



Ignite Procurement

Forprosjektrapport *Gruppe 47*



Innhold

1. Presentasjon	3
1.1 Oppdragsgiver	3
1.2 Kontaktperson	3
1.3 Veileder tildelt hos Ignite	3
1.4 Intern veileder	3
1.5 Gruppemeddlemmer	3
1.6 Oppgave	4
2. Sammendrag	4
3. Dagens situasjon	4
4. Mål og rammebetingelser	4
4.1 Mål for applikasjonen	4
4.2 Rammebetingelser	5
4.3 Teknologier/Verktøy	5
5. Løsninger/alternativer	6
6. Analyse av virkninger	9

1. Presentasjon

1.1 Oppdragsgiver

Ignite

Grundingen 2, 0250 Oslo

Tlf: +47 944 88 923

Ignite er et SaaS-selskap og startet opp i 2016. De legger vekt på å rette for økt lønnsomhet og en enklere hverdag i arbeidet med strategisk innkjøp.

1.2 Kontaktperson

Sigbjørn Nome | CEO & Product Director

Epost: sigbjorn.nome@ignite.no

Tlf: +47 944 88 923

1.3 Veileder tildelt hos Ignite

Valdemar Rolfsen | CTO

Epost: valdemar.rolfsen@ignite.no

Tlf: +47 993 22 797

1.4 Intern veileder

Qian Meng

Epost: qiangmeng@oslomet.no

Telefon: +47 413 93 020

1.5 Gruppemedlemmer

Huy Minh Le
s315271@oslomet.no
+47 916 13 284

Piotr Jan Kusnierz
s326178@oslomet.no
+47 483 87 279

Tam Thu Caroline Vo
s326167@oslomet.no
+47 476 41 271

1.6 Oppgave

Utvikle en applikasjon som:

- Gir brukeren(ansatt hos en bedrift) muligheten til å vurdere leverandører bedriften har hatt kontrakt med.
- Gir brukeren oversikt over eksisterende kontrakter.
- Gir brukeren mulighet til å endre på sin egen profil.

2. Sammendrag

Vi skal utvikle en mobilapplikasjon for Ignite som har målgruppen ansatte hos bedrifter. Med applikasjonen kan ansatte hos bedrifter vurdere andre bedrifter som de har hatt kontrakt med, gi tilbakemeldinger og søke opp bedrifter som har en tjeneste eller produkt de trenger. Hovedfokuset kommer til å være på vurderingsfunksjonen hvor vurderingen vil gi verdifull informasjon til fremtidige bedrifter som skal inngå kontrakter, i tillegg til å hjelpe Ignite sine algoritmer. Mobilapplikasjonen skal skrives i Typescript og React Native slik at den kan kjøres på både Android og IOS enheter. Backend API-er skal skrives i Django (Python).

3. Dagens situasjon

Ignite i dagens situasjon er en analyse selskap som har mange store og små bedrifter som kunder. Ignite tilbyr avanserte algoritmer og imponerende brukergrensesnitt for å presentere verdifull data om innkjøp og kontrakter for bedrifter. Tjenesten gir brukeren mulig til å ha full kontroll over de mange kontraktene bedriften har og hjelper brukeren til å finne leverandører som passer dem best i forhold til strategisk innkjøp.

Vanligvis er brukeren av Ignite sine tjenester personer som har kontroll over bedriftens kontrakter og innkjøp. Nå vil Ignite, i samarbeid med oss, lage en mobilapplikasjon som er for ansatte. Denne applikasjonen har i hovedfokus å gjøre det mulig for brukeren å vurdere leverandører som brukerens bedrift har vært i samarbeid med, noe som gir god og verdifull data til fremtidige bedrifter som også skal inngå avtaler med leverandøren og til Ignite sine algoritmer.

4. Mål og rammebetingelser

4.1 Mål for applikasjonen:

- Bruker skal ha mulighet til å logge inn med bruker fra Ignite.
- Bruker skal ha mulighet til å se og endre sin egen profil.
- Brukeren skal ha oversikt over kontrakter med leverandører, i tillegg til å filtrere kontraktene.
- Det skal være mulig å vurdere leverandører som brukeren har hatt kontrakt med. Det skal være mulig å vurdere ulike aspekter som kommunikasjon, verdi osv.

