

---

# Dispositioner - Software Design

---

AARHUS SCHOOL OF ENGINEERING

EKSAMENSDISPOSITIONER TIL BRUG VED EKSAMEN I SOFTWARE DESIGN

Bjørn Nørgaard  
IKT  
201370248  
bjornnorgaard@post.au.dk

Joachim Andersen  
IKT  
20137032  
joachimdam@post.au.dk

## Indholdsfortegnelse

<b>1</b>	<b>Solid 1 - SRP, ISP og DIP</b>	<b>1</b>
1.1	Fokuspunkter . . . . .	1
<b>2</b>	<b>Solid 2 - OCP, LSP og DIP</b>	<b>2</b>
2.1	Fokuspunkter . . . . .	2
<b>3</b>	<b>Patterns 1 - GoF Strategy + GoF Template Method</b>	<b>3</b>
3.1	Fokuspunkter . . . . .	3
<b>4</b>	<b>Patterns 2 - GoF Observer</b>	<b>4</b>
4.1	Fokuspunkter . . . . .	4
<b>5</b>	<b>Patterns 3 - GoF Singleton + Method/Abstract Factory</b>	<b>5</b>
5.1	Fokuspunkter . . . . .	5
<b>6</b>	<b>Patterns 4 - State patterns</b>	<b>6</b>
6.1	Fokuspunkter . . . . .	6
<b>7</b>	<b>Patterns 5 - Model-View-Controller og Model-View-ViewModel</b>	<b>7</b>
7.1	Fokuspunkter . . . . .	7
<b>8</b>	<b>Patterns 6 - Redegør for følgende concurrency mønstre</b>	<b>8</b>
8.1	Fokuspunkter . . . . .	8
<b>9</b>	<b>Domænemodeller og Domain Driven Design</b>	<b>9</b>
9.1	Fokuspunkter . . . . .	9
<b>10</b>	<b>Software arkitektur</b>	<b>10</b>
10.1	Fokuspunkter . . . . .	10

## List of Figures

## List of Tables

# 1 Solid 1 - SRP, ISP og DIP

## 1.1 Fokuspunkter

- Redegør for designprincipperne:
  - Single Responsibility Principle (SRP).
  - Interface Segregation Principle (ISP).
  - Dependency Inversion Principle (DIP)
- Redegør for, hvordan du mener anvendelsen af principperne fremmer godt SW design.
- Vis et eksempel på anvendelsen af et eller flere af principperne i SW design.
- Redegør for konsekvenserne ved anvendelsen af principperne - har det nogle ulemper?

## 2 Solid 2 - OCP, LSP og DIP

### 2.1 Fokuspunkter

- Redegør for:
  - Open-Closed Principle (OCP).
  - Lisskov's Substitution Principle (ISP).
  - Dependency Inversion Principle (DIP).
- Redegør for, hvordan du mener anvendelsen af principperne fremmer godt SW design.
- Vis et eksempel på anvendelsen af et eller flere af principperne i SW design.
- Redegør for konsekvenserne ved anvendelsen af OCP, LSP og/eller DIP - har det nogle ulemper?

## 3 Patterns 1 - GoF Strategy + GoF Template Method

### 3.1 Fokuspunkter

- Redegør for, hvad et software design pattern er.
- Sammenlign de to design patterns GoF Strategy og GoF Template Method - hvornår vil du anvende hvilket, og hvorfor?
- Vis et designeksempel på anvendelsen af GoF Strategy.
- Redegør for, hvordan anvendelsen af GoF Template fremmer godt SW design.
- Redegør for, hvilke(t) SOLID-princip(per) du mener anvendelsen af GoF Strategy understøtter.

## 4 Patterns 2 - GoF Observer

### 4.1 Fokuspunkter

- Redegør for, hvad et software design pattern er.
- Redegør for opbygningen af GoF Observer.
- Sammenlign de forskellige varianter, af GoF Observer - hvilken vil du anvende hvornår?
- Redegør for, hvordan anvendelsen af GoF Observer fremmer godt software design.
- Redegør for fordele og ulemper ved anvendelsen af GoF Observer.
- Redegør for, hvilke(t) SOLID-princip(per) du mener anvendelsen af GoF Observer undersøger.

## 5 Patterns 3 - GoF Singleton + Method/Abstract Factory

### 5.1 Fokuspunkter

- Redegør for, hvad et software design pattern er.
- Redegør for opbygningen af GoF Factory Method og GoF Abstract Factory.
- Giv et designeksempel på anvendelsen af GoF Abstract Factory.
- Redegør for opbygningen af GoF Singleton.
- Redegør for fordele og ulemper ved anvendelsen af GoF Singleton



## 6 Patterns 4 - State patterns

### 6.1 Fokuspunkter

- Redegør for, hvad et software design pattern er.
- Redegør for de forskellige måder at implementere en state machine på.
- Redegør for opbygning af GoF State Pattern
- Sammenlign switch/case-implementering med GoF State
- Redegør for fordele og ulemper ved anvendelsen af GoF State
- Redegør for, hvordan et UML (SysML) state machine diagram mapper til GoF State.

## 7 Patterns 5 - Model-View-Controller og Model-View-ViewModel

### 7.1 Fokuspunkter

- Redegør for, hvad et software design pattern er.
- Redegør for Model-View-Control mønstret og dets variationer
- Redegør for Model-ViewModel mønstret

## 8 Patterns 6 - Redegør for følgende concurrency mønstre

### 8.1 Fokuspunkter

- Parallel Loops
- Passing data
- Producer/consumer
- Mapreduce
- Shared state

## 9 Domænemodeller og Domain Driven Design

### 9.1 Fokuspunkter

- Hvad er en domændemodel?
- Hvordan dokumenteres den?
- Hvad bruges domænemodellen til?
- Hvilke metoder kan man bruge til at finde de konceptuelle klasser?
- Redegør for begrebet Domain Driven Design.

## 10 Software arkitektur

### 10.1 Fokuspunkter

- Redegøre for begrebet softwarearkitektur.
- Hvordan er den typiske software arkitektur?
- Hvordan udarbejdes en software arkitektur?
- Hvordan dokumenteres en software arkitektur?
- Hvorledes udarbejdes og dokumentes en concurrency model?