

# Documentatie Race Game v1.0

---

*Gemaakt door: Emiel de Brouwer & Robbert Goey*

## Gebruiksaanwijzing 3D Race Game

Gebruik de vierpunddruktoetsen op het toetsenbord om de auto te bewegen: om de auto naar voren te laten rijden, druk op “omhoog”, voor het sturen druk op “links” of “rechts” en om achteruit te gaan, druk op “onder”.

## Requirements

De volgende requirements zijn in de game geïmplementeerd:

- Er is een auto met wielen
- De wielen draaien conform de snelheid van de auto
- De auto volgt een eenvoudig parkours
- De tijd per ronde wordt aangegeven
- Er is een HUD dat de verstreken tijd en de huidige snelheid laat zien
- Er wordt gebruik gemaakt van modellen
- Er wordt textuur gebruikt op de modellen
- Er is een skybox in de game
- Er is geluid bij de volgende evenementen:
  - o De auto staat stil
  - o De auto rijdt
  - o De auto rijdt uit de wereld

## Engine

De engine is compleet overgenomen van “Micro Game Engine with HUD”. Daarin werd een wereld aangemaakt met twee hoofden van apen die geroteerd konden worden, terwijl ook de camera van de wereld aangepast kon worden d.m.v. toetsen.

Met dit principe hebben we de engine aangepast, zodat het meer een racegame is geworden. In de volgende hoofdstukken wordt uitgelegd welke onderdelen zijn toegevoegd/aangepast/verwijderd zijn.

## Game

In de Game.cpp zijn de belangrijkste aanpassingen gebeurd.

De hoofden van apen zijn vervangen door modellen van een auto en wielen. Om ervoor te zorgen dat de wielen kunnen roteren, hebben we twee aparte modellen gemaakt: de auto en de wielen. De wielen zijn twee keer aangemaakt in GameObject: twee voorwielen en twee achterwielen. Deze bewegingen afhankelijk van de snelheid van de auto. Als je snelheid naar voren hebt zullen de wielen naar voren draaien, bij snelheid naar achteren draaien de wielen ook achteruit.

Ook is er een start/finish en een checkpoint aangemaakt als GameObject. Deze GameObjecten hebben geen mesh gekregen en zijn dus niet zichtbaar. Ze hebben wel een collider. En bij een collision met de start/finish wordt de tijd onthouden. Dit wordt niet elke keer zomaar gedaan, het moet of de eerste keer zijn (dan wordt de starttijd opgeslagen) of er moet eerst het checkpoint gepasseerd zijn dan wordt de rondetijd berekent en opgeslagen.

Het plaatje van de vloer is vervangen met een plaatje van een racebaan.

Ook hier wordt de camera aangemaakt. We hebben ervoor gekozen om de camera op een vast standpunt te plaatsen, zodat het lijkt alsof je het spel speelt in een semi-topdown view.

De skybox wordt ook hier aangemaakt.

## HUD

In de HUD zijn de volgende elementen opgenomen:

- FPS
- Stuurrichting van de auto
- Snelheid van de auto
- De tijd voor de huidige ronde
- Rondetijden van afgelegde ronden
- Of de auto binnen de wereld rijdt of niet

## RaceCar

De RaceCar is een klasse die erft van GameObject. Deze klasse heeft naast alles wat het van GameObject erft nog een snelheid en een stuurrichting en methoden om deze te beïnvloeden.

## KeysBehaviour

In deze klasse zit de besturing van de auto. In deze klasse wordt de auto ook verplaatst volgens de stuurrichting en snelheid die de auto op dat moment heeft. En verder wordt in deze klasse geluid afgepeeld met behulp van SFML om de auto meer realisme te geven.

