

Alfie Fish - C2207339

Eclipse studios

Beta arcade 2024 - Teesside University

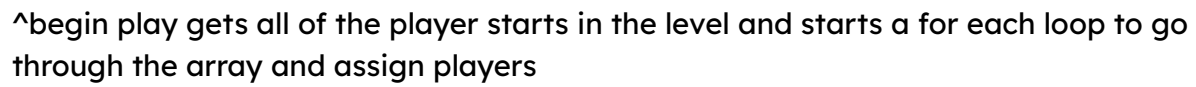
Beta Arcade self reflective report

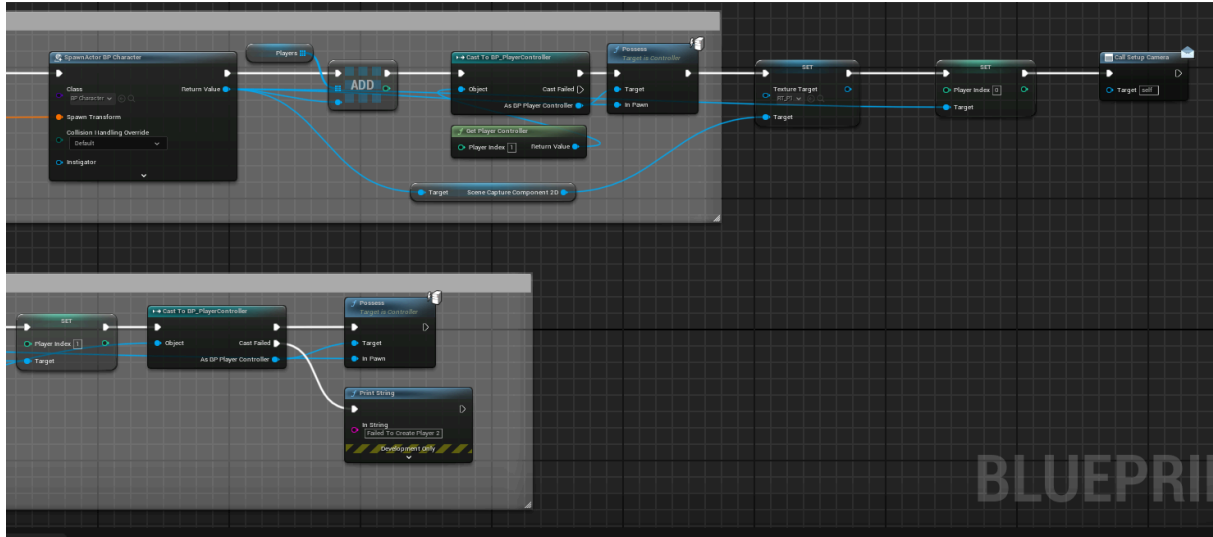
As part of the team eclipse studios we created a multilevel couch co-op top down shooter based around an arcade style game. The project we decided to pursue was heavily influenced by games such as Brotato and the Call of Duty Dead Ops Arcade. During this project I was assigned a programming role with a main focus on character and gameplay scripting. Due to this I split my time focusing between creating a character controller, player blueprint and creating the mechanics within game modes. Once I had completed these aspects of the project, I focused on developing a camera manager to ensure a fluid and dynamic multiplayer experience.

Individual contribution

Gamemode

I was tasked with making our game have a two player local multiplayer level as well as a single player alternative. I first had issues with assigning the enhanced input system to the second player so to resolve this, I had to create a new gamemode. This could be used at the start of game play to detect the number of player spawn points present in the level. It then creates a local player for each spawn point and possesses the player with their own character controller. This method worked well as it created an easily scalable gamemode which would allow our team to decide the number of players for each level if we decided to expand the game concept. Creating the new game mode was the most challenging part of this project for me to figure out however, I am happy with the result as it works well for the purposes and implementation we need, allowing for both single and multiplayer action.





^ this blueprint then uses the number of array elements to create 2 local players and possess the with character controllers before finally calling the set up event in the camera manager allowing play to begin

[Blueprint UE link](#)

Character Blueprint

As part of my role as a gameplay developer I was asked to script our character blueprint. For this part of my role, I created the input system and assigned the required input actions. After creating the basic movement mechanics for our player and an initial mounted camera on a spring arm, I discussed with my designers what mappings we required and the desired mechanics for our player character.

