 og 12 år ikke har samme lengde mellom fingertupper som fra hode til fot.

Undersøkelser

Forsøk og eksperiment

Plan:
 Vi vil skrive brev til lærerne i fjerde til sjuende-trinn og spørre om vi kan få måle elevene deres.



Observasjoner og datainnsamling:
 Gjennomsnittlig forskjeller i cm:

- Gutter: 1.9 cm
- Jenter: 2.6 cm

Figur 150

6.5.2 ADMINISTRASJONGRENSESNIETTET

6.5.2.1 AKSESSERING

For å få tilgang til administrasjonsgrensesnittet så er man avhengig av at brukeren man har logget på med har administratorrettigheter. Administratorrettigheter kan bare settes i administrasjonsgrensesnittet, og dermed bare av brukere som allerede har administratorrettigheter.

Administrasjonsgrensesnittet kan nås ved å skrive /admin på enden av URLen til nettsiden, **med** dvs. koble seg til:

<http://np-frontend.s3-website.eu-central-1.amazonaws.com/admin>

6.5.2.2 INNLOGGING

Bruker samme innlogging som brukergrensesnittet. Se [Innlogging](#) under brukergrensesnitt hovedkapittelet.

6.5.2.3 DET GENERELLE ADMINISTRASJONGRENSESNIETTET

Navigasjonslinjen på venstre side linker brukeren til forskjellige sider i administrasjonsgrensesnittet. På brukersiden og prosjektsiden har man mulighet til å søke i tabellen og eksportere all metadata til en CSV fil.

Id	Fornavn	Efternavn	Email	Rolle	Opprettet
Id-1	Olav	Vestlie	olav@milehigh.no	USER	02-03-2019
Id-2	Yanis	Mustis	forever@young.com	USER	01-03-2019
Id-3	Vegardo	Leivoso	livin@briskeby.no	USER	03-03-2019
Id-4	Thomas	Toger	livets@hardescole.no	USER	05-03-2019
Id-5	Dummy	Boy	user@user.com	USER	10-03-2019
Id-7	Guðnún	Gul	guo@vesterdahls.com	USER	10-03-2019
Id-8	Fritof	Frik	fr@het.no	USER	10-03-2019
Id-9	Gunnar	Gunn	mail@erprivatslivingipen.no	USER	10-03-2019
Id-10	Petter	Prosa	admin@nysgjerriger.no	USER_ADMIN	11-03-2019
Id-6	Petter	Prosa	admin@admin.no	USER_ADMIN	11-03-2019

Figur 151 Administrasjonsgrensesnittet

6.5.2.4 BRUKERSIDEN

Den første siden du kommer til etter at du har skrevet inn URLen ovenfor er Brukerinformasjonssiden. Brukerinformasjonssiden viser informasjon om alle brukerne av systemet.

Id	Fornavn	Etternavn	Email	Roller	Opprettet
Id-1	Olav	Vestfie	olav@milehigh.no	USER	02-03-2019
Id-2	Yanis	Mussis	forever@young.com	USER	01-03-2019
Id-3	Vegardio	Leviosa	livin@briskeby.no	USER	03-03-2019
Id-4	Thomas	Toger	livets@hardeskole.no	USER	05-03-2019
Id-5	Dummy	Boy	user@user.com	USER	10-03-2019
Id-7	Gudrun	Gul	gul@westerdahlts.com	USER	10-03-2019
Id-8	Fritjof	Frik	fri@het.no	USER	10-03-2019
Id-9	Gunnar	Gunnis	mail@erprivatstivingripen.no		10-03-2019
Id-10	Petter	Prosa	admin@nysgjerrigper.no	USER_ADMIN	11-03-2019
Id-6	Petter	Prosa	admin@admin.no	USER_ADMIN	11-03-2019

Figur 152 Brukerinformasjonssiden viser informasjon om alle brukerne av systemet.

Det er mulig å velge om man vil se samtlige brukere, eller filtrere på lærerbrukere. På siden som viser samtlige brukere har man mulighet til å slette brukere og endre brukerrettigheter, se underkapitlet [nedanfor](#). Mens på lærersiden har man mulighet til å se alle prosjektene som læreren har opprettet, se [Siden som viser alle lærere](#).

Id	Fornavn	Etternavn	Email	Roller	Opprettet
Id-1	Olav	Vestfie	olav@milehigh.no	USER	02-03-2019
Id-2	Yanis	Mussis	forever@young.com	USER	01-03-2019
Id-3	Vegardio	Leviosa	livin@briskeby.no	USER	03-03-2019
Id-4	Thomas	Toger	livets@hardeskole.no	USER	05-03-2019
Id-5	Dummy	Boy	user@user.com	USER	10-03-2019
Id-7	Gudrun	Gul	gul@westerdahlts.com	USER	10-03-2019
Id-8	Fritjof	Frik	fri@het.no	USER	10-03-2019
Id-9	Gunnar	Gunnis	mail@erprivatstivingripen.no		10-03-2019
Id-10	Petter	Prosa	admin@nysgjerrigper.no	USER_ADMIN	11-03-2019
Id-6	Petter	Prosa	admin@admin.no	USER_ADMIN	11-03-2019

Figur 153 Her kan man velge å vise samtlige brukere, eller filtrere på lærerbrukere (1)

6.5.2.5 SIDEN SOM VISER ALLE BRUKERE

På brukersiden som viser alle brukere har man mulighet til å slette brukere, og man mulighet til å endre rettighetene til brukere i dette grensesnittet. Denne muligheten får man ved å trykke på den brukeren man ønsker å endre.

Brukerinformasjon						
id	Fornavn	Efternavn	Email	Rolle	Opprettet	
Id-1	Olav	Vestlie	olav@milisigh.no	USER	02-03-2019	
1: Olav Vestlie Opprettet dato: 02-03-2019 ROLLER: USER ... <div style="text-align: center;"> 1 → SLETT ENDRE RETTIGHETER ← 2 </div>						
Id-2	Yanis	Musis	forever@young.com	USER	01-03-2019	
Id-3	Vegardio	Leivosa	livin@briskeby.no	USER	03-03-2019	
Id-4	Thomas	Toger	livets@hardeskoie.no	USER	05-03-2019	
Id-5	Dummy	Boy	user@user.com	USER	10-03-2019	
Id-7	Guðrun	Gul	gud@westerdahl.com	USER	10-03-2019	
Id-8	Fritjof	Frik	fri@het.no	USER	10-03-2019	
Id-9	Gunnar	Gunnis	mail@erprivatslivingipen.no	USER	10-03-2019	
Id-10	Petter	Prosa	admin@mysgjerrigper.no	USER_ADMIN	11-03-2019	
Id-6	Petter	Prosa	admin@admin.no	USER_ADMIN	11-03-2019	

Figur 154 Her kan man slette brukere (1), og endre rettighetene til hver enkelt bruker (2)

6.5.2.6 SIDEN SOM VISER ALLE LÆRERE

På lærersiden som viser alle lærere har man mulighet til å se alle prosjektene læreren har opprettet.

Lærerinformasjon						
id	Fornavn	Efternavn	Skole	Email	Opprettet	
Id-5	Dummy	Boy	Vålerenga	user@user.com	10-03-2019	
5: Dummy Boy Opprettet dato: 10-03-2019 Antall prosjekter: 4 Antall fullførte prosjekter: 1 <div style="text-align: center;"> 1 → PROSJEKTER </div>						
Id-7	Guðrun	Gul	Ornestad	gud@westerdahl.com	10-03-2019	
Id-8	Fritjof	Frik	Gamlebyen	fri@het.no	10-03-2019	
Id-9	Gunnar	Gunnis	Vålerenga	mail@erprivatslivingipen.no	10-03-2019	
Id-10	Petter	Prosa	Vålerenga	admin@mysgjerrigper.no	11-03-2019	
Id-6	Petter	Prosa	Vården	admin@admin.no	11-03-2019	
Id-11	asdasdasdasd	asdasdasdasd	asdasdasd	asdasd@asdasd.no	24-04-2019	
Id-12	asdasdasd	asdasdasd	asdasdasd	asdasdasd@asdasdasd.no	24-04-2019	
Id-13	Marit	Malhausen	Marit skole	mimo@rcn.no	24-04-2019	
Id-14	Trude	Hauge	Petsund	trh@rcn.no	24-04-2019	

Figur 155 Her har man mulighet til å se alle prosjektene som en lærer har opprettet

6.5.2.7 PROSJEKTSIDEN

Prosjektsiden viser metadata om alle prosjektene som er lagret i applikasjonen. Typisk interessante felter er hvem som står som eier av prosjektet, når det ble opprettet og når det ble ferdigstilt.

BRUKERE								
PROSJEKTER								
STATISTIKK								
EKSPORTER TIL CSV								
NULLSTILL SØK								
Søkefelt...								
Id	Titel	Antall Studenter	Skole	Tittel	L-Id	Påbegynt	Fullført	
Id-80	awdawdawwad	23	Bergen Administrasjonsskole	awdawdawwad	L-72	2019-05-13		
Id-84	Demo	12	Fyrverkeriet barneskole	2. barneskole	L-6	2019-05-13		
Id-16	Demo	12	Fyrverkeriet barneskole	3d	L-6	2019-04-06		
Id-14	Superforskerne	23	Sem skole	Forskere 4. og 8. klasse	L-6	2019-04-05	2019-04-25	
Id-11	Nysgjerriger demo	1	Vålerenga	4c	L-5	2019-04-04		
Id-12	Explorer	5	Bjørnsletta skole	8	L-5	2019-04-05		
Id-6	Gruppe 56	56	Re	5 klasse	L-6	2019-04-04		
Id-10	Nysgjerriger demo	1	Vålerenga	4c	L-10	2019-04-04		
Id-13	awdawdawwad	13	awdawwad	dawdawwad	L-6	2019-04-05		
Id-1	Prname	23		4A	L-6	2019-03-10		

Antall rader per side: 10 25 30 50 Viser fra 1 til 10 av 77 resultat 1 2 3 4 5 > >>

Figur 156 Prosjektsiden

Trykker man på et prosjekt får man mulighet til å slette prosjektet.

BRUKERE								
PROSJEKTER								
STATISTIKK								
EKSPORTER TIL CSV								
NULLSTILL SØK								
Søkefelt...								
Id	Titel	Antall Studenter	Skole	Tittel	L-Id	Påbegynt	Fullført	
Id-80	awdawdawwad	23	Bergen Administrasjonsskole	awdawdawwad	L-72	2019-05-13		
<div style="border: 1px solid #ccc; padding: 5px;"> <p>ID: awdawdawwad</p> <p>Navn på lærer: Yedimislav Zadmorov Id: 72</p> <p>Oppdatert sist: 2019-05-13</p> <p>Beskrivelse:</p> <p>Problemløsing:</p> <p>Konklusjon:</p> <p style="text-align: center;">1 SLETT</p> </div>								
Id-84	Demo	12	Fyrverkeriet barneskole	2. barneskole	L-6	2019-05-13		
Id-16	Demo	12	Fyrverkeriet barneskole	3d	L-6	2019-04-06		
Id-14	Superforskerne	23	Sem skole	Forskere 4. og 8. klasse	L-6	2019-04-05	2019-04-25	
Id-11	Nysgjerriger demo	1	Vålerenga	4c	L-5	2019-04-04		
Id-12	Explorer	5	Bjørnsletta skole	8	L-5	2019-04-05		
Id-6	Gruppe 56	56	Re	5 klasse	L-6	2019-04-04		
Id-10	Nysgjerriger demo	1	Vålerenga	4c	L-10	2019-04-04		
Id-13	awdawdawwad	13	awdawwad	dawdawwad	L-6	2019-04-05		
Id-1	Prname	23		4A	L-6	2019-03-10		

Antall rader per side: 10 25 30 50 Viser fra 1 til 10 av 77 resultat 1 2 3 4 5 > >>

Figur 157 Ved å trykke på et prosjekt får man mulighet til å slette prosjektet (1)

6.5.2.8 STATISTIKKSIDEN

Statistikksiden gir innsikt i hvor mange prosjekter som har blitt opprettet i forhold til hvor mange som er ferdigstilt, og hvor mange brukere som har blitt lagret den siste perioden.

Den viser også tall over det totale antallet prosjekter opprettet, det totale antall prosjekter ferdigstilt og det totale antallet brukere på systemet.



Figur 158 Statistikk siden

6.6 BRUKERTESTER

6.6.1 SAMTYKKESKJEMA

Forespørsel om deltakelse i brukertest/intervju

- Om Nysgjerrigper Prosjektveileder

Bakgrunn og formål

Nysgjerrigper er en forskningsmetode utviklet av Norges forskningsråd som har som mål å lære elever til å bruke den vitenskapelige metoden på en morsom og interessant måte.

Forskningsrådet skal lage en løsning for å gjøre det lettere for lærere å bruke Nysgjerrigpermetoden. Løsningen er en prosjektveileder som både kan benyttes til å planlegge et forskningsprosjekt med elevene, og som veileder deg trinn for trinn gjennom Nysgjerrigpermetoden. Prosjektrapporten eller prosjektloggen lagres, og kan brukes som utgangspunkt for en skriftlig forskningsrapport. Netcompany har fått i oppgave å utvikle denne nettapplikasjon på vegne av Forskningsrådet.

Forskningsrådet utforsker et nytt brukergrensesnitt med ny funksjonalitet. Som et ledd i denne prosessen er det viktig å få innspill og innsikt av brukere og potensielle brukere av metoden.

Hva innebærer deltakelse i testen?

Deltakelsen i studien vil innebære diskusjon rundt konsepter, i tillegg vil du få teste en prototype. Totalt tar dette ca. 30-40 minutter. Av personlige opplysninger ber vi kun om fornavn. **Deltageren vil, med samtykke, bli fotografert under intervjuet for bruk til dokumentasjon og presentasjon av prosjektet.**

Hva skjer med informasjonen om deg?

Alle personopplysninger vil bli behandlet konfidensielt. Dokumentasjonen vil bli oppbevart på passordbeskyttede maskiner, og vil ikke bli delt med andre.

Dersom du ønsker å bli anonymisert i forbindelse med dokumentasjon og presentasjon kan du velge dette nå. (Egen avkrysning lenger nede i dokumentet.)

Alt av datamateriale vil bli anonymisert ved prosjektets slutt. Forventet sluttdato for prosjektet er medio mai.

Frivillig deltakelse

Det er frivillig å delta i prosjektet, og du kan når som helst trekke ditt samtykke uten å oppgi noen grunn. Dersom du trekker deg, vil alle opplysninger om deg bli anonymisert.

Har du spørsmål til deltakelse i prosjektet, ta kontakt med Yanislav Zagorov (405 73 277) yanisz@netcompany.com.

Samtykke til deltakelse i studien

Jeg har mottatt informasjon om studien, og er villig til å delta

Jeg ønsker å bli anonymisert i forbindelse med dokumentasjon og publisering.

Sted og dato

Signatur

6.6.2 BRUKERTESTDOKUMENTET

PROSJEKTBEKRIVELSE

Forskningsrådet ønsker å senke terskelen for at lærere skal ta i bruk Nysgjerrigpermetoden, som er Forskningsrådets forskningsmetode for barn i 1.-7. trinn. Dette skal de gjøre ved å utvikle en ny webapplikasjon som skal veilede lærerne gjennom oppbygning og innlevering av forskningsrapporter.

Dagens webapplikasjon er tung i bruk og lite intuitiv. Man er i stor grad avhengig av å kjenne til brukergrensesnittet for å finne frem og man må vite hva det er som skal til for å produsere en god forskningsrapport.

I tillegg til å forbedre brukergrensesnittet, ønsker de blant annet å gjøre løsningen sikrere gjennom to-faktor autentisering.

De siste årene har det kommet flere andre konkurrenter på markedet. Det er derfor enda viktigere enn før å kunne tilby et godt brukergrensesnitt og en konkurransedyktig løsning til brukerne.

HOVEDMÅL

- Designe et brukervennlig brukergrensesnitt for Nysgjerrigpermetoden
- Ta høyde for ny funksjonalitet
- God designprosess med mulighet til å gjøre endringer på et allerede utviklet produkt.

FORBEREDENDE FASE

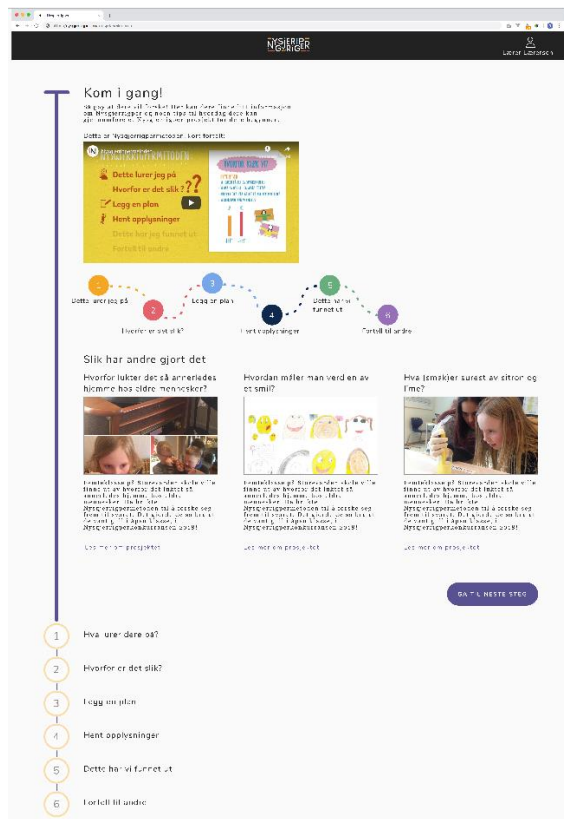
BAKGRUNN

Dagens situasjon bærer preg av en tung innleveringsprosess. Nysgjerrigpermetodens veiledningsdokumenter er lange og tunge å lese, dessuten har man med dagens løsning bare mulighet til å levere en PDF fil som man har laget på forhånd. Nysgjerrigper har gitt oss i oppgave å lage en prosjektveilederen. Denne prosjektveilederen skal både gi lærere en stegvis veiledning i hva som forventes av en oppgave, og kunne benyttes til å lage prosjekter. Dessuten skal Nysgjerrigper kunne gi brukerne sine kundestøtte, men da systemene er kompliserte og vanskelige å tilby kundesupport til er det behov for at en ekstern datakyndig person tar seg av dette.

