

**Projektopgave efterår 2012 - jan 2013**  
**02312-14 Indledende programmering og 02313 Udviklingsmetoder til IT-Systemer.**

Projekt navn: del2

Gruppe nr: 21.

Afleveringsfrist: Mandag den 12/11-2012 kl. 05:00

Denne rapport er afleveret via Campusnet (der skrives ikke under)

Denne rapport indeholder 15 sider incl. denne side.

*Studie nr, Efternavn, Fornavne*

s122996, Caspersen, Martin

Kontakt person (Projektleder)



s100182, Baltzersen, Jesper Engholm



# Indhold

<b>Indhold</b>	<b>2</b>
<b>1 Indledning[1]</b>	<b>3</b>
1.1 Ansvarfordeling . . . . .	3
<b>2 Krav</b>	<b>4</b>
2.1 Udgangspunkt . . . . .	4
2.2 Mål . . . . .	4
<b>3 Design</b>	<b>4</b>
<b>4 Implementering</b>	<b>4</b>
<b>5 Test</b>	<b>4</b>
<b>6 Konklusion</b>	<b>4</b>
Referencer . . . . .	5

# 1 Indledning[1]

Denne rapport er udarbejdet i kurserne 02313 Udviklingsmetoder til IT-systemer, 02312 Indledende programmering og 02314 Indledende programmering på første semester af Diplomineniør IT. Opgaven er del et af tre CDIO opgaver på første semester, det overordnede formål med opgaverne på semesteret er at lave et Matador spil. Denne første opgave fokuserer på at implementere en terningklasse og en MatadorRafleBaeger klasse til at holde terning instanser. Den nærmere kravspecificering til opgaven findes i "CDIO\_opgave\_del2.pdf"[Nyb12], de relevante dele fra denne vil blive omhandlet i rapporten. Opgaven bygger i på undervisning modtaget i ovenstående kurser hvor der blev benyttet bøgerne [Lar04] og [LL12]. Igennem opgaverne vil klasser blive vist på følgende måde *KlasseNavn* og metoder på følgende måde *metodeNavn()*.

Rapporten er sat med L<sup>A</sup>T<sub>E</sub>X og alle UML diagrammer er lavet med TikZ-UML.

Bemærk venligst at Martin Caspersen tidligere var med i gruppe 14 og Jesper Engholm Baltzersen tidligere var med i gruppe 17. Begge skrev størstedelen af rapporterne til CDIO del 1 og derfor kan der forekomme passager i denne rapport som minder om passager fra gruppe 14 og 17s rapporter til CDIO del 1.

## 1.1 Ansvarfordeling

Ansvarsfordelingen i opgaven er anført i overskrifterne til de forskellige afsnit og benytter følgende numre for at beskrive gruppens medlemmer:

1. Martin Caspersen
2. Jesper Engholm Baltzersen

Ansvarsfordelingen overskues nemmest i indholdsfortegnelsen. Hvis der er nummer på et af hovedafsnittene og ikke nogen numre på underafsnit er det pågældende medlem ansvarlig for alle underafsnittene også.

Ansaret for kodningen af de forskellige klasser er beskrevet herunder:

- Die [1, 2]
- MatadorRafleBaeger [1, 2]
- Game [1, 2]
- Player [1, 2]
- BoundaryToGUI [1, 2]
- BoundaryToPlayer [1, 2]
- Main [1]
- TestMatadorRafleBaeger [1]
- Board [1, 2]

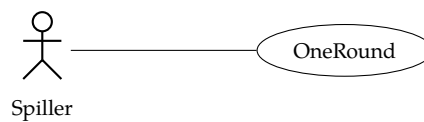
- Field og underklasser [1, 2]
- TestDie [1, 2]

