

# Oblig 1

DAFE2200 Systemutvikling

Amna Dastgir (s364520)

Mathangi Murugesu (s364580)

Nana Agyeiwaa Buabeng (s359009)

# Innholdsfortegnelse

<b>OPPGAVE 1 – BAKGRUNN FOR SYSTEMET .....</b>	<b>2</b>
A).....	2
B).....	3
<b>OPPGAVE 2 – INTERESSENER FOR SYSTEMET .....</b>	<b>3</b>
A).....	3
B).....	3
C).....	4
<b>OPPGAVE 3 – UTVIKLINGSPROSESS FOR SYSTEMET .....</b>	<b>4</b>
A).....	4
B).....	4
C).....	5
D).....	5
<b>OPPGAVE 4 – KRAVSPESIFIKASJON FOR BILLETTSYSTEMET .....</b>	<b>5</b>
A).....	5
B).....	6
C).....	6
<b>OPPGAVE 5 – USE CASE FOR BILLETTSYSTEMET.....</b>	<b>8</b>
A).....	8
B).....	8
<b>BIBLIOGRAFI .....</b>	<b>10</b>

# Obligatoriske oppgave 1 – Foranalyse og kravhåndtering

## Oppgave 1 – Bakgrunn for systemet

a)

Fordeler:

- i. Ved å bruke det eksisterende kino-billettsalgssystemet vil fordelene være at man ikke trenger å starte fra bunnen av igjen ved utviklingen. Ettersom det allerede eksisterer et system, er en alternativ løsning å forbedre på det som trengs, og eventuelt legge inn nye funksjoner og oppdateringer.
- ii. De fleste gamle systemer er kjente, noe som leverer følelser av kjennskap og komfort. Hele arbeidsflyten er jevn, det er enklere hvordan funksjonaliteter skal aksessereres, samtidig som utføring av funksjonene er raskt å få seg gjennom (Folio3, u.d.).
- iii. Det er ikke alltid å erstatte det gamle systemet som burde være en prioritet, og det også fordi andre prosjekter har mer verdi knyttet til dem. Man kan fokusere på de nye funksjonalitetene som skal bringe inn inntektsstrømmer eller øke selskapets totale produktivitet.

Ulemper:

- iv. Massevis av ny teknologi tar over industrier globalt. Å bytte ut det gamle systemet til helt ny teknologi kan da være en ressurskrevende prosess for enhver bedrift. Avskaffing av ny programvare krever nye ansettelser, og opplæring til flere spesialister, noe som krever mer penger, tid og ressurser (Brown, u.d.).
- v. Med tanke på kompatibilitetsproblemer er det slik at eldre systemer kan støtte visse filer og formater som videre i tid kan bli utdatert. Gammel programvare vil dermed slite med å støtte nye teknologiske krav, som betyr at bedriftene vil streve med å nå sine forretningsmål.
- vi. Eldre systemer tilbyr vanligvis begrenset fleksibilitet, noe som gjør dem utfordrende å endre eller tilpasse. Som et resultat sliter bedrifter med å takle industriens påkjenninger, og heve frem noen av de primære ulempene ved eldre systemer, og deres fleksibilitet.

b)

Gruppen vår velger å benytte oss av et helt nytt system. Det kan muligens ta mer tid å skjønne et system og sette seg inn i det fra bunnen av, sammenliknet med å heller opprette ett helt nytt et. På denne måten er også alt mer nøye og det er dermed sjans for mindre feil og kravhåndteringen blir enklere å utvikle.

## Oppgave 2 – Interessenter for systemet

a)

b)

En interessent kan bli beskrevet som noen som har interesse og er involvert i en bedrift på en eller annen måte. Disse interessentene trenger ikke nødvendigvis å være en enkelt person, men kan også være en større organisasjon. Dette kan blant annet være en sponsorgruppe. Disse sponsorgruppene har ikke et ansvar knyttet til utviklingen og eller driften av systemet. En aktør er noen som har en direkte rolle i en bedrift. Derfor vil en ansatt være en aktør da deres jobb blant annet går ut på å drive systemet ved enkelte roller som å dele ut VR-briller, og skanne billetter ved inngangen. Med dette kan man si at en aktør kan være en interessent, men en interessent ikke nødvendigvis er en aktør. Noen andre interessenter vi finner i systemet til Virtuoso Kino i tillegg til «Ansatt» og «Sponsorgrupper» er «Ledelsen», «Kunde», «Medlem» og «Utviklergruppe».

