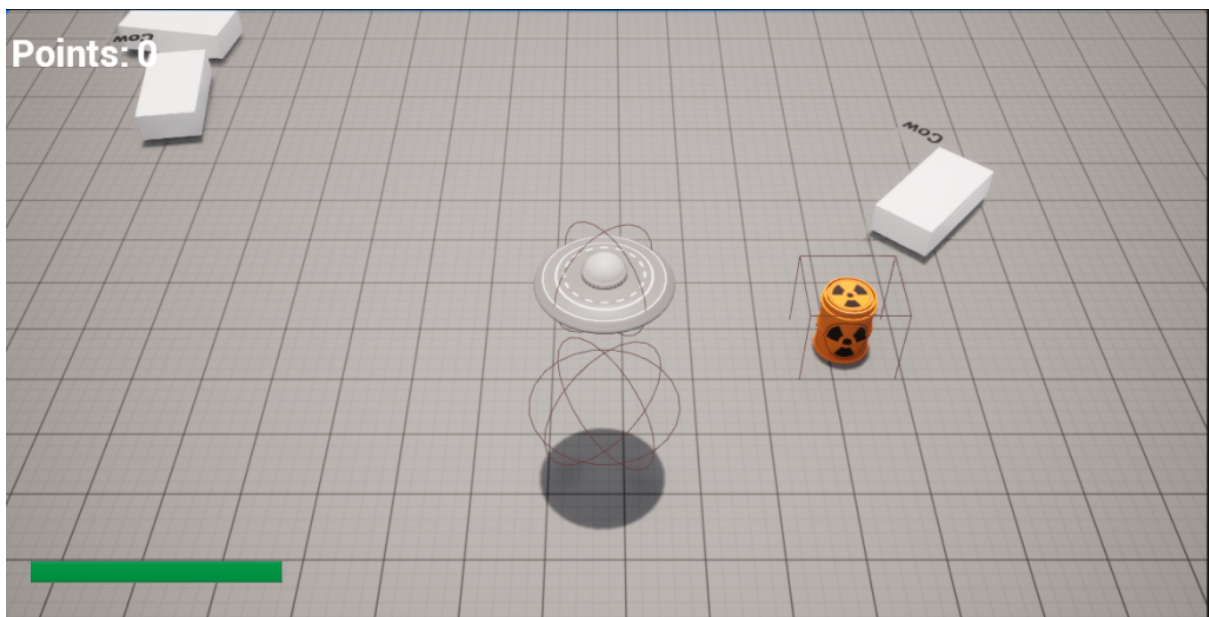


They Came From Mars Dev Log

(Note this is converted from the university forums we used for research and games design documentation)

My commercial game is a UFO game in which the player is playing as the UFO who has to abduct Cow's from an American ranch while the US military tries to destroy you, you will have X amount of time to abduct as many cows as possible to get the highest score you can while trying not to get hit by all the enemies (US military's) weapons there is radioactive barrel pickups you can get to restore health however these have a respawn time so you need to be careful when you use them. During the pre-production of this project, I did research into the audience and market and then into other games in the sci-fi genre.

During this forum I will go into depth the research for game mechanics and design and the audience/market research of the project and the challenges I faced and how I thought it went during the development cycle of this UFO commercial game project.



Research

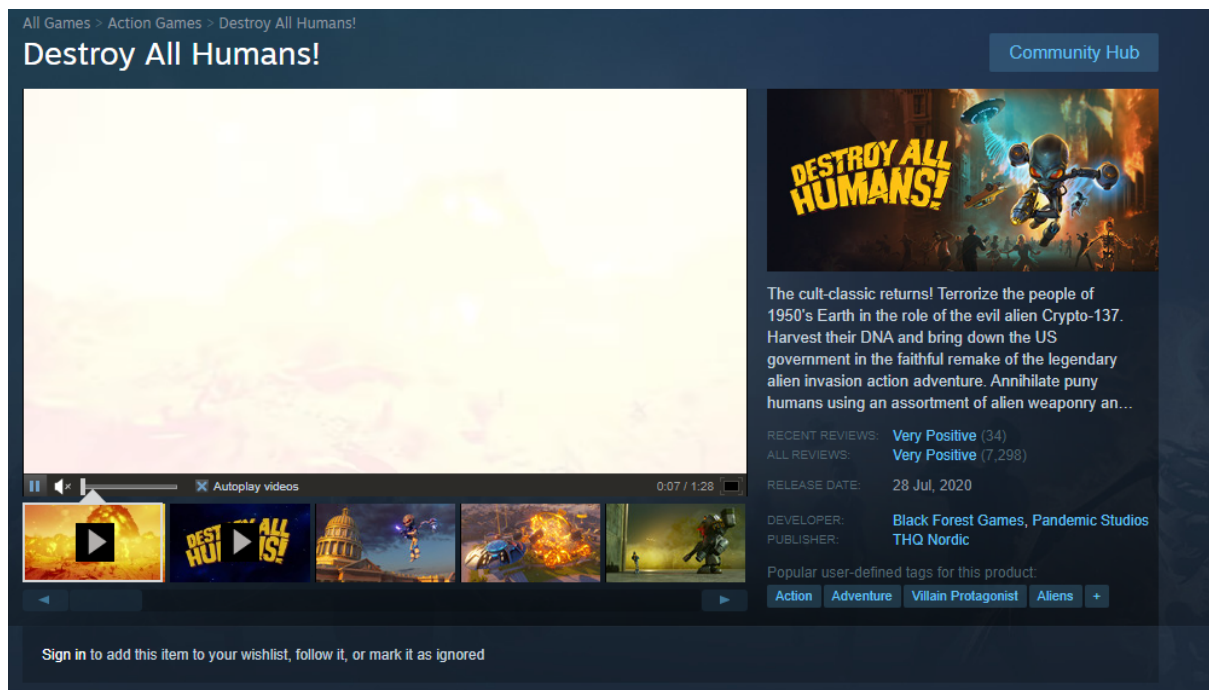
For this part of my forum posts I will be doing extensive research and analysis into the game mechanics I would like to create, the design principles and marketing and audience research including games I researched into that are Sci-fi.

Destroy all Humans

Destroy all humans is an action adventure game where you play as a alien who must terrorise 1950's earth, I looked into Destroy all Humans because for me it was a great example of a 1950's America involving Aliens with a lot of interesting and unique gameplay mechanics and systems involved, my commercial game called 'They came from Mars!' is inspired by this 1950's America theme with alien conspiracy's and comedy, which is why I've heavily investigated and researched into this game to help me design and development my game. Destroy all Humans is what my game could potentially be like if I had the funding, team and scope to create this, however for the vertical slice I decided to go with an arcade style 1950's UFO abducting game.

In Destroy all humans the player has a mechanic which means they can command their flying saucer which uses death rays to kill their opponents, I took this idea and morphed it into something I could use for my own mechanic, for example as it's a vertical slice I've kept it to a small area, and instead of killing your opponents, you use your tractor beam to abduct cows while the US military attack you.

The target audience for this game when researching into it is players who have a love for alien conspiracy theories and 1950's comedy along with SCI-FI genre in general, Which allows the audience to experience playing as an alien causing havoc on earth, this game however is allows you to leave your UFO and cause destruction around you with a wide range of weapon mechanics and systems, I would defiantly like to have included this to my game if i had the scope, budget and team as a sandbox game. This game chose to do a third person perspective however I choose to go with top down perspective as I believed this to be a far interesting perspective for the player when when abducting cows from the ranch.



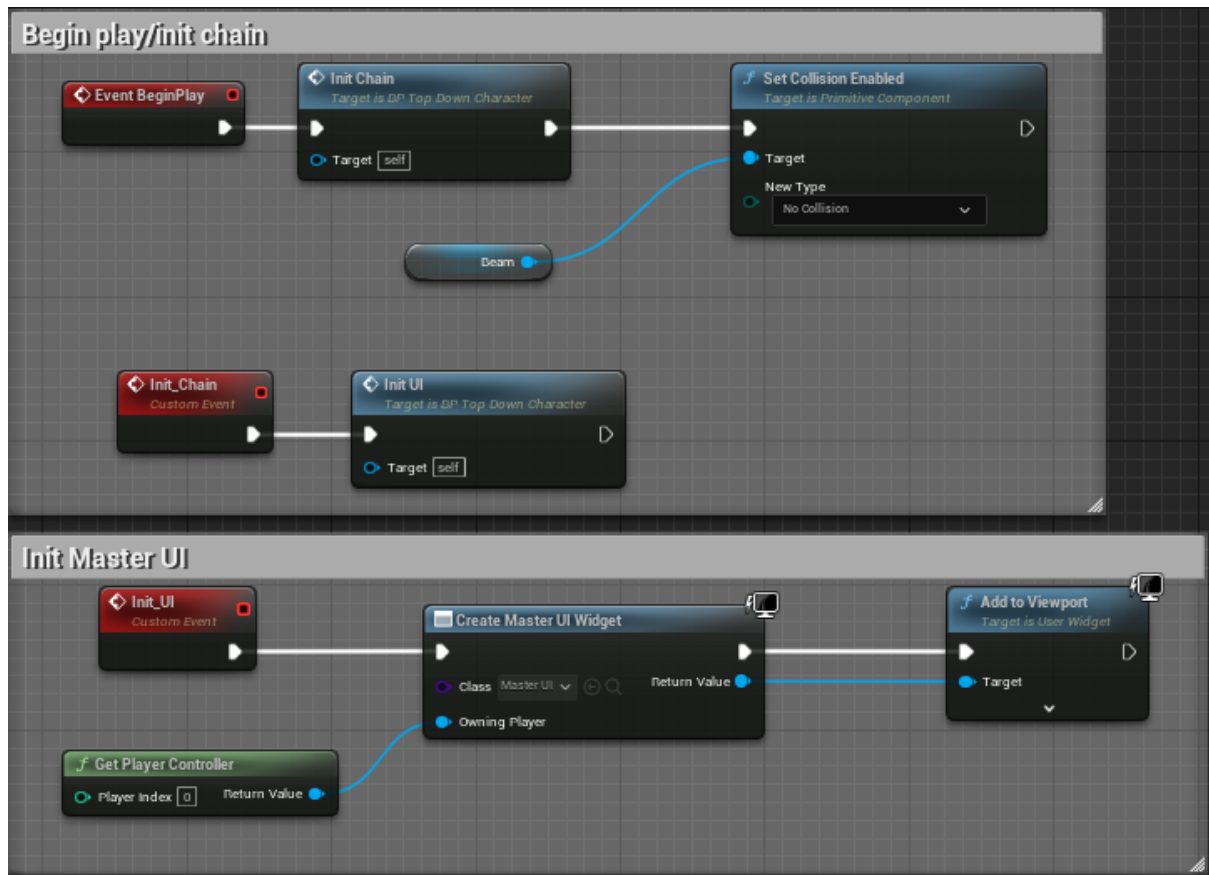
Technical & design development log #1

Player character

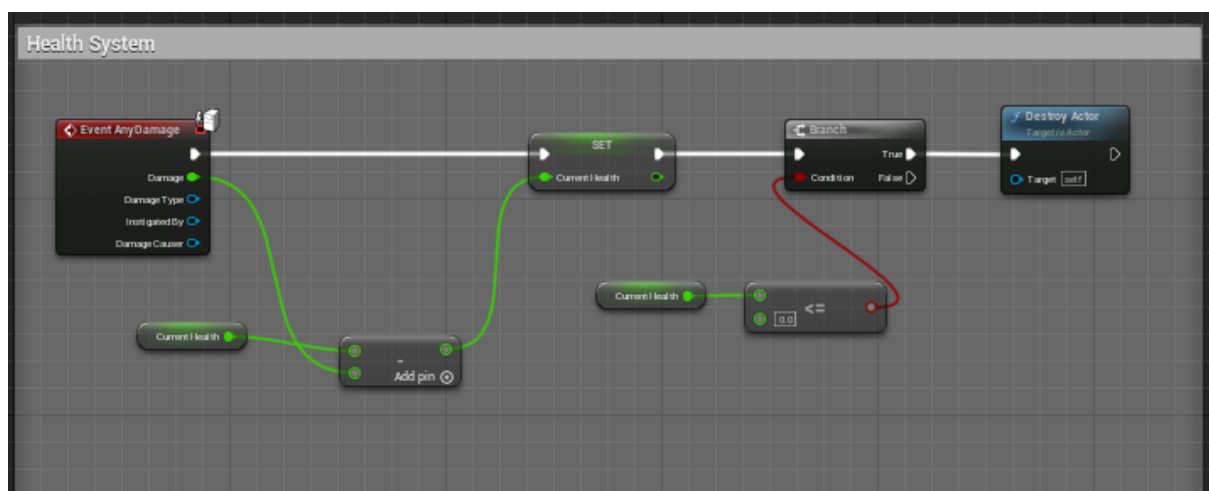
I will be discussing the technical development I have made in my player character below!

This is my begin play & init chain blueprint, where on begin play it will call the custom event init chain then set collusion enabled of beam to no collusion this was to avoid beam working when you start up the game before using the beam. then on the init chain we call the custom

event init UI and on custom event init ui we create master UI widget and add it to the viewport

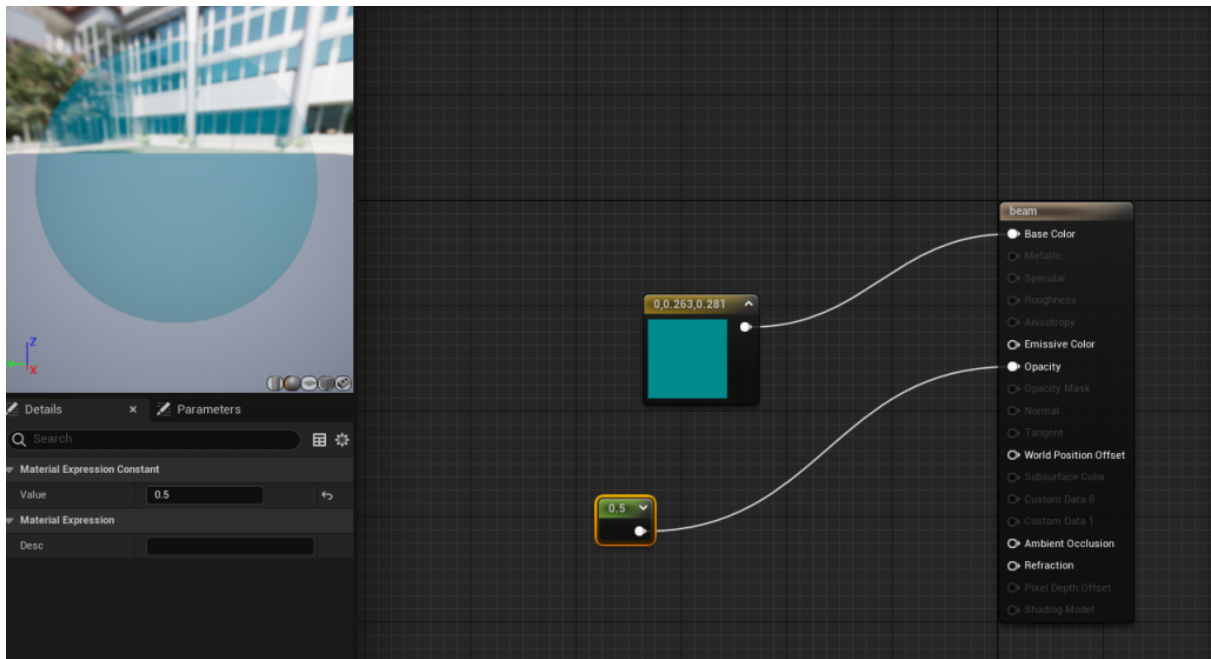


This is my health system for my UFO which is set too 100 and apou reaching 0 health you will die.

