

# Sensorveiledning IDATx1003

Høst 2023

## Om emnet IDATx103 Programmering 1

### Emnebeskrivelse

(<https://www.ntnu.no/studier/emner/IDATG1003>)

Emnet er felles på tvers av campusene Gjøvik, Ålesund og Trondheim. Emnet har egen emnekode pr campus: IDATG/IDATT/IDATA 1001, vi refererer til emnet som **IDATx1003**.

IDATx1003 Programmering 1 dekker følgende læringsutbytter i OOP:

- ☐ Klasser og objekter, samhandling av objekter
- ☐ Datatyper, variabler og konstanter
- ☐ Metoder med og uten retur og parametre
- ☐ Betingelser (if- og switch uttrykk)
- ☐ Løkker (for-each, for og while), primært i forbindelse med **samlinger** (ArrayList, HashMap HashSet)
- ☐ Samlinger av objekter og klasser for å håndtere disse (ArrayList, HashMap og HashSet osv)
- ☐ Iterator-klassen som hjelp ifm å iterere over en samling
- ☐ Designprinsippene **coupling**, **cohesion** og **responsibility driven design**.
- ☐ Dokumentasjon, kodelstil og robust kodedesign (studentene har krav på seg å kjøre **CheckStyle** for å sikre god kodelstil.
- ☐ Testing (enhetstesting/JUnit), debugging og robusthet
- ☐ Enkelt tekstbasert/menybasert brukergrensesnitt.
- ☐ Grunnleggende diagrammer i UML (for eks, aktivitetsdiagram, klassediagram, sekvensdiagram, pakkediagrammer)
- ☐ Grunnleggende versjonskontroll med Gitlab/Github

Emnet dekker **ikke**:

- ☐ Maven (men prosjektet skal være satt opp i maven)
- ☐ Unntakshåndtering (exceptions)
- ☐ Arv, polymorfi, interfaces
- ☐ Filhåndtering
- ☐ Grafisk Brukergrensesnitt
- ☐ For versjonskontroll dekkes ikke greiner (branches) og merge-konflikter

### Lærebok og Verktøy/IDE

Lærebok benyttet i Ålesund og Gjøvik er «Objects First with Java» av David J. Burnes og Michael Kölling.

I Trondheim benyttes « Core Java Volume I - Fundamentals 11. edition» av Cay S. Horstmann.

Studentene bruker profesjonelle IDE'er som IntelliJ, VSCode, Eclipse, Netbeans etc. (studentene velger selv) men er pålagt å bruke Maven-oppsett som et byggesystem.

---



## Om Mappen

Mappen som er levert for sensur består av:

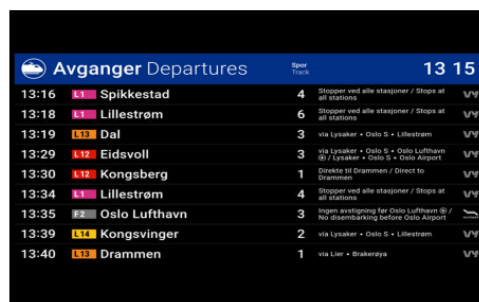
- En prosjektoppgave i programmering som kandidaten har jobbet **individuellt** med i ca 10 uker. Kandidaten har fått mulighet for **tilbakemelding muntlig** på sitt arbeid 3 ganger i løpet av de første 8 ukene, i tillegg til bistand i lab-timene ved behov. Prosjektoppgaven er lastet opp i Inspira sammen rapport.  
- *Prosjektet (kode) teller 70% av endelig karakter*
- En rapport, der kandidaten skal beskrive sin løsning, hvilke valg kandidaten har gjort under utviklingen av løsningen, og hvilke endringer (refaktoring) som er gjort. Kandidaten skal også begrunne sitt design basert på teoriene rundt **coupling**, **cohesion** og **responsibility driven design**, med konkret henvisning til eksempler fra koden.  
- *Rapporten teller 30% av endelig karakter*

Kravspesifikasjonen til prosjektet er levert ut til kandidaten i 3 deler:

- Del 1 i uke 40 - Beskriver prosjektet i sin helhet, med hva man skal utvikle i løpet av prosjektet. For Del 1 skal kandidaten fokusere på å implementere **entitetsklassen** som skal representere en **togavgang**.
- Del 2 i uke 43 - Fokuserer på **register av togavganger** og nødvendig funksjonalitet i registeret
- Del 3 i uke 45 - Med fokus på **brukerinteraksjon**. I denne delen åpnes også oppgaven opp betydelig og gir rom for at kandidaten skal foreslå egne utvidelser, egne endringer til opprinnelig kravspek osv. for å sette sitt eget personlige preg på løsningen.