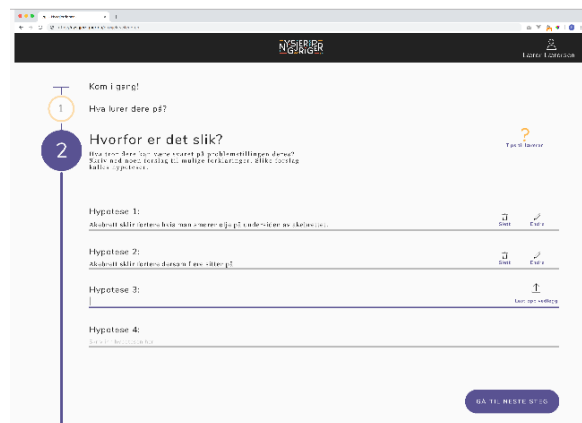
Nysgjerrigper ønsker seg derfor et administrasjonsgrensesnitt som viser hvor mange det er som tar i bruk metoden, de ønsker også at dette grensesnittet skal inneholde funksjonalitet som tillater at Nysgjerrigper blant annet kan legge til brukerprofiler.

FØRSTE DESIGN UTKAST

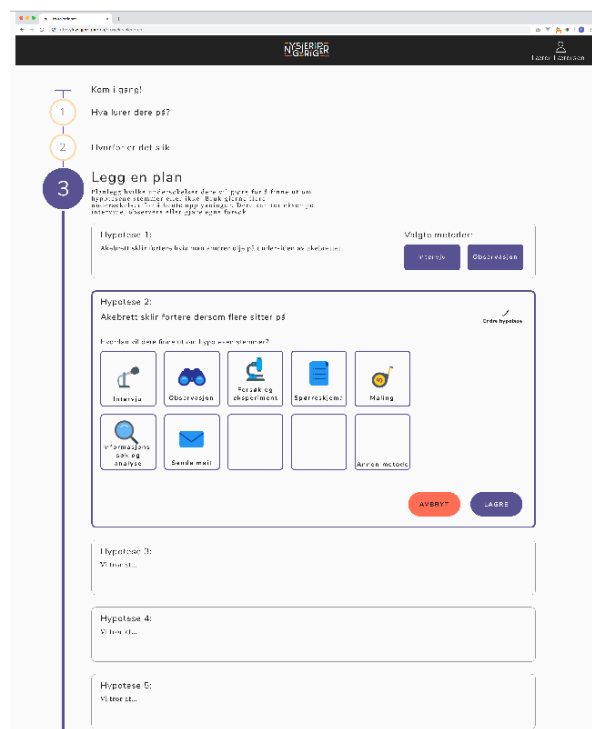
Målet med designet er å gjøre applikasjonen enkel og mer effektiv i bruk.



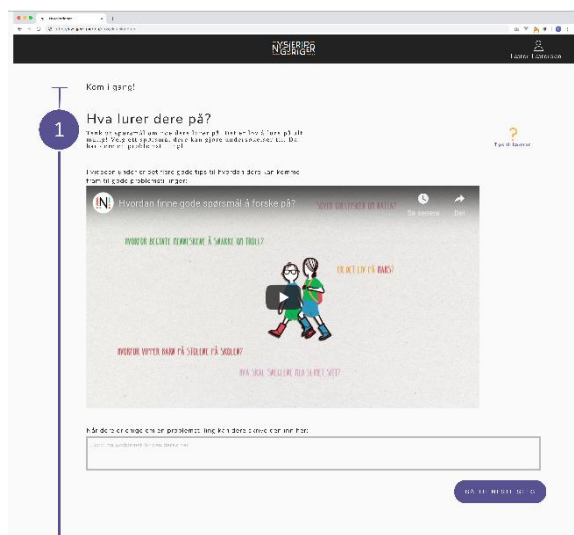
Figur 159 Kom i gang



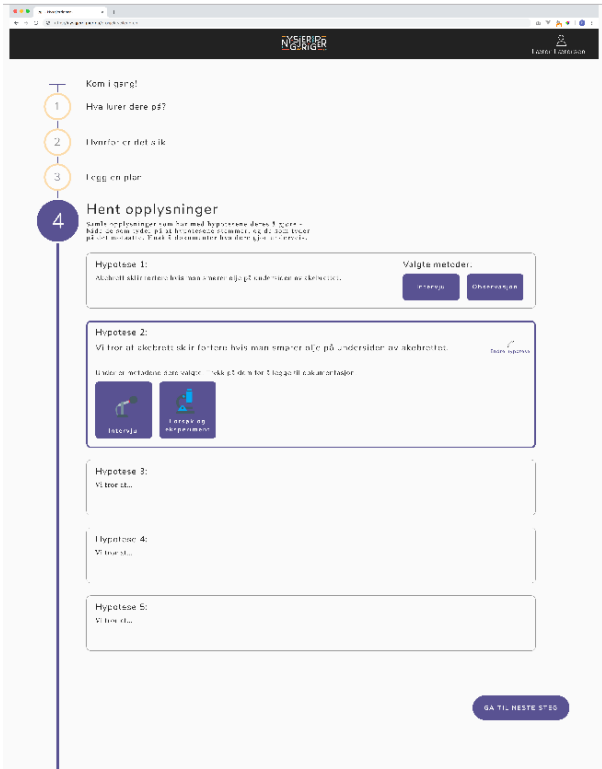
Figur 161 Steg 2



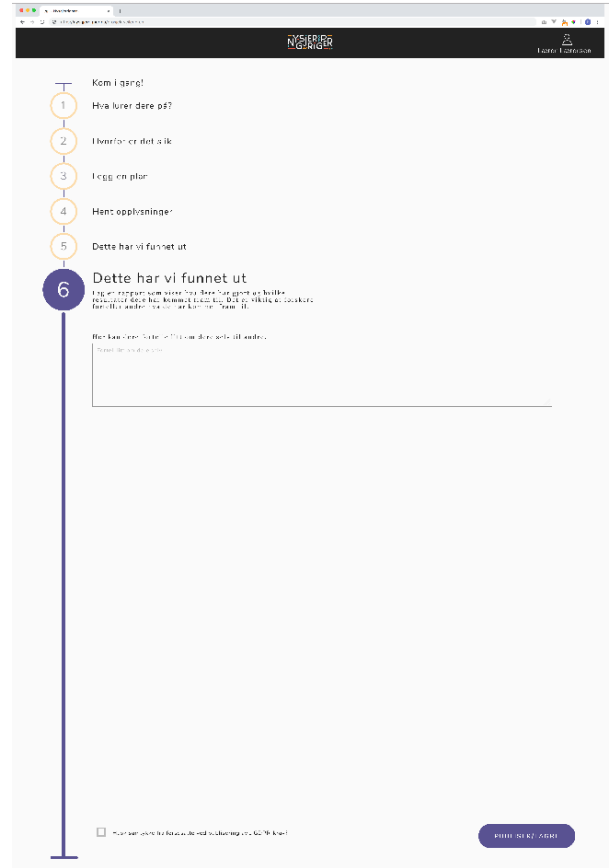
Figur 162 Steg 3



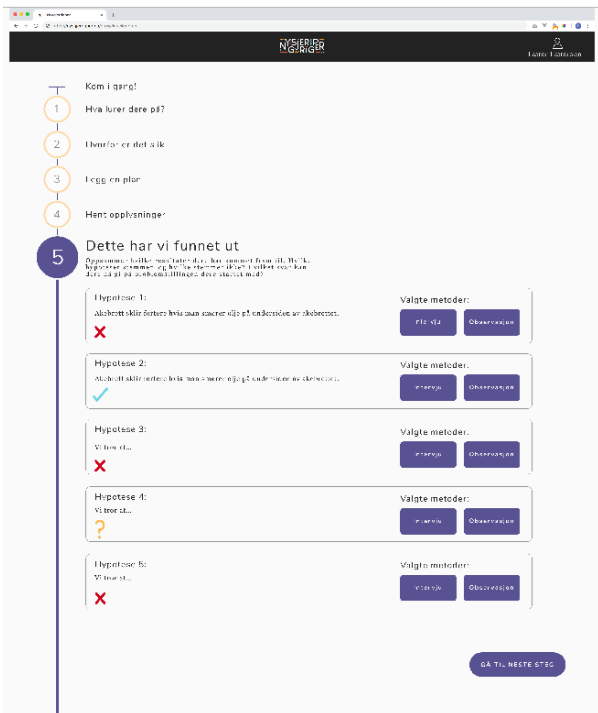
Figur 160 Steg 1



Figur 163 Steg 4



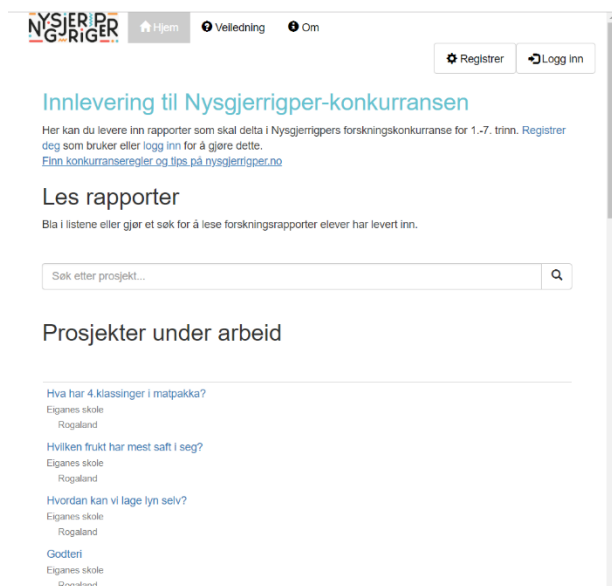
Figur 165 Steg



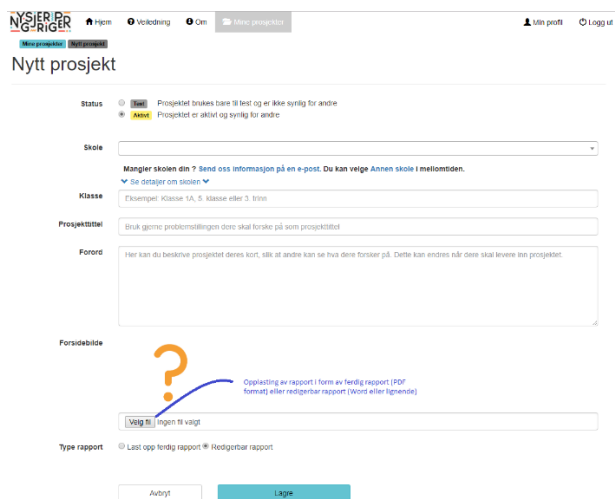
Figur 164 Steg 5

DAGENS LØSNING

Nysgjerrigpermetodens veiledningsdokumenter er lange og tunge å lese, dessuten må man navigere seg rundt på nettsiden for å finne fram til disse. I tillegg har man med dagens løsning bare mulighet til å levere en PDF fil som man har laget på forhånd.



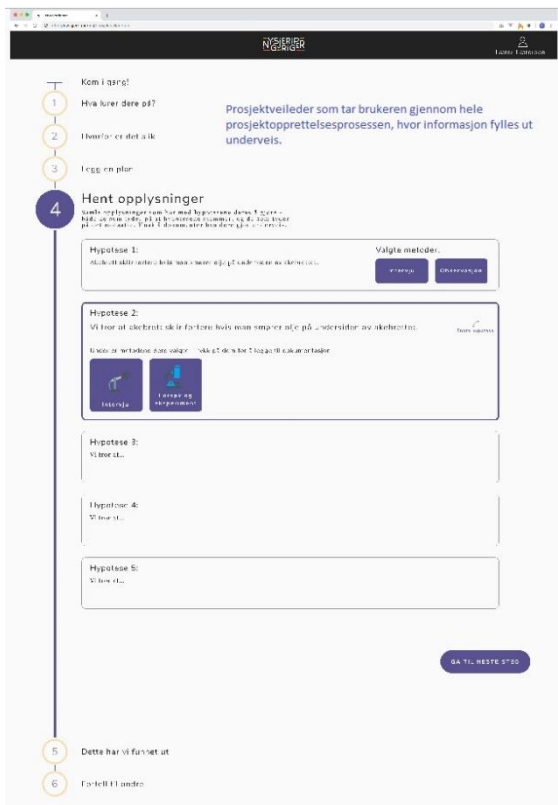
Figur 8 Forsiden til nysgjerrigpermetoden.no



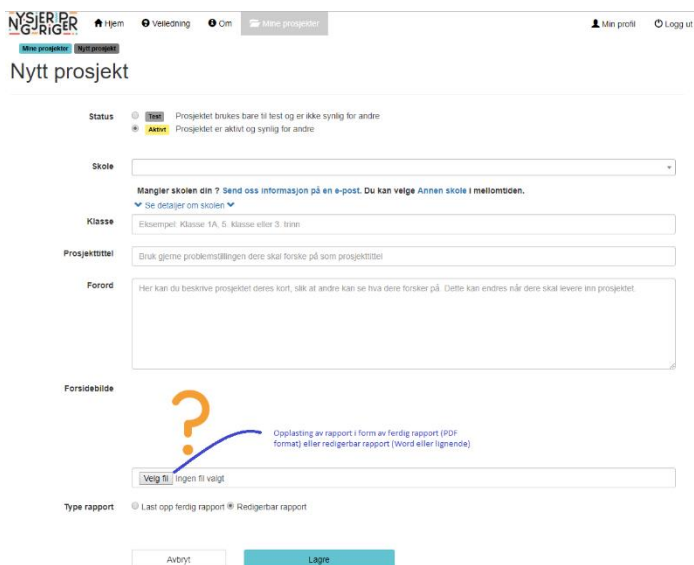
Figur 166 Etter pålogging kommer man til prosjektopprettelsesvinduet

DAGENS VS. NYTT DESIGN

I det nye brukergrensesnittet, har vi fokusert på å lage det så intuitivt som mulig. Navigasjonsknapper som hjelper brukeren gjennom de 6 stegene som Nysgjerrigpermetoden inneholder. Dette minimerer mulige lenkevalg brukeren har og minsker muligheten for feilklikk.



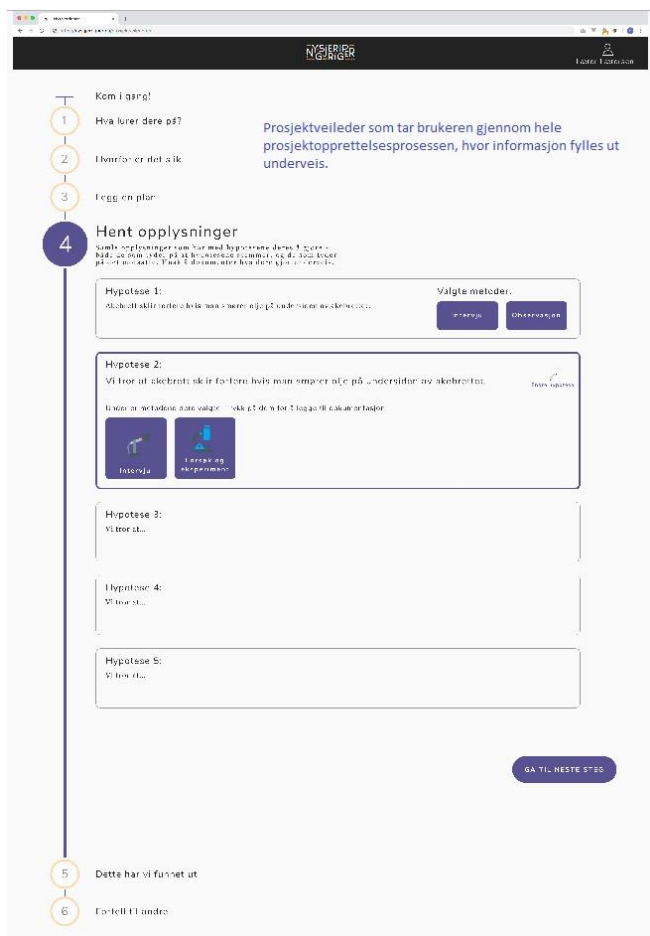
Figur 167 Prosjektveilederen tar brukeren gjennom hele prosjektopprettelsesprosessen.



