

# FORPROSJEKT - RAPPORT

## FOR BACHELOROPPGAVE

TITTEL:  
MÅOYI MOBIL APP (APPLIKASJON FOR HANDEL AV SJØMAT)

KANDIDATNUMMER(E):

DATO:	EMNEKODE: * <b>IE303612</b>	EMNE: <b>Bacheloroppgave (Data)</b>	DOKUMENT TILGANG: - Åpen
STUDIUM: <b>BACHELOR I INGENIØRFAG - DATA</b>	ANT SIDER/VEDLEGG: 15/0	BIBL. NR: - Ikke i bruk -	

OPPDRAGSGIVER(E)/VEILEDER(E):

Kurt Louis Skjong (Giske kystfiske AS)  
Mikael Tollefsen (NTNU)

OPPGAVE/SAMMENDRAG:

Gruppen skal utvikle en applikasjon for å knytte fiskere og potensielle kjøpere med hverandre. Applikasjonen skal ha mulighet for å kommunisere mellom partene for å bestemme møtested og pris.



## INNHold

<b>INNHold</b>	<b>3</b>
<b>1 INNLEDNING</b>	<b>4</b>
<b>2 BEGREPER</b>	<b>4</b>
<b>3 PROSJEKTORGANISASJON</b>	<b>4</b>
3.1 PROSJEKTGRUPPE	4
3.1.1 Oppgaver for prosjektgruppen - organisering	4
3.1.2 Oppgaver for prosjektleder	5
3.1.3 Oppgaver for sekretær	5
3.1.4 Oppgaver for øvrige medlem(mer)	4
3.2 STYRINGSGRUPPE (VEILEDER OG KONTAKTPERSON OPPDRAGSGIVER)	5
<b>4 AVTALER</b>	<b>5</b>
4.1 AVTALE MED OPPDRAGSGIVER	5
4.2 ARBEIDSTED OG RESSURSER	5
4.3 GRUPPENORMER – SAMARBEIDSREGLER – HOLDNINGER	6
<b>5 PROSJEKTBESKRIVELSE</b>	<b>7</b>
5.1 PROBLEMSTILLING - MÅLSETTING - HENSIKT	7
5.2 KRAV TIL LØSNING ELLER PROSJEKTRESULTAT – SPESIFIKASJON	7
5.3 PLANLAGT FRAMGANGSMÅTE(R) FOR UTVIKLINGSARBEIDET – METODE(R)	8
5.4 INFORMASJONSINNSAMLING – UTFØRT OG PLANLAGT	8
5.5 VURDERING – ANALYSE AV RISIKO	9
5.6 HOVEDAKTIVITETER I VIDERE ARBEID	10
5.7 FRAMDRIFTSPLAN – STYRING AV PROSJEKTET	11
5.7.1 Hovedplan	11
5.7.2 Styringshjelpemidler	12
5.7.3 Utviklingshjelpemidler	12
5.7.4 Intern kontroll – evaluering	12
5.8 BESLUTNINGER – BESLUTNINGSPROSESS	13
<b>6 DOKUMENTASJON</b>	<b>13</b>
6.1 RAPPORTER OG TEKNISKE DOKUMENTER	13
<b>7 PLANLAGTE MØTER OG RAPPORTER</b>	<b>14</b>
7.1 MØTER	14
7.1.1 Møter med styringsgruppen	14
7.1.2 Prosjektmøter	14
7.2 PERIODISKE RAPPORTER	14
7.2.1 Framdriftsrapporter (inkl. milepæl)	14
<b>8 PLANLAGT AVVIKSBEHANDLING</b>	<b>14</b>
<b>9 UTSTYRSBEHOV/FORUTSETNINGER FOR GJENNOMFØRING</b>	<b>14</b>
<b>10 REFERANSER</b>	<b>15</b>
<b>VEDLEGG</b>	<b>8</b>

## 1 INNLEDNING

Høsten 2020 bestemte gruppen seg for å jobbe sammen om bacheloroppgaven som skulle skrives våren 2021. Sammen rangerte gruppen tre oppgaver som virket interessant å skrive oppgave om. Gruppen tildelt en oppgave som går ut på å lage en applikasjon for salg av fisk og skalldyr.

Oppdragsgiver er Giske Kystfiske AS er en lokal aktør innenfor fiskerinæringen, og er oppdragsgiver for dette prosjektet. I hovedsak baserer bedriften seg på teinefiske etter hummer, krabbe og kreps.

Vi skal planlegge og utvikle en handelsplattform, som skal forenkle handel mellom fisker og sluttbruker. Konkret vil tjenesten bestå av en applikasjonsserver driftet av oppdragsgiver og en mobilapplikasjon installert hos brukerne.

