

# Animaties

Jordy Teunis

## Inhoud

Inleiding.....	1
Methodes .....	2
Uitvoering .....	2
1. Onderzoek naar animaties in Unity .....	2
• Verschillende animatoren.....	2
• Animaties gebruiken in Unity .....	4
2. Issue bord in GitLab.....	6
3. Animaties in het project .....	7
• Speler animatie tree .....	7
• Animaties activeren via code .....	8
Reflectie .....	8
Bronnen .....	9

## Inleiding

Het doel van deze sprint was om animaties te maken voor zowel de vijanden als de speler in de game. Hiervoor hebben we gebruikgemaakt van gratis assets van verschillende websites, Mixamo om animaties te maken, en Unity om ze werkend te krijgen. Tijdens deze sprint hebben we gekeken hoe we animaties het beste konden maken en toepassen, en deze uiteindelijk toegevoegd aan het project. Dit zorgt ervoor dat de bewegingen van de speler en vijanden er realistischer en soepeler uitzien, wat de game een stuk beter maakt.

# Methodes

1. Analyseren: Onderzoek naar animaties in Unity
  - Type: Literatuur- en audiovisuele onderzoek.
  - Beschrijving: Onderzoeken hoe je animaties in Unity toepast en deze animaties verandert wanneer de speler of vijand rent of stilstaat.
  - Bronnen: Unity Documentation en YouTube
2. Ontwerpen: Issue bord in GitLab
  - Type: Regels issue voor project
  - Beschrijving: De regels uitleggen van de issues die we voor het project hanteren.
3. Realiseren: Animaties in het project
  - Type: Animaties integreren in het project.
  - Beschrijving: Animaties ontwerpen en toevoegen aan het project en de animaties bruikbaar integreren in de game.

## Uitvoering

### 1. Onderzoek naar animaties in Unity

- **Verschillende animatoren**

Er zijn verschillende soorten animatoren die te gebruiken zijn voor Unity, je kan natuurlijk Unity zelf gebruiken als animator voor je 3d modellen, maar een software zoals Blender is gemaakt om 3d modellen aan te passen en te animeren. Je hebt meerdere alternatieven voor Blender, maar Blender is het meest gedocumenteerde Modelleringssoftware door gebruikers, hierdoor is er vaak makkelijk een oplossing te vinden voor de problemen en heb ik daarom ook Blender gebruikt voor mijn project. Er is ook Mixamo die gemaakte animaties op een 3d model voor je zetten,

- *Unity*

Unity biedt basisfunctionaliteiten voor het creëren van animaties via de Animation Window, waarmee je keyframes kunt instellen en eenvoudige bewegingen direct in Unity kunt animeren. Dit is handig voor snelle prototypes of het animeren van eenvoudige objecten, zoals deuren die openen of lichten die flikkeren. Echter, het maken van complexe en vloeiende animaties, zoals karakterbewegingen of cinematografische scènes, is in Unity beperkt. Voor meer gedetailleerde en realistische animaties wordt vaak Blender gebruikt.

- *Blender*

Blender is een open source 3d modellering software die door vele gebruikt wordt om 3d modellen te sculpturen en te animeren voor games of films. Blender is erg groot en er zijn heel veel verschillende features die je kan gebruiken om te animeren, hierdoor is het moeilijk om als beginner te beginnen in Blender door de vele features in Blender, ook gaan de meeste tutorial ervan uit dat de gebruiker al goed met Blender overeen kan. Maar door de vele features worden in Blender erg mooie animaties gemaakt, met mooie transities.

Zelf heb ik geprobeerd om in Blender een aantal animaties te creëren voor de speler, deze animaties waren dan in first person, ik heb daarvoor 2 modellen van het internet gepakt dat na nader inzien niet zo verstandig was, want het model voor de handen miste een aantal belangrijke tools voor animeren, bones, bones zijn eigenlijk de botten in model, deze bones kan per frame verplaatsen van locatie zodat Blender een animatie maakt. Ik kwam er zo laat achter dat dit een belangrijk onderdeel was dat ik het maar had opgegeven voor nu. In plaats van de animaties zelf te maken heb ik uiteindelijk een iemand gevonden die erg mooie animatie maakt voor first person shooters die hij gratis distribueerden.

