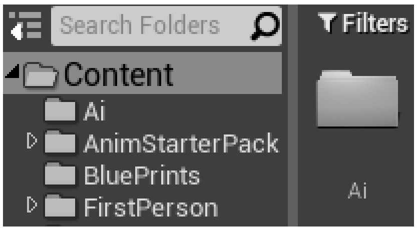
Creating enemy AI

In this section I will show how I created an enemy Ai step by step:

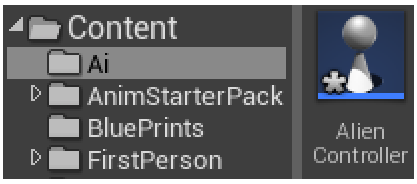
For the ai I simply needed the enemy to chase and hit the player. The enemy will also wonder around the level to make the game seem more dynamic.

Getting started:

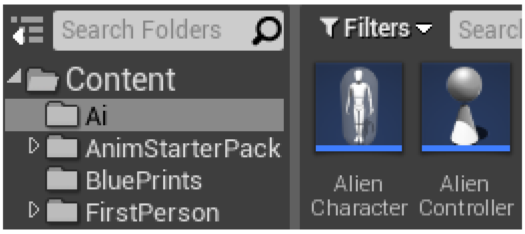
The first step was to create a new folder named 'AI'



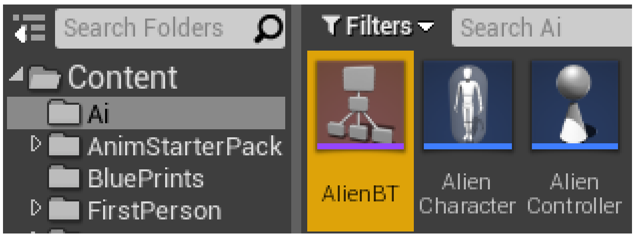
And in this folder, I added a new 'Blueprint Class' as a 'AIController' named 'AlienController'.



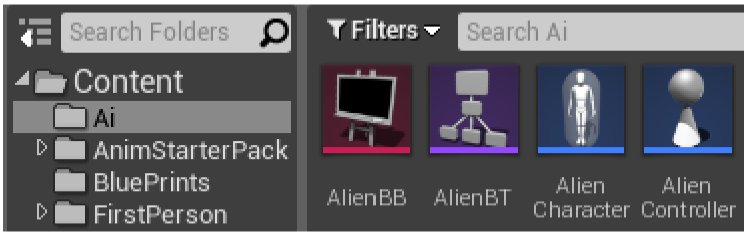
Now that I had this, I needed an Ai character. so, I added another 'Blueprint class' as a 'Character' named 'AlienCharacter'.



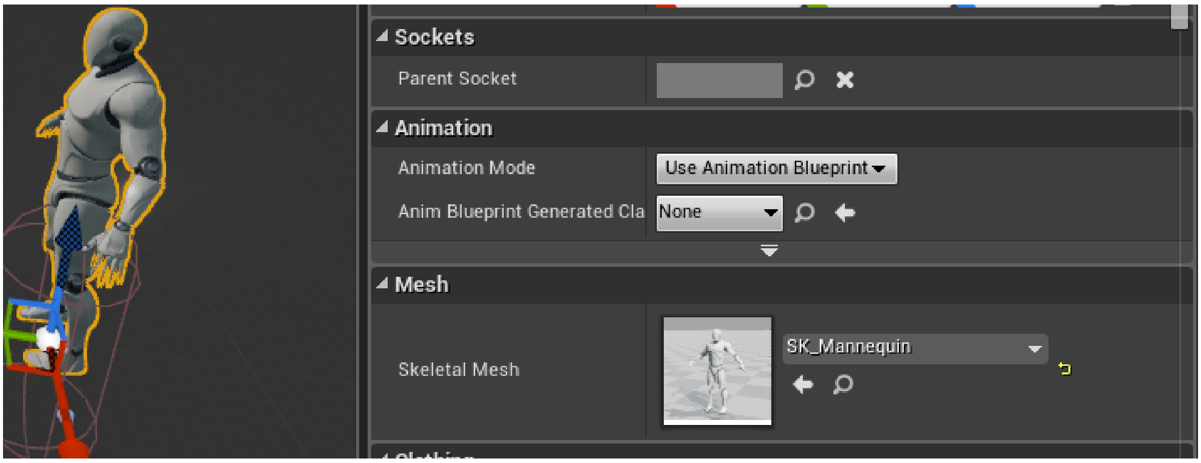
The next step was to add a 'Behaviour tree' named 'AlienBT', this was found by right clicking and hovering over 'Artificial Intelligence'.



Within this I needed a ‘Blackboard Asset’ however, I first needed to create a ‘Black Board’ named ‘AlienBB’.



I then added a ‘mesh’ to the ‘Aliencharacter’. To do this I first made the ‘SK_Mannequin’ the ‘Mesh’


Sockets

Parent Socket

Animation

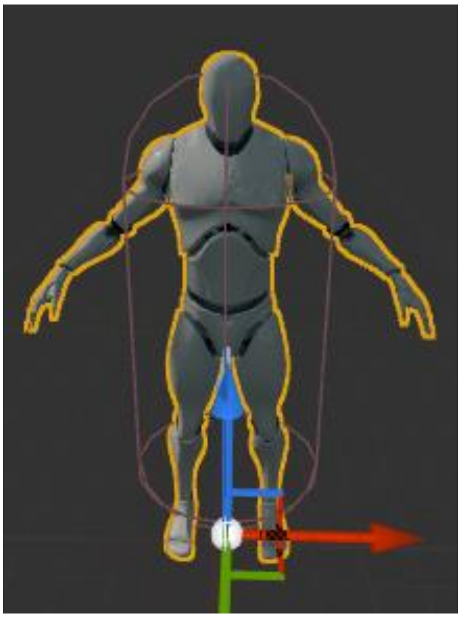
Animation Mode: Use Animation Blueprint

Anim Blueprint Generated Cla: None

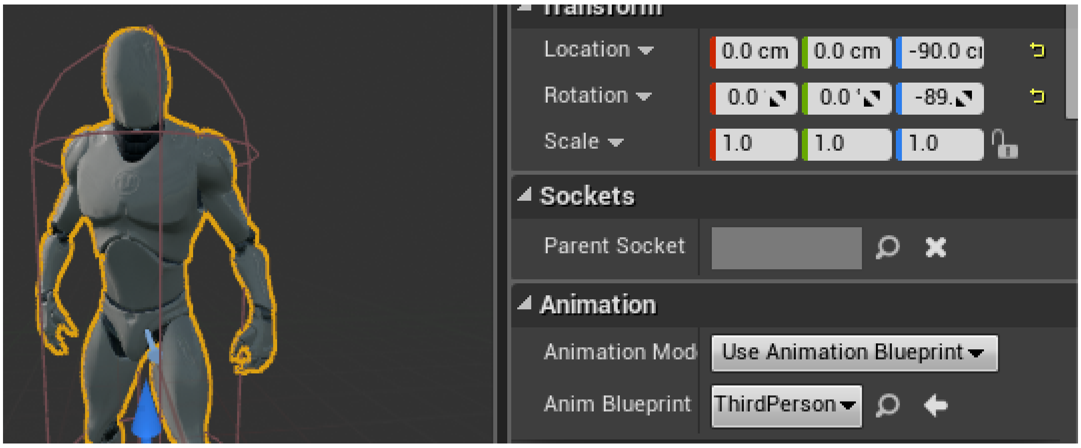
Mesh

Skeletal Mesh: SK_Mannequin

He was then rotated (-90°) and dragged down to (-90°) to fit into the ‘Capsule’.



I then decided to share the ‘ThirdPerson_AnimBP’ animations with this new mesh for now. This was done in ‘Anim Blueprint Generated Class’.



Transform

Location 0.0 cm | 0.0 cm | -90.0 cı

Rotation 0.0 | 0.0 | -89.

Scale 1.0 | 1.0 | 1.0

Sockets

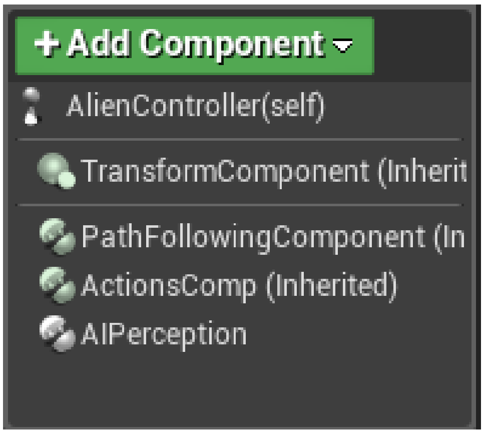
Parent Socket

Animation

Animation Mod: Use Animation Blueprint

Anim Blueprint: ThirdPerson

The next step was to work on the AI now. I started by adding senses so that my AI could see and hear. This was done in the ‘AlienCharacter’ folder. The first thing to do hear was to add a new ‘Componant’ as ‘AIPerception’.



AlienController(self)

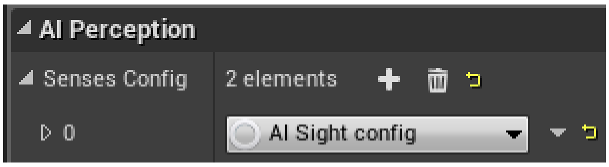
TransformComponent (Inherit

PathFollowingComponent (In

ActionsComp (Inherited)

AIPerception

Now under AI Perception I needed to add two new ‘Elements’, these are what will act as the enemy senses. The first was set to ‘AI Sight config’

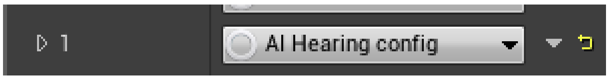


AI Perception

Senses Config: 2 elements

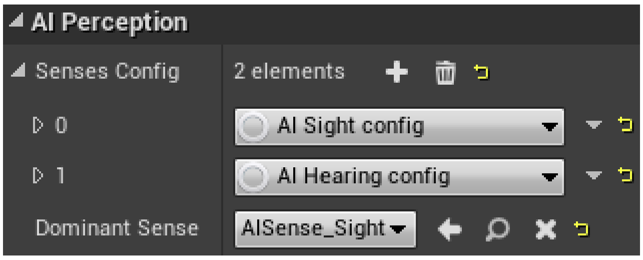
0: AI Sight config

And the second was set to ‘AI Hearing config’.



1: AI Hearing config

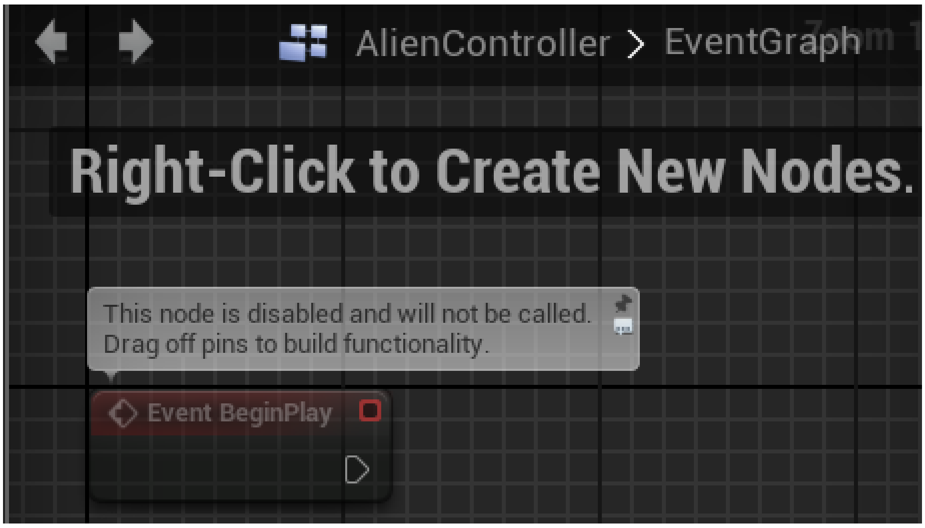
The ‘Dominant Sense’ was then set to ‘AISense_Sight’ because I wanted the main thing to cause my enemy to attack the player to be its line of sight.



AI Perception

Senses Config	2 elements
0	AI Sight config
1	AI Hearing config
Dominant Sense	AISense_Sight

The next step was to tell the ‘controller’ to use the ‘behaviour tree’. This was done in the ‘AlienController – EventGraph’.

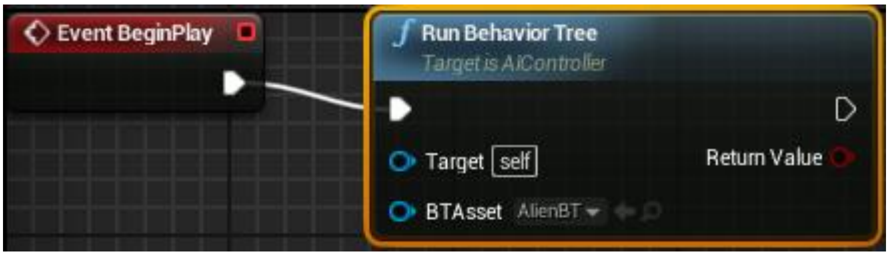


AlienController > EventGraph

Right-Click to Create New Nodes.

This node is disabled and will not be called.
Drag off pins to build functionality.

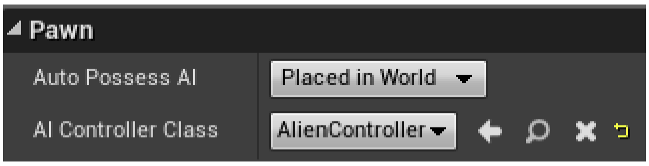
The first ‘Node’ added here was a ‘Run Behaviour Tree’ with the ‘BTAsset’ being the ‘AlienBT’.



Target self

BTAsset AlienBT

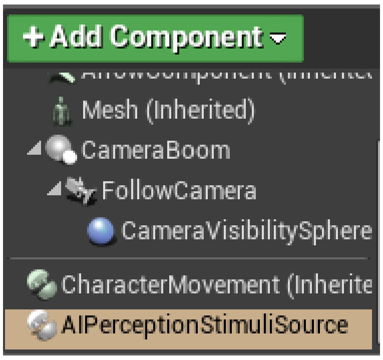
Now I needed to make the ‘AlienCharacter’ use the ‘AI Controller’ instead of looking for an ‘input controller’. This was done under ‘Pawn’ by setting the ‘AI Controller Class’ to ‘AlienController’.



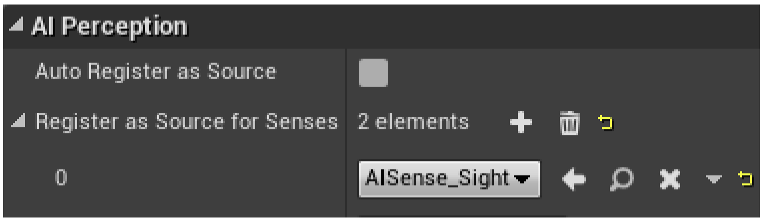
Pawn

Auto Possess AI	Placed in World
AI Controller Class	AlienController

I now needed to add a ‘source’ for the ‘senses’ the first being the playable character and the second being a projectile for if the player throws something the AI will run towards it. To do this I needed to add a new component to the ‘ThirdPersonCharacter’ set as a ‘AIperceptionStimuliSource’.



And in this I added two elements, one being set to ‘AISense_sight’

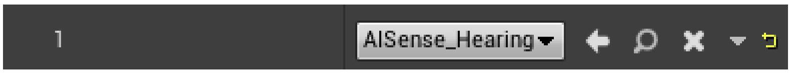

AI Perception

Auto Register as Source □

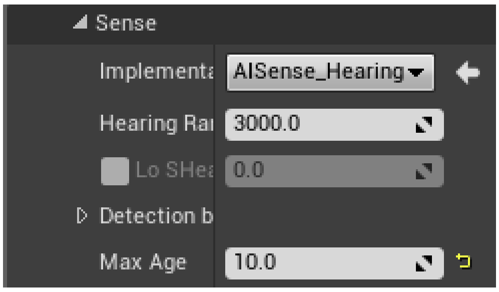
Register as Source for Senses 2 elements

0 AISense_Sight

And another set to ‘AISense_Hearing’.


1 AISense_Hearing

Now I needed to set up how the senses work on the controller this was done in the ‘AlienController’. For the hearing sense I set the ‘Max Age’ as (10.0) this is the number of ticks that go by where if the enemy can’t see the object it will ignore it.


Sense

Implementa AISense_Hearing

Hearing Ra 3000.0

□ Lo SHe 0.0

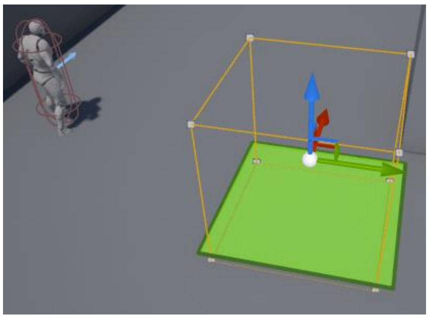
Detection b

Max Age 10.0

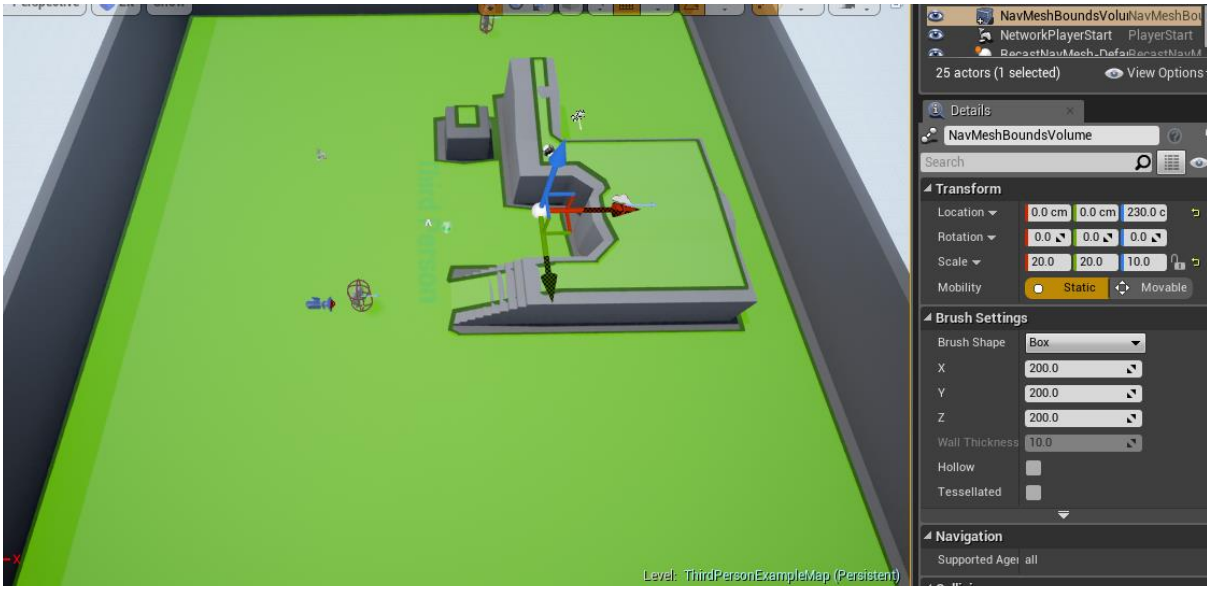
This was the end of my reference video ‘Creating a Zombie AI in UE4 [4.25] / Getting Started [Part 1]’ (O, Odyssey 2020)

Tasks, Keys & Behaviour Trees:

In this next section I needed to set up tasks in the behaviour tree to allow the enemy to wander around the map. To do this I needed to add a 'nav mesh bounds volume' this is the area that the enemy will wander around.



The 'nav mesh bounds volume' was then set to a scale of X value – 20, Y value – 20 and value Z – 10. This covered the default area of Unreals layout and will be changed later to accommodate my level design.



NavMeshBoundsVolu NavMeshBo
NetworkPlayerStart PlayerStart
RecastNavMesh-Defa RecastNavM
25 actors (1 selected) View Options

Details
NavMeshBoundsVolume

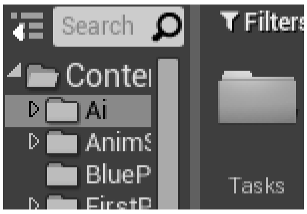
Transform
Location 0.0 cm 0.0 cm 230.0 c
Rotation 0.0 0.0 0.0
Scale 20.0 20.0 10.0
Mobility [✓ Static] Movable

Brush Settings
Brush Shape Box
X 200.0
Y 200.0
Z 200.0
Wall Thickness 10.0
Hollow □
Tessellated □

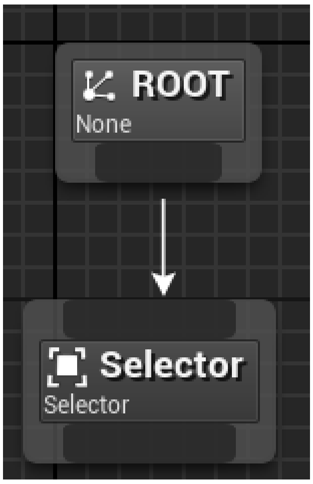
Navigation
Supported Age all

Level: ThirdPersonExampleMap (Persistent)

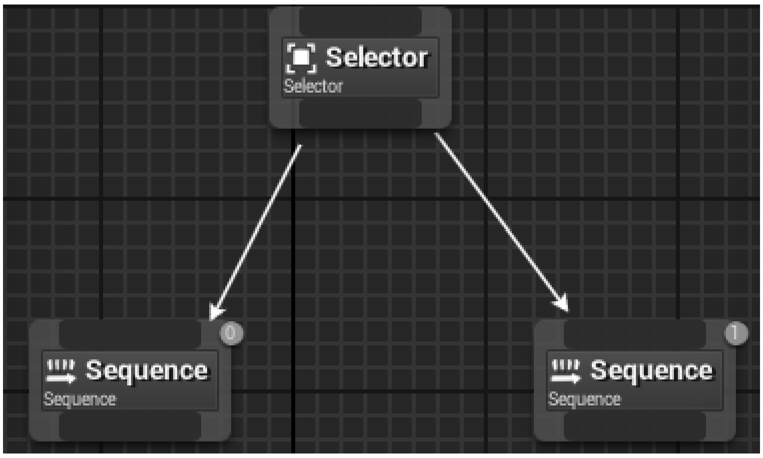
The next step was to add a new folder named 'Tasks' for now this was left blank as the tasks were set up in the 'Behaviour Tree'.



