Technisch Verslag

Z-Rush



Door: Calvin Carter, Nicky Hagens,

Mark Verburg, Pim Goes

Versie: 1.0

Datum: 31-1-2021

Inhoud

[Doel Game 3](#_Toc63017523)

[Features 3](#_Toc63017524)

[Game Flow 4](#_Toc63017525)

[Klassen diagram 4](#_Toc63017526)

[Klassen 5](#_Toc63017527)

[Klasse MainMenu: 5](#_Toc63017528)

[Klasse GameOver: 5](#_Toc63017529)

[Klasse ScoresScreen: 5](#_Toc63017530)

[Klasse Player: 5](#_Toc63017531)

[Klasse Zombie: 5](#_Toc63017532)

[Klasse Static: 5](#_Toc63017533)

[Klasse Drawable: 5](#_Toc63017534)

[Klasse AssetManager: 5](#_Toc63017535)

[Klasse Game: 6](#_Toc63017536)

[Doxygen 7](#_Toc63017537)

# Doel Game

Het doel van de game is het zo lang mogelijk overleven van de zombies. De speler kan punten verdienen door “vijanden”(zombies) dood te schieten. Het uiteindelijke doel van de game is een zo hoog mogelijke score te behalen.

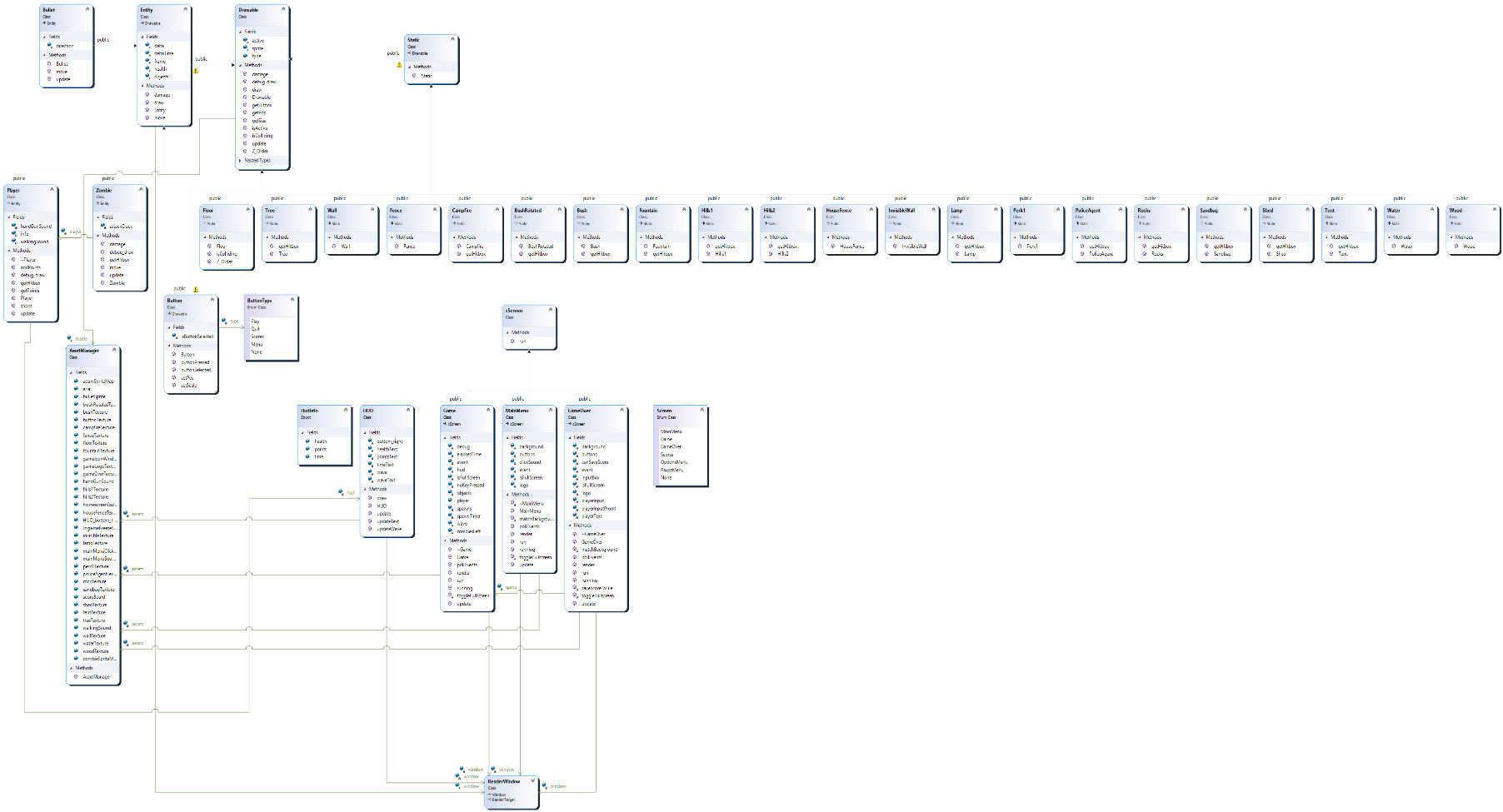
# Features

* Third person top down view
* 1 Enemy
* 1 Wapen
* Wave shooter
* Menu scherm
* Scores worden opgeslagen
* Scores inzien
* Top 8 scores lijst
* Geluidseffecten
* Hoofdmenu muziek
* Een HUD, met informatie van de huidige spel, zoals tijd, levens en wave-nummer.

