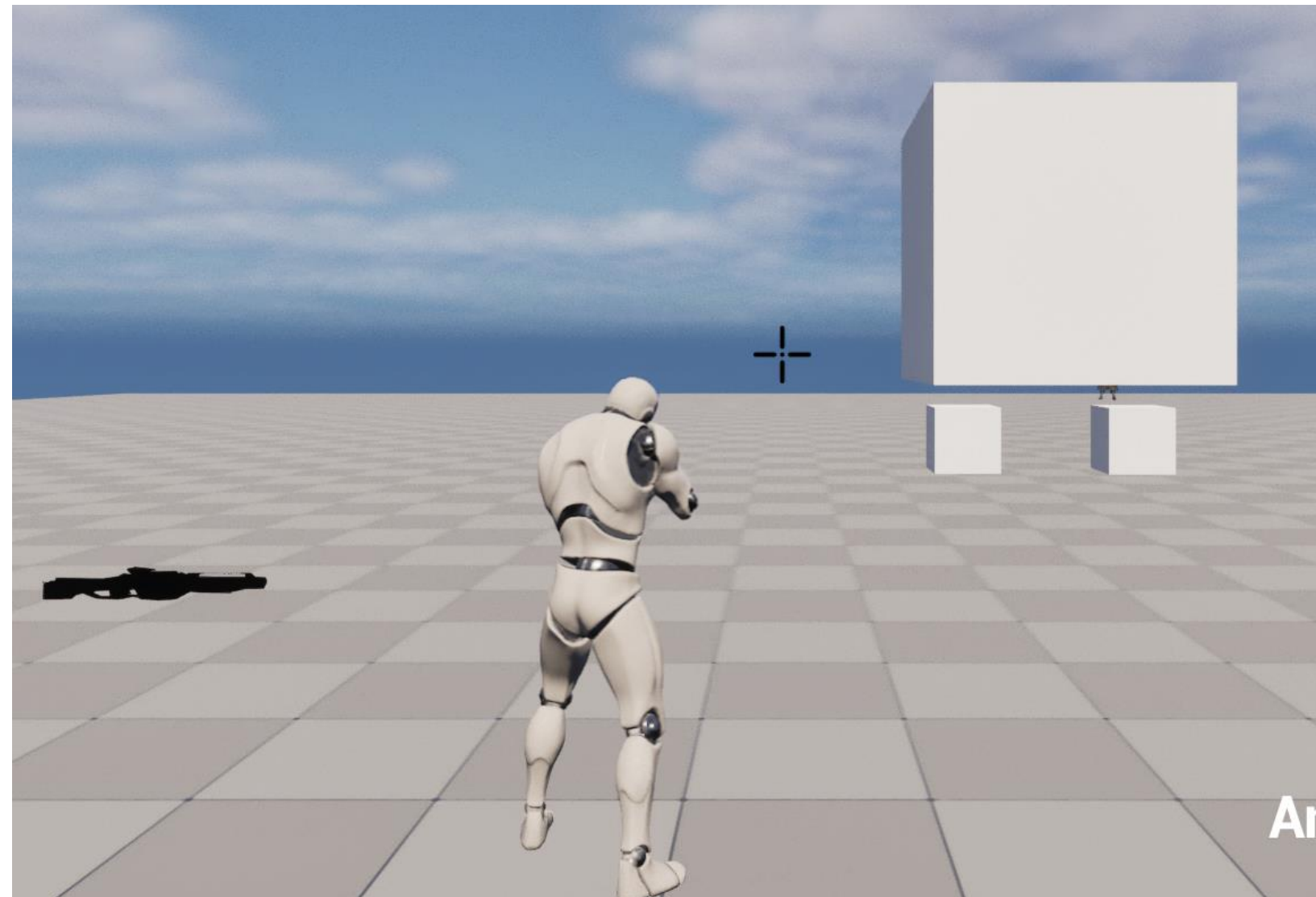
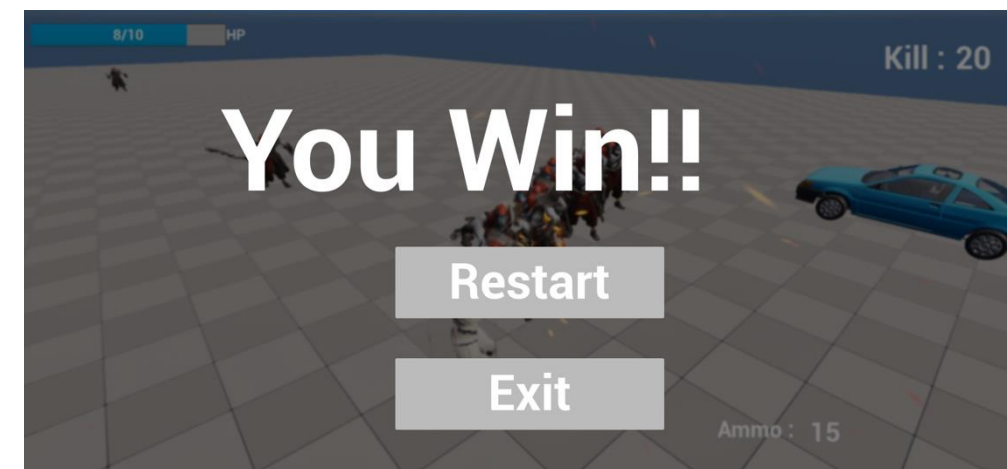