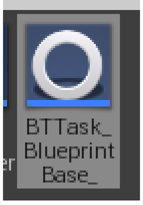
Now I needed to add a ‘selector’ in the ‘behaviour tree’ this will base which path the tasks go down, based on a ‘condition’.



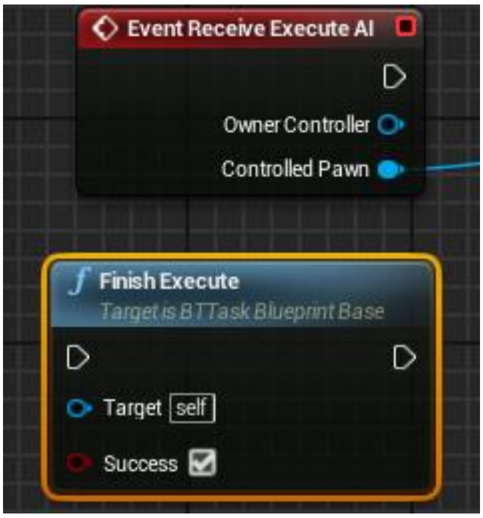
Next, I needed to add a ‘Sequence’ for it to carry out a sequence of things for the enemy to do such as to find a random location and move towards that location and a second to find the player and run towards them.



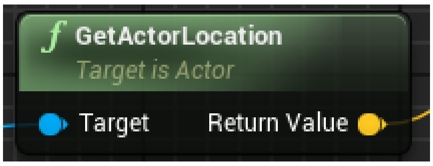
Now I needed to be able to add a custom ‘Task’ by creating a new ‘task’ named ‘BTTask_BlueprintBase_FindRandomLocation’. This I what will be telling the enemy AI to move to a random location in a ‘navical point’ determined by the ‘nav mesh bounds volume’.



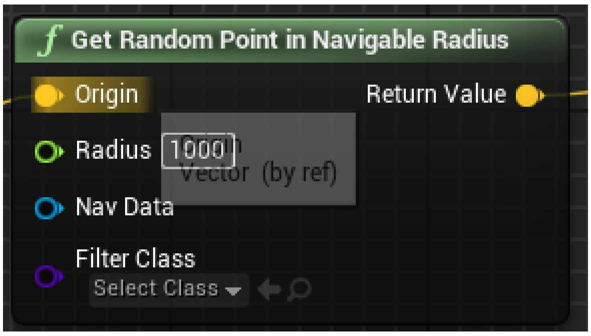
Within this I first needed to add two things them being firstly an ‘Event: Receive Execute AI’ node and secondly a ‘finish Execute’ this will be at the end set to ‘Success’.



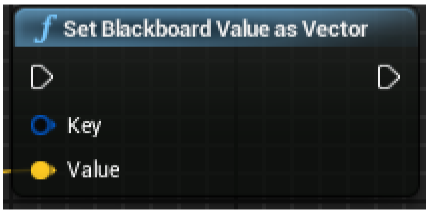
From the ‘Event: Receive Execute AI’ I then added a ‘GetActorLocation’ this will get the location of the enemy character.



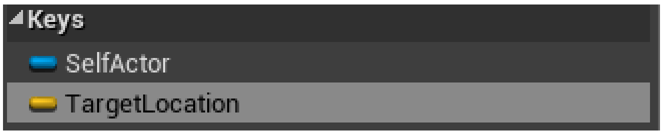
And from this I added a ‘Get Random point In Navigable Radius’ set to a ‘Radius’ of ‘1000’ so that the enemy will only run for a short period of time.



Next, I needed to add a ‘Set Blackboard Value as Vector’ this will automatically get the ‘data type’ and set that as the value.

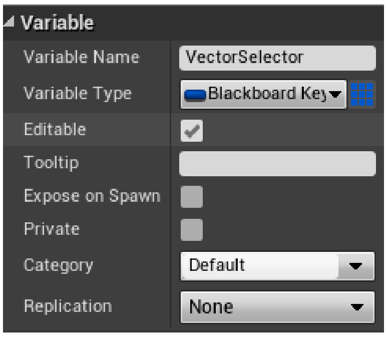


Now I needed the behaviour tree to inherit something from the ‘Task’. So, to do this I first needed to add a ‘New Key’ as a ‘Vector’ in the ‘Alien_BB’ named ‘TargetLocation’.



Keys
- SelfActor
- TargetLocation

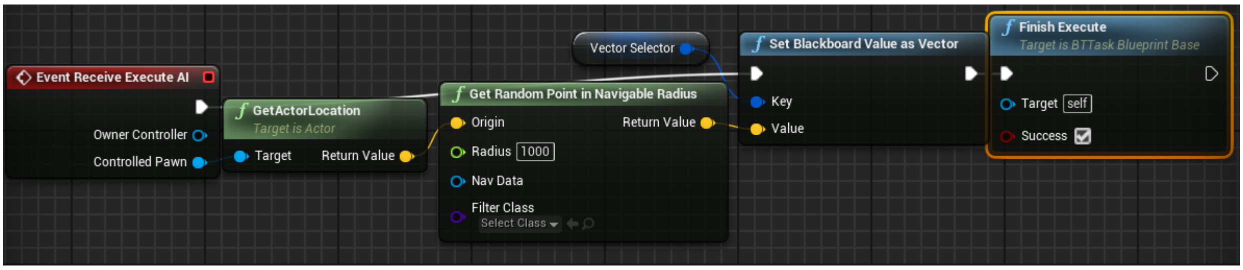
This is now on the behaviour tree so now I needed to create a new variable in the ‘BTTask_BlueprintBase_FindRandomLocation’ set as a ‘Blackboard key Selector’ named ‘VectorSelector’. This was also set as a ‘public variable’ so it can be seen outside of the ‘Task’.



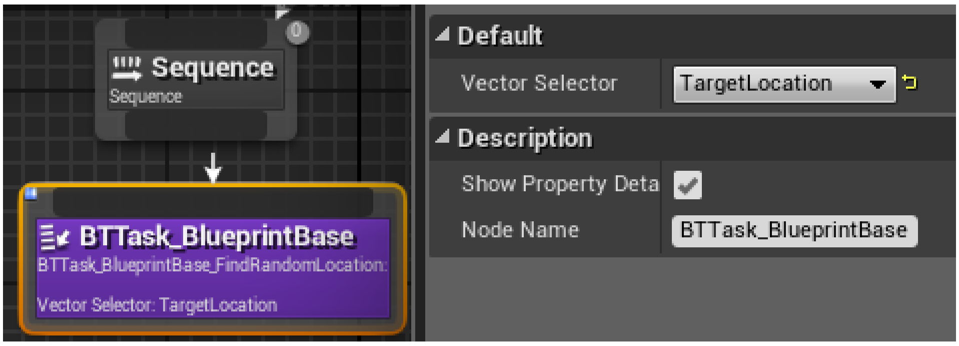
Variable

Variable Name	VectorSelector
Variable Type	Blackboard Key
Editable	☑
Tooltip	
Expose on Spawn	□
Private	□
Category	Default
Replication	None

This is then fed into the ‘key’ on the ‘Set Blackboard Value as Vector’. And everything else was connected like so…



Now this task will allow the enemy to find a random location and move to the random location. Next, I needed to add this ‘Task’ to the behaviour tree by dragging from the left sequence and setting the ‘Vector selector’ from ‘SelfActor’ to ‘TargetLoction’.


Sequence
Sequence
0
BTTask_BlueprintBase
BTTask_BlueprintBase_FindRandomLocation
Vector Selector: TargetLocation

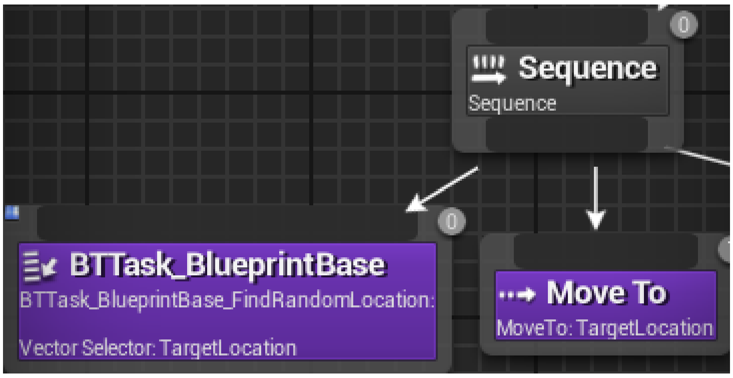
Default

Vector Selector	TargetLocation

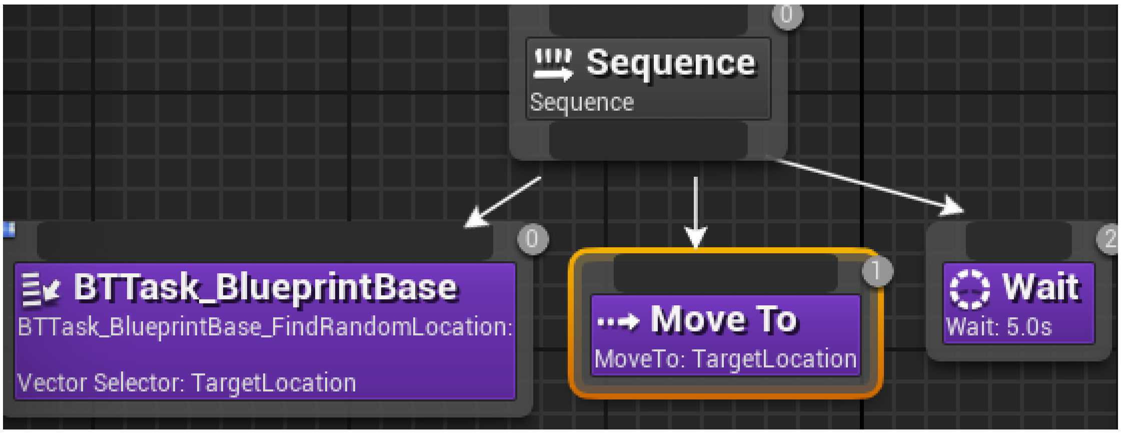
Description

Show Property Deta	☑
Node Name	BTTask_BlueprintBase

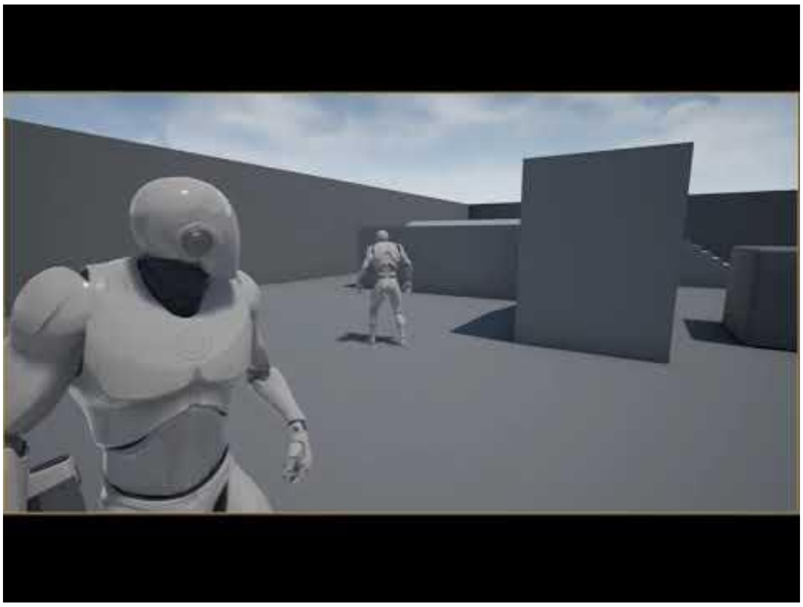
I then added a ‘Move to’ that is also set to ‘targetLocation’


0
Sequence
Sequence
0
BTTask_BlueprintBase
BTTask_BlueprintBase_FindRandomLocation:
Vector Selector: TargetLocation
Move To
MoveTo: TargetLocation

And lastly, I added a ‘Wait’ and left it as the default time of five seconds.



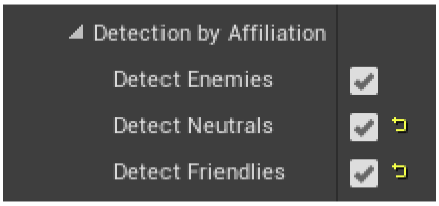
So now the ‘AlienCharacter’ is linked to the ‘AlienController’ which is linked to the ‘Alien_BT’ which is linked to the ‘Alien_BB’.



This was the result of my reference video ‘Creating a Zombie AI in UE4 [4.25] / Tasks, Keys & Behaviour Trees! [Part 2]’ (O, Odyssey 2020)

More Tasks, More Keys & Sight perception:

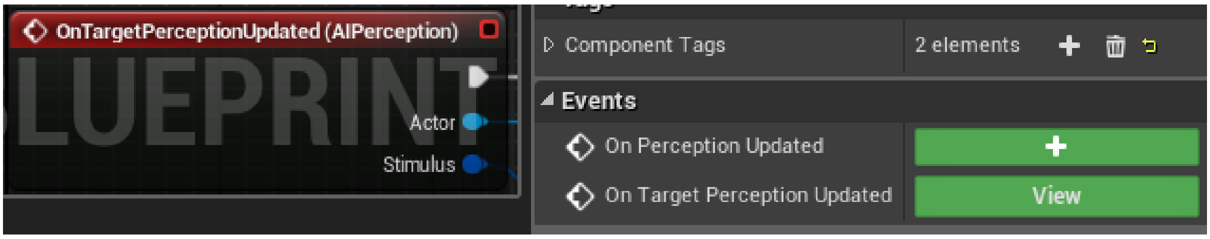
In this next section I will be continuing with the enemy Ai behaviour tree aswell as setting up the perception for the 'AlienController'. Firstly, in the 'AlienController' there were the 'detection by affiliation' boxes that all needed to be ticked.



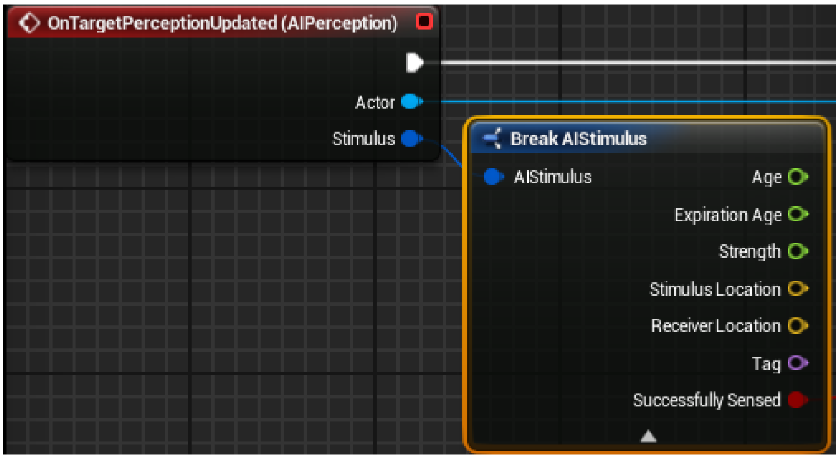
Detection by Affiliation

Detect Enemies	☑
Detect Neutrals	☑
Detect Friendlies	☑

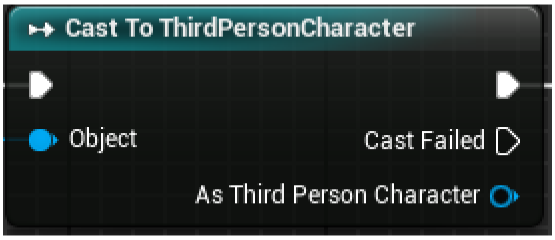
The first node that needed to be added to start the move to player logic was a 'OnTargetPerceptionUpdated' node this will receive the stimulus from the component.


OnTargetPerceptionUpdated (AIPerception)
Actor
Stimulus
Component Tags
2 elements
Events
On Perception Updated
On Target Perception Updated
View

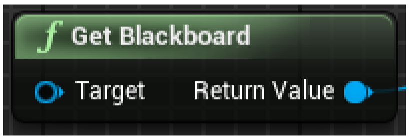
From the 'stimulus' a 'Break AIStimulus' was added



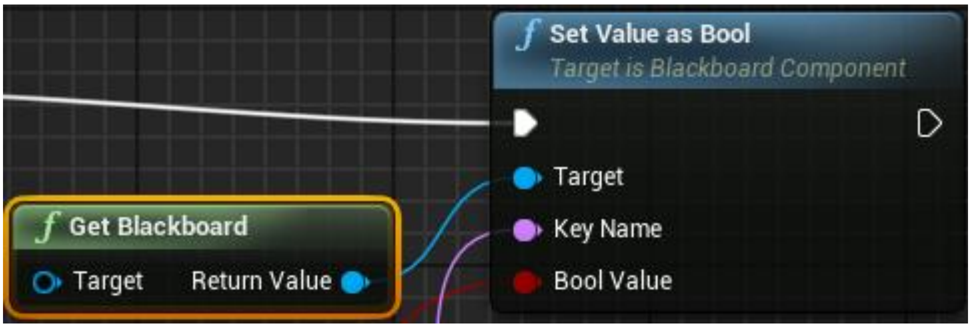
And from the 'Actor' a 'Cast To ThirdPersonCharacter' was added.



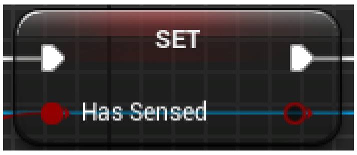
Next the value had to be set to a 'Boolean' but this was only obtainable from the 'Blackboard'. So, a 'Get Blackboard' was added



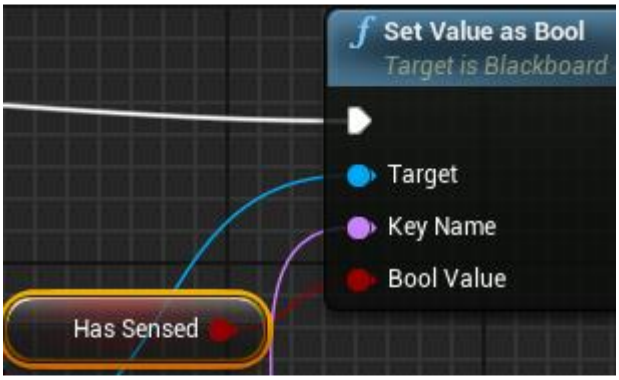
and from this a 'set Value as Bool' was created.



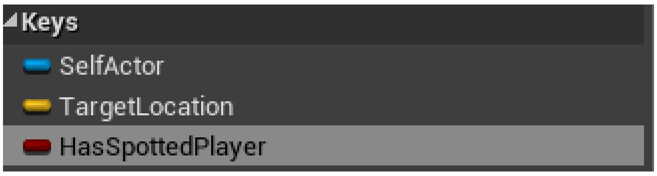
Now I needed to promote the 'Break AIStimulus' to a variable named 'HasSensed'.



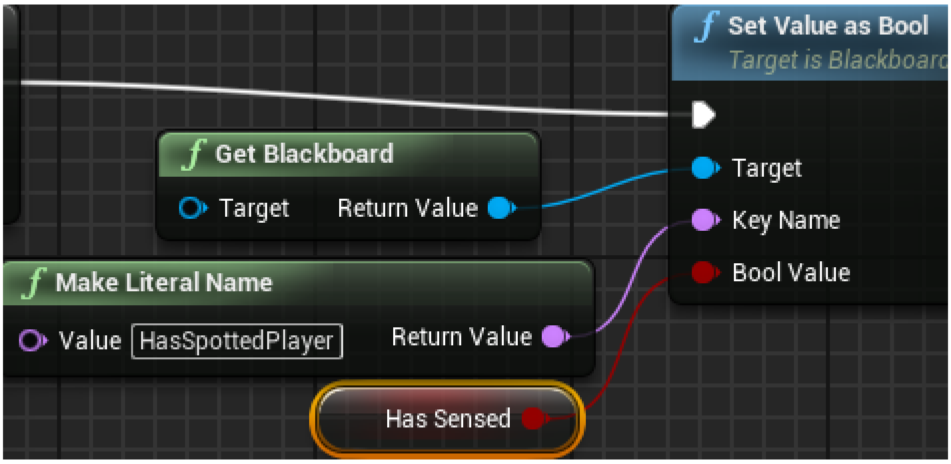
This variable is also set as the 'Bool value'.

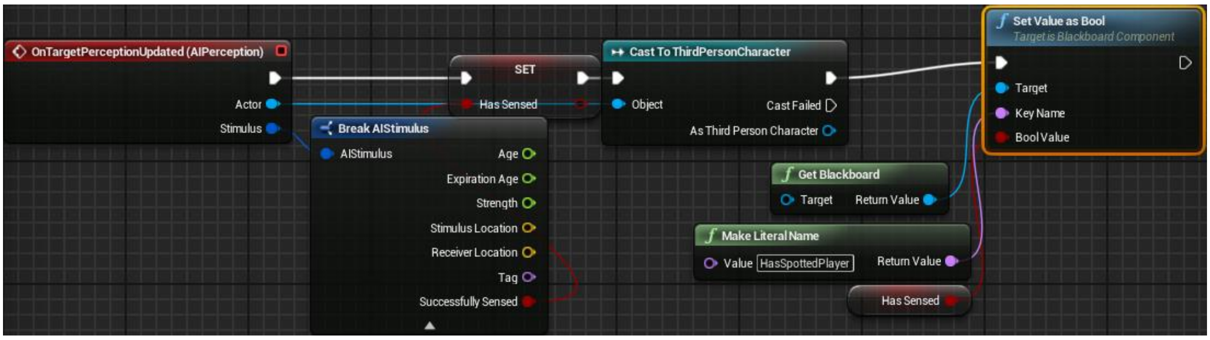


Next, I needed to create a new black board key for the condition that was just set up. This key was set up to be a 'Boolean' named 'HasSpottedPlayer'.



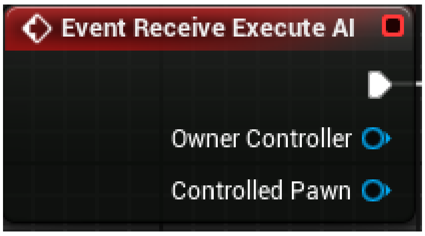
This was now ready to be added with a ‘Make Literal Name’ as the ‘Key name’ back in the ‘Aliencontroller – Events Graph’.



