TECHNISCH Ontwerp

Dungeon Crawler Game ‘Tomb of the Slime King’

Lodewijk Klaassen, Jasper Kooy & Patrick Ubbink

Studentnummers: 676682, 674152 & 669278  
Mitch Holewijn  
7 april 2022

Inhoud

[1. Inleiding 2](#_Toc100217866)

[2. KlassenDiagram 3](#_Toc100217867)

[2.1 Eerste versie 3](#_Toc100217868)

[2.2 Eind versie 4](#_Toc100217869)

[2.3 Wijzigingen ten opzichte van de eerste versie 4](#_Toc100217870)

[4. Conclusie 5](#_Toc100217871)

[5. Bronvermelding 6](#_Toc100217872)

# Inleiding

In dit document gaan we kijken naar de technische structuur van het spel. We gaan een klassendiagram bekijken die in kaart brengt wat de structuur van het spel is en hoe ieder onderdeel samenhangt met elkaar. Ook wordt er kort beschreven wat de wijzigingen waren die we hebben moeten maken aan het klassendiagram en waardoor dat is gekomen. Als laatste is er nog een conclusie en een bronvermelding.

# KlassenDiagram

## 2.1 Eerste versie

Diagram, engineering drawing

Description automatically generated

\*Zie bijlage “SlimeGame\_ClassDiagram\_first\_version.drawio.png”

## 2.2 Eind versie

Diagram, engineering drawing

Description automatically generated

\*Zie bijlage “SlimeGame\_ClassDiagram\_final.drawio.png”

## 2.3 Wijzigingen ten opzichte van de eerste versie

De voornaamste wijzigingen zijn het weghalen van onnodige variabelen, het toevoegen van meer buttons en tekst entities en het verplaatsen van functionaliteit naar het hoofd spel en scenes. Dit laatste was nodig door de meerdere obstakels die we tegenkwamen door het gebruik van de nieuwe game engine. Wij hadden nog te weinig kennis over wat er wel en niet mogelijk was en hebben daardoor functionaliteiten moeten verplaatsen en omwegen moeten bouwen. Dit, samen met het opschonen van de klassen van ongebruikte variabelen en functies en de nieuwe klassen die we toe hebben moeten voegen, heeft uiteindelijk geleden tot een complexere klassenstructuur en bijbehorend diagram. Ook is door voornamelijk het verplaatsen van functionaliteit jammergenoeg onze code wat ongestructureerder geworden.

# Conclusie

Bij het schrijven van de code zijn we erachter gekomen dat een aantal dingen niet zouden werken op de manier hoe wij in gedachten hadden. Hierdoor moesten wij een aantal functionaliteiten moeten verplaatsen wat verder benoemd is in punt 2.3. Desalniettemin hebben we toch best veel van onze ambitieuze game waar kunnen maken en ons op veel plekke toch aan ons originele klassendiagram kunnen houden.

# Bronvermelding

*Primitive Data Types.* Oracle.com. Laatst geraadpleegd op 6 april 2022 , van <https://docs.oracle.com/javase/tutorial/java/nutsandbolts/datatypes.html>

*Han-Yaeger Tutorial.* Han-yaeger.github.io. Laatst geraadpleegd op 5 april 2022, van

<https://han-yaeger.github.io/yaeger-tutorial/introduction.html>

*Han-Yager Book.* Han-yaeger.github.ino. Laatst geraadpleegd op 3 april 2022, van

<https://han-yaeger.github.io/yaeger/book/introduction.html>

*Han-Yaeger Github Repository.* Github.com. laatst geraadpleegd op 24 maart 2022, van

<https://github.com/han-yaeger/yaeger-tutorial>

*Javadocs.* www.jetbrains.com. Laatst geraadpleegd op 6 april 2022, van

<https://www.jetbrains.com/help/idea/working-with-code-documentation.html>