## 2 BEGREPER

API – (Application Programming Interface), programmeringsgrensesnitt

JPA – Jakarta Persistence, en del av Jakarta EE

SDK – Software development kit, samling av programvare utviklings verktøy

APK – Android package, pakkefilformat brukt av Android

Sprint – del av Scrum; en fast leveransesyklus, der utviklerne planlegger og utvikler neste iterasjon

Scrum-master – rolle i Scrum, sørger for at metodikken blir etterlevd i prosjektet

## 3 PROSJEKTORGANISASJON

### 3.1 Prosjektgruppe

Studentnummer	Navn	Mobiltelefon	e-post
498789	Erlend Solbakken Nikolaisen	40625699	<a href="mailto:erls@stud.ntnu.no">erls@stud.ntnu.no</a>
507879	Trygve Johansen Woldseth	45484139	<a href="mailto:Trygveiw@stud.ntnu.no">Trygveiw@stud.ntnu.no</a>
498788	Nils-Jarle Haugen	95927996	<a href="mailto:nilsjarh@stud.ntnu.no">nilsjarh@stud.ntnu.no</a>

#### 3.1.1 Oppgaver for prosjektgruppen – organisering

Prosjektgruppen har bestemt seg for at Trygve blir prosjektleder og hele gruppen rullerer på rollen som sekretær.

### 3.1.2 Oppgaver for prosjektleder

- Scrum-master
- Planlegge progresjonen slik at hele prosjektet blir ferdig i tide
- Følge med at mål blir oppnådd i tide
- Kommunikasjon med oppdragsgiver/ veileder
- Lede agenda under møte med veileder og/eller oppdragsgiver

### 3.1.3 Oppgaver for sekretær

- Sørge for at nødvendig informasjon blir notert ned til bruk videre i prosjektet
- Føre referat under møter med veileder og/eller oppdragsgivere

## 3.2 Styringsgruppe (veileder og kontaktperson oppdragsgiver)

Veileder:

- Mikael Tollefsen, mikael.tollefsen@ntnu.no

Oppdragsgiver:

- Kurt Louis Skjong, kurt@giskeskjong.no

## 4 AVTALER

### 4.1 Avtale med oppdragsgiver

Oppdragsgiver har ikke hatt mulighet til å ha møte med prosjektgruppen fordi han jobber på båt. Første møte med oppdragsgiver vil ikke skje før innleveringsfristen for denne rapporten, og derfor vil det ikke være mulig på nåværende tidspunkt å skrive om noen avtale.

Prosjektgruppen ser for seg å inngå avtale med oppdragsgiver så snart dette er mulig.

### 4.2 Arbeidssted og ressurser

Prosjektgruppen hadde sett for seg i hovedsak å arbeide sammen på campusområdet, hovedsakelig rom L167. Grunnet koronasituasjonen per dags dato, vil det meste av arbeid foregå fra hjemmekontor. Gruppen bruker discord for kommunikasjon.

De fleste resursene som gruppen ser for seg å bruke i dette prosjektet er enten kostnadsfrie, eller har studentløsninger som gruppen ikke trenger å betale for. Mye av redskapene prosjektgruppen tenker å bruke er gratis programvare. Derfor ser vi ikke for oss å bruke penger på programvarelisenser.

Utviklingsprosess og drift av applikasjonen vil foregå på standard PC-plattform. Derimot vil kompilering til Apple sin iOS-plattform kreve utviklingsverktøyet Xcode og Apple SDK-er, som bare kan kjøres på Macintosh-maskiner. Strengt lisensieringskrav forhindrer kjøring av Xcode på PC-plattform. Ved absolutt krav fra oppdragsgiver om å kompilere applikasjon mot iOS, kan dette medføre økt kostnad i form av innkjøp eller leie av nødvendig Macintosh-maskinvare (Apple 2019).

Prosjektgruppen vil undersøke mulighetene for å gjøre dette på billigst mulig måte. Hvis oppdragsgiver ser behov for å publisere applikasjon til Apple App Store, vil årlig lisensavgift komme i tillegg.

Utover dette vil mesteparten av arbeidet utføres på utstyr prosjektgruppen i hovedsak benytter daglig i sammenheng med studiene. Gruppen ser ikke for seg at prosjektet vil innbringe noen kostnader i hardware.