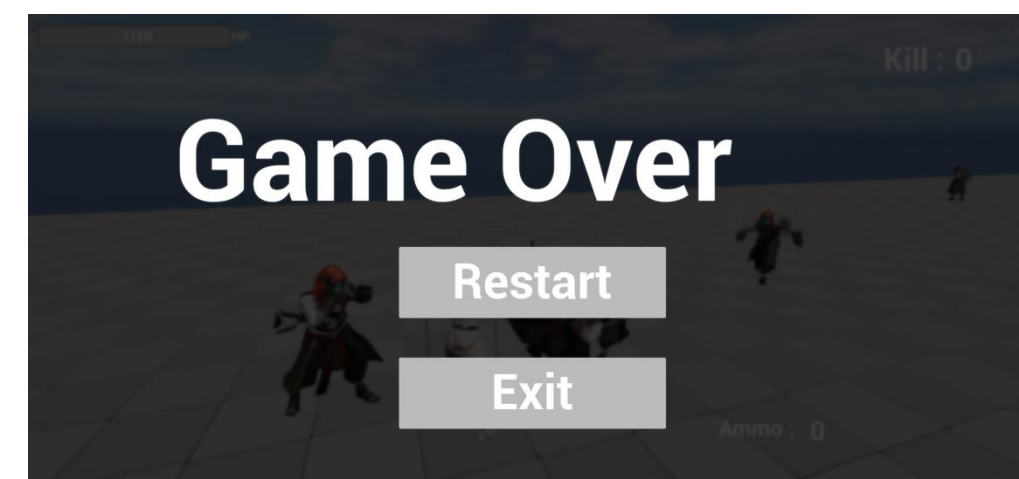
TPS 프로젝트



승리조건 : 적 20킬 하기



패배조건 : 캐릭터 HP가 0이 되는 경우



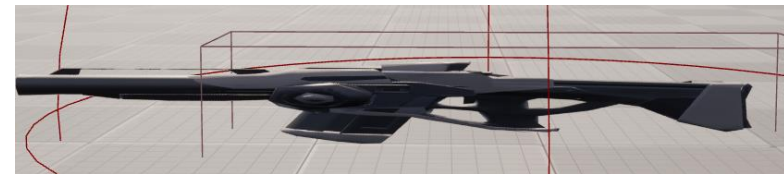
1. 무기

1) Rifle



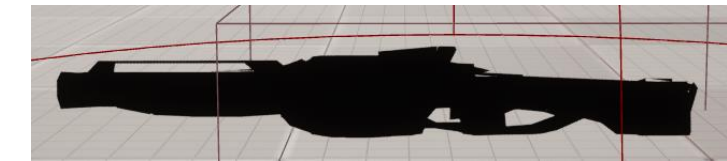
- 단발/연사 기능 구현
- 라인트레이스로 발사 구현

2) Sniper



- 스나이퍼모드 ui 구현
- 라인트레이스로 발사 구현

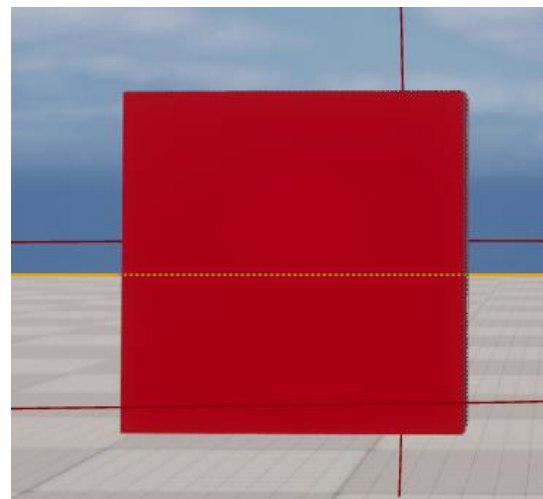
