

# Rapportmal for mappeoppgave

Navn Navnesen og Navn Navnesen

19. februar 2024

## **Sammendrag**

[En kort oppsummering over hva denne rapporten handler om, de viktigste resultatene og konklusjonene. De fleste vil lese dette sammendraget for å vurdere om det er interessant nok til å sette seg ned og lese hele rapporten. Bruk derfor god tid på å formulere sammendraget på en slik måte at du trigger leseren sin interesse til å lese hele rapporten. Sammendraget skal kunne leses isolert, uten at man behøver å ha kjennskap til rapporten forøvrig.]

## Innhold

<b>1 Innledning</b>	<b>3</b>
1.1 Bakgrunn . . . . .	3
1.2 Kravspesifikasjon . . . . .	3
1.3 Avgrensninger . . . . .	3
1.4 Begreper . . . . .	3
<b>2 Teori</b>	<b>3</b>
<b>3 Metode</b>	<b>5</b>
3.1 Utviklingsprosess . . . . .	5
3.2 Verktøy . . . . .	5
3.3 Bruk av KI-verktøy . . . . .	5
<b>4 Resultat</b>	<b>5</b>
4.1 Teknisk design . . . . .	5
4.2 Implementasjon . . . . .	6
4.3 Testing . . . . .	7
4.4 Utrulling til sluttbruker . . . . .	7
<b>5 Drøfting</b>	<b>7</b>
5.1 Løsning . . . . .	7
5.2 Prosess . . . . .	7
5.3 KI-verktøy . . . . .	7
<b>6 Konklusjon</b>	<b>8</b>
<b>A Vedlegg</b>	<b>8</b>
<b>B KI-deklarasjon</b>	<b>8</b>

## Figurer

1 Use-case-diagram. . . . .	4
2 Klassediagram. . . . .	6

## Kodelister

1 Java-kode fra fil . . . . .	6
-------------------------------	---

## 1 Innledning

### 1.1 Bakgrunn

[Dette er første kapitlet i den faglige rapporten. Det bør behandle bakgrunnen for oppgaven, eventuell oppdragsgiver, oppsummering av problemstillingen og/eller oppgaven som skal løses.]

### 1.2 Kravspesifikasjon

[Her beskriver du både de funksjonelle kravene og de ikke-funksjonelle kravene til løsningen du skal utvikle. Dersom det er gitt en kravspesifikasjon vil du kunne hente det meste av informasjon fra denne. Husk at du her *ikke* skal beskrive noen av de valg du har gjort i prosjektet, eller det du konkret har utviklet i prosjektet. Bruk her gjerne UML-diagrammer som use-case, aktivitetsdiagram osv for å beskrive krav til funksjonalitet, uten å dra inn hvordan du/dere har løst det. Se f.eks. figur 1]. Når du senere skriver drøfting og konklusjon, må du henvise tilbake til dette kapitlet og svare på om løsningen du har levert svarer på kravspesifikasjonen.