- *Mixamo*

Mixamo is volledig gratis en er is een mogelijkheid om verschillende 3d modellen kiezen om te gebruiken voor animaties of een eigen 3d model te uploaden en die op de animaties van Mixamo. Hierdoor is Mixamo erg beginner vriendelijk en is het makkelijk om een animatie toe te voegen aan een model die je op het internet hebt gevonden of zelf hebt gemaakt. Deze animaties kunnen daarna vanuit de website worden gedownload in een speciale Unity FBX-format die makkelijk toe te voegen is in Unity, er kunnen soms materiaalfouten voorkomen. Deze zijn makkelijk te verhelpen door in het model in Unity de materialen op external te zetten, hierdoor worden de materialen die op het FBX-model zaten automatisch overgezet naar de juiste materialen.

Mixamo is een gratis website voor animaties en 3d modellen die erg makkelijk is om te gebruiken, waardoor het een snelle oplossing is om mooie animaties toe te voegen aan je game.

- **Animaties gebruiken in Unity**

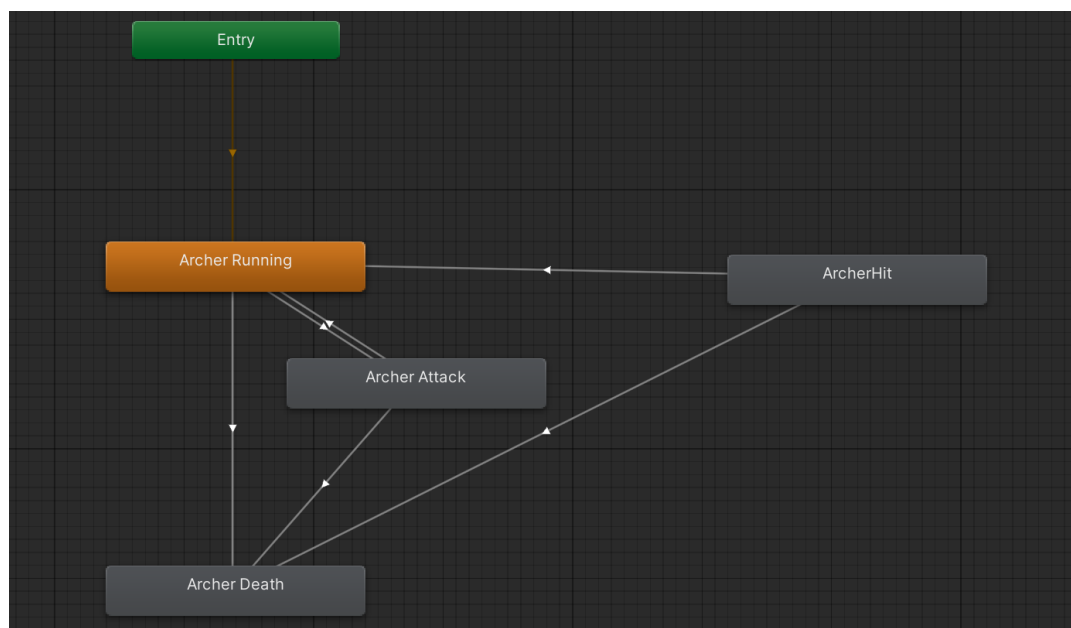
In Unity wordt de animator controller gebruikt om animaties te beheren via een animatie tree. Dit is een visuele state machine waarmee je animaties kunt koppelen aan verschillende toestanden (states) van een object, zoals lopen, rennen of aanvallen. Overgangen tussen animaties worden gecontroleerd met behulp van parameters, zoals bool, trigger of een float. Je kunt lijnen trekken tussen verschillende animaties waarin je de parameters kan toevoegen, zodat wanneer een parameter geactiveerd wordt verandert de animatie naar de lijn waar de pijl naar toestaat.

- *Basisfuncties animatie tree*

**Bool:** Een bool kan bijvoorbeeld gebruikt worden om een loop-animatie te starten wanneer `IsWalking` waar is, of te stoppen wanneer deze onwaar wordt.