4.2 Rammebetingelser:

- Mobilapplikasjonen skal utvikles i React Native slik at den kan kjøres på både Android og IOS.
- Applikasjonens grafiske design skal følge Ignite sitt fargepalett og design.
- Gruppemedlemmer skal benytte samme arbeidsmetodikk, teknologi og arbeidsspråk som oppdragsgiveren.
- Applikasjonen skal være lastet opp på deres private github hvor dokumentasjon og automatiserte tester skal kjøres.
- Vi må følge Ignite sin mappestruktur.
- Vi må implementere applikasjonen med styled components.
- Utviklingen og dokumentasjon skal være ferdig til 25.05.2020
- Sluttpresentasjonen skal være klar uke 24, 8-12 juni 2020

4.2 Rammebetingelser:

Frontend teknologier:

- **React:** *Et Javascript-rammeverk for oppbygging av brukergrensesnittet som er utviklet og vedlikeholdt av Facebook.*
- **React native:** *Utvikler mobilapplikasjoner ved å benytte JavaScript og React.*
- **Typescript:** *Et Open Source programmeringsspråk utviklet og vedlikeholdt av Microsoft. Er basert på Javascript, men har mulighet til å være et objektorientert språk. Typescript kode blir compilert til Javascript kode før den kjøres.*
- **Javascript:** *Et høynivå-programmeringsspråk, og benyttes til å utvikle nettsider og applikasjoner.*

Backend teknologier:

- **Python:** *Et høytnivå-programmeringsspråk og generell programmeringsspråk, der det benyttes på en server til å utvikle webapplikasjoner.*
- **Django:** *Et Open Source webrammeverk som er skrevet i Python.*

Verktøy

- **Github:** *En plattform for hosting for versjonskontroll og samarbeid. Det lar oss jobbe og samarbeide med prosjekter hvor som helst og når som helst.*
- **Trello:** *Et nettbasert prosjektverktøy som benyttes for å holde orden i gjøremålene for alle som jobber i grupper.*
- **Visual studio code:** *En Open-Source editor utviklet av Microsoft for Windows, Linux og macOS. Den er basert på Electron, et rammeverk som brukes til å utvikle Node.js-applikasjoner. [1]*
- **PyCharm:** *Et integrert utviklingsmiljø (IDE) som brukes i programmering, spesielt for Python-språket.*
- **Android Studio:** *Et integrert utviklingsmiljø som brukes i utvikling av android applikasjoner. Den kommer innebygd med en Android Emulator som vi bruker for å teste hvordan applikasjonen vår ser ut i android enheter.*
- **Expo:** *Verktøy som er bygget rundt react native som gjør det lettere å lage applikasjoner og teste applikasjonene på telefoner uten å laste ned store mengder data hver gang applikasjonen skal testes.*
- **CircleCI:** *Verktøy som vi knytter til vår github for å automatiserer tester. Dette er samme verktøy som Ignite bruker.*

5. Løsninger/alternativer

Ignite var bestemt fra starten at de ønsker seg en mobilapplikasjon. Vi har da ikke mange alternativer på hvordan sluttproduktet blir, men vi har fått frihet til å velge arbeidsmetodikk som vi skal benytte i selve utviklingsprosessen. Alternativene vi har kommet oss frem til er mellom Scrum eller Kanban metodikk.

Scrum - bygger på samarbeide innenfor team fra tre til ni medlemmer som bryter ned oppgaver til prosjekter som kan ferdigstilles innen gitte tidsrom, såkalte «sprint», som vanligvis strekker seg fra to uker til én måned. Metoden omfatter også daglige vurderings- og planleggingsmøter som ikke skal vare lenger enn ett kvarter.[\[2\]](#)

Kanban - er en måte for å begrense igangsatt arbeid i et produksjonsanlegg. En ny jobb må vente hvis antall påbegynte jobber har nådd en grense (MAX). Så snart en jobb er ferdig, kan neste ventende jobb settes i gang. [\[3\]](#)

	Fordeler	Ulemper
Scrum	<ul style="list-style-type: none"> • Faste og tette ngen • Gjennomgang av hver sprint før man går over til neste trinn. • Resulterer vanligvis raskere tid til marked. 	<ul style="list-style-type: none"> • Daglige Scrum-møter og anmeldelser krever betydelige ressurser. • Vellykket prosjekt er avhengig av deltakere, der det må opprettholdes gjennomgående kommunikasjoner gjennom prosjektet.
Kanban	<ul style="list-style-type: none"> • Enkelt å lære og implementere. • Øker produktivitet og effektivitet. 	<ul style="list-style-type: none"> • Utdatert “Kanban board” kan lede til problemer i utviklingsprosess. • Kanban antar stabil og repeterende produksjonsplaner.

