

---

3주차 베타  
분류 3인칭 슈팅  
제목 미정  
작성자 김호곤

---

## ○ 주요 이슈

### 1. 행동트리 만들었다.

적 행동을 만들기 위해 행동트리를 만들었다. 행동트리 노드를 생성할 때는 가독성을 위해 모두 함수로 호출하는 것으로 했고, 외부에서 노드 클래스에 직접 접근할 일은 없기 때문에 모든 노드는 C\_BehaviourTree의 private 내부 클래스로 만들었다. 현재 적의 행동은 순찰 및 이동, 적 발견, 적 공격으로 세 가지밖에 없어서 상태패턴을 쓰는 것도 무방하나, 만약에 앞으로 더 복잡한 행동을 구현하려면 행동트리가 좋을 것이다.

### 2. 적은 NavMesh를 쓴다.

순찰 중 이동하거나 플레이어의 기지로 이동할 때 장애물을 피하기 위해 유니티에서 제공하는 NavMesh를 이용한다. NavMesh는 bake해서 쓰고 Navigation area는 Walkable, Not Walkable 두 가지다.

### 3. 플레이어 이동은 Rigidbody를 쓴다.

물체의 위치 값을 직접 바꾸면 Rigidbody의 물리 충돌이 적용되지 않는다. 따라서 물리적 충돌에 의한 속력 변화까지 적용하기 위해 모든 이동은 Rigidbody의 속력 값을 바꾸는 것으로 했다. HUD에서 속력 표시 또한 Rigidbody의 속력을 읽어오는 것으로 했다.

### 4. Actor 소환 시 정점의 y 좌표를 기준으로 나타난다.

Actor 소환, 소멸을 시각화하기 위해 Actor가 서서히 나타나거나 사라지는 효과를 줬다. 이 효과는 Dissolve를 주는 것과 비슷한데, 원래 Dissolve는 노이즈 텍스처를 이용했다면 여기서는 정점의 y 좌표를 사용해서 위에서부터 아래로 서서히 나타나는 효과를 줘서 마치 텔레포트하는 것같은 느낌을 줬다. 소멸할 때는 역순으로 아래에서 위로 서서히 사라진다.

### 5. C\_ObjectPool을 이용해서 적 소환한다.

적, 폭발 효과 등 소모되는 물체의 반복적인 생성, 파괴를 절약하기 위해 ObjectPool을 이용했다. 처음에 ObjectPool은 아무것도 가지고 있지 않다가 누군가가 어떤 물체를 요청할 때 그 물체를 생성해서 반환해준다. 그 물체를 다 써서 다시 돌려줄 때 ObjectPool은 그 물체를 스택에 담아두고 있다가 다음에 누군가가 또 그 물체를 요청하면 저장하고 있던 물체를 반환한다. 만약 저장하고 있던 물체가 다 떨어지면 새로 물체를 생성해서 반환한다. 물체를 담는 자료구조로 스택을 쓰고 있는데, 큐를 써도 차이는 없을 것 같고, 추가, 삭제가 많이 일어난다는 점에서 연결리스트 사용도 고려해볼 수 있으나 담아둘 물체

수가 많지 않으면 스택으로 유지하는 것이 더 나을 것이다.

#### 6. 일정 범위 안에서 플레이어는 상호작용할 수 있다.

ScriptableObject로 에디터 애셋으로 설정한 값에 따라 일정 거리 안으로 상호작용할 수 있는 물체가 들어오면 플레이어는 상호작용할 수 있다. 현재 상호작용은 적 기지의 어떤 장치를 조작해서 기지를 점령하는 것으로, 상호작용 버튼을 길게 누르고 있어야 된다. PC에서는 키보드로도 상호작용할 수 있고 PC가 아닌 플랫폼을 위해서 UI에 버튼을 만들어 키보드를 쓰는 것과 똑같이 동작하도록 했다.

#### 7. 가이드 미사일의 초기 속력은 비행기 속력과 일치한다.

가이드 미사일에 비행기의 관성을 적용하기 위해 처음 속력은 비행기의 속력과 일치시켰다.

### ○ 부족한 점

#### 1. NavMeshAgent에서 목적지 갱신이 느리다.

NavMeshAgent에서 목적지를 갱신할 때 불규칙적으로 갱신 속도가 다르다. 어떤 때는 시야 내에 적을 발견해놓고 그냥 지나치다가 나중에 가서 멈춰서는 현상이 생긴다.