

# IN-GAME 강화학습 전투 시스템

# 목차

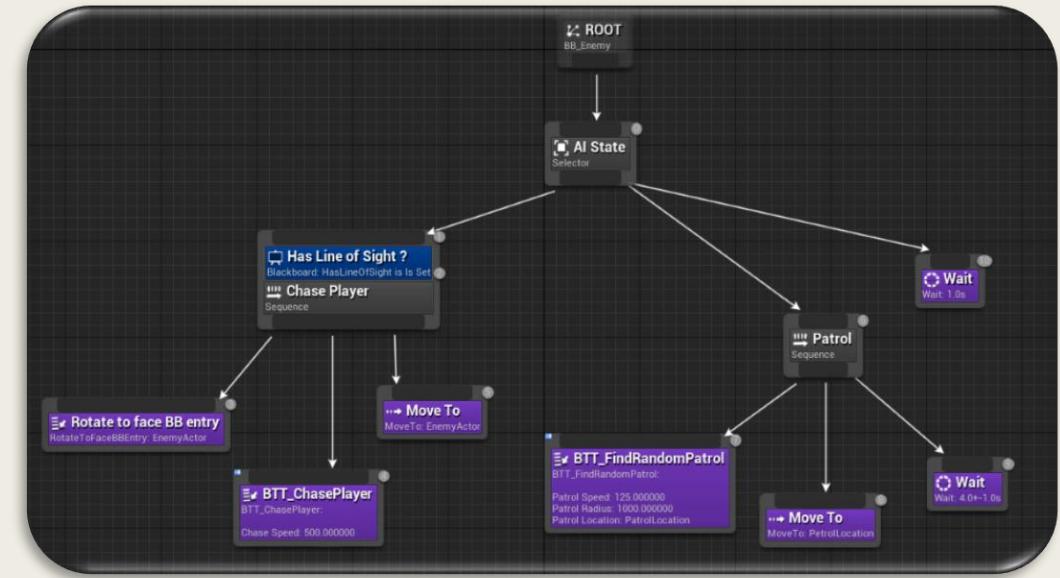
01. 프로젝트 개요 및 목적
02. 기대 효과 및 차별점
03. 요구 명세
04. 설계 및 구현
05. 실행 결과
06. 개선 사항
07. 참고 문헌

# 01. 프로젝트 개요 및 목적 - 개요

- In-Game 강화학습 전투 시스템은
- Unreal Engine 5의 Learning Agents 플러그인을 기반으로
- 보상의 가중치와 학습 횟수를 조절하여 캐릭터의 전투 AI를 훈련시키고, 학습된 AI를 전투에 적용하는 방식으로 강화학습을 게임 플레이 경험의 일부로 통합하여
- 플레이어가 AI 학습 과정에 직접 참여하고 자신만의 AI를 성장시키는 경험을 제공하고자 한다.

# 01. 프로젝트 개요 및 목적 - 동기

- 이전의 게임에서 사용하는 캐릭터의 AI는 유한상태기계 (FSM)나 행동 트리 (Behavior Tree)처럼 사람이 직접 규칙과 조건을 설계하는 규칙 기반 방식이었다.
- 이후, 현재 주목받는 AI 기술은 학습 기반으로 발전하여 사람이 모든 규칙을 직접 설계하지 않아도 AI가 데이터를 통해 스스로 규칙을 학습하고 성능을 개선할 수 있게 되었다.



# 01. 프로젝트 개요 및 목적 - 동기

- 이러한 AI 기술의 발전 흐름에 따라, 최근 게임 산업에서도 학습 기반 AI를 적용하려는 시도가 이루어지고 있다. 그 중 강화학습이 실제로 다양한 상용 게임에 적용되고 있다.
- AI가 변화하는 게임 환경에 자동 적응하여 AI 유지 · 보수를 효율화, 학습된 에이전트 행동 분석을 통해 게임 밸런스 검증을 한다.