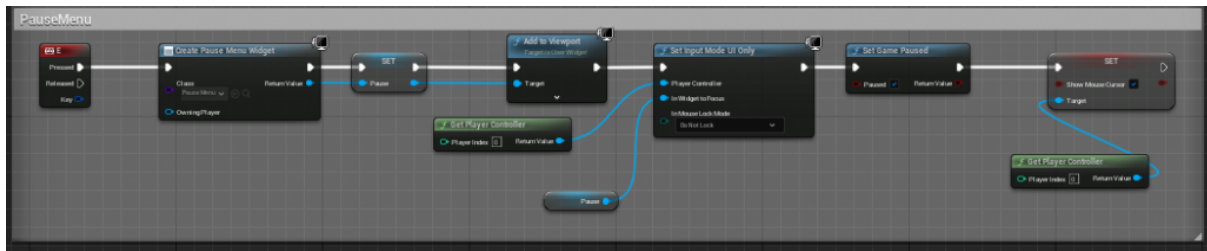


This is my old game timer which I used for my game which was set in the character however I remove this as it wasn't efficient and didn't work as intended so I have reiterated on this and it is now hosted within the

[illegible]

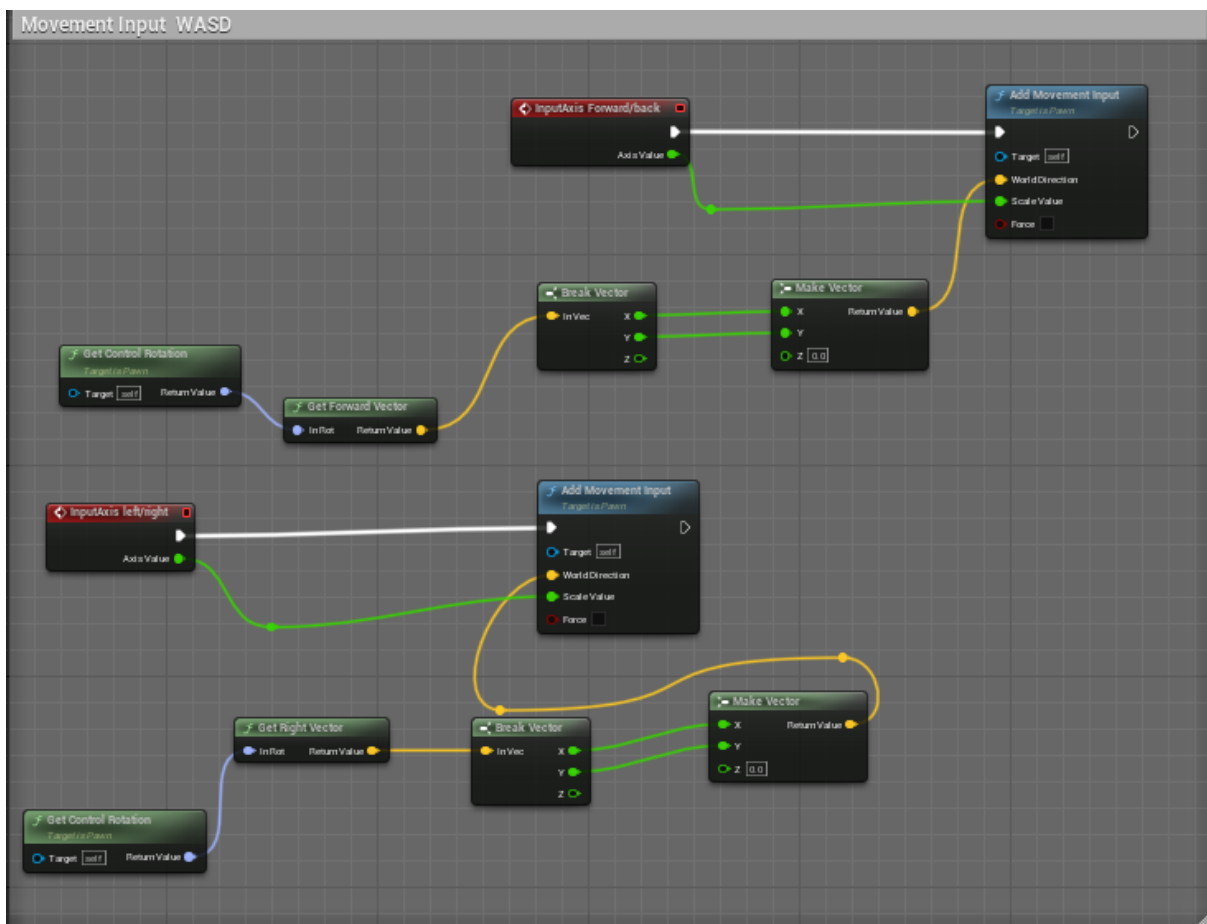


This is my pause menu blueprint which when you press action input e it creates a pause menu widget then sets the reference and adds to viewport and sets the input mode to UI only so you cannot interfere with the game after this we set the game to paused and set show mouse cursor.

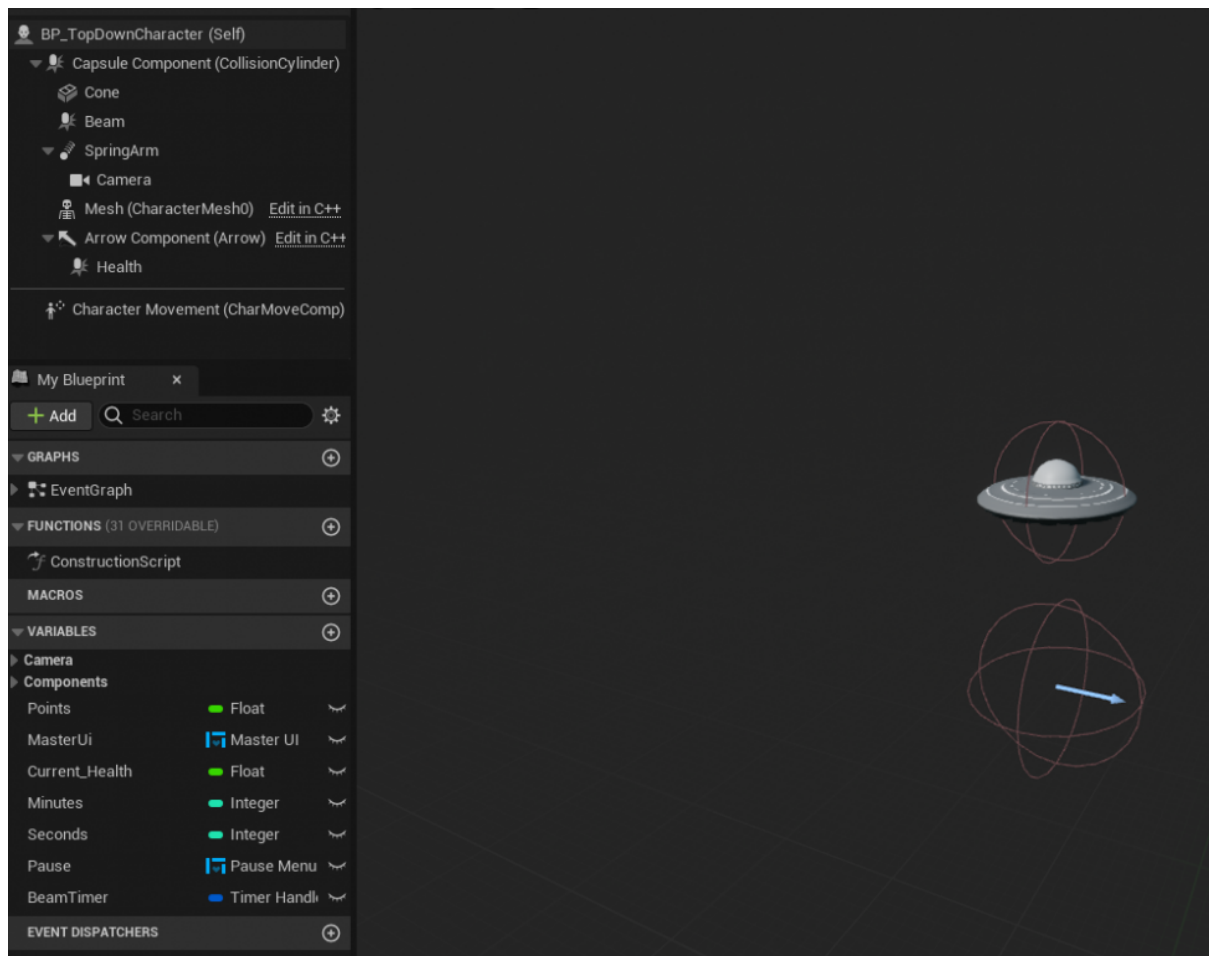


As I used the top down project template I didn't have WASD movement keys and was touch to move, however I engineered my character so that it would use WASD to move the character around, on world direction it would get make vector then the Y and X would go into a break vector and the in vec would get forward vector and get control rotation.

The input axis left and right (A and D) would set movement input and make vector on the world direction and break it again and this time get the right vector and get control rotation.



This is the design view of my UFO player character, I had to use multiple sphere collisions so that I could get the player health and hit overlap separate from the tractor beam.

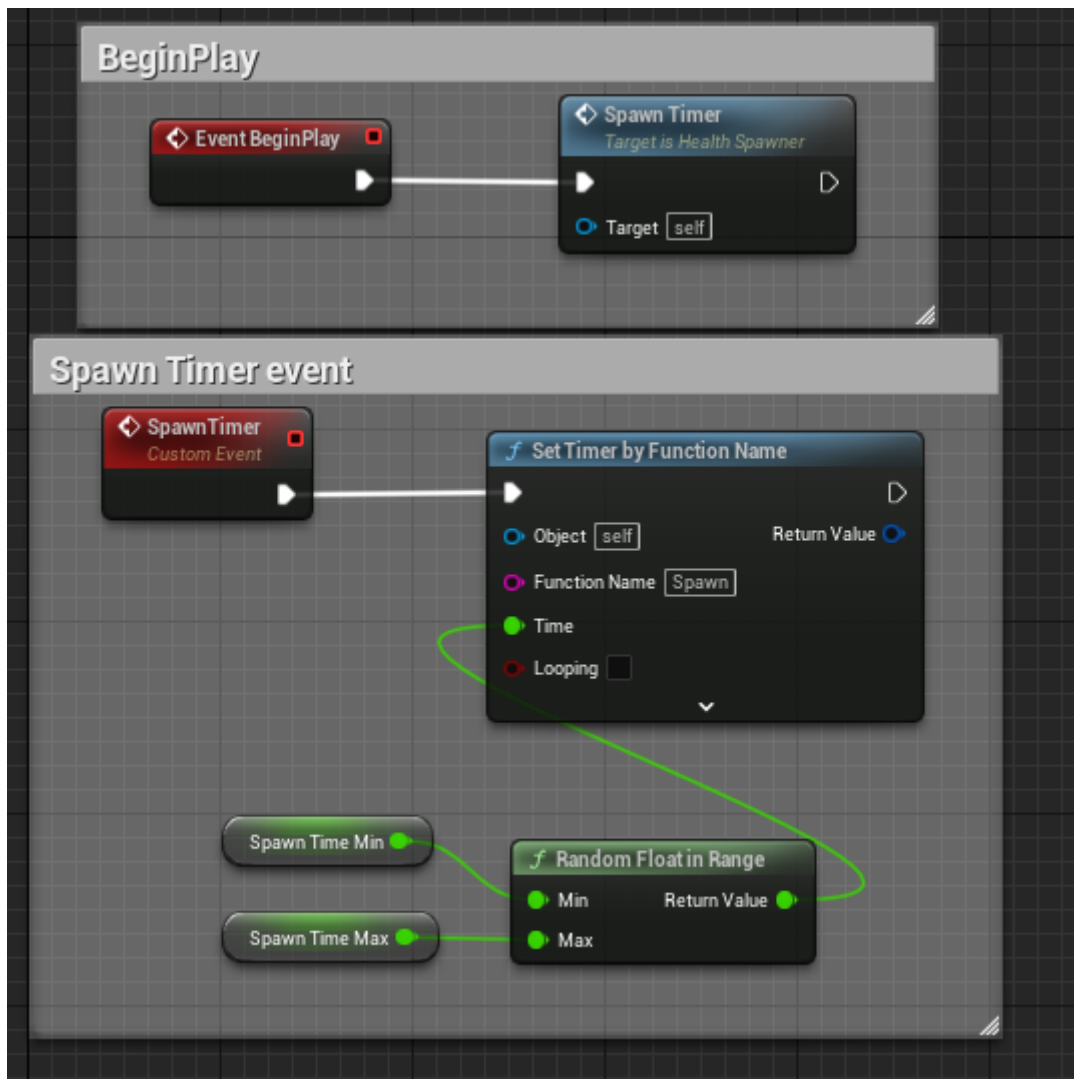


Technical & design development log #2

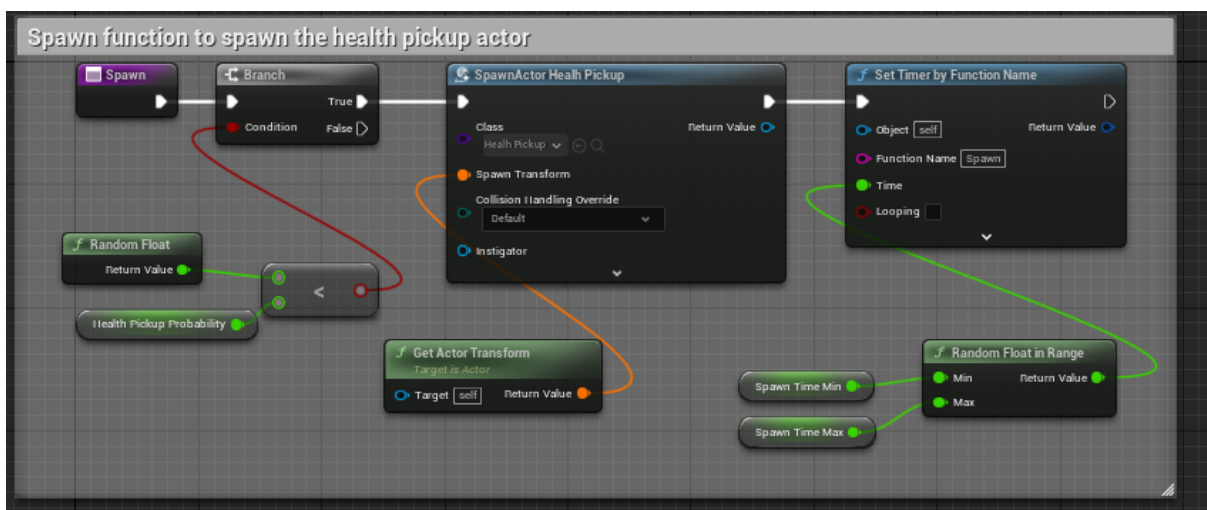
Spawners

I will be discussing the technical development I have made in my spawners below!

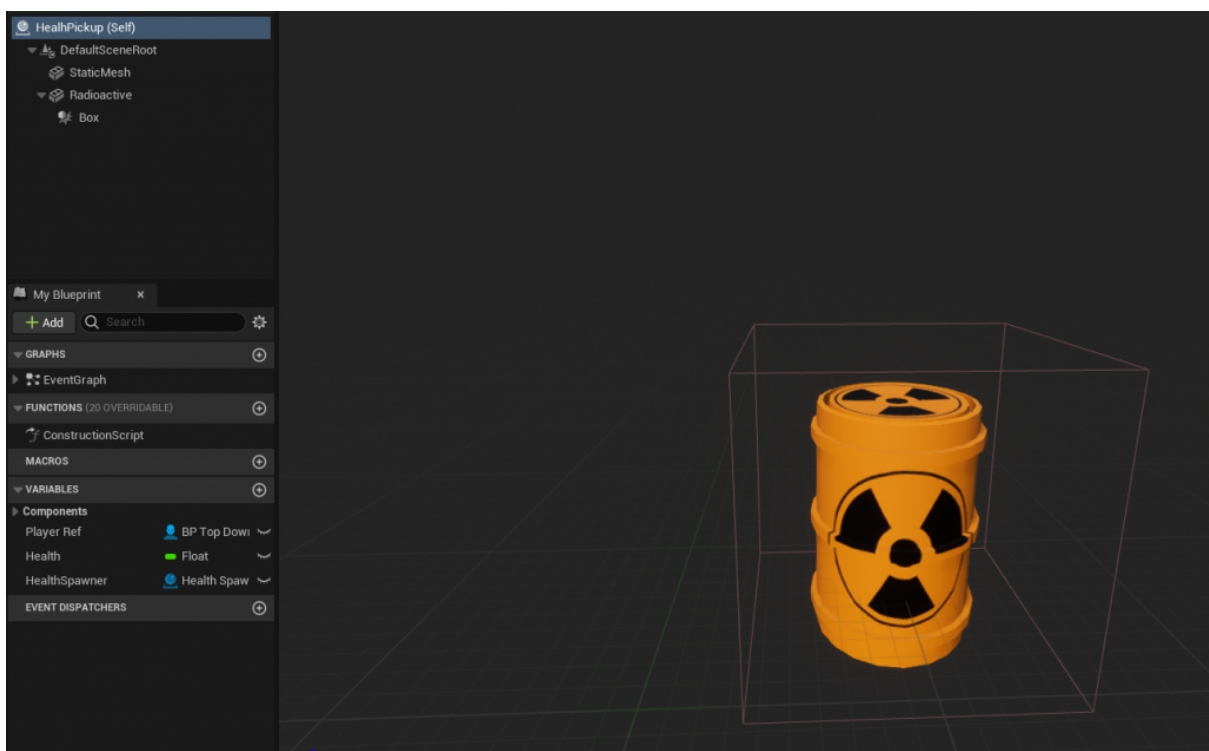
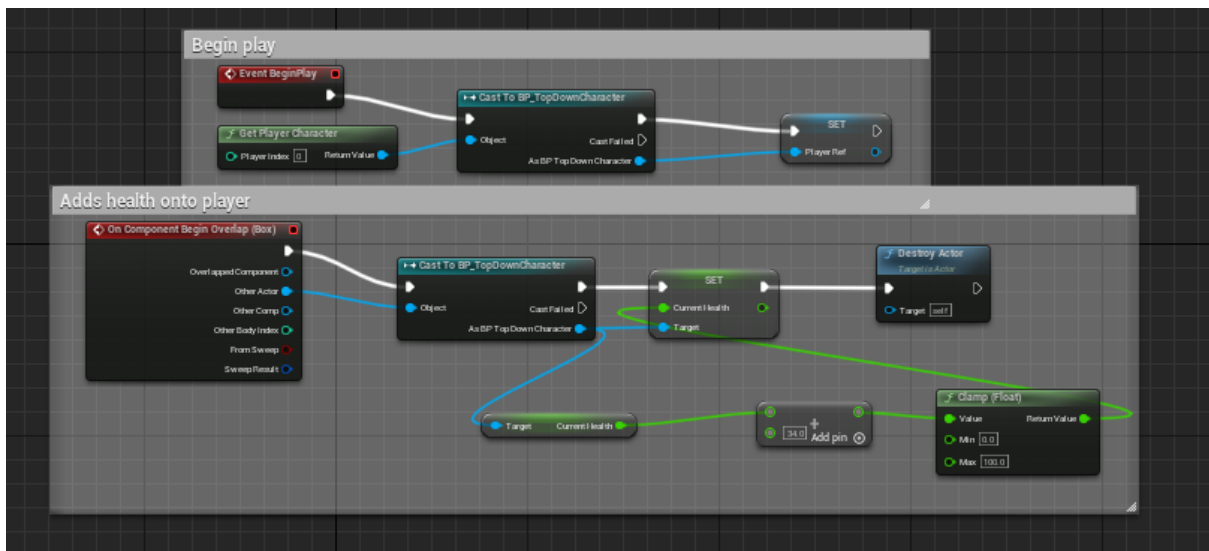
This is my Health spawner actor eventgraph blueprint on begin play we call the event spawn timer, which on spawn timer event we set timer by function name, with the function name being Spawn which is what our function is called, then from time we get random float in range and then two float variables called spawn time min and spawn time max, which get the minimum and maximum time it takes for our health spawner to spawn