## Prosjektoppgaven:

Kandidaten skal utvikle en applikasjon for et forenklet system for avvikling av togavganger (engelsk: "Train Dispatch System"). Et eksempel på et slikt system vises nedenfor.



Avganger Departures		Spør Tolk	13 15
13:16	Spikkestad	4	Stopper ved alle stasjoner / Stops at all stations
13:18	Lillestrøm	6	Stopper ved alle stasjoner / Stops at all stations
13:19	Dal	3	via Lysaker • Oslo S • Lillestrøm
13:29	Eidsvoll	3	via Lysaker • Oslo S • Oslo Lufthavn (B) / Lysaker • Oslo S • Oslo Airport
13:30	Kongsberg	1	Direkte til Drammen / Direct to Drammen
13:34	Lillestrøm	4	Stopper ved alle stasjoner / Stops at all stations
13:35	Oslo Lufthavn	3	Ingen avgang for Oslo Lufthavn (B) / No disembarking before Oslo Airport
13:39	Kongsvinger	2	via Lysaker • Oslo S • Lillestrøm
13:40	Drammen	1	via Lier • Brakerøya

Figur 1 Toginfotavle på en stasjon

Brukergrensesnittet er **tekstbasert**, og kan enten være en **meny** med ulike valg, eller implementert som **kommandoer/kommandolinje**. Full prosjektbeskrivelse gis sensorer i eget dokument.

## Om Verktøy/IDE

Vi har ikke stilt krav om ett bestemt IDE, men studentene skal sett opp prosjektet som et Maven prosjekt. Studentene får selv velge hvilket IDE de ønsker å bruke. Følgende IDE'er benyttes typisk av studentene:

- ☐ IntelliJ
- ☐ VSCode

## Karakterskala ved NTNU

Følgende karakterskala benyttes ved NTNU:

Symbol	Betegnelse	Generell, ikke fagspesifikk beskrivelse av vurderingskriterier
A	Fremragende	Fremragende prestasjon som klart utmerker seg. Kandidaten viser svært god vurderingsevne og stor grad av selvstendighet.
B	Meget god	Meget god prestasjon. Kandidaten viser meget god vurderingsevne og selvstendighet.
C	God	Jevnt god prestasjon som er tilfredsstillende på de fleste områder. Kandidaten viser god vurderingsevne og selvstendighet på de viktigste områdene.
D	Nokså god	En akseptabel prestasjon med noen vesentlige mangler. Kandidaten viser en viss grad av vurderingsevne og selvstendighet.
E	Tilstrekkelig	Prestasjonen tilfredsstiller minimumskravene, men heller ikke mer. Kandidaten viser liten vurderingsevne og selvstendighet.
F	Ikke bestått	Prestasjon som ikke tilfredsstiller de faglige minimumskravene. Kandidaten viser både manglende vurderingsevne og selvstendighet.

## Sensurskjema

Her følger sensurskjema som er lagt til grunn for sensureringen av mappene ved alle 3 campus (Trondheim, Gjøvik, Ålesund).

Ved sensurering benyttes poenger og vekting, men kun som **veiledende** (obs - ingen «eksakt matematisk vitenskap»). Det vil alltid gjøres en total helhetsvurdering av kandidatens besvarelse før endelig karakter settes.