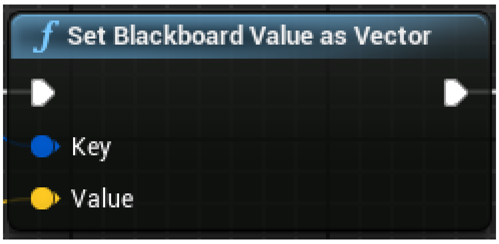


The next thing I needed to do was create a new ‘Task’ named ‘BTTask_BlueprintBase_FindPlayerLocation’. This new task handles the logic for the enemy chasing the player.

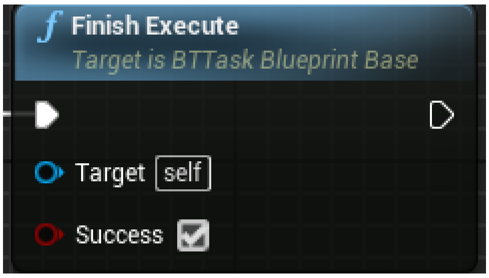
Within this the first thing that needed to be created was a ‘Event: Receive Execute AI’ node.



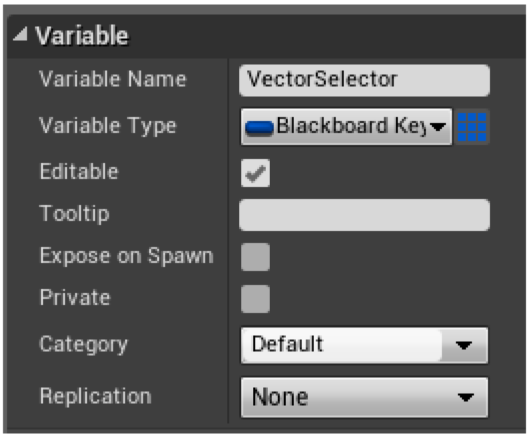
From this the black board key needed to be set as a ‘Vector’



And from that a ‘Finish Execute’ was created as a ‘Success’.



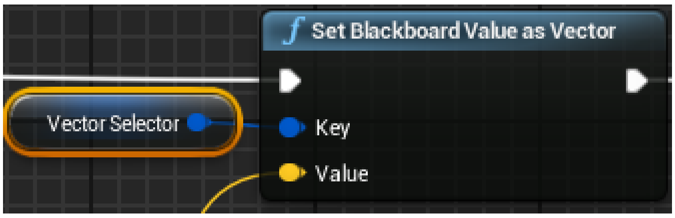
Next, I needed to make a ‘Vector Selector’ to do this I needed to create a new variable set as a ‘Blackboard key selector’ named ‘VectorSelector’, this also needs to be a ‘Public variable’ by ticking the ‘Editable’ box.



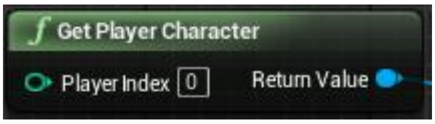
Variable

Variable Name	VectorSelector
Variable Type	Blackboard Key
Editable	☑
Tooltip	
Expose on Spawn	□
Private	□
Category	Default
Replication	None

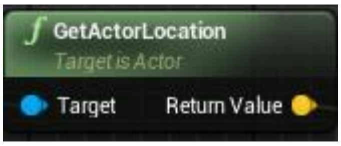
This was then set as the ‘Key’ on the ‘Set blackboard value as vector’ node.



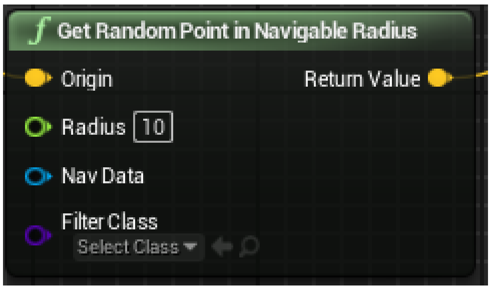
Now I needed a ‘Get Player Character’



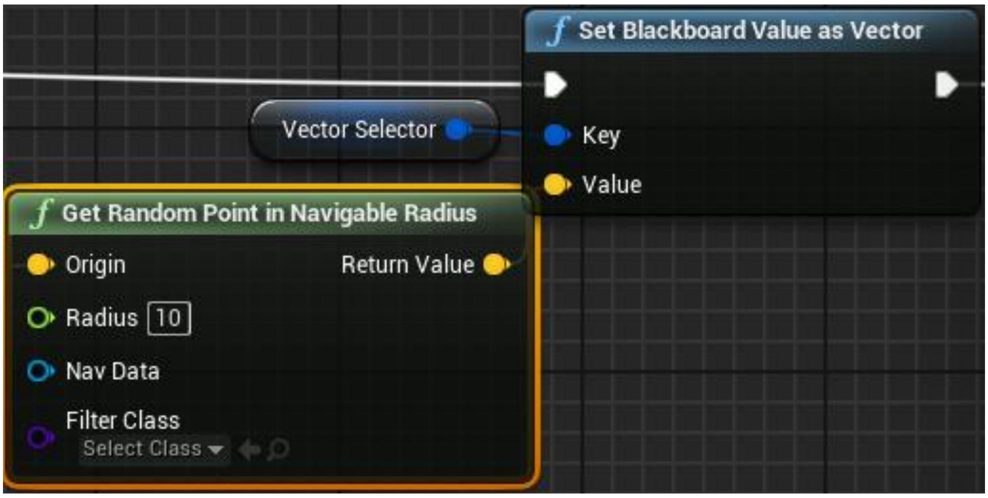
And from this I needed to get the location of the player by adding a ‘GetActorLocation’ this will return a ‘Vector Value’.

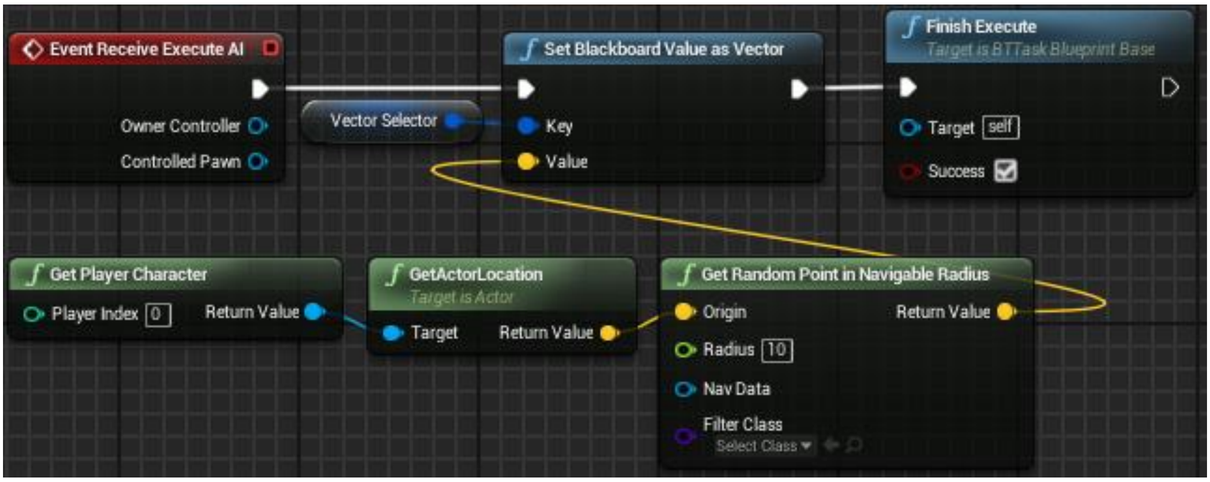


However, I needed something to stop the enemy from clipping through the player character as this would cause collision and realism issues. So, I needed to add a perimeter around the character by adding a ‘GetRandomPointInNavigableRadious’ and setting the ‘Radius’ to (10.0).

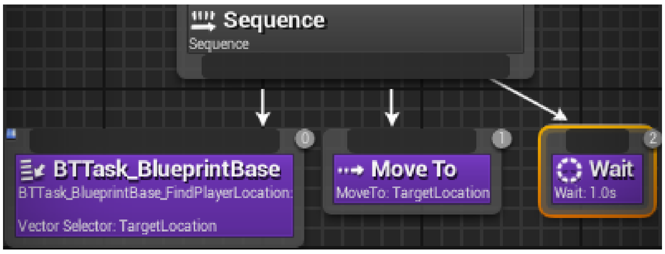


This was then set as the ‘Value’ on the ‘Set blackboard value as vector’ node.





This now needed to be implemented to the behaviour tree. By doing the same thing as before.



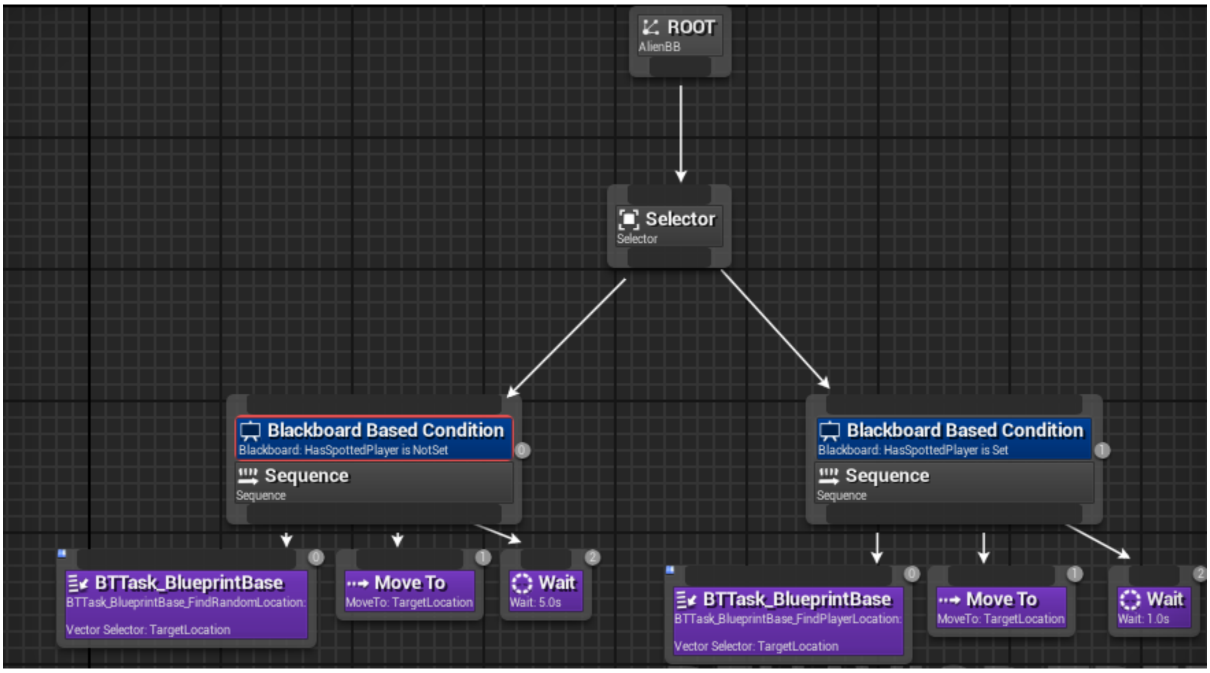
Sequence
Sequence

BTTask_BlueprintBase
BTTask_BlueprintBase_FindPlayerLocation:
Vector Selector: TargetLocation

Move To
MoveTo: TargetLocation

Wait
Wait: 1.0s

Next, I needed to add a 'decorator' to both sequences based on the condition in the 'AlienController – Event Graph' by right clicking on the sequence and adding the 'blackboard' option. With the self-actor set as 'HasSpottedPlayer'. One is 'set' and one is not 'set'.



ROOT
AlienBB

Selector
Selector

Blackboard Based Condition
Blackboard: HasSpottedPlayer is Not Set
Sequence
Sequence

BTTask_BlueprintBase
BTTask_BlueprintBase_FindRandomLocation:
Vector Selector: TargetLocation

Move To
MoveTo: TargetLocation

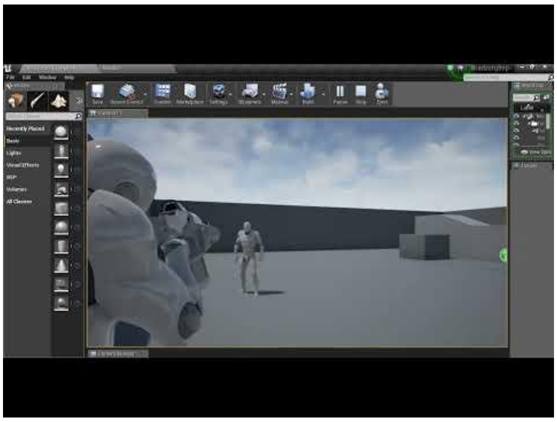
Wait
Wait: 5.0s

Blackboard Based Condition
Blackboard: HasSpottedPlayer is Set
Sequence
Sequence

BTTask_BlueprintBase
BTTask_BlueprintBase_FindPlayerLocation:
Vector Selector: TargetLocation

Move To
MoveTo: TargetLocation

Wait
Wait: 1.0s

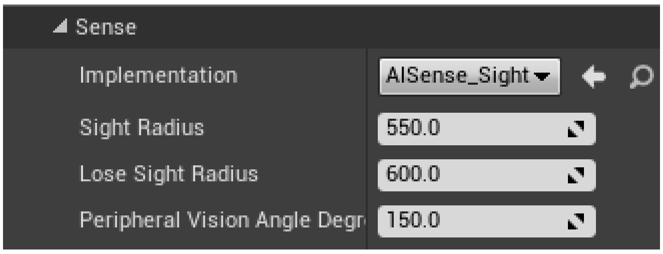


This was the result of my reference video 'Creating a Zombie AI in UE4 [4.25] / More Tasks, More Keys & Sight Perception! [part 3]' (O, Odyssey 2020)

Perception Refinements, Mesh & Animations:

In this section I will show the process of refining the sense of perception with the alien AI and importing my alien mesh, along with some animations that I downloaded from Mixamo. https://www.mixamo.com/#/

Firstly, I fixed the sight of the enemy Ai by simply changing a few values to the ‘AI perception’ in the ‘AlienController’. The ‘sight radius’ was changed to 550.0, The ‘lose sight Radius’ was changed to 600.0 and the ‘Peripheral Vision Angle degree’ was changed to 150.0. This made it easier for the enemy to find the player.



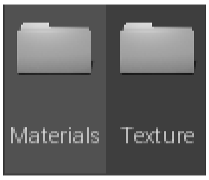
Sense

Implementation	AISense_Sight
Sight Radius	550.0
Lose Sight Radius	600.0
Peripheral Vision Angle Degr	150.0

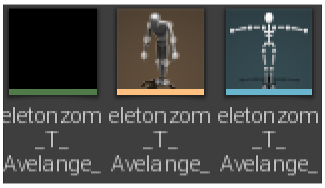
Now it was time to import the enemy mesh from Mixamo. Firstly, a new folder was created under ‘AI’ named ‘Alien’.

Next the mesh was imported on its own to prevent any confusion.

When this was imported it added a bunch of textures and materials so to keep organised these were moved into newly crated files.

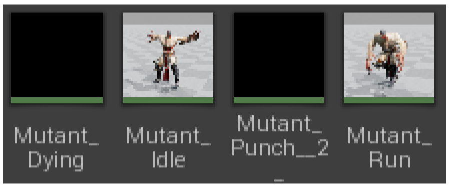


A new ‘skeleton’ and ‘physics asset’ were also automatically created by unreal upon download of the mesh.

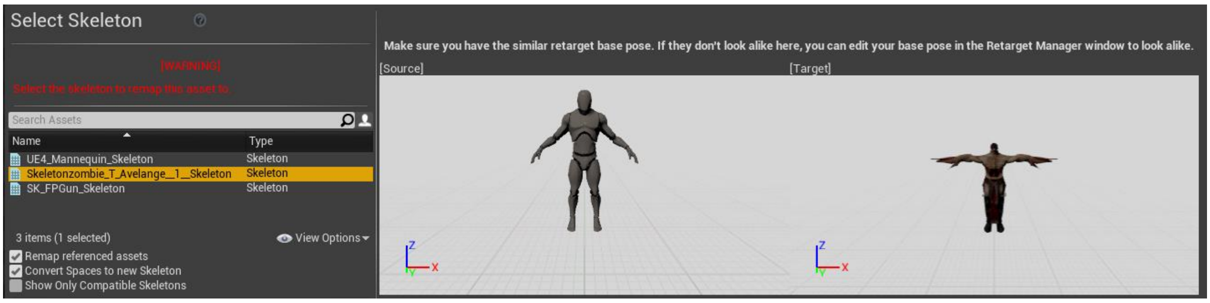


Next, I needed to import the downloaded animations however, this needed it own file to be created aswell.

These were the animations that I imported.



The ‘ThirdPerson_AnimBP’ needed to be retargeted to the Alien mesh.



Select Skeleton

[WARNING]

Select the skeleton to remap this asset to.

Search Assets

Name	Type
UE4_Mannequin_Skeleton	Skeleton
Skeletonzombie_T_Avelange__1__Skeleton	Skeleton
SK_FPGun_Skeleton	Skeleton

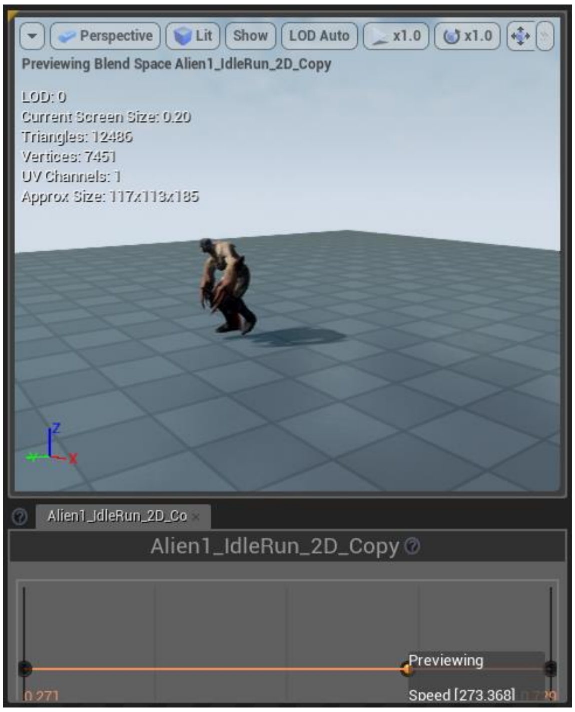
3 items (1 selected)

☑ Remap referenced assets
☑ Convert Spaces to new Skeleton
□ Show Only Compatible Skeletons

Make sure you have the similar retarget base pose. If they don't look alike here, you can edit your base pose in the Retarget Manager window to look alike.

I now needed to create a new anim_BP for the Alien. This was done by making a copy of the ‘ThirdPerson_AnimBP’. This was renamed to ‘Alien1_AnimBP’.

The animations would now be added in the ‘Alien1_IdleRun_2D_Copy’ with the ‘Mutant_Idle’ animation at speed 0 and Mutant_Run’ at the max speed.

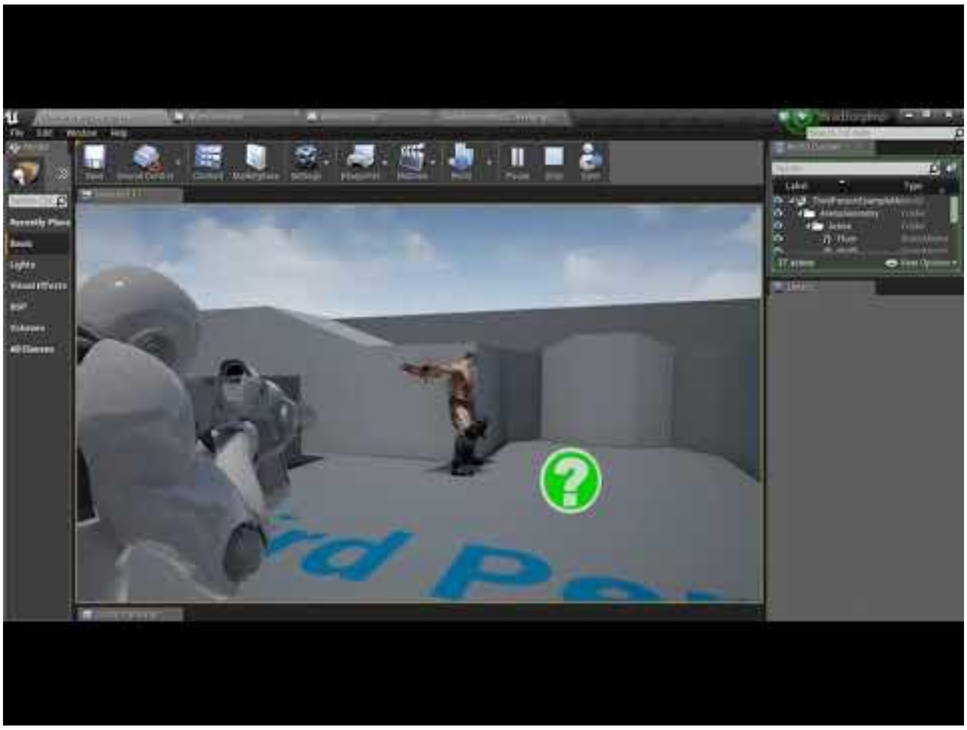


Previewing Blend Space Alien1_IdleRun_2D_Copy

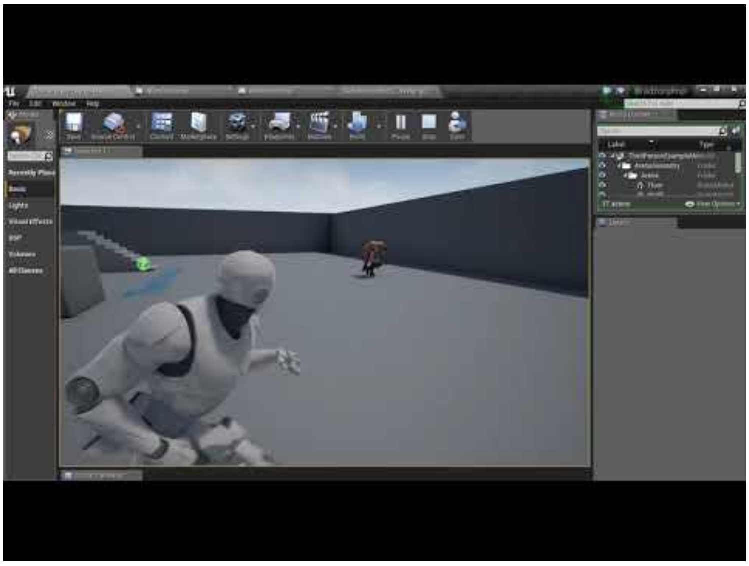
LOD: 0
Current Screen Size: 0.20
Triangles: 12486
Vertices: 7451
UV Channels: 1
Approx Size: 117x113x185

Speed [273.368]

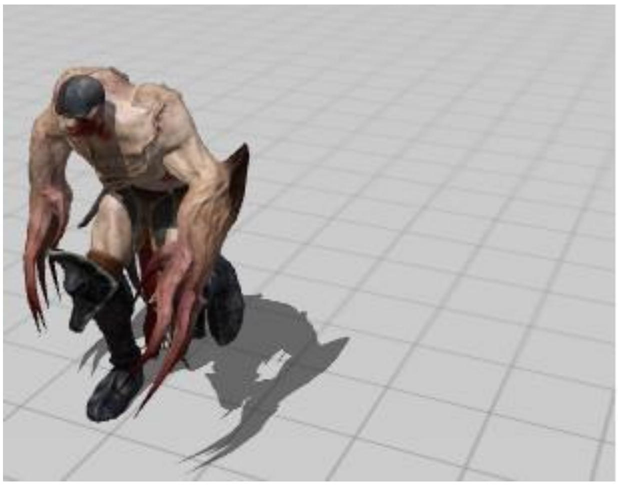
This should have created a smooth animation for the enemy however, the enemy would just slide around while it plays the idle animation.

