

Mads Sørensen, Bertram Kjær, Lucas Schoubye, Ismail Ali, Mark Nielsen

14. november 2021

# Indhold

<b>1 Indledning</b>	<b>4</b>
<b>2 Projektplanlægning</b>	<b>4</b>
<b>3 Design</b>	<b>5</b>
3.1 Designklassediagram . . . . .	5
3.2 Sekvensdiagram . . . . .	7
3.3 GRASP . . . . .	7
3.3.1 Creator . . . . .	7
3.3.2 Information Expert . . . . .	7
3.3.3 Low Coupling High Cohesion . . . . .	7
3.3.4 Polymorphism . . . . .	7
3.3.5 Pure Fabrication . . . . .	7

## **Abstract**

# **1 Indledning**

I denne rapport vil vi dokumentere vores arbejde, herunder analyse, design og implementering af projektet. Opgaven består af at lave junior matador spillet med de engelske regler 2-4 spillere. Spillet går på tur hvor der skiftevis slås med terninger for at bevæge sin spillebrik. Når en spiller lander på et felt købes det for et specifikt beløb, hvis ingen anden spiller ejer det. Når en spiller lander på et felt ejet af en anden spiller skal de betale et beløb til spilleren. Spillet bliver ved indtil den første spiller er løbet tør for penge. I rapporten dokumenteres b.l.a. de forskellige artefakter, vi har udarbejdet, under de forskellige faser; analyse, design, implementering og test. Der er anvendt agile udviklingsmetoder, herunder at udvikle i små iterationer. Programmet er udviklet i Java med værktøjet IntelliJ, og skal bruges i DTU's databare.

# **2 Projektplanlægning**

**Tidsplan**

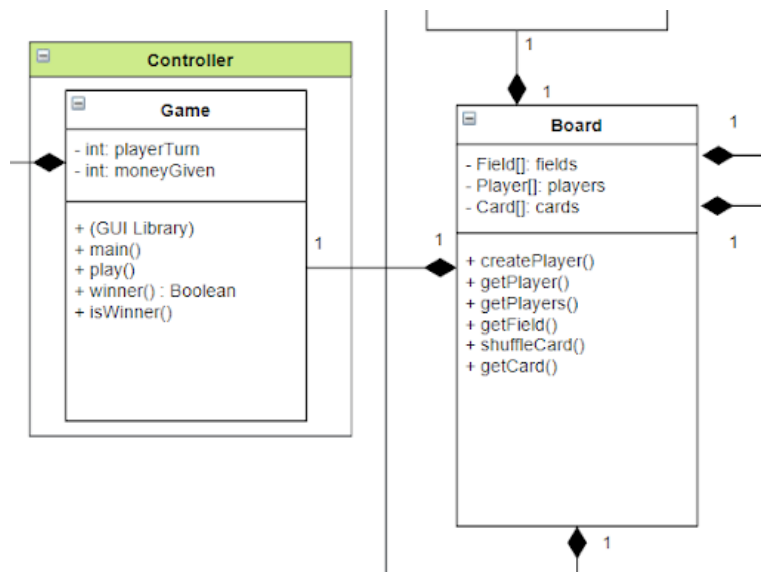
**Samarbejdsaftale**

**Mødereferater**

**Værktøjer**

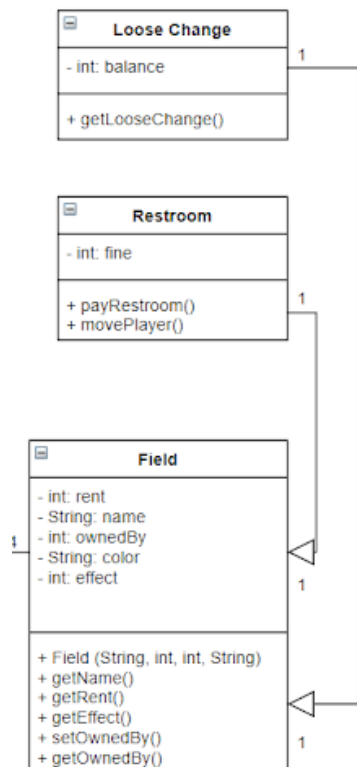
### 3 Design

#### 3.1 Designklassediagram



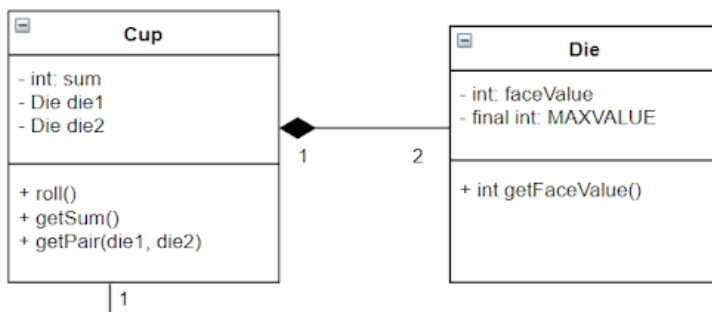
Figur 1: Design klasse diagram udsnit af Game og Board