## 2 Krav

### 2.1 Udgangspunkt

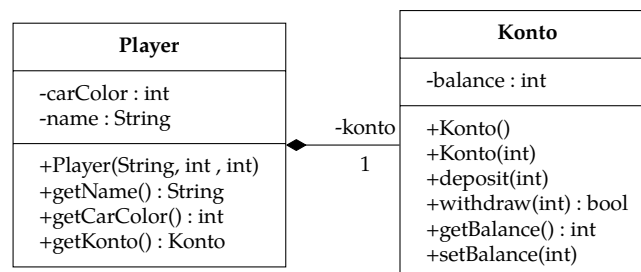
### 2.2 Mål

**Figur 2.1:** Usecasediagram.



## 3 Design

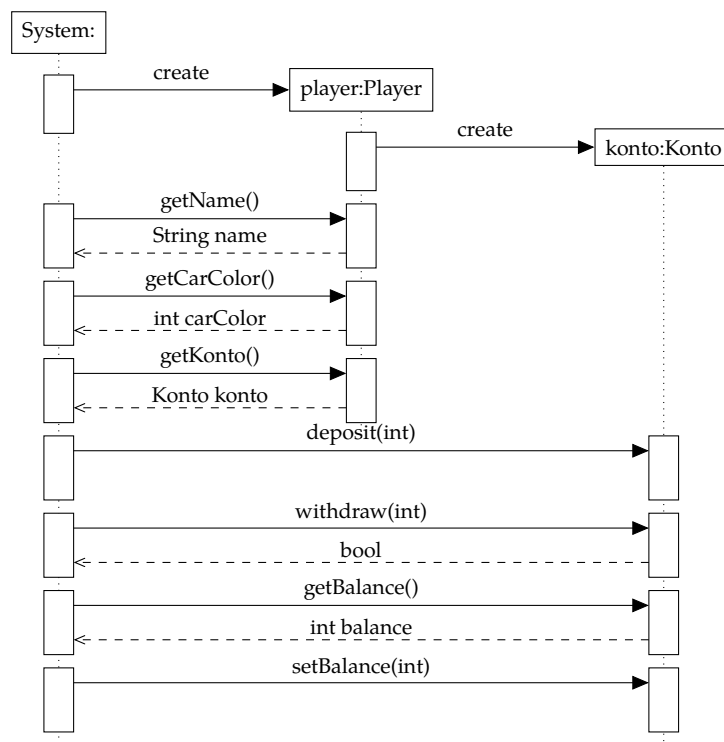
**Figur 3.1:** Designklassediagram der viser *Player* og *Konto* klasserne.