**Trigger:** Triggers worden vaak gebruikt voor eenmalige gebeurtenissen, zoals het uitvoeren van een aanval. Wanneer een trigger wordt geactiveerd, speelt de gekoppelde animatie af en keert daarna terug naar de vorige toestand zonder behulp van de trigger parameter, je kan dus in de terugkeer na de trigger andere parameters gebruiken om de animaties er netter eruit te laten zien.

**Float:** Floats worden gebruikt om waardes door te geven, zoals de snelheid van een speler, Hierdoor kan je een val animaties maken wanneer een speler zijn y-as verandert. Dit is handig voor het werken met Blend Trees, waarmee de overgangen tussen animaties soepel verlopen op basis van snelheid.



Figuur 1: Voorbeeld van een animatie tree.

- *Animaties in scripts beheren*

Je kunt via de code de animatie tree aansturen doormiddel van de animator-klasse. Er zijn verschillende soorten functies zoals Play(), SetBool() en SetTrigger().

Play(): Hiermee kan je een specifieke animatie starten die meteen wordt afgespeeld en andere animaties overlapt zelfs wanneer hij zelf al wordt afgespeeld. Dit is erg handig voor bijvoorbeeld een schiet animatie waarbij je al sneller weer kan schieten dan de animatie lang is.

```
gunAnimator.Play("shoot", -1, 0f); // Play the shoot animation from the beginning
```

Figuur 2: voorbeeld van een Play() functie.

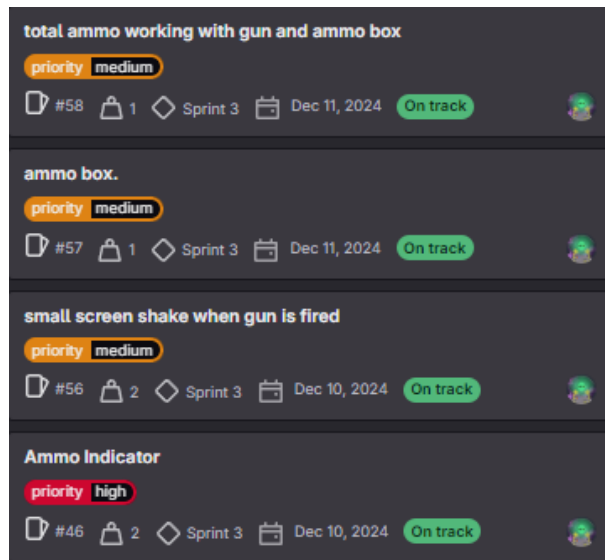
SetBool(), SetTrigger() en SetFloat(): Deze functies kunnen worden gebruikt om parameters in de Animator Controller aan te passen vanuit een script.

```
gunAnimator.SetFloat("moving", speed);  
gunAnimator.SetBool("ismoving", speed > 0);
```

Figuur 3: voorbeeld van Set() functie.

## 2. Issue bord in GitLab

In GitLab hebben we als team een aantal voorwaarden gesteld voor onze issues. Deze voorwaarden zijn: Een prioriteit, een weight, een datum en wat de toestand is van de issue. Doormiddel van deze voorwaarden kan er naar het issuebord gekeken worden en kan er op één oogopslag zien hoe het eraan toegaat voor de issue.



Figuur 4: Issue

In de issue zelf zetten we ook voorwaarden voor wanneer de issue klaar is (Definition of Done). Hierin wordt duidelijk gezet wat er af moet zijn voor de specifieke issue, hierdoor kunnen de teamleden ook nakijken wat er wordt gemaakt voor de issue.