3) Bazooka



- Bullet 클래스를 직접 스폰하여 발사 구현

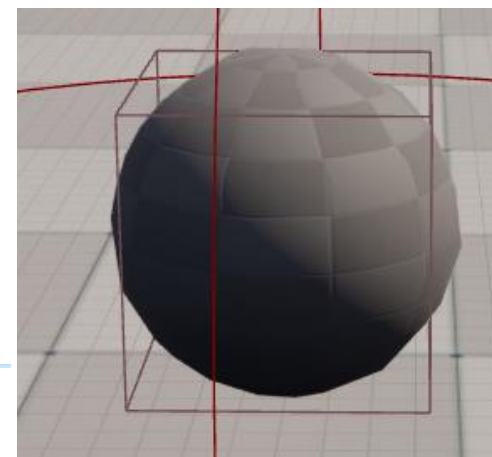
2. 아이템

1) HPPack



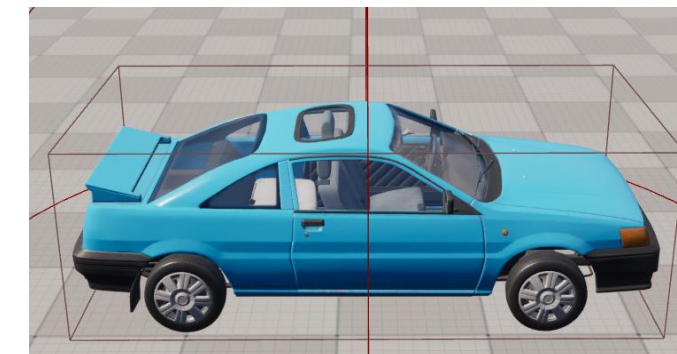
- 플레이어 체력 회복

2) Ammo



- 플레이어가 현재 가지고 있는 총 탄창 충전

3) Car

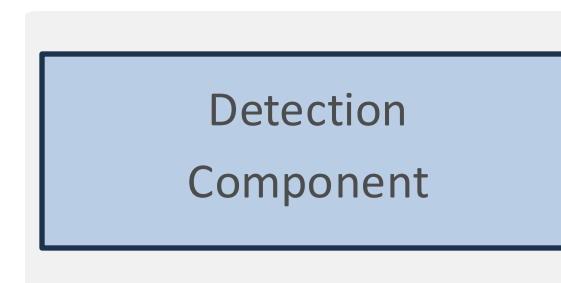
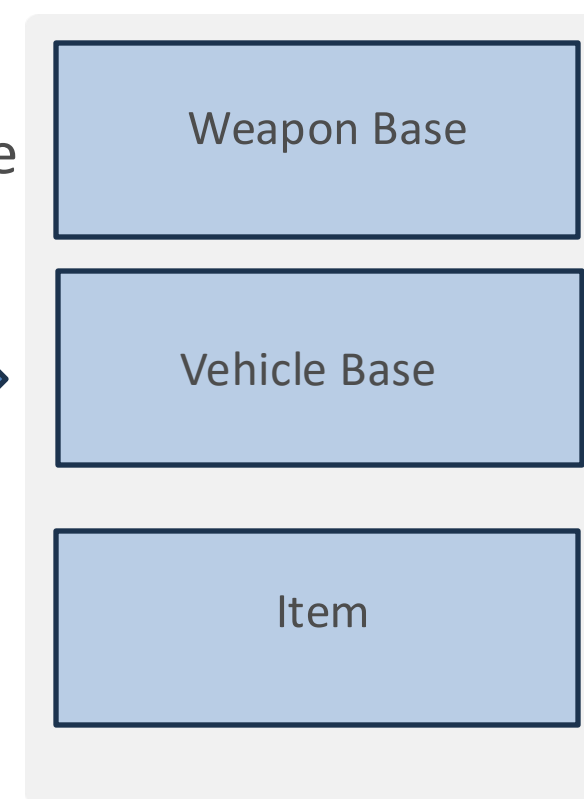
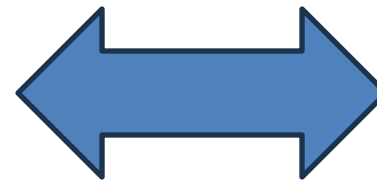