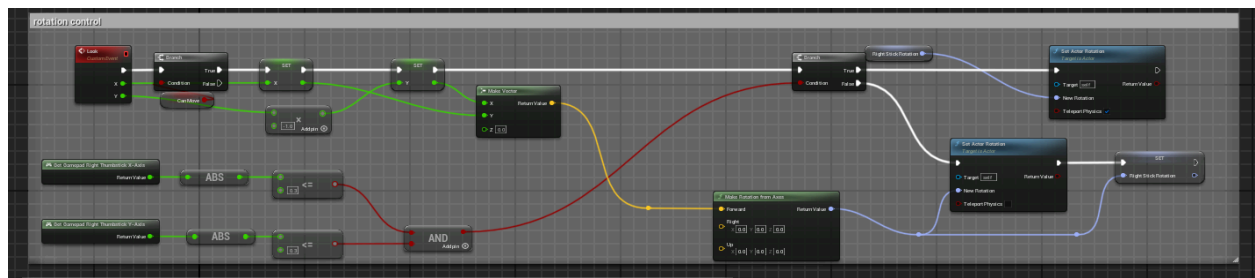
The features we required for our player were;

- Movement based on controller input
- Ability to fire a weapon
- A dash mechanic linked to a button press
- Death mechanic
- Spring arm top down camera (later was adapted)

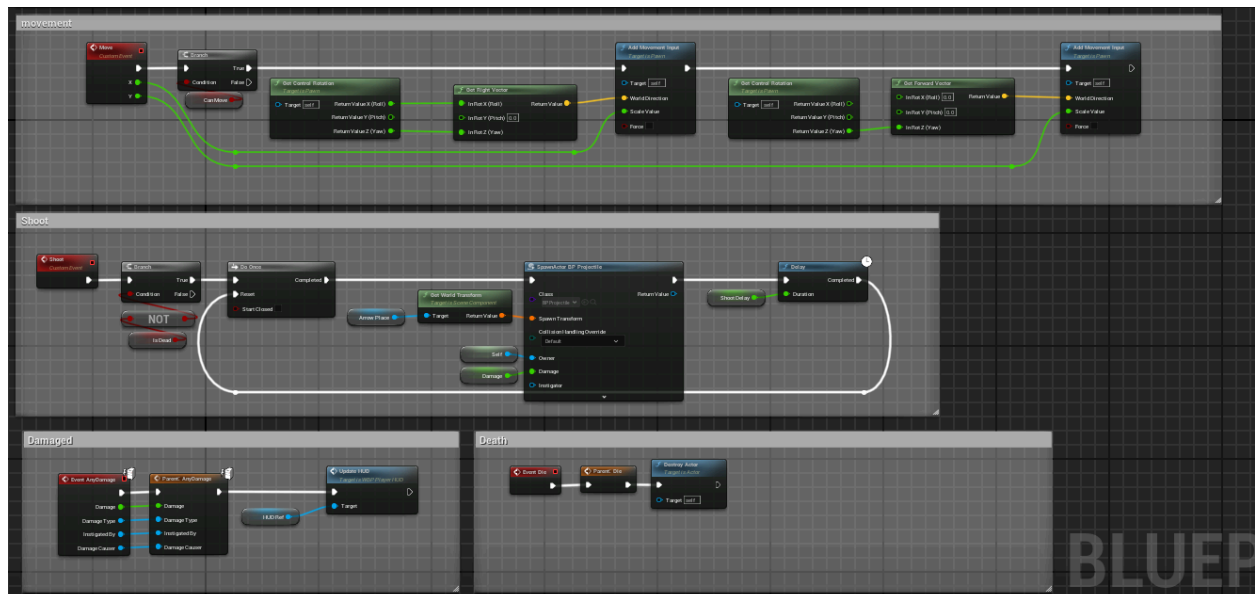
We decided to map our control scheme to a gamepad as we thought it would better fit the audience we would be targeting on an initial release. Our target audience consists of younger or more casual players. We also created a couch co-op implementation. Due to these factors, the team decided that the game would be more suitable as a console release. This allowed me to create an intuitive control scheme which would be easily understood by players of all ages as well as allow for fun friendly competition between players. This led to the input system seen below.

The character controller was edited and optimised by our programmer Tristan and so I will not display it here. It now functions through event dispatchers, which fire to the player blueprint, taking inputs from the input actions from the players gamepad.