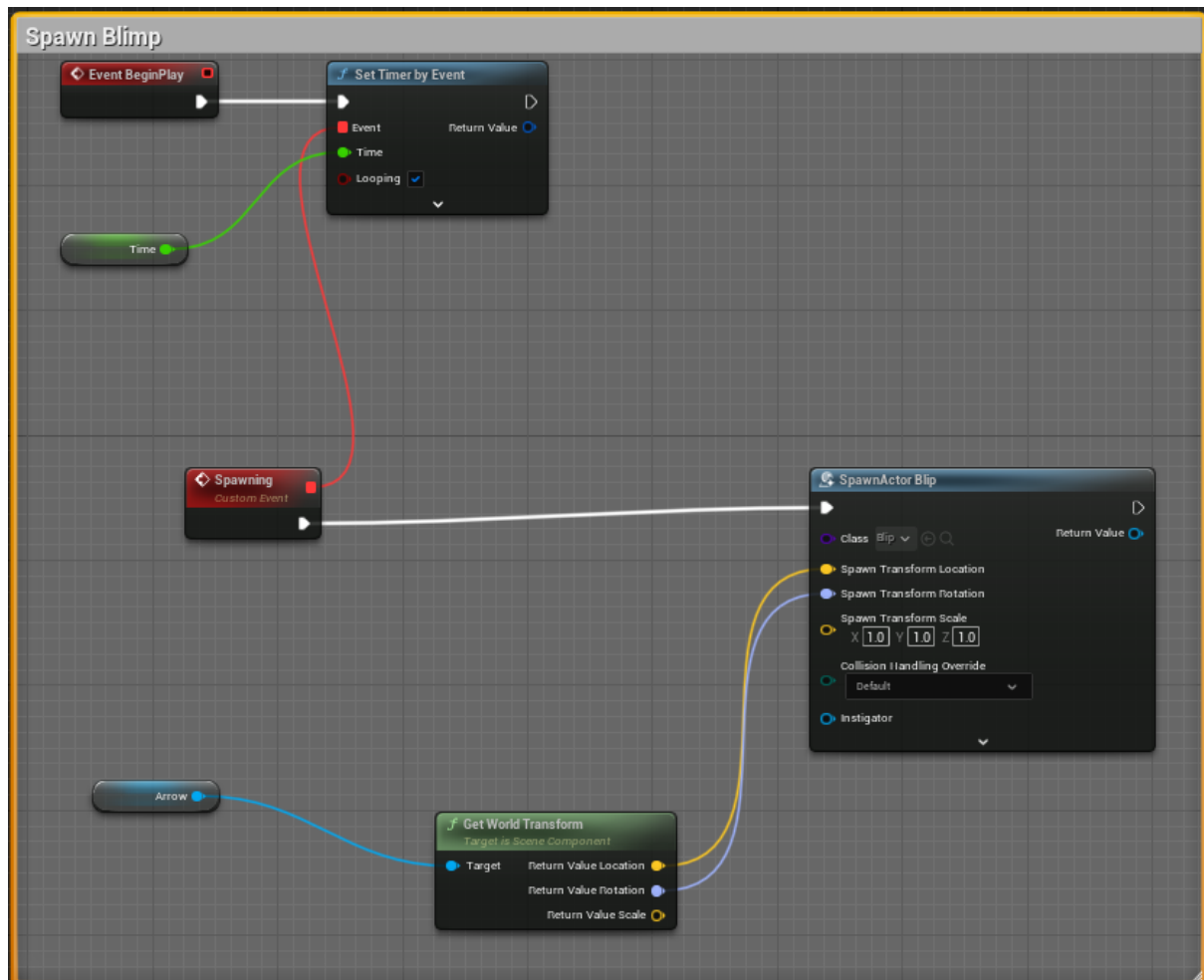
This is our function we call in our timer by function, we branch off from the spawn function event, and on the condition of our branch we have if less then, which connects into random float and then our health pickup probability which is set to 100, from the true branch we spawn actor health pickup where we get the spawn transform of get actor transform so we spawn it at the actors location. After this we go into set time by function name with spawn then set the spawn time min and max in the random float range.



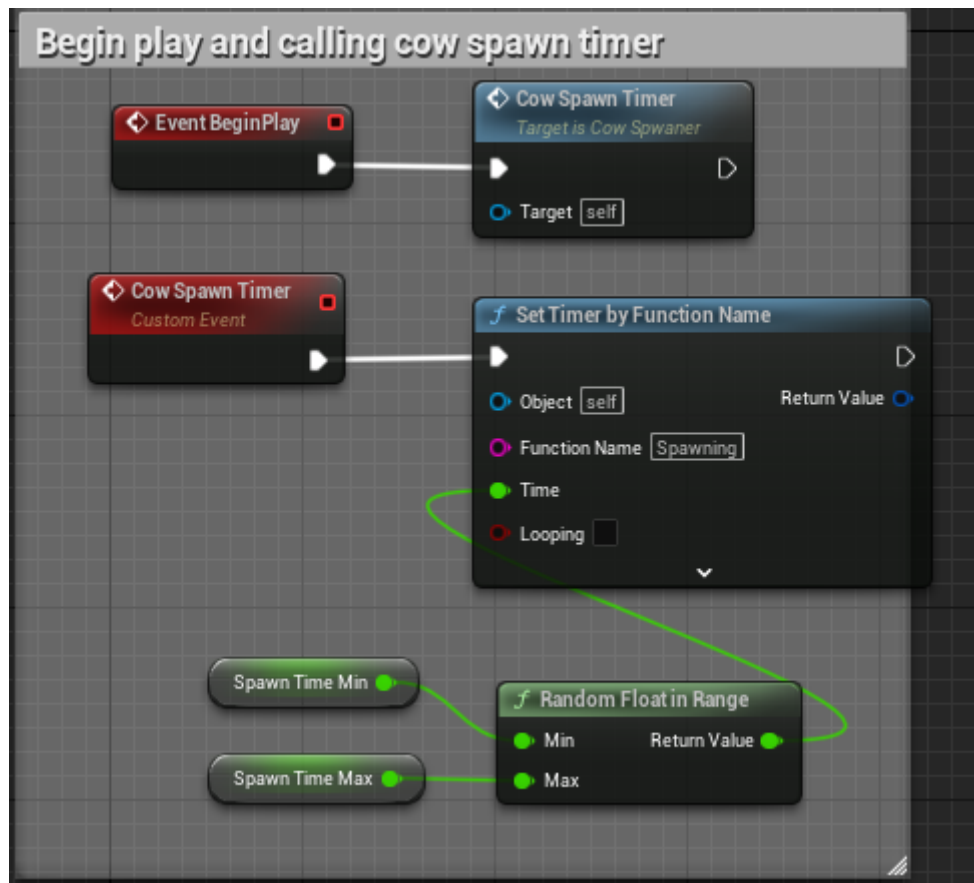
This is where I hold the health pickup actor, on begin play we cast to the top down character and get and set the player references so we can easily use the references there, then on component begin overlap (on my box collision) we cast to top down character and set the current health and clamp it into the return value float, from the current health float variable, and after the execute we destroy the actor so it can then later respawn. from the top down character reference in the cast we get current health then use an add node, I've balanced the add node to 34 so we can get three lives before we die. then this plugs into the clamp value



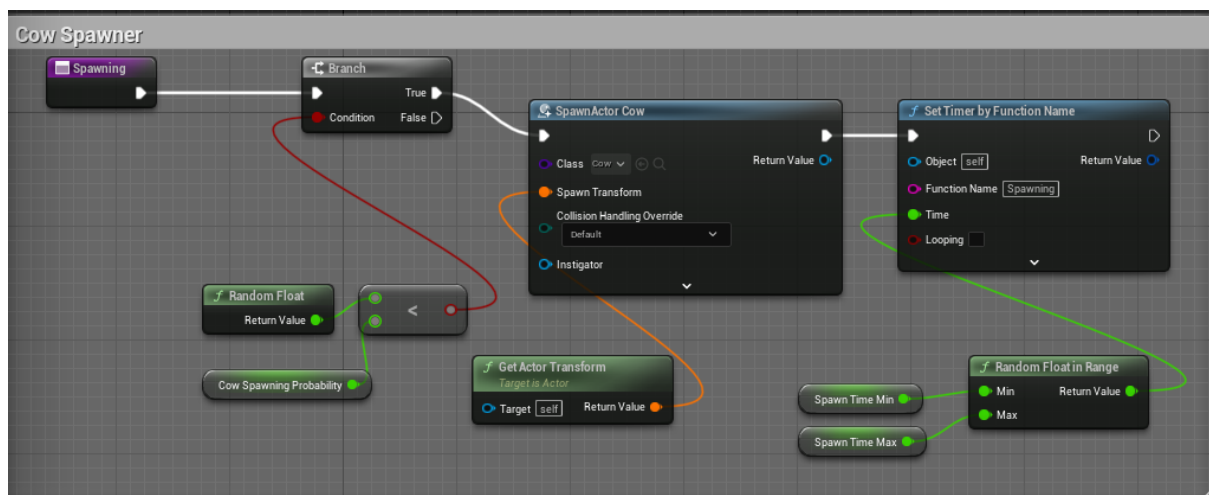
This is my blueprint for spawning the enemy blimps out of the spawner, on begin play we set timer by event and have a custom float time variable which is set to 6 seconds and we have it set to looping so it will always spawn after 6 seconds, then on event we create a custom event called spawning and spawn actor blip where the spawn transform location and rotation will be plugged into the get world transform node and then the target is my arrow which is hidden in game but it will come out of the arrows location this is good for placing where you want actors to spawn.



This is my Cow spawner actor eventgraph blueprint on begin play we call the event spawn timer, which on spawn timer event we set timer by function name, with the function name being Spawn which is what our function is called, then from time we get random float in range and then two float variables called spawn time min and spawn time max, which get the minimum and maximum time it takes for our health spawner to spawn



This is our function we call in our timer by function, we branch off from the spawn function event, and on the condition of our branch we have if less then, which connects into random float and then our cow spawning probability which is set to 100, from the true branch we spawn the cow actor where we get the spawn transform of get actor transform so we spawn it at the actors location. After this we go into set time by function name with spawn then set the spawn time min and max in the random float range.

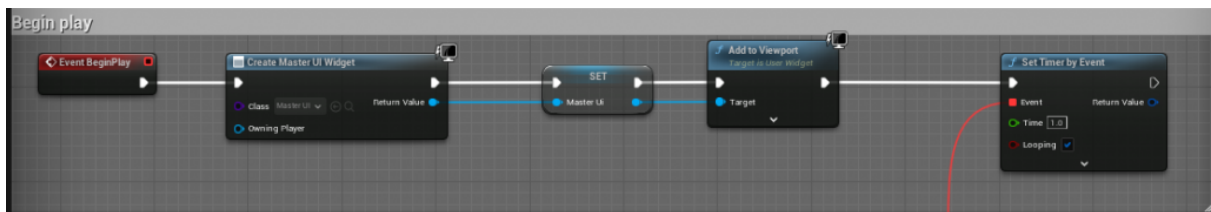


Technical & design development log #3

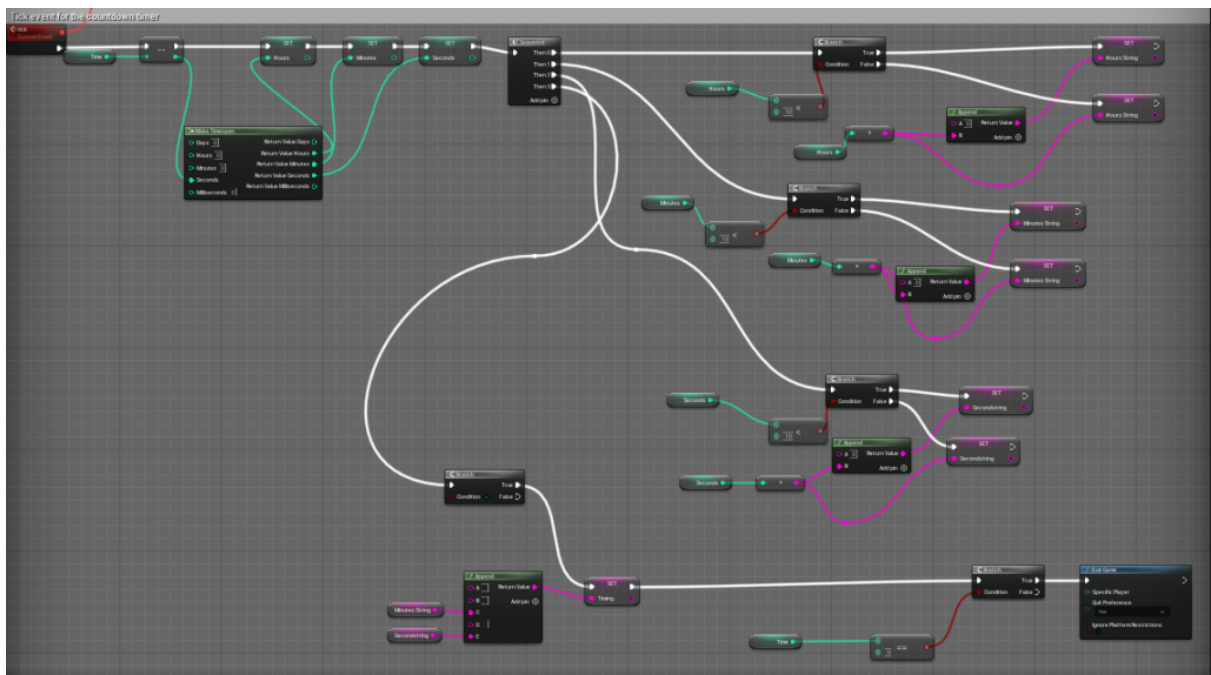
Timer

I will be discussing the technical development I have made in my timer below!

This is my begin play event in my gamemode, where on begin play it creates the master UI widget and sets the reference variable then adds to view port from here we set timer by event which loops, and goes to a custom event called tick.

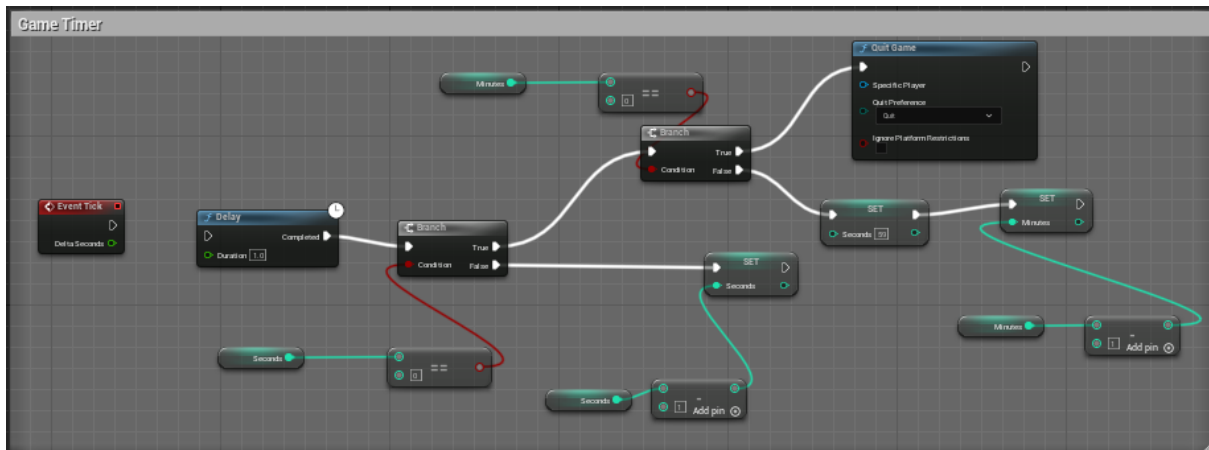


This is the new timer blueprint from tick we subtract 1 and get the time integer variable and then make timespan for seconds, then from the execute of the subtract one we set the hours, minutes and seconds and plug those into the retrospective variable integer handles in make timespan, from here we go into a sequence node where for example we branch and set the condition to if less then then 10 and then for example the hour integer variable then from true we set the hours string and append it into the hours variable this is repeated for minutes and seconds too. Until we get to then 3 where we branch and if true we set the timing string and append it for minutes and seconds, as were not using hours we unplug this from the pin in the append, then from the set timing we go into another branch and do equals to on the condition with the time integer and then on true we quit the game however I will change this to change level to main menu.