Figur 168 I den gamle prosjektveilederen er man avhengig av at brukeren vet hva som skal til for å oppfylle kravene til et prosjekt, og at brukeren vet hvordan man genererer en rapport.

ENDRINGER I BRUKERGRENSESNITT OG FUNKSJONALITET

Designet følger Nysgjerrigpers design guide, som inneholder et sett med regler for fargevalg og skrifttyper. I tillegg til et forbedret brukergrensesnitt, inneholder applikasjonen flere funksjoner enn den gamle applikasjonen. I den gamle applikasjonen tok man for gitt at lærerne var kapable til å generere en PDF fil på egenhånd, for så å laste opp denne. I det nye brukergrensesnittet fyller man inn informasjon og dataene lagres i en database underveis. Når prosjektet blir ferdigstilt vil Forskningsrådet kunne se og hente ut rapporten fra databasen.



Figur 169 Prosjektveilederen tar brukeren gjennom hele prosjektopprettelsesprosessen.

Nytt prosjekt

Status: **Innlagt** (Prosjektet brukes bare til test og er ikke synlig for andre) / **Aktive** (Prosjektet er aktivt og synlig for andre)

Skole:
 Mangler skolen din? Send oss informasjon på en e-post. Du kan velge annen skole i mellomtiden.
 ▼ Se detaljer om skolen ▼

Klasse:
 Eksempel: Klasse 1A, 5. Klasse eller 3. Innr.

Prosjekttittel:
 Bruk gjerne problemstillingen dere skal forske på som prosjekttittel

Forord:
 Her kan du beskrive prosjektet deres kort, slik at andre kan se hva dere forsker på. Dette kan endres når dere skal levere inn prosjektet

Forsidebilde:
 Opplasting av rapport i form av ferdig rapport (PDF-format) eller redigerbar rapport (Word eller lignende)

Type rapport: Last opp ferdig rapport / Redigerbar rapport

Avbryt | Lagre

Figur 170 I den gamle prosjektveilederen er man avhengig av at brukeren vet hva som skal til for å oppfylle kravene til et prosjekt, og at brukeren vet hvordan man genererer en rapport.

RESULTAT

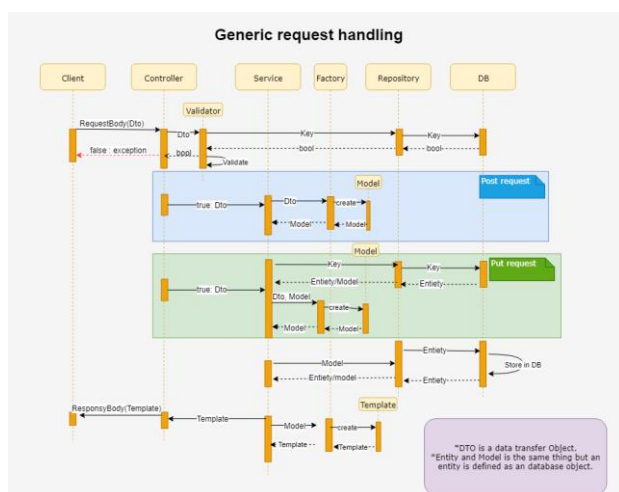
Flere iterasjoner har blitt gjort, og resultatet av prosjektet kan ses her:

<http://np-frontend.s3-website.eu-central-1.amazonaws.com>

METODER OPPSUMMERT:

- Workshops
- «Low fidelity» prototyping
- «High fidelity» prototyping
- Brukerhistorier
- Intervjuer
- Brukertester

FLYT



Figur 171 Flytskjema

BRUKERTEST: [TESTPERSONENS NAVN]

BRUKERE

Vi skal gjennomføre brukertesting sammen med fem lærere. Brukertesting vil hjelpe oss å kartlegge mulig forbedringspotensial og eventuelle mangler.

TESTLEDER

- [Testlederens navn].
 - Testlederen har ansvar for utførelsen av testene.
 - Testlederen forklarer prosessen, intensjonen og målet for testingen til brukerne.
 - Testlederen forklarer at det er applikasjonen, ikke brukeren som vil bli testet og evaluert.
 - Testlederen oppfordrer brukerne til å snakke og fortelle om det de tenker høyt.
 - Stiller spørsmålene/gir oppgavene og instruksjoner.
-

OBSERVATØRER

- [Observatørens navn]
 - Tar notater og svarer på eventuelle spørsmål som blir stilt underveis.
 - Referenten skriver ned alt det brukeren gjør for å få til den tildelte oppgaven.
-

INTRODUKSJON

Vi holder på å lage en ny applikasjon for Nysgjerrigper som skal senke terskelen for at lærere velger å ta i bruk Nysgjerrigpermetoden. Det er viktig for oss å få testet kravene som er satt for applikasjonen. Å den beste måten å teste disse kravene på er ved å teste dem på sluttbrukerne, som i dette tilfelle er lærere. For deg som tester er det viktig å tenke på at det er applikasjonen som testes å ikke deg som bruker.

Testen vil utføres som følger:

- Innledende spørsmål om dagens løsning og første inntrykk av den nye applikasjonen.
 - Læreren vil bli gitt praktiske oppgaver som han/hun vil prøve å løse etter beste evne.
 - Spørsmål om løsningen og tilbakemeldinger (generelle og spesifikke).
-

INNLEDENDE SPØRSMÅL

Benytter du en form for forskningsmetode i undervisningen i dag?

Dersom du har benyttet Nysgjerrigpermetoden, vil du si at dagens løsning er tilstrekkelig?

FØRSTEINNTRYKK

Hva ser du her?

PRAKTISKE OPPGAVER

Du vil bli spurt om å navigere deg rundt på siden, registrere bruker, logge på bruker, opprette prosjekt og fylle inn tekst i prosjektet.



Figur 172 Landingssiden til Prosjektveilederen

E1 INNLOGGING OG SIKKERHET

Hovedtema	Oppgave	Ønsket resultat	Oppnådd resultat/diskusjon	Forbedringspotensial
E1.1	Føler du at landingssiden du havner på før du har logget inn i applikasjonen er godt utformet med tanke	Testpersonen forklarer hvordan han/hun oppfatter at han/hun skal gå frem, og		

	på den skal gi en kort introduksjon til metoden og hva som må gjøres for å komme i gang?	forklaringen stemmer med en måte som vil løse oppgaven		
E1.2	Lag deg en bruker.	Det skal opprettes en bruker basert på innfylte data gitt av brukeren.		
E1.3	Når du nå har laget deg en bruker, prøv å logg inn.	Prosjektveilederen tillater at brukeren logger seg på, og tofaktor autentisering skal være påkrevd.		

E2 INNLOGGET BRUKER

Hovedtema	Oppgave	Ønsket resultat	Oppnådd resultat/diskusjon	Forbedringspotensial
E2.1	Finn fram til egen brukerinformasjon.	Brukeren finner frem til egen informasjon innen 30 sekunder.		

E4 PROSJEKTVEILEDER

E4.1 BRUKER-/PROSJEKTHÅNDTERING

Hovedtema	Oppgave	Ønsket resultat	Oppnådd resultat/diskusjon	Forbedringspotensial
E4.1.1	Prøv å opprett et nytt prosjekt.	Brukeren oppretter et nytt prosjekt.		

E4.1.2	Hva trenger du av informasjon og tips nå? Hvordan finner du det du trenger?	Brukeren finner frem til tips/informasjon gitt i de		
--------	---	---	--	--

		forskjellige stegene		
E4.1.3	Føler du at utformingen av "kom i gang" siden er god med tanke på at siden skal gi en introduksjon til Nysgjerrigpermetoden.	Brukeren skal kunne få en introduksjon til Nysgjerrigpermetoden.		
E4.1.4	Får du veiledning til steget og mulighet til å legge inn problemstilling i steg 1?	Brukeren skal kunne få veiledning til steget og mulighet til å legge inn problemstilling.		
E4.1.5	Får du informasjon om hva som kan være gode hypoteser, og mulighet til å fylle inn tekst i steg 2?	Brukeren skal kunne få veiledning til steget og mulighet til å føre inn en eller flere hypoteser.		
E4.1.6	Får du informasjon/tips om hvordan de ulike metodene kan benyttes, og mulighet til å velge metoder i steg 3?	Brukeren skal kunne få veiledning til steget og mulighet til å velge en eller flere metoder for testing av hypotesene valgt i steg 2.		
E4.1.7	Har du mulighet til å fylle inn tekst i steg 4?	Brukeren skal kunne få veiledning til steget og mulighet til å beskrive gjennomføring av planen lagt.		
E4.1.8	Får du fylt inn tekst og valgt resultat i steg 5?	Brukeren skal kunne få veiledning til steget og mulighet til å konkludere per		

		hypotese, same konkludere på problemstillingen på bakgrunn av arbeidet gjort med hypotesene.		
--	--	--	--	--

E4.1.9	Har du mulighet til å fylle inn tekst for introduksjon av klassen og beskrivelse av prosessen i steg 6?	Brukeren skal kunne få veiledning til steget og mulighet til å skrive en introduksjon og en beskrivelse av gjennomføringen av prosjektet.		
E4.1.10	Er det mulig å se et utkast av rapporten (altså en prosjektoppsummering)?	Prosjektveilederen skal vise en presentasjon av prosjektet.		
E4.1.11	Hvis du skal legge til et bilde underveis i prosessen, hvordan vil du løse det? Prøv å legge til bilder i prosjektet.	Når brukeren velger å laste opp et bilde til et steg i metoden, så skal det legges inn ett bilde knyttet til stedet jeg velger å laste det opp.		
E4.1.12	Hvordan vil du gå frem for å endre innhold i rapporten underveis?	Brukeren endrer innhold i rapporten.		
E4.1.13	Hvordan vil du gå frem for å slette innhold i rapporten underveis?	Brukeren sletter innhold i rapporten.		
E4.1.14	Hva ble du møtt av når du prøvde å laste opp et bilde?	Brukeren blir bedt om å innhente samtykke.		
E4.1.15	Hvordan vil du gå frem for å slette opplastet innhold (for eksempel bilder)?	Brukeren sletter opplastet innhold fra rapporten.		

E4.2 PROSJEKTVEILEDEREN SKAL HA EN PROSJEKTOVERSIKT

Hovedtema	Oppgave	Ønsket resultat	Oppnådd resultat/diskusjon	Forbedringspotensial
E4.2.1	Hvordan vil du gå fram for å se en oversikt over dine prosjekter?	Brukeren finner fram til oversikten over sine prosjekter.		
E4.2.2	Hvordan vil du gå fram for å åpne et av dine allerede eksisterende prosjekter?	Brukeren åpner et av sine allerede eksisterende prosjekter		

AVSLUTTENDE SPØRSMÅL TIL BRUKERNE

Syns du navigeringen på siden var intuitiv? - Er det lett å forstå hva slags muligheter du har i applikasjonen?

Har du noen tips til forbedringer eller andre kommentarer?

6.6.3 BRUKERTEST: ANONYM

Vi skal gjennomføre brukertesting sammen med fem lærere. Brukertesting vil hjelpe oss å kartlegge mulig forbedringspotensial og eventuelle mangler.

TESTLEDER

- Vegard Skullestad Bjørkgård.
- Testlederen har ansvar for utførelsen av testene.
- Testlederen forklarer prosessen, intensjonen og målet for testingen til brukerne.
- Testlederen forklarer at det er applikasjonen, ikke brukeren som vil bli testet og evaluert.
- Testlederen oppfordrer brukerne til å snakke og fortelle om det de tenker høyt.
- Stiller spørsmålene/gir oppgavene og instruksjoner.

OBSERVATØR

- Vegard Skullestad Bjørkgård.
- Tar notater og svarer på eventuelle spørsmål som blir stilt underveis.
- Referenten skriver ned alt det brukeren gjør for å få til den tildelte oppgaven.

INTRODUKSJON

Vi holder på å lage en ny applikasjon for Nysgjerrigper som skal senke terskelen for at lærere velger å ta i bruk Nysgjerrigpermetoden. Det er viktig for oss å få testet kravene som er satt for applikasjonen. Og den beste måten å teste disse kravene på er ved å teste dem på sluttbrukerne, som i dette tilfelle er lærere. For deg som tester er det viktig å tenke på at det er applikasjonen som testes å ikke deg som bruker.

Testen vil utføres som følger:

- Innledende spørsmål om dagens løsning og første inntrykk av den nye applikasjonen.
- Læreren vil bli gitt praktiske oppgaver som han/hun vil prøve å løse etter beste evne.
- Spørsmål om løsningen og tilbakemeldinger (generelle og spesifikke).

INNLEDENDE SPØRSMÅL

Benytter du en form for forskningsmetode i undervisningen i dag?

Nei. Ikke per nå.

Dersom du har benyttet Nysgjerrigpermetoden, vil du si at dagens løsning er tilstrekkelig?

Nei.

FØRSTEINNTRYKK

Hva ser du her?

Landingsside med påloggingsmulighet.

PRAKTISKE OPPGAVER

Du vil bli spurt om å navigere deg rundt på siden, registrere bruker, logge på bruker, opprette prosjekt og fylle inn tekst i prosjektet.



Figur 173 Landingsiden til Prosjektveilederen

E1 INNLOGGING OG SIKKERHET

Hoved-tema	Oppgave	Ønsket resultat	Oppnådd resultat/diskusjon	Forbedringspotensial
E1.1	Føler du at landingssiden du havner på før du har logget inn i applikasjonen er godt utformet med tanke på den skal gi en kort introduksjon til metoden og hva som må gjøres for å komme i gang?	Testpersonen forklarer hvordan han/hun oppfatter at han/hun skal gå frem, og forklaringen stemmer med en måte som vil løse oppgaven	Litt vanskelig å forstå at det er mer informasjon på denne siden. Litt uklart at det er mere informasjon tilgjengelig.	Bør gjøre det mer tydelig at det er mer informasjon lenger ned på siden.
E1.2	Lag deg en bruker.	Det skal opprettes en bruker basert på innfylte data gitt av brukeren.	Ikke testet.	
E1.3	Når du nå har laget deg en bruker, prøv å logg inn.	Prosjektveilederen tillater at brukeren logger seg på, og tofaktor autentisering skal være påkrevd.	(2 faktor juksa til) Ikke autofokus. Masse passord som må fjernes før innføring. Ingen tilbakemelding på at man er blitt logget inn/ikke.	Bør implementeres autofokus på første tekstfelt. Gjøre det tydeligere at man har logget inn/ikke.

E2 INNLOGGET BRUKER

Hoved-tema	Oppgave	Ønsket resultat	Oppnådd resultat/diskusjon	Forbedringspotensial
E2.1	Finn fram til egen brukerinformasjon.	Brukeren finner frem til egen informasjon innen 30 sekunder.	Funka	Ingen

E4 PROSJEKTVEILEDER

E4.1 BRUKER-/PROSJEKTHÅNDTERING

Hoved-tema	Oppgave	Ønsket resultat	Oppnådd resultat/diskusjon	Forbedringspotensial
E4.1.1	Prøv å opprett et nytt prosjekt.	Brukeren oppretter et nytt prosjekt.	Gikk fint.	Ingen
E4.1.2	Hva trenger du av informasjon og tips nå? Hvordan finner du det du trenger?	Brukeren finner frem til tips/informasjon gitt i de forskjellige stegene	Informasjonen i toppen blir oversett. Går rett til neste overskrift.	Implementere en mer intuitiv tips/informasjon knapp.

E4.1.3	Føler du at utformingen av "kom i gang" siden er god med tanke på at siden skal gi en introduksjon til Nysgjerrigpermetoden.	Brukeren skal kunne få en introduksjon til Nysgjerrigper-metoden.	Spørsmål må omskrives.	
E4.1.4	Får du veiledning til steget og mulighet til å legge inn problemstilling i steg 1?	Brukeren skal kunne få veiledning til steget og mulighet til å legge inn problemstilling.	Ja. Tips til lærer ikke implementert.	
E4.1.5	Får du informasjon om hva som kan være gode hypoteser, og mulighet til å fylle inn tekst i steg 2?	Brukeren skal kunne få veiledning til steget og mulighet til å føre inn en eller flere hypoteser.	Tips til lærer ikke implementert. Gjerne automatisk stor bokstav først i inputfeltet.	Kan implementere løsning som automatisk setter stor bokstav i hver setning som skrives.
E4.1.6	Får du informasjon/tips om hvordan de ulike metodene kan benyttes, og mulighet til å velge metoder i steg 3?	Brukeren skal kunne få veiledning til steget og mulighet til å velge en eller flere metoder for testing av hypotesene valgt i steg 2.	Velg først en metode panelet bør være det uthevede. Litt mer veiledning/tilbakemelding ved tillegg av metoder. Ingen informasjon om at man kan legge til en eller flere. Tilbakemelding når steget per metode er gjennomført.	Metodepanelet bør være åpent til enhver tid. Bedre tilbakemelding ved opprettelse av metoder. Gi tilbakemelding når hver enkelt metode er gjennomført.
E4.1.7	Har du mulighet til å fylle inn tekst i steg 4?	Brukeren skal kunne få veiledning til steget og mulighet til å beskrive gjennomføring av planen lagt.	Autofokus. Knapp for lagring misvisende. Uklart hva man skal gjøre for å lagre/endre. Ingen informasjon om hvilke metoder/hypoteser steget er gjennomført for.	Bør implementeres autofokus i første tekstfelt. Bør gjøres mer intuitivt hva man skal gjøre for å lagre/endre data. Bør gis tilbakemelding på hvilke metode/hypotese som gjennomføres.
E4.1.8	Får du fylt inn tekst og valgt resultat i steg 5?	Brukeren skal kunne få veiledning til steget og mulighet til å konkludere per hypotese, same konkludere på problemstillingen på bakgrunn av arbeidet gjort med hypotesene.	Fremgangsmåten som det er ment at man skal bruke i dette steget må beskrives på starten av siden. Intro til «første og andre steg» litt lange, og kan være mer uthøvet. Teksten tar bare imot vanlige bokstaver.	Bør beskrive fremgangsmåten for steget på toppen av siden. Komprimere introduksjonen til første og andre steg, samt fremheve den.

