

KGDEV1 - Reinier Maartense 3028323

Datum: 19-09-2019

Avalanche 3D:

Dit spel - Avalanche in 3D

<https://spele.nl/klim-omhoog-spel/>

Er vallen random blokken naar beneden en je moet de toren die hieruit ontstaat beklimmen.

Dit terwijl er lava druk creëert waardoor je snel moet blijven klimmen.

er is een highscore voor de hoogte die je bereikt.

Controls:

WASD/Arrow Keys - Lopen

Spacebar - springen

Spacebar wanneer tegen blok aan en in de lucht - wall jump

UML & Activity Diagram(2 tabs):

https://drive.google.com/file/d/1U1KmpHL_iIFPoKDTcYjA9CTqzrAymMgd/view?usp=sharing

Uitleg code

Mijn GameManager is momenteel vooral een plek om belangrijke references en delegates op te slaan. Ik heb geëxperimenteerd met delegates en het delen van references.

De InstanceManager was voor mij een experiment met delegates. Het idee hiervan is dat ik van alle scripts die ik vaak nodig heb hierin een instance kan maken met een enkele regel code. Vervolgens kan ik via de InstanceManager ook met een enkele regel code deze instance ophalen.

Mijn ObjectPooler had eerst een string class pool die je op het GameObject aan moest maken. Dit was niet heel erg failsafe en slecht compatible met version control. Hierom heb ik ervoor gekozen het met Scriptable Objects op te lossen. Er word voor elk ScriptablePool object in de resources folder automatisch een pool gemaakt.

Verder wilde ik ook van tags af in mijn project. Dit heb ik gedaan door interfaces te gebruiken.

Op elke plek waar ik voorheen een zou gebruiken, gebruik ik nu een interface.

In de HighscoreManager heb ik voor het eerst met System.Linq gewerkt. Voordat ik de lijst met scores weergeef sorteert ik deze eerst met de Sort functie van Linq

Ook tijdens het opslaan en laden gebruik ik een linq functie om de oude en nieuwe list te me