

Forprosjektrapport

Januar 2022

Gruppe 15 – *Intern kompetanseplattform for Fremtind Forsikring*

Av

Erik Storås Sommer, Adrian Tokle Storset,
Mats Nome Sommervold og Mathias Rundgreen

Innholdsfortegnelse

Forprosjektrapport	1
1. Presentasjon	2
1.1 Foreløpig prosjektittel	2
1.2 Kontaktpersoner	2
1.6 Om Fremtind	2
2. Sammendrag	2
3. Dagens situasjon	3
4. Mål og rammer/kravspesifikasjon MVP	3
5. Løsning	4
5.1 Oppsett	4
5.2 Hosting	5
5.3 Testing	5
5.4 Logging	5
6. Analyse av virkninger	5
7. Arbeidsplan	6
8. Fremdriftsplan	8

1. Presentasjon

1.1 Foreløpig prosjektittel

Intern kompetanseplattform for Fremtind Forsikring. Løsningens navn er "Juvet".

1.2 Kontaktpersoner

Norun Christine Sanderson
Førsteamanuensis,
Fakultet for teknologi, kunst og design,
OsloMet
Epost: nsand@oslomet.no

Erik Storås Sommer
Student

Studentnummer: s341870
Epost: erikssommer@gmail.com

Adrian Tokle Storset
Student

Studentnummer: s341859
Epost: adriantstorset@gmail.com

Mats N. Sommervold
Student

Studentnummer: s341829
Epost: mats@sommervold.net

Mathias Rundgreen
Student

Studentnummer: s341885
Epost: rundgreen@me.com

Fremtind Forsikring AS
Hammersborggata 2,
PB 778 Sentrum, 0106 Oslo
Org.nummer: 915 651 232

Erlend Tunheim
Utviklingsleder

Tlf: +47 906 49 674
Epost: erlend.tunheim@fremtind.no

Glenn A. Brownlee
Utviklingsleder

Tlf: +47 912 43 937
Epost: glenn.brownlee@fremtind.no

1.6 Om Fremtind

«Den 1. januar 2019 fusjonerte forsikringsselskapene til SpareBank 1 og DNB, og Fremtind ble født. Fremtind har et komplett forsikringstilbud til det norske markedet der vi dekker hele kundens forsikringsbehov; person, liv, eiendom og ting.» (fra fremtind.no)

2. Sammendrag

Fremtind har et behov for å samle kanaler for kompetanseheving i form av tilgjengelige kurs, fagmennesker, arrangement, fagforum, etc. i en intern webløsning. En web basert kompetanseplattform-applikasjon er spesielt etterspurt av utviklere og designere, men dette er også ønskelig å kunne implementeres på andre områder hvor det er behov i Fremtind Forsikring.

Vi skal utvikle en MVP (minimum viable product; enkleste brukbare produkt) etter gjennomføring av én ukes design sprint i starten av januar. Design sprinten vil ledes og fasiliteres av designere i Fremtind. Vi vil deretter videreutvikle denne løsningen iterativt gjennom den perioden vi har til rådighet. Løsningen skal kunne settes i produksjon og vil benyttes i det daglige i bedriften. Teamet som veileder oss er bevisst på begrensningene vi har i tid og ressurser, og ønsker å legge til rette for at vi skal kunne levere et produkt/oppgave som passer innen disse rammene.

3. Dagens situasjon

I dag ligger kurs, fagforum, påmelding til kurs, info om arrangementer og oversikt over fagpersoner spredt i ulike kanaler som Confluence, Teams, Yammer, epost, ol.. Ansatte opplever det vanskelig å holde oversikt og utfordrende å søke opp info eller fagstoff som er gitt. Aktiviteter for kompetanseheving foregår både internt og eksternt i organisasjonen, og det er ikke noen samlet plass disse blir annonsert. Fremtind betaler påmeldingsavgiften for relevante eksterne kurs/aktiviteter for den ansatte, men enkelte ansatte opplever det «knotete» å få innvilget deltakelse for eksterne aktiviteter, slik at påmeldingsprosessen rundt eksterne kurs/aktiviteter som en barriere for den type aktiviteter.

Samtidig er det en policy internt i organisasjonen at det ikke er lov å sende invitasjoner via epost til hele bedriften, slik at de ansatte er nødt til å følge med på aktuelle hendelser i relevante forum/kanaler for å holde seg oppdatert på hva som skjer.

4. Mål og rammer/kravspesifikasjon MVP

Ledestjerne:

«Om 2 år er alle ansatte i Fremtind med på min fagutvikling gjennom å bruke og oppdatere Juvet hver dag.»

