

Amber Meijers

Media Designer. Traditional Artist.

Open Innovation
Unity



Contents

3. *Documentbeschrijving*

4. *Opzet*

5. *Physics + Colliders*

6. *Animeren + Coderen*

7. *Bronnen*

*Document***beschrijving**

In dit document heb ik mijn reis door 2D animatie in Unity van begin tot eind gedocumenteerd. Ik ben begonnen als leek met geen enkele ervaring en kan inmiddels zelf voor een karakteranimatie zorgen die veranderd wanneer er een interactie plaats vindt.



Test 1

Voor mijn eerste POC heb ik gebruik gemaakt van bestaande sprites.

Het eerste keuze die ik heb gemaakt was de keuze tussen skeletal animation rigs of frame by frame animaties⁰. Bij een Skeletal Rig maak je letterlijk een skelet waar lichaamsdelen aan vast zitten. Het skelet is relatief moeilijk te maken maar als het eenmaal in elkaar zit, is het makkelijk en snel te gebruiken voor animaties. Daarnaast is er ook frame by frame animatie, waar je elk onderdeel van de animatie los tekent. Omdat ik graag pixel art maak en ook al een beetje had ge-experimenteerd met animatie in Aseprite, heb ik gekozen om eerst aan de slag te gaan met 2D animatie op een frame by frame basis.

Omdat ik geen enkele ervaring met Unity had, liep ik tijdens de eerste 2D animatie tutorial al tegen het probleem aan dat ik, toen ik een asset pack had gedownload, geen enkel idee had van het installeren en gebruiken van dit asset pack. De eerste stap die ik dus moest zetten was het kijken van een tutorial over asset packs en hoe ik ze kon installeren¹. Vervolgens heb ik een tutorial gekeken waarin uitgelegd werd hoe ik assets aan mijn Scene kon toevoegen². Daarnaast heb ik een aantal keer advies opgezocht over kleine problemen waar ik tegenaan liep³.

Ten slotte is het laatste dat ik moet klaar maken het karakter zelf⁴. Omdat de tutorial gebruik maakte van een volledig spritesheet i.p.v. losse afbeeldingen, wilde ik (voor mijn eerste test) dit idee volgen. Ik heb dit spritesheet van Shovel Knight gedownload.

Met de automatische slice tool van Unity had ik meteen een serie losse afbeeldingen in mijn assets staan. Op dit moment ziet het spel er uit zoals de foto links bovenin, maar als je het 'speelt' gebeurt er niets.