After functionality was added to the player controller, I then turned my attention to the player blueprint. I used a basic actor with a projectile movement component and damage output on overlapping with another actor. This actor is spawned from the character blueprint when the weapon is fired using the transform of an arrow component attached to a socket at the end of the player's crossbow. I then concentrated on creating the player camera. I spoke with the team at length to get the correct angle and distance. This was to prevent negatively affecting the players field of view and allowing the correct game feel and vision of the level.



^character rotation



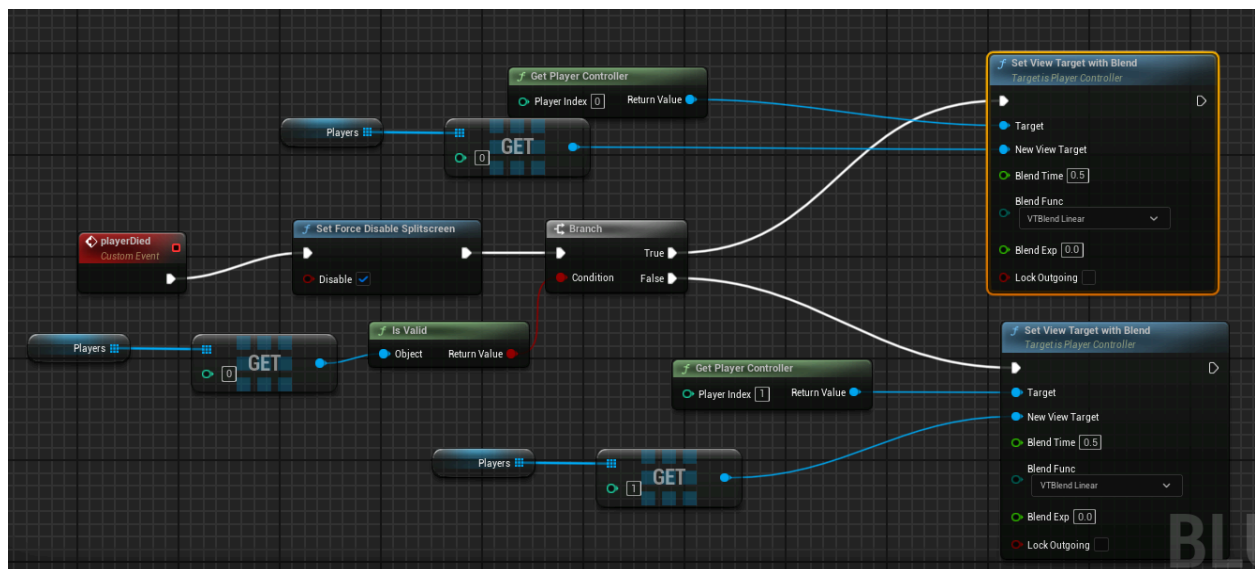
^character movement, shooting mechanic and damage taking

[Link to blueprint UE](#)