02:10

This is the old game timer blueprint as discussed in the player character development log

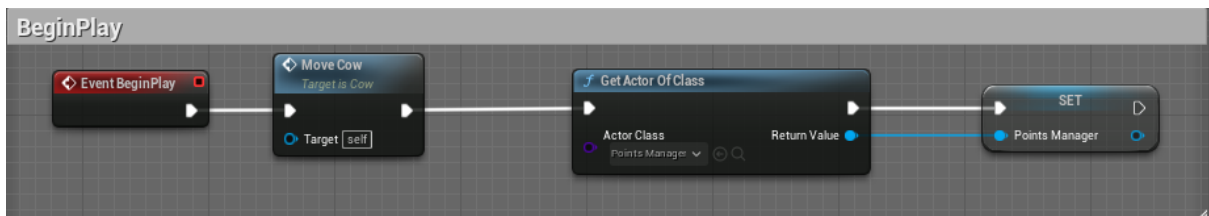


Technical & design development log #4

Cow and points manager

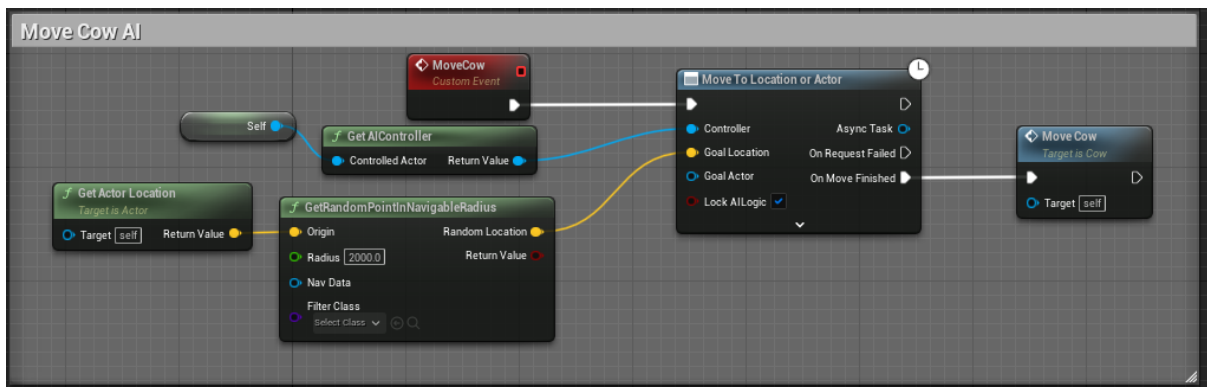
I will be discussing the technical development I have made in my Cow actor and points manager actor below!

This is my event begin play in my cow actor blueprint event graph after being play we call the custom event called Move cow which is where my blueprint is held for the cow being able to move around the map, after this we get actor of class points manager and set it, so the points are referenced from the cow.

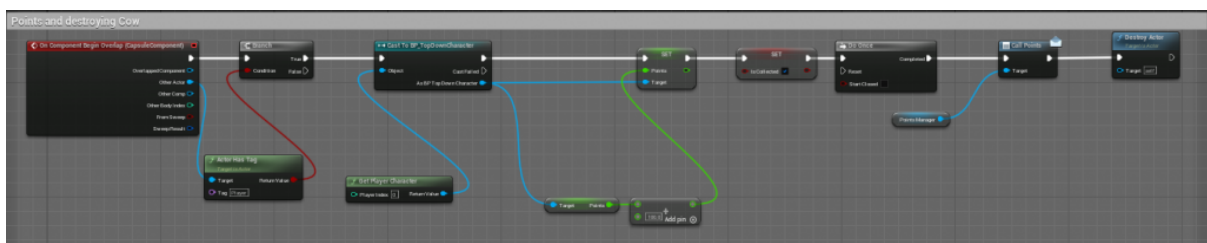


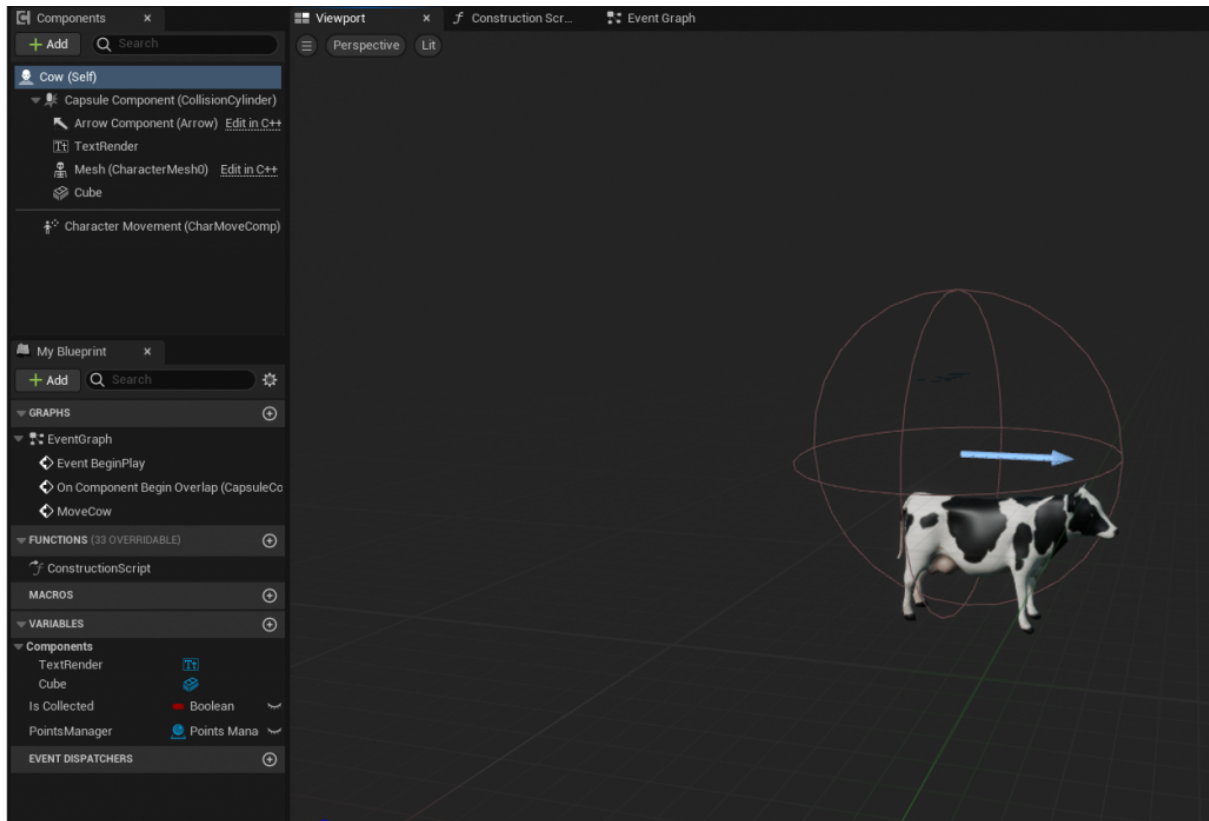
As mentioned above, this is the move cow custom event which controls moving the cow throughout the map, after move cow custom event we move to location actor and the controller is get AI controller and

target is self, the goal location is get random point in navigable radius of the origin get actor location and a radius of 2000 this will allow the cow to move randomly around the map within this radius, and then it calls move cow again at the end.

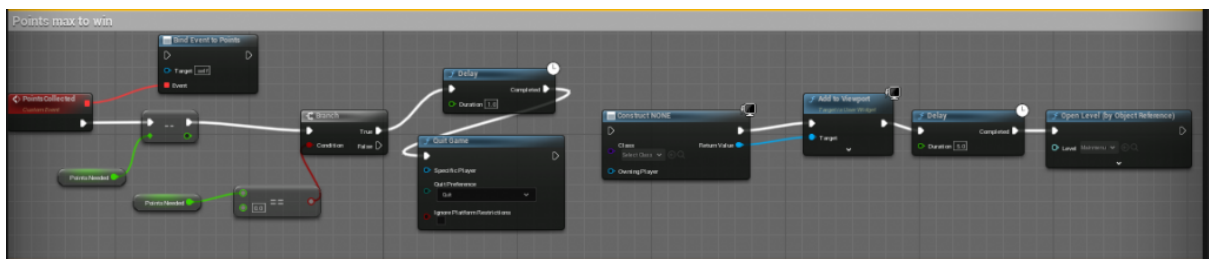
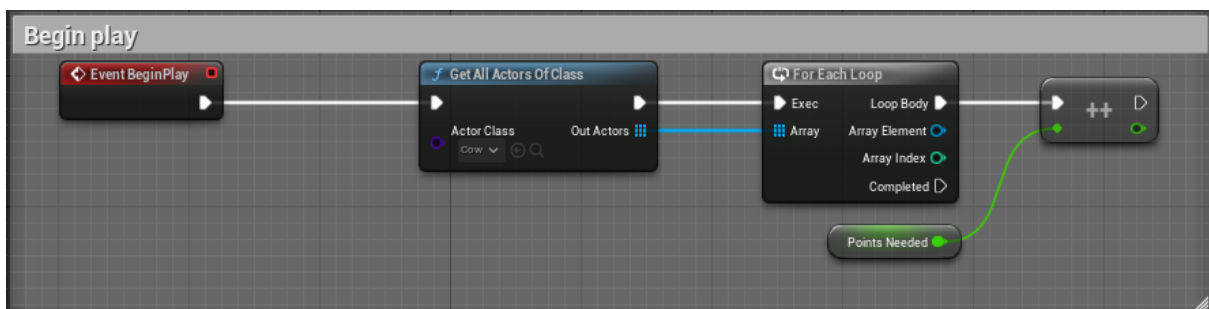


This is the blueprint for destroying the cows when abducted and giving the player points, we use a capsule component on component begin overlap here with the other actor being actor has tag (which is Player) and that goes into our condition branch from there we cast to our top down character with the object being the get player character, after casting to the top down character we set the points with that float set having the points pin go into an add node which adds 100 points then target being top down character, This gives the player 100 points for abducting this cow. after the set points float, we set is collected as true for are boolean, then do once so it only happens once on each cow, then on completed call points with target being point manager and then destroy actor.





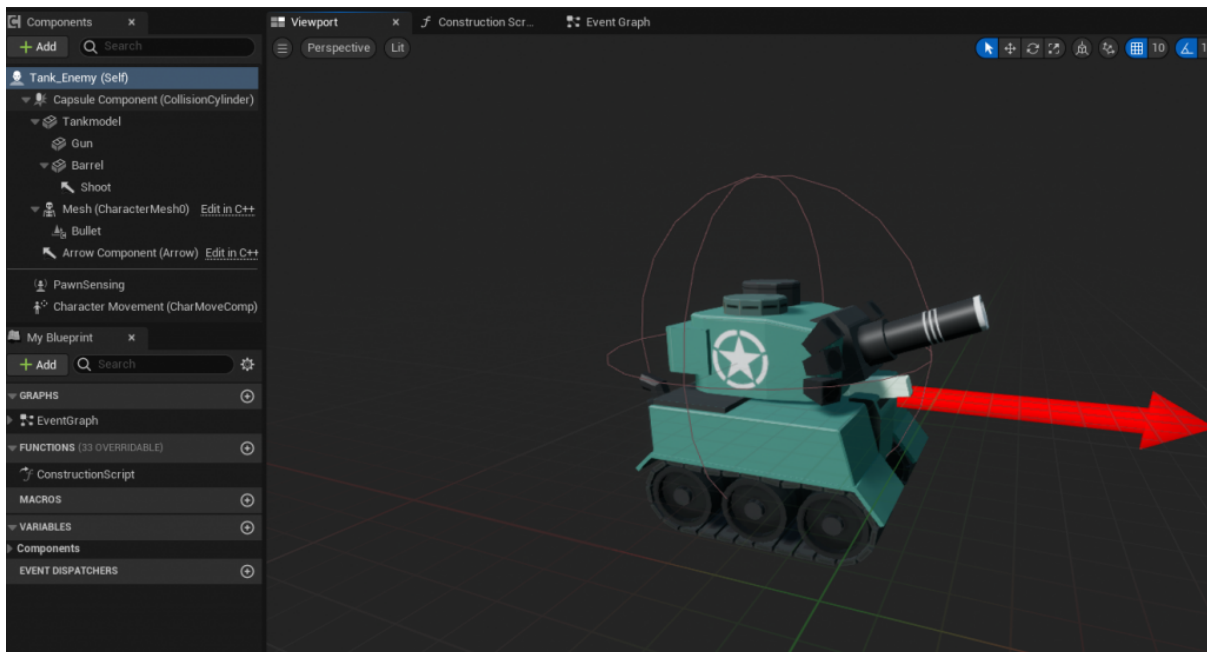
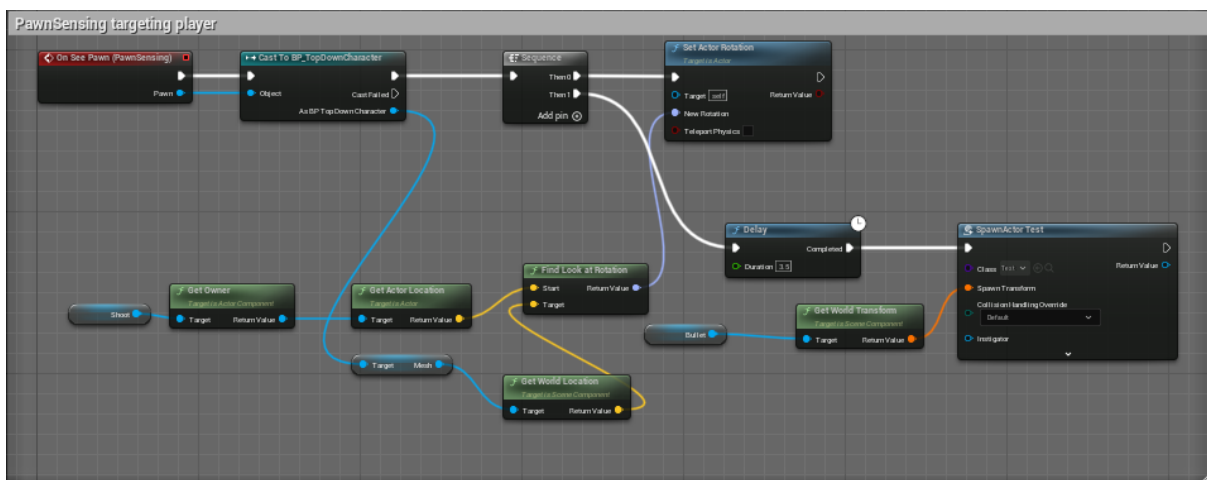
My game is now timer based as discussed in a previous post so you must get as many points as possible in the time limit or before you die, this below is my old iteration of my points system which let you win after reaching a max amount of points needed.



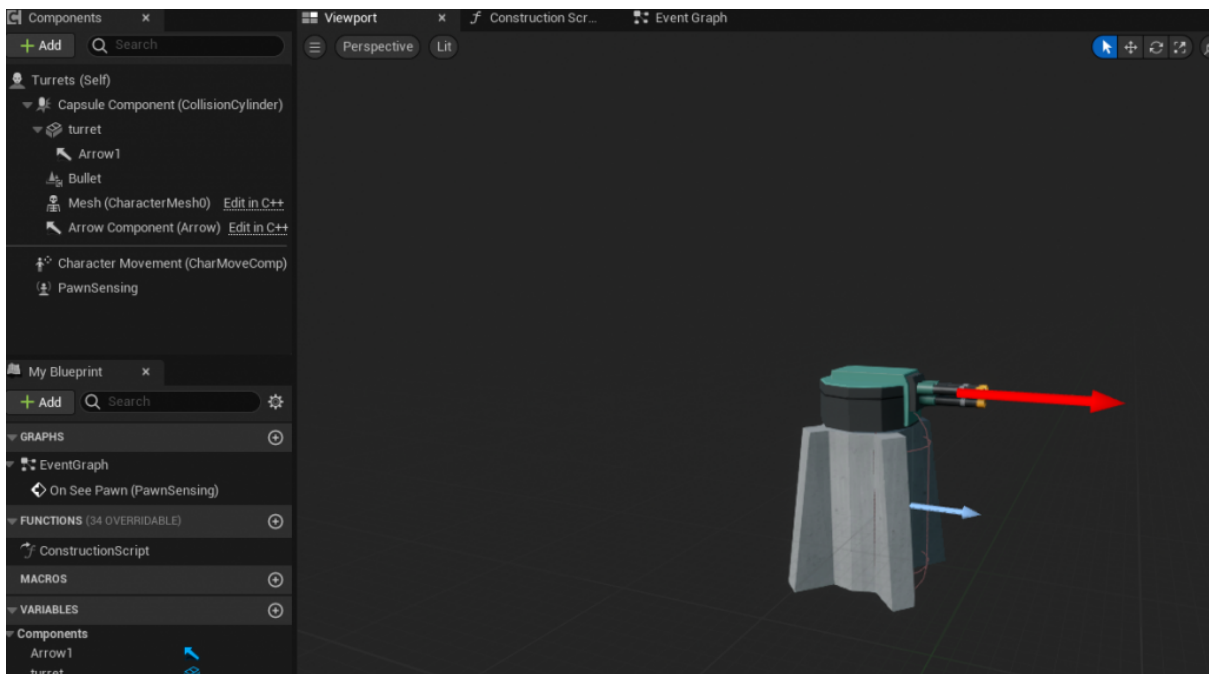
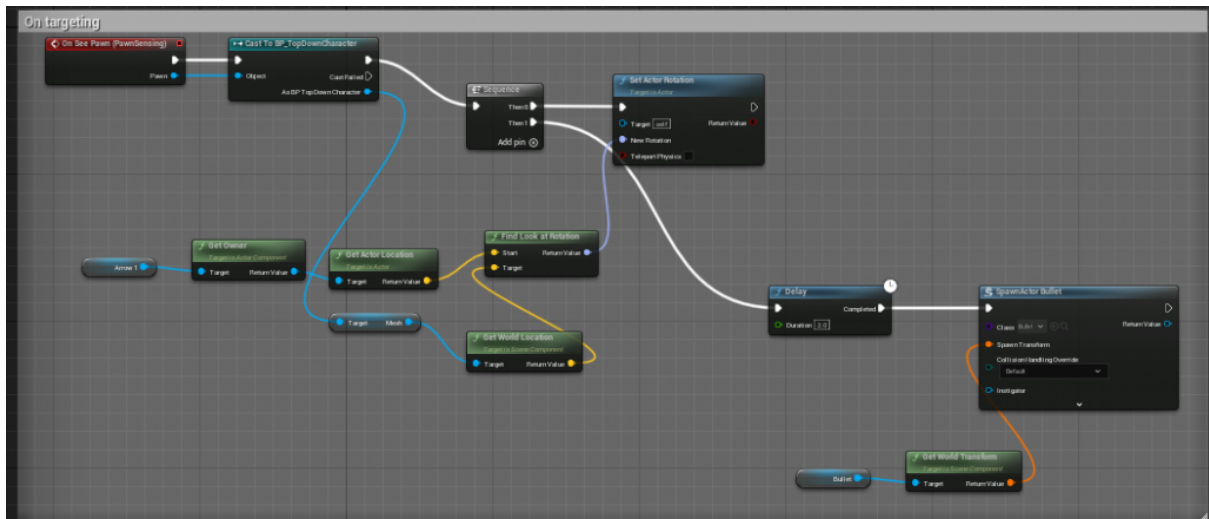
Technical & design development log #5