## 4 Implementering

## 5 Test

## 6 Konklusion

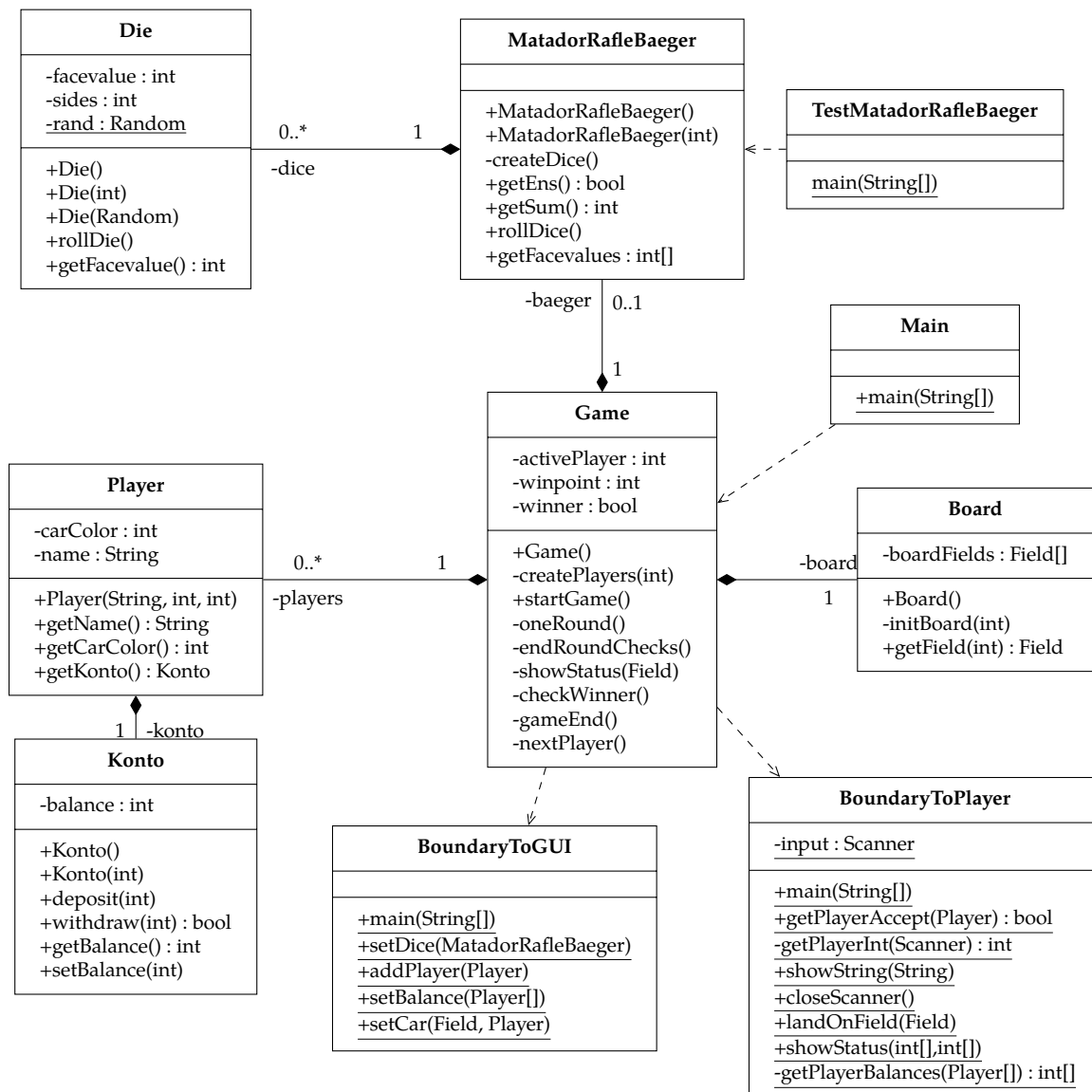
**Figur 3.2:** Sekvensdiagram der viser *Player* og *Konto* klasserne.