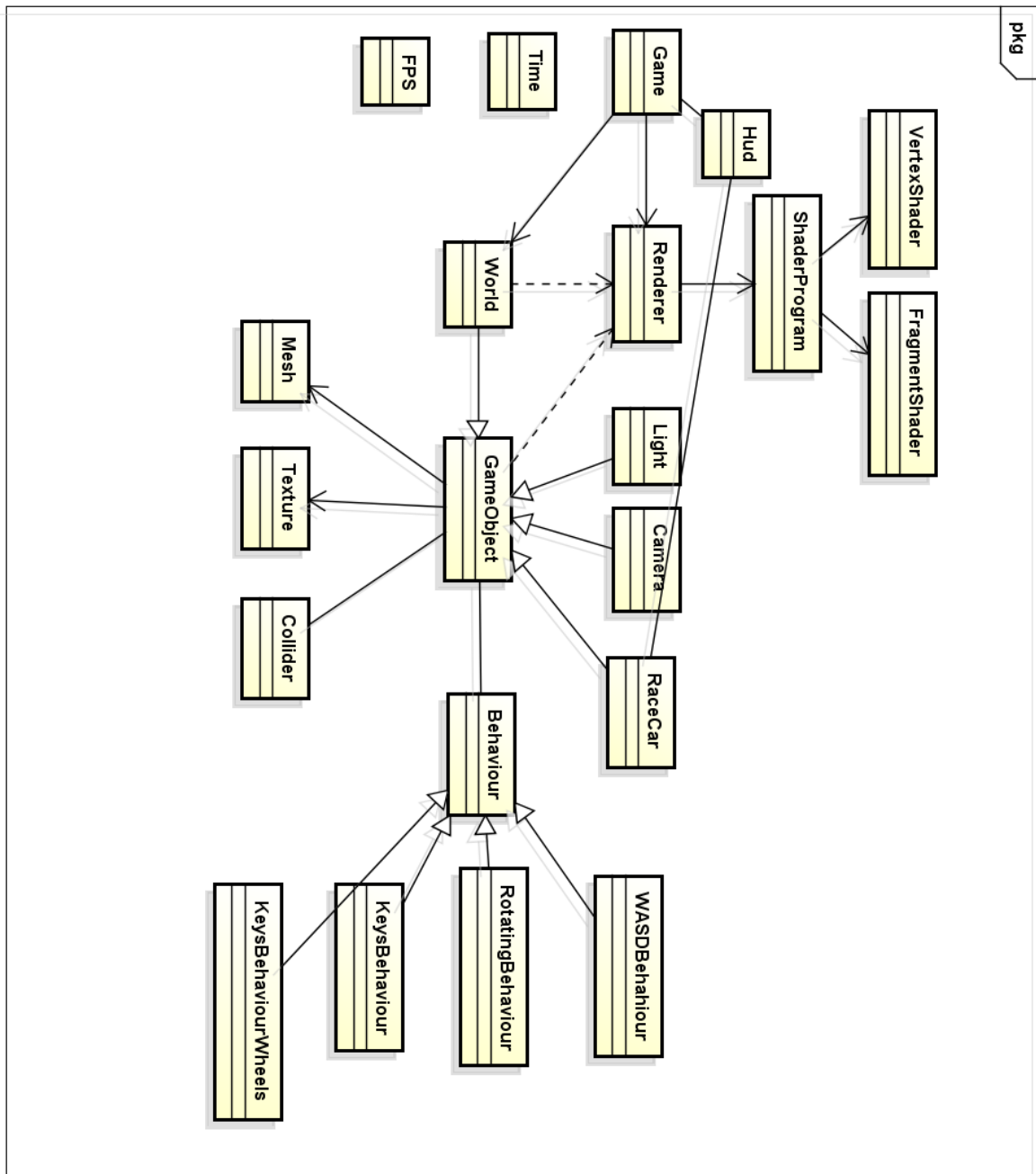
## KeysBehaviourWheels

Deze klasse krijgt de wielen en de auto mee. Vervolgens zorgt deze klasse ervoor dat de wielen rondraaien afhankelijk van de snelheid van de auto.

## Skybox

De skybox is een kubus die om het speelveld heen zit. Deze kubus heeft de normalen naar binnen staan. Waardoor textures aan de binnenkant van de kubus getekend worden. Dit geeft het effect dat de wereld groter is dan het in daadwerkelijkheid is.

# Klassendiagram



Objectdiagram