Enemies

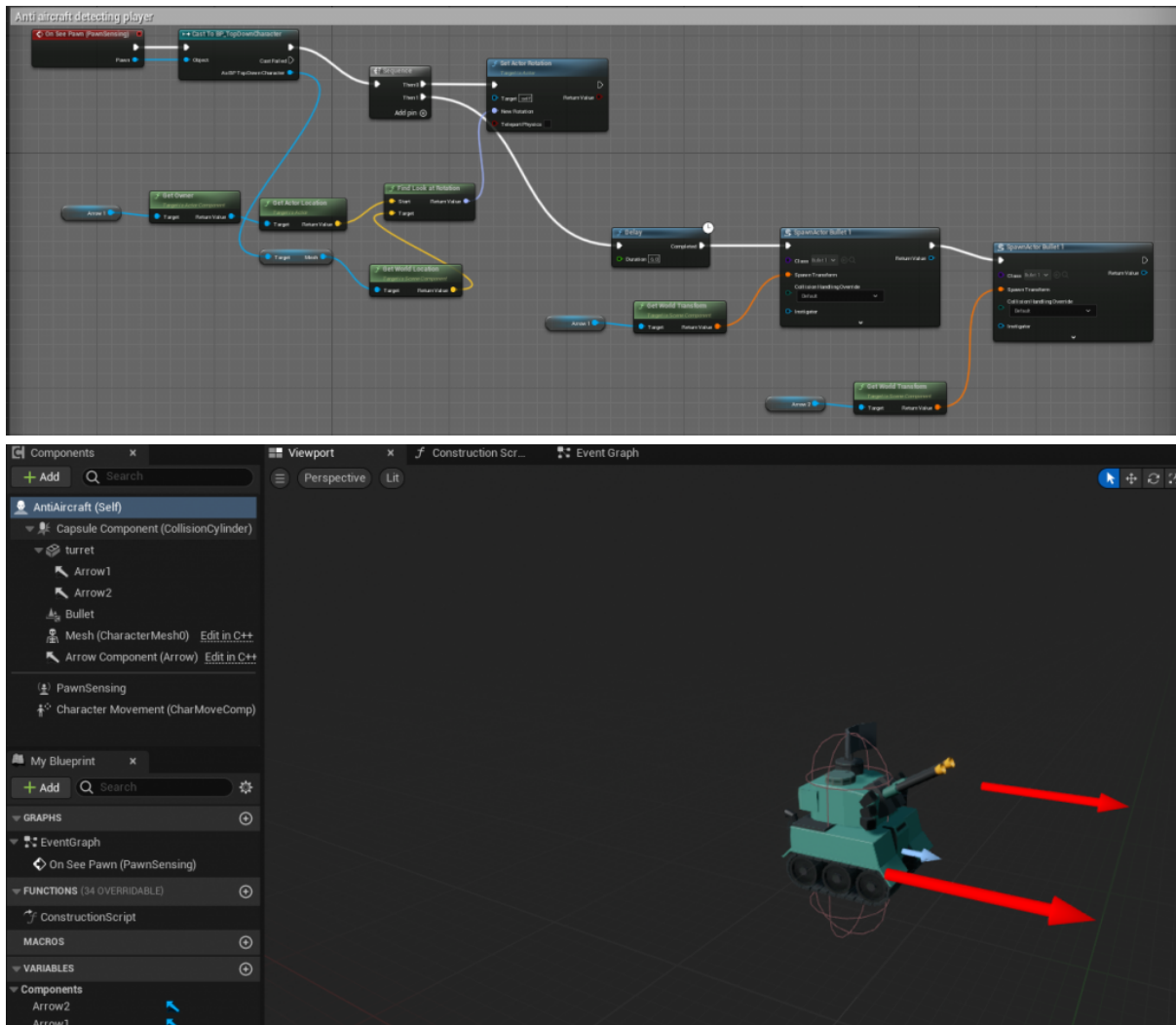
This is my tank enemy pawn sensing blueprint which is where it will target the player from on see pawn we cast to top down character and go into a sequence with then 0 being set actor rotation and on new rotation we find look at rotation which target is world location and target mesh of top down character and the start is get actor location and then target get owner which is shoot. on the then 1 of the sequence we have a delay of 3.5 seconds then it spawns actor test (will rename to bullet_shell) and gets world transform and bullet reference



This is my turret actor blueprint which works exactly the same way my tank on see pawn event works.

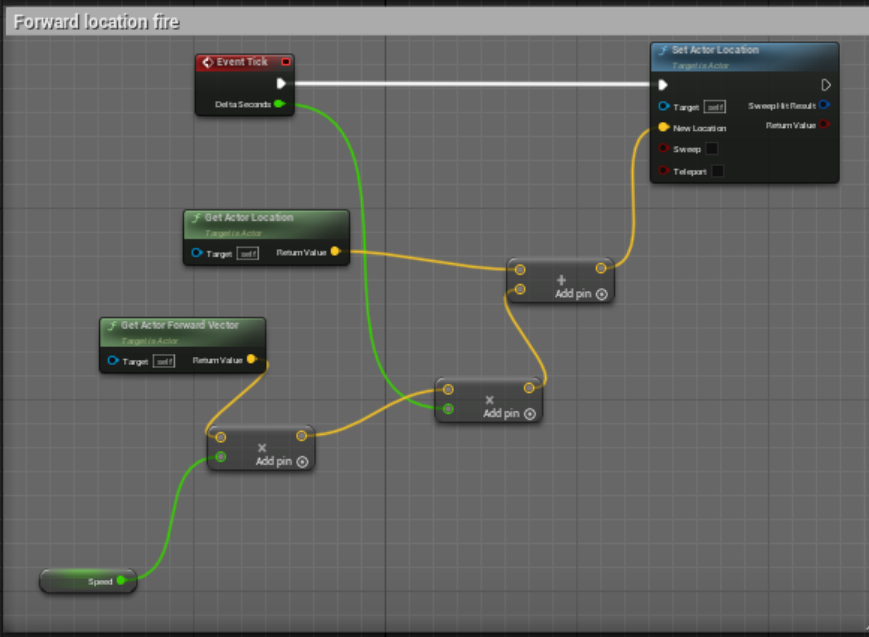
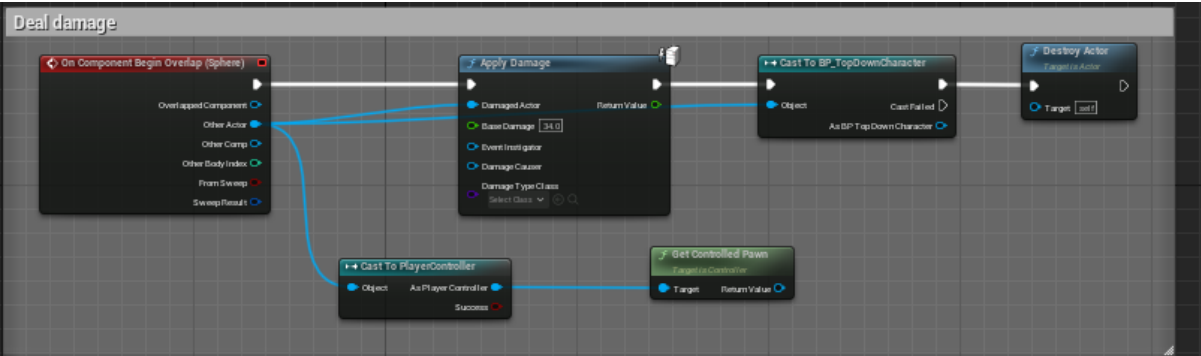


This is my anti aircraft turret actor blueprint which works exactly the same way my tank & turret on see pawn event works, however in this instance we spawn two bullets instead of one.

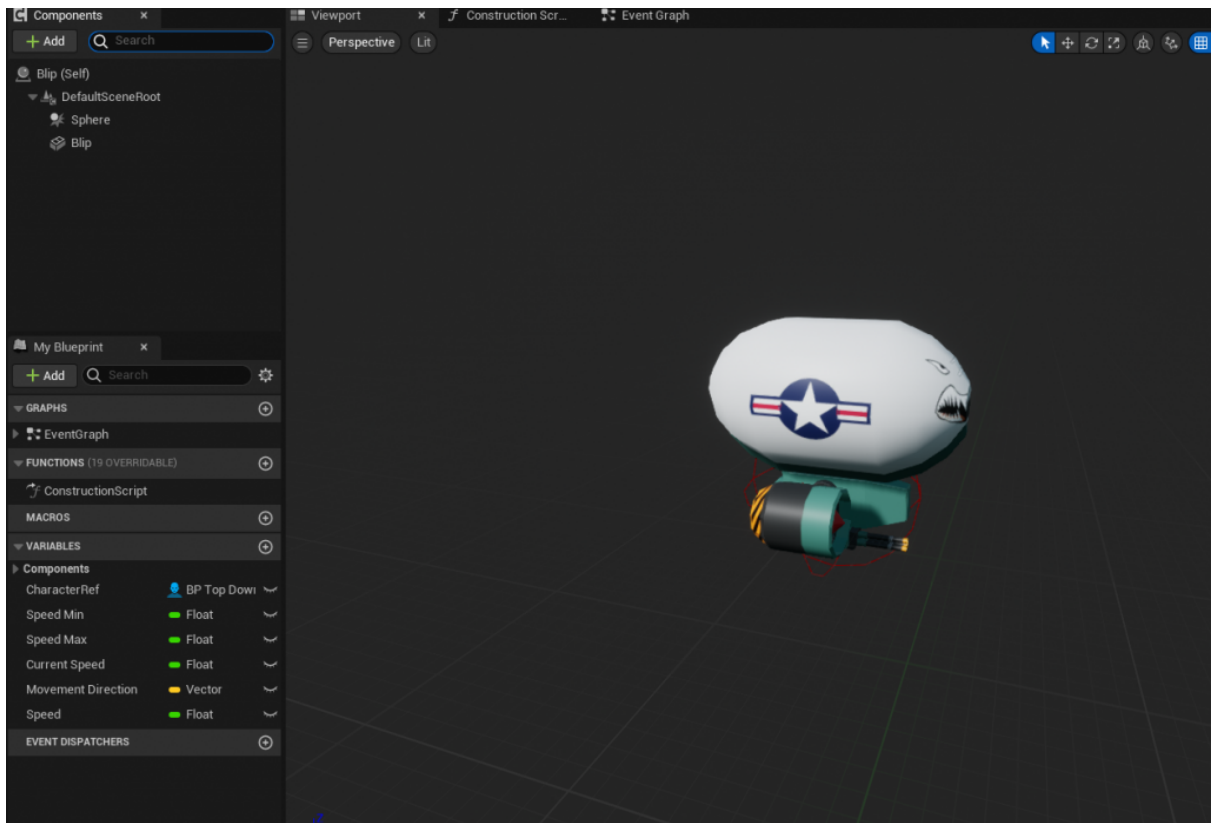


This is my Blimp blueprint which for on the deal damage overlap event we apply damaged with the damaged actors being other which is cast to player controller and get controlled pawn (which is our players controller) and then after apply damage we get cast to top down character and destroy actor so that the blimp explodes on impact destroying itself.

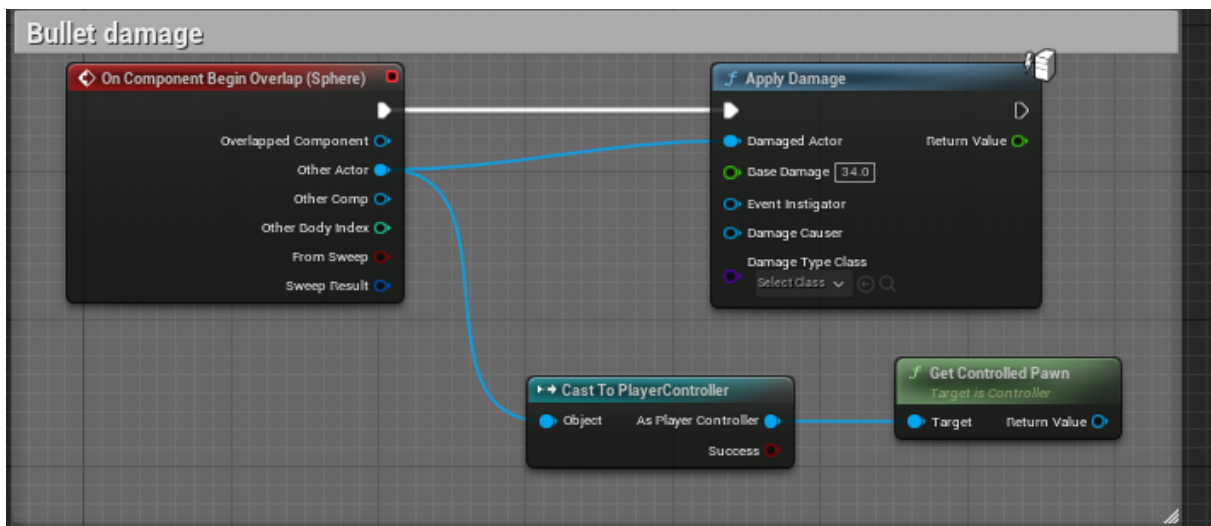
We also have my event tick which sets the actors location and on new location we add node by vectors, and get actor location and then multiply the add node to delta seconds in event tick and then another multiply node into get actor forward vector and our float speed variable.

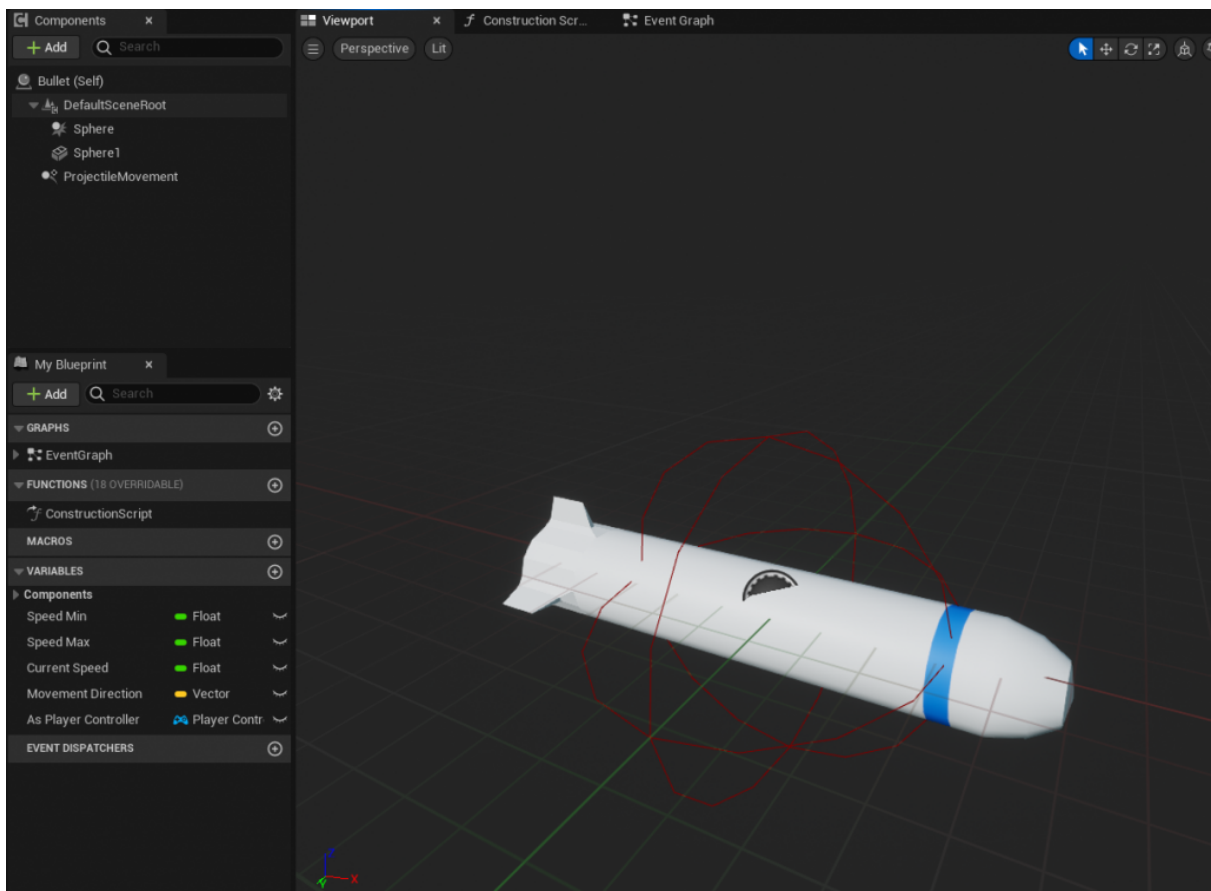
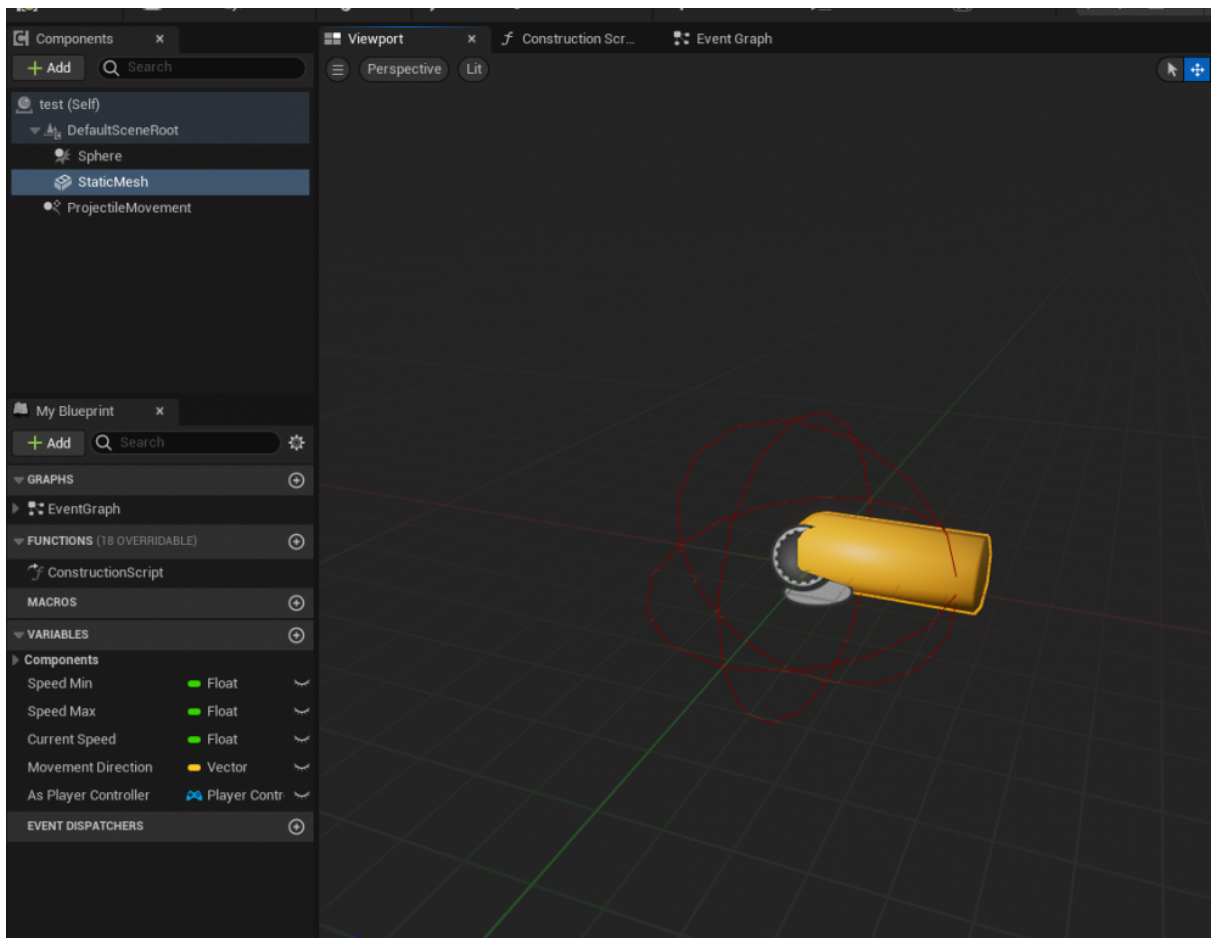