MMORPG 블레이드 앤 소울에서 강화학습 기반 전투 AI를 NPC에 적용한 무한의 탑 콘텐츠



강화학습 기반으로 학습된 AI 에이전트를 게임 봇에 적용한 스포츠 게임 Roller Champions

# 01. 프로젝트 개요 및 목적 - 동기

- 대부분의 강화학습 적용 사례는 게임 회사가 중앙에서 AI를 사전에 학습시키고, 그 결과를 게임에 적용하는 방식으로 이루어진다.
- 이러한 구조에서는 플레이어가 학습 과정에 직접 참여할 수 없기 때문에, AI의 학습 특성을 활용하거나 자신의 플레이 스타일을 반영하기 어렵다.
- 본 프로젝트는 플레이어가 직접 학습 과정에 참여할 수 있는 구조를 도입하여, AI의 학습 특성을 이용하고 AI가 개인 맞춤형 플레이 경험을 제공할 수 있도록 하고자 한다.

# 01. 프로젝트 개요 및 목적 - 동기

- 플레이어의 학습 참여를 구현하기 위해 플레이어가 캐릭터에게 명령을 내리고 캐릭터가 해당 목표를 달성하기 위해 세부 행동을 스스로 결정하는 방식의 게임 구조를 선택하였다. 이는 기존에 규칙 기반 AI가 주로 사용되는 장르이기 때문에 플레이어에게 익숙하면서도, 학습 기반 AI의 장점을 자연스럽게 드러낼 수 있다고 생각한다.
- 그리고 학습 참여는 공격 중심인지, 방어 중심인지와 같이 보상의 가중치 조절로 큰 방향을 조절하는 방식을 선택하였다

## 02. 기대 효과 및 차별점

유저 관점의 변화	규칙 기반 AI	In-Game 강화학습
캐릭터 행동 다양성	캐릭터가 정해진 규칙으로 반응하여 낮다	각 캐릭터가 마다 학습 경험에 따라 차별화된 행동 습득 하므로 높다
결과 예측성	미리 정의된 규칙으로 행동하므로 행동 패턴이 쉽게 예측됨	AI가 학습을 통해 스스로 전략을 조정 하므로 겉보기에는 동일한 상황처럼 보여도 다른 행동 가능
플레이 경험의 개별화	모든 플레이어에게 동일한 AI 경험 (획일적)	플레이어의 명령과 피드백이 학습에 반영되어 개인화된 플레이 경험 제공

- 회사는 기존의 중앙적인 학습 방식으로 개발 · 유지보수 효율성 향상과 게임 밸런스 개선 같은 장점을 확보할 수 있다. 이에 더해 In-Game 강화학습을 도입하면 개별화된 플레이 경험을 제공하여 유저들의 유지율을 높일 수 있다.