Navn	Interesse	Eventuelt ansvarsområde
Ledelsen	<ul style="list-style-type: none"><li>• Styre kinoen på en måte som tilfredsstiller alle medarbeiderne</li></ul>	<ul style="list-style-type: none"><li>• passe på at de underordnede gjør jobben sin, samt å styre driften av bedriften.</li></ul>
Kunde	<ul style="list-style-type: none"><li>• Oppleve en virtuell kinoopplevelse</li></ul>	<ul style="list-style-type: none"><li>• Betale nødvendig beløp og å oppføre seg</li></ul>
Medlem	<ul style="list-style-type: none"><li>• Motta ekstra fordeler og raskere billett kjøp</li></ul>	<ul style="list-style-type: none"><li>• Registrere seg i kundeklubben, og gi fra seg kontaklinformasjon</li></ul>
Utviklergruppe	<ul style="list-style-type: none"><li>• Danne et system på en effektiv måte</li></ul>	<ul style="list-style-type: none"><li>• Passe på at systemet møter og krav.</li></ul>
Ansatt	<ul style="list-style-type: none"><li>• Utføre arbeidsoppgavene de har fått utdelt</li></ul>	<ul style="list-style-type: none"><li>• Passe på at alle besøkende har en trivelig opplevelse.</li></ul>
Sponsorgruppe	<ul style="list-style-type: none"><li>• Investerer i bedriften</li></ul>	<ul style="list-style-type: none"><li>• Støtte bedriften økonomisk.</li></ul>

c)

Interessentene «Ledelsen», «Ansatt», «Kunde» og «Medlem» er både interessenter og aktører. Dette er grunnet deres direkte rolle i bedriftens gang som vil si at det er de som tar i bruk samt påvirker systemet på en daglig basis. Begrepet aktør har to delgrupper kalt primære aktører og sekundære aktører. Interessentene nevnt tidligere er eksempler på primære aktører, men «Utviklergruppe» og «Sponsorgruppe» er sekundære aktører. Med dette menes at gruppene har en indirekte rolle i bedriften i motsetning til de primære aktørene. Uten sekundær aktørene hadde ikke Virtuoso Kino blitt dannet da de trenger både økonomisk støtte, og utviklere til å opprette systemet.

### Oppgave 3 – Utviklingsprosess for systemet

a)

Hovedforskjellen mellom plandrevne og smidige prosessmodeller er måten planleggingen gjennomføres. I plandrevne prosesser er alle prosessaktivitetene planlagt på forhånd, dermed vil progresjonen måles i henhold til denne planen. Plandrevne prosesser er ofte tunge. Det vil si at de vektlegger aktiviteter som vanligvis gjøres tidlig i prosessen (planlegging, analyse og design). I smidige prosesser gjøres planleggingen gradvis og det gjør det lettere å endre prosessen for å reflektere eller tilpasse omarbeidet krav for kunden. Dette regnes som lette prosesser og her fokuseres det på fundamentale prosesser (f.eks. «kontinuerlig testing»), og det har færre formelle dokumenter og er ofte mer iterative.

b)

Scrum er en metodikk som strukturerer arbeidsflyt og teamkultur for å levere prosjekter på korte tidslinjer. Fordelene ved Scrum er at systemet blir delt opp i ulike mengder som både er forståelig og håndterbar. Det blir god gruppearbeid og kommunikasjon da hele teamet rette seg etter hva som skjer i prosjektet. Kunden får oppdateringer levert til avtalt tid og får fortløpende tilbakemelding på hvordan deler av systemet fungerer. Dette fører til at tillitt mellom kunder og utviklere etableres. Ulempen er at Scrum leverer biter av leveranser i en til fire ukers perioder.