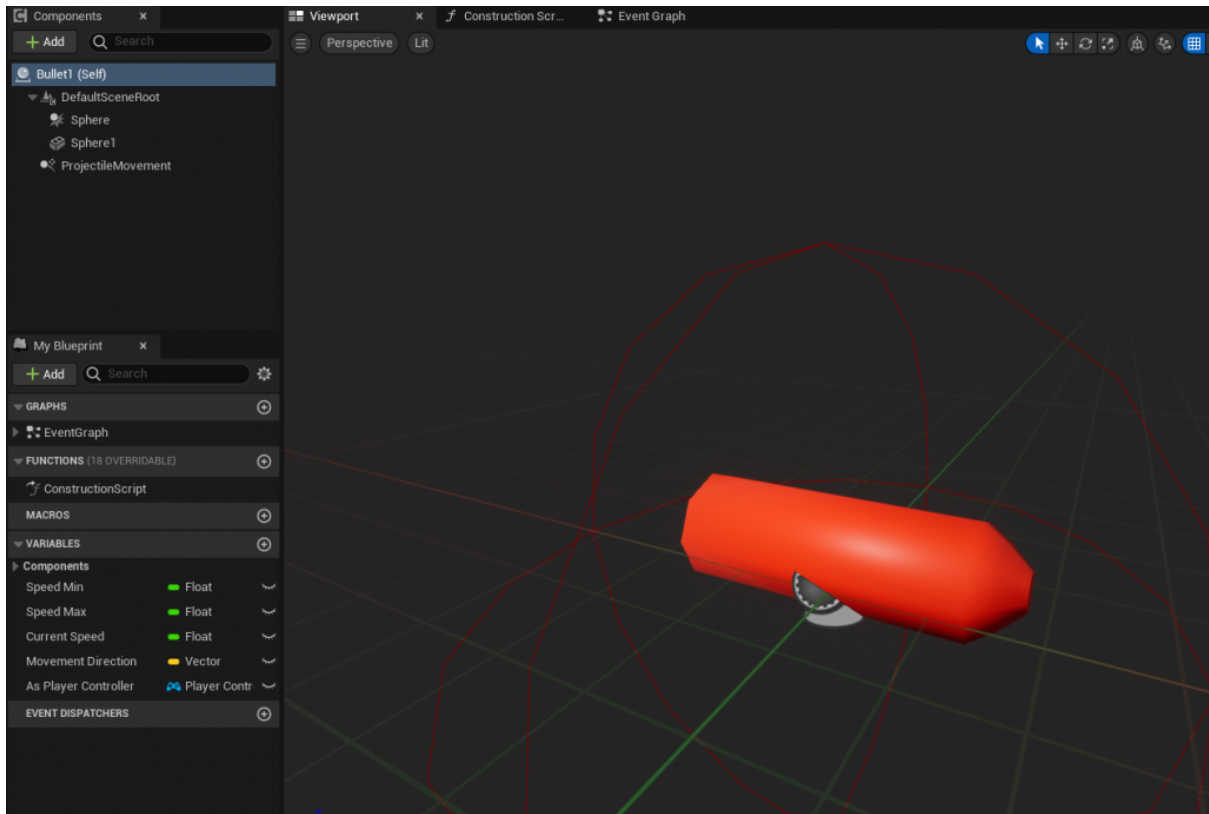
BLU



This is my bullet damage which is the same for all of my bullets as they all do the same damage as the player has three lives before they die.







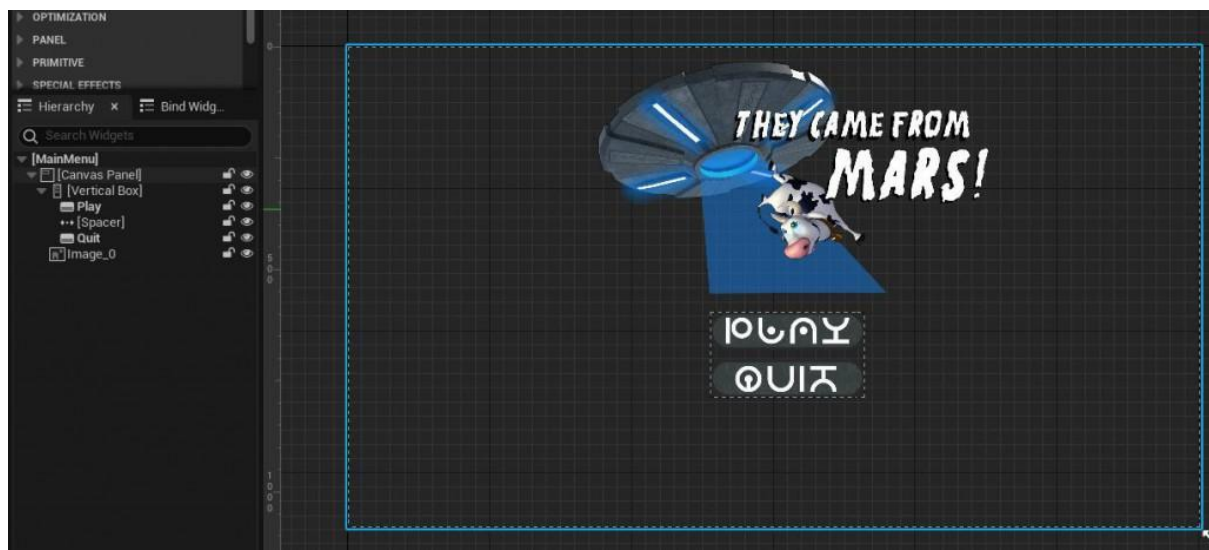
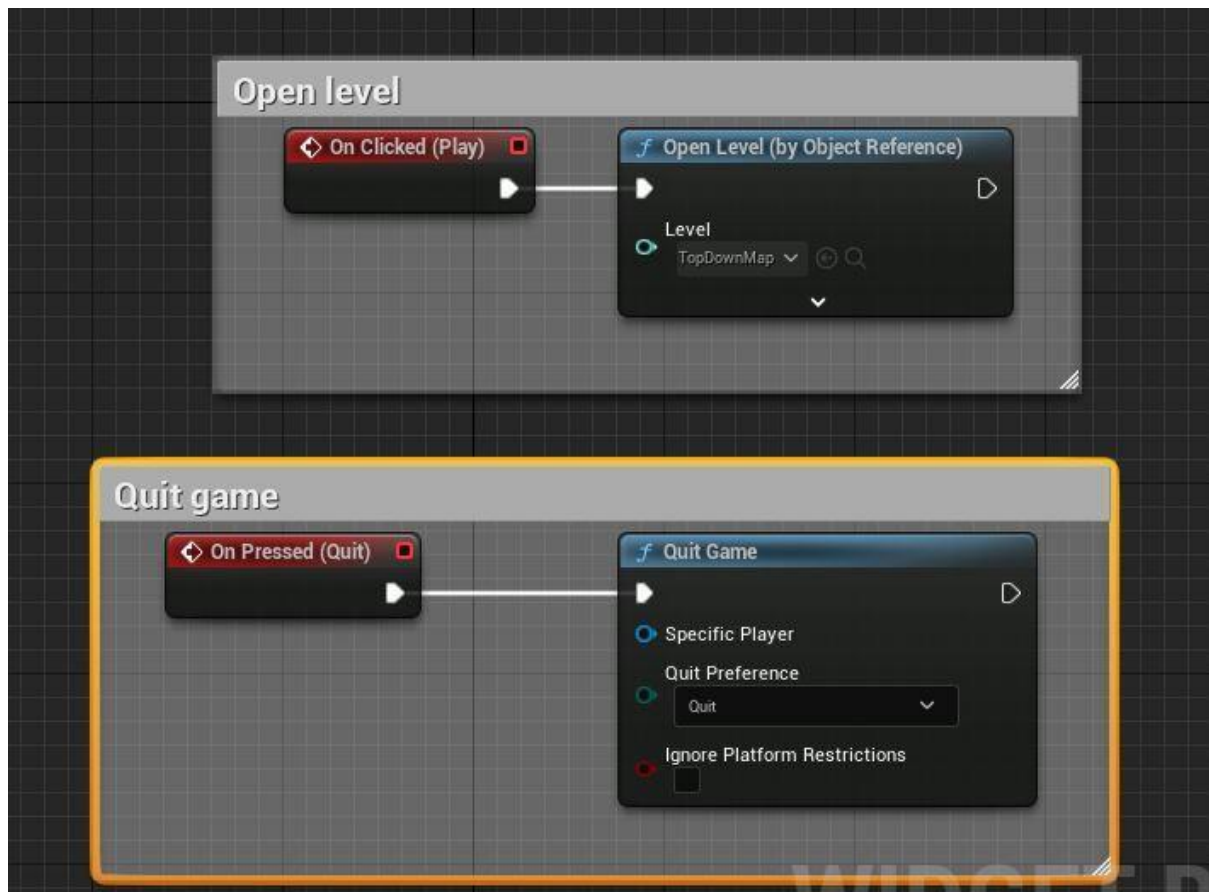
Video showcasing enemies

Technical & design development log #6

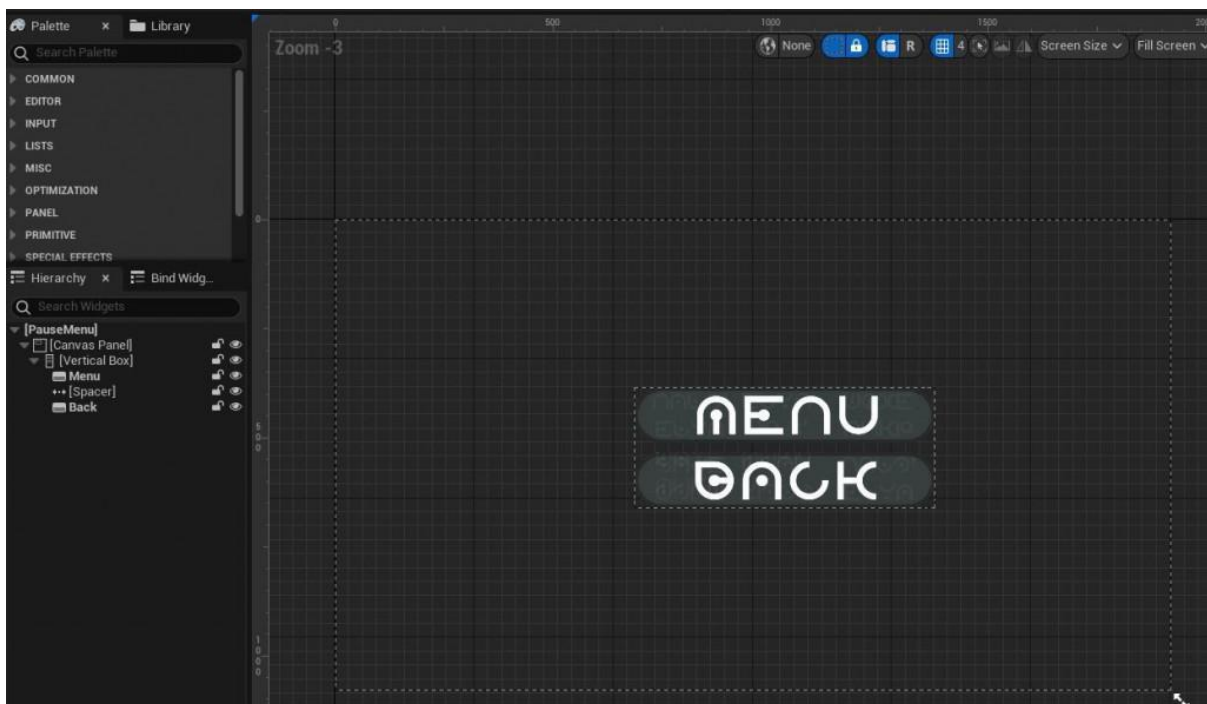
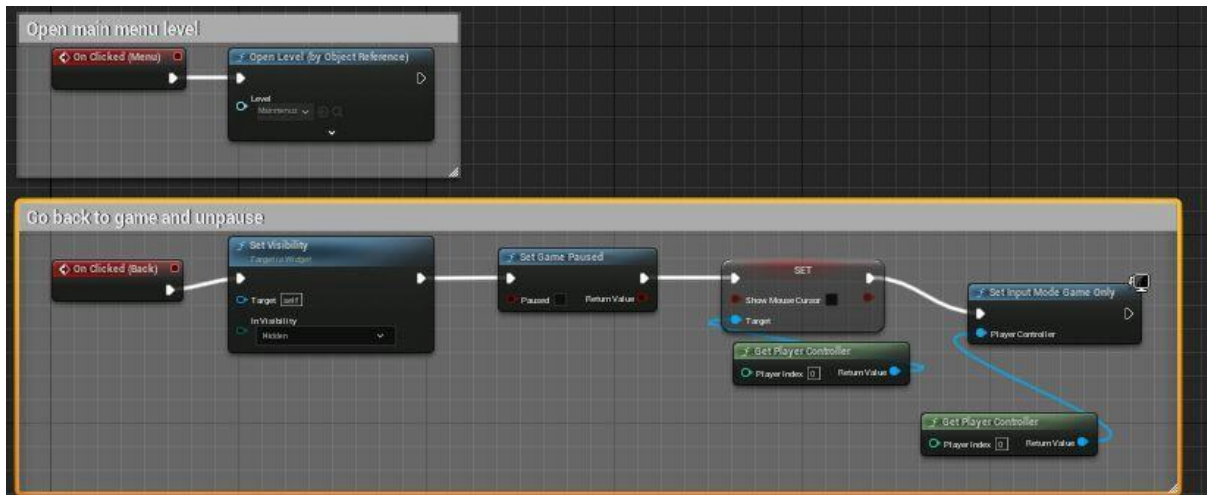
Widgets

This is my widgets that I created in my project.