# 02. 기대 효과 및 차별점

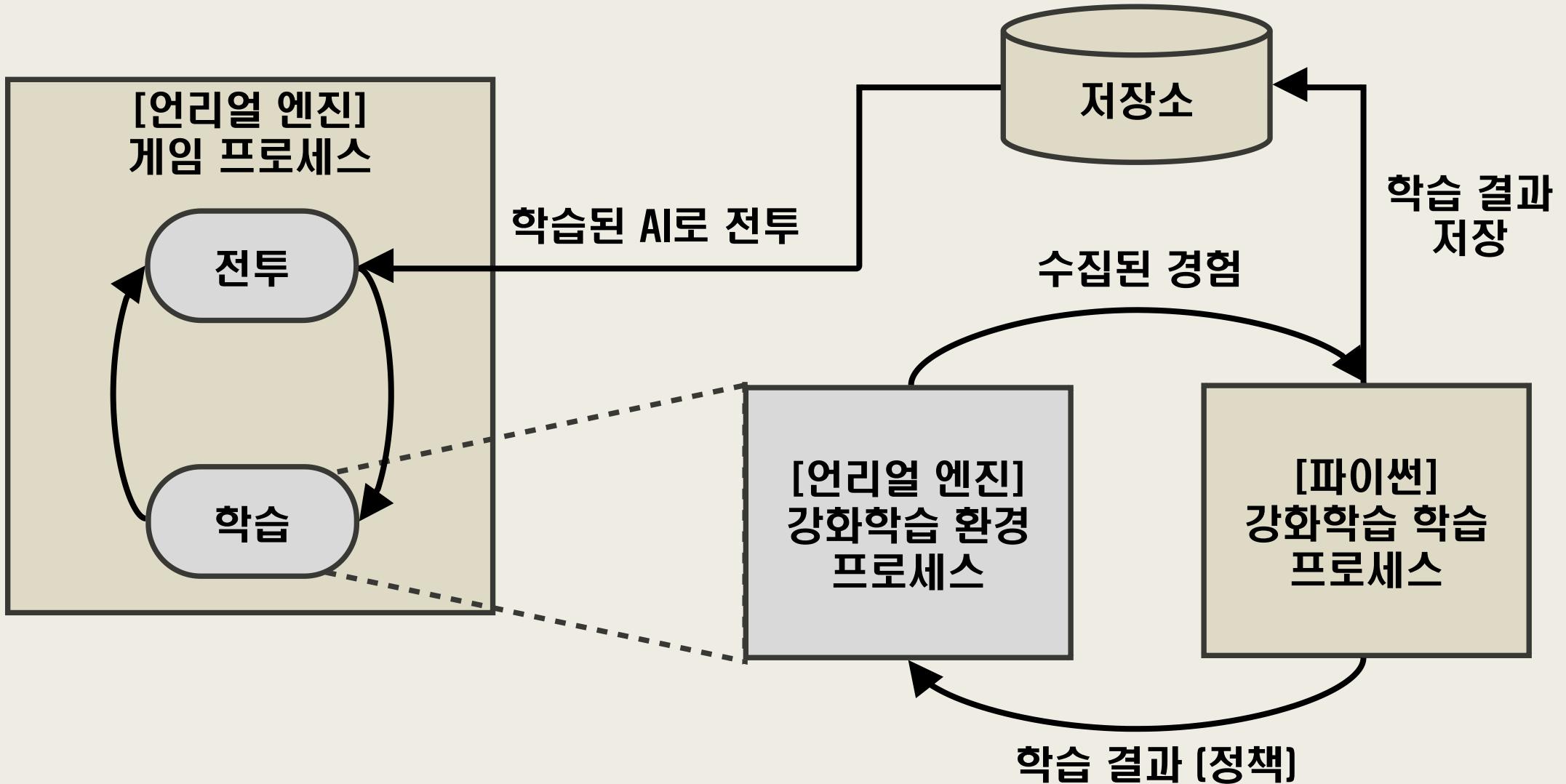
- 플레이어의 AI 학습 참여 시스템이 소수의 캐릭터에 집중하는 육성 게임처럼 캐릭터의 세부 행동 개선에 쓰인다면

구분	기존 캐릭터 육성 게임 (규칙 기반)	In-Game 강화학습 (학습 기반)
조정 방식	능력치나 행동 확률의 수치 조정 (예: 오른팔 공격 확률 +10%)	특정 행동에 대한 보상 가중치를 조정하여 행동을 유도
결과 반영 시점	조정 즉시 반영되어 결과가 바로 나타남	학습 과정을 거치며 점진적으로 변화
변화 예측 가능성	설정된 수치에 따라 일정하게 동작하므로 결과 예측 가능성이 높다	축적된 학습, 상태 변화 등 다양한 요인에 따라 학습이 진행되어 변화의 방향은 예상할 수 있어도, 구체적 결과는 예측하기 어렵다
행동 확률 구조	고정 확률 혹은 조건부 확률 — 사람이 정의한 상태(예: HP<30%)에 따라 미리 지정된 확률만 변동 가능	동적 확률 — AI가 상태(state)를 스스로 해석하고, 상황에 따라 행동 확률을 실시간으로 조정 (예: 55~80%)
플레이 감각 / 몰입 포인트	효율적 수치 조정 중심	학습 관찰 중심 - 점진적으로 변화하는 생동감 있는 성장 경험

# 03. 요구 명세

- User Requirements
  - 사용자는 AI를 직접 학습시킬 수 있는 훈련 모드에 진입할 수 있어야 한다.
  - 사용자는 학습된 AI를 전투 모드에서 전투 캐릭터로 적용할 수 있어야 한다.
  - 보상 함수의 구성 요소(공격, 방어 등)에 대한 가중치를 사용자가 조정할 수 있어야 한다.
  - 학습 과정은 시각적으로 확인 가능해야 하며, 필요 시 중단 및 리셋이 가능해야 한다.
- System Requirements
  - 학습 환경은 병렬 학습을 지원해야 하며, 학습 시간은 현실적이어야 한다.
  - 학습 데이터의 저장 및 불러오기, 데이터 딕렉토리 관리를 지원해야 한다.
  - 에디터 내 UI 위젯을 통해 학습 시작/중지/초기화/스냅샷 선택 기능이 제공되어야 한다.