Sensurskjema er delt inn i 3 deler:

- ☐ Del 1 – Funksjonelle krav, og brukervennlighet
  - ☐ Del 2 – Implementasjon/kode/løsning
  - ☐ Del 3 – Rapport
-

## Del 1 - Funksjonelle krav

<b>Funksjonelle krav</b>
Er de <b>funksjonelle kravene</b> i kravspesifikasjonen oppfylt (del 1 - 3) ?
<b>Brukervennlighet/UI-design</b>
Prosjektet skal leveres med et tekstbasert brukergrensesnitt. Følgende skal vurderes i forhold til brukervennlighet og design av brukergrensesnitt: <input type="checkbox"/> Fremstår brukerinteraksjonen som "brukervennlig"/intuitiv? <input type="checkbox"/> Er det greit å forstå informasjonen som presenteres? <input type="checkbox"/> Håndteres feil inntasting av bruker på en god måte?

## Del 2 - Implementasjon/Design

KRITERIUM	Ikke bestått (tilsvarer F)	Svak (tilsvarer D/E)	Middels (tilsvarer C)	Utmerket (tilsvarer B/A)
<b>Grunnleggende OOP</b>				
Kan kandidaten deklare en klasse?	Kandidaten viser betydelig manglende forståelse for hvordan deklare en klasse	Kandidaten viser noen mangler, f.eks. ved: <input type="checkbox"/> Varierende Navnekonvensjon <input type="checkbox"/> Varierende grad av Innkapsling	Kandidaten deklarerer klasser med: <input type="checkbox"/> Riktig navnekonvensjon for klasser, variabler og pakker <input type="checkbox"/> Nødvendige konstruktører <input type="checkbox"/> God innkapsling	Kandidaten har i tillegg vist: <input type="checkbox"/> Robust(e) konstruktør(er)
Datatyper og variabler		Kandidaten viser noen manglende forståelse for datatyper: <input type="checkbox"/> Velger å bruke <b>String</b> der det typisk burde vært brukt <b>int</b> eller <b>double</b> eller en <b>klasse-type</b> (spesielt i retur fra metoder)	Kandidaten benytter fornuftige datatyper til felt- og variabler: <input type="checkbox"/> Riktige datatyper i forhold til hva variablene representerer.	Kandidaten har: <input type="checkbox"/> Hensiktsmessig bruk av <b>final/static</b> <input type="checkbox"/> Har opprettet egne datatyper i form av klasser der dette er fornuftig.
Get- og set-metoder		Kandidaten viser noe manglende forståelse for fornuftig bruk av get- og set metoder: <input type="checkbox"/> Har implementert get- og public set metoder for samtlige felt helt ukritisk. Ingen vurdering av hvilke set-metoder som <b>det er naturlig</b> gjøres tilgjengelig (public) <input type="checkbox"/> Har ikke, eller i svært liten grad validert parametre i set-metodene <input type="checkbox"/> Benytter ikke seg av set-metoder i konstruktør.	Kandidaten har en fornuftig bruk av set- og get metoder: <input type="checkbox"/> En get-metode pr felt det er naturlig å gjøre tilgjengelig <input type="checkbox"/> Kun public set-metoder for felt det er naturlig at skal endres etter at objektet er opprettet. <input type="checkbox"/> Noen grad av validering av parametre.	Kandidaten har i tillegg: <input type="checkbox"/> Set-metoder for <b>samtlig</b> e felt i klassen, <b>MEN</b> kun et fåtall er satt til <b>public</b> <input type="checkbox"/> Set-metoder der det <b>alltid</b> utføres validering av parameter <input type="checkbox"/> Aktiv bruk av set-metoder i konstruktør(ene) for å redusere/forhindre duplisering av kode.