### 1.3 Avgrensninger

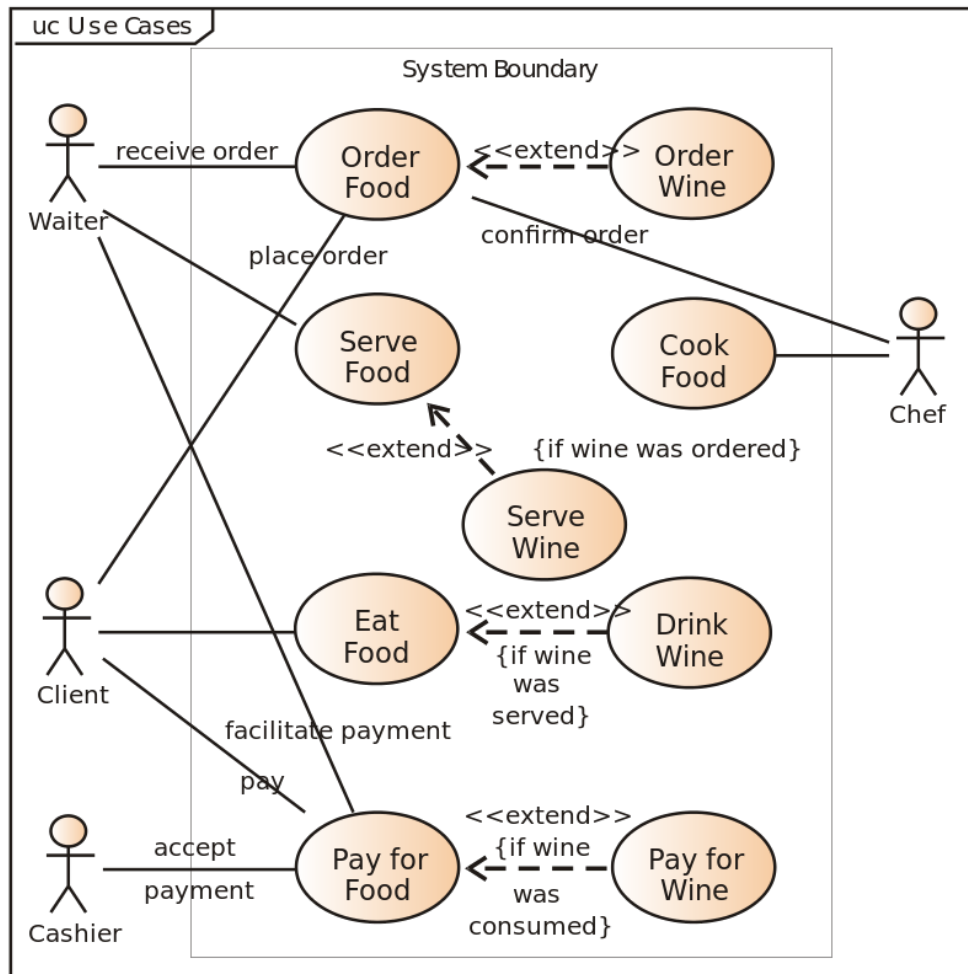
[Er det gitt noen avgrensninger/begrensninger i oppgaven? Beskriv i så fall disse her. Avgrensninger kan f.eks. være teknologier dere må bruke, el.l. Dersom prosjektet ikke har noen avgrensninger, kan dette kapitlet utelates.]

### 1.4 Begreper

[Når man utvikler programvare for en kunde, er det viktig å etablere en felles forståelse for begreper/terminologi/ord som benyttes av kunden. Det er derfor svært vanlig å lage en «ordliste» og/eller en «domenemodell». Denne ordlisten er også et svært godt utgangspunkt for å finne frem til hvilke mulige klasser det kan være aktuelt å implementere i løsningen når denne skal utvikles. Bruk tid på denne slik at du har en god forståelse for begrepene. Her kan man også bruke klasse-diagrammer for å illustrere hvordan begreper henger sammen. Klasser i dette diagrammet er ikke nødvendigvis klassene du har valgt å implementere i prosjektet. Kanskje er det også på sin plass med en henvisning til læreboken [1] eller andre kilder.]

## 2 Teori

[Her beskriver du de teoriene og beste praksiser som dere har benyttet ved utvikling av en programvare, som kobling, kohesjon osv. Beskriv hvilke teorier, men ikke hvordan dere har brukt teoriene. Det hører hjemme i resultat og diskusjon. Her må dere referere til kilder (som legges til i kapitlet Referanser), f.eks. slik [1]. Når du senere skal skrive drøfting og konklusjon, må du henvise tilbake til dette



**Figure 1:** Use-case-diagram.

kapittelet og svare på om løsningen du har levert er løst i henhold til disse teoriene/beste praksisene. Unngå å beskrive om grunnleggende teorier som «hva er OOP», «Programmeringsspråket Java» osv. Skriv kun teoripunkter som du faktisk har benyttet i ditt design, og som du faktisk diskuterer i diskusjonen.]