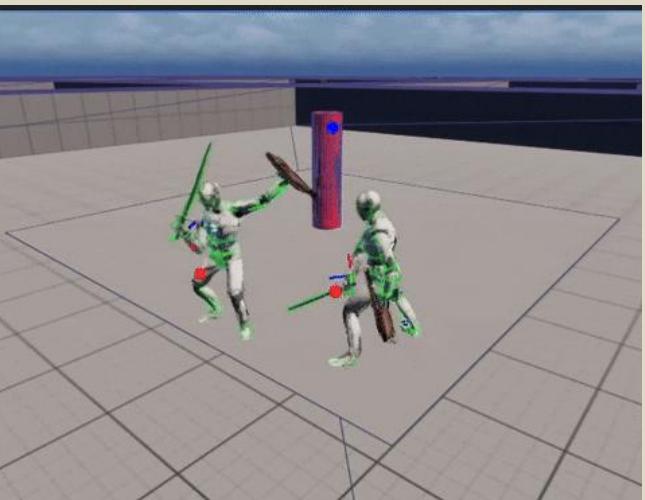
# 04. 설계 및 구현 - 설계



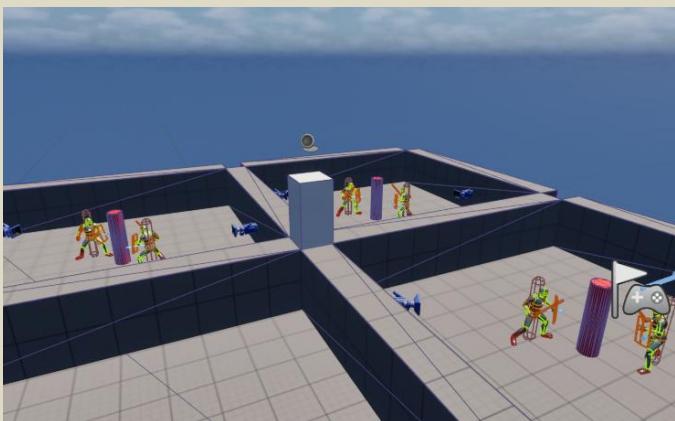
# 04. 설계 및 구현 - 구현

## 강화학습 환경 프로세스

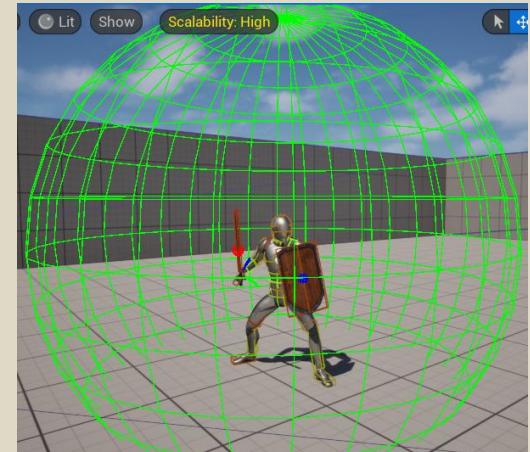
학습 시간이 길기 때문에, 초기에는 미리 학습된 기본 AI를 제공하고 이후에 플레이어가 추가 학습에 참여하도록 설계하였다.