Visjoner som understøtter ledestjernen:

- Sørge for at personer som ikke kunne delta på et kurs/aktivitet også får et læringsutbytte
- Skape en lav terskel for å delta på aktiviteter
- Lage en søkefunksjon som går på tvers av personer, aktiviteter og annet innhold
- Lage en søkefunksjon som gjør det mulig å søke i tidligere kurs
- Med løsningen spre relevant informasjon uten å overvelde eller komplisere
- Få en bedre oversikt over hvem som sitter på hvilken kunnskap
- «Gameifie» systemet slik at det blir mer attraktivt å delta regelmessig

Kravspesifikasjon MVP:

- Utvikle en webapplikasjon som gjør det mulig for ansatte i Fremtind å søke i aktiviteter (kurs/seminar/webinar ol. som skjer både i fortid, nåtid og fremtid) og blant kolleger som innehar relevant kompetanse for søket.
- Aktivitetene i Juvet skal være mulig å lagre på «Min side», hvor hver ansatt kan samle det innholdet fra Juvet som er interessant for den enkelte. På denne måten vil det bli lettere å få oversikt over aktiviteter som virker interessante, men som man ikke nødvendigvis har tid til å sette seg ned med på det aktuelle tidspunktet.
- Ansatte i Fremtind skal enkelt kunne opprette aktiviteter i Juvet og gjøre dette tilgjengelig for alle andre i organisasjonen.
- Ansatte skal kunne angi i sin Juvet-profil hvilken kunnskapen den enkelte ansatte er villig til å dele og lære bort til andre kolleger. Denne kompetanse-«tag'en» vil være søkbar i søkefunksjonen.
- Ansatte som ønsker å kontakte en kollega med fagkunnskap via Juvet, skal ved få trykk kunne sette opp en avtale for peer-to-peer-læring.
- Bruke Fremtinds egenutviklede design system i løsningen.

5. Løsning

Juvet vil være et oppslagsverk for kompetanseheving i Fremtind. Brukeren vil komme til en landingsside med et søkefelt, presentasjon av ulike kategorier, og foreslåtte aktiviteter som er relevante for den aktuelle brukeren. Ulike aktiviteter utgjør innholdet i løsningen, og disse vil bli presentert med en overskrift, beskrivelse, varighet, vanskelighetsgrad, dato og produsent av innholdet. En bruker skal kunne favorisere aktiviteter slik at det blir lagret til brukerens egne «min side» i *Juvet*. På denne måten vil de aktivitetene som er interessant for brukeren alltid være lett tilgjengelig på brukerens «min side». I søket på landingssiden skal brukeren også få opp kolleger med relevant kunnskap i søket. Disse skal enkelt kunne kontaktes for en rask prat om emnet, da det vil være en «kontakt meg»-lenke knyttet til denne personen. Dette vil være kjernefunksjonene i vår løsning.

Det var ingen tekniske krav fra oppdragsgiver om løsningene vi skulle bruke, men de rammeverkene Fremtind bruker sammenfaller med de vi har lært og kan godt. Derfor var valgene av disse ganske opplagte. Det vil også gi oppdragsgiver et bedre utgangspunkt for vedlikehold og videreutvikling av løsningen etter vi har levert prosjektet.

Det finnes mange ulike rammeverk for å utvikle denne type webapplikasjon. Det betyr at vi kunne ha bygget vår løsning på flere fullgode alternative måter. Men det at Fremtind er godt kjent med disse rammeverkene og at vi på gruppa ikke behøver å ha en større opplæring ved prosjektstart, gjør at dette er det foretrekkende oppsettet.

5.1 Oppsett

- GitHub
Git er en versjonshåndteringskontroll som er mye brukt. GitHub er en plattform som legger opp til deling og samskriving av kode basert på Git versjonskontrollen. Vårt prosjektbrett (Kanban board) vil også ligge her. Valget av GitHub var enkelt da dette er noe som oppdragsgiver bruker og de vil ha innsikt i vårt arbeid underveis. På denne måten vil tilbakemeldinger fra Fremtind bli enklere. I tillegg er dette et verktøy studentene er godt kjent med.
- React
React er et komponentbasert JavaScript-bibliotek for å enkelt og raskt kunne bygge logikk på klient-siden til en nettside. Fordelen med dette er at vi hurtig kan få en klient-side opp og kjøre.
- Spring Boot
Spring Boot er et Java-rammeverk for å raskt opprette og kjøre uavhengige applikasjoner. Vår backend-løsning vil basere seg på dette. Fordelen med dette er et raskt og enkelt oppsett som gjør at vi kan få en server-side av løsningen raskt opp og kjøre.
- Database
Hvilken type database vil bli bestemt på et senere tidspunkt på grunn av at vi ennå ikke vet hvilke krav som stilles til databasen.

