

Designdokument: Post Human Evolution

Name: Marius Kögel
Matrikelnummer: 262607
Studiengang: MIB
Veranstaltung: Prototyping interaktiver Medien-Apss und Games



Overview

Das Spiel ist ein 2D Shooter mit Topdown View. Es spielt in einer Postapokalyptischen Welt, die von Zombies geplagt wird. Als einer der letzten noch verbleibenden Menschen ist es das Ziel des Spielers trotz aller Gefahren zu überleben. Um das zu erreichen muss er sich durch endlose Gegnerwellen kämpfen.

Gameplay

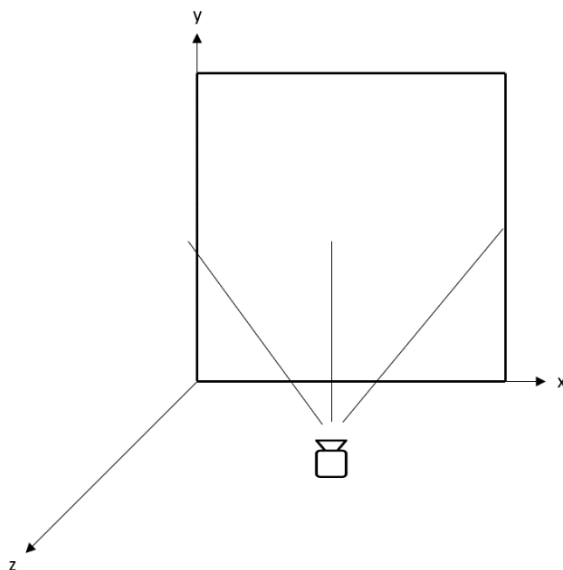
Der Spieler startet auf einer rechteckigen Map auf der sich mehrere Feinde befinden. Das Ziel ist es die Feinde so schnell wie möglich auszuschalten, ohne dabei zu sterben. Es werden während des Spielverlaufs immer mehr Gegner auf der Map erscheinen. Das Zeitintervall hierfür wird, bis es seinen Minimalwert erreicht, mit jedem Durchlauf kürzer, d.h. nach immer kürzerer Zeit erscheinen neue Gegner. Sobald ein Gegner den Spieler erfasst verfolgt er diesen. Ist er unmittelbar vor ihm greift er an und der Spieler verliert somit Leben.

Der Spieler bewegt sich mittels der Tasten „W, A, S, D“. Die Schussrichtung wird über die Maus gesteuert. Mit der linken Maustaste wird geschossen. Die Waffe lädt er mit „R“ nach. Außerdem verfügt er über einen Dash, den er mit der Umschalt-Taste bedienen kann. Der Dash hat zudem einen kurzen Cooldown. Jedesmal wenn der Spieler einen Gegner besiegt steigt der Score an. Stirbt der Spieler wird das Level neu geladen und der Highscore aktualisiert.

Map und Kamerapositionierung

Die Map auf der sich der Spieler und die Gegner bewegen ist rechteckig und von Wänden umschlossen, die der Spieler nicht verlassen kann. Sie ist mittig in den Weltkoordinaten (0/0/0) positioniert.

Wie man auf der nächsten Abbildung erkennt befindet sich die Map auf der X-Y-Ebene. Die Kamera befindet sich entlang der Z-Achse und schaut auf die X-Y-Ebene.



Anforderungstabelle

Nr	Bezeichnung	Inhalt
	Titel	Post Human Evolution
	Name	Marius Kögel
	Matrikelnummer	262607
1	Nutzerinteraktion	Das Spiel wird über einen Mausklick gestartet. Während des Spiels kann der Spieler wie folgt gesteuert werden: Bewegung: WASD, Schussrichtung: Maus, Schießen: linke Maustaste, Nachladen: R, Dash: Umschalt- / Shift-Taste.
2	Objektinteraktion	Eine Kollisionprüfung findet zwischen folgenden Objekten statt: Spieler und Wände: der Spieler kann sich nicht durch Wände bewegen. Spieler und Gegner: Gegner greifen an wenn eine Kollision erkannt wurde. Bullet und Gegner: Schusskollision.
3	Objektanzahl variabel	Gegner werden nach Ablauf eines variablen Zeitintervalls erstellt. Der Zeitintervall hierfür wird immer kürzer, bis er seinen Minimalwert erreicht. Außerdem werden beim Schießen Kugeln/Bullets erstellt und in Schussrichtung verschoben.
4	Szenenhierarchie	An die root wird das level gehängt. Alle Objekte der Spielwelt sind children des level. Somit muss beim Neustart des Spiels lediglich das level von der root entfernt werden und anschließend wieder ein neues level an die root gehängt werden. Außerdem ist die Kamera ein child des Avatars und folgt diesem so.
5	Sound	Das Spiel verfügt über einen Soundtrack, der dauerhaft ab Programmstart wiedergegeben wird. Startet der Spieler das Spiel im Startscreen wird zudem ein Ambiencesound wiedergegeben, der die Atmosphäre des Spiels unterstützt. Bei verschiedenen Interaktionen werden zudem Geräusche abgespielt, wie z.B. dem Schießen, dem Nachladen, oder beim Angriff eines Gegners.
6	GUI	Der Spieler kann das Spiel bei Programmstart mit Mausklick starten. Während des Spiels werden Spielerwerte, wie Gesundheit, Score, Ammo und Highscore angezeigt.
7	Externe Daten	Der Spieler hat die Möglichkeit die Schwierigkeitsstufen einzustellen.
8	Verhaltensklassen	Die verschiedenen Klassen haben eigene Methoden, wie z.B. move, rotate und shoot beim Avatar. Weitere Klassen mit individuellen Methoden sind z.B. Enemy und Bullet.
9	Subklassen	Die Klasse GameObject dient als Oberklasse der meisten Spielobjekte. Eine Unterklasse davon ist die Klasse Moveable, zu der Objekte gehören, die sich bewegen können (z.B. Avatar, Enemy und Bullet). In der Klasse Moveable ist die Methode checkCollision mit derer die Kollision der jeweiligen Objekte überprüft werden kann.
10	Maße & Positionen	Der Spieler hat die Ausgangsgröße (1/1/2). D.h. er ist ca. 2 Meter groß. Davon abgeleitet sind alle anderen Größen der Objekte. Die Spielwelt befindet sich mittig auf der Position (0/0/0) und reicht horizontal von -18 bis 18 und vertikal von 10 bis -10.
11	Event-System	Über das Event-System wird der Avatar gesteuert. So wird z.B. die Mausbewegung verfolgt und der Methode hndMouse übergeben. Diese rechnet die Bildschirmkoordinaten dann zunächst in Weltkoordinaten um und übergibt sie der Methode rotateTo der Klasse Avatar. Dort erfolgt die Berechnung des notwendigen Drehwinkels und der Drehrichtung, um den Avatar in Mauseichtung schauen zu lassen.

Gameplay Bilder

