

1. Schritt Basic 3D Setup

Als erstes habe ich ein grundlegendes 3D-Environment geschaffen, noch ohne Texturen und Details. Wichtig waren hierbei folgende Elemente: Der Raum selbst inklusive Collider, 3 bewegbare Türen im Boden für die Planeten sowie drei Spheres inklusive Podeste. Die Spheres haben Spherecollider und Rigidbodies.

2. Schritt Function-List

Danach habe ich mir eine Liste erstellt, wo ich alle wichtigen Punkte, die mit Animations oder Scripts gelöst werden müssen, aufgezählt habe. Die Liste ist wie folgt: VR-Player-rig mit Debugmodus, Interactable Spheres, Sphere-Hover, Sphere-Door-Trigger mit Destroy, Sphere-Counter, Timer mit Wandanzeige, Win und Lose mit Timer/Sphere Verkoppelung.

3. Schritt Steam-VR Import

Bevor ich zu Coden begonnen habe, musste ich noch Steam-VR aus dem Assetstore importieren und den Player in mein Environment platzieren. An diesem Punkt habe ich von meinem Macbook auf meinen Windows-Standpc gewechselt, da am Macbook keine Verbindung zum Assetstore möglich war. Am Standpc hat dann alles funktioniert.

4. Schritt Script-Aufteilung

Ich habe folgende Scripte erstellt: Ballhoverscript, Doorscript und Gamemangerscript. Das Ballhoverscript regelt das Schweben der Planeten am Podest, sowie die Aufhebung der Upforce wenn die Sphere entnommen wird. Das Doorscript regelt die Interaktion zwischen Sphere und Doors und der Gamemanager führt alle Werte zusammen und regelt zusammen mit dem Timer die Win oder Lose Situation.

5.Schritt Ballhoverscript

Zuerst habe ich die Variablen definiert. Danach rechne ich Kraft gegen die Schwerkraft, um die Planeten schweben zu lassen.

6.Schritt Doorscript

Zuerst habe ich wie immer die Variablen definiert. Danach regelte ich die Tags für die Planeten/Türen. Nur wenn der richtige Planet die richtige Tür berührt sollte die zuvor gemachte Türanimation ausgeführt, der Planet zerstört und der Punkt gezählt werden. Am Ende machte ich die im vorherigen Satz genannten Funktionen. Nun musste ich nur noch das Script auf die Objekte legen und die richtigen Tags verwenden. Ich führte einen Test durch und es funktionierte (überraschender Weise).

7.Schritt Gamemanager

Der Gamemanager hat zwei Aufgaben: Die Planeten, welche die richtigen Türen berührt haben, zählen und den Timer ausführen. Sind alle drei Planeten in der richtigen Tür, hat man gewonnen und wenn der Timer

vorher abläuft hat man verloren. Der Timer startet beim Drücken von Play und läuft für 1 Minute. Das Format des Timers habe ich von Leon, um Zeit zu sparen (kompliziert).

8. Schritt Planeten und Türen Texturieren

Im 8ten und letzten Schritt habe ich die Türen und die Planeten texturiert. Es war wichtig, dass die Planeten texturiert sind, damit ich die Aufgabenstellung „Es muss ein -Globus- vorkommen“ erfülle. Die Türen mussten ebenfalls texturiert werden, damit man weiß welcher Planet zu welcher Tür gehört.

Problematiken

Zu Beginn stieß ich wie bereits zuvor erläutert auf das Problem mit dem Assetstore auf meinem Macbook. Der Assetstore lud nicht, deshalb wechselte ich das Betriebssystem. Danach war das Problem, das ich nicht der beste Programmierer bin, das heißt ich musste mir hin und wieder Hilfe auf Youtube suchen oder Klassenkollegen fragen. Nach einigen Tagen Arbeit hatte ich endlich den vorliegenden Prototypen fertig, mehr Zeit konnte und wollte ich neben der Diplomarbeit nicht hineinstecken. Ich bin dennoch zufrieden mit meinem Projekt.