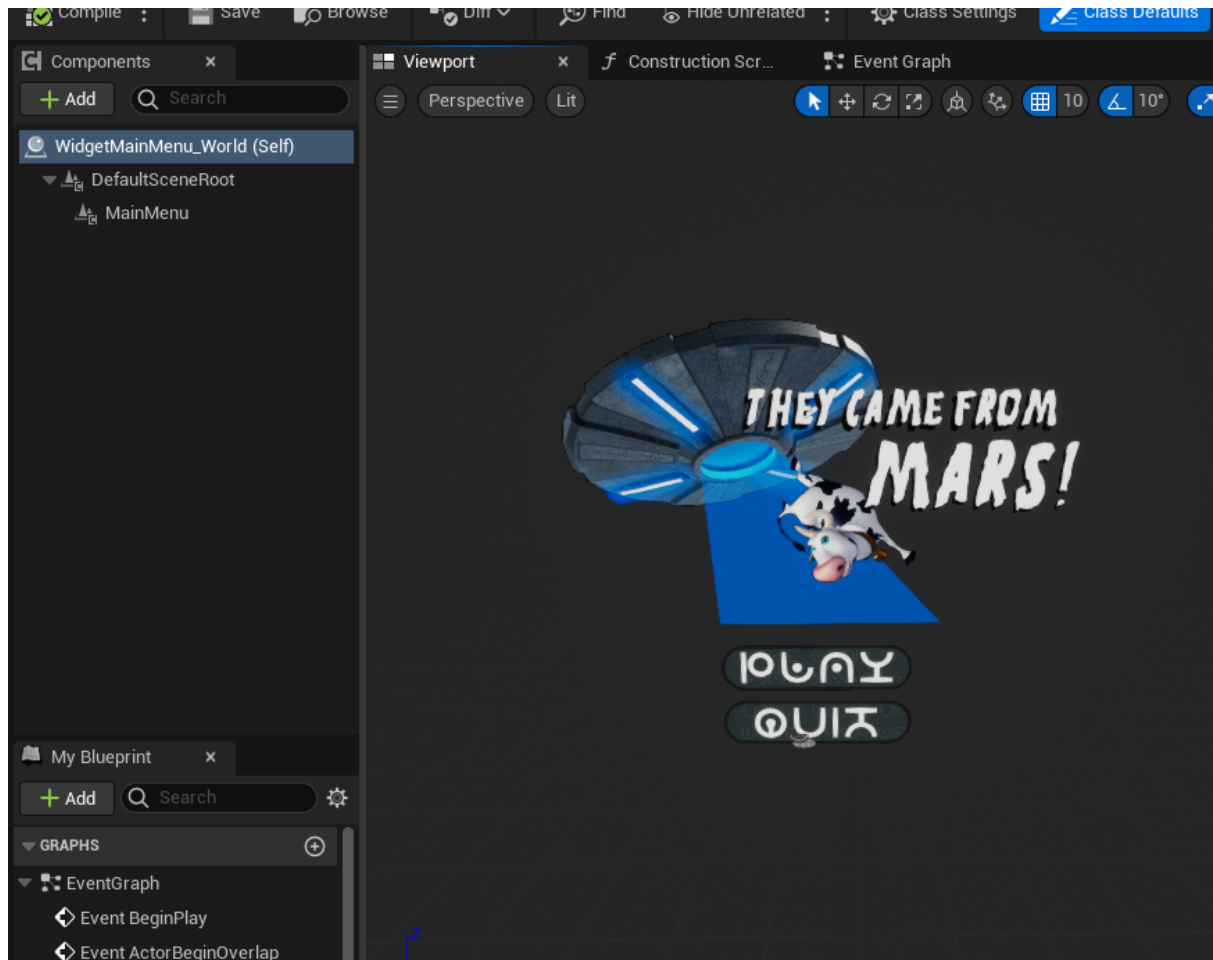
The blueprint within is basic but it achieves the functions needed for the widget, the on clicked play opens up the level and the quit game on pressed quits the game



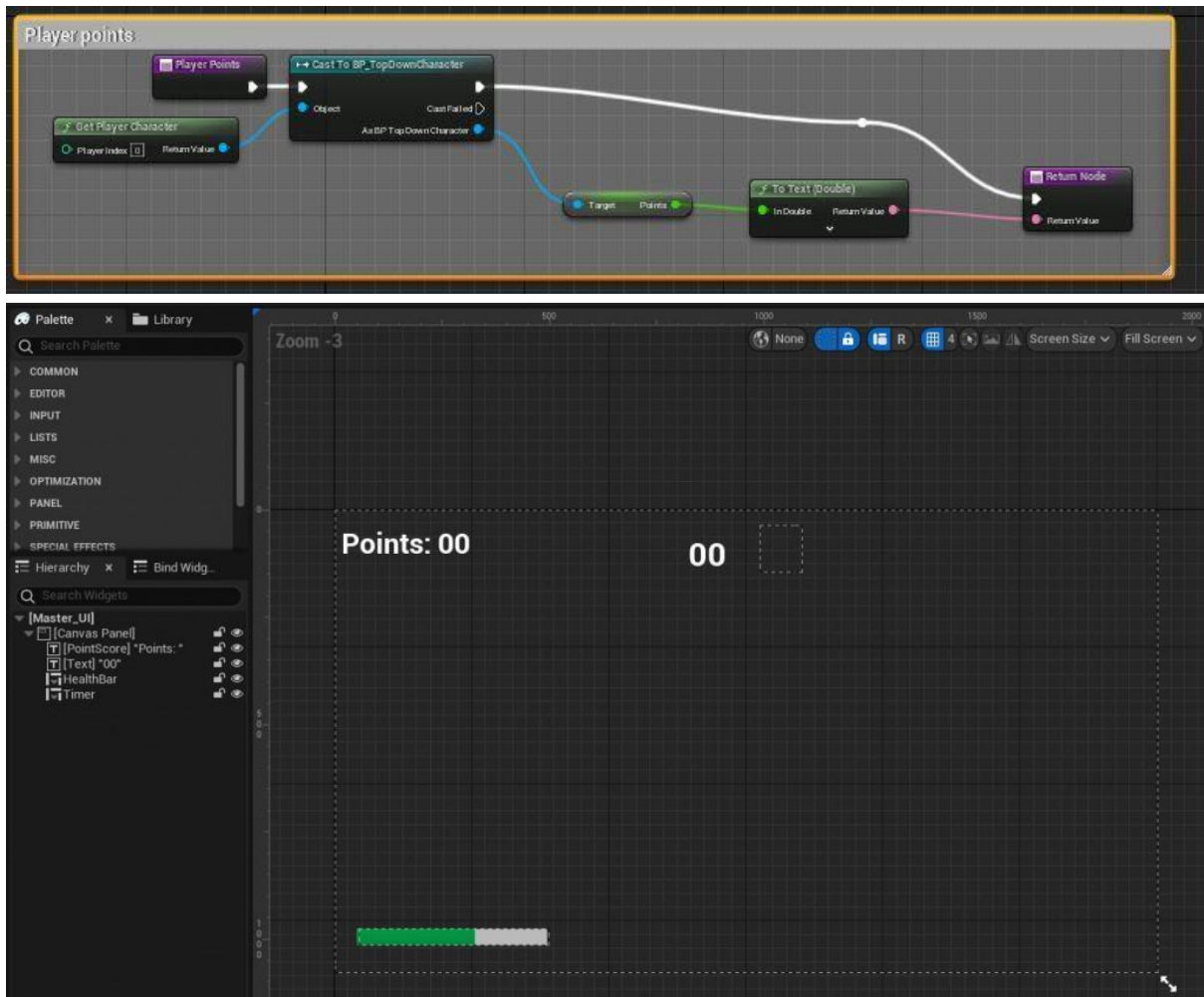
This is my pause menu blueprint and widget design on clicked menu we open up the menu level and on clicked back we set visibility of the widget to hidden and set game to unpaused and set mouse cursor to not show then set input mode back to game only.



This is my world widget actor which contains a widget component so i can get my widget to display in the world



This is my master UI design view then the event graph blueprint which cast to the player character and targets points which goes to a text double and returns it. So it displays the amount of points my character currently has.



This is what the final main menu looks like.

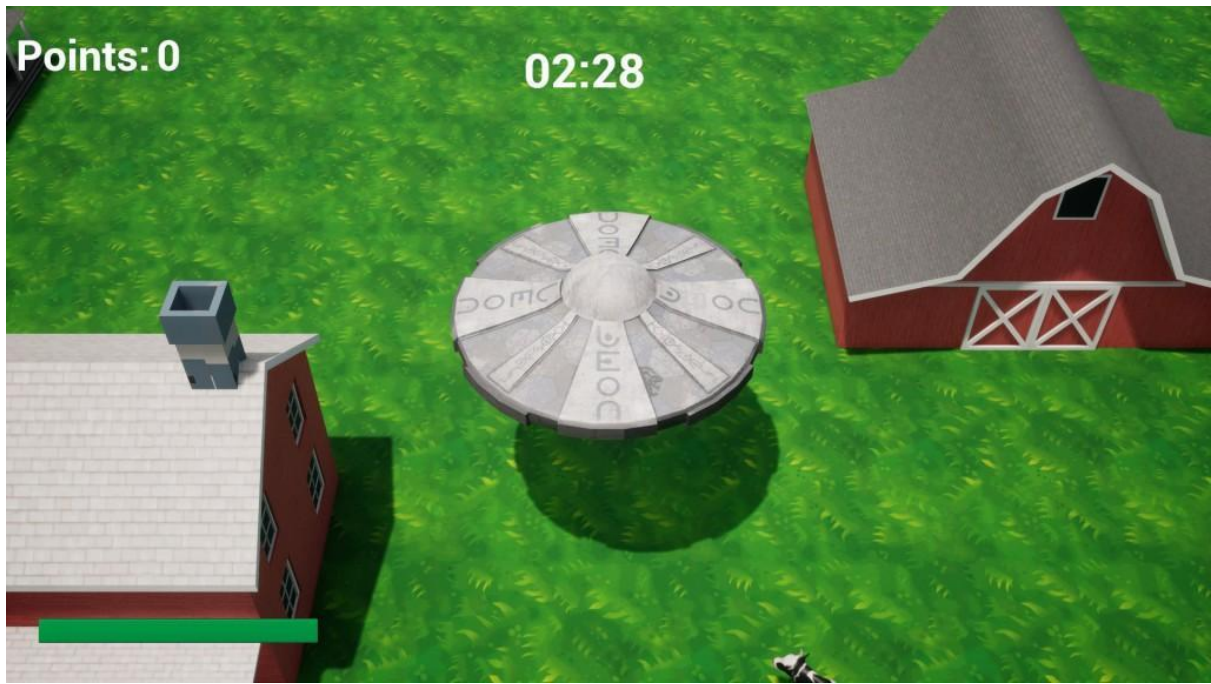


Technical & design development log #7

These are the in game screenshots from within my final level!



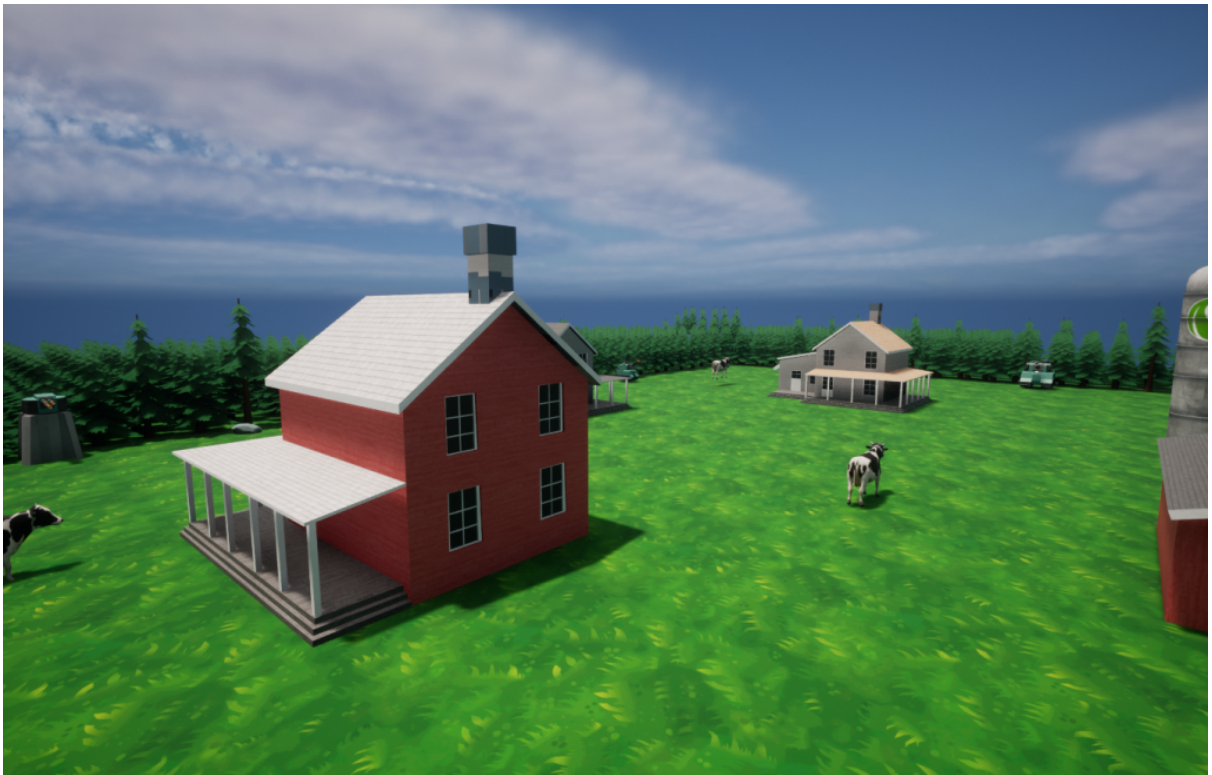




Technical & design development log #8

These are the screenshots of my assets within my final level!





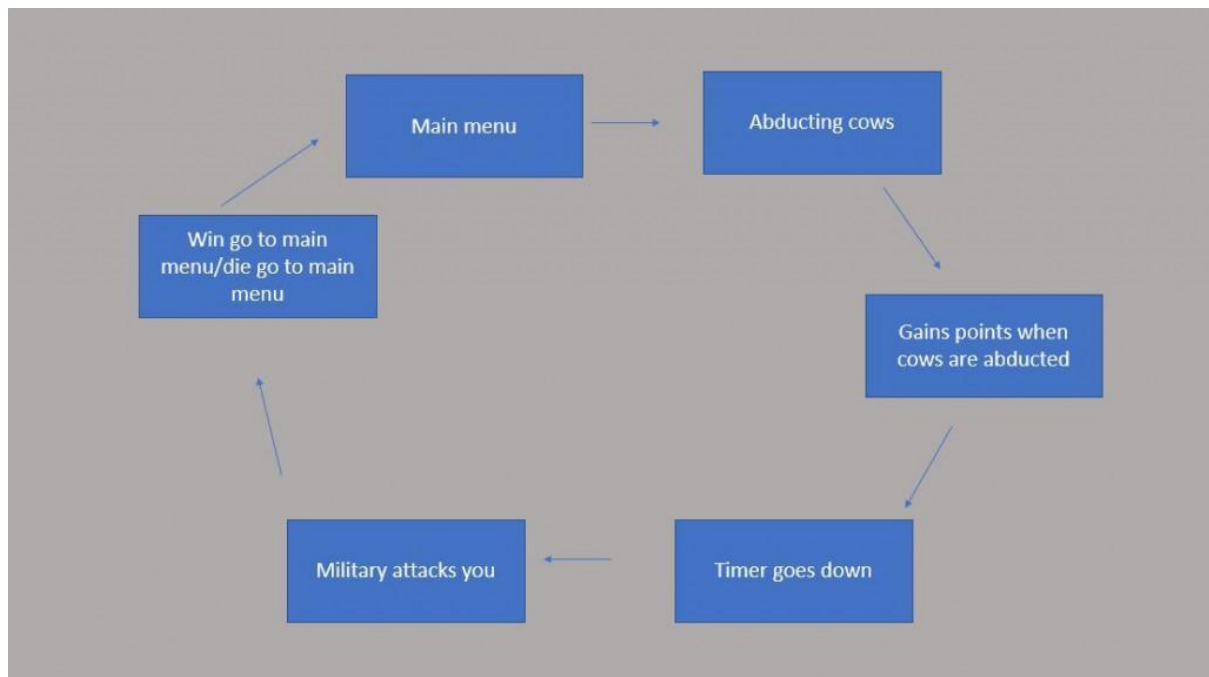




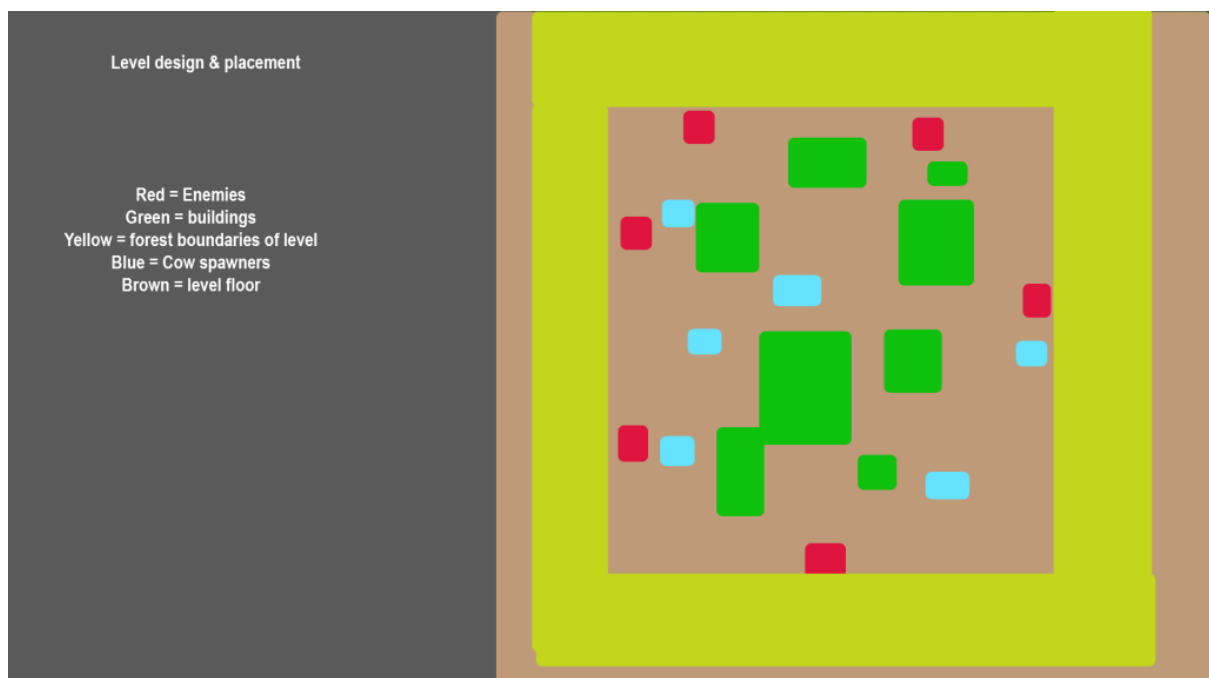
Technical & design development log #9

These are some of my designs for my game post production (forgot to post these before)

This is my gameplay loop I designed for my commercial UFO game, from the main menu you would go into game and start abducting cows, gain points, timer counts down, military attacks you, you survive the time you win and go main menu you die you lose and go main menu.