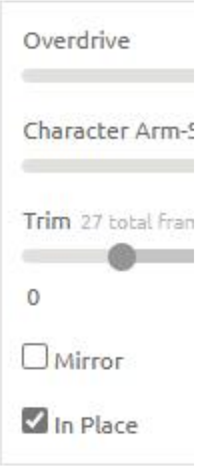


So, I thought if I just set the enemy to be running constantly by deleting the idle animation from the ‘Alien1_IdleRun_2D_Copy’. This made the enemy worse because the animation would move forward then reset to where it originally was when it loops back to the beginning of the ‘Mutant_Run’ animation giving a lag effect.



I spent a while trying to fix this, so I went back to Mixamo and noticed there was an option for ‘In Place’ so I redownloaded the run and an additional animation of the enemy walking animation with this ‘In place’ option ticked.





Overdrive

Character Arm-S

Trim 27 total fran

0

☐ Mirror

☑ In Place

I then imported these to the ‘Skeletonzombie_T_Avelange__1__Skeleton’ and replaced the old version of the animations with the new ones however, with the idle, walk and run animations all in use it would still slide around in the idle animation. So, I took that out for now for later re-implementation. My character will just walk around now. The issue seemed to be a character speed issue as the enemy would only play one animation because the speed was not changing.

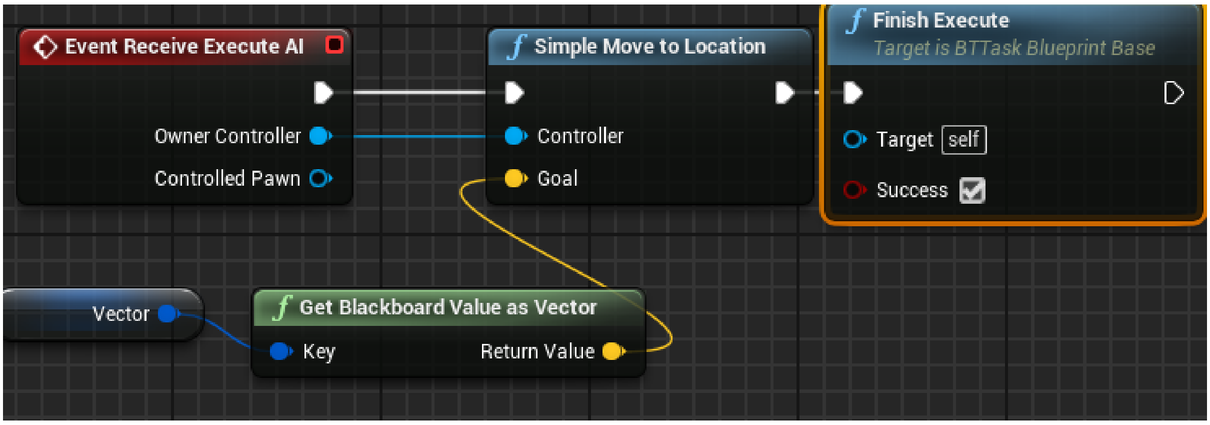
This was the end of my reference video ‘Creating a Zombie AI in UE4 [4.26] / Perception Refinements, Mesh & Animations [Part 4]’ (O, Odyssey 2020)

New Player Model, Smooth Rotation and Player Chasing:

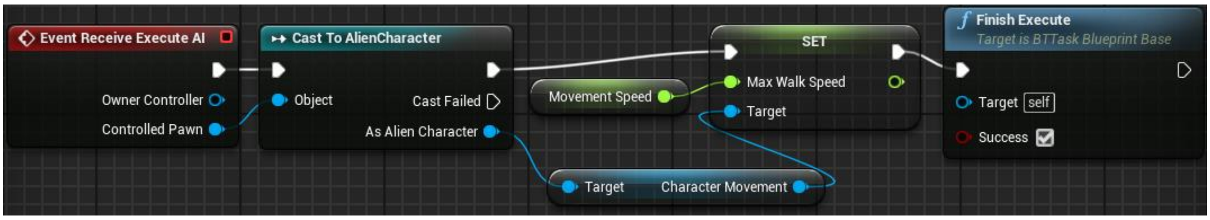
In this section I will be showing how I fixed the animations for the alien character, adding different movement states, setting up a new chase mechanic and adding a new player mesh.

So, the first thing I started on here was tidying up the behaviour tree by creating two new tasks one named ‘BTTask_BlueprintBase_ChasePlayer’ and another named ‘BTTask_BlueprintBase_UpdateMovementSpeed’.

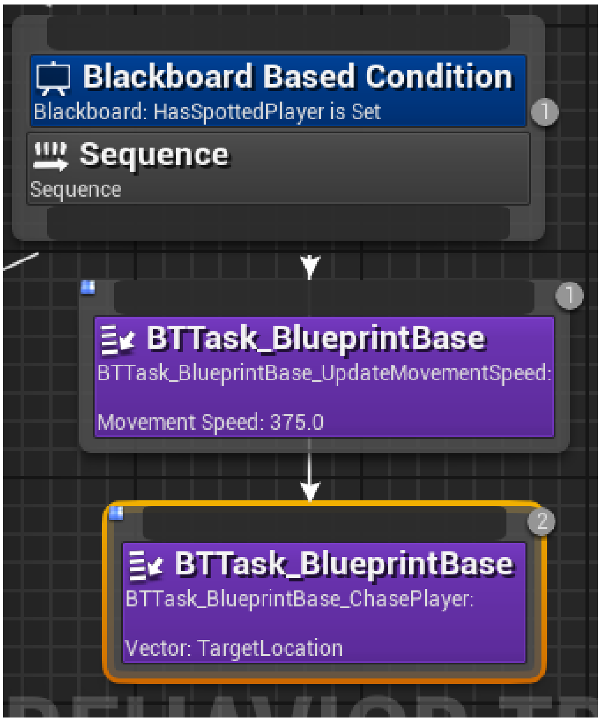
This is the logic for ‘BTTask_BlueprintBase_ChasePlayer’.



And this is the logic for ‘BTTask_BlueprintBase_UpdateMovementSpeed’.

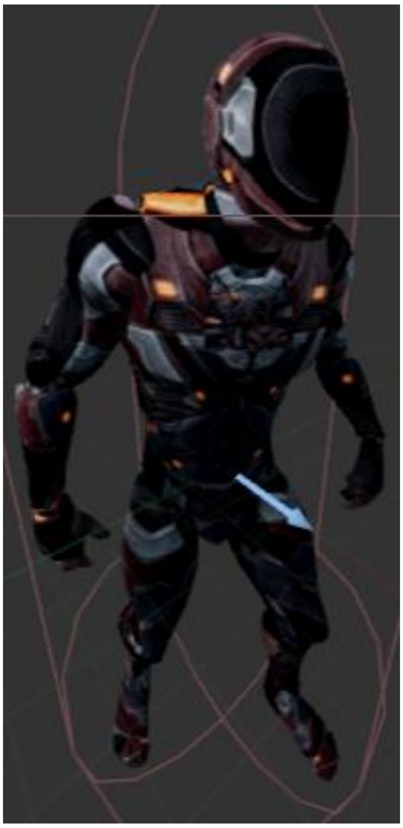


These were then updated in the behaviour tree with the ‘Chase Player’ task set to ‘TargetLocation’. This should have allowed the run animation to play when the enemy spotted the player and played the walk animation when the enemy can’t find the player, but it wasn’t working, and I couldn’t figure out why.

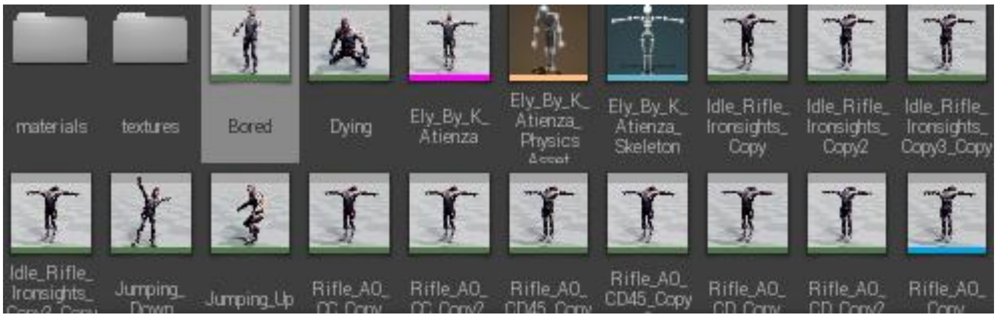

Blackboard Based Condition
Blackboard: HasSpottedPlayer is Set
1
Sequence
Sequence
1
BTTask_BlueprintBase
BTTask_BlueprintBase_UpdateMovementSpeed:
Movement Speed: 375.0
2
BTTask_BlueprintBase
BTTask_BlueprintBase_ChasePlayer:
Vector: TargetLocation

It was now time to add a new mesh to my playable character, so it wasn't just a white model however following the tutorial messed up my socket, Actor and Aim offset logic that was created in the third person character section, so I took the textures from the model I downloaded from Mixamo.

And applied the texture to the 'ThirdPersonCharacter' mesh. This ended up looking better than intended and good enough to keep as it is.



However, I imported and retargeted the mixamo mesh and he is there if I can find an actor, socket and aim offset fix.

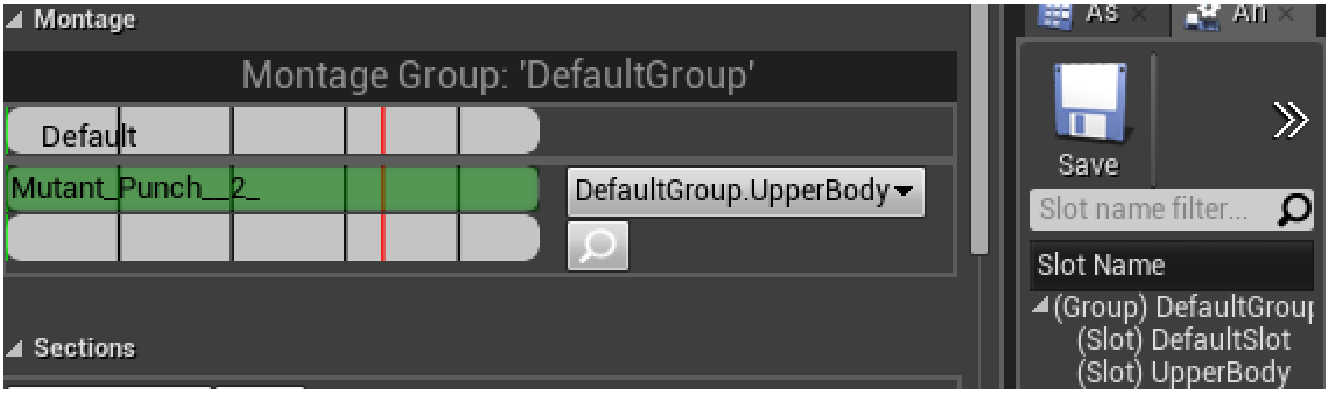


This was the end of my reference video 'Creating a Zombie AI in UE4 [4.26] / New Player Model, Smooth Rotation and Player Chasing [Part 5]' (O, Odyssey 2020)

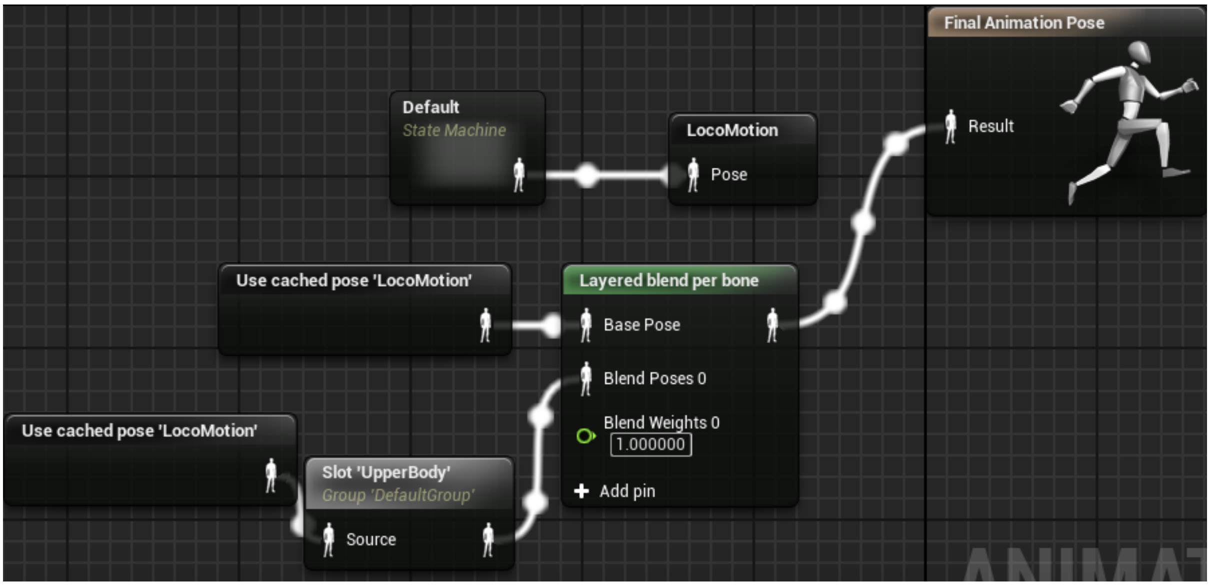
Creating the Attack Logic:

For this section I will be integrating my attack animation into the enemy AI. This was done by creating a 'Montage' and adding my attack animation 'Mutant_Punch'

I then added a new 'Slot' within this named 'UpperBody'

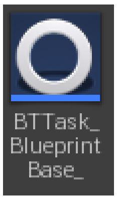


This new logic was the created.

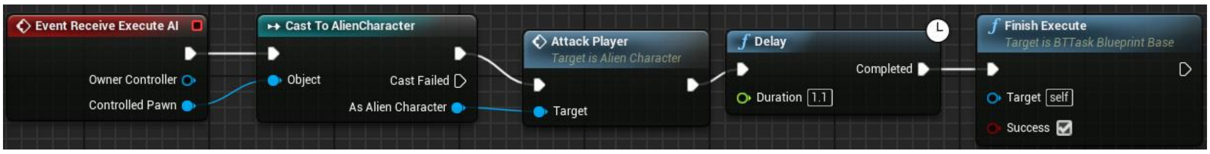


Next, I needed to make a new 'Key' in the 'Alien_BB'. It was set as a 'Boolean' named 'PlayerIsInRange'

It was now time to make a new 'Task' in the 'Alien_BT' named 'BTTask_BlueprintBase_AttackPlayer'



This logic was then created for that.

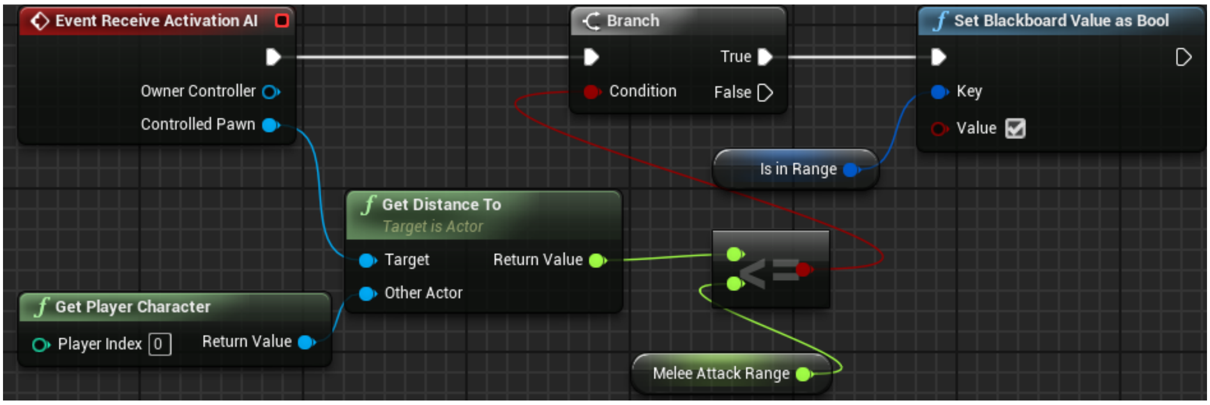


A new 'service' was then created named 'BTService_BlueprintBase_PlayerIsInRange'

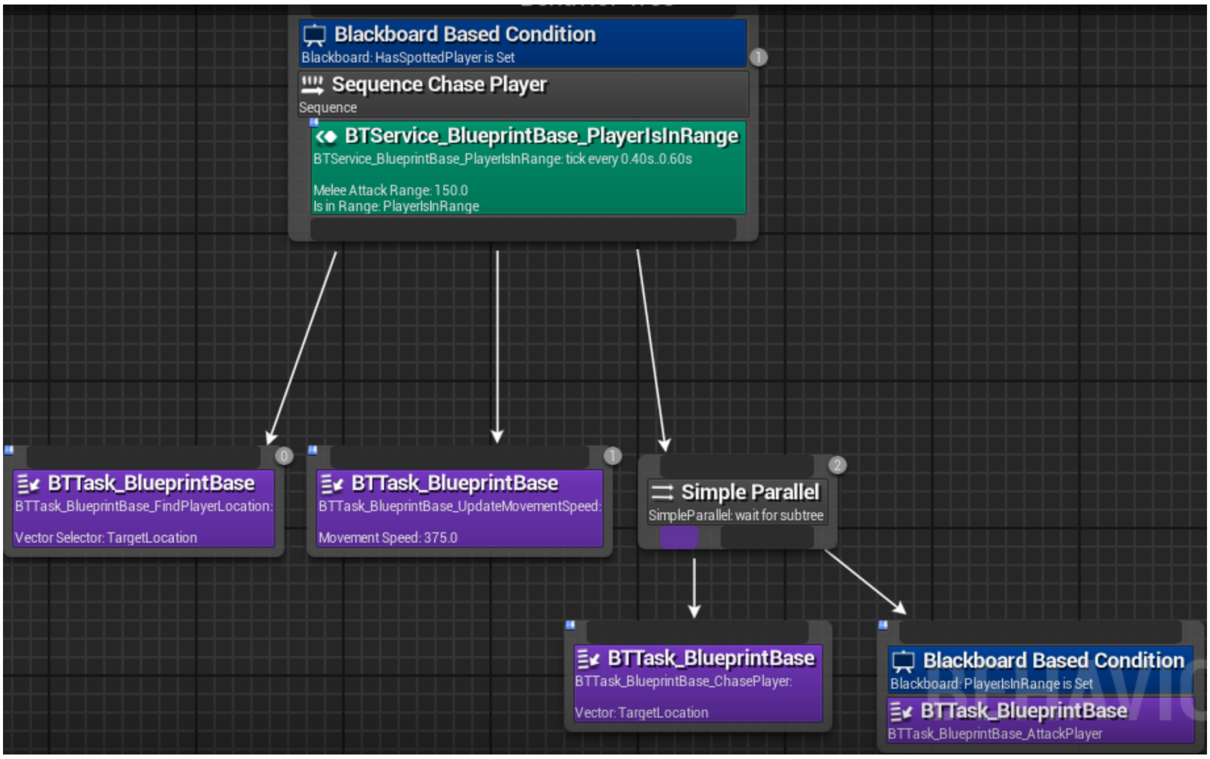
Two new variables were then created, the first being a ‘float’ named ‘MeleeAttackRange’

And the second being a ‘BlackBoardKeySelector’ named ‘IsInRange’

This logic was then created.



It was then time to add this to the behaviour tree.