				Teksten bør kunne ta imot spesialtegn og tall.
E4.1.9	Har du mulighet til å fylle inn tekst for introduksjon av klassen og beskrivelse av prosessen i steg 6?	Brukeren skal kunne få veiledning til steget og mulighet til å skrive en introduksjon og en beskrivelse av gjennomføringen av prosjektet.	Ja. Ikke tips.	Ingen
E4.1.10	Er det mulig å se et utkast av rapporten (altså en prosjektoppsummering)?	Prosjektveilederen skal vise en presentasjon av prosjektet.	Ikke implementert.	
E4.1.11	Hvis du skal legge til et bilde underveis i prosessen, hvordan vil du løse det? Prøv å legge til bilder i prosjektet.	Når brukeren velger å laste opp et bilde til et steg i metoden, så skal det legges inn ett bilde knyttet til stedet jeg velger å laste det opp.	Ikke implementert.	
E4.1.12	Hvordan vil du gå frem for å endre innhold i rapporten underveis?	Brukeren endrer innhold i rapporten.	Ikke implementert, men diskutert underveis.	
E4.1.13	Hvordan vil du gå frem for å slette innhold i rapporten underveis?	Brukeren sletter innhold i rapporten.	Ikke implementert.	

E4.1.14	Hva ble du møtt av når du prøvde å laste opp et bilde?	Brukeren blir bedt om å innhente samtykke.	Ikke implementert.	
E4.1.15	Hvordan vil du gå frem for å slette opplastet innhold (for eksempel bilder)?	Brukeren sletter opplastet innhold fra rapporten.	Ikke implementert.	

E4.2 PROSJEKTVEILEDEREN SKAL HA EN PROSJEKTOVERSIKT

Hovedtema	Oppgave	Ønsket resultat	Oppnådd resultat/diskusjon	Forbedringspotensial
E4.2.1	Hvordan vil du gå fram for å se en oversikt over dine prosjekter?	Brukeren finner fram til oversikten over sine prosjekter.	Ikke implementert.	
E4.2.2	Hvordan vil du gå fram for å åpne et av dine allerede eksisterende prosjekter?	Brukeren åpner et av sine allerede eksisterende prosjekter	Ikke implementert.	

AVSLUTTENDE SPØRSMÅL TIL BRUKERNE

Syns du navigeringen på siden var intuitiv? - Er det lett å forstå hva slags muligheter du har i applikasjonen?

Diskutert underveis. Navigering forståelig. Muligheter og hva man skal gjennomføre per steg uklart.

Har du noen tips til forbedringer eller andre kommentarer?

Tips til forbedringer diskutert og skrevet inn underveis. Ordbruk på knapper for «lagre/legg til» «endre» osv. bør være likt for alle input muligheter underveis. Klarere.

6.6.4 BRUKERTEST: KATHERINE

Vi skal gjennomføre brukertesting sammen med lærere. Brukertesting vil hjelpe oss å kartlegge mulig forbedringspotensial og eventuelle mangler.

TESTLEDER

- Yanislav Zagorov.
- Testlederen har ansvar for utførelsen av testene.
- Testlederen forklarer prosessen, intensjonen og målet for testingen til brukerne.
- Testlederen forklarer at det er applikasjonen, ikke brukeren som vil bli testet og evaluert.
- Testlederen oppfordrer brukerne til å snakke og fortelle om det de tenker høyt.
- Stiller spørsmålene/gir oppgavene og instruksjoner.

OBSERVATØR

- Olav Sørli.
- Tar notater og svarer på eventuelle spørsmål som blir stilt underveis.
- Referenten skriver ned alt det brukeren gjør for å få til den tildelte oppgaven.

INTRODUKSJON

Vi holder på å lage en ny applikasjon for Nysgjerrigper som skal senke terskelen for at lærere velger å ta i bruk Nysgjerrigpermetoden. Det er viktig for oss å få testet kravene som er satt for applikasjonen. Og den beste måten å teste disse kravene på er ved å teste dem på sluttbrukerne, som i dette tilfelle er lærere. For deg som tester er det viktig å tenke på at det er applikasjonen som testes å ikke deg som bruker.

Testen vil utføres som følger:

- Innledende spørsmål om dagens løsning og første inntrykk av den nye applikasjonen.
- Læreren vil bli gitt praktiske oppgaver som han/hun vil prøve å løse etter beste evne.
- Spørsmål om løsningen og tilbakemeldinger (generelle og spesifikke).

INNLEDENDE SPØRSMÅL

Benytter du en form for forskningsmetode i undervisningen i dag?

Syns det er masse jobb bare å komme i gang med Nysgjerrigpermetoden. Mye som skal være på plass før man kan starte et prosjekt.

Dersom du har benyttet Nysgjerrigpermetoden, vil du si at dagens løsning er tilstrekkelig?

FØRSTEINTRYKK

Hva ser du her?

Hva er dette er det første hun tenker. Litt godt gjemt på siden den Les mer knappen på Kom i gang siden.

PRAKTISKE OPPGAVER

Du vil bli spurt om å navigere deg rundt på siden, registrere bruker, logge på bruker, opprette prosjekt og fylle inn tekst i prosjektet.



Figur 174 Landingssiden til Prosjektveilederen

E1 INNLOGGING OG SIKKERHET

Hovedtema	Oppgave	Ønsket resultat	Oppnådd resultat/diskusjon	Forbedringspotensial
E1.1	Føler du at landingssiden du havner på før du har logget inn i applikasjonen er godt utformet med tanke på den skal gi en kort introduksjon til metoden og hva som må gjøres for å komme i gang?	Testpersonen forklarer hvordan han/hun oppfatter at han/hun skal gå frem, og forklaringen stemmer med en måte som vil løse oppgaven	Hva er dette er det første hun tenker. Litt godt gjemt på siden den Les mer knappen på Landingssiden siden. Litt godt skjult.	Les mer knappen bør være mer synlig.
E1.2	Lag deg en bruker.	Det skal opprettes en bruker basert på innfylte data gitt av brukeren.	Bør være en tilbakeknapp til landingssiden/forrige side. Passordkravene er litt for sterke. Mulighet for å se hva man har skrevet inn i passordfeltet gjerne med et øyeikon.	Ønsker tilbakeknapp til forrige side. Justere ned styrken på passordet.
E1.3	Når du nå har laget deg en bruker, prøv å logg inn.	Prosjektveiledere n tillater at brukeren logger seg på, og tofaktor autentisering skal være påkrevd.	Påloggingen ser ut til å gå fint. Kanskje utheve overskriften litt.	Utheve overskriften litt.

E2 INNLOGGET BRUKER

Hovedtema	Oppgave	Ønsket resultat	Oppnådd resultat/diskusjon	Forbedringspotensial
E2.1	Finn fram til egen brukerinformasjon.	Brukeren finner frem til egen informasjon innen 30 sekunder.	Det gikk ganske greit å komme fram til brukerinformasjon. Ingen spesifikke kommentarer	Ingen

E4 PROSJEKTVEILEDER

E4.1 BRUKER-/PROSJEKTHÅNDTERING

Hovedtema	Oppgave	Ønsket resultat	Oppnådd resultat/diskusjon	Forbedringspotensial
E4.1.1	Prøv å opprett et nytt prosjekt.	Brukeren oppretter et nytt prosjekt.	Ikke like lett å finne fram til prosjektopprettelsen. Prosjektnavnet er ikke med i dette feltet. Prosjektet bør ha et navn, noe som tilsynelatende ser ut som gruppens navn sånn det er nå. Kanskje heller ha bekreftelses-vindu istedenfor informasjonen man akkurat har fylt inn her.	Legge til navn i prosjekt. Bytte ut informasjonssiden med et bekreftelsesvindu.
E4.1.2	Hva trenger du av informasjon og tips nå? Hvordan finner du det du trenger?	Brukeren finner frem til tips/informasjon gitt i de forskjellige stegene		
E4.1.3	Føler du at utformingen av "kom i gang" siden er god med tanke på at siden skal gi en introduksjon til Nysgjerrigmetoden.	Brukeren skal kunne få en introduksjon til Nysgjerrigmetoden.	Litt lite intuitiv i forhold til at det er vanskelig å vite at man har mer på siden, da GDPR feltet og det ikke er noe tydelighet i at det er mer lenger ned på siden.	Gjøre utformingen mer intuitiv, slik at det blir mer tydelig at det finnes mer informasjon lenger ned på siden.

E4.1.4	Får du veiledning til steget og mulighet til å legge inn problemstilling i steg 1?	Brukeren skal kunne få veiledning til steget og mulighet til å legge inn problemstilling.	Problemstillingen har man allerede valgt seg ut før man starter opp et prosjekt. Så å velge seg en problemstilling blir litt i overkant. Ønsker mulighet for å endre i stegene	Bør ha annen beskrivelse på hypotesene enn problemstilling, da prosjektet har en problemstilling.
E4.1.5	Får du informasjon om hva som kan være gode hypoteser, og mulighet til å fylle inn tekst i steg 2?	Brukeren skal kunne få veiledning til steget og mulighet til å føre inn en eller flere hypoteser.	Rammen rundt informasjon om hvorfor er det slik er rød, noe som kan føre til at brukeren tenker at det er noe som er glemt. Man kobler ikke sammen at problemstillingen henger sammen med hypotesen. Ikke mulig å fjerne hypoteser. Fjerne at det står nye hypoteser. Bedre å skrive at her kan man skrive sine hypoteser.	Bør ha en annen farge en rødt rundt Usannsynlig. Samme som E4.1.5. Bør være mulig å fjerne hypoteser. Bør ikke stå nye hypoteser.
E4.1.6	Får du informasjon/tips om hvordan de ulike metodene kan benyttes, og mulighet til å velge metoder i steg 3?	Brukeren skal kunne få veiledning til steget og mulighet til å velge en eller flere metoder for testing av hypotesene valgt i steg 2.	Hvordan skal dette gjøres tro? Litt usikker på hva som skal trykkes på. Både panelet som trykkes på for å åpne metodevalgene. Og selve metoden. Samtidig vanskelig å finne tilbake til metodene som er valgt. Da de er skjult bak enda et panel. Det er ikke intuitivt at man må trykke på panelene.	Panelene bør være åpne til enhver tid. Fortelle brukeren at plan må legges før metoder kan velges. Bør være mer åpenbart hvor det er man fyller ut selve utførelsen av selve undersøkelsen. Bør være mer plass til å fylle inn

			<p>Ingenting forteller heller brukeren at først må man legge en plan for så å velge metoden. Litt vanskelig å vite hvor man skal fylle ut utførelsen av selve undersøkelsen. Vil gjerne ha litt mer plass til å fylle inn planleggingsbiten. Det mangler en knapp for neste steg og gå tilbake og endre i modalen for trinn 3. Bør lagres når man hopper mellom stegene i venstre navigasjonsbar, og ikke ha behov for å trykke på lagre i bunn av steget.</p>	<p>planleggingsdelen .</p> <p>Ønsker knapp for neste og tilbake i modalen til trinn 3.</p> <p>Autolagring bør implementeres.</p>
E4.1.7	Har du mulighet til å fylle inn tekst i steg 4?	<p>Brukeren skal kunne få veiledning til steget og mulighet til å beskrive gjennomføring av planen lagt.</p>	<p>Det står hent opplysninger som overskrift og samle informasjon rett under, bør rettes opp. To forskjellige ting sier norsklærer. Tips knappen ser ikke ut som en knapp. Bør være tydeligere at det er informasjon og hente i dette panelet. Foreslår at man bør kunne ta opp lyd og bilde i intervju delen. Om ikke bilde så hvert fall ta opp lyd. Bør være en knapp for å kunne gå tilbake også her. Ønsker</p>	<p>Endre ord «samle informasjon», alternativt «hent opplysninger».</p> <p>Endre design på tips knappen.</p> <p>Bør være mulig å ta opp lyd i intervju metoden.</p> <p>Bør være en tilbake-knapp også her.</p> <p>Bør være mer fleksibilitet når man beveger seg mellom punktene.</p>

			mer fleksibilitet mellom punktene.	
E4.1.8	Får du fylt inn tekst og valgt resultat i steg 5?	Brukeren skal kunne få veiledning til steget og mulighet til å konkludere per hypotese, same konkludere på problemstillingen på bakgrunn av arbeidet gjort med hypotesene.	Liker at det er forskjellige farger på siden i stegene. Bedre å liste opp i beskrivelsen. Dette gjør det lettere for barn å lese. Ville likt å se animasjoner. Lurer på hvorfor det står sannsynlig og usannsynlig, bør kanskje heller stå Ja eller Nei. Vanskelig å åpne panelene, ikke intuitivt at man må klikke seg inn i disse. Bildeknappen sier ikke så veldig masse. Fint at den er der dog.	Bør implementere teksten som lister, gjør det lettere for barn å lese. Siden bør inneholde animasjoner. Bør stå «Ja» og «Nei» istedenfor «Sannsynlig» og «Usannsynlig». Panelene bør være åpne hele tiden. Bør gi litt mer informasjon rundt bilder.
E4.1.9	Har du mulighet til å fylle inn tekst for introduksjon av klassen og beskrivelse av prosessen i steg 6?	Brukeren skal kunne få veiledning til steget og mulighet til å skrive en introduksjon og en beskrivelse av gjennomføringen av prosjektet.	Vil ha en slags forhånds-visning på lagringen og ha en tilbakeknapp. Ønsker seg en tilbakemelding på lagring eller at man bare går videre til neste steg ved å trykke lagre. To knapper er i overkant. Syns det er bedre om man kan skrive om gruppen osv. helt i starten av prosjektet. Trudde	Bør være forhånds-visning på rapporten. Bør gis en tilbakemelding på lagring. Bør holde med en knapp for lagring og lukking. Bør skrive om gruppen helt i begynnelsen istedenfor helt i

			hun allerede har skrevet om gruppen tidligere.	slutten av rapporten.
E4.1.1 0	Er det mulig å se et utkast av rapporten (altså en prosjektoppsummering)?	Prosjektveilederen skal vise en presentasjon av prosjektet.		
E4.1.1 1	Hvis du skal legge til et bilde underveis i prosessen, hvordan vil du løse det? Prøv å legge til bilder i prosjektet.	Når brukeren velger å laste opp et bilde til et steg i metoden, så skal det legges inn ett bilde knyttet til stedet jeg velger å laste det opp.	Bilde-bekreftelsesvinduet er litt i overkant. Den kan være litt forvirrende der den ligger nå. Det kommer ikke så tydelig frem at teksten hører til et eventuelt bilde. Lite intuitivt hele bildeopplaster grensesnittet. Bildet blir litt lite etter man har valgt det. Forslag til at det kan være en popup istedenfor en liten thumbnail.	Bør redesigne bilde-opplasteren. Gjør bildet litt større når man har lastet det opp. Bilde-opplasteren kunne vært en popup istedenfor en liten thumbnail.
E4.1.1 2	Hvordan vil du gå frem for å endre innhold i rapporten underveis?	Brukeren endrer innhold i rapporten.	Auto lagring ikke implementert, fører til problemer under innføring i prosjektet, brukeren navigerer mellom vinduene ved å benytte stegene på høyre side av prosjektveilederen. Teksten lagres kun dersom brukeren	Prosjektveilederen bør ha autolagring.

			trykker på lagreknapp nederst i hver komponent som inneholder tekstfelter.	
E4.1.1 3	Hvordan vil du gå frem for å slette innhold i rapporten underveis?	Brukeren sletter innhold i rapporten.	Ikke implementert.	
E4.1.1 4	Hva ble du møtt av når du prøvde å laste opp et bilde?	Brukeren blir bedt om å innhente samtykke.	Ikke implementert.	
E4.1.1 5	Hvordan vil du gå frem for å slette opplastet innhold (for eksempel bilder)?	Brukeren sletter opplastet innhold fra rapporten.	Ikke implementert.	

E4.2 PROSJEKTVEILEDEREN SKAL HA EN PROSJEKTOVERSIKT

Hovedtema	Oppgave	Ønsket resultat	Oppnådd resultat/diskusjon	Forbedringspotensial
E4.2.1	Hvordan vil du gå fram for å se en oversikt over dine prosjekter?	Brukeren finner fram til oversikten over sine prosjekter.	Ingen kommentarer på dette. Virket intuitivt for brukeren	Ingen
E4.2.2	Hvordan vil du gå fram for å åpne et av dine allerede	Brukeren åpner et av sine allerede eksisterende prosjekter	Ingen kommentarer på dette. Virket	Ingen

	eksisterende prosjekter?		intuitivt brukeren	for	
--	-----------------------------	--	-----------------------	-----	--

AVSLUTTENDE SPØRSMÅL TIL BRUKERNE

Syns du navigeringen på siden var intuitiv? - Er det lett å forstå hva slags muligheter du har i applikasjonen?