5.2 Hosting

- Docker-container
Docker-container er en måte å pakke og kjøre koden på, slik at den kan kjøre på samme måte uavhengig av infrastrukturen i bunn. Fordelen med dette er at det gjør løsningen mer fleksibel, bla. til å flyttes fra Heroku til AWS. I tillegg gir det fordeler mtp fleksibilitet og skalerbarhet.
- Heroku (i test)
Heroku er en skytjeneste for å «hoste» applikasjoner. Det gir oss mulighet til å teste vår løsning som en skytjeneste. Vi har valgt Heroku fordi det er gratis og relativt «simpelt» i forhold til AWS som er mer kompleks i oppsettet.
- AWS (til produksjon)
AWS står for Amazon Web Services, og er en skytjeneste for å «hoste» applikasjoner. Fremtind hoster sine tjenester i AWS, og vil også legge vår løsning her. Grunnen til at vi ikke benytter dette under utviklingen er fordi AWS er mer komplisert enn Heroku og det koster penger. Vi får heller ikke tilgang til Fremtinds AWS-område da dette krever spesielle tilganger.

5.3 Testing

- JUnit
JUnit-testing er enhetstesting av Java-koden og er et veldig komplett test rammeverk hvor det er mulig å teste så og si alt på klasse- og metode-nivå. Fremtind bruker dette for enhetstesting av sine løsninger, i tillegg legger Spring Boot opp til bruk av dette, og er derfor et naturlig valg. Dette vil bli brukt til testing av vår backend og vil benyttes gjennom hele utviklingsfasen.
- Mockito eller Wiremock
Begge disse verktøyene er brukt til mock testing. Dette er testing med «mock»-innhold, det vil si oppdiktet data som er relevant for testingen av vår løsning. Disse er brukt av Fremtind i utvikling av sine løsninger, slik at det vil være lettere å få veiledning for disse verktøyene dersom det blir aktuelt. Hvilken av de som vil bli brukt avgjøres på et senere tidspunkt da det gjenstår å se hvilken av de som passer best.

5.4 Logging

- Log4J eller Logback
Begge disse verktøyene blir brukt til å logge hendelser i applikasjoner og webtjenester. Hvilken av disse to verktøyene som blir brukt er ikke avgjort ennå, men begge disse er benyttet i Fremtind fra før.

6. Analyse av virkninger

Gjennom en design sprint gjort sammen med Fremtind, kom vi fram til et tydeligere bilde av hvordan applikasjonen vil se ut. Denne sprinten tok oss gjennom fire dager med arbeid som bestod av at alle jobbet hver for seg på samme oppgave og så sammenlignet med hverandre med jevne mellomrom. I

løpet av de første to dagene fikk vi tydeliggjort hvilke problemer vi ville løse med vår applikasjon, samt jobbet oss fram til et konsept vi likte. Dette konseptet laget vi så en prototype til på dag tre og fikk testet den på fem brukere fra Fremtind dagen etter.

Det meste av analysen av hvilke virkninger denne applikasjonen vil ha for Fremtind tok plass i løpet av den første dagen. Vi kom fram til fire hovedspørsmål vi ville svare på.

- Kan vi få bedre oversikt over hvem som sitter på ulike kunnskaper?
- Kan vi spre relevant informasjon kontinuerlig uten å overvelde eller komplisere?
- Klarer vi å hjelpe 'Fremtindere' til å finne fagpersoner, kurs eller emner via Juvet?
- Kan vi "gamifye" systemet slik at det blir mer attraktivt å delta (*på aktiviteter*) regelmessig?

Disse spørsmålene representerer også i stor grad hvilke virkninger applikasjonen vil ha. Gjennom brukertesten vi avsluttet sprinten med, fikk vi tydelige svar på disse samt flere andre spørsmål. Ved å bruke vår applikasjon vil det bli enklere for ansatte å få en tydeligere oversikt over hvem som sitter med hvilken kunnskap i Fremtind. Den vil også gjøre det mye enklere å ta kontakt med fagpersoner, sette opp og delta i kurs og andre faglige aktiviteter både internt i Fremtind og eksternt. Vi vil også kunne oppfordre brukere til å ta del i faglige aktiviteter uten å skyve brukere vekk med press om konstant faglig forbedring. Når det gjelder hvordan slik oppfordring gjennomføres burde vi trå varsomt rundt 'game theory' konsepter, og være sikker på at et system er implementert på attraktiv måte.