Blackboard Based Condition
Blackboard: HasSpottedPlayer is Set

Sequence Chase Player
Sequence

BTService_BlueprintBase_PlayerIsInRange
BTService_BlueprintBase_PlayerIsInRange: tick every 0.40s..0.60s

Melee Attack Range: 150.0
Is in Range: PlayerIsInRange

0 BTTask_BlueprintBase
BTTask_BlueprintBase_FindPlayerLocation:

Vector Selector: TargetLocation

1 BTTask_BlueprintBase
BTTask_BlueprintBase_UpdateMovementSpeed:

Movement Speed: 375.0

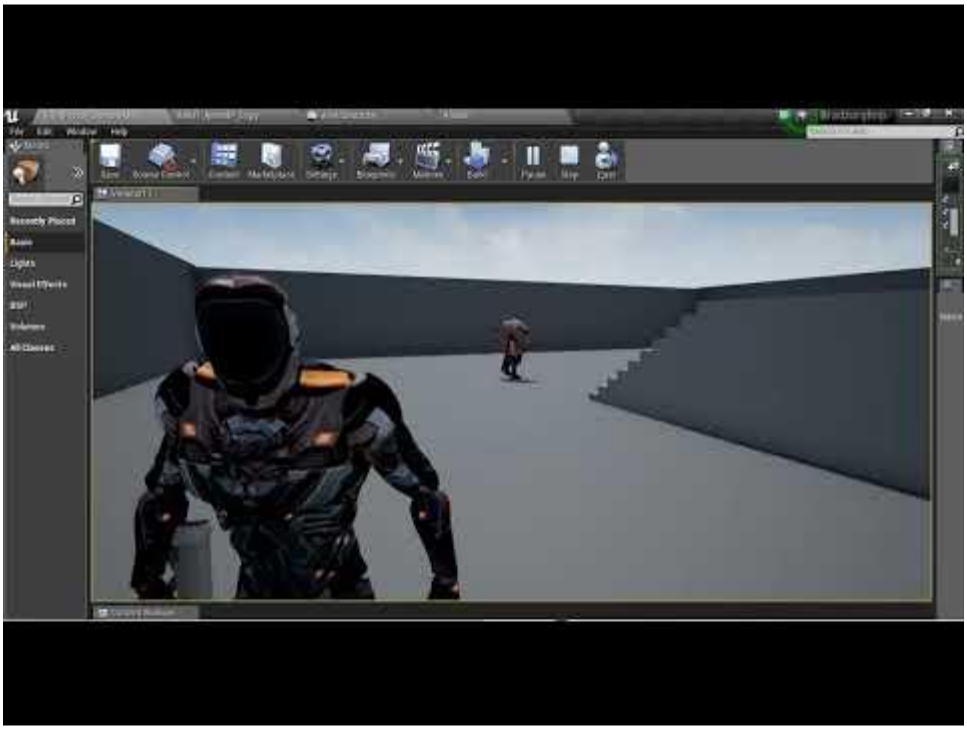
2 Simple Parallel
SimpleParallel: wait for subtree

BTTask_BlueprintBase
BTTask_BlueprintBase_ChasePlayer:

Vector: TargetLocation

Blackboard Based Condition
Blackboard: PlayerIsInRange is Set
BTTask_BlueprintBase
BTTask_BlueprintBase_AttackPlayer

This is the end of my reference video ‘Creating a Zombie AI in UE4 [4.26] / Creating the Attack Logic’ [Part 6] (O, Odyssey 2020)



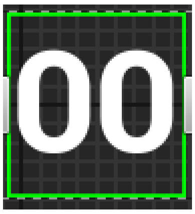
Adding the HUD and Health:

in this section I will be adding a heads-up display and creating a health system.

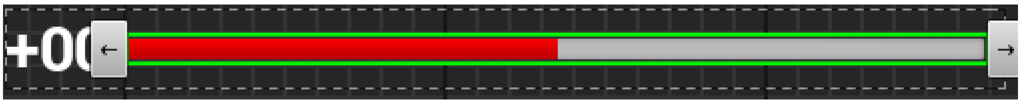
The first step was to create a HUD this was done by firstly adding a ‘Widget BluePrint’ named ‘WBP_HUD’ in the ‘ThirdPerson_BP’

Within this I added two new textboxes aligned at the top left of the screen. One to signify that the value being shown is health

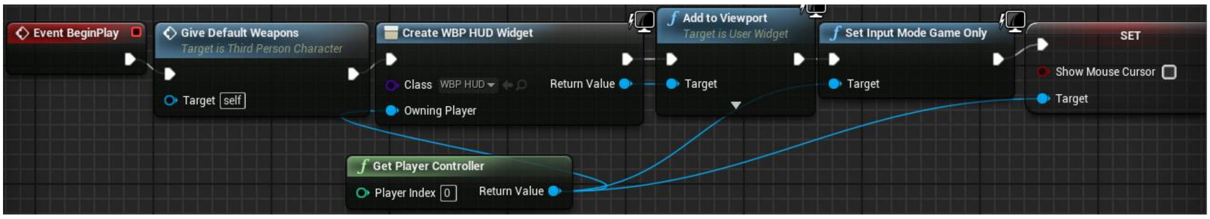
And another being for the 0-100 health ‘float’.



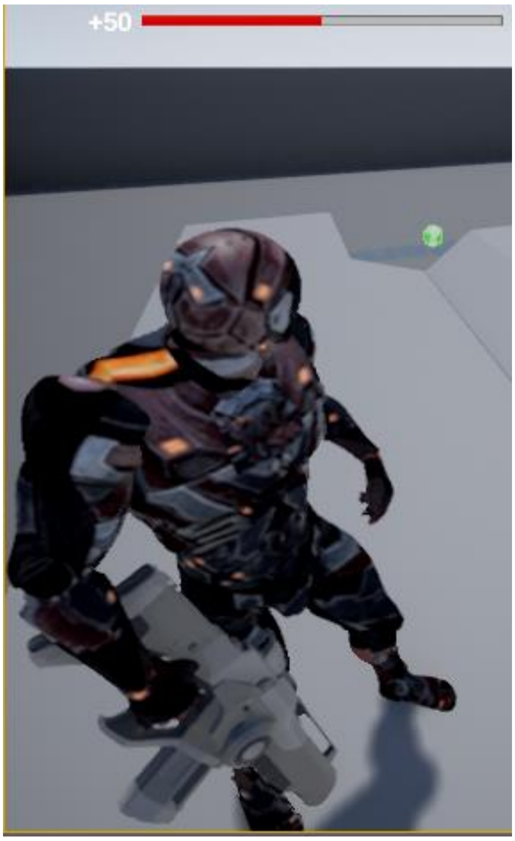
Lastly, I needed to add a ‘progress bar’. I made this red to let the player know that this is their health bar.



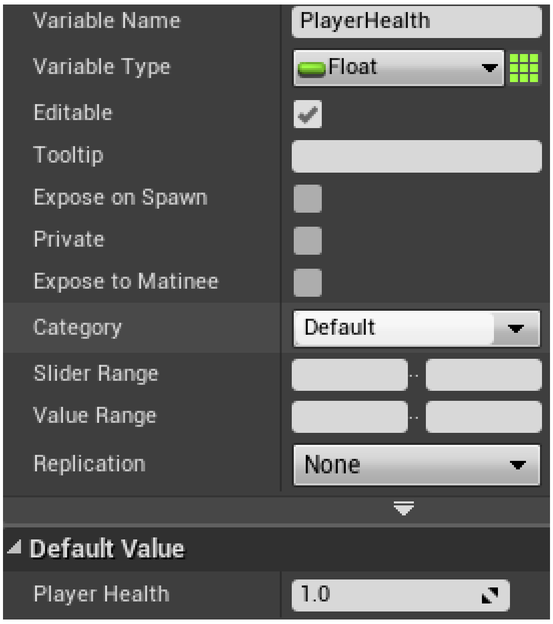
Now in the ‘ThirdPersonCharacter – Events Graph’ I created this logic to draw the ‘HUD’ on the player’s screen.



This would now show up on the players screen when play testing.



however, it would be stuck to half-filled this was because it needed to be binded to a newly created ‘Float’ variable named ‘PlayerHealth’ with a default value of ‘1’.

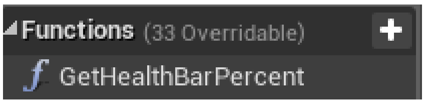


Variable Name	PlayerHealth
Variable Type	Float
Editable	☑
Tooltip	
Expose on Spawn	□
Private	□
Expose to Matinee	□
Category	Default
Slider Range	..
Value Range	..
Replication	None

Default Value

Player Health	1.0

Once binded unreal automatically made a new function



Functions (33 Overridable) +

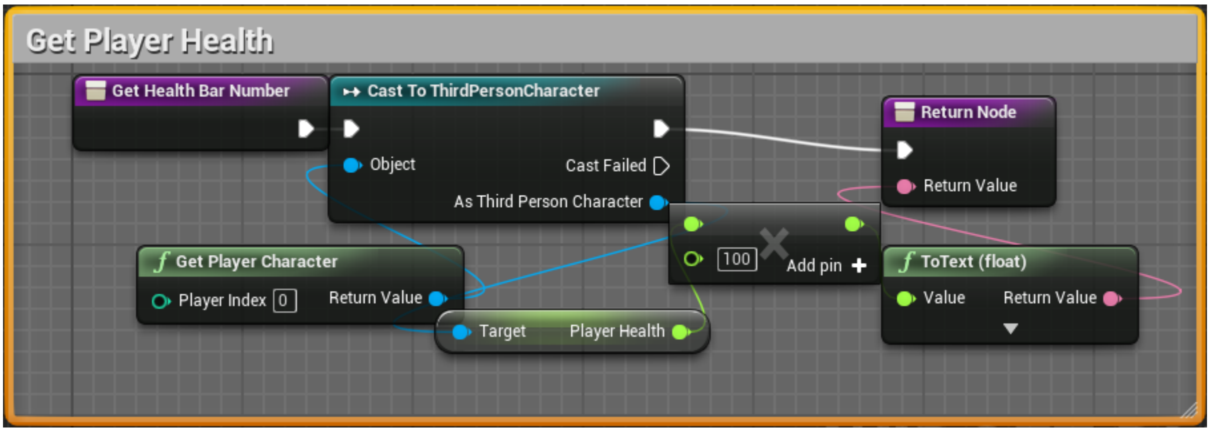
f GetHealthBarPercent

And a new variable.

Variables +

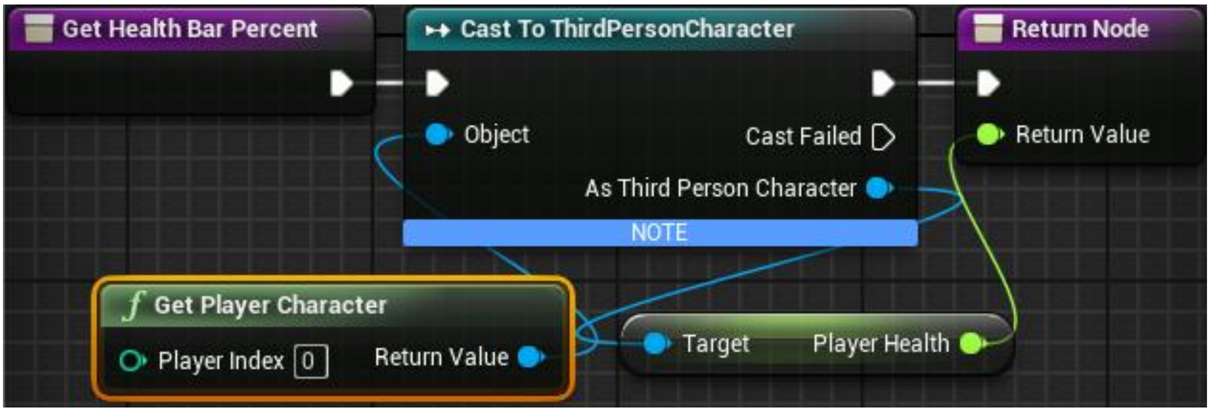
ProgressBar_0

Within the ‘GetHealthBarPercent’ function this logic was created.

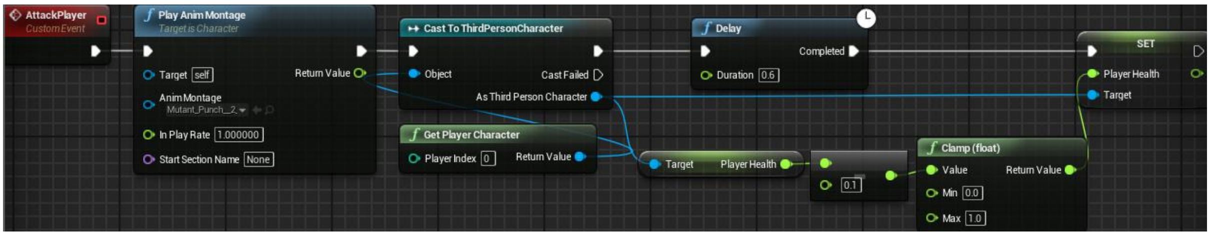


Now I needed to do the same binding process to the 0-100 text box. This created a new function named 'GetHealthBarNumber'

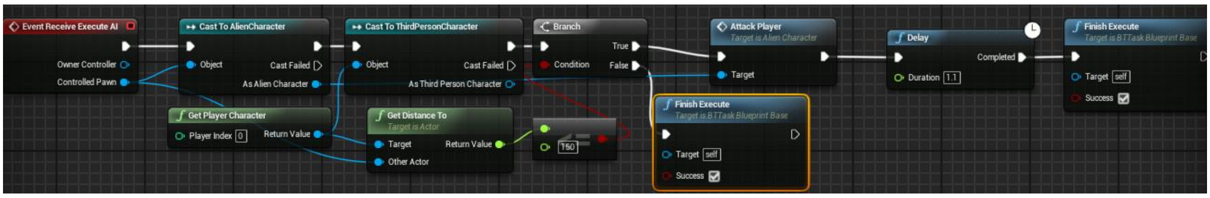
I then created this logic within this new function.



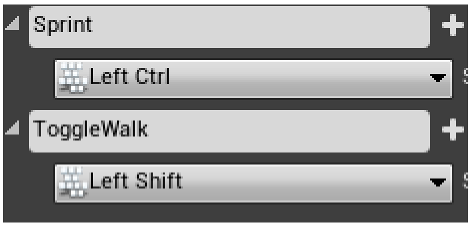
It was now time to make the enemy take health off the player. That was done with this logic created in the 'AlienCharacter – Events Graph'.



However, once the enemy started to play the attack animation it would not stop even when the player was out of range. This not only didn't look right but it would also take health off the player each time the attack animation played. I changed the logic to this, and it fixed it.

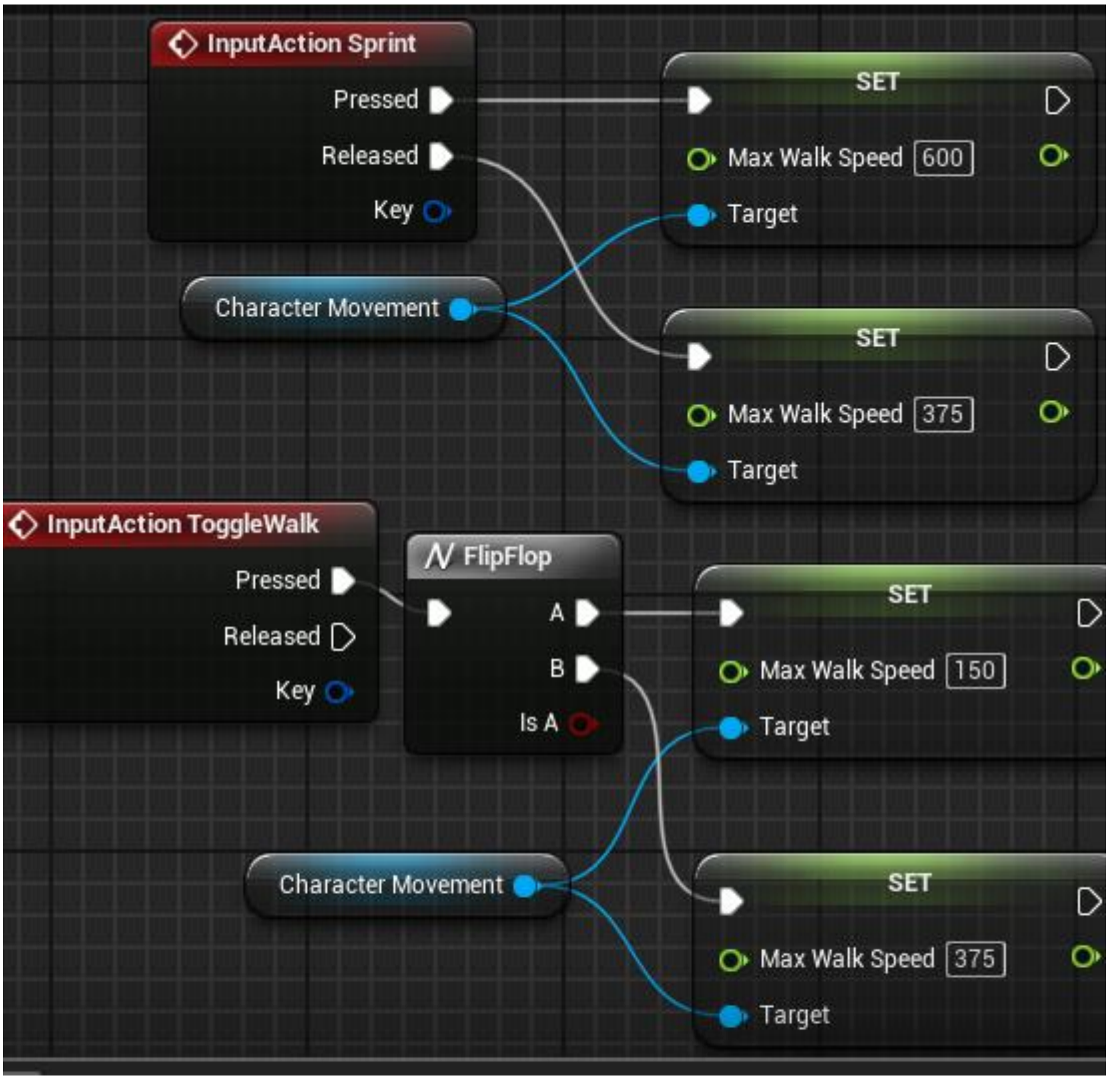


The last thing to do before ending this reference video was to add a sprint and walk input to the player character. This was done by creating new inputs in the project settings.

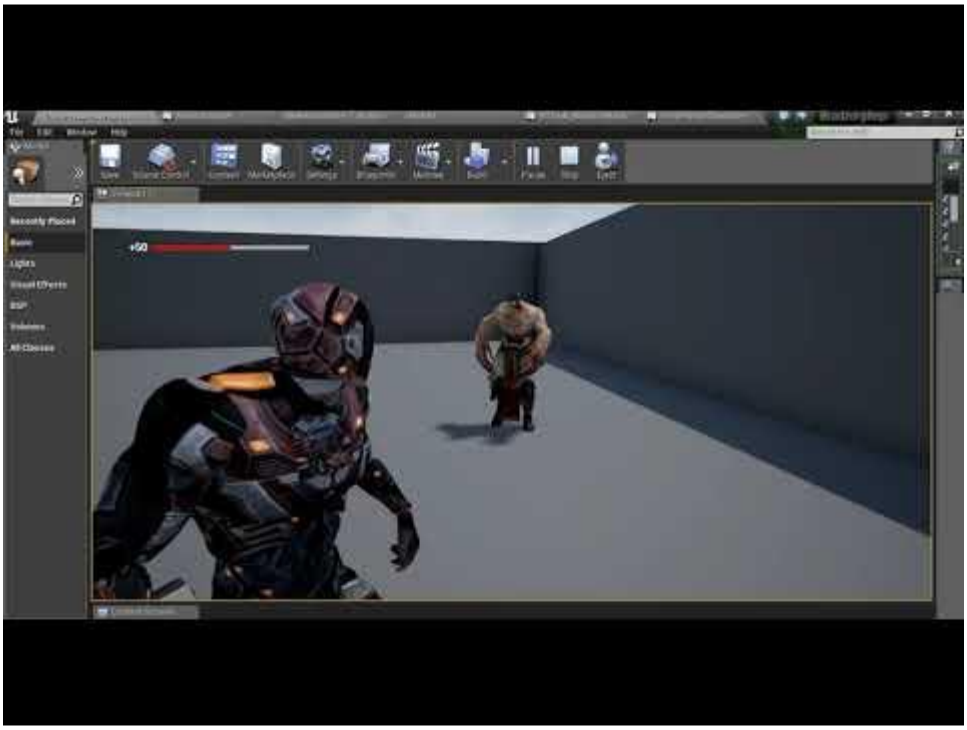


Sprint
Left Ctrl
ToggleWalk
Left Shift

This logic was then created for these two systems.

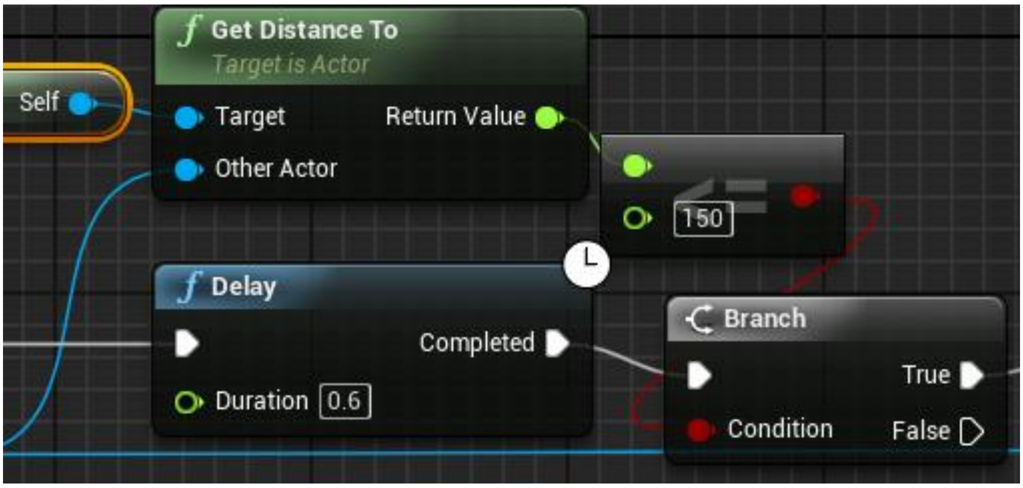


This is the end of my reference video ‘Creating a Zombie AI in UE4 [4.26] / Adding the HUD and Health [Part 7]’ (O, Odyssey 2020)



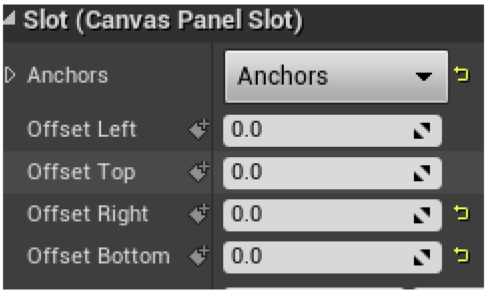
The Death Screen, Health Packs and Ragdoll:

Before moving onto the next section, I needed to fix the enemy damage range because when the enemy attack animation played it would take the players health. However, this caused the player to take damage even when they are not within range to the enemy for it to make sense and it just looked awful. So, I added this new logic to the ‘AlienCharacter’ to fix it.