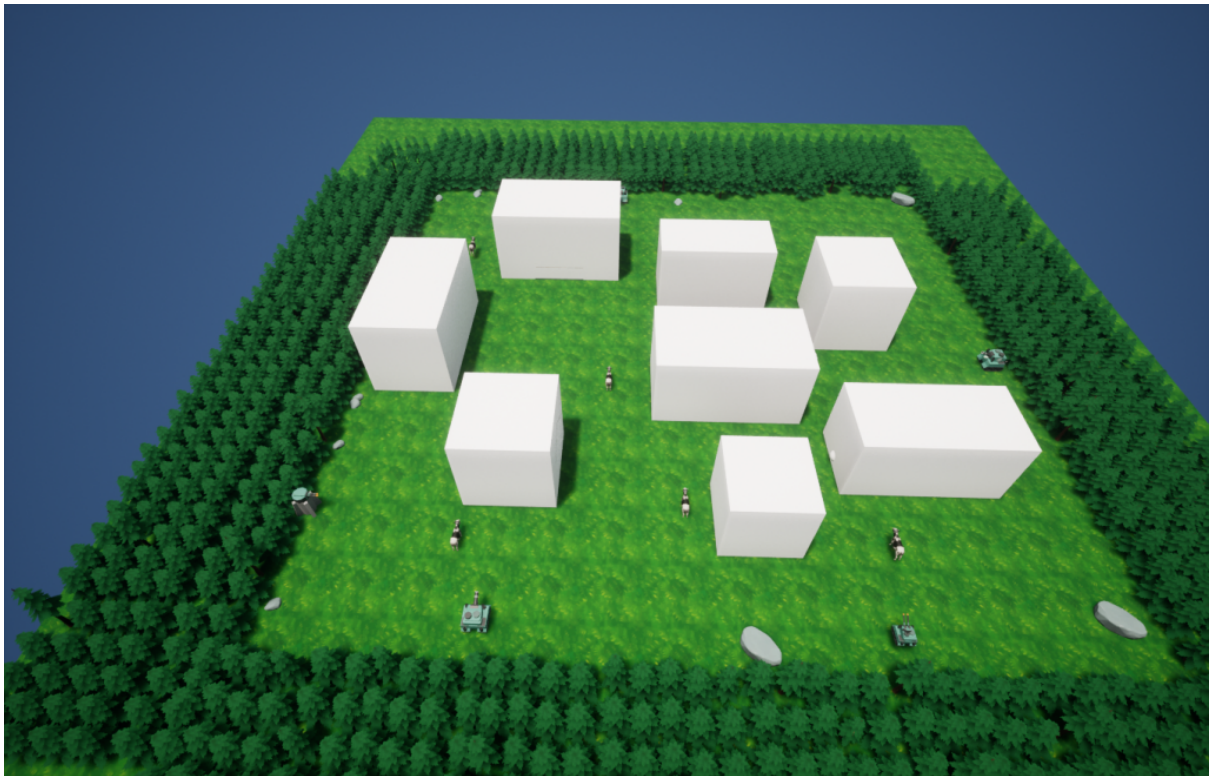


Level design process of designing how I want my level to be layout in photoshop with annotations.



This is when I blocked out where I want my building to be places once I had the assets in for level blocking. My design process thoughts for having buildings in my level was so while the enemy was

dodging the enemy military so they can abduct cows, they also need to swerve around the buildings so it's not as easy for them.



These below are my enemy balancing excel sheet I made to keep track of how much damage I wanted them to do and the delay between shooting the bullets, I ended up reiterating on my first iteration of the balancing as I found later on I wanted it so my character had 3 lives before they died instead of it varying rapidly depending on what you were hit by with diffren't damages, however perhaps in a future version if I added a difficulty setting I could have it adjust the scores in the data table and enumeration

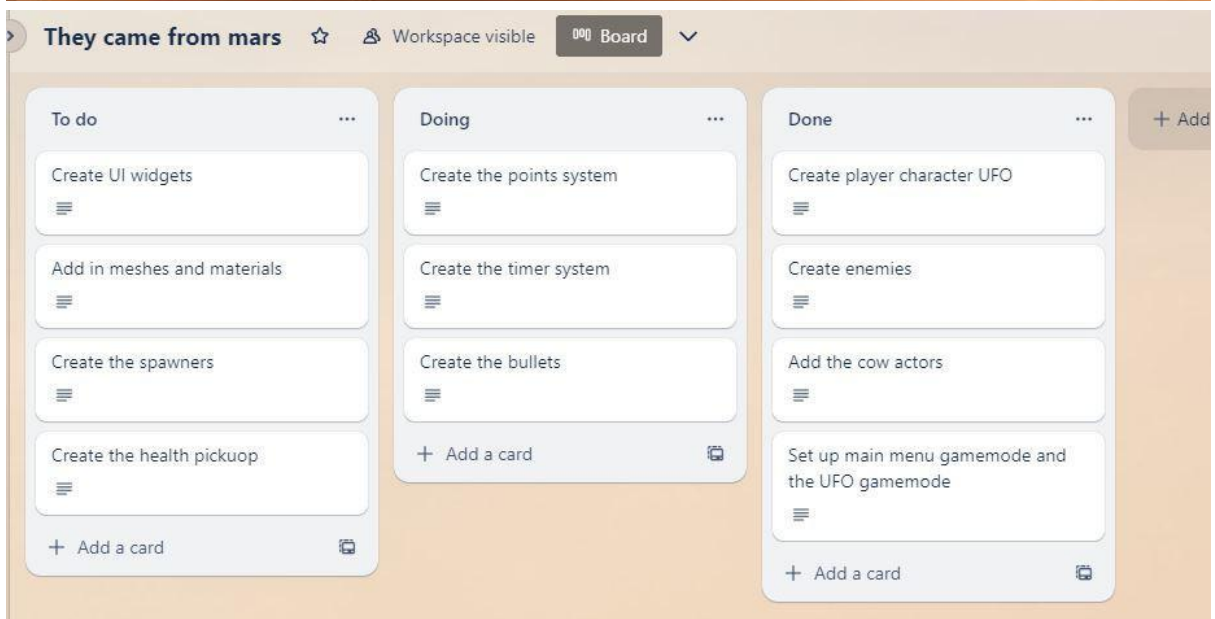
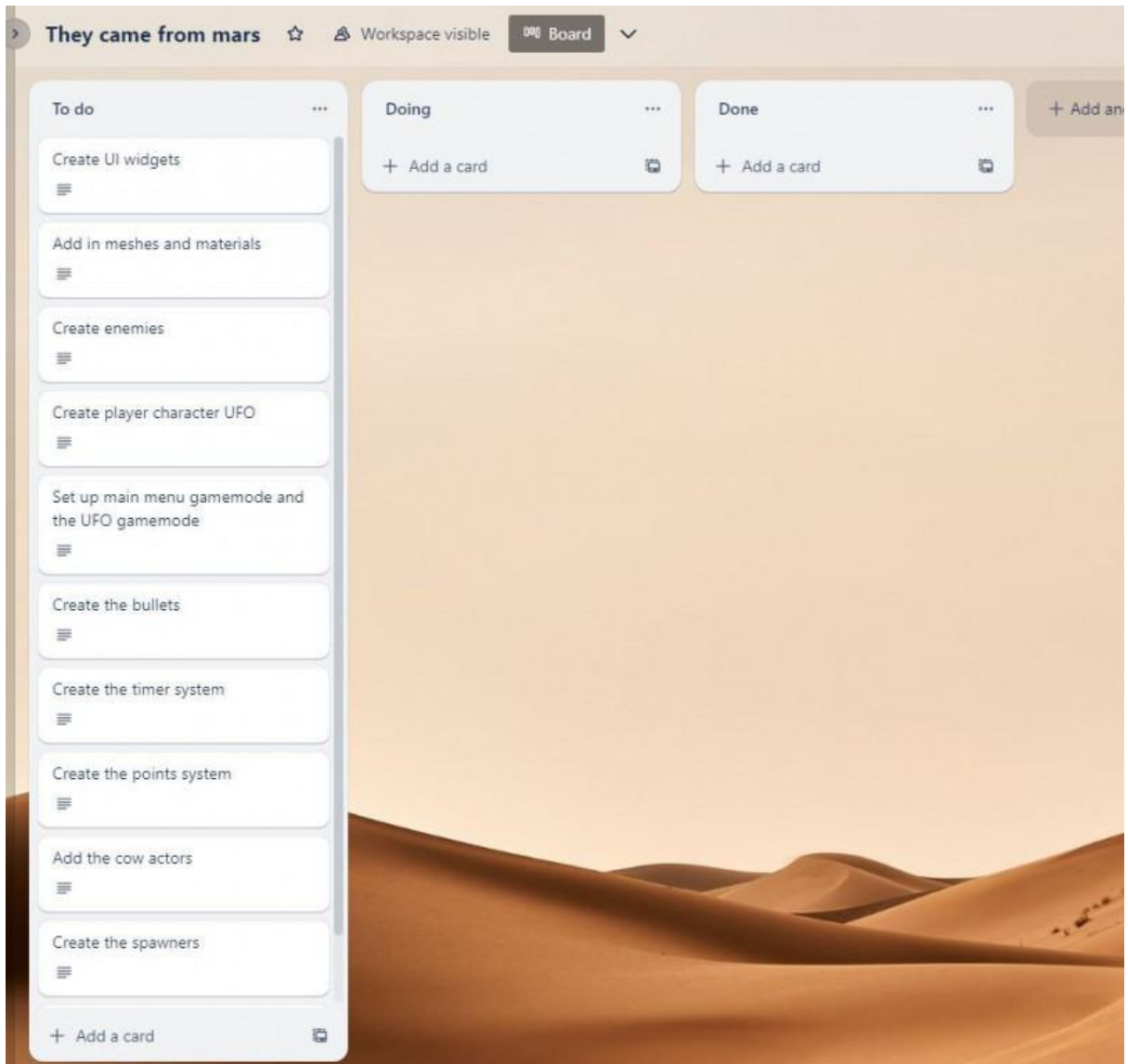
First iteration design balancing for enemies		
Column1	Column2	Column3
Enemy Name	Damage	Delay in shooting
Anti Aircraft	34	6 seconds
Tank	45	3.5 second
Turret	25	2 seconds
Blimp	60	5 seconds

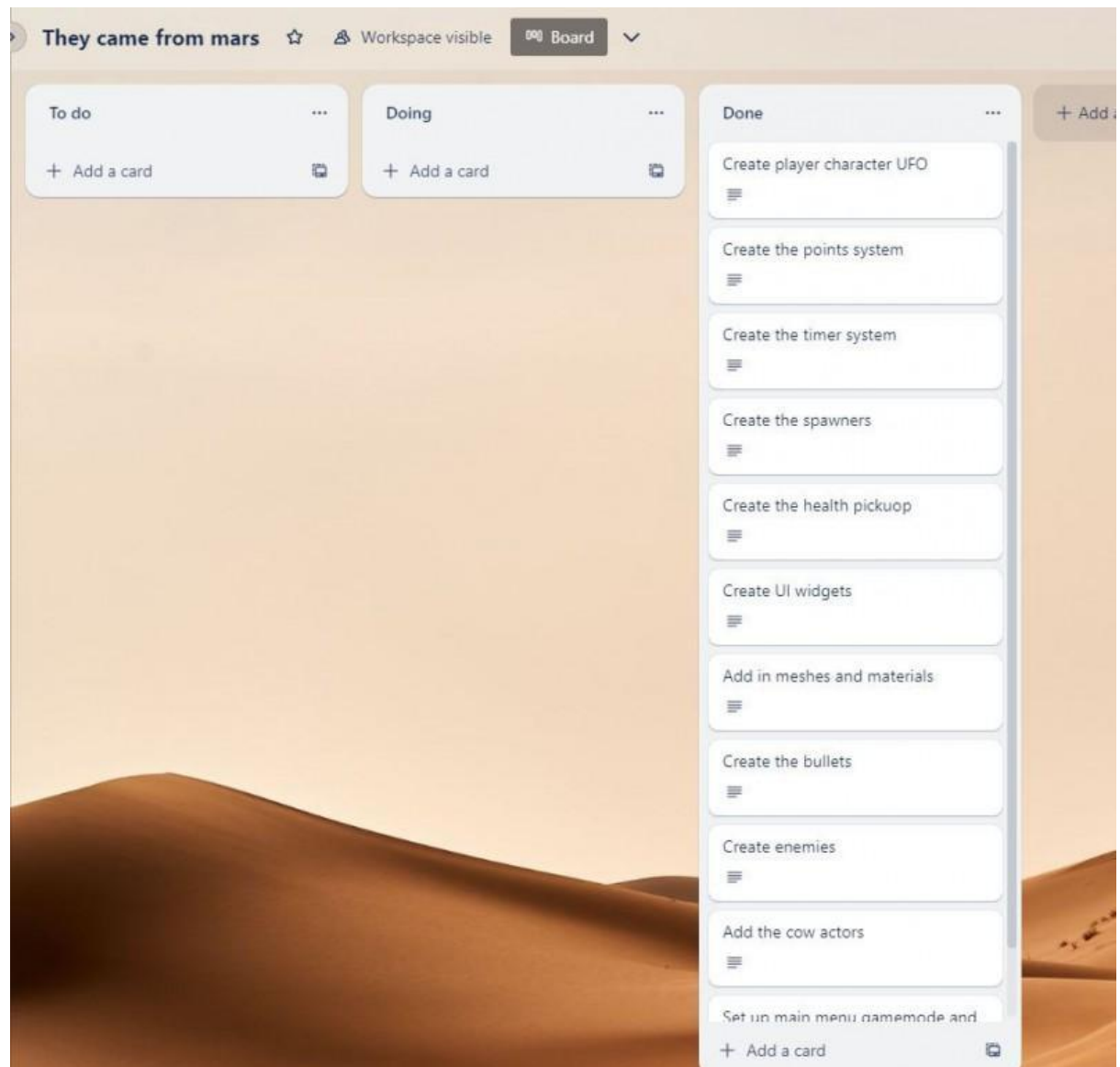
Final design balancing for enemies		
Column1	Column2	Column3
Enemy Name	Damage	Delay in shooting
Anti Aircraft	34	6 seconds
Tank	34	3.5 second
Turret	34	2 seconds
Blimp	34	1 second

Edited May 18 by KERSHAW Leon A

Technical & design development log #10

These are my trello project tracking screenshots where i used it to create tasks for myself and i used github mainly too which was the prime way of keeping track of my project





Create player character UFO

in list [Done](#)

Notifications

 Watch

Description [Edit](#)

Create the player character and set up the blueprint inside

- WASD movement
- UFO beam
- set up UI
- create health system

Activity

[Show details](#)

Create the points system

in list [Done](#)

Notifications

 Watch

Description [Edit](#)

Create a point system that tracks how many points you have when you abduct a cow

Activity

[Show details](#)

Create the timer system

in list [Done](#)

Notifications

 Watch

 Description [Edit](#)

Create a countdown timer that ends the game upon reaching 00:00

 Activity

[Show details](#)

Create the spawners

in list [Done](#)

Notifications

 Watch

 Description [Edit](#)

Create the spawners for

- Cows
- Health pickup
- Blimps

 Activity

[Show details](#)

Create the health pickup

in list [Done](#)

Notifications

 Watch

Description [Edit](#)

Create the healthpick up so the player can restore their health when damage

Activity

[Show details](#)

Create UI widgets

in list [Done](#)

Notifications

 Watch

Description [Edit](#)

Create the UI widgets for

- Health
- Timer
- Points
- Main Menu
- Pause menu

Activity

[Show details](#)

 **Add in meshes and materials**
in list [Done](#)

Notifications

👁 Watch

 **Description**

Edit

Add the meshes and materials for

- Buildings
- enemies
- UFO player
- bullets
- forest boundary
- Cows

 **Activity**

Show details

 **Create the bullets**
in list [Done](#)

Notifications

👁 Watch

 **Description**

Edit

Creates 4 types of the same bullet for each enemy with a differrn't mesh and same damage

 **Activity**

Show details

Create enemies

in list [Done](#)

Notifications

 Watch

Description [Edit](#)

Create the enemies listed below

- Tank
- Anti aircraft
- Turret
- Blimps

Add the cow actors

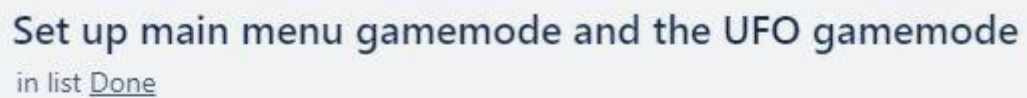
in list [Done](#)

Notifications

 Watch

Description [Edit](#)

Create the cow actor with the AI so it randomly moves around the map and then add it so it has 100 point upon abduction



© Watch

Description Edit

Set up the main menu gamemode and the UFO gamemode so that these can be separated to avoid issues on build

Activity

[Show details](#)

The screenshot displays the Unity Hub interface with the 'Changes' tab selected. The 'Changes' list on the left shows a commit 'Doroth01' with 8 changed files. The 'History' tab on the right shows a list of files, including 'Config/DefaultEngine.ini', 'Assets/Scripts/TransitionMap.cs', and 'Assets/Scripts/TransitionMap.cs.meta'. The 'Assets/Scripts/TransitionMap.cs' file is highlighted, showing its content.