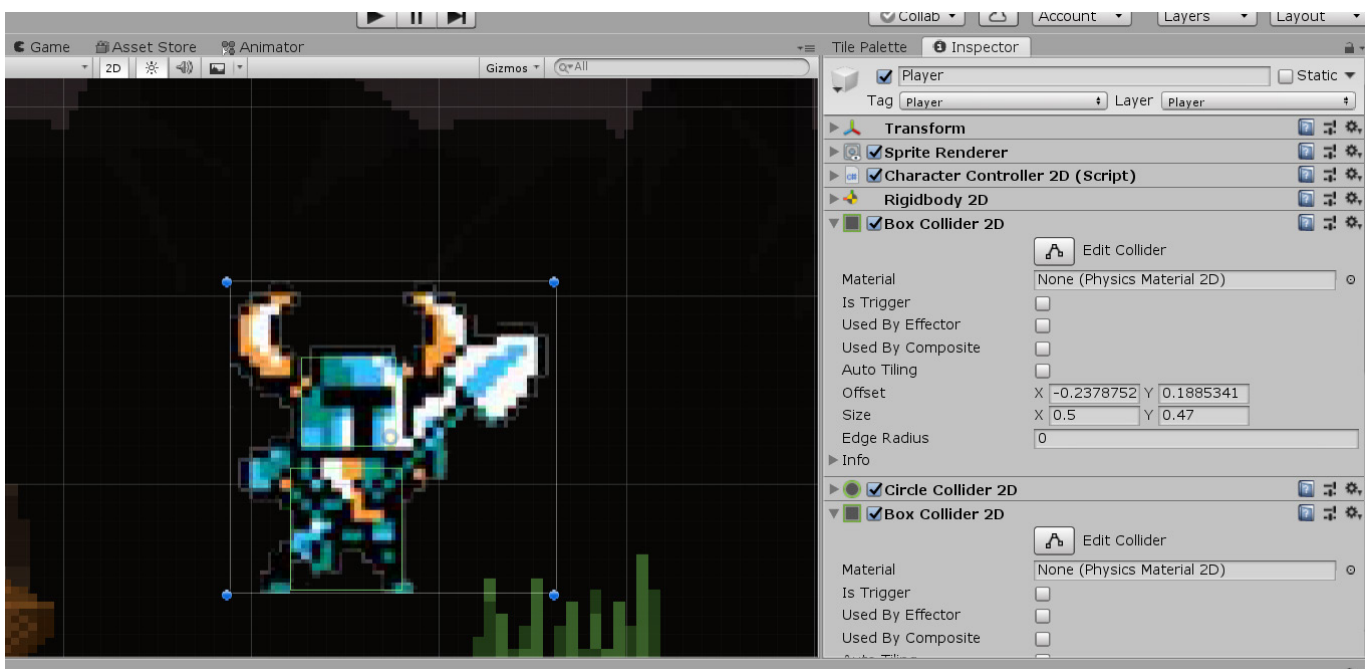
De volgende stap is het toevoegen van physics aan mijn karakter en de achtergrond.



Physics+**Colliders**

Op dit moment heeft het karakter geen enkele interactie met zijn omgeving. Hij kan niet bewogen worden en zelfs zwaartekracht bestaat niet. Zelfs al zou er zwaartekracht zijn, de ondergrond is geen vast object en het karakter zou er dus zo doorheen vallen. Daarom moet ik eerst een aantal Components toevoegen. Unity heeft een hoop van zulke 'Components' al in het programma zelf staat. Het zijn onderdelen die je toe kunt voegen aan je objecten. In principe zijn het methodes en parameters die meegegeven worden aan het object.⁵

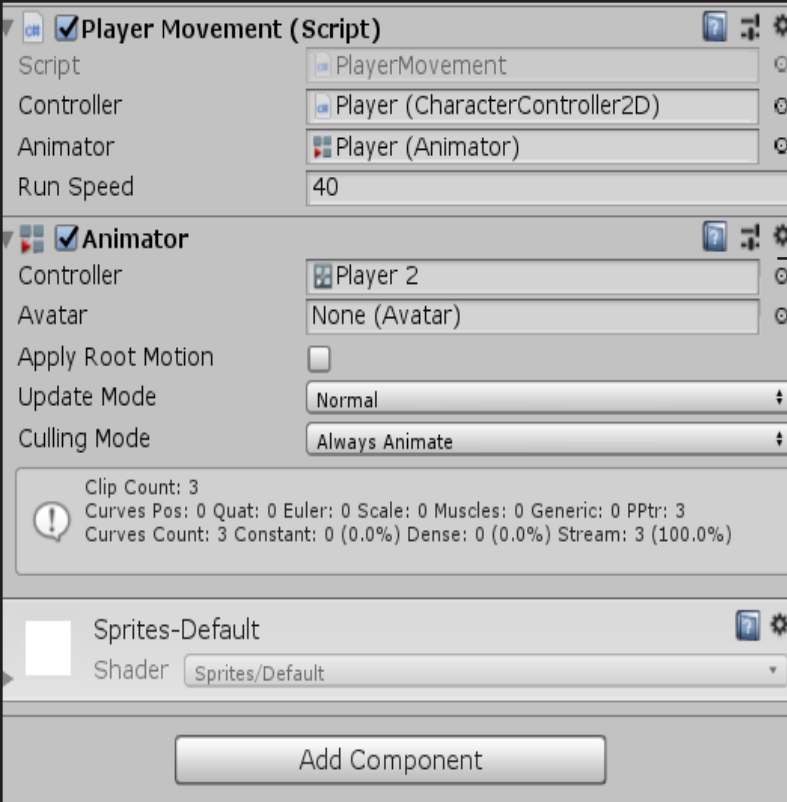
Als eerste geef ik het karakter een aantal 'Colliders' mee, box en circle om de juiste vorm te krijgen die binnen de sprite past. Deze colliders zorgen ervoor dat het karakter straks op de grond kan lopen en tegen de muren kan springen. Daarnaast geef ik het een Rigidbody mee, het onderdeel waar ik de massa en zwaartekracht voor dit object mee kan bepalen. Als ik het spel nu start, valt mijn shovel knight door de grond heen, want er is nog geen grond. Ik geef dus een box collider aan de grond mee en een edge collider aan de buitenrand van het spel zodat hij er niet af kan vallen, behalven in het gat dat ik expres in het spel heb gelaten.