Som der ses på billedet ovenover er vores main metode er i game klassen sammen med reglerne for logik, hvor board er en controller der udfører logik. Man kunne tage main-metoden ud i en separat klasse, men det har vi valgt ikke at gøre, da vi vil have vores main metode i vores game klasse. Der er valgt at lave nedrivning fra field klassen til Loose Change klassen samt Restroom klassen, da Loose Change klassen og Restroom klassen er deler flere attributter med field klassen dog er de begge lidt forskellige fra field klassen, som ses her:



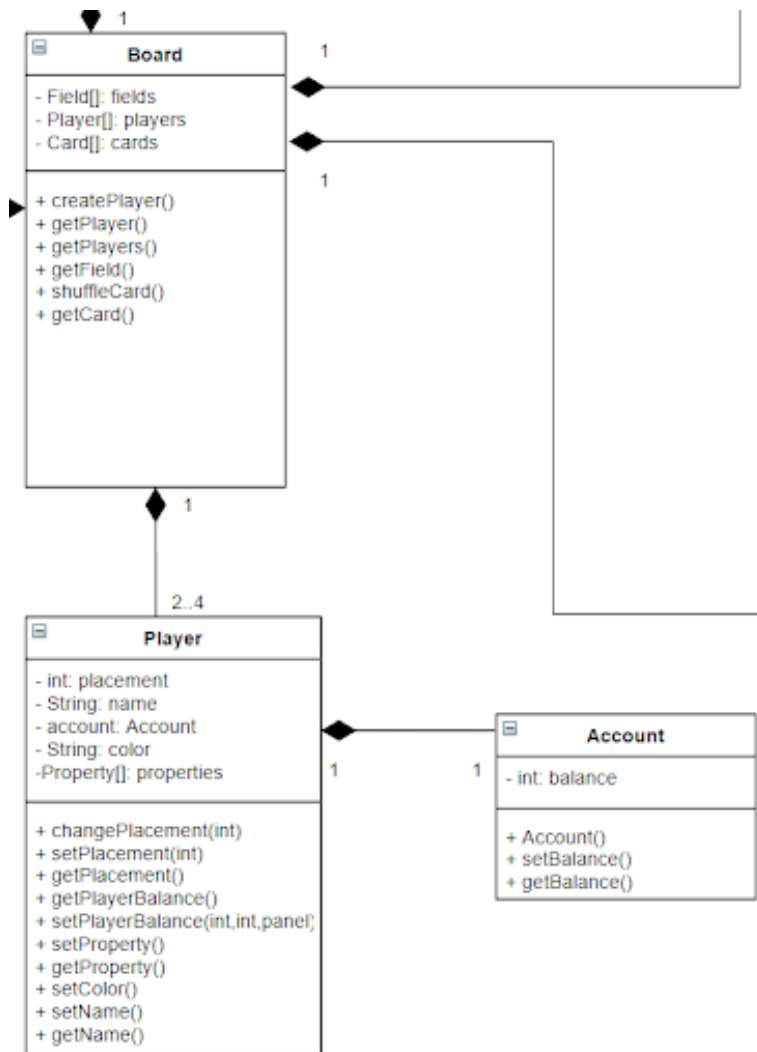
Figur 2: Design klasse diagram udsnit af Field, Restroom og Loose Change

Der er valgt at have en Cup klasse samt en Die klasse hvor Die klassen indeholder en metode der giver et tilfældigt tal mellem 1 og 6 som simulere en rigtig terning. Det er så Cup klassen simulere et rafflebæger som slår med terningen ved hjælp af roll metoden hvor der tages to terninger fra Die klassen og skabes 2 værdier. Herefter er der getSum metoden der finder en sum af de to terninger, hvilket er det slag vores spiller har lavet. Dette ses her i design klassesdiagrammet:



Figur 3: Design klasse diagram udsnit af Cup og Die

Der er valgt at have en player klasse som simulere vores spillere der bliver tildelt navn, farve, placering og ejede grunde med mere. Til player klassen er der koblet en account klasse som styrer spillerens balance, det er også denne attribut der er valgt til i Board klassen at bestemme om en spiller er gået bankerot og derfor har tabt spillet.



Figur 4: Design klasse diagram udsnit af Board, Player og Account

## 3.2 Sekvensdiagram

## 3.3 GRASP

### 3.3.1 Creator

### 3.3.2 Information Expert

### 3.3.3 Low Coupling High Cohesion

### 3.3.4 Polymorphism

### 3.3.5 Pure Fabrication