Figuur 5: DOD in een issue

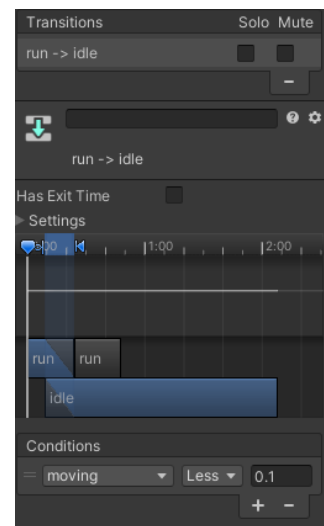
### 3. Animaties in het project

- Speler animatie tree

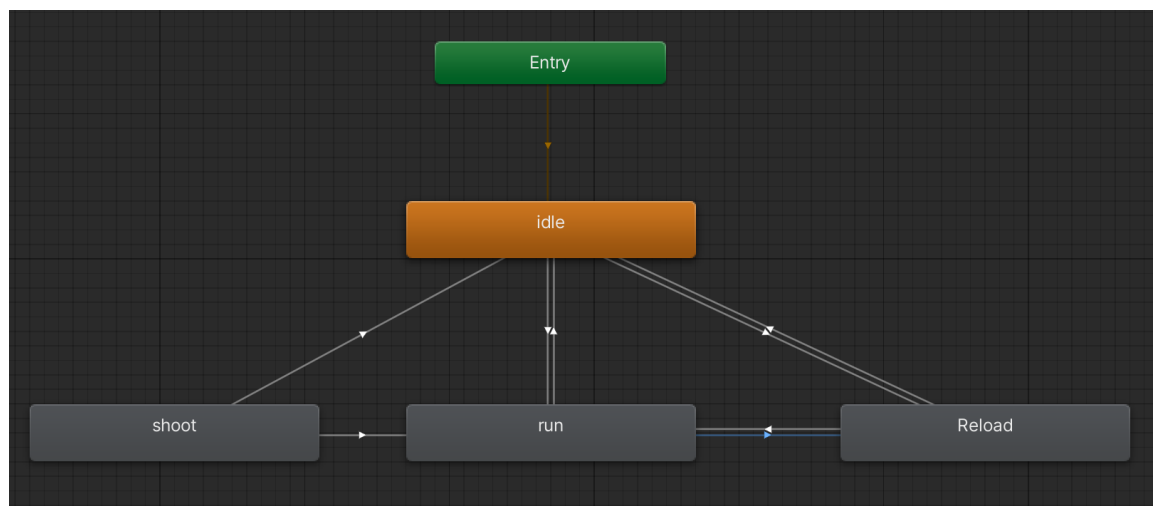
Voor het project heb ik een animatie tree gemaakt voor de speler gemaakt. Ik heb een speler model van het internet gebruikt. Dit model had al animaties erbij, dus deze gebruiken we ook voor het project.

In de animatie tree voeg je 5 animatie toe vanuit het 3d model, de animaties: Draw, idle, shoot, run en reload. Zet de idle animatie als entry animatie, zodat deze altijd afspeelt.

Maak de transitielijnen zoals ze op het **voorbeeld** worden weer gegeven. Maak voor de transitielijnen naar de Reload animatie een boolean aan die reload wordt genoemd. Maak een float aan die moving wordt genoemd, hiermee kijken we of de speler zich voortbeweegt. Voeg deze aan de transitielijnen richting run en terug toe, zie figuur 6 hoe de conditie moet worden ingevuld. Je kan natuurlijk altijd andere variaties gebruiken om de animaties aan te sturen, zoals een bool te gebruiken om de run animatie te activeren, hierdoor kan je de animatie stoppen wanneer je bijvoorbeeld de run knop niet meer aan raakt.



Figuur 6: voorbeeld transitie.



Figuur 7: voorbeeld animatie tree

- **Animaties activeren via code**

Variables van de animatie tree moet je veranderen doormiddel van code, dus in het script van de speler veranderen we een aantal variables naar aanleiding van de input van de speler.

- *Shoot animatie activeren*

De shoot animatie wordt geactiveerd met de Play() functie van de Animator klasse. Door Play() te gebruiken wordt de schiet animatie altijd uitgevoerd wanneer er wordt geschoten.

```
private void GunShotEffects()
{
    gunAnimator.Play("shoot", -1, 0f); // Start de schietanimatie
    // Overige effecten zoals muzzle flash en geluid
}
```

Figuur 8: voorbeeld shoot animatie.

- *Reload animatie activeren*

De Reload animatie wordt geactiveerd met de SetTrigger() functie.

```
gunAnimator.SetTrigger("reload"); // Activeer de herlaadanimatie
```

Figuur 9: voorbeeld reload animatie.

