

# Exercise 7

Software Development 2018  
Department of Computer Science  
University of Copenhagen

Emil Møller Hansen <ckb257@alumni.ku.dk>,  
Casper Bresdahl <whs715@alumni.ku.dk>,  
Torben Olai Milhøj <vrw704@alumni.ku.dk>

Version 1;  
**Due:** Friday, March 16th

## 1 Analyse

Vi har indelt problemet i forskellige koncepter, der hver har deres eget ansvar, resultatet kan ses herunder.

Responsibility description	Concept name
Læser fil og returnerer Level objekt med info herfra	File reader
Class til at beskrive Level objekter	Level class
Singleton der skal indeholde en dictionary af Levels	Levels keeper
Sørger for alle levels loades før start på spil	Level loader
Laver en entity container til level baseret på Level objekt	Level creator
Returnerer en entity som repræsenterer blok i spillet	Entity creator

**Table 1:** Dele af vores løsning inddelt i responsibilities og koncepter

Vi har opdelt koncepterne efter koncepttyperne Know og Do, så hvert koncept har netop én type. Vores Do koncepter har hver ansvar for én opgave, eksempelvis file reader der kun skal lave et level ud fra informationen fra en fil. Vi har 6 koncepter hvoraf 4 er Do- og 2 er know-koncepter.

Vores entity creator laver for nu kun firkanter der alle er lige store, og som har et billede, da vi antager vi ikke skal implementere collision detection i denne uge. På længere sigt skal der være forskel på størrelsen og formen på de forskellige objekter.

I tabel 2 herunder ses det hvordan de forskellige koncepter skal kommunikere med hinanden.

Concept pair	Association description	Association name
Level Loader og file reader	Giver filnavn og får level tilbage	Hent data
File reader og level class	Laver Level objekt	Genererer
Level loader og levels keeper	Levels keeper gemmer Level fra loader	Gem data
Level creator og levels keeper	Får Level objekt fra Keeper	Hent data
Level creator og entity creator	Får entity med givne parameter	Genererer

**Table 2:** Associationer mellem forskellige koncepter

Det ses at vi har besluttet at Level loader skal kalde en funktion i file reader for at denne returnerer et Level som Level loader gemmer i levels keeper. I starten af planlægningsprocessen var det egentlig planen at file reader skulle sende Level-objektet til levels keeper, men da vi fik et bedre overblik over de forskellige ansvarsområder kunne vi se at det gav bedre mening at gøre det som beskrevet ovenfor. På denne måde får vi nemlig en klarere adskillelse mellem Level loaders og file readers ansvar, i og med at file readers eneste job er at læse og parse filen, hvorefter level loader behandler Level-objektet.

I tabel 3 ses det hvilke attributer vi tænker at hvert concept skal have.

Det ses at det ikke er alle vores koncepter der har attributer, dette skyldes at det ikke er dem alle der skal gemme informationer. Eksempelvis skal file reader blot læse filen og generere objekter på baggrund af dette, den behøver altså

Concept	Attribute	Attribute description
Level loader	Levels	En liste af filnavnene på level layouts
File reader	Ingen attributer	
Level class	Level array	2D-struktur der indeholder level design
	Name	Navn på level
	Costommer	Information om costommers i dette level
	Platforms	Info om hvad der er platforme
	Decoder	Gemmer sammenhængen mellem karakterer og billedenavne
Levels keeper	Levels	Container til alle levels
	Index	Tæller der holder styr på hvor mange levels der er i levels
Level creator	Levels	Reference levels keepers
Entity creator	Ingen attributer	

**Table 3:** Attributer for hvert koncept

ikke gemme informationen til senere brug. I modsætning til dette skal hvert Level objekt gemme informationen om den selv.