Now it was time to create a death screen for when the players health reaches zero. To do this I firstly needed to crate a new ‘Widget’ in the ‘ThirdPersonCharacter_BP’ named ‘WBP_DeathScreen’

Within this a boarder was added and anchored to cover all the available space.



Slot (Canvas Panel Slot)

Anchors: Anchors

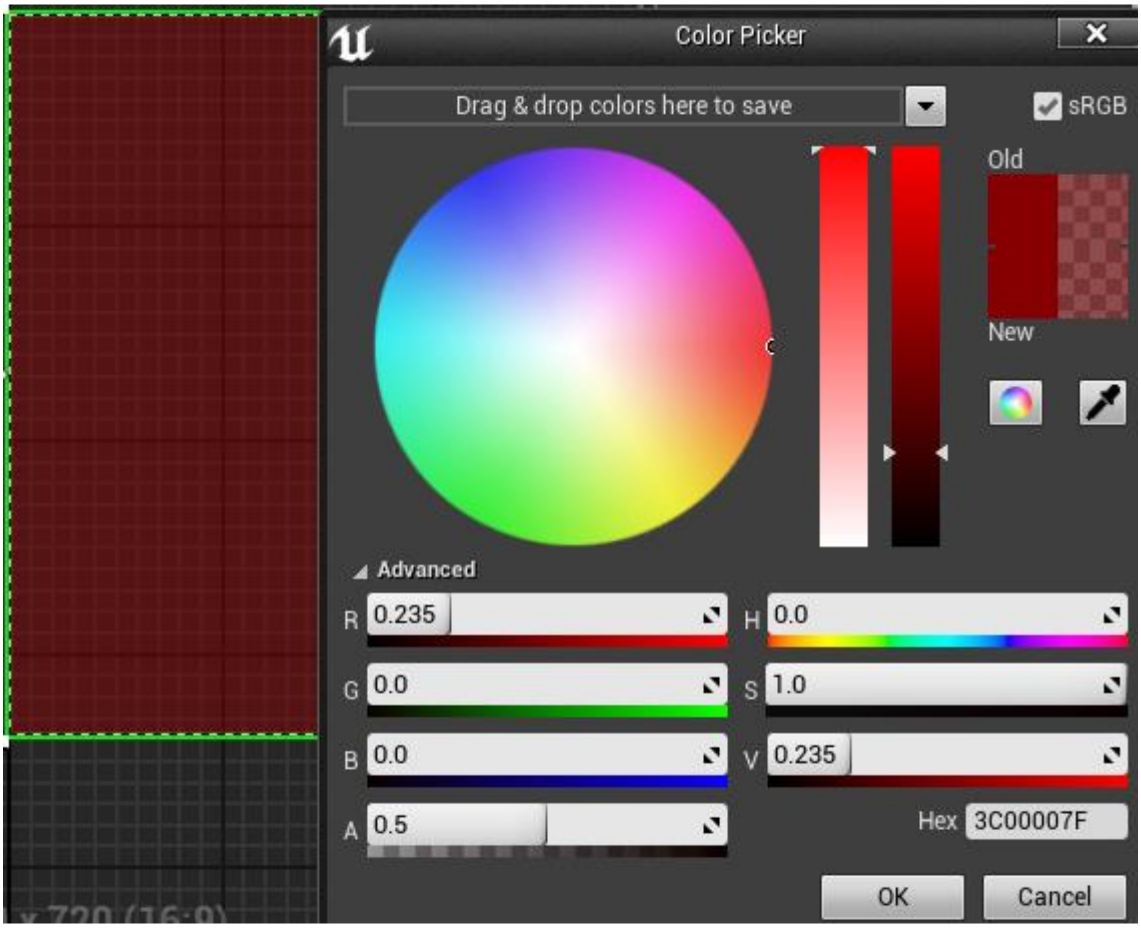
Offset Left: 0.0

Offset Top: 0.0

Offset Right: 0.0

Offset Bottom: 0.0

This was then set to a red tint to signify player Blood/Death.



☑ sRGB

R 0.235

G 0.0

B 0.0

A 0.5

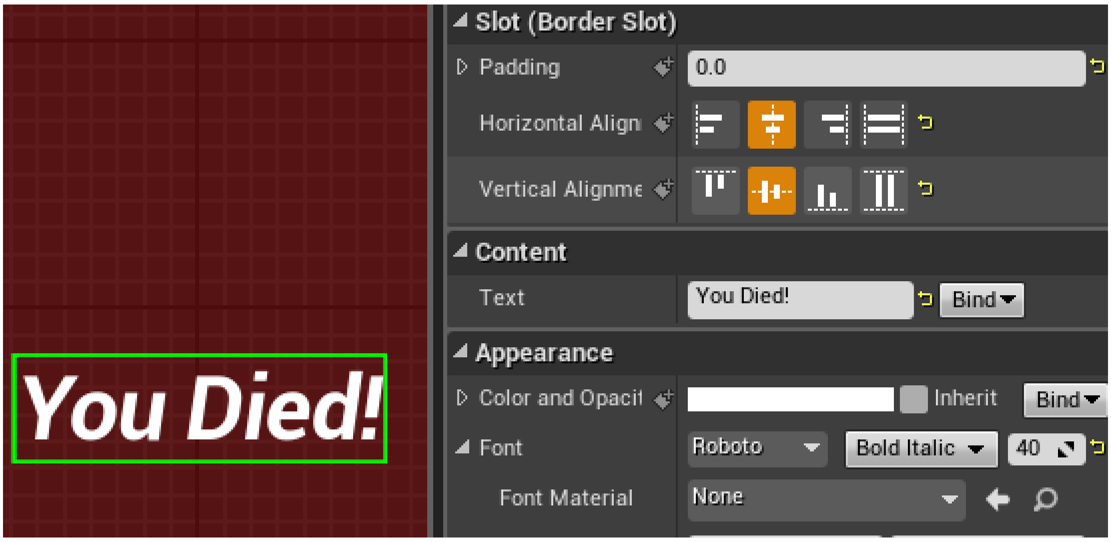
H 0.0

S 1.0

V 0.235

Hex 3C00007F

Next, I added a text box that says, ‘You Died!’ and aligned it to the middle of the boarder.



You Died!

Slot (Border Slot)

Padding 0.0

Horizontal Align

Vertical Alignme

Content

Text You Died! Bind

Appearance

Color and Opacit Inherit Bind

Font Roboto Bold Italic 40

Font Material None

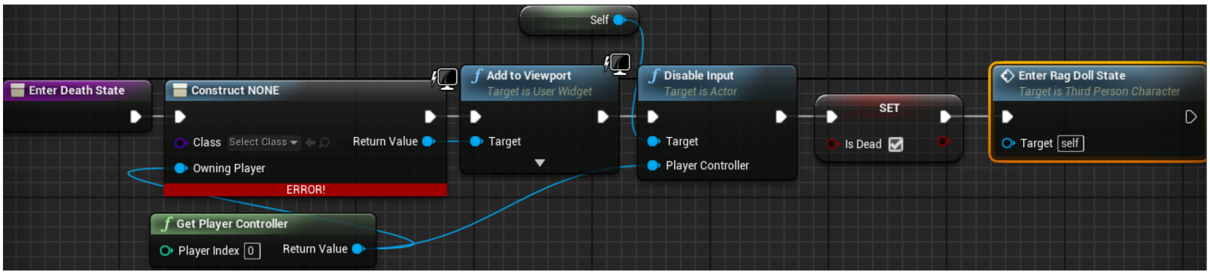
Now within the ‘ThirdPersonCharacter’ a new function was created named ‘EnterDeathState’.

EnterDeathState

A new ‘Boolean’ variable was then created within the new function named ‘IsDead’.

IsDead

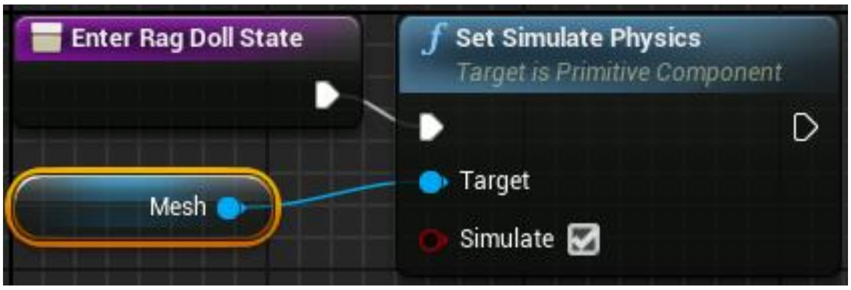
This logic was then created.



A second new function was then created named ‘EnterRagDollState’.

EnterRagDollState

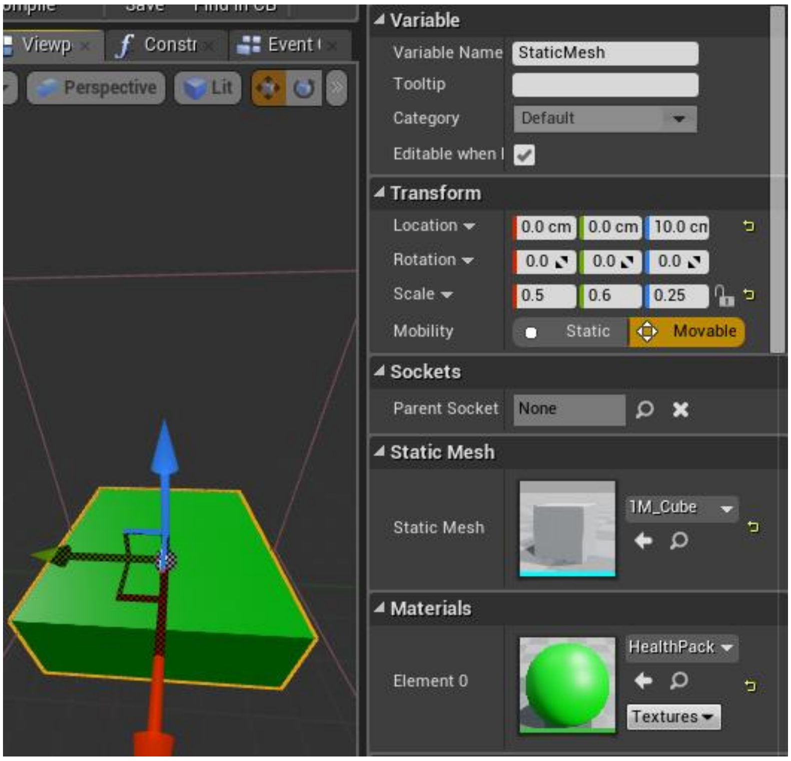
Then this logic was created within that.



This should have made the character ragdoll as the player died however it does not work and I can’t figure out how to fix it. This is probably because I’m on a later version of unreal than the tutorial that I am following. The Death screen still takes away player input though so this will have to do for now.

Now it was time to create a med kit for the player to regain health. To do this firsly I created a new ‘BluePrint Class’ in the ‘BluePrint’ folder named ‘MedKit’.

Within this a ‘box mesh’ was created with a green ‘material’ over it. This is how the medkit will look.



Variable
Variable Name: StaticMesh
Tooltip:
Category: Default
Editable when I ☑

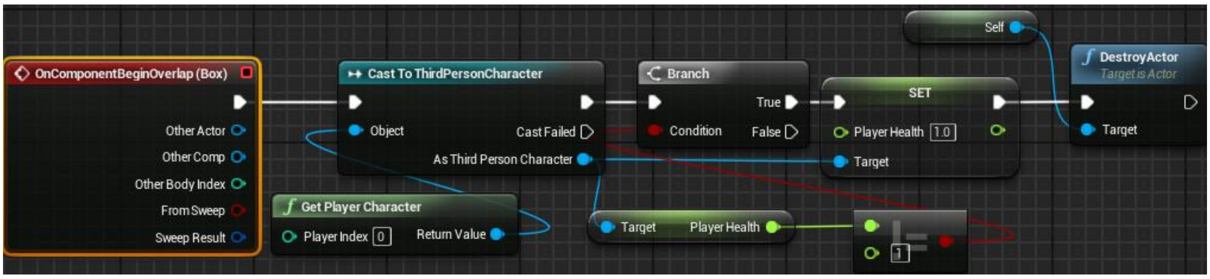
Transform
Location: 0.0 cm | 0.0 cm | 10.0 cm
Rotation: 0.0 | 0.0 | 0.0
Scale: 0.5 | 0.6 | 0.25
Mobility: Static | Movable

Sockets
Parent Socket: None

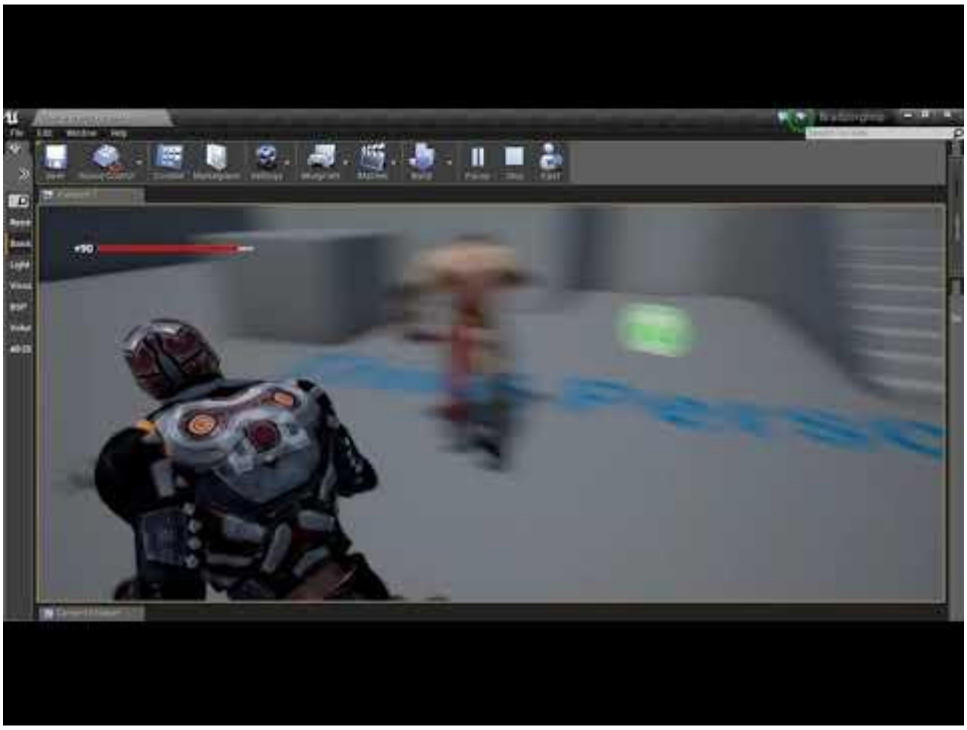
Static Mesh
Static Mesh: 1M_Cube

Materials
Element 0: HealthPack
Textures

This logic was then created for the medkit to give the player health.



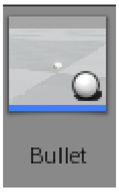
The medkits were then dragged into the environment. This is the result of my reference video 'Creating a Zombie AI in UE4 [4.26] / The Death Screen, Health Packs and Ragdoll! [Part 8]' (O, Odyssey 2020)



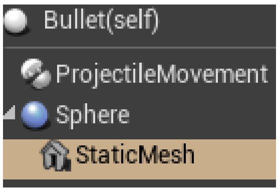
Adding the bullet and enemy health system:

Now at the end of these two reference playlists there was still a bullet ‘Actor’ missing and an enemy health system.

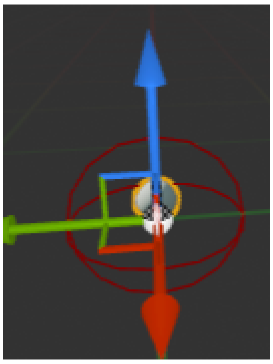
The first step was to create a new ‘Blueprint’ named ‘Bullet’ to act as the projectile that will damage the enemy.



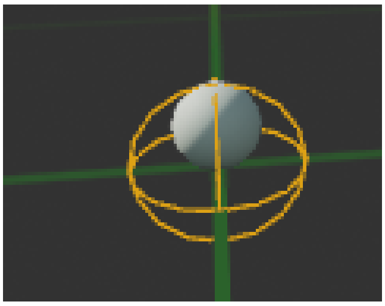
Within this I first needed to add three new components, one being a ‘static mesh’ another being a sphere and lastly a ‘Projectile movement’ component.



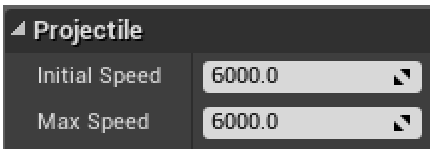
The next step was to set the scale needed for the ‘static mesh’.



Next, I needed to set the Sphere Radius to encase the static mesh. In this case the radius was set to ‘2’.



After that the initial and max speeds of the 'Projectile movement' were set to 6000 each.

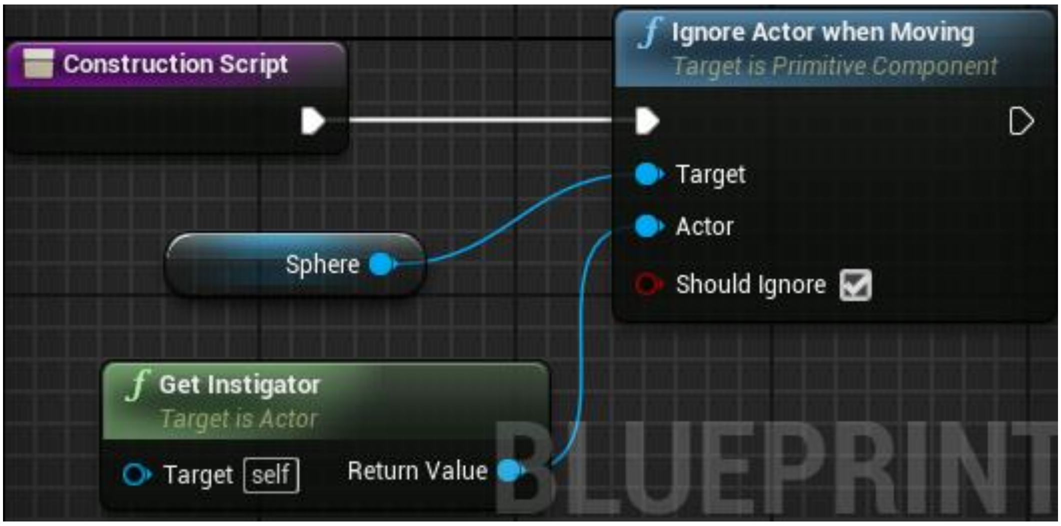


Projectile

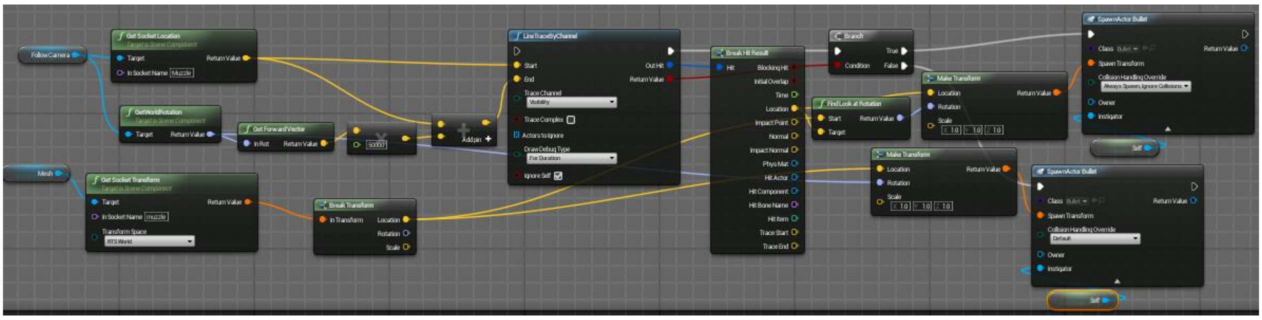
Initial Speed 6000.0

Max Speed 6000.0

It was now time to add some logic into the 'construction script' to allow the bullet to know what character owns it, in this case it is the 'Third Person Character'.

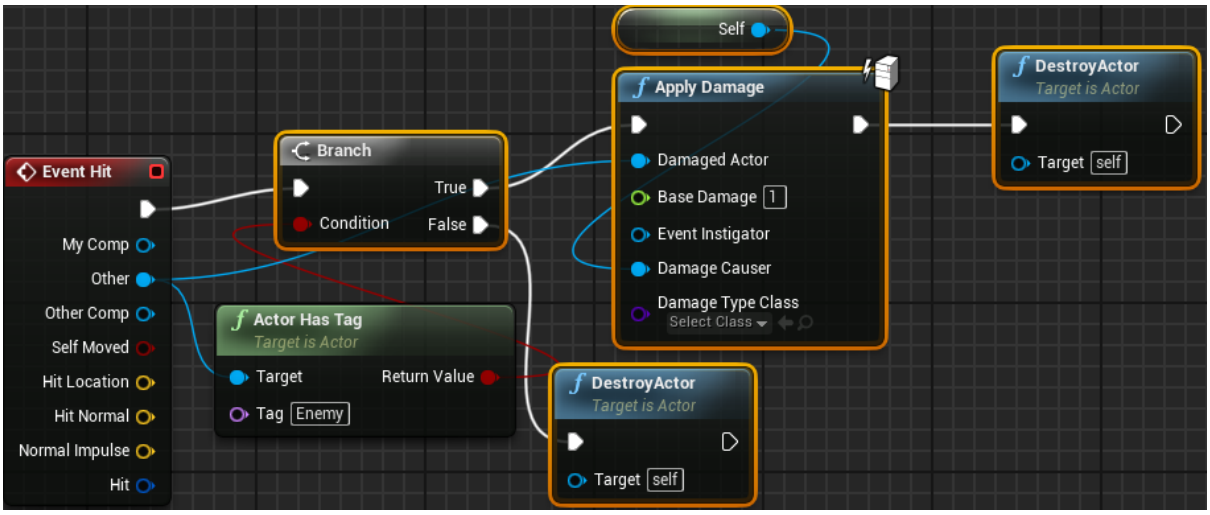


It was now time to create new logic in the 'ThirdPersonCharacter' event graph to handle this new bullet blueprint in the sense of which direction the bullet should go in and where it spawns from to begin with.