Som nevnt i punkt 4.1, er situasjonen slik at møte med oppdragsgiver er utsatt. Dermed kan ikke spørsmål vedrørende offentlighet avklares før dette møtet har funnet sted. Prosjektgruppen har derfor bestemt seg for å følge føre-var prinsippet, og behandler prosjektinformasjon (kildekode og dokumentasjon) som ikke-offentlig frem til at en har diskutert problemstillingen med oppdragsgiver.

For å sørge for god flyt i prosjektet har gruppen kommet frem til å holde et oppdateringsmøte med veileder hver andre uke. En vil der diskutere den nåværende progresjonen i prosjektet, eventuelle endringer, og planlegging videre. Roller som sekretær rulleres på mellom gruppemedlemmene. Mot oppdragsgiver ser gruppen for seg å rapportere i etterkant av hver sprint, i beste fall som møte, eller i skriftlig form som e-post hvis dette ikke er mulig.

### **4.3 Gruppenormer – samarbeidsregler – holdninger**

I bacheloroppgaven skal prosjektgruppen følge arbeidsmetoden *Scrumban*, og det skal brukes sprinter på 2 uker. Gruppen skal arbeide minst 30 timer hver uke, kjernetid fra 09.00 - 14.30 med mulighet for fleksitid. Hver morgen skal gruppen gjennomføre et stand-up møte der det fortelles hva som ble gjort forrige dag, hva som skal fokusere på den dagen, og hvilke utfordringer hvert gruppemedlem har. Oppgaver som registreres i *Scrumban*, legges inn i en *backlog* så utviklingsarbeidet blir loggført.

Ved *Scrumban* blir en arbeidsoppgave tildelt en enkelt person, men gruppemedlemmene er enige om at det ikke betyr at ansvaret for at oppgaven blir godt utført faller kun på personen som blir tildelt oppgaven. Alle gruppemedlemmene har et ansvar for å ha en helhetlig forståelse for hverandres arbeid og alle deler av det ferdige prosjektet.

Gruppemedlemmene har utarbeidet en arbeidskontrakt seg imellom som alle på gruppen skriver under på. Dette sikrer gode holdninger til samarbeid.

Gruppemedlemmene forplikter seg også til å påta seg nødvendig arbeid utenfor avtalt arbeidstid ved behov. Dette inkluderer også ekstra tid som må brukes for å tilegne seg nytt materiale. Prosjektgruppen skal gjøre sitt beste for å imøtekomme krav og ønsker fra oppdragsgiver, til den grad det er fornuftig. Ellers sikter gruppen på å holde kommunikasjonen med oppdragsgiver så åpen som mulig, slik at de er innforstått med alle avgjørende valg som tas underveis. Dette hjelper også med å sikre at det ferdige produktet tilfredsstiller alle parter.

## 5 PROSJEKTBESKRIVELSE

### 5.1 Problemstilling - målsetting - hensikt

Giske Kystfiske AS er en lokal aktør innenfor fiskerinæringen, og er oppdragsgiver for dette prosjektet. I hovedsak baserer bedriften seg på teinefiske etter hummer, krabbe og kreps.

Som oppdragsgiver, ser de behov for en enklere løsning for salg og kjøp av sjømat. Prosjektgruppen sin problemstilling blir da i hovedsak å utvikle en applikasjon som skal være et mellomledd mellom kjøper og selger. Den skal gjøre transaksjonene enklere og tryggere mellom partene.

Effektmål:

- Gjøre det mulig for fiskere å legge dagens fangst ut til salg
- Gjøre det mulig å kjøpe fersk fisk rett fra en app
- Gjøre det mulig med forhåndsbestilling av fangst

Resultatmål:

- Ferdig applikasjon skal være fungerende etter kravspesifikasjonen
- Brukervennlig applikasjon
- Applikasjonsserver skal være enkel å drifte
- Mobilapplikasjonen skal være enkel å vedlikeholde

Prosessmål:

- Tilegne seg mer kunnskap innenfor applikasjonsutvikling
- Få mer kunnskap av prosessen fra idé til ferdig produkt
- Få en god karakter på prosjektet
- Lære seg å bruke Flutter til applikasjonsutvikling

### 5.2 Krav til løsning eller prosjekresultat – spesifikasjon

I applikasjonen skal en selger ha mulighet til å legge dagens fangst ut til salgs med ønsket pris. En kjøper skal kunne bla gjennom de tilgjengelige annonsene for å se hva som er til salgs. En kjøper skal også kunne legge ut en bestilling på hva han ønsker å kjøpe. Selger og kjøper skal ha mulighet til å kommunisere med hverandre. Applikasjonen skal og være tilgjengelig både på Android og iOS.