KRITERIUM	Ikke bestått (tilsvarer F)	Svak (tilsvarer D/E)	Middels (tilsvarer C)	Utmerket (tilsvarer B/A)
Metoder i klasser		Kandidaten viser manglende forståelse for deklarasjon av metoder: <ul style="list-style-type: none"> <li><input type="checkbox"/> Navnekonvensjon</li> <li><input type="checkbox"/> Metoder returnerer feil/ufornuftig datatype (typisk String der det burde vært returnert int eller double osv.</li> <li><input type="checkbox"/> Misbruk av <b>toString()</b> - benyttes for å lage en streng som skal presenteres sluttbruker.</li> </ul>	Kandidaten viser god forståelse for deklarasjon av metoder i en klasse: <ul style="list-style-type: none"> <li><input type="checkbox"/> Samtlige metoder starter med liten bokstav</li> <li><input type="checkbox"/> Metoder returnerer <b>void</b> der dette er naturlig.</li> <li><input type="checkbox"/> Metoder returnerer fornuftige datatyper der det er fornuftig</li> </ul>	Kandidaten har i tillegg: <ul style="list-style-type: none"> <li><input type="checkbox"/> I metoder som tar parameter valideres parameter(ne) før øvrig kode i metoden ("guard condition")</li> </ul>
<b>Samlinger</b> <b>(ArrayList, HashMap osv)</b>				
Grunnleggende forståelse av <b>samlinger</b>		Kandidaten har: <ul style="list-style-type: none"> <li><input type="checkbox"/> Ikke egen klasse som representerer vareregister, men bruker isteden ArrayList (el.l.) direkte i Main-klassen el.l.</li> <li><input type="checkbox"/> Har brukt <b>primitiv array</b> istedenfor en av klassene i <b>collection-biblioteket</b>.</li> </ul>	Kandidaten viser grunnleggende forståelse for <b>samlinger</b> : <ul style="list-style-type: none"> <li><input type="checkbox"/> Bruker riktig samling for å håndtere register.</li> <li><input type="checkbox"/> Benytter innebygde metoder i samling-klassen der det er mulig.</li> </ul>	Kandidaten har i tillegg: <ul style="list-style-type: none"> <li><input type="checkbox"/> Benyttet avanserte samlinger for håndtering av register</li> <li><input type="checkbox"/> Eventuelt: kandidaten beskriver at hen har vurdert bedre løsninger og konkludert med annen løsning.</li> </ul>
<b>Løkker</b>				



KRITERIUM	Ikke bestått (tilsvarer F)	Svak (tilsvarer D/E)	Middels (tilsvarer C)	Utmerket (tilsvarer B/A)
For-each og while-løkker		Bruker ikke løkker iht anbefalingene: <ul style="list-style-type: none"> <li><input type="checkbox"/> Avbryter for/for-each løkke med <b>break</b> eller <b>return</b> i løkken</li> <li><input type="checkbox"/> I while-løkker: samler ikke alle betingelser for å iterere i selve <b>while-statmenetet</b> (f.eks. bruker return og/eller break inne i while løkka.</li> <li><input type="checkbox"/> for/while-løkke med index istedenfor den bedre løsningen med Iterator</li> </ul>	Kandidaten viser grunnleggende forståelse av løkker: <ul style="list-style-type: none"> <li><input type="checkbox"/> for-each for samlinger - når <b>hele</b> samlingen skal itereres over</li> <li><input type="checkbox"/> while-løkke, når det <b>søkes i en samling</b></li> </ul>	Kandidaten har i tillegg: <ul style="list-style-type: none"> <li><input type="checkbox"/> Konsekvent bruk av kun for-each/for løkker når man skal iterere over samtlige objekter, og ikke ellers.</li> <li><input type="checkbox"/> Bruker <b>while med iterator</b></li> <li><input type="checkbox"/> Kandidaten bruker <b>streams</b> og <b>filters</b> i stedenfor løkker. Må da gå ut ifra at kandidaten også behersker for- do while-løkke selv om kandidaten ikke bruker dette i sin løsning.</li> </ul>
<b>Kodekvalitet/Design</b>				
Er koden godt dokumentert iht JavaDoc-standard?	Ingen dokumentasjon	Kandidaten har: <ul style="list-style-type: none"> <li><input type="checkbox"/> Sporadisk dokumentasjon av klassene</li> <li><input type="checkbox"/> Sporadisk dokumentasjon av metoder</li> <li><input type="checkbox"/> Dokumentasjon av typen "This metode...", eller dok som beskriver hvordan og ikke hva</li> <li><input type="checkbox"/> Mangelfull/fraværende bruk av "@param", "@return" osv</li> </ul>	Kandidaten viser grunnleggende forståelse for dokumentasjon ved at: <ul style="list-style-type: none"> <li><input type="checkbox"/> Samtlige klasser er dokumentert med god beskrivelse av rolle/ansvar.</li> <li><input type="checkbox"/> De fleste metoder som er public er godt dokumentert.</li> <li><input type="checkbox"/> Grei "ordlyd" i dokumentasjonen (Ikke "This method...", "Gets the.." osv)</li> <li><input type="checkbox"/> Gjennomgående bruk av "@param" og "@return" osv</li> </ul>	Kandidaten viser i tillegg: <ul style="list-style-type: none"> <li><input type="checkbox"/> Gjennomført høy kvalitet i dokumentasjon</li> <li><input type="checkbox"/> God formulering som klart dokumenterer klasser og metoder</li> <li><input type="checkbox"/> "@param", "@return" konsekvent brukt</li> <li><input type="checkbox"/> Innslag av HTML-formatering for bedre lesbarhet</li> <li><input type="checkbox"/> CheckStyle med Google-styles rapporterer 0 eller minimalt med feil</li> </ul>
Er koden robust?	Ingen verifisering	Kandidaten har: <ul style="list-style-type: none"> <li><input type="checkbox"/> Sporadisk eller ingen sjekk av parameterverdier</li> </ul>	Kandidaten viser grunnleggende forståelse av robust kode ved: <ul style="list-style-type: none"> <li><input type="checkbox"/> Gjennomgående grei verifisering av parameterverdier.</li> <li><input type="checkbox"/> Håndterer ugyldige verdier på en god måte.</li> </ul>	Kandidaten har i tillegg: <ul style="list-style-type: none"> <li><input type="checkbox"/> Verifiserer samtlige parametre.</li> <li><input type="checkbox"/> Har fornuftig håndtering av situasjoner der parameter har ugyldig verdi</li> <li><input type="checkbox"/> Dersom kandidaten benytter Exceptions er det OK, men ikke poenggivene.</li> </ul>