Kanban er en metodikk sentret rundt visualisering av oppgaver, den leverer oppgaver fortløpende til prosjektet er ferdig. Fokuset er å hindre gjennomstrømning fremfor å begynne på flere oppgaver som vil hope seg opp. Det er mulig å bruke smidig utvikling uten å bruke tid. Denne metoden er nyttig der det er svært vanskelig å gjennomføre oppgavene. Ulempen er at det ikke alltid er fleksibelt (Coursera, 2022).

c)

Det er ikke lett å forutsi endringene som kan komme underveis i utviklingen av systemet. Siden det er et nytt system som blir lagd er det mulig å bruke fleksible metoder for å kunne tilpasse de nye ideene og kravene fra kunden. Fleksibilitet fører også til at utviklingsprosessen blir mer effektiv. Dersom aktuelle ideer oppstår under prosessen, vil ikke tidligere behov fra kunden være like relevant for det nye systemet.

d)

Vi tenker at smidig prosess er mest egnet for utviklingen av systemet til Virtuoso Kino. Siden det er et nytt system er det lettere med smidig prosess da det sørger for fleksible og tilpasningsdyktige strukturer for utviklingen av systemet. Scrum er effektivt å bruke da de støtter fleksibel produktutvikling. Hvis vi hadde videreutviklet et gammelt system hadde det vært nyttig med plandreven modell, det hadde vært mindre krav for endringer og implementeringer da det er et system som vi kjenner til.

## Oppgave 4 – Kravspesifikasjon for billettsystemet

a)

Funksjonelle krav:

- i. Systemet skal **samle inn all data** (dette vil senere bli brukt for å generere statistikk og videreutvikling)
- ii. Man skal kunne **kjøre en eller flere standardbilletter** til ønsket filmvisning
- iii. **Billettens strekkode skannes** ved inngangen til kinosalen
- iv. En får **utdelt virtuelle virkelighetsbriller** (VR-briller) som gir dem følelsen av å være med i filmen som vises

- v. Når betalingen er godkjent skal **billetten(e) være tilgjengelige** for kunden i PDF-format med en strekkode
- vi. Hvis kunden ønsker ekstra fordeler og raskere billett kjøp kan man **registrere seg** som medlem av Virtuoso Kinos Kundeklubb.

b)

Hvordan skal systemet implementere de ikke-funksjonelle kravene?

- i. Brukervennlighet: 90% (eller flere) skal kunne bestille billetter på under fem minutter etter å ha brukt systemet minst 3 ganger
- ii. Ytelse: Systemet skal kunne håndtere inntil flere 1000 samtidige brukere
- iii. Språk: Skjerminnhold, meldinger, databaser, tabeller, rapport navn, online og offline dokumentasjon skal være på norsk bokmål
- iv. Teknisk dokumentasjon: Koden burde dokumenteres på en måte slik at utviklere kan skjønne koden uten særlige komplikasjoner
- v. Hastighet og pålitelighet: Alt med responstid og oppdatering av skjermen skal være på sitt best. Det skal gjennomsnittlig være få feil, sannsynligheten for utilgjengelighet er minimalistisk, og feilrate skårer bra.

Produktkrav: Brukervennlighet, hastighet, ytelse og språk

Organisatoriske krav: Budsjett, teknisk dokumentasjon

Eksterne krav: Regler og lovverk, pålitelighet

c)

Stresstesting kan brukes for å sjekke hastighet. Flere brukere kan utføre de samme transaksjonene på samme data. For eksempel kan flere klienter være koblet til serverne med forskjellige arbeidsbelastninger. Tenketiden kan reduseres til «null» for å stresse serverne til maksimalt stress. Tenketiden er for eksempel som tidsintervallet mellom å skrive brukernavn og passord (Software testing help, 2022).

Volumtesting kan brukes for å sjekke ytelse for eksempel når programvaren er underlagt store datamengder, kontrollerer grensen hvor programvaren svikter. Hvis applikasjonen behandler

databasen for å lage en rapport, vil en volumtest for eksempel være å bruke et stort resultatsett og sjekke om utskriften til rapporten er korrekt (Software testing help, 2022).

Brukervennlighet kan testes ved bruk av noe så enkelt som brukervennlighetstesting. Da evalueres systemet for menneskelig bruk eller sjekker om den er egnet til bruk. Det kan da sjekkes om resultatet er riktig og meningsfylt, og om det møter opp til virksomhetens krav. Er feilene diagnostisert er GUI riktig og konsistent med standarden? Og sist, men ikke viktigst, er applikasjonen enkel å bruke? Her kan det ikke-funksjonelle kravet «Språk» også passe inn (Software testing help, 2022).