Kart og GPS integrasjon skal være i appen, slik at selger kan i annonser legge ved hvor kjøper må dra for å ta imot fisken.

For at både kjøpere og selgere skal være motivert til å ta i bruk applikasjonen, er det viktig at den er enkel å bruke.

Som nevnt tidligere i denne rapporten, har ikke gruppen mulig til å ha møte med oppdragsgiver på dette tidspunktet. Dette har ført til at det ikke har blitt diskutert hva

som skal anses som fullføring av prosjektet. For gruppemedlemmene anses prosjektet som fullført når applikasjonen fungerer ut fra kravspesifikasjonen laget av oppdragsgiver.

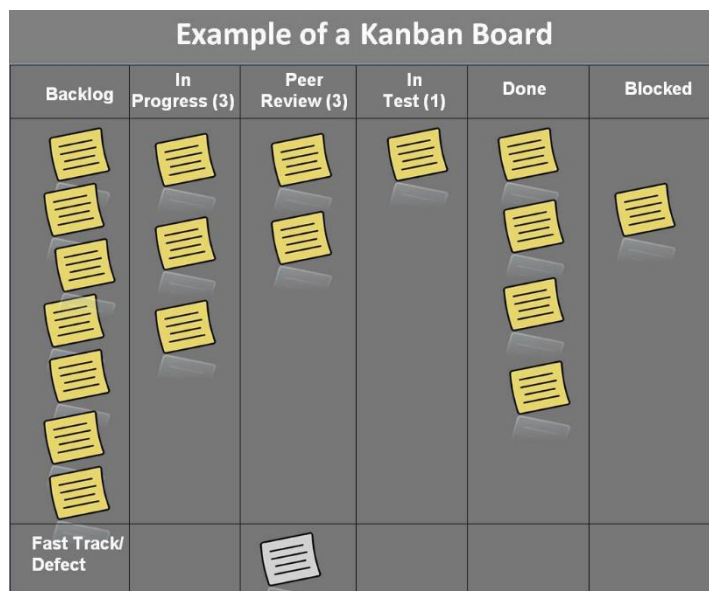
### 5.3 Planlagt framgangsmåte(r) for utviklingsarbeidet – metode(r)

Prosjektgruppen har sett på flere agile utviklingsmetoder, deriblant Extreme programming (XP), Scrum, og Lean software development. Det ble bestemt å bruke en agil metode fordi nødvendige endringer oppdages rask på grunn av en kortere utviklingssyklus. Den kortere syklusen gjør også at oppdragsgiver blir en del av utviklingen og kan rette opp i misforståelser før de blir et problem.

Prosjektgruppen ble enig om å bruke den agile utviklingsmetoden Scrum, med en tavle fra Kanban. Scrum ble valgt blant annet fordi XP har en del forhåndsbestemte utviklingsmetoder som omfattende kodegjennomgang og enhetstesting av all kode. Gruppen var enig i at slike arbeidsmetoder ville ta ekstra tid som heller kunne brukes på annet. Scrum er også en arbeidsmetode som alle gruppemedlemmene allerede er kjent med.

I Scrum blir arbeidsoppgaver brutt opp i mål som kan fullføres på korte tidsiterasjoner, kalt sprinter. Scrum-teamet sporer fremgangen ved å ha daglige 15 minutters møter. Ved slutten av hver sprint har teamet en "sprint-review" og "retrospective", for å vise utført arbeid og kunne kontinuerlig forbedre der som er gjort (Schwaber and Sutherland, 2020).

Kanban brukes til å visualisere arbeidsoppgaver og -mengde i et prosjekt. Dette gjøres ofte ved å bruke en kanbantavle for å vise oppgaver og hvilken tilstand oppgavene er i (Kanban, 2021). Figur 1 viser et eksempel på en slik tavle.



Figur 1: Eksempel på Kanbantavle (hentet fra [3])

### 5.4 Informasjonsinnsamling – utført og planlagt

Gruppen fant ingen løsning på markedet per dags dato som dekker noen stor andel av kravspesifikasjonen. Det nærmeste som ble funnet var bestillefisk.no som tilbyr bestilling av fisk til dør, men denne tjenesten er en nettside og ikke en app.



Gruppen har så langt i prosjektet gjort undersøkelser angående hvilke programvare-rammeverk som vil være nødvendig for å kunne gjennomføre oppgaven. Gruppen har også undersøkt de forskjellige løsningene for å integrere betaling i appen, siden det er noe ingen på gruppen har gjort før.