Nu kan ik animatie toevoegen. Wanneer ik mijn karakter 'Player' selecteer, kan ik een animatie toevoegen onder het window 'Animation'. Deze animatie sla ik op met een relevante naam. Voor een idle animatie kies ik player_idle en voor rennen noem ik ze player_run. Vervolgens kan ik alle sprites voor de animatie die ik wil uitvoeren in een tijdlijn zetten en aangeven hoe lang de animatie moet spelen. Dit process herhaal ik voor elke animatie die ik wil aanmaken.

In het volgende onderdeel breng ik een stukje User Interaction aan en animaties die hieraan verbonden zijn.





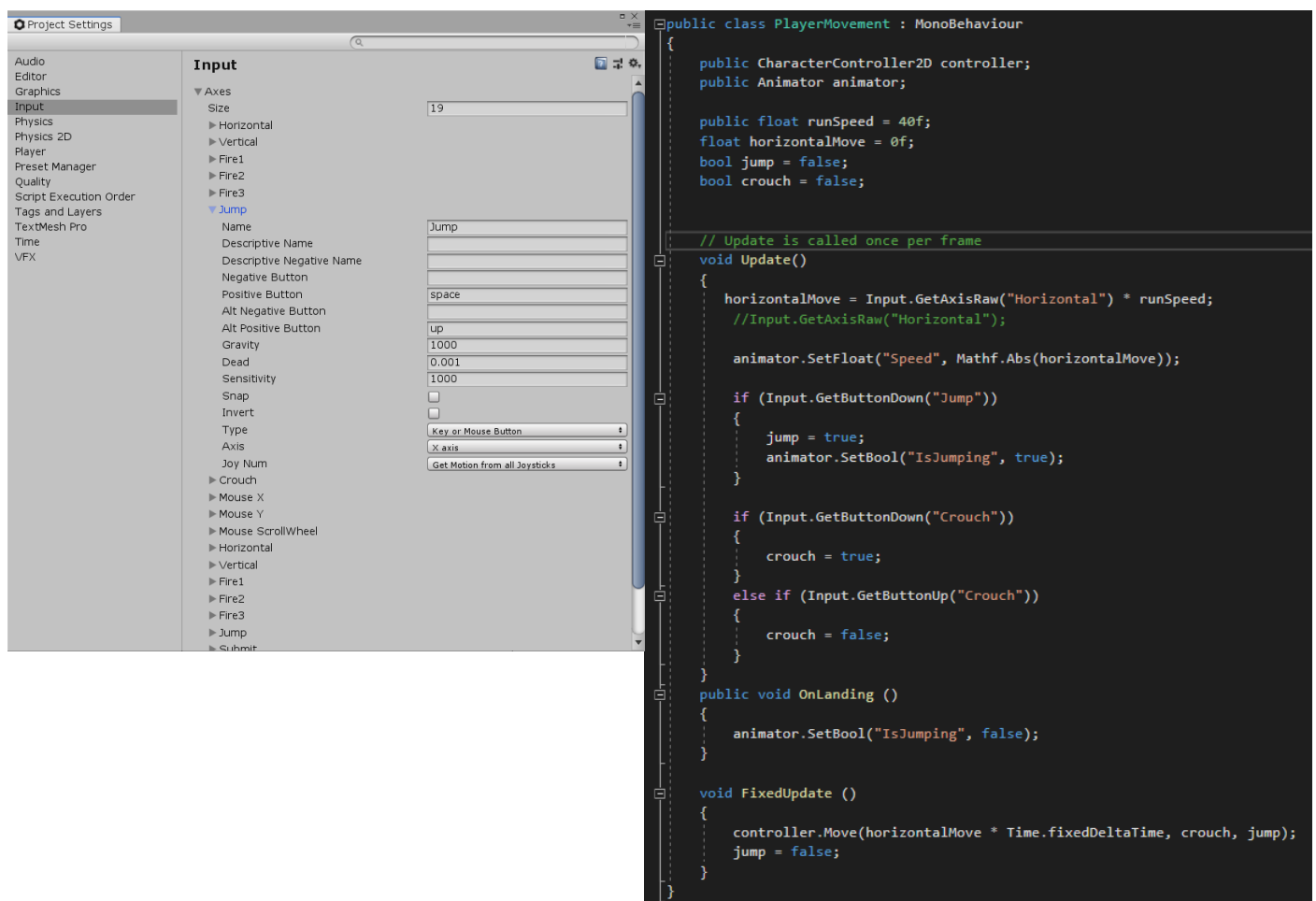
Animeren + Code

Player Movement + Animator

Deze twee components werken samen om onze speler te laten lopen wanneer wij willen dat hij loopt en hierbij een animatie te spelen.

Omdat er geen Player Movement Component bestond, moest ik zelf een stukje code schrijven, ondersteund door bestaande classes en methodes die al bestaan in Unity zelf en een aantal die ge-upload waren op Github.

<https://www.youtube.com/watch?v=4nppyg7DNUe>



Bronnen

⁰ <https://assetstore.unity.com/packages/essentials/unity-anima2d-79840> - Skeletal Rig 2D Animating