KRITERIUM	Ikke bestått (tilsvarer F)	Svak (tilsvarer D/E)	Middels (tilsvarer C)	Utmerket (tilsvarer B/A)
Har variabler, metoder og klasser beskrivende navn?		Kandidaten har: <ul style="list-style-type: none"> <li><input type="checkbox"/> Svært mye bruk av variabelnavn som ikke gjenspeiler data de representerer/holder.</li> <li><input type="checkbox"/> Navn på metoder gjenspeiler i ingen eller liten grad <b>hva</b> metoden gjør.</li> </ul>	Kandidaten viser grunnleggende forståelse: <ul style="list-style-type: none"> <li><input type="checkbox"/> Gjennomgående gode variabelnavn som gjenspeiler data de representerer/holder.</li> <li><input type="checkbox"/> Gjennomgående gode navn på metoder som gjenspeiler <b>hva</b> metoden gjør.</li> <li><input type="checkbox"/> Klassenavn beskriver godt rollen/ansvaret til klassen</li> </ul>	Kandidaten har i tillegg: <ul style="list-style-type: none"> <li><input type="checkbox"/> Svært gode variabelnavn som gjenspeiler data de representerer/holder.</li> <li><input type="checkbox"/> Svært gode navn på metoder som gjenspeiler <b>hva</b> metoden gjør.</li> <li><input type="checkbox"/> Alle klassenavn beskriver godt rollen/ansvaret til klassen</li> </ul>
Har koden god struktur, med løse koblinger og høy kohesjon?		Kandidaten viser manglende forståelse: <ul style="list-style-type: none"> <li><input type="checkbox"/> Det er tett kobling mellom klasser som ikke burde hatt kobling (high coupling)</li> <li><input type="checkbox"/> Mange metoder som utfører flere oppgaver (low cohesion)</li> <li><input type="checkbox"/> Mange klasser med flere roller/ansvar</li> </ul>	Kandidaten viser grunnleggende forståelse: <ul style="list-style-type: none"> <li><input type="checkbox"/> Generelt god løs kobling mellom klasser som ikke burde hatt kobling (low coupling)</li> <li><input type="checkbox"/> Jevnt over de fleste metoder utfører en enkelt oppgave (high cohesion). Der en metode må utføre flere del-oppgaver, benyttes <b>deligering</b> til sub-metoder.</li> <li><input type="checkbox"/> De fleste klasser har <b>en</b> klart definert rolle/ansvar</li> </ul>	Kandidaten har i tillegg: <ul style="list-style-type: none"> <li><input type="checkbox"/> Gjennomgående svært god løs kobling mellom klasser som ikke burde hatt kobling (low coupling)</li> <li><input type="checkbox"/> Samtlige metoder utfører en enkelt oppgave (high cohesion). Der en metode må utføre flere del-oppgaver, benyttes <b>deligering</b> til sub-metoder.</li> <li><input type="checkbox"/> Samtlige klasser har <b>en</b> klart definert rolle/ansvar</li> </ul>
<b>Versjonskontroll og enhetstesting</b>				
<b>Enhetstesting</b>	Har ingen enhetstester	<ul style="list-style-type: none"> <li><input type="checkbox"/> Har noen enhetstester/tester noen få av klassene</li> <li><input type="checkbox"/> Mangler negative tester/har ikke forstått "negativ test"</li> <li><input type="checkbox"/> Skriver tester som medfører at tester må kjøres i en bestemt rekkefølge</li> <li><input type="checkbox"/> Ikke gode beskrivende navn på testene</li> </ul>	<ul style="list-style-type: none"> <li><input type="checkbox"/> Har gode beskrivende navn på testene</li> <li><input type="checkbox"/> Følger prinsippet om Arrange-Act-Assert</li> <li><input type="checkbox"/> Har enhetstester for de grunnleggende klassene</li> <li><input type="checkbox"/> Har helt greie negative tester (viser at kandidaten har forstått hovedpoenget med positive/negative tester)</li> </ul>	<ul style="list-style-type: none"> <li><input type="checkbox"/> Har også gode negative tester for alle test-klassene</li> <li><input type="checkbox"/> Tester Exceptions</li> <li><input type="checkbox"/> Benytter seg av "@BeforeAll", "@BeforeEach" osv - der det er hensiktsmessig</li> <li><input type="checkbox"/> Har svært høy "test coverage" for aktuelle klasser</li> <li><input type="checkbox"/> Har svært gode negative tester</li> </ul>
<b>Versjonskontroll</b>				

