**Multifaceted**

**Perspective**

**Report**

게임 소프트웨어 학과

C193354

이원용

**목차**

1. **Main Concepts**
   1. **시점 변환**
2. **References**
3. **Resources**

**Main Concepts**

* 3인칭, 1인칭, 2D, 2.5D 등을 총 망라한 플레이를 시도하고 싶었습니다.
* 따라서 상황에 따라 player controller 와 game condition을 변화시키는 것을 시도했습니다.
* 창의적인 게임 보다는 무엇을 할 수 있는지를 보여주는 것에 좀 더 초점을 맞추었습니다.

**시점 변환**

* 게임에서 가장 크게 보여드리고 싶었던 부분입니다.

1인칭 <-> 3인칭의 시점 변환은 크게 어렵지 않았습니다. 가장 대중적으로 사용되는 방법은

만화 영화, 블루이(가) 표시된 사진

자동 생성된 설명

이러한 방식으로 3인칭 카메라와 1인칭 카메라를 따로 설정한 뒤

스크린샷, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명

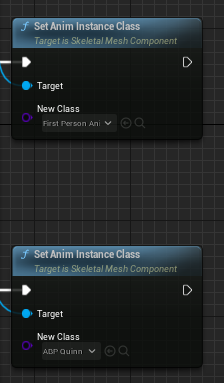
지정된 키 (프로젝트 에서는 v)를 할당하고 해당 키가 눌렸을 때 flipflop 함수를 사용하여 각 카메라의 activate를 전환하고, 1인칭 시점일 때에 자연스럽게 모델이 플레이어의 마우스를 따라 움직일 수 있도록 Set -> Use Controller Rotation Yaw를 해주는 것이었습니다.

해당 방법은 간단하지만, 제가바라는 1인칭과는 괴리감이 있었습니다. 저는 fps에 가까운 1인칭을 원했으나 카메라는 플레이어의 목에 붙어 FPS와 비슷한 느낌만 줄 뿐이었습니다.



그에 따른 해결책으로 제가 생각 한 것은 Player 모델의 Skeletal Mesh를 시점이 바뀌는 순간마다 바꾸어 주는 것이었습니다.

1인칭 모션에서는 플레이어의 팔만 존재하면 되었고, 모델의 Skeletal Mesh만 변경하면 된다는 생각에, 바꾸는 과정은 그리 어렵지 않았습니다.

본래의 Mesh를 시점 변환에 따라 Skeletal Mesh와 함께 변경시켜주는 부분
 라인이(가) 표시된 사진

자동 생성된 설명

다만 적용 이후의 문제점은, firstPerson의 Animation이 필요하다는 것이었지만… 애니메이션이 제대로 작동하지 않는 문제점이 있었습니다.

거기에 더해 플레이어의 모델을 바꾸어 버리니 그림자도 이상해 지는 문제점이 있었습니다.

따라서 본래의 방법을 사용하는 대신, FirstPersonCamera 의 Fov(Field of View) 값을 조정하는 조금의 편법을 사용하였습니다.

**References**

* **How to make a 2.5D Platformer in Unreal Engine 5 - Beginner Tutorial: by Cobra Code**

**(**[**https://youtu.be/ZGQoo9frBUA**](https://youtu.be/ZGQoo9frBUA)**)**

* **How to Switch between Third Person and First Person in Unreal Engine 5 - In 3 mins!: Gorka Games**

**(https://youtu.be/lMuinhr0SXU)**