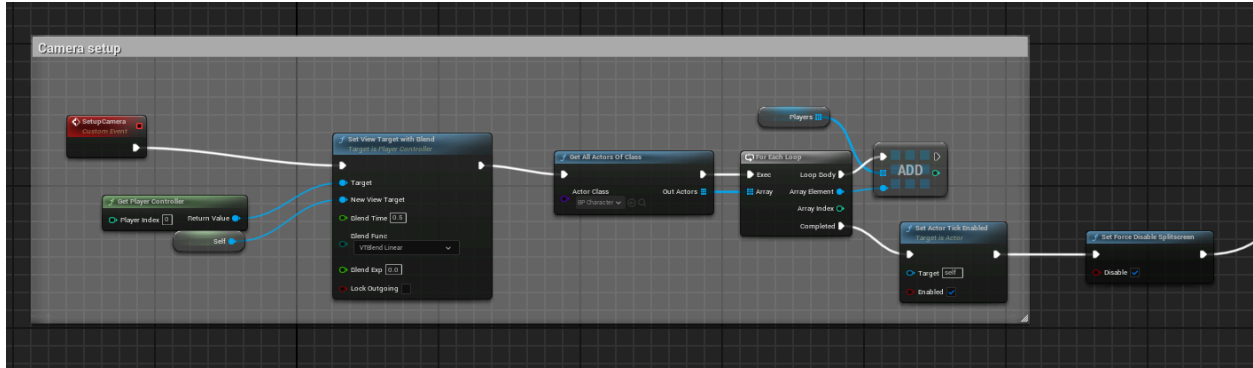
Camera manager

As a stretch goal for this project I was assigned the task of creating a dynamic split screen system. Initially, I planned to make a system which used the voronoi shape concept to fill the screen with both player views in a similar way to that used by the lego games. Upon further research I had realized this was not realistic within the time I had remaining to complete this project and could create confusion for the players if not fully polished before the end of the module. I instead opted for a method using a separate camera manager which could be placed within the level. This camera would update the length of the spring arm thus expanding the view of the camera as players moved away from each other and would then enter the split screen view once they had left the camera's maximum field of view. This camera would also update its position to remain central in between two players as they moved around the level.

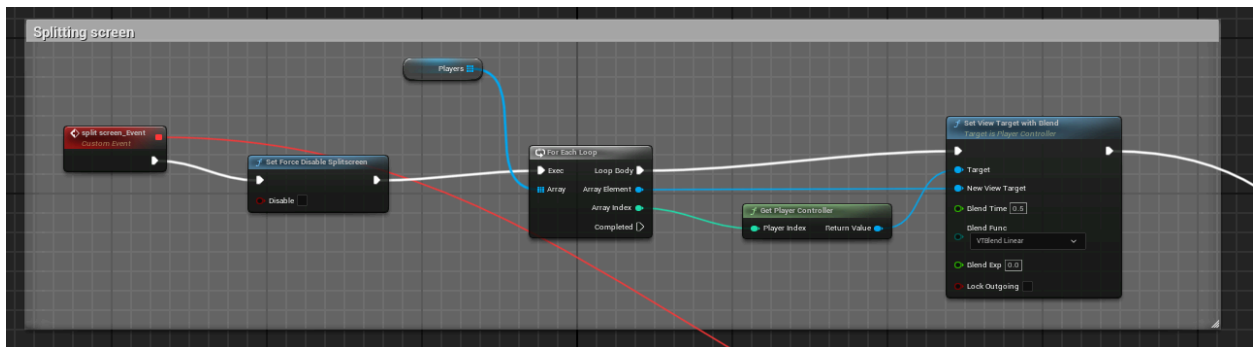
Technically the implementation worked well however could have been more optimised by not relying on the event tick. It worked by creating variables using the two players in the level and then finding their locations and the distance between the manager then used those distances to adjust the length of the camera arm to keep both players in view. Also using this distance, it would then alter the location of the camera to remain an equal distance from both players and the camera view expanded and contracted.



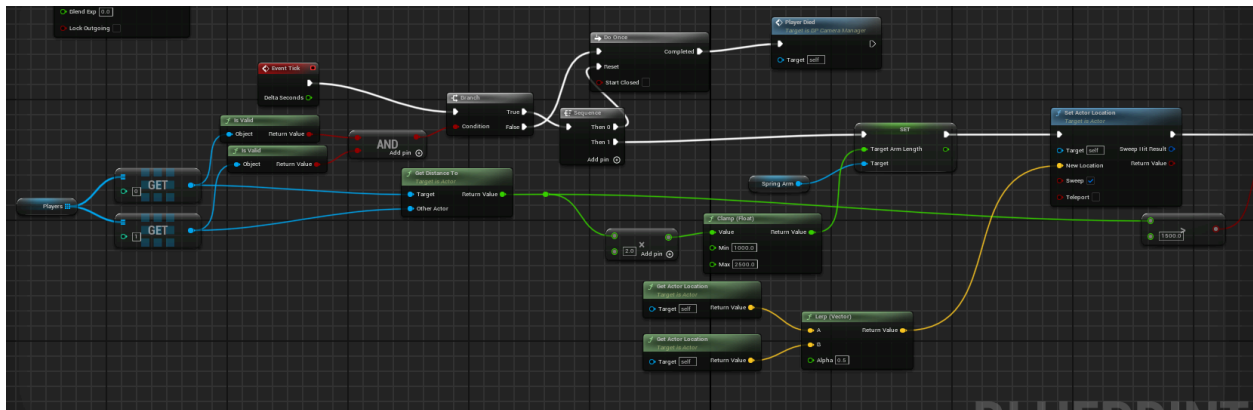
^This event handles the reset of the camera after one player has died



^Initial camera set up, this event is called to set up the camera on the game start. It also handles combining the cameras when the players return to close proximity



^This code split the camera when the players move out of the distance parameters



^This is the main function of the BP which registers the distance and adjusts both location and the height of the spring arm.