¹ <https://www.youtube.com/watch?v=VUhqrxH0aE> - Introduction to Unity - Importing Assets and Scene Setup.

² <https://www.youtube.com/watch?v=OFOHGjrxhNA> - Unity 2D Game Design Tutorial - Adding Assets to your scene.

³ <https://forum.unity.com/threads/2d-sprite-always-behind-background.421276/> - Background layering forum post

Omdat ik mijn achtergrond niet op de juiste laag kreeg.

⁴ <https://www.youtube.com/watch?v=m-J5sCRipA0> - How to make a 2D platformer - Character.

⁵ <https://www.youtube.com/watch?v=wDR6jvRuLEA> - Unity 2D Game Design Tutorial - Adding physics and colliders to objects.

<https://www.youtube.com/watch?v=hkaysu1Z-n8> -

2D Animation in Unity

https://assetstore.unity.com/packages/2d/characters/sunny-land-103349?aid=1101lPG-j&cid=1101l7rjFhxy&utm_source=aff

Voorbeeld karakter

<https://www.youtube.com/watch?v=dwcT-Dch0bA>

2D Movement in Unity

<https://github.com/Brackeys/2D-Character-Controller> - 2D character controller

Onderdeel van de bovenstaande tutorial

<https://docs.unity3d.com/ScriptReference/Physics2D.CircleCast.html> - Circlecasting

<https://puu.sh/EXKEE/21e4920999.gif>

Spel Eyeball

Spel Shovelknight

<https://puu.sh/EXLFa/e862130b26.gif>