- 플레이어가 차에 빙의되어 조작

전반적인 코드 구조



BaseInteractive
인터페이스



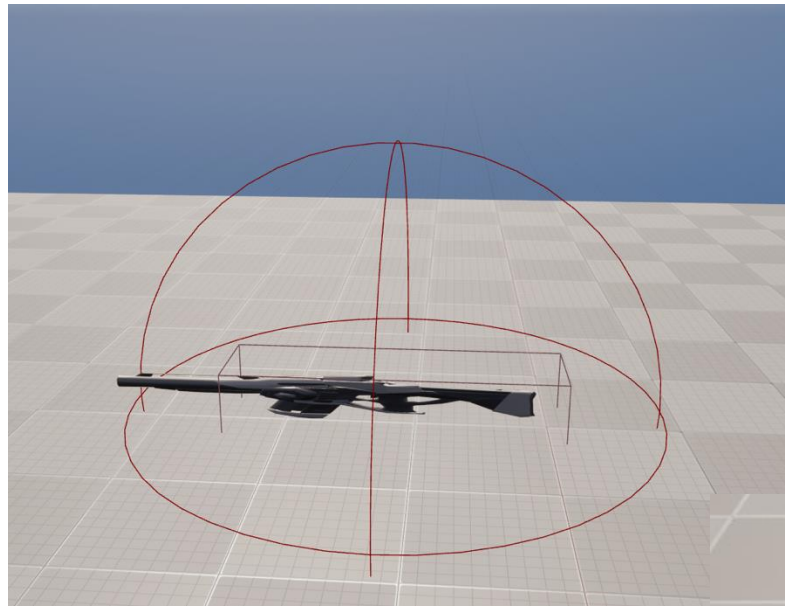
- My Character와 여러 아이템과의 상호작용(줍기, 타기, 장착하기 등)을 구현하려고 했습니다.

- 이 때 다양한 종류의 클래스(Pawn, Actor)와 클래스에 맞는 상호작용을 구현하기 위해 인터페이스를 상속받아 이용했습니다.

- 각 무기들은 WeaponBase를 상속받고, 자동차는 VehicleBase를 상속받았습니다.

- 또한 각 아이템들은 mycharacter에게 감지당했을 때 ui를 띄우기 위해 이 기능을 detection component로 구현하여 각 아이템에 자식컴포넌트로 설정했습니다.

Detection Component



- Detection Component는 플레이어가 감지됐을 때 상호작용 ui를 띄우는 역할을 합니다.

- 부모 클래스에 DetectionSphere(Sphere Component)를 설정한 후 플레이어가 해당 클래스의 DetectionSphere 안에 들어가면(OnOverlapBegin 이벤트 발생) 타이머(매 0.1초마다)를 통해 CheckIfInView()함수를 호출합니다.

- CheckIfInView()함수는 플레이어의 시점이 라인트레이스를 통해 아이템과 부딪쳐서 감지가 검출되면 상호작용 ui를 띄우도록 구현하였습니다.

무기 장착 및 해제

무기 장착

F키를 누르면 무기 장착을 하도록 하였습니다.
장착할때는 attachcomponent 함수를 사용하여 미리 설정한 플레이어의 스켈레톤 소켓에 부착하도록 하였습니다. 따라서 이미 생성된 무기 액터를 그대로 사용하도록 구현하였습니다.

플레이어는 두가지 무기를 장착할 수 있습니다. 그래서 각 무기는 1번 슬롯과 2번 슬롯에 할당하였는데 1번 키를 누르면 1번슬롯에 장착된 무기, 2번 키를 누르면 2번 키에 장착된 무기를 사용하도록 했습니다.

무기 해제

G키를 누르면 무기장착을 해제하도록 하였다.

이 때 플레이어의 앞에 있는 한 점으로부터 아래방향으로 한 벡터를 설정했다. 그리고 라인트레이스를 이용해 그 벡터와 바닥과 충돌한 지점에 무기를 배치하도록 했다.

Enemy & Car

Enemy

EnemyFSM

네비게이션 시스템

- Enemy는 자식컴포넌트로 EnemyFSM을 설정하였습니다. 이 클래스는 상태머신을 이용해서 enemy의 상태를 나타냈습니다. (Idle, Move, Attack, Damage, Dead)
- 또한 네비게이션 시스템을 활용하여 플레이어가 감지되면 자동으로 최적의 길을 찾아 플레이어를 찾도록 구현하였습니다
- Enemy는 일정 범위 안에 플레이어가 있으면 단거리 공격을 하도록 설정했습니다.

Car

Detection Component

Vehicle Movement
Component

- 자동차는 ChaosVehiclesPlugin을 사용하여 구현했습니다.
- 자동차로 빙의하기 위해서는 입력처리를 플레이어에게 했지만 자동차에서 빙의를 풀 때는 컨트롤러가 자동차에 설정되어있기 때문에 자동차 클래스에서 차에서 나오는 입력처리를 했습니다.



감사합니다!