KRITERIUM	Ikke bestått (tilsvarer F)	Svak (tilsvarer D/E)	Middels (tilsvarer C)	Utmerket (tilsvarer B/A)
Lokalt og sentralt repo, commits	Har ikke benyttet versjonskontroll	<input type="checkbox"/> Prosjektet er lagt til versjonskontroll lokalt <input type="checkbox"/> Sporadiske innsjekkinger (commits) - mye som er endret mellom hver commit <input type="checkbox"/> Mangelfulle commit-meldinger (enten tomme, eller ikke beskrivende for endringene som er utført) <input type="checkbox"/> Ikke sentralt repo	<input type="checkbox"/> Prosjektet har sentralt repo (GitHub/GitLab) <input type="checkbox"/> Fornuftig jevnlig innsjekking (commit) av endringer <input type="checkbox"/> Gode commit-meldinger som beskriver kort hvilke endringer som er gjort/hvilke problem som er løst	<input type="checkbox"/> Svært gode commits og svært gode commit-meldinger. <input type="checkbox"/> Har benyttet tags for å merke versjoner
<b>Totalintrykk</b>				
Her vurderes totalintrykket til kandidaten. Dette kriteriet kan benyttes for vurdering som ikke fanges opp av de øvrige kriteriene.		Kandidaten presterer under gjennomsnitt. Løsningen er unødvendig tungvint implementert, eller er mangelfull.	En prestasjon som er midt på treet totalt sett. Kandidaten viser at han/hun har forstått de fleste læringsmål og viser god forståelse.	En fremragende løsning på alle måter. Kandidaten benytter elementer i sin løsning som er ut over det som er forventet innenfor gjeldende pensum. NB! Å benytte arv, og grafisk brukergrensesnitt inngå <b>ikke</b> i "...over forventet.."

## Del 3 - Rapport

Kandidatene fikk utlevert en mal for Word. I malen var det gitt en mengde *hjelpetekst* for å hjelpe kandidaten til å forstå hva som er forventet i de ulike kapitlene i rapporten. Malen baserer seg på strukturen ofte benyttet i vitenskapelige artikler og i Bachelor rapporter.