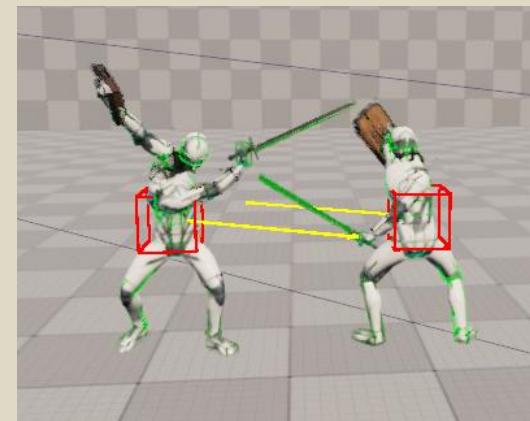
물리 엔진



Multi-Worker 병렬 학습



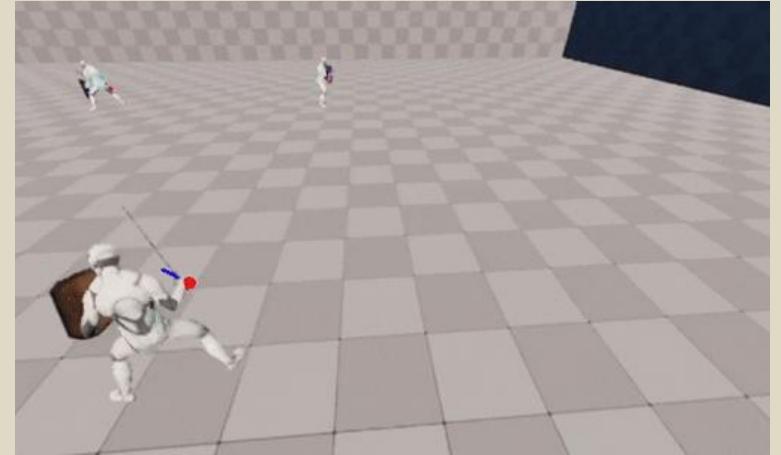
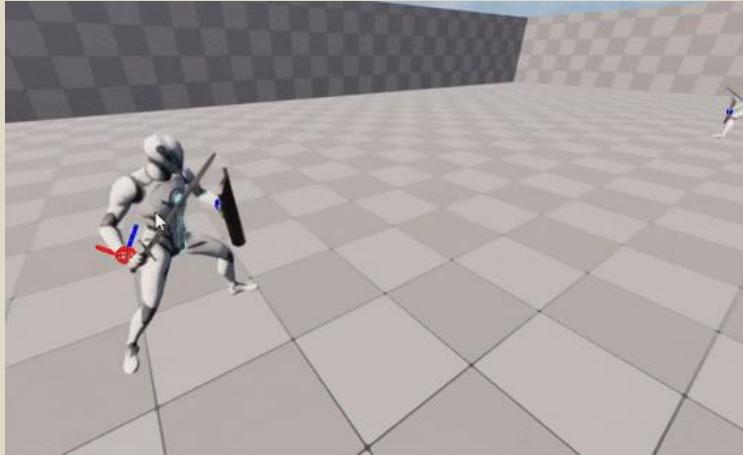
검술 기반