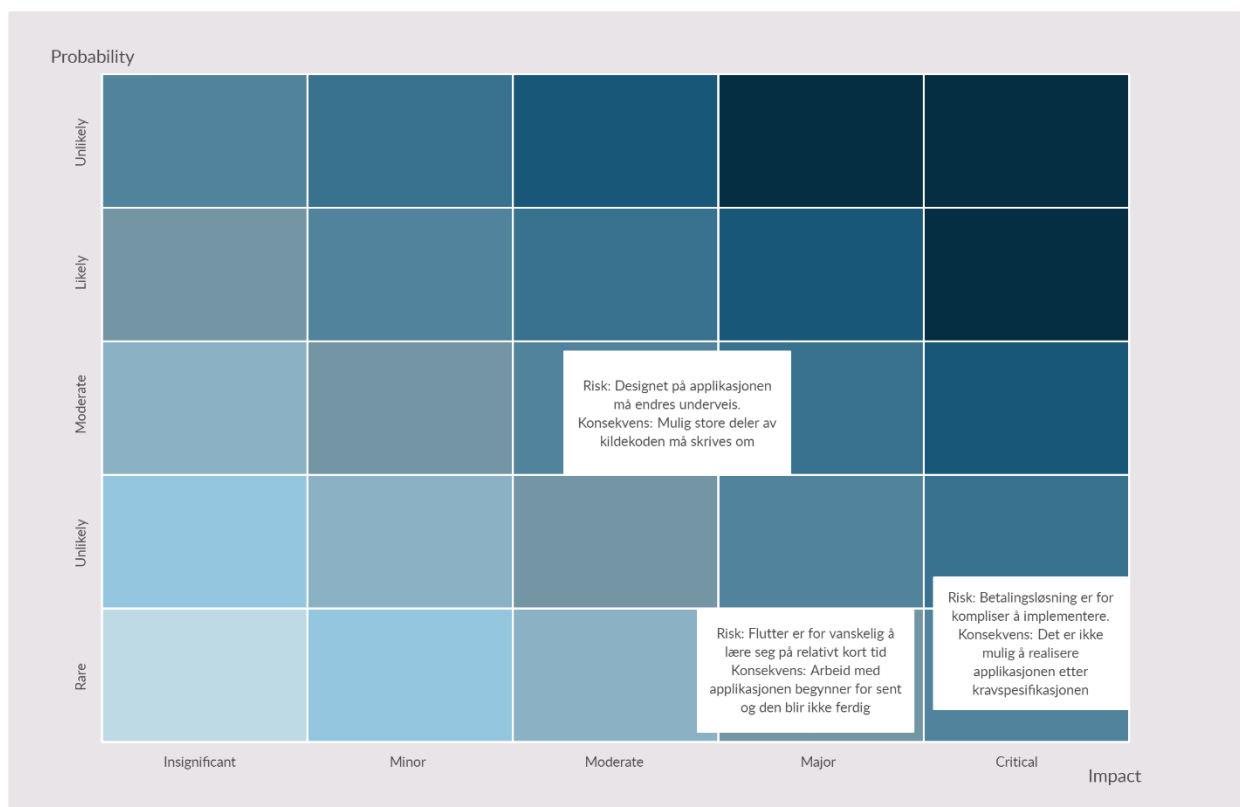
Framover vi gruppen trenge å undersøke videre hvilke betalingsløsninger gruppen skal bruke, og hvordan implementere de på en sikker og stabil måte.

### ***5.5 Vurdering – analyse av risiko***

Gruppen har laget lignende applikasjoner tidligere, og mener derfor at risikoen for at prosjektet ikke realiseres innenfor den rammen som er gitt er lav. Et element som øker risikoen for at prosjektet ikke realiseres, er at ingen av gruppemedlemmene har implementert betalingsløsninger før. Om det oppstår store problemer med å implementere betalingsløsning, som ønsket av oppdragsgiver, vil det være en trussel for at prosjektet ikke kan realiseres før tidsfristen.

Gruppemedlemmene har ikke hatt mulighet til å holde møte med oppdragsgiver før arbeidet med applikasjonen har startet. Dette, sammen med hvordan applikasjoner designes i Flutter, gjør at gruppen må ta en avgjørelse på utseende til applikasjonen. Det er da en reell risiko for at oppdragsgiver ønsker et annet design, som fører til at deler av kildekoden må skrives om.

Aktuelle risikoer vises i risiko-diagrammet nedenfor. Jo mer sannsynlig at en risiko inntreffer, desto lengre opp i matrisen vil risikoen være plassert. Jo større konsekvens risikoen vil ha, desto lengre mot høyre i matrisen vil risikoen være plassert.