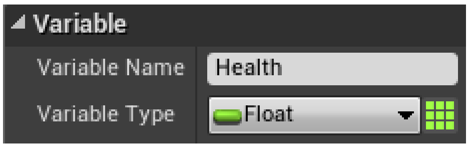


This was the end of my reference video 'Unreal Engine – Projectile Shooting – Part 29' (D, Shape 2016)

The final thing to do to the bullet was to create logic in the event graph to tell the bullet how much damage it will do then destroy itself. A tag named 'Enemy' was also added so the bullet will only damage a mesh if it has that tag.



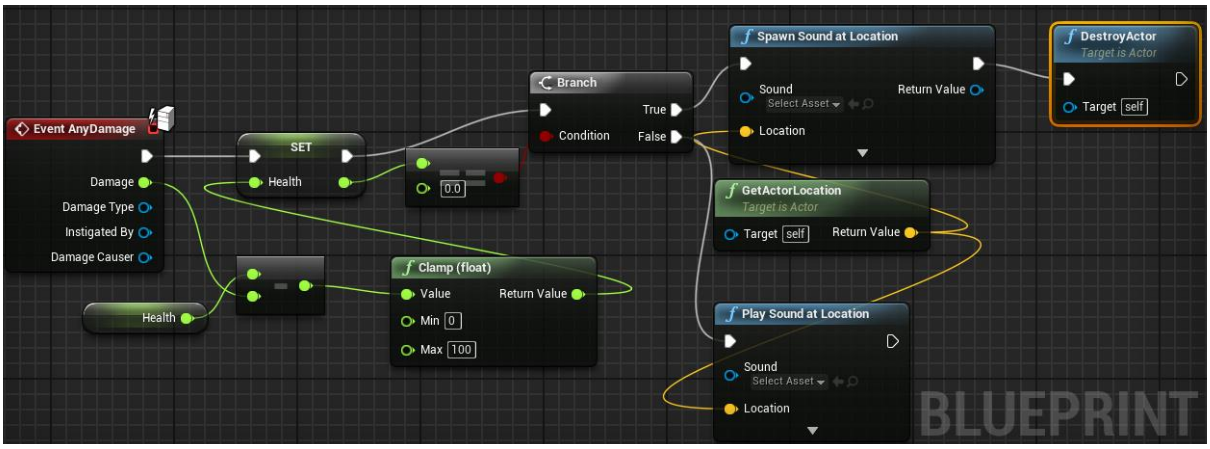
Now logic needed to be created in the 'AlienCharacter' event graph which allows the enemy to have health and when that health hits zero the enemy will be destroyed. For this a new 'Float' variable needed to be created named 'Health'


Variable

Variable Name: Health

Variable Type: Float

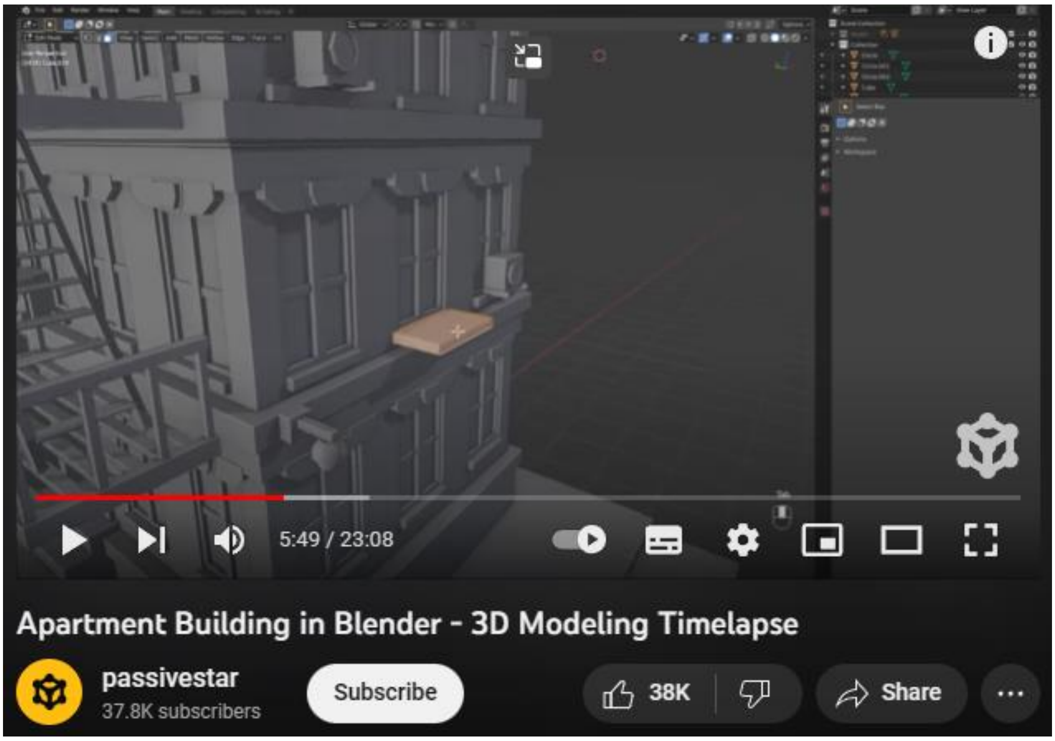
The rest of the logic was then created.



This was the end of my reference video 'Shooting And Killing Our Enemy AI / Shooter Minigame Part 5 – Unreal Engine Tutorial' (M, Aspland 2021)

3D Modelling:

In this section I will show how I made a lacklustre first attempt at modelling a 3D building in blender. First, I set out to watch a couple of blender tutorials them being ‘Apartment Building in Blender – 3D Modeling Timelapse’ (P, Passivestar 2020).

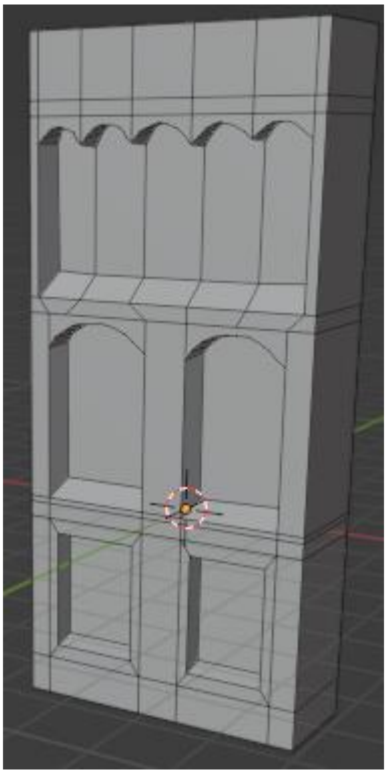


And ‘Learn Blender 3D in 20 Minutes! Blender Tutorial for Absolute Beginners (2023)’ (3, 3DGreenhorn 2021).

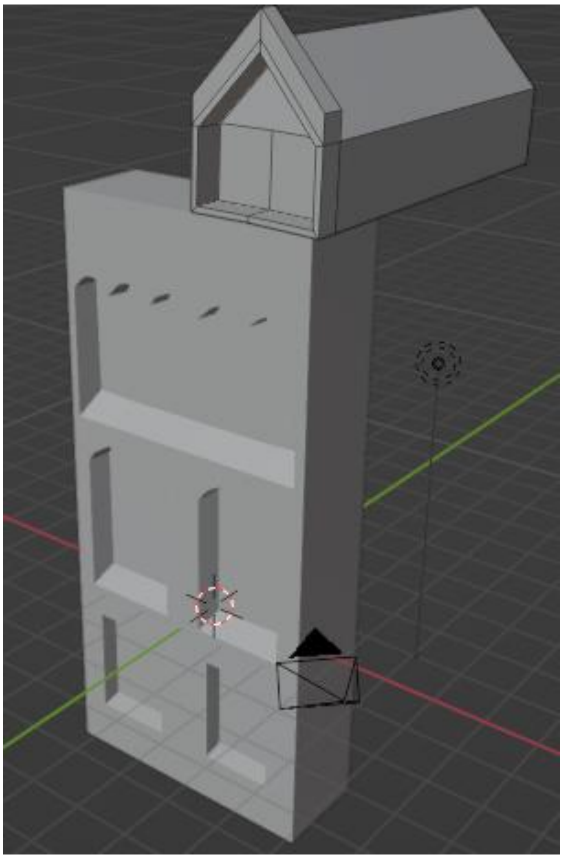
After watching these I started to attempt to model Bradfords town hall. The first thing to do was to get a reference photo of the building that I was modelling. I used this one for the alignment and the initial walls and windows.

and this one for the front of the building.

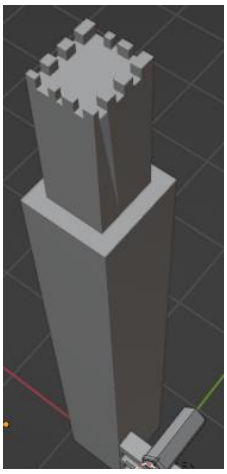
Now that I had images to reference while modelling, I created my first section that I intended to duplicate and place next to one another.



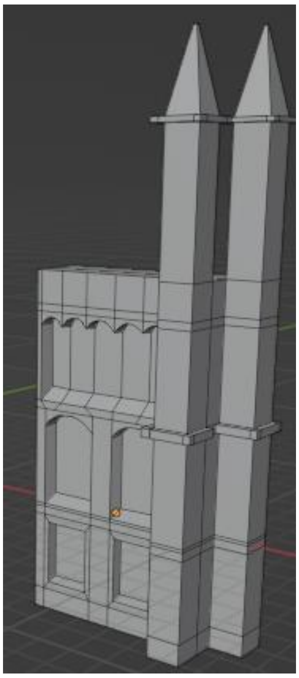
Next, I modelled the triangular sections on the top of the building.



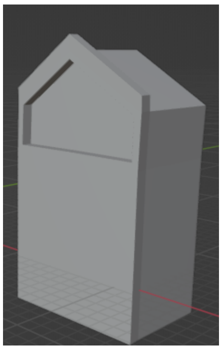
As these past two were to be duplicated I decided to make a simple tower to mark the front of the model.



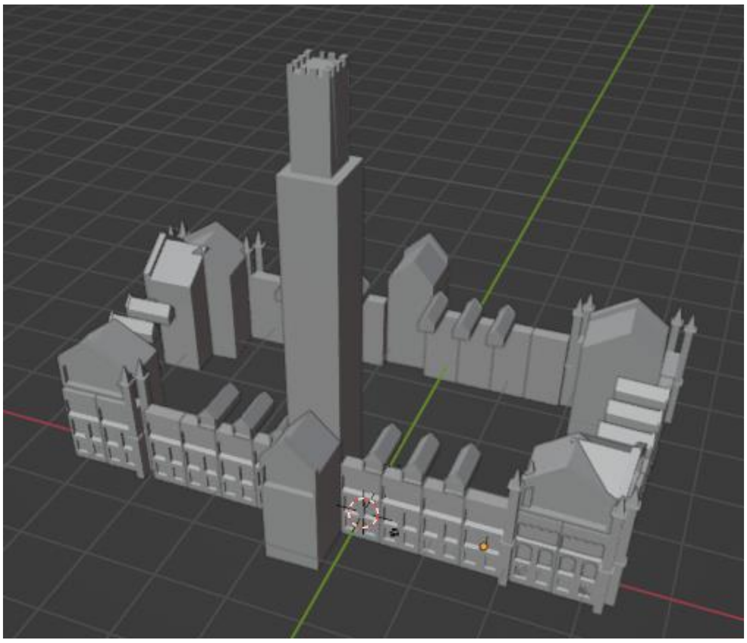
The triangular posts were then modelled out of the side of a copy of the first segment.



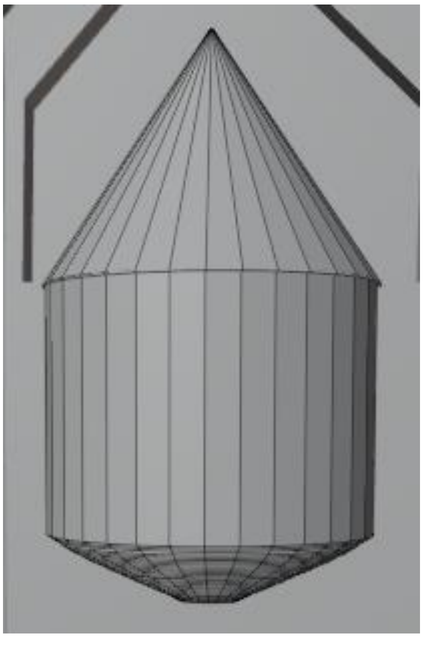
Next was the bigger section between the triangular posts.



I now had the building blocks to make a simple version of Bradfords town hall.

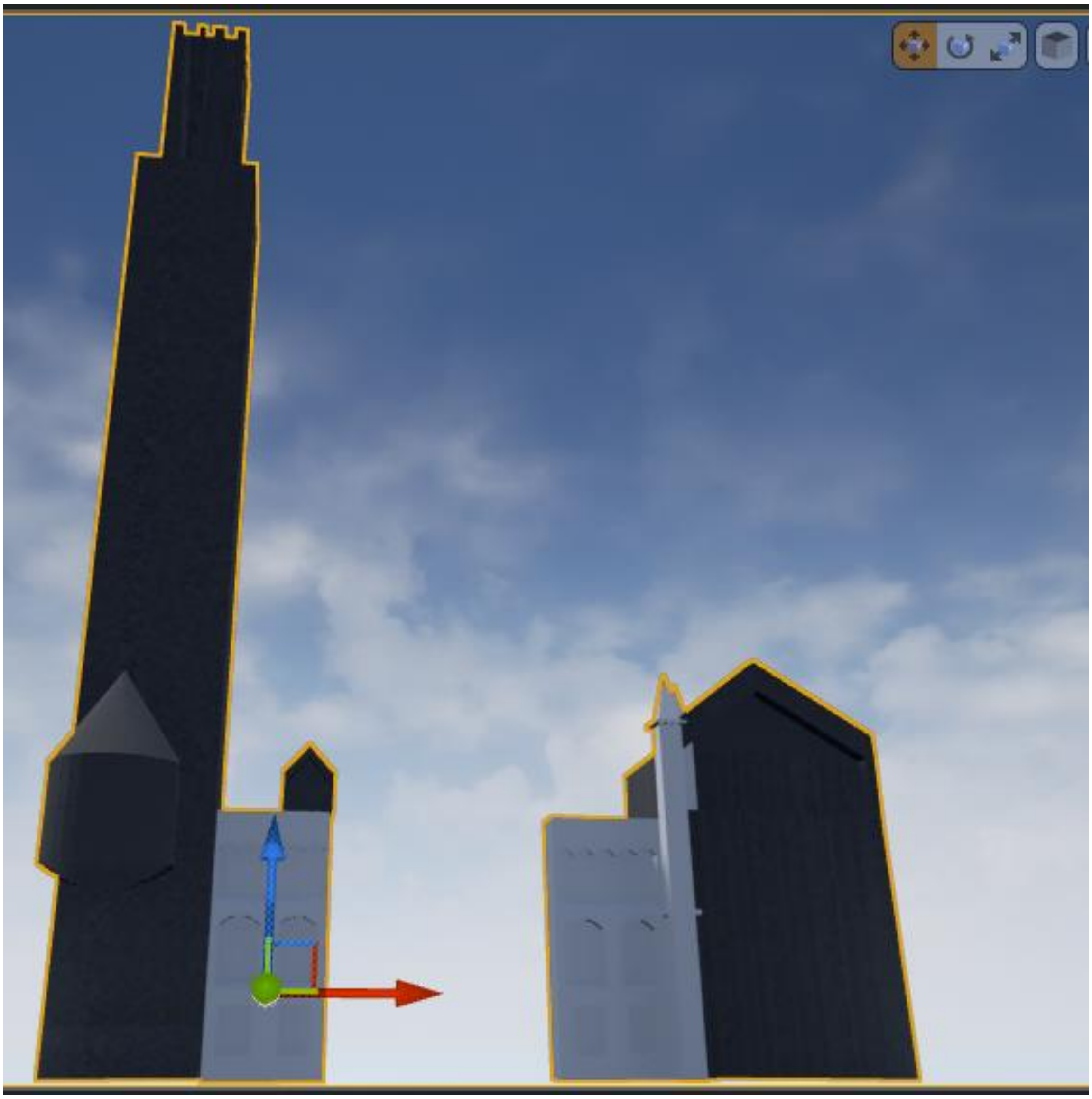


I then needed to seed shape for the front of the building.



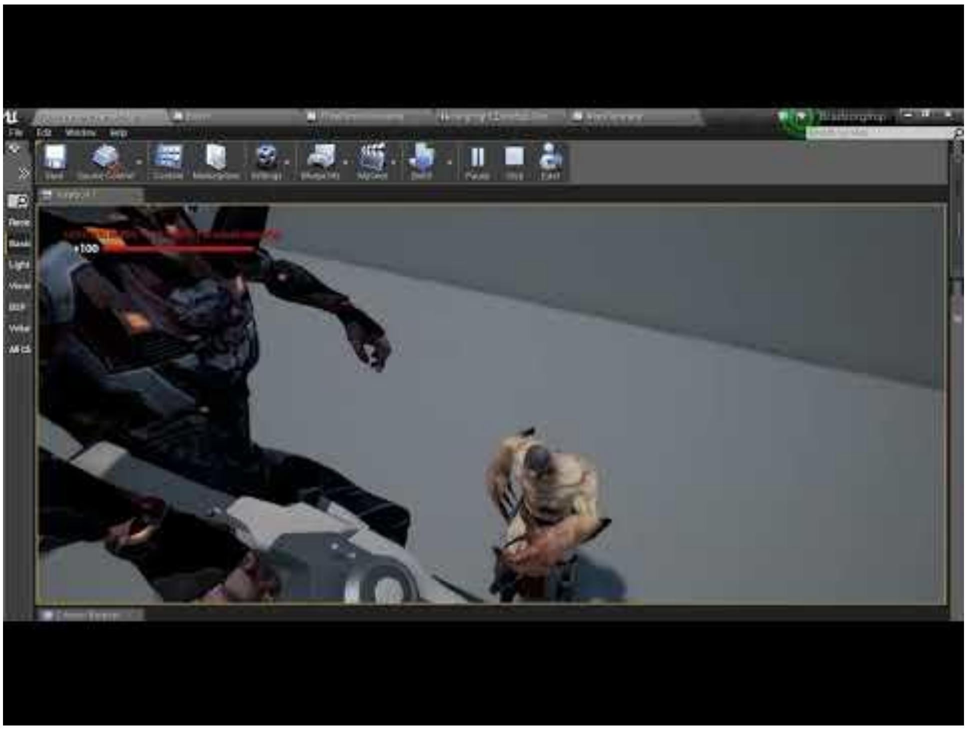
It was now time to add brick and glass textures to the model, so I researched how to do so by watching ‘How to TEXTURE in under 60 seconds.. (Blender)’ (R, RoBuilder 2021). However, after seeing this I concluded that it would probably be easier to texture the model in unreal engine. So, I used (How To Export/Import From Blender To Unreal Engine 4’. (M, Aspland 2020)

This was the point where I realised, I may have made a huge mistake when it comes to the fundamentals of blender as no duplicated layers were exported over and this is what I was left with.



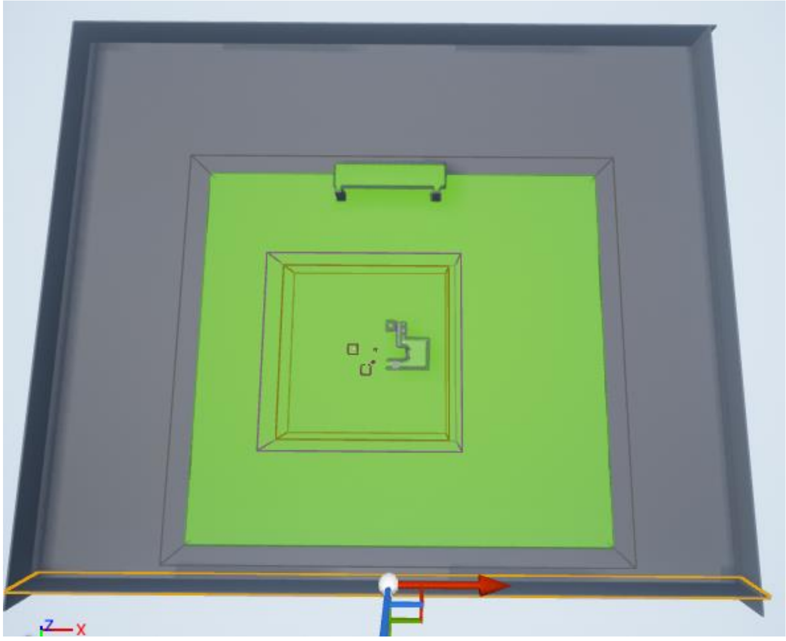
With time running out I decided it was better to make a game out of the tools that I already had and using unreal engines models to create a basic layout instead of modelling.

Before I started on the layout, I made some variable changes to the enemy. He is now faster, can spot the enemy more easily and has a faster attack. I did this to make the game a little harder to give it a little bit more replay ability.