# Game Flow

1. De speler start het spel, de speler komt in het hoofdmenu van de game. De speler heeft de mogelijkheid om 1. Het spel kan starten. 2. De scores bekijken. 3. Het spel afsluiten.
2. De speler start het spel
   1. De speler wordt in de map gezet. En er spawnen zombies.
   2. De speler moet de zombies dood maken, de speler krijgt per geraakte shot. Een bepaald aantal punten. En als de zombie doodgaat krijgt de punt een bepaald aantal punten.
   3. Als de speler doodgaat, komt de game over scherm tevoorschijn.
3. Het spel start, de speler start linksonder in de map.
4. Speler krijgt een standaard pistool.
5. Speler moet alle vijanden dood maken.
6. De speler dood alle vijanden in de wave.
   1. De speler gaat dood. De game is afgelopen
7. De nieuwe wave start
8. Er komt een nieuwe “wave” van vijanden.
9. Stap 5,6,7 wordt herhaald totdat de speler doodgaat.
10. Eindscherm

# Klassen diagram



Het klasse diagram is ook te vinden in onze GitHub repository onder de folder documents.

# Klassen

## Klasse MainMenu:

De klasse MainMenu laat het start scherm zien. Dit werkt in samenwerking met de code in de main.cpp. Als de functie Run wordt aangeroepen, dan wordt in het geval van de MainMenu het hoofdmenu laten zien. Als de functie een 0 (of lager getal) teruggeeft, wordt het gehele spel afgesloten. Als de run functie een andere getal returned, dan wordt het spel gestart (zie klasse game, de functie run).

## Klasse GameOver:

De klasse GameOver wordt geactiveerd als de speler 0 levens heeft (wat veroorzaakt wordt als de zombies de speler “dood” maken). De speler kan dan zijn naam invullen om zijn score op te slaan (score van 0 wordt niet opgeslagen). De score wordt opgeslagen in het bestand: “scores.json”.

## Klasse ScoresScreen:

De klasse ScoreScreen laat de top 8 scores zien ingeladen van “scores.json” bestand. Alle scores worden in dat bestand in een json array opgeslagen (wordt gedaan in de klasse: GameOver, als de speler doodgaat). De library die gebruikt wordt om de scores te serialiseren en deserialiseren is: [nlohmann/json](https://github.com/nlohmann/json).

## Klasse Player:

De speler klasse laat de speler in het spel zien. De speler verstuurd ook elke frame de data naar de HUD-klasse om de health etc. op het scherm te tonen. Hier wordt de sprite animatie geregeld en bepaalt welke kant de speler opkijkt aan de hand van de muis positie.

## Klasse Zombie:

In de zombie klasse wordt vooral het updaten van de zombie gereld. In de **update** functie wordt de positie elke frame geüpdatet zodat de zombie naar de speler toe loopt, ook worden de sprite animatie hier geregeld en de kijkrichting van de sprite aan de hand van de directie dat de zombie op loopt. Hier bevindt zich ook de health van de zombie een functie **damage** die bepaalt wat er gebeurt zodra de zombie geraakt wordt door een kogel.

## Klasse Static:

Deze klasse is niet zo bijzonder, wordt gebruikt voor alle objecten die niet geüpdatet hoeft te worden, en dus ook nooit zou bewegen. Het enigste wat deze klasse doet is een texture inladen en toe voegen aan een sprite zodat deze in het spel zichtbaar zal zijn.

## Klasse Drawable:

Dit is de hoofdklasse van alle tekenbare objecten op het scherm. Alle objecten die te zien zijn op het scherm hebben een aantal functies gemeen, namelijk: **draw**, **debug\_draw**, **isColliding**, **isActive**, **getPos**, **getSize** en **getHitbox**. Hierdoor kunnen we gemakkelijk met een enkele lijst en loop voor collisies checken, de entiteiten tekenen en bepalen op welke volgorde ze getekend worden zodat de illusie ontstaat dat objecten voor of achter elkaar staan.

## Klasse AssetManager:

De Asset manager, heeft de verantwoordelijkheid om alle textures, fonts en afbeeldingen in te laden. Deze klasse wordt aan elke klasse die textures nodig heeft als reference gegeven zo dat de desbetreffende klasse bij de textures kan komen en zo bij de juiste bestanden te komen.

## Klasse Game:

Game, deze klasse knoopt alles aan elkaar. Hier bevindt het systeem waarmee de rondes worden geregeld, het inspawnen van de zombies en de speler input. Game bevat een lijst van alle entiteiten, deze worden elke frame hier geüpdatet en getekend.

# Doxygen

De Doxygen is te generen door in de GitHub folder het programma Doxygen te runnen.

[Doxygen Download](https://www.doxygen.nl/download.html)

[Doxygen gebruik](https://www.doxygen.nl/manual/doxygen_usage.html)