Figur 2: Risiko-matrise (laget med [4])

### 5.5.1 Korrigerende tiltak for å kontrollere risiko

**Risk:** Designet på applikasjonen må endres underveis.

**Konsekvens:** Mulig store deler av kildekoden må skrives om

**Tiltak:** Sørge god kommunikasjon mot oppdragsgiver, slik at en til enhver tid er orientert om ønsket design.

**Risk:** Betalingsløsning er for komplisert å implementere.

**Konsekvens:** Det er ikke mulig å realisere applikasjonen etter kravspesifikasjonen

**Tiltak:** Undersøke valgmuligheter for betalingsleverandør tidlig i prosessen. Dette muliggjør at valgt betalingsleverandør kan involveres tidlig i prosjektet.

**Risk:** Flutter er for vanskelig å lære seg på relativt kort tid

**Konsekvens:** Arbeid med applikasjonen begynner for sent og den blir ikke ferdig.

**Tiltak:** Legge opp til at medlemmene kan lære seg Flutter, ved å sette av tid til opplæring og utforsking.

### 5.6 Hovedaktiviteter i videre arbeid

Nr	Hovedaktivitet	tid
<b>A0</b>	<b>Innhenting av informasjon</b>	<b>N/A</b>
<b>A1</b>	<b>Prosjektering av applikasjon og system</b>	<b>3t</b>

A1.1	Lage User Stories basert kravspesifikasjon	1t
A1.2	Wireframes for utseende for app	1t
A1.3	Design modell for datastruktur	1t
<b>A2</b>	<b>Sette opp driftsmiljø for applikasjonsserver</b>	<b>11t</b>
A2.1	Sette opp kontainerbasert kjøremiljø	8t
A2.2	Installere og sette opp database (relasjonsdatabase)	1t
A2.3	Forberede server for kommunikasjon (med HTTPS)	2t
<b>A3</b>	<b>Utvikling av applikasjonsserver</b>	<b>208t</b>
A3.1	API-ende punkt - utarbeide API-spesifikasjon	4t
A3.2	JPA datastruktur - går dette under	4t
A3.3	Utvikling	200t
<b>A4</b>	<b>Utvikling av mobilapplikasjon</b>	<b>602t</b>
A4.1	API kommunikasjon	100t
A4.2	Hovedside med Tilgjengelige bestillinger/leveringer	150t
A4.3	Transaksjon skjerm med mulighet for kommunikasjon med motpart	100t
A4.4	Transaksjonsskjerm med mulighet for betaling og rating ved fullført kjøp	100t
A4.5	Mine sider kjøper	75t
A4.6	Mine sider selger	75t
<b>A5</b>	<b>Testing</b>	<b>180t</b>
A5.1	Ferdigstilling av produkt	180t
<b>A6</b>	<b>Hovedoppgave – rapport</b>	<b>N/A</b>
A6.1	Utarbeide rapport	N/A
A6.2	Ferdigstille rapport	270t

## 5.7 Framdriftsplan – styring av prosjektet

### 5.7.1 Hovedplan

#### A0 innhenting av informasjon

Ved siden av forprosjektet trenger gruppemedlemmene å lære seg programvareutviklingssettet Flutter. Videre i prosjektet blir gruppen nødt til å hente informasjon om betalingsløsninger som kan brukes i applikasjonen.

#### A1 Prosjektering av applikasjon og system

Det skal lages en backlogg med arbeidsoppgaver. Det skal også lages wireframes for å få en idé av designet til den ferdige applikasjonen.

#### A2 Sette opp driftsmiljø for applikasjons server

For å kunne kjøre applikasjonsserveren kreves et driftsmiljø som kjøres på en vertsmaskin. I dette tilfellet er dette basert på docker-compose, et kontainerbasert kjøremiljø, der hver applikasjon kjøres isolert fra vertsmaskinen.

#### A3 Utvikling av applikasjonsserver

Det må bestemmes hvilke API-endepunkter som applikasjonsserveren trenger og den underliggende datastrukturen. Skrive kode.

#### **A4 Utvikling av mobilapplikasjon**

Selve mobilapplikasjonen skal lages i henhold til kravspesifikasjonen levert av oppdragsgiver. Skrive kode og bygging mot plattformer.

#### **A5 Testing**

Testing gjøres gjennom hele prosjektet, men mot slutten skal hele applikasjonen testes slik at gruppen er sikker på at den fungerer optimalt.

#### **A6 Hovedoppgaverapport**

Ved siden av utviklingen av applikasjonen skal gruppen også skrive hovedoppgaven.