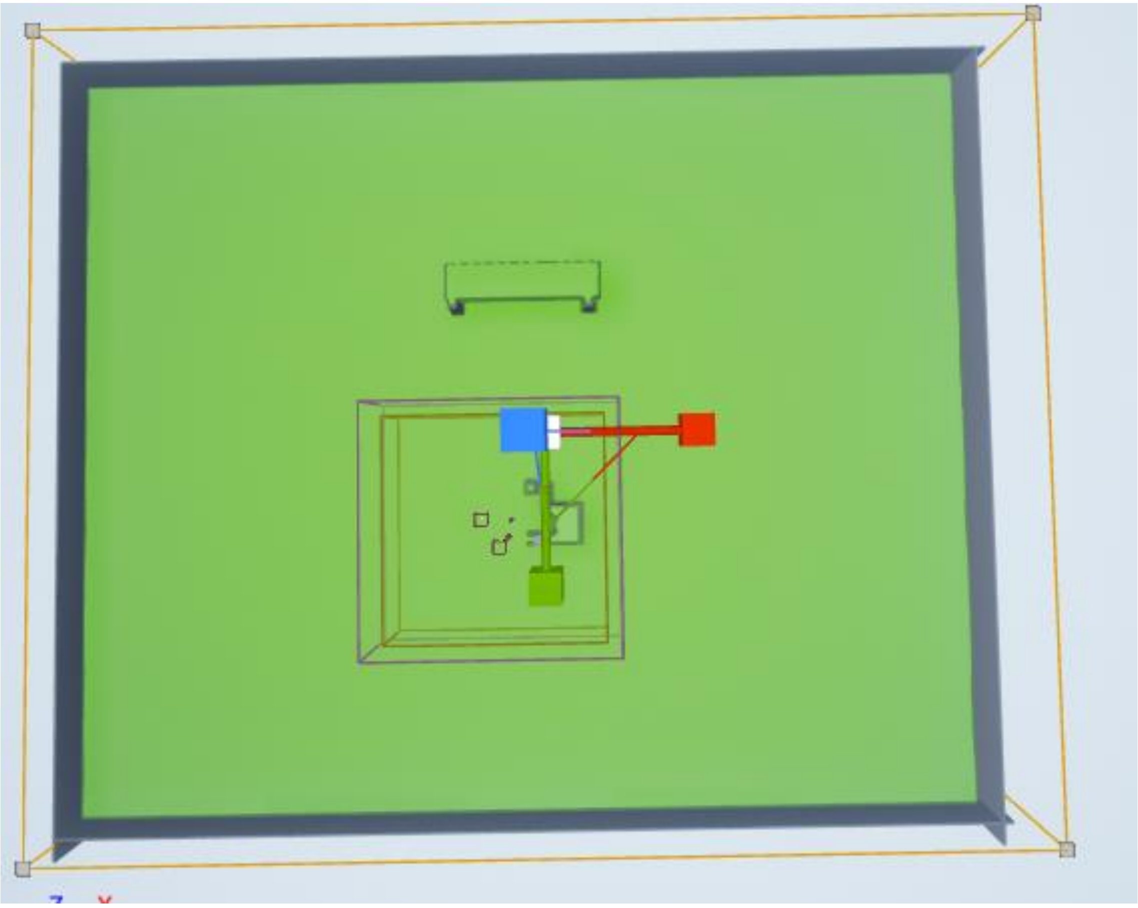


Level layout:

I decided to still make the basic layout of Bradford city centre just without the models. For now, I just put basic shapes where the buildings are supposed to be. I started off by expanding the playable area.

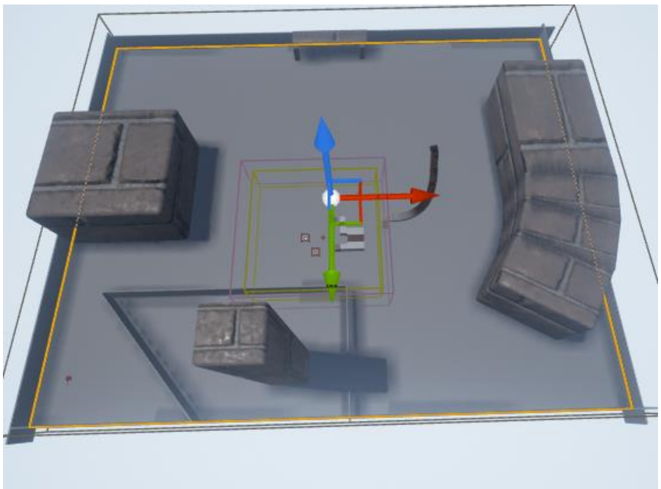


This was then followed by the ‘NavMeshBoundsVolume’. This allows the enemy to reach the expanded space.

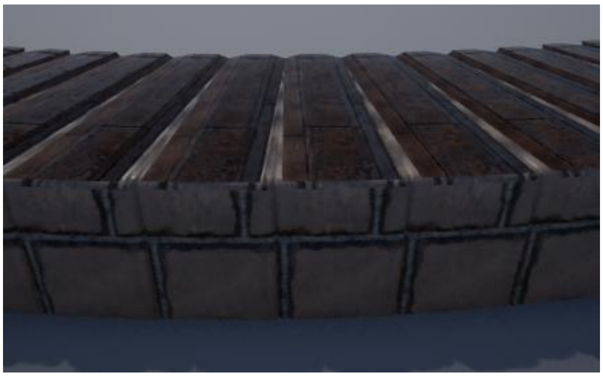


I then spent a while placing static meshs around to make the basic layout for the level. I used this image as reference for my layout.

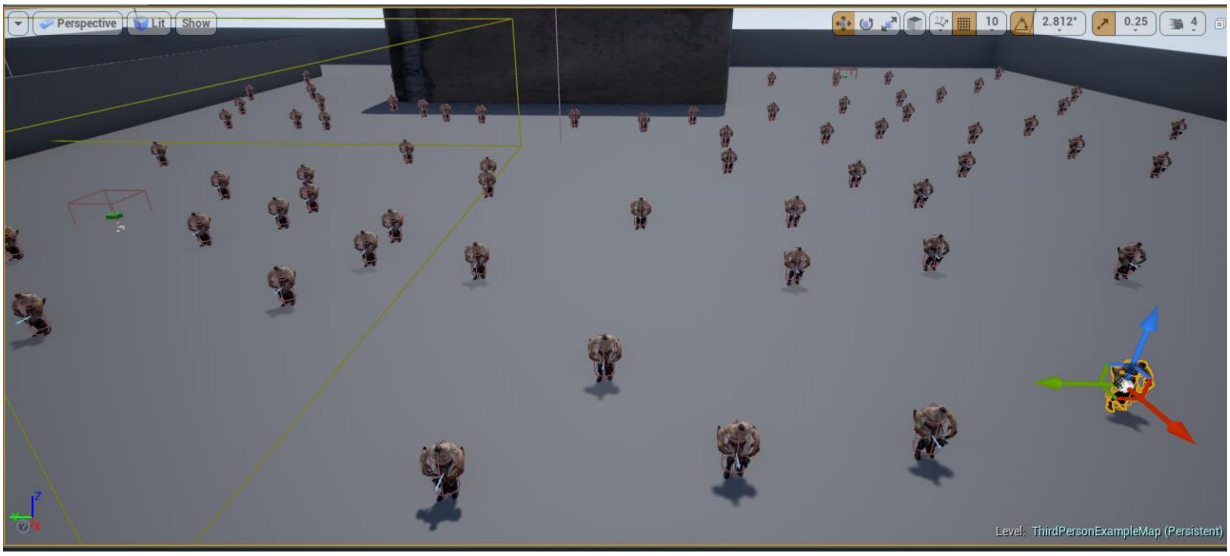
I added basic brick textures to the main four areas to signify that they should be buildings.



For the circle in the middle, I got a box with a stone texture and with a flattened mesh with a wood texture on it to combine them. I copied and pasted this with slight adjustments every time to make a more circle shape to make a wooden walkway.

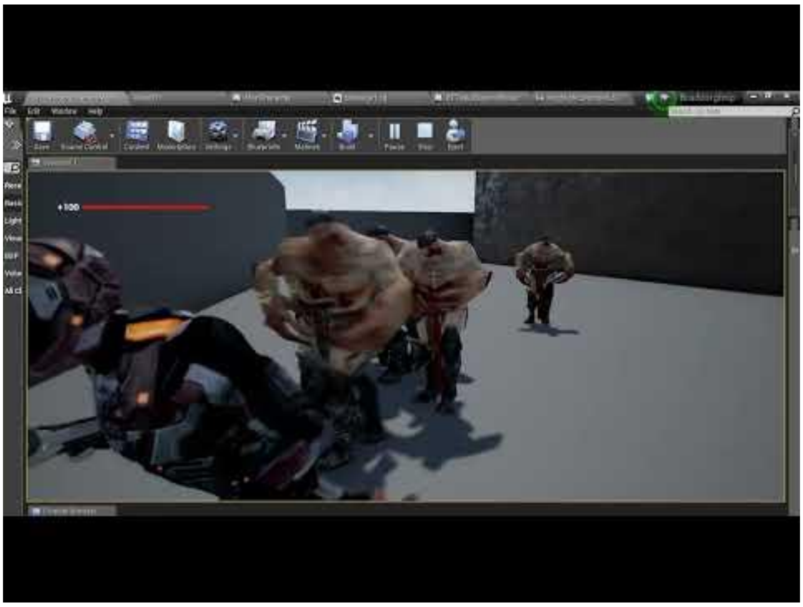


Now that I had a level layout, I needed to fill it with my ‘Enemy Character’.

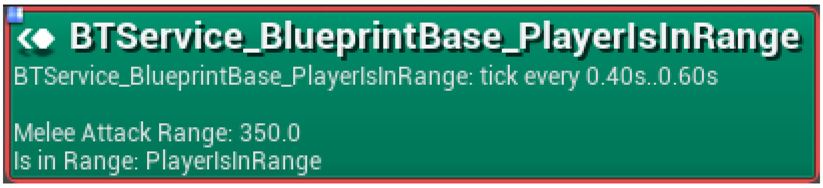


Level: ThirdPersonExampleMap (Persistent)

However, when I play tested it the enemies could not damage the player character without then being idle.



I concluded that the best way to fix this would be to give the enemy character a wider ‘attack range’. I did this by gradually increasing some variables that were set earlier in development until I felt like it was more of a realistic attack range for the enemy to have. One in the ‘Behaviour tree’,

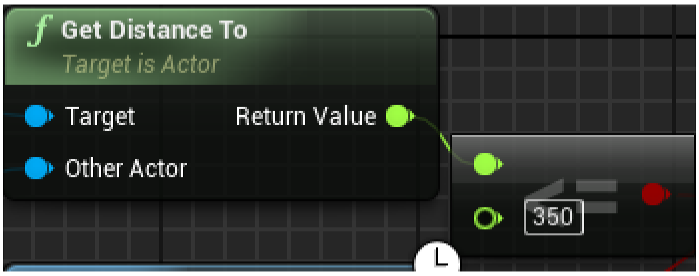


BTService_BlueprintBase_PlayerIsInRange

BTService_BlueprintBase_PlayerIsInRange: tick every 0.40s..0.60s

Melee Attack Range: 350.0
Is in Range: PlayerIsInRange

Another in the ‘Alien Character – Event Graph’,



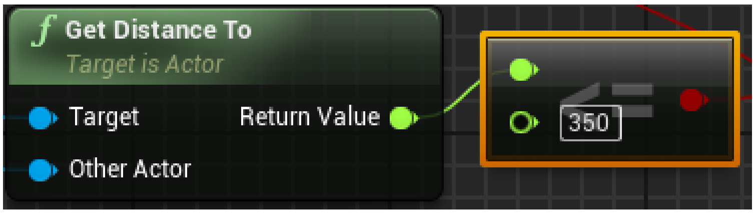
Get Distance To
Target is Actor

Target — Return Value

Other Actor

350

And the final change in the ‘BTTask_BlueprintBase_AttackPlayer’.



Get Distance To
Target is Actor

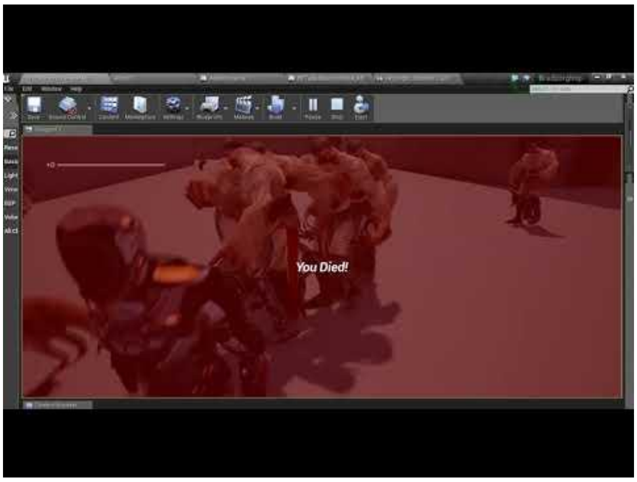
Target — Return Value

Other Actor

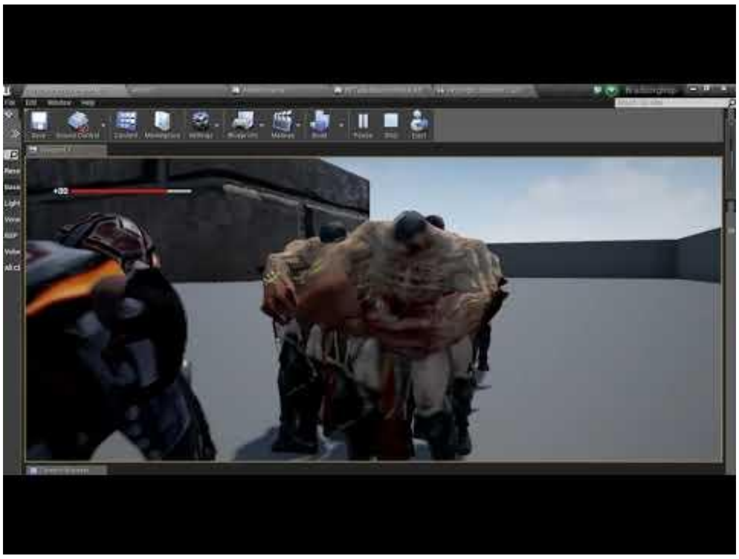
350

This resulted in something accidentally positive for the game overall. As now the player will one hundred percent get damaged from the enemy if they are in the ‘walk’ state. However, if the player is in the default ‘run’ state they can use movement such as jumping and smart herding or as people who play Zombie type games would call it ‘hording’. Player bases that play games such as Call of Duty: zombies’ reference to it as ‘training’. Shown in a reference video named ‘Black Ops 3 Zombies – “The GIANT” SOLO Train Method for High Rounds’ (A, Avxry 2015)

I also left a semi-secret sprint mechanic being, if the player holds ‘CTRL’ the player will speed up just enough to not get hit by the enemy but the change in speed is so slight it’s not noticeable unless it is held for a long period of time.

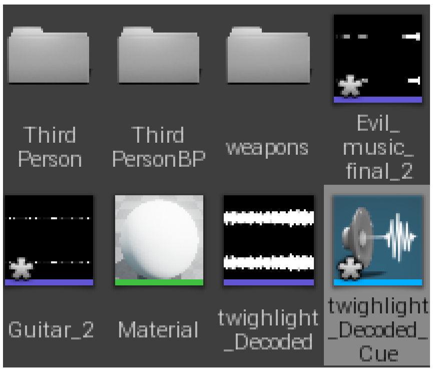


This was now my nearly finished version of ‘Bradzorg!’. This video shows all the mechanics and areas.

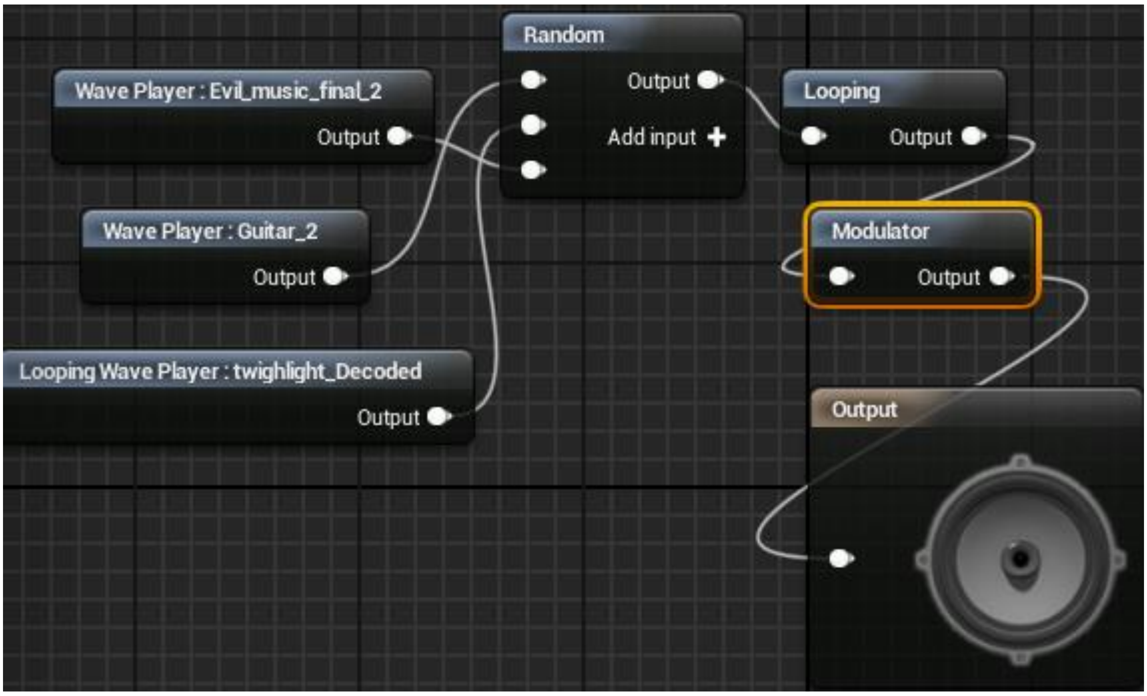


Community aspect:

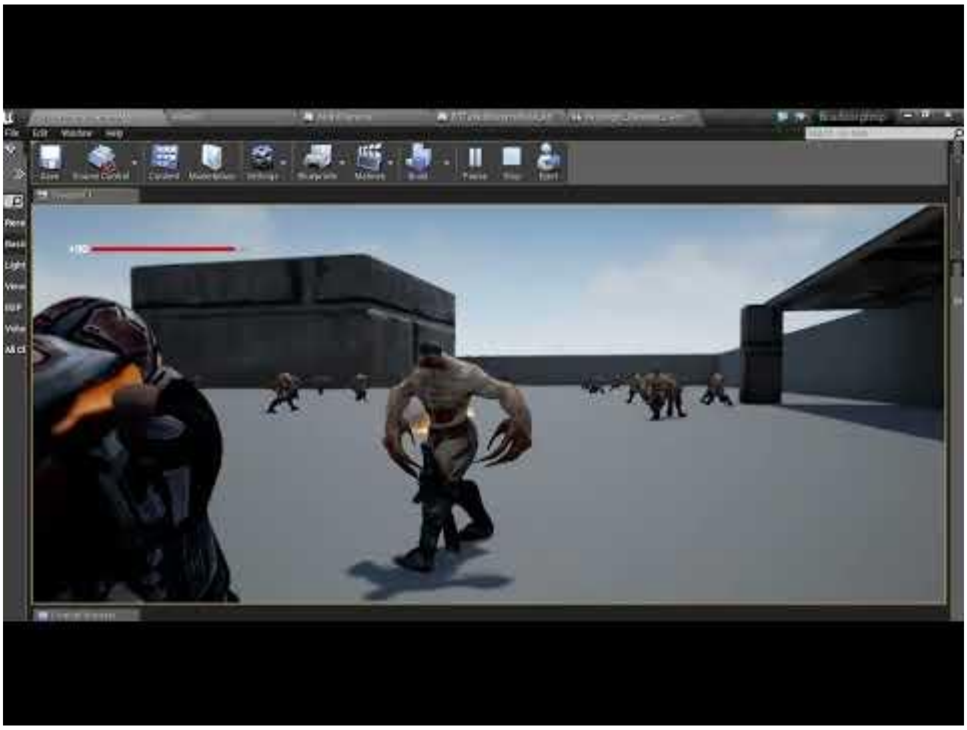
Now with the basic level layout sorted and the deadline being close I needed to add my community aspect. My community aspect is a collaboration with Pipeline Productions. They gave me copyright free songs that I felt best fit my game. To add this as background music it needed to firstly be imported.



I now needed another research source to figure out how to have all the music be able to play randomly and on loop. I used ‘Random Background Music Playlist – Unreal Engine 4’

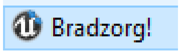


This is the final finished version of bradzorg!



It was now time to export the game as the finished version. To be sure on how to do this I followed a research video, that video being ‘How To Build Package And Export Your Game – Unreal Engine Tutorial’ (M, Aspland 2022)

The first step was to ‘Build’ the game then check that all the project settings were set up as needed. After this was done my game was exported.



Bradzorg!

Bibliography

Dail, K. (2015). Third person shooter build introduction. [online] www.youtube.com. Available at: https://youtu.be/0psqqL68TiY [Accessed 8 mar.2023]

Odyssey, O. (2020). *Creating a Zombie AI in UE4 [4.25] | Getting Started [Part 1]*. [online] www.youtube.com. Available at: https://www.youtube.com/watch?v=TjoE7Pmank0&list=PLZyvGU0G457qe1EJ6KX9NhxDrK8EoeOvo [Accessed 21 Apr. 2023].

Shape, D. (2016). *Unreal Engine 4 - Projectile Shooting - Part 29*. [online] www.youtube.com. Available at: https://www.youtube.com/watch?v=evk4JuFmMmg. [Accessed 26th March 2023].

Aspland, M. (2021). *Shooting And Killing Our Enemy AI | Shooter Minigame Part 5 - Unreal Engine Tutorial*. [online] www.youtube.com. Available at: https://www.youtube.com/watch?v=evGU3geInvg [Accessed 1st May 2023].

Passivestar, P. (2020). *Apartment Building in Blender - 3D Modeling Timelapse*. [online] www.youtube.com. Available at: https://www.youtube.com/watch?v=wq5Xf4s_0ZU&t=349s [Accessed 10th May 2023].

3DGreenhorn, 3DGreenhorn (2021). *Learn Blender 3D in 20 Minutes! Blender Tutorial for Absolute Beginners (2023)*. [online] www.youtube.com. Available at: https://www.youtube.com/watch?v=Rqhtw7dg6Wk&t=227s [Accessed 10th May 2023].

RoBuilder, R. (2021). *How to TEXTURE in Under 60 Seconds.. (Blender)*. [online] www.youtube.com. Available at: https://www.youtube.com/watch?v=1VBRA-OwtPY [Accessed 10th May 2023].

Aspland, M. (2020). *How To Export/Import From Blender To Unreal Engine 4*. [online] www.youtube.com. Available at: https://www.youtube.com/watch?v=5svTBl8dw9o&t=81s [Accessed 10th May 2023].

Avxry, A. (2015). *Black Ops 3 Zombies - 'The GIANT' SOLO Train Method for High Rounds*. [online] www.youtube.com. Available at: https://www.youtube.com/watch?v=g4R8FYtBXBQ [Accessed 13th May 2023].

Interactive, R. (2020). *Random Background Music Playlist - Unreal Engine 4*. [online] www.youtube.com. Available at: https://www.youtube.com/watch?v=T_mfFnBFoWI [Accessed 14th May 2023].

Aspland, M. (2022). *How To Build Package And Export Your Game - Unreal Engine Tutorial*. [online] www.youtube.com. Available at: https://www.youtube.com/watch?v=O2PKihXmejw&t=198s [Accessed 14th May 2023].