Diskutert underveis. Litt lite oversiktlig hva som er hva når man har vært gjennom alle stegene. Steg 1 er helt låst, man kan ikke endre noe, dette bør man ha mulighet til. Syns man bør ha en nesteknapp i flere av stegene. Bildet kommer opp når man velger det i filutforskeren. Ønsker at man har mulighet til å endre på bildestørrelse, bilder og innhold i Se rapport vinduet. Gjerne mulighet til å hoppe mellom vinduer.

Har du noen tips til forbedringer eller andre kommentarer?

Diskutert underveis. Forslag til maskot som er animert på siden, kan gjøre det litt morsommere for barna.

6.6.5 BRUKERTEST: RAGNHILD

Vi skal gjennomføre brukertesting sammen med lærere. Brukertesting vil hjelpe oss å kartlegge mulig forbedringspotensial og eventuelle mangler.

TESTLEDER

- Vegard Skullestad Bjørkgård.
- Testlederen har ansvar for utførelsen av testene.
- Testlederen forklarer prosessen, intensjonen og målet for testingen til brukerne.
- Testlederen forklarer at det er applikasjonen, ikke brukeren som vil bli testet og evaluert.
- Testlederen oppfordrer brukerne til å snakke og fortelle om det de tenker høyt.
- Stiller spørsmålene/gir oppgavene og instruksjoner.

OBSERVATØR

- Olav Sørli.
- Tar notater og svarer på eventuelle spørsmål som blir stilt underveis.
- Referenten skriver ned alt det brukeren gjør for å få til den tildelte oppgaven.

INTRODUKSJON

Vi holder på å lage en ny applikasjon for Nysgjerrigper som skal senke terskelen for at lærere velger å ta i bruk Nysgjerrigpermetoden. Det er viktig for oss å få testet kravene som er satt for applikasjonen. Og den beste måten å teste disse kravene på er ved å teste dem på sluttbrukerne, som i dette tilfelle er lærere. For deg som tester er det viktig å tenke på at det er applikasjonen som testes å ikke deg som bruker.

Testen vil utføres som følger:

- Innledende spørsmål om dagens løsning og første inntrykk av den nye applikasjonen.
- Læreren vil bli gitt praktiske oppgaver som han/hun vil prøve å løse etter beste evne.
- Spørsmål om løsningen og tilbakemeldinger (generelle og spesifikke).

INNLEDENDE SPØRSMÅL

Benytter du en form for forskningsmetode i undervisningen i dag?

Benytter vanlige forsøksrapporter og et interaktivt verktøy for å lage videoer.

Dersom du har benyttet Nysgjerrigpermetoden, vil du si at dagens løsning er tilstrekkelig?

Har ikke benyttet Nysgjerrigpermetoden i undervisningen tidligere.

FØRSTEINTRYKK

Hva ser du her?

En skjerm med pålogging, og ny bruker. Personvernserklæringa er litt i veien, antyder at det er slutten av siden. Forslag: Kanskje kutte det lilla feltet på landingssiden litt, slik at man ser at det er mer nedover.

PRAKTISKE OPPGAVER

Du vil bli spurt om å navigere deg rundt på siden, registrere bruker, logge på bruker, opprette prosjekt og fylle inn tekst i prosjektet.



Figur 175 Landingssiden til Prosjektveilederen

E1 INNLOGGING OG SIKKERHET

Hovedtema	Oppgave	Ønsket resultat	Oppnådd resultat/diskusjon	Forbedringspotensial
E1.1	Føler du at landingssiden du havner på før du har logget inn i applikasjonen er godt utformet med tanke på den skal gi en kort introduksjon til metoden og hva som må gjøres for å komme i gang?	Testpersonen forklarer hvordan han/hun oppfatter at han/hun skal gå frem, og forklaringen stemmer med en måte som vil løse oppgaven	Vanskelig å se at det er mer på siden. Litt liten sti mellom punktene.	Kan flytte det lille feltet litt lenger opp slik at man ser at det er mer innhold nedover på siden. Kanskje den lille siden kan bli til en overskrift når man scroller nedover.
E1.2	Lag deg en bruker.	Det skal opprettes en bruker basert på innfylte data gitt av brukeren.	Litt i overkant sterkt passord. Hopper tilbake til landings-siden etter pålogging, vanskelig å se at man er pålogget når den ser omtrent helt lik ut som før pålogging.	Gjøre det litt mer åpenbart at man er pålogget etter pålogging, for eksempel ved å la bruker-menyen komme frem på landingssiden.
E1.3	Når du nå har laget deg en bruker, prøv å logg inn.	Prosjektveilederen tillater at brukeren logger seg på, og tofaktor autentisering skal være påkrevd.	Påloggingen gikk greit, litt vanskelig å se at man har blitt pålogget.	Samme som E1.2.

E2 INNLOGGET BRUKER

Hovedtema	Oppgave	Ønsket resultat	Oppnådd resultat/diskusjon	Forbedringspotensial
E2.1	Finn fram til egen brukerinformasjon.	Brukeren finner frem til egen informasjon innen 30 sekunder.	Går ganske greit, nevner at det er Hent Opplysninger er skrevet feil.	Rett opp i skrivefeil i Hent Opplysninger

E4 PROSJEKTVEILEDER

E4.1 BRUKER-/PROSJEKTHÅNDTERING

Hovedtema	Oppgave	Ønsket resultat	Oppnådd resultat/diskusjon	Forbedringspotensial
E4.1.1	Prøv å opprett et nytt prosjekt.	Brukeren oppretter et nytt prosjekt.	Forslag til at det kan være et plusstegn for nytt prosjekt.	Implementere et plusstegn for nye prosjekt på navigasjons-linjen øverst i prosjektveilederen.
E4.1.2	Hva trenger du av informasjon og tips nå? Hvordan finner du det du trenger?	Brukeren finner frem til tips/informasjon gitt i de forskjellige stegene	Tips knappen kan kanskje ha en annen farge, og kanskje være litt større. At siden har veldig mange forskjellige fonter blir nevnt.	En mer fremhevet/intuitiv tips knapp. Minimere antall fonter på siden.
E4.1.3	Føler du at utformingen av "kom i gang" siden er god med tanke på at siden skal gi en introduksjon til Nysgjerrigpermetoden.	Brukeren skal kunne få en introduksjon til Nysgjerrigpermetoden.	Forslag til at man ser at man er logget inn ved at brukermenyen blir med i kom i gang siden når man har logget inn.	Samme som E1.2
E4.1.4	Får du veiledning til steget og mulighet til å legge inn problemstilling i steg 1?	Brukeren skal kunne få veiledning til steget og mulighet til å legge inn problemstilling.	Litt stygt design på informasjonssidene man kommer på etter at man har fylt inn informasjon og. Behøver kanskje heller ikke å være en like stegvis prosess ved at man blir vist informasjon om hva man har fylt inn i hvert steg.	Forslag til at informasjonssidene gjøres penere, eller fjernes.

E4.1.5	Får du informasjon om hva som kan være gode hypoteser, og mulighet til å fylle inn tekst i steg 2?	Brukeren skal kunne få veiledning til steget og mulighet til å føre inn en eller flere hypoteser.	Må ha litt klarere instruksjoner for hva de forskjellige knappene gjør. Forslag om å kunne endre hypotesen direkte. Forslag til tilbakeknapp i tillegg til neste steg.	Bedre instruksjoner for de forskjellige knappene. Kan legge inn mulighet til å endre hypotesene direkte i tekstboksen. Legge til en tilbakeknapp.
E4.1.6	Får du informasjon/tips om hvordan de ulike metodene kan benyttes, og mulighet til å velge metoder i steg 3?	Brukeren skal kunne få veiledning til steget og mulighet til å velge en eller flere metoder for testing av hypotesene valgt i steg 2.	Istedenfor at man bare hadde en hvor teksten sto, så kunne man hatt flere, en som het intervju, en som het observasjon etc. på toppen i tips steget. Så ikke tipsene over de forskjellige hypotesene. Alle metode tipsene bør være i toppen. Dveler litt ved at det er to forskjellige fonter i tipsene til metodene. Forslag til at vi benytter hypotese teksten som overskrift i de forskjellige metode typene. Forslag til opplasting av filer og ikke bare bilder i de forskjellige stegene. Litt tydeligere når man har valgt en metode og mulighet til å	Forslag til at man kan ha alle metodetipsene i toppen av steg 3 istedenfor at de ligger inni hver enkelt metode. Hypotese-teksten bør være overskriften i metode popupen. Applikasjonen bør støtte opplasting av filer generelt ikke bare bilder. Bør være mulig å fjerne metoder etter de er valgt. Bør være tydeligere at en metode er valgt. Må fikse feil ved henting av informasjon fra Confluence, observasjonstipsene kommer opp i forsøksdelen

			fjerne metoder. Feil i henting av informasjon fra Confluence, observasjonstips ene kommer opp i forsøksdelen. Når man velger et steg, så kan det være at noen blir forvirret når store tall forstørres og teksten forsvinner.	Bør beholde teksten i stegmenyen når den er valgt.
E4.1.7	Har du mulighet til å fylle inn tekst i steg 4?	Brukeren skal kunne få veiledning til steget og mulighet til å beskrive gjennomføring av planen lagt.	Fint at teksten fylt inn i metoden kommer tilbake i steg 4. Litt tungvint at man både må trykke lagre og lukke. Kanskje knappen burde hete lagre og lukke. Kanskje likegreit at metodeplanen ligger åpen, med et lite scrollefelt. Istedenfor at man har et vindu som kollapser.	Bør ha en felles knapp som både lagrer og lukker. Panelet bør ligge åpent, ikke kollapset.
E4.1.8	Får du fylt inn tekst og valgt resultat i steg 5?	Brukeren skal kunne få veiledning til steget og mulighet til å konkludere per hypotese, same konkludere på problemstillingen på bakgrunn av arbeidet gjort med hypotesene.	Samme her bør også det skrevet i metoden komme opp med en gang man åpner panelet. Litt mye klikking. Liker effekten når man velger sannsynlig, vet ikke og usannsynlig. Fin touch at rammene forblir på hypotesene når man har valgt sannsynlig etc.	Panelet bør være åpent til enhver tid. Bør ha mulighet til å legge inn så mange bilder man har lyst til.

			blir med når man går tilbake til foregående steg. Syns også man bør ha mulighet til å legge inn så mange bilder som man har lyst til her.	
E4.1.9	Har du mulighet til å fylle inn tekst for introduksjon av klassen og beskrivelse av prosessen i steg 6?	Brukeren skal kunne få veiledning til steget og mulighet til å skrive en introduksjon og en beskrivelse av gjennomføringen av prosjektet.	Kanskje det bør stå noe i heading teksten at man kan motivere litt, for eksempel kanskje sende til lokalavisen. Det burde være mulig at disse lages hver for seg, slik at man kan lage den ene den ene dagen også lage den andre en annen dag for eksempel. Når man lagre kanskje gå rett til neste vindu.	Motivere litt i steg 6 steget. Burde være mulig å lagre tekstfeltene hver for seg.
E4.1.10	Er det mulig å se et utkast av rapporten (altså en prosjektoppsummering)?	Prosjektveilederen skal vise en presentasjon av prosjektet.	Forslag til at man har en mulighet til å skrive ut fra denne siden. For å gjøre det enklere. Forslag til at man kan velge skrifttyper og problematikk med formatering. Forslag til at det er mer en forside, for så at informasjonen kommer inn. Forslag til at man deler inn i sider. Barna vil kanskje si at uten forside så er det ikke noe	Knapp for utskrift av rapporten. Fint om man kan endre fonttyper, og formatere rapporten selv. Bør være mulig å legge inn en forside i rapporten. Kan legges inn sidetall. Ønsker mulighet til å laste ned dokumentet som docx.

			<p>rapport. Nysgjerriger, hvem er vi dette har vi forsket på og litt om prosjektet. Med en mulighet dersom man ville redigere det så kan man kanskje kunne ha mulighet til å laste ned docx. Mange lærere hadde satt pris på en lever nå knapp. Også er man ferdig. Kan være litt moro i det å kunne sende selve rapporten i plenum sammen med klassen. Alltid lurt at man kan gå tilbake også endre på ting. Forslag til at det blir lagt inn en gå hjem knapp.</p>	<p>Bør være en leveringsknapp, slik at man har mulighet til å gå tilbake og endre på ting.</p>
E4.1.1 1	<p>Hvis du skal legge til et bilde underveis i prosessen, hvordan vil du løse det? Prøv å legge til bilder i prosjektet.</p>	<p>Når brukeren velger å laste opp et bilde til et steg i metoden, så skal det legges inn ett bilde knyttet til stedet jeg velger å laste det opp.</p>	<p>Ikke implementert. – Ser ut som at man prøver å lure brukeren med sjekkboksen.</p>	<p>Bør gjøre «samtykke»-knappen penere.</p>
E4.1.1 2	<p>Hvordan vil du gå frem for å endre innhold i rapporten underveis?</p>	<p>Brukeren endrer innhold i rapporten.</p>	<p>Ikke implementert.</p>	

E4.1.1 3	Hvordan vil du gå frem for å slette innhold i rapporten underveis?	Brukeren sletter innhold i rapporten.	Ikke implementert.	
E4.1.1 4	Hva ble du møtt av når du prøvde å laste opp et bilde?	Brukeren blir bedt om å innhente samtykke.	Ikke implementert.	
E4.1.1 5	Hvordan vil du gå frem for å slette opplastet innhold (for eksempel bilder)?	Brukeren sletter opplastet innhold fra rapporten.	Ikke implementert.	

E4.2 PROSJEKTVEILEDEREN SKAL HA EN PROSJEKTOVERSIKT

Hovedtema	Oppgave	Ønsket resultat	Oppnådd resultat/diskusjon	Forbedringspotensial
E4.2.1	Hvordan vil du gå fram for å se en oversikt over dine prosjekter?	Brukeren finner fram til oversikten over sine prosjekter.	Kunne gjort om fargene på knappene her. Litt problematisk med google auth rett over slett bruker. Problematisk med slett bruker i det heletatt, fordi dersom lærer er pålogget kan barna slette brukeren til læreren.	Ønsker andre farger på knappene. Ønsker at informasjon om Google Authentication flyttes bort fra «Slett bruker» panelet. Ønsker at slett bruker funksjonalitet fjernes.
E4.2.2	Hvordan vil du gå fram for å åpne et av dine allerede	Brukeren åpner et av sine allerede		

	eksisterende prosjekter?	eksisterende prosjekter		
--	-----------------------------	----------------------------	--	--

AVSLUTTENDE SPØRSMÅL TIL BRUKERNE

Syns du navigeringen på siden var intuitiv? - Er det lett å forstå hva slags muligheter du har i applikasjonen?

Diskutert underveis. To knapper for å komme seg videre og sånt ikke bra.

Har du noen tips til forbedringer eller andre kommentarer?

Diskutert underveis.

6.7 AKSEPTANSETEST

Følgende brukerhistorier er levert i løpet av studentprosjektet.

Det er ikke alle brukerhistoriene som skal akseptansetestes da noen av de er tekniske. Disse vil være merket slik at det er synlig hvilke man skal akseptanseteste.

Det er ønskelig at eventuelle feil som avdekkes under akseptansetesting blir prioritert (A, B og C feil), slik at vi enkelt ser hvilke feil vi skal starte med først. Forslag til hvordan vi arbeider med feil som meldes inn:

- A-feil: Brukerhistorier fungerer ikke som forventet eller total svikt i systemet (sider som ikke er tilgjengelige, input som ikke blir sendt til backend og lagret). Disse skal fikses.
- B-feil: Features som ikke fungerer helt og det oppstår feil (feks valideringsfeil). Dette er feil som vi fikser hvis det er tid til det.
- C-feil: Endringsønsker (f.eks. ønsker rundt farger, tekst eller lignende). Dette er feil som det er naturlig å jobbe videre med ved produksjonsetting.

Grunnlaget for en slik måte å prioritere eventuelle feil på er at det er behov for tid til dokumentering. Dette gjelder både bacheloroppgaven, men også dokumentasjon som skal overleveres til Forskningsrådet.

Svar fra utviklere fargelegges basert på status, og får følgende melding:

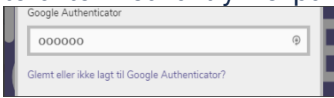
- **B-1: Klar til Publish**
- **B-1: Ikke klar til akseptanse**
- **B-1: Klar til Akseptanse**

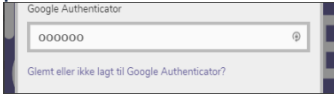

Lenke til Nysgjerrigper Prosjektveileder: <http://np-frontend.s3-website.eu-central-1.amazonaws.com/>




OBS! Testscriptene for brukerhistorier NP-20, NP-22, NP-24, NP-26, NP-29 og NP-31 er koblet sammen og akseptansetesting av disse skal utføres i sammenheng med hverandre.