#### **Milepæler**

Siden gruppen skal jobbe med den agile utviklingsmetoden Scrum, settes det per nå ingen milepæler. Det tas hensyn til at Scrum backloggen legges opp slik at prosjektet blir ferdig i tide. Hvilke oppgaver som prioriteres blir avgjort etter behov.

### **5.7.2 Styringshjelpemidler**

- Overleaf – skriving av rapporter
- Confluence - For utforming og lagring av møtereferat og –agenda, og andre hjelpemidler
- Jira – For utforming av Scrum sprinter og for å holde oversikt over arbeidsoppgavene underveis.
- Git – versjonskontroll av kildekode
- GitHub – Plattform for drift av git-repository
- GitHub Pull Requests – Sporing og revisjon av innsendt kode
- Discord – VoIP- og chatteprogram for kommunikasjon
- Zoom (NTNU) – Plattform for videomøter med styringsgruppen

### **5.7.3 Utviklingshjelpemidler**

- JetBrains IntelliJ - IDE for Java-programmering
- Docker Compose - kontainerbasert-miljø for distribuerte applikasjoner
- Windows Subsystem for Linux (WSL) - Miljø for å kjøre fullverdig Linux rett i Windows
- Android Studio – For emulering av Android som brukes til testing
- Xcode – Kompilering og pakking av iOS-applikasjon
- PostgreSQL – DBMS (Database Management System) (?)

### **5.7.4 Intern kontroll – evaluering**

Siden gruppen valgte å bruke Scrum som arbeidsmetodikk vil det holdes "sprint retrospective"-møter etter hver 2-ukers periode. Her vil fåreløping fremgang evalueres og eventuelle endringer implementert i planen fremover. I forbindelse med Scrum blir

også gjennomført er stand-up møte hver dag hvor foreløpig progresjon blir gjennomgått. Gruppen anser et mål som gjennomført når koden har bli sett over av de andre gruppemedlemmene, gjennom pull request systemet på GitHub, og koden er "merget" inn i develop branchen.

### **5.8 Beslutninger – beslutningsprosess**

Oppdragsgiver hadde en detaljert oppgavebeskrivelse som i stor grad har gitt gruppen et inntrykk av hvordan oppdragsgiver ønsker prosjektet. I forprosjekt fasen hadde ikke gruppen mulighet til å møte oppdragsgiver, siden han var ut på sjøen.

Hvis det oppstår et behov for å ta noen store beslutninger, vil gruppen diskutere dette og prøve å bli enig. Hvis enighet ikke blir nådd vil flertallet velge, siden gruppen har 3 medlemmer vil dette løse de fleste problemer. Hvis den nedstemte part ønsker det, kan han "anke" til veileder som kan si sin mening. Gruppen stemmer så på nytt og resultatet blir stående.

## **6 DOKUMENTASJON**

### **6.1 Rapporter og tekniske dokumenter**

For å forenkle videre utvikling og drift av tjenesten, er det nødvendig med dokumentasjon som sikrer at utenforstående enkelt kan få innsikt i oppbyggingen av systemet.

Følgende dokumenter vil bli utarbeidet:

- Kravspesifikasjon
- UI-mockup
- Databasemodell (E/R-diagram)
- UML-diagram
- API-spesifikasjon

Hvorvidt teknisk dokumentasjon kan publiseres offentlig eller ikke, blir i stor grad avgjort om kildekoden og funksjonaliteten i systemet er offentlig eller lukket. Dette vil bli tatt stilling til så snart gruppen har mulighet til å møte oppdragsgiver.

Underveis i utviklingsprosessen vil også følgende rapporter bli utarbeidet

- Sprint-retrospective rapporter som en del av Scrum
- Fremdriftsrapporter til styringsgruppen (hvis ikke møte)
- Forprosjektrapport (dette dokumentet)
- Bacheloroppgave

Dokumentasjon og rapporter er foreløpig lagret på OneDrive via NTNU, i Overleaf (via git-repository), og på NTNU Ålesund sin Jira-plattform. Grunnet uavklarte omstendigheter vedrørende offentlighet, er det en viss mulighet for at lokasjon for lagring senere endres.

Alle i prosjektgruppen er ansvarlig for at dokumentasjon er oppdatert, og lagret på avtalt lokasjon. Dokumenter som ikke skal endres i etterkant, eksporteres til et skrivebeskyttet format.