[Link to Blueprint UE](#)

Team evaluation

Team break down

- 1 game developer
- 4 game designers
- 1 games programmer
- 2 3D artists
- 2 animators

Our team was mostly made up of an even division of disciplines which allowed for even development of the project in all areas. During the beginning stages of the game, development was quick. Due to the simplicity of our core mechanics, a working demo was created by week four. The character art was also created quickly, the main character was finished and ready for rigging shortly after. Prop art however lacked behind with one of our artists missing check in messages and deadlines for initial props to assist the level designers. This then became a common issue throughout the remainder of the project with team attendance beginning to decline as the weeks progressed leading to a breakdown of communication. Our group lead was absent for multiple lab sessions which created issues with assigning tasks and enforcing deadlines, this created the need for a few members of the team (myself, a designer and an artist) to step up into a more leadership focused role. As part of this, we oversaw project management alongside developing our personal contributions to the project. We became more vocal within the team discord and had knowledge of the different team members and the issues they faced which halted their development. The team members responded well to this change and it allowed us to move forward and resume our initial pace without overburdening one lead. The team was split under 3 leads by discipline which made multi discipline communication easier due to the differing technical knowledge we held as well as offering more opportunities for discourse within the team.

One area we were particularly lacking was playtesting. This could be attested to the communication breakdown which occurred in the middle of the project as organising completed mechanics became difficult and therefore getting a fully merged gameplay loop for playtesting was often not possible. We were able to complete in house play testing which allowed the opportunity for the team to voice valuable feedback. However, outside opinions on the game feel and development milestones would have been invaluable and helped the team whilst polishing and finishing the projects in the final weeks. This lack of outside playtesting and opinion was our greatest issue as we had to rely on in-house feedback which could have been tainted by a personal experience with the game and inside knowledge of development and gameplay.

Despite these issues work did continue and I feel we completed the brief and created a functional, accessible and fun game. Many team members who had issues and lower attendance were still working on the project in their own time and created useful and technically competent designs, scripts and assets which have been used in the final project. Although communication was an issue, workflow for many of the team stayed on track throughout the module and we were able to create a finished game ready for presentation.

Personal evaluation

My role for this project was that of a gameplay scripter, as part of this my main focus was to create and implement a character as well as a system to handle the multiplayer and camera elements of our game. I feel that I completed my tasks within this role to a high standard and was able to work within the team well. My responsibilities included creating a game mode to work around a bug which has been found within the unreal engine version 5 and carried through to version 5.4. This bug had caused the local multiplayer and input assignment to not properly function within the project settings. Due to this I was required to implement a manual creation of local players and assignment of player controllers. I found this to be the most difficult part of the project and was proud of my ability to adapt to the issues I encountered and create a solution which fit with my team's needs. The creation of the multiplayer functionality was not the only challenge encountered during the development of our project however I feel that I was able to adapt and overcome creating more complex and optimised scripts than I have in previous modules. This project has allowed me to take on a technical role which i hope to carry forward into my time within the games industry

I also took on a more leadership focussed role for the technical development side of our project working with our lead and other members of the team to create and give our presentations to the cohort. This task was not originally meant for me however it became a more efficient way for the team to work allowing for team members to discuss issues with others of their own discipline and work towards solutions better than with a singular generalised lead. I feel I helped the team to communicate more effectively and have personally developed my skills and confidence discussing and explaining technically developed concepts within both small groups and in front of large crowds. My confidence in my ideas and implementations has grown as well as my ability to work alongside others both aiding my peers and asking for help and learning new methods and scripting techniques. The beta arcade module has allowed me to take lessons and methods learned from previous projects such as journeyman and apply them to my workflow within my current team, previously I have been required to step into leadership roles and organise communication and the assignment of tasks.

This project has given me the opportunity to develop both my personal and professional skills which can be taken into industry as my career progresses after university. I have progressed as a technical developer and increase my skills and proficiency within the unreal engine, i have been able to implement optimised and high level solutions to many problems which have arisen during the project as well as discuss and give explanations of the methods behind my blueprints to members of the team from multiple disciplines. I was also able to seize opportunities to present both my own and my team's work to my peers at university. I helped the team to communicate to our tutor Frank who was detrimental in the creation of our project.