KRITERIUM	Ikke bestått (tilsvarer F)	Svak (tilsvarer D/E)	Middels (tilsvarer C)	Utmerket (tilsvarer B/A)
Struktur	Kaos	Elementer mangler eller er plassert på ulogiske steder	Grei struktur, fin rød tråd og ting er beskrevet der de hører hjemme	Ekstra elementer som f.eks. <input type="checkbox"/> Kryssreferanser <input type="checkbox"/> bibliografi
Kravspesifikasjon (Kapittel "Innledning - Problemstilling")	Mangler	Vanskelig å forstå hva som skal utvikles basert på beskrivelsen.	Har beskrevet de viktigste funksjonelle kravene.	Utfyllende med f.eks. <input type="checkbox"/> Avgrensninger. <input type="checkbox"/> Passende UML diagrammer <input type="checkbox"/> Andre typer krav (systemkrav, ikke-funksjonelle krav)
Teoretisk grunnlag  Hvilke <b>teorier</b> er benyttet ved utviklingen av løsningen?	Mangler	Mangelfullt beskrevet. Beskriver bare deler av teorigrunnlaget.	God beskrivelse av teoriene om god design. Men beskriver teorier som kandidaten senere ikke refererer til	Har i tillegg inkludert: <input type="checkbox"/> Teori om en eller flere elementer <input type="checkbox"/> Teori og praksis rundt <b>kodestil</b> <input type="checkbox"/> Teori rundt god brukerinteraksjon Har kun omtalt teorier som senere diskuteres i rapporten.
Metode  Hvilke <b>verktøy</b> er benyttet for å løse oppgaven?  Hvilken <b>prosess/prosjektmodell</b> ble fulgt? I denne mappen har vi f.eks. gitt prosjektet i 3 deler med tilbud om 3 tilbakemeldinger.	Mangler	Mangelfull beskrivelse av IDE og prosess/metode	<input type="checkbox"/> Beskriver hvilke verktøy (IDE etc.) som er brukt. <input type="checkbox"/> Versjonskontroll	<input type="checkbox"/> Beskriver i tillegg greit arbeidsmetoden (hvilken arbeidsmetode kandidaten har fulgt) <input type="checkbox"/> Versjonskontroll og refaktorering

KRITERIUM	Ikke bestått (tilsvarer F)	Svak (tilsvarer D/E)	Middels (tilsvarer C)	Utmerket (tilsvarer B/A)
Design og Implementasjon (i kapittel "Resultat")	Mangler	Utydelig, manglende diagram. Skriver «dagbok», og ikke en beskrivelse av rolle- og ansvar til de ulike klassene.	Godt beskrevet <input type="checkbox"/> med bruk av klassediagram <input type="checkbox"/> med god bruk av f.eks. kodeeksempler og skjermdump <input type="checkbox"/> med god beskrivelse av hvilke <b>refaktoring</b> er gjort og hvorfor?	Ekstra utfyllende med f.eks. <input type="checkbox"/> Diagrammer og illustrasjoner <input type="checkbox"/> Resultat av gjennomførte <b>brukertester</b> <input type="checkbox"/> Ryddig og oversiktlig.
Drøfting og refleksjon, Konklusjon	Mangler	Diskusjon: Knappt beskrevet. Ingen reell <i>drøfting</i> av egen løsning i forhold til teoriene beskrevet i Teori-avsnittet..  Konklusjon: Svært kort, der kandidaten ikke peker på de viktigste oppnådde målene i prosjektet	Drøfter greit endelig resultat i forhold til <b>teoriene</b> . God konklusjon der kandidaten oppsummerer prosjektet i forhold til: <input type="checkbox"/> Om kandidaten mener prosjektet er iht de prinsipper og standarder adressert i emnet <input type="checkbox"/> Om kandidaten har fått vist sine ferdigheter igjennom prosjektet	Drøfter <b>svært godt</b> endelig resultat i forhold til <b>teoriene</b> fra læreboken.  Konklusjon: Har også fått med f.eks. hva som burde vært gjort annerledes om det skulle gjøres igjen