## 3 Metode

[I dette kapittelet skal dere beskrive hva som skal til for å kunne reproducere resultatet dere har fått. I programvareutvikling koker det som oftest ned til prosess/metodikk, og verktøy.]

### 3.1 Utviklingsprosess

I dette kapittelet skal du fortelle hvilken prosess du/dere planla å følge. Den skal dekke prosessmodellen, hvorfor den ble valgt og hvordan den ble implementert. Har dere jobbet i gruppe, så si noe om hvordan dere planla å organisere arbeidet (hvor ofte planla dere å møtes å jobbe?). Hvordan planlagt prosess faktisk fungerte og hvilke endringer dere eventuelt gjorde skal beskrives i resultat-kapittelet og drøftes under Drøfting. I en typisk mappe-oppgave der dere har jobbet gjennom flere stadier/deler av prosjektet, og fått tilbud om tilbakemelding, bør dette beskrives som en del av prosessen (altså at dere jobbet i f.eks. 3 deler med muntlig tilbakemelding etter hver del). Vis gjerne figurer med skjermdumper av f.eks. kanban-tavler eller andre prosessverktøy som er benyttet.]

### 3.2 Verktøy

[Beskriv verktøy du/dere har benyttet for å løse prosjektet. Lag gjerne en tabell med navn på verktøy, versjon og hva verktøyet er benyttet til. Få med samtlige verktøy (IDE, versjonskontroll, prosjektplanlegging/gjennomføring osv)]

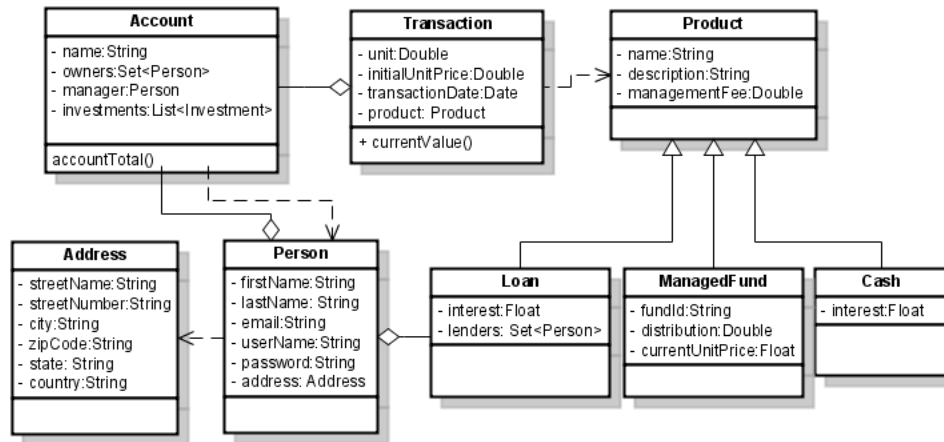
### 3.3 Bruk av KI-verktøy

[Som et vedlegg til denne rapporten *skal* dere fylle ut en KI-deklarasjon. Du/dere kan selv velge om dere vil beskrive hva og hvordan dere har benyttet KI-verktøy enten i skjemaet, eller her i dette kapittelet. Dersom dere velger å beskrive det her, så henviser du/dere til dette kapittelet i KI-deklarasjons-skjemaet, vedlegg B. Beskriv også her hvorfor du/dere valgte å benytte KI-verktøy. Hva ønsket dere å oppnå?]

## 4 Resultat

### 4.1 Teknisk design

[Kapittelet om teknisk design beskriver det store bildet av valgt løsning. For et programvareutviklingsprosjekt vil dette vanligvis inneholde systemarkitekturen



Figur 2: Klassediagram.

(klient-server, sky, databaser, tjenester, desktop-applikasjon osv.); både hvordan det ble løst, og, enda viktigere, hvorfor denne arkitektur ble valgt.]

