# The Movies 01

|  |  |
| --- | --- |
| **Læringsmål** | * Læringsmål fra 1. semester |
| **Forventet læringsudbytte [SOLO]** | * Læringsudbytter fra 1. semester |
| **Din forberedelse** | Denne opgave omhandler tidligere emner fra 1. semester, som du skal repetere i øvelserne gennem en case, som du allerede har arbejdet med i systemudvikling, nemlig TheMovies.  I forbindelse med TheMovies-casen:   * **VIGTIGT:** Genlæs case ”TheMovies” tidligere udleveret i systemudvikling * **VIGTIGT:** Genopfrisk TheMovies-afleveringen fra uge 49, som dit team udarbejdede   Designsporet (efter behov):   * Genlæs kapitlerne i Larman om DCD, pakkediagram, og sekvensdiagrammer samt GRASP-principper/-mønstre   Programmeringssporet (gennemgå specielt):   * Persistens via tekstfiler (CSV-format) med brug af StreamReader og StreamWriter * CSV-format (bemærk at separatoren ikke behøver at være et komma)   + [Comma-separated values](https://en.wikipedia.org/wiki/Comma-separated_values)   + [What is a CSV file (overview)](https://www.youtube.com/watch?v=OGOD2Fqy5k8) (video: 3:04) * Repository (og indkapsling) * Interface (med brug af generics: <T>) samt nedarvning * C# klasse (felt, property, metode, constructor)   Derudover (om nødvendigt for dig) læs generelt om:   * Lagdeling, MVVM, databinding, XAML, code-behind * Se på dit landkort (nu med fokus på design- og implementeringsartefakter) |

Denne opgave dækker 3 programmeringsdage for hele uge 5 og er tænkt som en rolig opstart primært med praktisk repetition af udvalgte emner fra 1. semester. Du genopfrisker de emner, du selv vurderer, du skal styrkes i, dog sørg for som minimum at sættes dig ind i TheMovies-casen inklusiv din aflevering i uge 49.

# Dagens ord:



# Øvelse 1: Teamdannelse

Det er første gang i 2. semester, du skal have programmering i dit nye team, og det er selvfølgelig vigtigt, at du møder dine nye ”kolleger” og får skabt et fælles grundlag for et godt samarbejde i en ny konstellation.

Hvis du ikke allerede har gjort det i dit team, så:

* Lav en præsentationsrunde i dit team, hvor hver kan præsentere sig selv og sine forventninger til det nye samarbejde, svage og stærke sider (Belbin, faglige emner, mm.); lær hinanden at kende.
* Påbegynd arbejdet med at etablere fælles kvalitetskriterier, kodestandarder og programmerings­relevante overvejelser. Du bliver ikke færdig med dette i dit team på en gang; det tager noget tid at få plads. Så til at starte med, tal sammen om, hvilken proces du og dit team vil anvende og følge for at finde et fælles fodslag med aftaler, gruppekontrakt og lignende.

Som sagt bliver du og dit team ikke færdige med dette inden for tidsrammen; der bliver mulighed for at fortsætte med denne aktivitet de næste gange. Husk, at der er 3 undervisningsdage til alle øvelserne i denne opgave, så du kan genbesøge denne teamøvelse, når du starter hver dag med dit team.

*Tidsramme: 30 minutter*

# Øvelse 2: Præsentér TheMovies analyseartefakter

Du har tidligere (i uge 49) med dit forrige team udarbejdet og afleveret et analysearbejdet i forbindelse med TheMovies-casen. Dette forarbejde danner grundlaget for denne opgave og betyder, at du og delta­gerne i dit nye team hver kan bidrage med forskellige analyser af samme case i form af analyse­artefakter (objektmodel, domænemodel, use cases, SSD’er og operationskontrakter).

* Brug **Ordet rundt** i teamet til at præsentere og kort reflektere over den tidligere TheMovies-aflevering, som hver enkelt har været med til at udarbejde. I denne øvelse skal I kun præsentere for hinanden, hvad I hver især har været med til at lave.  
  Sørg for, at alle får mulighed for at tale. Max. 5 minutter hver.

*Tidsramme: ca. 30 minutter*

# Øvelse 3: TheMovies – Design og Implementering

Du skal sammen med dit nye team designe og implementere (dele af) et it-system for TheMovies baseret på udvalgte artefakter fra jeres forskellige analysebidrag. Du kan opfatte bidragene præsenteret i øvelse 2 som et analysearbejde udarbejdet fra ”fremmed” hold, som du og dit team skal designe og udvikle videre fra.