Vi fikk også gjennom brukertesten drøftet på andre positive virkninger vår løsning kan ha for Fremtind. En plattform for konstant faglig utvikling gjort på en attraktiv måte kan gjøre Fremtind mer attraktiv som arbeidsplass for søkere, og dermed gjøre rekruttering enklere. Forenklingen av muligheten til å opprette kurs og aktiviteter vil også, sammenlignet med deres nåværende system, oppfordre ansatte til å ta del i kollegers faglige forbedring. Vår applikasjon vil da altså gjøre det mer attraktivt for ansatte hos Fremtind å heve egen faglig kompetanse, samt ta del i kollegers faglige utvikling.

7. Arbeidsplan

Med 20 uker til rådighet fra oppstart til innlevering vil vi fordele arbeidsoppgavene i ulike sprinter. Hver sprint varer i 2 uker og har hver sine hovedmål. Vi kan ikke forutse alle detaljer for arbeidsplanen, men i grove trekk vil vi fordele det på 9 sprinter etter oppstarten. Etter å ha utviklet en MVP, som er hovedmålet, vil vi videreutvikle løsningen iterativt gjennom de neste sprintene.

Oppstart:

- Gjennomføre design sprint (uke 1)
 - o Avgjøre ledestjerne og hovedmål
 - o Lage prototype
 - o Brukertest av prototype
 - o Analysere innsikt i brukertest
- Planlegge gjennomføring (uke 2)
 - o Avklare primær arbeidstid for gruppa
 - o Definere MVP
 - o Avklare hva som skal inngå i det tekniske oppsettet
- Sette opp utviklingsmiljø (uke 2)
 - o Gjennomføre React-intro

Sprint 1: Oppstart utvikling av MVP (uke 3 og 4)

- Få innføring i hvordan Fremtind setter opp Spring Boot
- Lage prototyper versjon 2 - få et klart bilde på hvilke sider som skal lages
- Starte på utvikling av frontend og backend for de avtalte sidene, med hovedvekt på landingssiden, søk og «min side»

Sprint 2: Utvikling MVP (uke 5 og 6)

- Ferdigstille første utkast på frontend og backend
- Avgjøre database-løsning, og verktøy for mock-testing
- JUnit-testing underveis
- Mock-testing underveis

Sprint 3: Ferdigstilling av MVP (uke 7 og 8)

- Utvikling av MVP som svarer til kravspesifikasjonen
- JUnit-testing underveis
- Mock-testing underveis
- Brukertesting av MVP

Sprint 4: Første iterasjon for forbedring av MVP (uke 9 og 10)

- Forbedringer av løsningen basert på brukertest
- JUnit-testing underveis
- Mock-testing underveis

Sprint 5: Andre iterasjon med videreutvikling (uke 11 og 12)

- Avgjøre hva som skal videreutvikles/evt implementeres av ny funksjon
- Implementere valgte videreutvikling/funksjon
- JUnit-testing underveis
- Mock-testing underveis
- Brukertest 1 av løsningen

Sprint 6: Tredje iterasjon med videreutvikling (uke 13 og 14)

- Avgjøre hva som skal videreutvikles/evt implementeres av ny funksjon
- Implementere valgte videreutvikling/funksjon
- JUnit-testing underveis
- Mock-testing underveis

Sprint 7: Ferdigstille løsningen (uke 15 og 16)

- Starte på sluttrapport (2 av teammedlemmene jobber kun med denne i slutfasen)
- Få et klart bilde på hva som er endelig mål for siste iterasjon. Hvilken funksjonalitet blir den siste vi implementerer?
- Mock-testing underveis
- Brukertest 2 av løsningen

Sprint 8: Fokus på sluttrapport og finpuss på løsning (uke 17 og 18)

- I stor grad ferdigstille sluttrapport
- Ferdigstille løsning og flytte løsning fra Heroku til AWS (hvis aktuelt)

Sprint 9: Finjustering (19 og 20)

- Finjustering av løsning og sluttrapport
- Overlevere løsning til prosjektgive
- Forberede presentasjon

Gjennom hele prosjektet vil vi dokumentere prosessen med en prosjektdagbok i Word og et kanban-board for arbeidsflyten på GitHub.

8. Fremdriftsplan

Følgende Grantt-diagram viser de ulike fasene, hva som inngår av aktiviteter og periode for gjennomførelse.