Bruker-historie	Akseptanse-kriterie	Testscript	Teste t av og dato	Aksept -anse- testet ok/feil	Kommentar fra testing
EPIC: Innlogging og sikkerhet					
NP-9 - E1.1 - Som en besøkende ønsker jeg en landingsside slik at jeg kan få en kort introduksjon til Nysgjerrigpe rmetoden og hva jeg kan gjøre for å komme igang	<p>Gitt at jeg ikke er innlogget eller ikke har en bruker</p> <p>Når jeg velger å navigere til applikasj onens nettadrese</p> <p>Så skal appli kasjonen vise meg en landingss ide med informasj on og mulighet til å komme videre.</p>	<ol style="list-style-type: none"> Gå til http://np-frontend.s3-website.eu-central-1.amazonaws.com/ i en nettleser Bekreft at du får mulighet til å logge inn eller opprette ny bruker Scroll nedover for å se video og mer informasjon om Nysgjerrigpermetoden 	<p>24.4./ MM/ Edge/ PC</p> <p>30/4/ MM/ Chrom e/PC</p>	OK	<p>C-feil flyttet til forbedringspunkter av - KUNDEN</p> <p>UTVIKLER _____: Oppdatering 07.05.19: "Les mer"-lenken fungerer fortsatt ikke på iPad, hverken i Chrome eller på Safari (etter 3-5 klikk fungerer den)</p> <p>UTVIKLER _____: "Les mer" skal nå være klikkbar uansett skjermstørrelse - Klar til Akseptanse</p> <p>Retestet KAF 7.5 OK</p>
NP-11 - E1.2 - Som en besøkende ønsker jeg å kunne opprette en bruker for applikasjone n slik at jeg kan bruke den i undervisning en.	<p>Gitt at jeg ønsker å opprette bruker</p> <p>Når jeg velger opprett bruker funksjone n</p> <p>Så skal appli kasjonen opprette en bruker</p>	<p>Her bør man teste med gyldig og ugyldig data. Test først med data som du forventer at applikasjonen skal kunne opprette bruker av.</p> <ol style="list-style-type: none"> Prøv å opprette bruker med ugyldig informasjon <ol style="list-style-type: none"> Prøv å opprett bruker med navn som inneholder tall Prøv å opprett bruker hvor "Skole" inneholder tall 	<p>24.4./ MM/ Edge/ PC</p> <p>--</p> <p>Ny test:</p> <p>28.4/ MM/ Chrom e/PC</p> <p>--</p>	OK	<p>4 - B/C: Passord som huskes</p> <p>Passord som blir husket er nå fikset. Ikke publisert ennå.</p> <p>KUNDEN _____: Det ser ut som om dette er publisert? Testet OK.</p> <p>5 - A: Tall i fornavn</p> <p>KUNDEN _____/ KUNDEN: Kan lage bruker med tall i fornavn og etternavn.</p>

	<p>basert på innfylte data fra brukeren</p>	<p>c. Prøv å opprett bruker der passord er ulikt i feltene "Passord" og "Gjenta passord"</p> <p>2. Nytt script 29.04</p> <p>a. Etter å ha navigert deg til startside, klikk på "Registrering" knappen</p> <p>b. Fyll ut brukerskjemaet med gyldig input</p> <p>c. Klikk på "Lag bruker"</p> <p>d. Da vil du få informasjon om å laste ned google-autentisering.</p> <p>e. Scann QR-koden.</p> <p>f. Bruk bekreftelses koden og trykk på bekreft.</p> <p>g. Prøv å logg på med brukeren og ved bruk av Autentiserings kode</p> <p>3. Prøv deretter å avbryte oppretting av bruker (NB! Ikke klar til akseptanse)</p> <p>a. Etter å ha navigert deg til startside, klikk på "Registrering"-knappen</p> <p>b. Fyll ut brukerskjemaet med gyldig input (under testen er det viktig å huske denne brukerens epost-adresse og passord)</p> <p>c. Trykk på "Lag Bruker"</p> <p>d. Lukk fane/nettleser før du har skannet Google Authenticator QR-koden</p> <p>e. Gå tilbake til startside</p> <p>f. Trykk på "Logg Inn"-knappen</p> <p>g. Prøv å logg inn uten å bruke tofaktor... SKAL IKKE GÅ.</p> <p>h. Trykk på lenken for "mistet eller ikke lagt til"</p>  <p>i. Følg instruksjonene</p>	<p>KAF/iP ad</p>	<p>OK med tall i skolenavn fordi det kan eksistere, men usikker på om dette skal være lov.</p> <p>5-A: Klar til Akseptanse 02.05-TAD</p> <p>5-A Får fortsatt lov til å registrere tall i fornavn. 05.05. MM/Chrome. Men dette er ikke noe jeg ser at vi trenger å prioritere.</p> <p>5-A: Klar til Akseptanse 06.05-TAD</p> <p>Retestet av KAF/PC/Chrome: Ok</p> <p>2.g - B: Henger igjen på Google authenticator-side</p> <p>KUNDEN: Hvis man har vært innlogget tidligere, kommer man direkte til "Legg til Google Authenticator" hvis man velger "Lag bruker", og må refreshe for å komme til registrerings skjema. Det er mulig dette er en ikke-feil, men fint om dere vurderer dette.</p>  <p>Google Authenticator vellykket. Nå kan du logge inn!</p> <p>Dette var en feil ja.</p> <hr/> <p>Klar til Akseptanse 02.05.19-UTVIKLER</p> <p>OK - testet 5.5./MM/CHrome</p>
--	---	--	------------------	--

		<p>j. Prøv å logg på pånytt ved bruk av tofaktor.</p> <p>Da gjenoppretting av passord ikke er implementert, så er ikke det at man alltid kan lage ny autentisering regnet som en feil. Dette vil da bli bakt inn i en reset av passord løsning.</p>			
<p>NP-13 - E1.3 - Som en besøkende ønsker jeg å kunne logge på med min registrerte bruker Slik at jeg kan bruke applikasjonene</p>	<p>Gitt at jeg har en registrert bruker</p> <p>Når jeg velger logg inn funksjonen</p> <p>Så skal applikasjonen logges på med tofaktor og brukeren får sine definerte autorisasjoner.</p>	<p>Gå til startside: http://np-frontend.s3-website-eu-central-1.amazonaws.com/</p> <p>Pålogging lærer</p> <ol style="list-style-type: none"> 1. Trykk på "Logg inn"-knappen 2. Skriv inn e-postadresse, passord og kode for Google Authenticator. Hvis du ikke har opprettet egen bruker kan du bruke følgende innlogging: <ol style="list-style-type: none"> a. E-postadresse: <u>SENSURERT</u> b. Passord: <u>SENSURERT</u> c. Om du ikke har tofaktor autentiseringen for brukeren, så kan du lage ny tofaktor ved å trykke på. 3. Etter å ha logget inn skal knappene "Min side" og "Nytt prosjekt" vises under logoen til prosjektveilederen. 	<p>24.4./MM/Edge/PC</p> <p>20.4:</p> <p>KAF/ iPad / Safari</p> <p>MMO/ Chrome</p>	OK	<p>2. A: Casesensitiv innlogging</p> <p>KUNDEN på iPad: Ved logg inn: Brukernavn/e-post er casesensitiv (får ikke logget inn med Katta@flcharms.com når katta@flcharms.com er registrert e-post).</p> <p>B: Klar til Akseptanse. Opprett ny bruker og test på nytt.</p> <p>6.5: Testet KAF/iPad/Safari: OK</p> <p>KUNDEN: Jeg klarer ikke helt å gjenskape denne feilen i Chrome på PC, jeg får logget inn med både fetter@fetter.no og Fetter@fetter.no, men får problemer dersom jeg bruker en stor bokstav i e-post ved oppsett av Authenticator under "Glemt eller ikke lagt til Google Authenticator"</p>

		<p>Det har tidligere vært mulig å logge inn med vilkårlig Google Authenticator-kode. Dette skal nå være fikset etter nyeste oppdatering.</p> <p>Pålogging admin</p> <ol style="list-style-type: none"> 1. Trykk på "Logg inn"-knappen 2. Skriv inn e-postadresse, passord og kode for Google Authenticator Hvis du ikke har opprettet egen bruker kan du bruke følgende innlogging: <ol style="list-style-type: none"> a. E-postadresse: <u>SENSURERT</u> b. Passord: <u>SENSURERT</u> c. Om du ikke har tofaktor autentiseringen for brukeren, så kan du lage ny tofaktor ved å trykke på.  <ol style="list-style-type: none"> 3. Etter å ha logget inn skal knappene "Min side" og "Nytt prosjekt" vises under logoen til prosjektveilederen. 		 <p>B-3. Klar til Akseptanse</p> <p>5.5.: Begge testet OK i Chrome av MM, men <u>KUNDEN</u> kan kanskje teste denne på iPad?</p> <p>OBS: Ikke testet stor bokstav i e-post ved oppsett av Google Authenticator</p>
<p><u>NP-93</u> - E 1.5 - Som utvikler ønsker jeg en databasemodell for bruker slik at denne er klar når funksjonalitet som behandler brukere produseres</p>	<p>Gitt at jeg er en utvikler</p> <p>Når jeg skal lage funksjonallitet som behandler brukere</p> <p>Så skal data basemodellen for brukere allerede være</p>		<p>Skal ikke testes, er en teknisk brukerhistorie</p>	

	tilgjengelig				
EPIC: Innlogget bruker					
<p>NP-16 - E2.1 - Som en lærer ønsker jeg å kunne se min brukerinformasjon slik at jeg verifiserer at opplysningene er korrekte.</p>	<p>Gitt at jeg har en bruker,</p> <p>Når jeg velger min sidefunksjonen</p> <p>Så skal applikasjonen vise min brukerinformasjon</p>	<p>1. Logg inn som en bruker med lærerrettigheter: email: <u>SENSURERT</u> passord: <u>SENSURERT</u></p> <p>2. Trykk på brukerens navn på navigasjonslinjen øverst i nettleservinduet.</p>  <p>3. Trykk så på Brukerinnstillinger.</p>  <p>4. Da skal du komme til siden med brukerinnstillinger/informasjon.</p> 	<p>24.4./MM/Edge/PC</p> <p>26.4./MM/EDGE/PC</p> <p>Ipad/KAF</p>	OK	
<p>NP-23 - E2.3 - Som en lærer ønsker jeg å se hvordan jeg skal gå frem for å få slettet brukeren min slik at jeg kan fjerne den ved ønske</p>	<p>Gitt at jeg har en registrert bruker og er inne på min side</p> <p>Når jeg ser informasjon om min egen bruker</p>	<p>1. Logg inn som en bruker med lærerrettigheter: NB! Dersom du har tenkt til å prøve å slette en bruker, vennligst gjør dette på en bruker du har opprettet selv. Dersom du ikke har tenkt til å teste selve slettefunksjonaliteten, kan du benytte denne brukeren: email: <u>SENSURERT</u> passord: <u>SENSURERT</u></p>	<p>24.4./MM/Edge/PC</p> <p>30/4MM Chrome</p>	OK	<p>B - Slettet bruker er fortsatt innlogget</p> <p>KUNDEN: Hvis du sletter brukeren, er man fortsatt innlogget og må selv logge ut. Refresher man siden endrer data seg til None None. Usikker på om det er en B-feil, eller C-forbedringspunkt.</p>

Så skal jeg kunne se hvordan jeg går frem for å få slettet brukeren min

2. Trykk på brukerens navn på navigasjonslinjen øverst i netteservinduet.



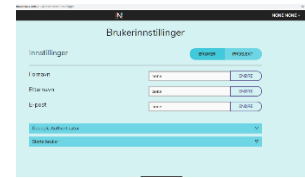
3. Trykk så på Brukerinstillinger.



4. Trykk så på Slette bruker.



5. Her vil du ha mulighet til å slette din egen bruker. Du vil bli advart når du trykker på slette knappen.



~~B-1: Klar til Akseptanse 02.05.19 UTVIKLER~~

~~B-1: Testet 5.5. MM/Chrome. Bruker logges ut når man sletter seg selv. OK.~~

Epic: Administratorgrensesnitt - Oversikt og statistikk

NP-25 - E3.1.1 - Som en administrator ønsker jeg et administrasjonsgrensesnitt slik at jeg kan gjøre administrative oppgaver




Gitt at jeg har administratorrettigheter
Når jeg logger inn
Så skal applikasjonen

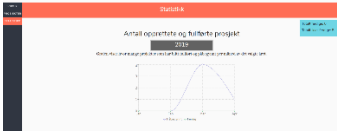
1. Logg in som en bruker med adminrettigheter: email: SENSURERT password: SENSURERT
2. i stedet for prosjektveileder. Skriv i nettadressefeltet. <http://np-frontend.s3-website.eu-central-1.amazonaws.com/admin>
3. Da skal du komme til

24.4./MM/Edge/PC

OK



Status oppdatert i Jira. /KGY



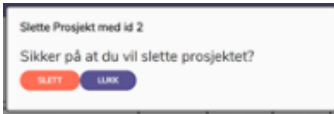
	vise meg ett administrasjonsgrensesnitt	administratorgrensesnitt. 			
NP-27 - E3.1.2 - Som en administrator ønsker jeg å kunne se alle opprettede prosjekter på en bestemt bruker slik at jeg har mulighet til å se prosjektene som hører til en bestemt person.	Gitt at jeg har administratorrettigheter Når jeg velger brukeren fra administratorgrensesnittet Så skal applikasjonen vise alle prosjektene som er knyttet til denne brukeren.	1. Logg in som en bruker med adminrettigheter: email: <u>SENSURERT</u> passord: <u>SENSURERT</u> 2. Gå inn på administratorgrensesnittet. Velg lærerfanen og finn en lærer med prosjekt. Admin/Petter Prosa har flere. 3. Trykk på raden til brukeren, da kommer et nedtrekksvindu, med en knapp; Prosjekter . Trykk på denne.  4. Da blir du flyttet til prosjektlistevinduet, hvor det allerede har blitt søkt på alle prosjekter til denne brukeren.  5. Prosjekttabellens søk kan nullstilles. Prøv å søk på Prosa, for å finne Prosa sine prosjekt.	24.4./MM/Edge/PC	OK	Status oppdatert i Jira. /KGY
NP-28 - E3.1.3 - Som en administrator ønsker jeg å kunne se tall over påbegynte, men ikke ferdigstilte prosjekter slik at jeg har	Gitt at jeg har administratorrettigheter Når jeg velger administratorforsiden	1. Logg in som administratorbruker. NP-25. Og gå inn på grensesnittet. 2. Gå inn på statistikk lenken, her kan man se statistikken for ferdige og ikke ferdige prosjekter. 3. For flere detaljer så kan man se på prosjektabellen, som har påbegynt og ferdig	24.4./MM/Edge/PC	OK	C: Gitt at jeg velger "prosjekter" på forsiden (kommer direkte til brukerliste) Status oppdatert i Jira. /KGY

<p>mulighet til å ha en formening om hvor mange prosjekter som blir påbegynt men ikke ferdigstilt</p>	<p>Så skal applikasjonen vise prosjekter som er påbegynt men ikke ferdigstilt</p>	<p>kollonner.</p> 			
<p>NP-30 - E3.1.4 - Som en administrator ønsker jeg å kunne se tall over ferdigstilte prosjekter slik at jeg har mulighet til å ha en formening om hvor mange prosjekter som blir fullført</p>	<p>Gitt at jeg har administratorrettigheter</p> <p>Når jeg velger administratorforsiden</p> <p>Så skal applikasjonen vise prosjekter som er ferdigstilt</p>	<p>1. Logg in som administratorbruker. NP-25. Og gå inn på grensesnittet.</p> <p>2. Gå inn på statistikk lenken, her kan man se statistikken for ferdige og ikke ferdige prosjekter.</p> <p>3. For flere detaljer så kan man se på prosjekt Tabellen, som har påbegynt og ferdig kollonner.</p> <p>4. Fyll ut konklusjon, om prosjekt og om deltagere i selvvalgte prosjekter.</p>	<p>24.4./MM/Edge/PC</p> <p>28.4./mm/Chrome/PC</p>	<p>OK</p>	

Epic: Administratorgrensesnitt - Bruker- og prosjekthåndtering

<p>NP-33 - E3.2.1 - Som en administrator ønsker jeg å kunne se alle brukere av applikasjonen slik at jeg får en oversikt over brukere av applikasjonen</p>	<p>Gitt at jeg er administrator av applikasjonen</p> <p>Når jeg velger vis brukere</p> <p>Så skal applikasjonen vise en liste med alle brukerne av</p>	<p>1. Logg inn på administratorgrensesnittet. Se NP-25</p> <p>2. Landingsvinduet fører til brukervisningstabellene. Hvor man kan se alle brukere. Fanen til venstre kan også brukes til å navigere hit, fra admin grensesnittet.</p>  <p>5. Prøv å søk etter en bruker ved bruk av</p>	<p>24.4./MM/Edge/PC</p>	<p>OK</p>	<p>Status oppdatert i Jira. /KGY</p>
--	---	---	-------------------------	-----------	--------------------------------------

	applikasjonen	søkefeltet. 6. Prøv å sortere kolonnene ved å trykke på kolumnefanen. 7. Prøv å bytt til lærervindu for å se lærerinformasjon.			
NP-34 - E3.2.2 - Som en administrator ønsker jeg å kunne slette en lærerbruker slik at jeg kan håndtere feil med brukere, eller ønske om å fjerne bruker	<p>Gitt at jeg er inne i administratorgrensesnittet og ser alle brukere av applikasjonen</p> <p>Når jeg velger slett bruker metoden for en valgt bruker</p> <p>Så skal applikasjonen slette denne brukeren og informasjon om denne fra applikasjonen</p>	<p>1. Lag en ny bruker -se NP-11 som skal slettes. Eks: Navn: Slettmeg Etternavn: Slettesen, Skole: Sletterud, epost: slett@meg.no, passord: Sl3ttM3g!</p> <p>2. Logg in som en bruker med adminrettigheter: email: SENSURERT passord: SENSURERT</p> <p>3. i stedet for prosjektveileder. Skriv i nettadressefeltet. http://np-frontend.s3-website.eu-central-1.amazonaws.com/admin</p> <p>4. I administratorgrensesnittet.</p> <p>5. Finn brukeren Slettmeg Slettesen.</p> <p>6. Trykk på cellen og åpne nedtrekksvinduet.</p>  <p>7. Trykk på slett knappen.</p>  <p>8. Verifiser at Slettmeg Slettesen ikke finnes i listen lengre.</p>	24.4./MM/Edge/PC	OK	Status oppdatert i Jira. /KGY
NP-36 - E3.2.3 - Som en administrator	<p>Gitt at jeg er i liste over alle påbegynt</p>	<p>1. Logg in som en bruker med adminrettigheter: email: SENSURERT passord:</p>	24.4./MM/Edge/PC	OK	Status oppdatert i Jira. /KGY