## Øvelse 3.1: Udvælgelse af analyse-artefakter

Udfør følgende:

* Udvælg de analyseartefakter, du og dit team finder mest relevante og dækkende som grundlag til design og implementering af TheMovies-systemet. Begrund de valg, I sammen laver.

Det er meget vigtigt, at I ikke gennemgår en fuld analysefase igen, men i stedet kun vælger og konsoliderer de analyseartefakter, dit team finder relevante. Måske har et team lavet en detaljeret domænemodel, et andet team nogle gode use cases, osv., så I får sammensat nogle gode udgangspunkter til de næste øvelser.

#### Udleveret CSV-fil med data

Ud over de valgte analyseartefakter, så er der i Git materiale-mappen udleveret et eksempel på konkrete data med både et regneark *Ex38-TheMovies.xlsx* og en CSV-fil *Ex38-TheMovies.CSV*, som har været anvendt til at registrere film, forestillinger og bookinger (det er de samme data i de to filer, bare med forskellig format). Det står frit for, hvordan data i regnearket anvendes i både design- og implementeringsøjemed; I kan anvende det, som det er er, eller supplere det med yderligere relevante data. Hele regnearket kan anvendes eller dele af det, som det passer dig og dit team bedst.

## Øvelse 3.2: Planlæg dit iterative design- og implementeringsforløb med tidsplan

Da der er 3 undervisningsdage til denne opgave, er det vigtigt, at der planlægges et iterativt forløb, som fokuserer på kun en eller to udvalgte use cases, som teamet baserer sit design og sin implementering ud fra. I kan dele teamet i to grupper, hvis I synes det er bedre, eller arbejde alle sammen.

Det er frit, hvilke use cases der designes og implementeres, men hold det forholdsvist simpelt. Det er vigtigst, at du og teamet arbejder med alle de centrale artefakttyper (se krav forneden). Sørg for, at alle deltager i de forskellige trin/produkter, så opgaven opfylder formålet med at repetere og genopfriske tidligere faglige temaer fra 1. semester.

Der stilles følgende uafvigelige krav til design- og implementeringsarbejdet med udgangspunkt i en valgt use case.

Krav til designfasen:

* Beskriv en lagdelt arkitektur via et pakkediagram (kun med lag, klassenavne og afhængigheder), hvor der er krav om, at MVVM-arkitekturmønstret skal anvendes i en WPF-applikation.
* Udarbejd en DCD med de klasser (med klassenavne, attributter og operationer), som relevante for use casen. Undgå at have alle domæneklasser med, men overhold et krav om, at DCD’et skal indeholde mindst én View-klasse, mindst én ViewModel-klasse, mindst én Repository-klasse og mindst én Model-klasse, så alle typer er repræsenteret.
* Udarbejd en SD for en udvalgt operationskontrakt, hvor kravet er, at SD’et tydeligt skal vise, hvordan den i SD’et beskrevne operation ændrer systemets tilstand fra kontraktens præ-betingelse (en: precondition) til dets post-betingelse (en: postcondition).
* Planlæg persistens i CSV-tekstfiler med relevant syntaks. Det skal være en Repository-klasse, som har ansvaret for at persistere. Sørg for enten, at I kan læse data fra en CSV-fil ind i it-systemet eller skrive data til en CSV-fil; gerne begge dele, hvis der er tid. Husk at holde det simpelt og vælg det letteste først. Benyt gerne data fra den udleverede CSV-fil.
* Design en WPF-brugergrænseflade med ét hovedvindue samt en dialogboks til indtastning af data; igen hold det simpelt via XAML. Der må gerne bruges databinding mellem View-laget og ViewModel-laget.

Krav til implementeringsfasen:

* Det er særdeles vigtigt, at du i denne fase får repeteret, hvordan man syntaktisk definerer en C#-klasse med dets 4 centrale medlemmer: fields, properties, constructors og methods, herunder også hvordan man implementerer de fire OOP-principper indkapsling, polymorfi, nedarvning og abstraktion i en C#-klasse.
* **VIGTIGT:** I den iterative proces, sørg for, at hver i dit team for mulighed for at lave en C#-klasse uanset hvilken type, det er; så husk at uddelegere opgaven.

Generelt hav fokus på at få ”hul” igennem fra en brugergrænseflade til persisteret data i en CSV-fil, dvs gennem alle lag. Opgaven repræsenterer et ultrakort sprint, så start simpelt, f.eks. med at vise alle film fra en CSV-fil, eller noget tilsvarende.

God fornøjelse!