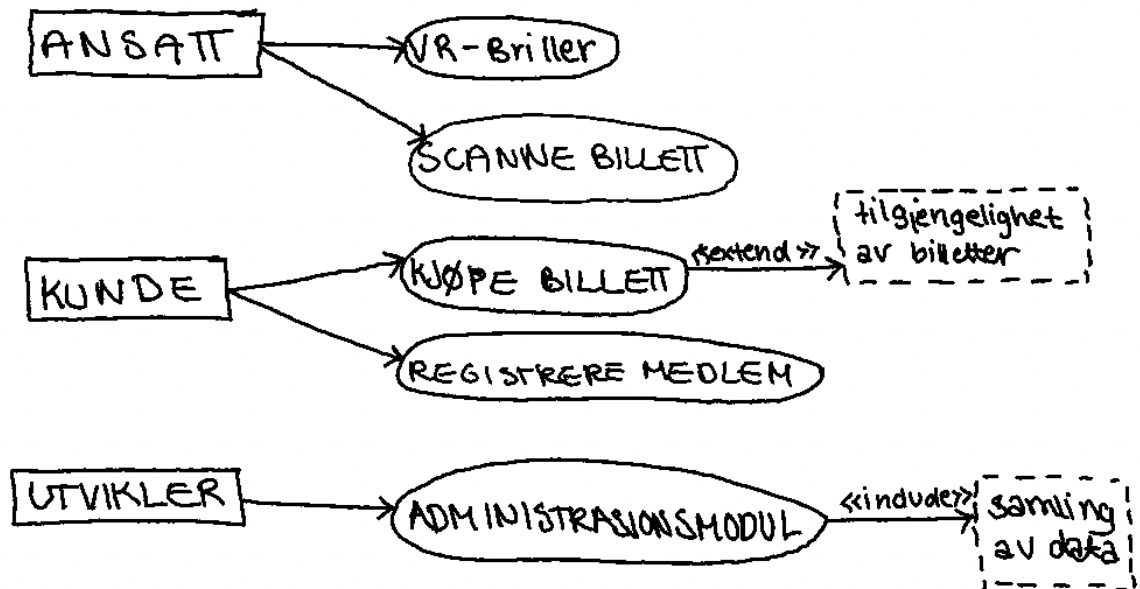
Teknisk dokumentasjon kan ta for seg tester for dokumentasjon, feilt, sikkerhet og stabilitet (Software testing help, 2022).



## Oppgave 5 – Use case for billettsystemet

a)

Bilde vedlagt:



b)

Tekstlig beskrivelse:

- 1) Navn: Kjøp billett.
- 2) Aktører: Kunde
- 3) Prebetingelse: ledig tid, ledig antall seter
- 4) Postbetingelse: Tilgjengeliggjøring av billettene på PDF-format

### Hoved flyt:

- 1) Velge film, dato, tid, antall seter
- 2) Bruker ber systemet om å finne valgt film på ønsket tid og med riktig mengde ledige seter
- 3) Systemet returnerer resultater på søket
- 4) Bruker registrer input (med f.eks navn, telefonnummer, epostadresse)
- 5) Systemet sjekker inputen
- 6) Systemet lagrer inputen
- 7) Systemet bekrefter forespørsel på bestilling av billetter
- 8) Systemet registrer billettene på valgt tid, valgt mengde seter og film

**Alternativ flyt 1:**

Steg 1: Det er enten ikke ledige seter i kinosalen på valgt film/ikke ledige dato/  
ikke ledig tid

Returnerer til steg 1.

**Alternativ flyt 2:**

Steg 4: Ugyldig brukerinformasjon

Returnerer til steg 4 med blanke input.

## Bibliografi

Brown, E. (u.d.). *Why you should keep your legacy system*. Hentet fra Unstoppable Software:

<https://unstoppablesoftware.com/why-you-should-keep-your-legacy-system/>

Coursera. (2022, August 10). *Kanban vs. Scrum: What's the difference?* Hentet fra Coursera:

<https://www.coursera.org/articles/kanban-vs-scrum?fbclid=IwAR0QNHoSObneSNXCq7SQWRykomfIZ1e3RnKrZzt0p64x1E4Kjngur5lg7KQ>

Folio3. (u.d.). *Benefits of replacing legacy systems in business*. Hentet fra Folio3:

<https://dynamics.folio3.com/blog/legacy-system/>

Alle forelesningspowerpointer og ukesoppgaver har blitt brukt som kilder på indirekte eller direkte bevis.