Vi har også alternativet til å velge hva slags spørringsspråk vi skal bruke. Ignite har nylig bestemt seg til å prøve å implementere GraphQL i fremtidige, og har gitt oss valget om vi også vil prøve oss på GraphQL hvis vi hadde lyst.

	Fordeler	Ulemper
GraphQL	<ul style="list-style-type: none"> • Hente mange ulike ressurser i et enkelt forespørsel. • Klienten har muligheten til å diktere nøyaktig hva de trenger fra serveren. 	<ul style="list-style-type: none"> • Mangel på innebygd cache-støtte. • Man kan tillate/ ikke tillate noen få felt. Noe som fører til ekstra spørring.

SQL	<ul style="list-style-type: none"> • Rask og effektivt for brukeren å hente en stor mengde data fra en database. • Enkelt å administrere i databasesystemet. • Brukeren kan lage ulike oversikt/views av databasestrukturen. 	<ul style="list-style-type: none"> • Kompleks struktur gjør det vanskelig for brukere å få tilgang til data i databasen. • SQL operasjoner er veldig kostelig.
------------	---	--

Når det gjelder CI (Continuous Integration) verktøy er vi fritt til å velge mellom hvilken som helst. Forslag som vi fikk fra oppdragsgiveren var TravisCI, Github Actions og CircleCI. Ignite bruker i dag CircleCI for deres systemer. Det er fordeler og ulemper ved alle disse verktøy.

	Fordeler	Ulemper
TravisCI	<ul style="list-style-type: none"> • Har integrert build matrix • Starter kjapt • Lett og leselig YAML konfigurasjonsfil • Har en gratis plan for open-source prosjekter • Er skybasert så trenger ingen dedikerte servere for å kjøre 	<ul style="list-style-type: none"> • Er dyrere enn andre CI hvis man skal integrere det i større prosjekter som ikke er open-source • Trenger 3. parti programvarer for å tilpasse noen funksjonaliteter
CircleCI	<ul style="list-style-type: none"> • Hurtig og lett å starte • Mulig å starte med en gratis plan • Tar ikke mye plass • Lett og leselig YAML konfigurasjonsfil • Er skybasert så trenger ingen dedikerte servere for å kjøre 	<ul style="list-style-type: none"> • Problemer kan forekomme når man prøver å tilpasse funksjonaliteter • Kan alltid slutte å støtte forskjellige programvarer
Github Actions	<ul style="list-style-type: none"> • Veldig lett å resonnerer og feilsøke. 	<ul style="list-style-type: none"> • Ikke så effektiv eller bra måte til unit-teste actions.

	<ul style="list-style-type: none"> • Allerede åpen kilde-kodeimplementerin av GitHub actions, f.eks en actions for lokal testing. 	<ul style="list-style-type: none"> • Kvaliteten og bredden av publiserte Github Actions er fremdeles ganske lav eller begrenset.
--	--	---

6. Analyse av virkninger

Siden vi er invitert til å delta i daglige standup møter så tenker vi at det blir spennende å følge deres arbeidsmetodikk som er Scrum. Vi tror også at faste oppfølgingsmøter kommer til å forbedre utviklingsprosessen. Vi forutseer at vi blir i stand til å utlevere resultater mye raskere, og siden hovedfunksjonalitetene allerede er bestemt i applikasjonen vår så ser vi for oss at Scrum passer best for oss.

Når det gjelder spørringsspråk så har vi ikke bestemt oss enda hvilken alternativ vi går for. For øyeblikket trenger vi å sette oss mer inn i GraphQL for å vurdere om det er verdt tidsinvesteringen når vi allerede har kompetanse i SQL. GraphQL har fordelen at man kan få mer data ved mindre spørringer, men det er vanskeligere å starte å lære det.

Til slutt så vi tenkte vi å velge CircleCI som vår CI. Dette er samme verktøy som Ignite bruker i dag. En stor fordel ved at vi også bruker CircleCI er at vi kan få veiledning og hjelp til å sette og vedlikeholde hvis noen feil dukker opp. Dette vil spare oss tid og krefter som kan bli fokusert på andre aspekter av prosjektet.

Kilder

[1]. https://en.wikipedia.org/wiki/Visual_Studio_Code

[2]. <https://no.wikipedia.org/wiki/Scrum>

[3]. <https://no.wikipedia.org/wiki/Kanban>