- *Run animatie activeren*

De run animatie wordt geactiveerd met de.SetFloat() functie, hierbij is de float speed de snelheid die de speler rent in de game, deze wordt in de update meegegeven.

```
private void RunAnimation(float speed)
{
    gunAnimator.SetFloat("moving", speed); // Stel de snelheid in
    gunAnimator.SetBool("ismoving", speed > 0); // Activeer de loopanimatie als de snelheid groter is da
}
```

Figuur 10: voorbeeld Run animatie.

## Reflectie

Ik denk dat ik zelf een goed onderzoek heb gedaan naar hoe je animatie kan toevoegen aan je project in Unity en dat ik het netjes heb toegevoegd aan Unity, maar ik heb laatst onderzoek gedaan naar statemachines voor vijanden. Als ik deze statemachine toepas in het huidige project is het veel duidelijker in code wanneer de vijand een animatie gaat afspelen.



# Bronnen

## Literatuur

Technologies, U. (z.d.). *Unity - Manual: Animation*.

<https://docs.unity3d.com/Manual/AnimationSection.html>

Technologies, U. (z.d.-b). *Unity - Manual: Animation Blend Trees*.

[https://docs.unity3d.com/Manual/class-BlendTree.html#:~:text=Use%20the%20plus%20icon%20to%20add%20animation%20clips%20or%20child%20blend%20trees.&text=The%20context%20menu%20when%20you, on%20a%20blend%20tree%20node.&text=This%20visualization%20changes%20as%20the, show%20the%20dominant%20animation%20clip\).](https://docs.unity3d.com/Manual/class-BlendTree.html#:~:text=Use%20the%20plus%20icon%20to%20add%20animation%20clips%20or%20child%20blend%20trees.&text=The%20context%20menu%20when%20you, on%20a%20blend%20tree%20node.&text=This%20visualization%20changes%20as%20the, show%20the%20dominant%20animation%20clip).)

Mixamo > *Unity does not import any textures*. (2019, 29 januari). Unity Discussions.

<https://discussions.unity.com/t/mixamo-unity-does-not-import-any-textures/691058/2>

Lampel, J. (2020, 8 juni). *How great first person animations are made - 10 tips for animating FPS characters in Blender*. CGCookie.com.

<https://cgcookie.com/posts/how-great-first-person-animations-are-made-10-tips-for-animating-fps-characters-in-blender>

*Editing Bones — Blender manual*. (z.d.).

<https://docs.blender.org/manual/en/2.81/animation/armatures/bones/editing/bones.html>

## Audiovisueel

iHeartGameDev. (2020, 31 mei). *How to Animate Characters in Unity 3D | Animator Explained* [Video]. YouTube. <https://www.youtube.com/watch?v=vApG8aYD5aI>

iHeartGameDev. (2020b, juni 28). *How to Animate Characters in Unity 3D | Animation Transitions With Booleans* [Video]. YouTube.

<https://www.youtube.com/watch?v=FF6kezDQZ7s>

UGuruz. (2021, 10 mei). *How to fix Pink Materials in Unity* [Video]. YouTube.

<https://www.youtube.com/watch?v=nB0r0c-SIVg>

thriftydonut. (2021, 3 januari). *How to make FPS Animations in Blender 2.8+* [Video].

YouTube. <https://www.youtube.com/watch?v=IV6XP-EDzw8>

Digvijaysinh Gohil. (2023, 14 augustus). *Simple way to make Muzzle flash* [Video].

YouTube. <https://www.youtube.com/watch?v=EiDxxH9wdq0>

Synty Studios. (2023, 14 maart). *How to create animations in Unity (Tutorial) by*

*#SyntyStudios* [Video]. YouTube. <https://www.youtube.com/watch?v=78IrmMtByAU>

Unity. (2014, 2 februari). *Blend Trees - Unity official tutorials* [Video]. YouTube.  
<https://www.youtube.com/watch?v=HeHvLEYpRbM>

## Assets

Speler model met animaties <https://sketchfab.com/ccransh>

Vijand zombie model zonder animaties <https://www.cgtrader.com/free-3d-models/character/fantasy-character/mutant-5ed2b9de-daeb-4005-af3f-38875267be55>