## 4.2 Implementasjon

[Her beskriver du de tekniske detaljene til løsningen du har endt opp med. Hvilke eksterne biblioteker og rammeverk, APIer osv. har du/dere benyttet?

Vis med klassediagrammer hvordan klassene i løsningen din henger sammen. Husk at du kan vise flere klassediagrammer som f.eks. i figur 2 for å vise ulike sider ved løsningen (kun klasser som benyttes til brukerinteraksjon f.eks., eller kun klasser som utgjør forretningslogikken). Bruk aktivitetsdiagramm(er) for å beskrive logikken/flyten i løsningen din. Fint også om du kan vise hvordan objekter av klassene dine samhandler for å løse de mest sentrale oppgavene i løsningen, da i form av sekvensdiagramm(er). Klassene dine bør beskrives i forhold til hvilken rolle og ansvar de har, men det er ikke nødvendig å beskrive hver enkelt metode eller felt til klassen. De står uansett dokumentert i Javadoc'en din.

Hvis dere har noen fikse kode-detalljer dere vil vise frem, kan det også gjøres som vist i kodeliste 1.]

Kodeliste 1: Java-kode fra fil

```

1  /**
2   * Hello World
3   */
4  class HelloWorld {
5      public static void main(String[] args) {
6          System.out.println("Hello, World!");
7      }
8  }
  
```

### 4.3 Testing

[Beskriv hvordan løsningen din har blitt testet, både i form av enhetstesting, men også brukertesting. Hvor mange brukere har testet løsningen? Hvordan ble brukertesting gjennomført? Vis også eventuelt resultatene fra brukertesting her. (svar på spørreskjema el.l.)

Enhetstesting: si noe om hvilke klasser du/dere har valgt å skrive tester for, og hva strategien som dere/du har lagt opp til for å sikre best mulig testet kode.]

### 4.4 Utrulling til sluttbruker

[Her beskriver du hvordan programvaren din gjøres tilgjengelig for sluttbruker. Rulles den ut å en web-server? I så fall hvordan? Lages det en desktop-applikasjon som bruker kan dobbeltklikke på for å starte? Eller kjøres applikasjonen fra Maven (mvn javafx:run) Osv.]

## 5 Drøfting

### 5.1 Løsning

[Her oppsummerer du/dere oppgaven. Hvor langt kom du/dere (resultat)? Hva fikk du/dere ikke gjort i forhold til oppgaveteksten? Hva var de store utfordringene/problemene du/dere møtte, etc. Spesielt viktig er det å drøfte din egen løsning i forhold til det du har lært om gode prinsipper for design av programvare (robust kode, kodelstil, designprinsipper osv) som beskrevet i teori-kapittelet.

I en godt skrevet rapport, er det ingen teorier som beskrives under teori-kapittelet som ikke drøftes under drøfting-kapittelet. Husk å være konkret: Det holder ikke å skrive «Jeg/vi har designet en løsning som er i trå med prinsippene om coupling og cohesion». Du/dere må «bevise» hvorfor dere kan påstå dette. Altså hente eksempler fra egen kode som underbygger teoriene om god design: «I klassen ..... har vi valgt å returnere .... fra metoden.... Dette bidrar til lav kobling....» Her bør man også gjøre seg tanker rundt kvaliteten av det arbeidet som er nedlagt. Er de kildene du/dere bruker pålitelige, er det sprik mellom forskjellige kilder (og i så fall hvorfor), er det andre forhold som kan være med å gjøre noen av de vurderinger og valg du/dere har gjort usikre?]

### 5.2 Prosess

[Fulgte dere prosessen som dere planla (og beskrev under kapittelet «Metode»)? Var det lurt, eller ikke? Hva fungerte bra hva fungerte mindre bra? Hva ville du/dere ha gjort annerledes neste gang?]