<p>r ønsker jeg liste over prosjekter med mulighet for sletting slik at jeg kan slette ett prosjekt for en bruker.</p>	<p>e prosjekter (3.1.1)</p> <p>Når jeg velger slett prosjekt funksjonen</p> <p>Så skal applikasjonen slette dette prosjektets og dets koblinger fra applikasjonen</p>	<p>SENSURERT</p> <p>2. Opprett nytt prosjekt, se NP-12. Sett gruppens navn til slettdenne.</p> <p>3. Gå inn på administratorgrensesnittet.</p> <p>4. Velg prosjektfanen fra navigeringskolonnen til venstre.</p>  <p>5. Finn prosjektet slettdenne i tabellen.</p> <p>6. Trykk på raden til prosjektet for å åpne nedtrekksvinduet.</p>  <p>7. Trykk på slett, deretter på nytt i verifikasjonsvinduet.</p>  <p>8. Verifiser at prosjektet ikke finnes i listen lengre.</p>			
--	---	---	--	--	--

EPIC: Prosjektveileder - Bruker- og prosjekthåndtering

<p>NP-12 - E4.1.1 - Som en lærer ønsker jeg å kunne opprette ett nytt prosjekt slik at jeg får planlagt og gjennomført Nysgjerrigprøvet metoden på en god måte</p>	<p>Gitt at jeg er lærer</p> <p>Når jeg velger opprett nytt prosjekt</p> <p>Så skal applikasjonen starte en ny prosjektveileder og opprette et prosjekt</p>	<p>Her bør man teste med gyldig og ugyldig data. Test først med ugyldig data som vi ønsker å få feilmelding på: Trykk på "Nytt Prosjekt"-knappen</p> <p>1. Fyll ut de følgende feltene med følgende informasjon</p> <p>a. <i>Gruppens navn</i>: <Ikke fyll inn noe informasjon i dette feltet></p> <p>b. <i>Antall forskere</i>: vi er mange</p> <p>c. <i>Ansvarlig veileder</i>: <Ikke fyll inn noe informasjon i dette feltet></p>	<p>24.4./MM/Edge/PC</p> <p>KAF/ipad/Safari</p>	<p>OK</p>	<p>Status oppdatert i Jira. /KGY (Ny feil lagt inn 30/4)</p> <p>B: Autoutfylling ved oppretting av prosjekt</p> <p><u>KUNDEN</u> på iPad:</p> <p>B: Ansvarlig veileder bør autoutfylles (innlogget bruker)</p> <p>B: Skoletrinn og skole er autoinnfylt, selv om jeg ikke har opprettet et prosjekt før (eller?)</p>
--	---	--	--	-----------	---

		<p>2. Trykk på "Lag Prosjekt"-knappen</p> <p>Forventet resultat av stegene over er at de tre første feltene blir røde og at brukeren forblir på "Kom igang!"-siden.</p> <p>Prøv deretter å lage en prosjekt med gyldig data;</p> <ol style="list-style-type: none"> Fyll ut feltene med følgende informasjon <ol style="list-style-type: none"> Gruppens navn: Superforskerne Antall forskere: 24 Ansvarlig veileder: <u>SENSURERT</u> Skoletrinn: 3. barneskole Skole: Forskningsrådet barneskole Trykk på "Lag Prosjekt"-knappen Verifiser at de riktige opplysningene ble lagret ved å se gjennom "kvitteringen" man får etter å ha lagret prosjektet. 			<p>-----</p> <p>-----</p> <p>Klar til Akseptanse 02.05.19 <u>UTVIKLER</u></p> <p>OK, testet av MM, 5.5./Chrome</p>
<p><u>NP-15</u> - E4.1.2 - Som en lærer ønsker jeg mulighet for veiledning i hvordan hvert steg i Nysgjerrigpe rmetoden kan gjennomføres slik at jeg får planlagt og gjennomført Nysgjerrigpe rmetoden på en god måte</p>	<p>Gitt at jeg er inne i ett av mine prosjekt i veiledere n</p> <p>Når jeg er ved et steg i metoden og velger å vise veiledning for steget</p> <p>Så skal applikasjonen vise tips</p>	<ol style="list-style-type: none"> Velg et vilkårlig steg (helst en som du ikke har fullført ennå) Les veiledningsteksten som ligger i boksen med farget ramme. <div data-bbox="475 1496 807 1541" style="border: 1px solid orange; padding: 2px; margin: 5px 0;"> <p><small>Tryk et spørsmål om noe du ikke forstår. Det er her du kan stille spørsmål om noe som ikke fungerer som det skal.</small></p> <p><small>Da har du en problemstilling!</small></p> </div> <ol style="list-style-type: none"> Trykk på "Tips til hvordan man kan finne en god problemstilling" under veiledningsteksten. 	25.4./MM/Edge/PC	OK	Status oppdatert i Jira. /KGY


	og veiledning for dette steget				
NP-18 - E4.1.3 - Som en lærer ønsker jeg en introduksjon til Nysgjerrigpermetoden slik at jeg får kjennskap til prosessen	<p>Gitt at jeg er lærer</p> <p>Når jeg oppretter ett nytt prosjekt</p> <p>Så skal prosjektveilederen vise en introduksjon til Nysgjerrigpermetoden «kom i gang»</p>	<p>1. Velg steg "Kom igang." 2. Les veiledningsteksten som ligger i boksen under tittelen på siden.</p> <p><small>Etter å ha trykket på "Kom igang" vil du bli informert om hvordan du kan bruke Nysgjerrigpermetoden i ditt prosjekt. Dette er et nytt steg i prosessen, og du kan finne mer informasjon om dette i brukerveiledningen.</small></p>	25.4./MM/Edge/PC	OK	Status oppdatert i Jira. /KGY
NP-20 - E4.1.4 - Som en lærer ønsker jeg å få informasjon/tips om hva som kan være gode problemstillinger og mulighet til å fylle inn tekst slik at jeg kan legge til en problemstilling og en beskrivelse av denne	<p>Gitt at jeg er i prosjektveilederen</p> <p>Når jeg velger steg 1, «Dette lurer vi på» i prosjektveilederen</p> <p>Så skal jeg kunne få veiledning til steget og mulighet til å legge inn problemstilling</p>	<p>OBS! Testscriptene for brukerhistorier NP-20, NP-22, NP-24, NP-26, NP-29 og NP-31 er koblet sammen og akseptansetesting av disse skal utføres i sammenheng med hverandre.</p> <p>1. Velg steg 1, "Dette lurer vi på." 2. Les veiledningsteksten som ligger i boksen med farget ramme. 3. Skriv følgende under "Problemstillingen vår." a. "Hvorfor lukter norskbøkene merkelig?" 4. Skriv følgende under "Hvordan vi kom frem til den." a. "Vi synes at norskbøkene våre lukter merkelig. Noen mener at dette er fordi bøkene er gamle, mens andre tror at</p>	25.4./MM/Edge/PC ----- 27.4./MM/Chrome / PC --- 28.4.19	OK	Status oppdatert i Jira. /KGY

		lukten kommer fra et kjemikalie brukt av produsenten. Vi ønsker å finne ut mer 5. Velg "Neste steg."			
NP-22 - E4.1.5 - Som en lærer ønsker jeg å få informasjon/tips om hva som kan være gode hypoteser, og mulighet til å fylle inn tekst slik at jeg kan legge til hypoteser	Gitt at jeg er i prosjektv eilederen Når jeg velger steg 2, «Hvorfor er det slik?» Så skal jeg kunne få veileding til steget og mulighet til å føre inn en eller flere hypoteser	1. Velg steg 2, "Hvorfor er det slik." 2. Les veiledningsteksten som ligger i boksen med farget ramme. 3. Skriv følgende hypotese under "Mulig forklaring" a. "Bøkene lukter støv fordi de er gamle" 4. Trykk på "Legg til"-knappen. 5. Gjenta steg 3-4 for å opprette en ny hypotese med følgende hypotese: 6. Velg "Neste steg."	25.4./MM/Edge/PC 30/4MM/Chrome	OK	
NP-24 - E4.1.6 - Som en lærer ønsker jeg å få informasjon/tips om hvordan de ulike metodene kan benyttes, og ha mulighet til å velge metoder slik at jeg får lagt en plan for gjennomføringen av undersøkels er	Gitt at jeg er i prosjektv eilederen Når jeg velger steg 3, «Legg en plan» Så skal jeg kunne få veileding til steget og mulighet til å velge en eller flere metoder for	1. Velg steg 3, «Legg en plan.» 2. Les veiledningsteksten som ligger i boksen med farget ramme. For hypotese 1 1. Trykk på "Velg først metoder for undersøkelse" for hypotese 1. 2. Velg Intervju som forskningsmetode på hypotese 1. 3. Klikk på "Legg til." 4. Trykk på "Legg så en plan for hver av undersøkelsene" for hypotese 1.	26.4./MMO/Chrome/PC	OK	

	testing av hypotesene valgt i steg 2	<p>5. Trykk på Intervju og skriv inn følgende under "Skriv inn planen her"</p> <p>a. "Vi har planlagt å snakke med rektor for å spørre om hvor gamle bøkene egentlig er."</p> <p>6. Trykk på "Legg til plan"</p> <p>For hypotese 2</p> <p>1. Trykk på "Velg først metoder for undersøkelse" for hypotese 2.</p> <p>2. Velg "Forsøk og eksperiment" som forskningsmetode på hypotese 2.</p> <p>3. Klikk på "Legg til"</p> <p>4. Trykk på "Legg så en plan for hver av undersøkelsene" for hypotese 2.</p> <p>5. Trykk på "Forsøk og eksperiment" og skriv inn følgende under "Skriv inn planen her"</p> <p>a. "Vi kommer til å bruke en <i>AdeTech Cotton Swab</i> kjemi-kit for å se på den kjemiske komposisjonen til fire norskebøker og analysere dette."</p> <p>6. Trykk på "Legg til plan"</p> <p>1. Velg "Neste steg."</p>			
<p>NP-26 - E4.1.7 - Som en lærer ønsker jeg å ha mulighet til å fylle inn tekst slik at jeg kan beskrive gjennomføringen av undersøkelsene som skal</p>	<p>Gitt at jeg er i prosjektveilederen</p> <p>Når jeg velger steg 4, «Hent opplysninger»</p> <p>Så skal jeg</p>	<p>1. Velg steg 4, «Hent opplysninger.»</p> <p>2. Trykk på "Legg inn observasjonene til denne hypotesens undersøkelse(r) her."</p> <p>3. Trykk på metoden du valgte å ta med i forrige trinn</p> <p>4. Fyll inn data i tekstfeltet "Observasjoner gjort for undersøkelsesmetoden"</p> <p>5. Trykk på "Legg til observasjoner"</p>	26.4./MMO/Chrome/PC	OK	

teste hypotesene	kunne få veiledning til steget og mulighet til å beskrive gjennomføringen av planen lagt i punkt 3	6. Velg "Neste steg."			
NP-29 - E4.1.8 - Som en lærer ønsker jeg å ha mulighet til å fylle inn tekst og velge resultat slik at jeg kan konkludere på hypotesene og skrive en endelig konklusjon på problemstillingen for prosjektet.	Gitt at jeg er i prosjektveilederen Når jeg velger steg 5, «Dette har vi funnet ut» Så skal jeg kunne få veiledning til steget og mulighet til å konkludere per hypotese, samt konkludere på problemstillingen på bakgrunn av arbeidet gjort med hypotesene	1. Velg steg 5, «Dette har vi funnet ut» 2. Velg om du synes hypotesene du har laget er Sannsynlig, Vet ikke, eller Usannsynlig 3. Ut ifra det du har funnet ut kan du legge til en konklusjon til prosjektet ditt, gjør det. 4. Velg "Neste steg"	26.4./MMO/Chrome/PC Kate/iPad / Safari	OK	B: Mange klikk for å se observasjonene <u>KUNDEN:</u> De må kunne vises sammen med undersøkelsesmetode, plan og observasjoner. Klar til Akseptanse 06.05.19
NP-31 - E4.1.9 - Som en lærer ønsker jeg	Gitt at jeg er i prosjektveilederen	1. Velg steg 6, «Fortell til andre» 2. Fyll inn data i feltet Fortell om gruppen og/eller fyll inn data i	26.4./MMO/Chrome/PC	OK	

<p>tekstfelter for introduksjon av klassen og beskrivelse av prosessen slik at dette kan legges til i rapporten.</p>	<p>Når jeg velger steg 6, «Fortell til andre»</p> <p>Så skal jeg kunne få veiledning til steget og mulighet til å skrive en introduksjon og en beskrivelse av vår gjennomføring av prosjektet</p>	<p>feltet Fortell om prosjektet</p> <p>3. Velg "Lagre fortelling"</p>	<p>---</p> <p>28.4./MM/Chrome/PC</p> <p>30/4./MM/Chrome/PC</p>		
<p>NP-32 - E4.1.10 - Som en lærer ønsker jeg å se et utkast av rapporten (prosjektoppsummeringen) slik at jeg tilpasse den om nødvendig</p>	<p>Gitt at jeg som lærer har fullført stegene</p> <p>Når jeg velger å se utkast funksjonene</p> <p>Så skal prosjektveilederen vise en presentasjon av prosjektet</p>	<p>1. Logg inn som en bruker med lærerrettigheter, epost: <u>SENSURERT</u></p> <p>password: <u>SENSURERT</u></p> <p>2. Opprett ett nytt prosjekt, eller bruk et eksisterende</p> <p>3. Naviger deg til Se rapport i menyen til venstre i Prosjektveilederen</p>	<p>26.4./MMO/Chrome/PC</p>	<p>OK</p>	
<p>NP-35 - E4.1.11 - Som en lærer ønsker jeg å kunne legge inn bilder underveis i</p>	<p>Gitt at jeg som lærer har et bilde som jeg vil laste opp</p>	<p>1. Logg inn som en bruker med lærerrettigheter, epost: <u>SENSURERT</u></p> <p>password: <u>SENSURERT</u></p> <p>2. Opprett ett nytt prosjekt, eller bruk et eksisterende</p>	<p>26.4./MMO/Chrome/PC</p>	<p>OK</p>	<p>A: Kan ikke legge til bilder over 0.8 MB med iPad/Safari, men får feilmelding om at bildet er for stort (å legge inn store bilder på</p>



<p>prosessen slik at jeg kan bruke beskrivende bilder av elevenes arbeid i rapporten.</p>	<p>Når jeg velger å laste opp bilde til et steg i metoden</p> <p>Så skal det legges inn ett bilde knyttet til stedet jeg velger å laste det opp</p>	<p>3. Legg til et eller flere bilde(r) (steg 1, 3, 5 og 6 tillater dette).</p> 	<p>7.5/KA F/Safari/iPad</p> <p>7.5/KA F/Chrome/iPad</p>	<p>iPad/Chrome fungerer fint)</p> <p>KUNDEN: på iPad: Kan ikke legge til bilde i trinn 1 når jeg bruker "Ta bilde eller video" eller "Velg fra bibliotek".</p> <p>B: Ingen lagre-knapp steg 4</p> <p>KUNDEN: Steg 4 Hente opplysninger. Når man laster opp bilder her, får man ikke opp noen knapp som sier Lagre endringer. Bildet lagrer seg uansett, men kan virke usikkert for bruker</p> <p>B: Manglende info og feilmelding om for store bilder</p> <p>KUNDEN: Ingen feilmelding eller respons hvis man laster opp et bilde som (jeg antar) er for stort. Prøvde å laste opp et på 4 MB og det skjer ingenting. Enten gi begrunnet feilmelding eller ha en kort infotekst om maks filstr., f. eks. bak tekst om personvern: (Maks filstr. er X)</p> <p>UTVIKLER: Bilder kan nå være opptil 10 MB store. Feilmelding for større bilder har også blitt implementert.</p> <p>-----</p> <p>-----</p> <p>---</p> <p>A-1: Klar til Akseptanse 02.05.19 UTVIKLER</p> <p>KUNDEN: Kan du sjekke dette på ipaden?</p>
---	---	--	---	--