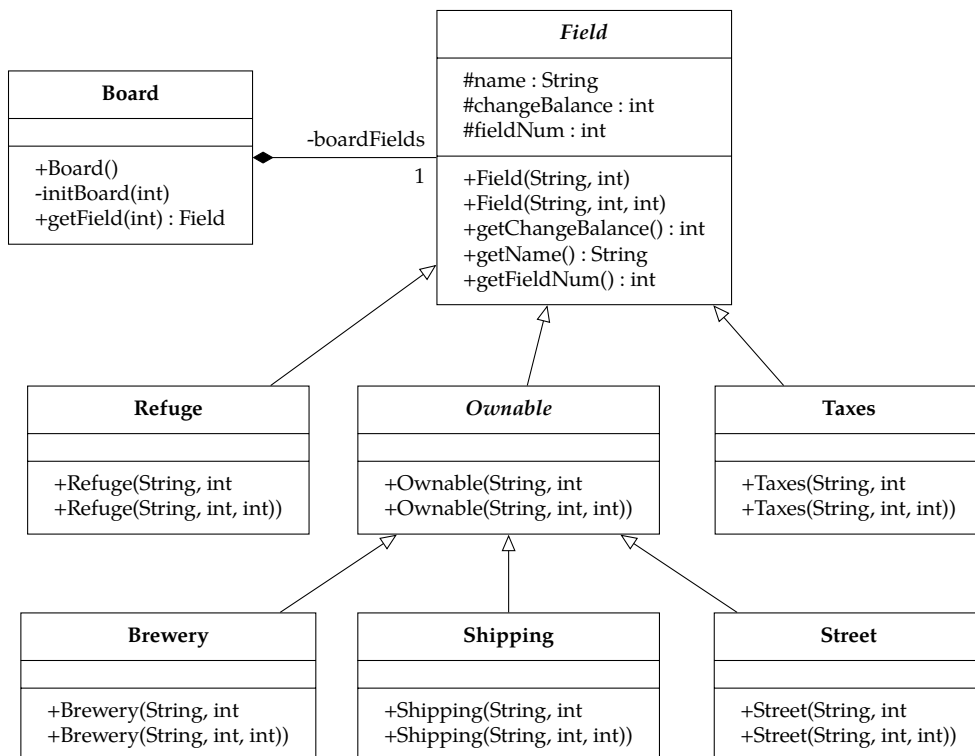
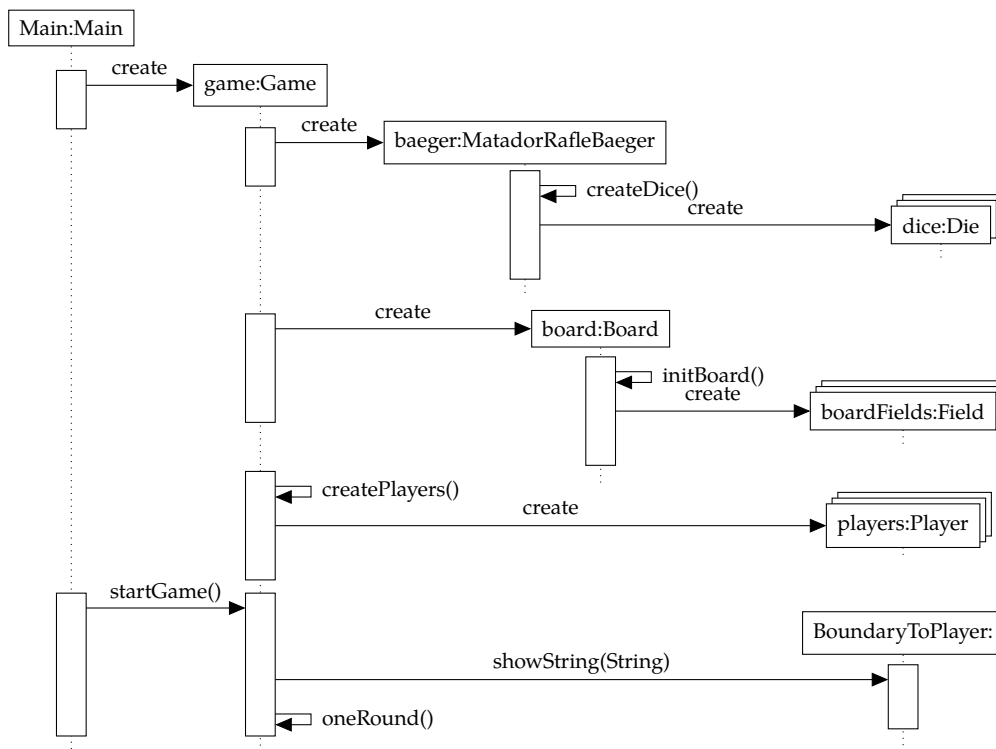
## Referencer

- [Lar04] Craig Larman. *Applying UML and Patterns. An Introduction to Object-Oriented Analysis and Design and Iterative Development*. 3. udg. Prentice Hall, 2004.
- [LL12] John Lewis og William Loftus. *Java Software Solutions Foundations of Program Design*. 7. udg. Pearson Education, 2012.
- [Nyb12] Mads Nybørd. *Del opgave 2. Anvendelse af klasser og relationer*. Vers. 2012-08-08. Opgavebeskrivelse udleveret i forbindelse med projektet. DTU, 8. aug. 2012.

## Rettelser

**Figur 3.3:** Designklassediagram der viser det meste af spillet. Field og dennes underklasser vises i figur 3.4 på næste side.



**Figur 3.4:** Designklassediagram der viser Board, Field og underklasser af Field.**Figur 3.5:** Sekvensdiagram som viser opstart af spillet.

Figur 3.6: Sekvensdiagram som viser en runde af spillet, MRB betyder MatadorRafleBaeger.