## **7 PLANLAGTE MØTER OG RAPPORTER**

### **7.1 Møter**

#### **7.1.1 Møter med styringsgruppen**

Møter mot veileder er planlagt i etterkant av hver utviklingssprint, normalt annenhver fredag, kl. 10.00 om ikke annet er spesifisert. Møter mot oppdragsgiver planlegges også i etterkant av hver sprint, men holdes etter avtale med oppdragsgiver. Dette kan være fysiske møter, eller digitale videomøter. Hvis det ikke skulle være passende for oppdragsgiver å holde møte, vil prosjektets fremgang bli rapportert over e-post.

#### **7.1.2 Prosjektmøter**

Gruppen følger den vanlige møtemetodikken brukt i Scrum. Dette betyr at gruppen møtes hver dag for å ha et stand-up møte for å kunne kjapt orientere seg på andre gruppemedlemmers progresjon og eventuelt re-distribuere noen av oppgavene. Gruppen vil også ha et mer omfattende møte etter hver 2-ukers bulk, hvor fremdriften i den avsluttede sprint-periode blir evaluert. Her vil det også gjennomgås hva som gikk bra/dårlig og potensielle forbedringer til neste sprint-periode. Når evalueringen av avsluttede periode er ferdig, vil oppgaver til neste sprint-periode bli delegert og den nye sprinten starter.

### **7.2 Periodiske rapporter**

#### **7.2.1 Framdriftsrapporter (inkl. milepæl)**

Når hver sprint er avsluttet, skal det lages en "sprint-retrospective"-rapport som oppsummerer den forrige sprinten. Dette gjøres annenhver uke. I tillegg skal gruppens medlemmer kontinuerlig jobbe med bacheloroppgaven som i detalj diskuterer fremgangsmåte, resultater, og videre drøfting m.m.

## **8 PLANLAGT AVVIKSBEHANDLING**

Dersom prosjektet ikke går som planlagt, blir gruppen nødt til å se på nøyaktig hva som senker framdriften. Hvis innholdet i kravspesifikasjonen skaper problemer, må gruppen diskutere med oppdragsgiver om den kan endres.

Prosjektleder har ansvar for å følge med at gruppen ligger an til å bli ferdig med prosjektmålene i tide.

## **9 UTSTYRSBEHOV/FORUTSETNINGER FOR GJENNOMFØRING**

Da formålet med prosjektet innebærer å utvikle av en applikasjon som tar for seg handel og transaksjoner av penger, er en betalingsleverandør nødvendig for å i hele tatt

gjennomføre prosjektet. Inntrykket her er at dette innebærer avtaler som må inngås mellom oppdragsgiver, bank og betalingsleverandør. Prosjektgruppen har ikke kunnskap på dette feltet i dag, så dette må undersøkes og avklares nærmere med oppdragsgiver.

For at tjenesten skal være tilgjengelig for brukerne, er det behov for å bygge mobilapplikasjonene på plattformene for Android og iOS. Som tidligere nevnt er iOS en lukket plattform, og for å kunne bygge mot iOS kreves en Macintosh-maskin. Da ingen av gruppemedlemmene eier dette i dag, må en skaffe tilgang til dette på en eller annen måte.

En annen problemstilling som gjelder, er distribuerings av applikasjon mot brukernes enheter. På Android-kommer vi forbi dette problemet, ved at programpakker (APK-filer) kan lastes ned og installeres av brukerne selv. På iOS finnes ikke denne muligheten, og applikasjoner kan bare distribueres via App Store eller ved innmelding i ett av Apple sine testprogrammer (TestFlight - Apple Developer, 2021). Detaljer og kostnader rundt dette må undersøkes nærmere.

Om det er mulig, kommer vi til å søke skolen for støtte om ressursene ovenfor. Kanskje en her også kan samarbeide med andre bachelorgrupper som også har tilsvarende problem, for å fordele kostnad og gjenbruke løsninger. Uansett er dette utfordringer som må bli tatt opp med styringsgruppen.

## 10 REFERANSER

1. 2019. *Xcode EULA*. [online] Available at: <<https://www.apple.com/legal/sla/docs/xcode.pdf>> [Accessed 18 January 2021].
2. Schwaber, K. and Sutherland, J., 2020. *Scrum Guide | Scrum Guides*. [online] Scrumguides.org. Available at: <<https://www.scrumguides.org/scrum-guide.html>> [Accessed 19 January 2021].
3. En.wikipedia.org. 2021. *Kanban (Development)*. [online] Available at: <[https://en.wikipedia.org/wiki/Kanban\\_\(development\)](https://en.wikipedia.org/wiki/Kanban_(development))> [Accessed 19 January 2021].
4. <https://createlly.com/>
5. Apple Developer. 2021. *Testflight - Apple Developer*. [online] Available at: <<https://developer.apple.com/testflight/>> [Accessed 21 January 2021].