					<p>(Chrome: Fortsatt OK 5.5./MM)</p> <p><u>UTVIKLER</u> : Skjermdumper lar seg lagre, ikke bilder (bilder sendt til <u>UTVIKLER</u> -</p> <p>I <u>TILLEGG</u> som diskutert med <u>UTVIKLER</u> : B-feil: Rekkefølgen på samtykke og last opp bilde er ikke intuitivt - du får laste opp bilde dersom du ikke har merket av for samtykke, får feilmelding om at samtykke mangler, merker av, men da har bildet forsvunnet og du må laste det opp på nytt, uten at du får beskjed om at det å lastes opp på nytt. Gjøre rekkefølgen "obligatorisk".</p> <p>B-1: Klar til Akseptanse 02.05.19 <u>UTVIKLER</u></p> <p>B-1: Usikker på hva som har blitt endret her, men siden bildet legger seg der, gjør det ikke så mye at Lagre-knappen er inaktiv hvis man kun har gjort endring på bildet, ikke teksten. 5.5./MM/Chrome</p> <p>Klar til Akseptanse 02.05.19 <u>UTVIKLER</u></p> <p>Apple iPad Pro 10.5" 256GB WiFi Sølv - iOS 12.1.4</p> <p>Maksimal filstørrelse på AWS er tilsynelatende 0,8MB. Lokalt klarer vi å laste opp større filer, vi kan evt. se på muligheten til å utvide denne</p>
--	--	--	--	--	--

					<p>grensen dersom dette er ønskelig.</p> <p>B-1: Får fortsatt ingen feilmelding eller beskjed om at bildefilen er for stor til å lastes opp. 5.5./MM/Chrome</p> <p>5-A: Klar til Akseptanse 06.05-TAD</p> <p>----</p> <p>KAF: Fikk ikke lastet opp store bilder på iPad/Safari, men Chrome.</p> <p>OK i Chrome/PC: Feilmelding ved for stort bilde, fil opp til 10 MB kan lastes opp og får opp melding "Laster Opp Bilde" når bildet lastes opp. 08.05. <u>KUNDEN</u></p>
<p><u>NP-37</u> - E4.1.12 - Som en lærer ønsker jeg å kunne endre data i utfylte felter slik at jeg har fleksibilitet i å endre data når jeg gjennomfører prosjektet</p>	<p>Gitt at jeg er i prosjektveilederen</p> <p>Når jeg velger et datafelt som er utfyllt</p> <p>Så skal jeg ha mulighet til å redigere teksten som står i feltet og rapporten oppdateres deretter</p>	<p>1. Logg inn som en bruker med lærerrettigheter, epost: <u>SENSURERT</u> passord: <u>SENSURERT</u></p> <p>2. Opprett ett nytt prosjekt eller benytt et eksisterende prosjekt</p> <p>3. Fyll inn data i prosjektveilederens felter.</p> <p>4. Prøv så å endre data i prosjektet.</p>	26.4./MM/Chrome/PC	OK	NB! Trinn 4

<p>NP-38 - E4.1.13 - Som en lærer ønsker jeg å ha mulighet til å slette utfylt data slik at jeg kan fjerne data som ikke passer inn i prosjektet</p>	<p>Gitt at jeg er i prosjektveilederen</p> <p>Når jeg sletter data fra utfylte felter</p> <p>Så skal rapporten oppdateres deretter</p>	<p>1. Logg inn som en bruker med lærerrettigheter, epost: <u>SENSURERT</u> passord: <u>SENSURERT</u></p> <p>2. Opprett ett nytt prosjekt eller benytt et eksisterende prosjekt</p> <p>3. Fyll inn data i prosjektveilederens felter.</p> <p>4. Prøv så å slette data som ikke ønskes i prosjektet.</p>			
<p>NP-39 - E4.1.14 - Som en lærer ønsker jeg å bli bedt om at samtykke innhentes før opplasting av bilde slik at jeg kan bekrefte at samtykke er innhentet.</p>	<p>Gitt at jeg er innlogget</p> <p>Når jeg er i Prosjektveilederen</p> <p>Så skal jeg ha mulighet til å legge til bilder mot at samtykke er innhentet</p>	<p>1. Logg inn som en bruker med lærerrettigheter, epost: <u>SENSURERT</u> passord: <u>SENSURERT</u></p> <p>2. Opprett ett nytt prosjekt eller benytt et eksisterende prosjekt</p> <p>3. Gå til et av disse stegene 1, 3, 5 og 6. Disse stegene vil tillate bildeopplasting. I bildekomponenten gis en beskjed om at samtykke må innhentes før bildeopplasting tillates</p> 	26.4./MM/Chrome/PC	OK	
<p>NP-41 - E4.1.15 - Som en lærer ønsker jeg å ha mulighet til å slette opplastet innhold slik at jeg kan fjerne innhold fra et påbegynt prosjekt.</p>	<p>Gitt at jeg er i prosjektveilederen</p> <p>Når jeg underveis i et prosjekt</p> <p>Så skal jeg ha mulighet til å slette</p>	<p>1. Logg inn som en bruker med lærerrettigheter, epost: <u>SENSURERT</u> passord: <u>SENSURERT</u></p> <p>2. Opprett ett nytt prosjekt, eller benytt et eksisterende prosjekt</p> <p>3. Legg til et eller flere bilde(r) (steg 1, 3, 5 og 6 tillater dette). Foreløpig pr 24/04 må man huke av for at samtykke er innhentet før man velger et bilde å laste opp.</p>	26.4./MM/Chrome/PC	OK	

	<p>innhold fra et påbegynt prosjekt og rapporten oppdateres deretter</p>	<p>4. Du skal nå ha mulighet til å slette innholdet etter at bilde dukker opp i bildekomponenten.</p>			
					
<p>NP-95 - E 4.1.17 - Som utvikler ønsker jeg et funksjonelt skjelett av applikasjonen slik at videre utvikling blir godt strukturert.</p>	<p>Gitt at jeg er en utvikler</p> <p>Når jeg skal lage ønsket funksjonalitet for applikasjonen</p> <p>Så skal stedet i applikasjonen der det skal inn allerede være opprettet.</p>			<p>Skal ikke testes, er en teknisk brukerhistorie</p>	
<p>NP-96 - E 4.1.18 - Som utvikler ønsker jeg en databasemodell for Prosjektet, slik at det er klart for videre utvikling av funksjonalitet.</p>	<p>Gitt at jeg er en utvikler</p> <p>Når jeg skal lage funksjonalitet som omhandler prosjekter</p> <p>Så skal data</p>			<p>Skal ikke testes, er en teknisk brukerhistorie</p>	

	basemodellen allerede være tilgjengelig				
EPIC: Prosjektveileder - prosjektoversikt					
NP-165 - E 4.2.1 - Som lærer ønsker jeg å kunne se en oversikt over egne prosjekter Slik at jeg har mulighet til å se egne prosjekter og det dermed er enklere å huske påbegynte prosjekter	Gitt at jeg har bruker Når jeg velger egne prosjekter Så skal applikasjonen vise alle prosjekter som er knyttet til min bruker	<ol style="list-style-type: none"> 1. Logg in som en lærer som har prosjekter(både fullstendige og ufullstendige). 2. Velg min side. 3. Trykk på prosjektknappen. 4. Verifiser at listen over prosjekter stemmer med brukerens prosjekter. 	28.4./MM / Chrome/PC	OK	
NP-166 - E 4.2.2 - Som lærer Ønsker jeg å kunne åpne eget eksisterende prosjekt slik at jeg kan ha flere prosjekter i gang samtidig.	Gitt at jeg har bruker Når jeg velger egne prosjekter Så skal applikasjonen åpne det valgte prosjektet	<ol style="list-style-type: none"> 1. Logg in som en lærer som har prosjekter(både fullstendige og ufullstendige). 2. Velg min side. 3. Trykk på prosjektknappen. 4. Trykk på prosjektet. 5. Trykk på velg prosjekt i nedtrekksmenyen. 6. Verifiser at du kommer til riktig prosjekt. 	28.4./MM / Chrome/PC	OK	



EPIC: Integrasjon

Som administrator

Ønsker jeg at prosjektveilederen implementerer Contentful

Slik at jeg har mulighet til å endre innhold uten å måtte endre tekst direkte i kildekode.

OBS! Denne ble gjort som en subtask i en annen brukerhistorie, og lenker derfor ikke til Jira.

Gitt at jeg har tilgang til Contentful

Når jeg går inn på Nysgjerrigspacet i Contentful

Så skal jeg ha mulighet til å endre ledeteksten i Prosjektveilederen

1. Logg inn i applikasjonen med vanlig bruker (ikke admin), og vel "Nytt prosjekt". Lenke til applikasjonen: <http://np-frontend.s3-website.eu-central-1.amazonaws.com/>
2. Legg merke til overskriften "Kom igang!", samt ledeteksten under
3. Gå til <https://www.contentful.com/>
4. Logg inn med bruker **SENSURERT** Hvis du er usikker på passordet, hør med **KUNDEN** eventuelt utviklerne
5. Velg "Content type" i menyen til venstre og klikk på ledeteksten med navn "steg0"
6. Endre overskriften eller selve ledeteksten i Contentful
7. Bekreft at teksten endret seg som forventet

28.4./MM / Chrome/PC

30/4/MM/ Chrome / PC

OK

EPIC: Tekniske krav

NP-19 - E6.1 - Som en prosjekteier ønsker jeg at applikasjonen bruker moderne teknologi tilsvarende

Gitt at jeg er prosjekteier

Når jeg vil ha utført vedlikehold

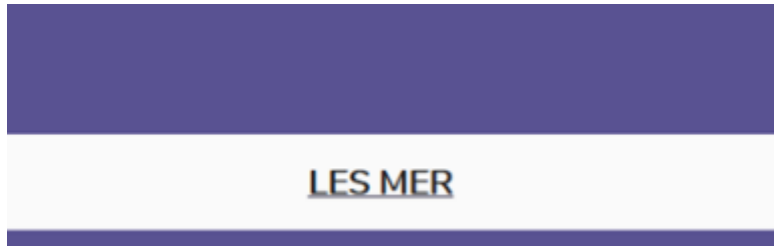
Skal ikke testes, er en teknisk brukerhistorie. Teknologivalg er

<p>lignende løsninger hos Forskningsrådet slik at eksisterende kompetanse kan benyttes og vedlikehold og drift går raskt og er enkelt</p>	<p>Id og drift av applikasjonen</p> <p>Så skal det gjøres med moderne og kjente verktøy</p>			<p>avklart med arkitekturforumet.</p>	
<p>NP-80 - E 6.2 - Som prosjekteier ønsker jeg at applikasjonen kan ligge i skyen, slik at alle kan nå den fra en nettleser</p>	<p>Gitt at jeg er prosjekteier</p> <p>Når jeg vil få tilgang til applikasjonen</p> <p>Så skal det være mulig å ha tilgang til hele applikasjonen gjennom en nettleser</p>	<p>1. Gå til http://np-frontend.s3-website-eu-central-1.amazonaws.com/</p> <p>2. Se at siden laster som forventet</p>	<p>MM</p>	<p>OK</p>	

C-feil til forbedringspunkter

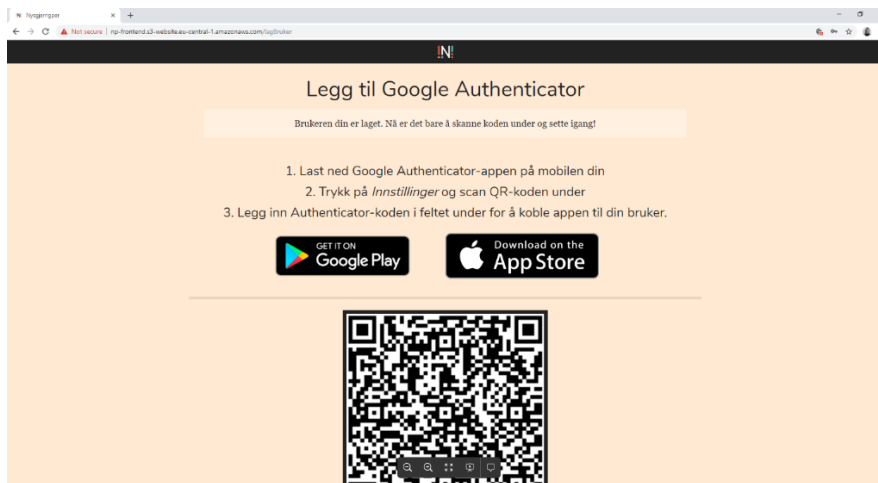
<p>NP-9 - E1.1 - Som en besøkende ønsker jeg en landingsside slik at jeg kan få en kort introduksjon til Nysgjerrigperm etoden og hva jeg kan gjøre</p>	<p>1-C: knapper ser annerledes ut i Edge. TAD: 1-C og 2-C er fikset.</p>  <p>3-C: Hvitt felt oppå det lilla når du holder musepeker over knapp "Les mer"</p>	<p>5-C: OK, testet av MM 5.5.</p>
---	--	-----------------------------------

for å komme igang



4-C: Hvis du klikker på lenkene "Les mer om prosjektet her" under prosjekt eksempene på forsiden, og deretter klikker backspace for å komme tilbake til prosjektveilederen, hadde det vært fint om du kom tilbake der du var, og ikke måtte scrolle ned igjen.

5 - C: Må QR-koden være så stor? For da må man scrolle ned for å se hele skjermbildet.



Klar til Akseptanse

6- C: Hjelpetekst punkt 2: Hos meg står det Konfigurer (ikke Innstillinger). Hvis man allerede har brukt appen virker det som man skal trykke på +tegnet og skann strekkoden. Jeg er usikker på om man skal skanne strekkoden hver gang, eller om det bare er første gang.

7- C: Hjelpetekst første linje: Bør det stå noe her a la: "For å logge inn på en sikker måte må du bruke en kode fra Google Autentisering-appen i tillegg til passordet ditt. Du kan bruke appen med alle e-postadresser." NB! For man må ikke ha en google-konto, ikke sant?

8- C: Hjelpetekst punkt 1. "Last ned Google Autentisering-appen (Google Authenticator) på mobilen din.

[NP-11](#) - E1.2 - Som en besøkende ønsker jeg å kunne opprette en bruker for applikasjonen slik at jeg kan

C: Endre heading Lag Bruker → Lag bruker (liten b i bruker)

Klar til Akseptanse

C: Gjenta Passord → Gjenta passord

OK, MM, 5.5.

<p>bruke den i undervisningen</p>		
<p>NP-24 - E4.1.6 - Som en lærer ønsker jeg å få informasjon/tips om hvordan de ulike metodene kan benyttes, og ha mulighet til å velge metoder slik at jeg får lagt en plan for gjennomføring av undersøkelser</p>	<p>iPad: C: Når jeg har valgt metode for en hypotese, vil jeg at denne tydelig merkes (Da jeg hadde valgt intervju var denne blitt blå, men også Egendefinert undersøkelse var markert med litt lysere lilla og en orange ramme).</p> <p>iPad: C: Når jeg har fylt inn planen, ønsker jeg at planen vises ved siden av undersøkelsesmetoden (intervju-ikon til venstre, forklaring til høyre).</p>	
<p>NP-29 - E4.1.8 - Som en lærer ønsker jeg å ha mulighet til å fylle inn tekst og velge resultat slik at jeg kan konkludere på hypotesene og skrive en endelig konklusjon på problemstillingen for prosjektet.</p>	<p>C: Vinduet hopper ikke til øverst på siden når jeg bytter trinn.</p> <p>C: Rart med placeholder under konklusjonen.</p>	
<p>NP-32 - E4.1.10 - Som en lærer ønsker jeg å se et utkast av rapporten (prosjektoppsommeringen) slik at jeg tilpasse den om nødvendig</p>	<p>C: Formattering av headings (SEO/UU)</p>	
<p>NP-38 - E4.1.13 - Som en lærer ønsker jeg å ha</p>	<p>B: Hypoteser kan ikke slettes eller settes blank (kan ikke sette tomt felt eller slette felt).</p>	

<p>mulighet til å slette utfylt data slik at jeg kan fjerne data som ikke passer inn i prosjektet</p>	<p>B: Undersøkelsermetoder kan ikke slettes helt (kan ikke sette tomt felt eller slette felt).</p> <p>B: Plan for metoder kan ikke slettes helt (kan ikke sette tomt felt eller slette felt).</p>	
<p>?</p>	<p>Dersom man ikke har opprettet prosjekt, men klikker på trinn 1 får man opp demo-data "Hvor lenge har det vært skogbruk i Norge?". . Det hadde vært fint om det var tydelig at dette var eksempeldata, og ikke reelle data.</p>	
<p>NP-37 - E4.1.12 - Som en lærer ønsker jeg å kunne endre data i utfylte felter slik at jeg har fleksibilitet i å endre data når jeg gjennomfører prosjektet</p>	<p>C: Det står ulikt på knappene ulike trinn. På 3 og 4 står det Endre plan/Endre observasjoner og så Lukk, mens på de andre står det Lagre Endringer.</p> <p>Klar til Akseptanse</p>	<p>OK, MM 5.5., Chrome</p>
	<p>FARGEKODER</p> <p>B-1: Klar til Publish</p> <p>B-1: Klar til Akseptanse</p> <p>B-1: Ikke klar til akseptanse</p>	
<p>NP-35</p>	<p>C: Endre tekst fra "Laster Opp Bilde" til "Laster opp bilde"</p>	