### 5.3 KI-verktøy

[Dersom du/dere benyttet KI-verktøy i denne oppgaven, drøft kort erfaringene dine/deres. Hva var KI-verktøyene nyttige for å løse? Hvilke svakheter oppdaget

du/dere? Har du/dere ikke benyttet KI-verktøy dropper du dette kapittelet.]

## 6 Konklusjon

[Overbevisninger /erfaring som en er kommet fram til på grunnlag av det presenterte materialet.

- Fikk du realisert hele problemstillingen fra kapittel «1.2 Kravspesifikasjon»?
- Hva ville du ha gjort annerledes dersom du kunne begynne på nytt?
- Hva slags begrensninger kan en forvente når en bruker løsningen?
- Hva skal tas opp i fremtidige arbeid dersom du eller noen andre ville ha tatt utvikling videre?]

## Referanser

- [1] D. J. Barnes og M. Kölling, *Objects First with Java: A practical introduction using BlueJ*, Sixth. Pearson/Prentice Hall, 2017.

## A Vedlegg

[Materiell som er utarbeidet eller innsamlet i tilknytning til rapporten, men som det ikke er naturlig eller hensiktsmessig å ta inn i hoveddelen, som feks brukerveiledning, skal tas inn som vedlegg. Vedleggene skal være nummererte og ha en overskrift. Har du/dere ingen vedlegg, så droppes dette kapittelet.]

## B KI-deklarasjon

[KI-deklarasjonen som *skal* leveres inn utfylt, er vedlagt på neste side. Last ned Word-malen og fyll den ut, og last opp generert PDF.]



# Deklarasjon om KI-hjelpemidler

Har det i utarbeidningen av denne rapporten blitt anvendt KI-baserte hjelpemidler?

☐ Nei

☐ Ja

Hvis *ja*: spesifiser type av verktøy og bruksområde under.

---

## Tekst

☐ **Stavekontroll.** Er deler av teksten kontrollert av:  
*Grammarly, Ginger, Grammarbot, LanguageTool, ProWritingAid, Sapling, Trinkai.ai* eller lignende verktøy?

☐ **Tekstgenerering.** Er deler av teksten generert av:  
*ChatGPT, GrammarlyGO, Copy.AI, WordAi, WriteSonic, Jasper, Simplified, Rytr* eller lignende verktøy?

☐ **Skriveassistanse.** Er en eller flere av ideene eller fremgangsmåtene i oppgaven foreslått av:  
*ChatGPT, Google Bard, Bing chat, YouChat* eller lignende verktøy?

Hvis *ja* til anvendelse av et tekstverktøy - spesifiser bruken her:

---

## Kode og algoritmer

☐ **Programmeringsassistanse.** Er deler av koden/algoritmene som i) fremtrer direkte i rapporten eller ii) har blitt anvendt for produksjon av resultaterslik som figurer, tabeller eller tallverdier blitt generert av: *GitHub Copilot, CodeGPT, Google Codey/Studio Bot, Replit Ghostwriter, Amazon CodeWhisperer, GPT Engineer, ChatGPT, Google Bard* eller lignende verktøy?

Hvis *ja* til anvendelse av et programmeringsverktøy - spesifiser bruken her:

---

## Bilder og figurer

☐ **Bildegenerering.** Er ett eller flere av bildene/figurene i rapporten blitt generert av: *Midjourney, Jasper, WriteSonic, Stability AI, Dall-E* eller lignende verktøy?

Hvis *ja* til anvendelse av et bildeverktøy - spesifiser bruken her:

---

## Andre KI-verktøy

☐ **Andre KI-verktøy.** har andre typer av verktøy blitt anvendt? Hvis *ja* spesifiser bruken her:

☐ Jeg er kjent med NTNUs regelverk: *Det er ikke tillatt å generere besvarelse ved hjelp av kunstig intelligens og levere den helt eller delvis som egen besvarelse.* Jeg har derfor redegjort for all anvendelse av kunstig intelligens enten i) direkte i rapporten eller ii) i dette skjemaet