학습 개선, curriculum 학습 적용

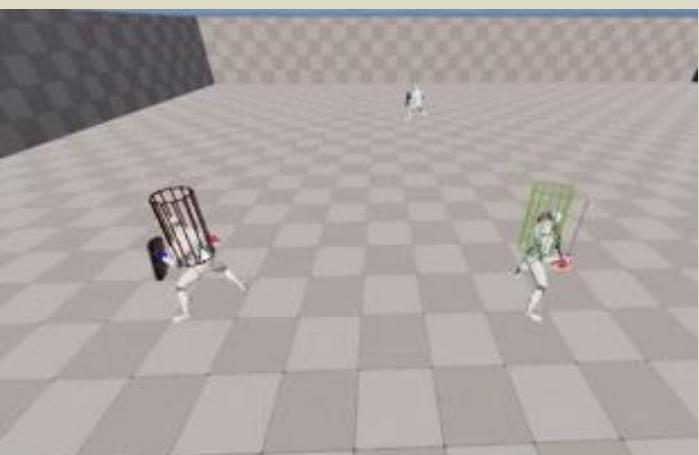
# 04. 설계 및 구현 - 구현

전투



캐릭터 배치

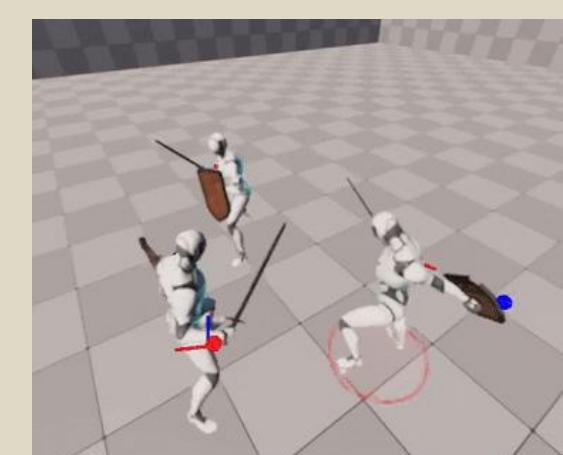
캐릭터 이동



전투 상태 전환

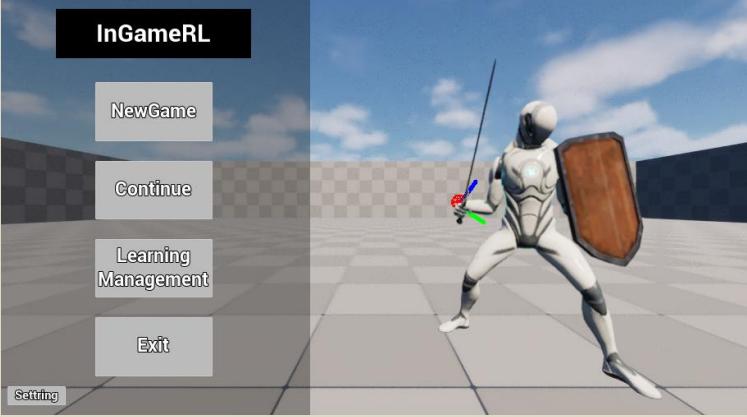


타겟팅 전환

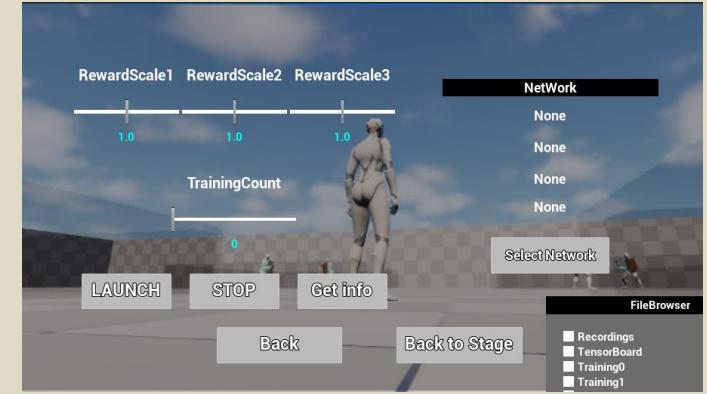


캐릭터 죽음

# 04. 설계 및 구현 - 구현



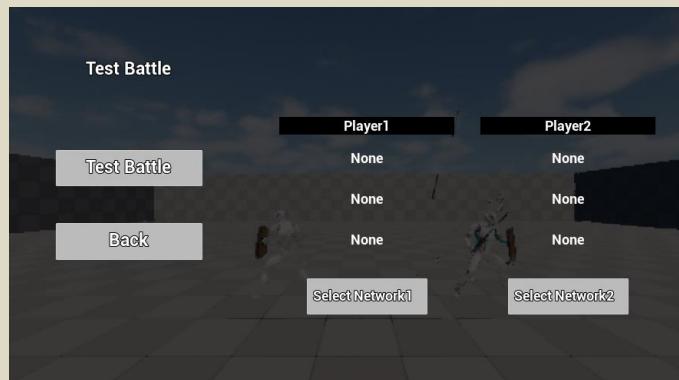
시작 화면



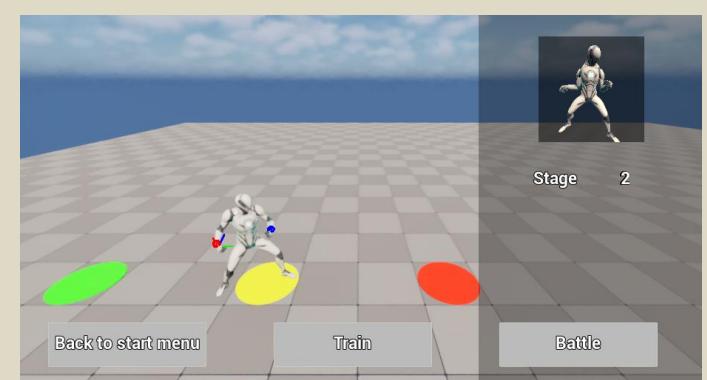
훈련 화면, tcp로 학습 프로세스 관리



전투 전 화면, 네트워크 선택

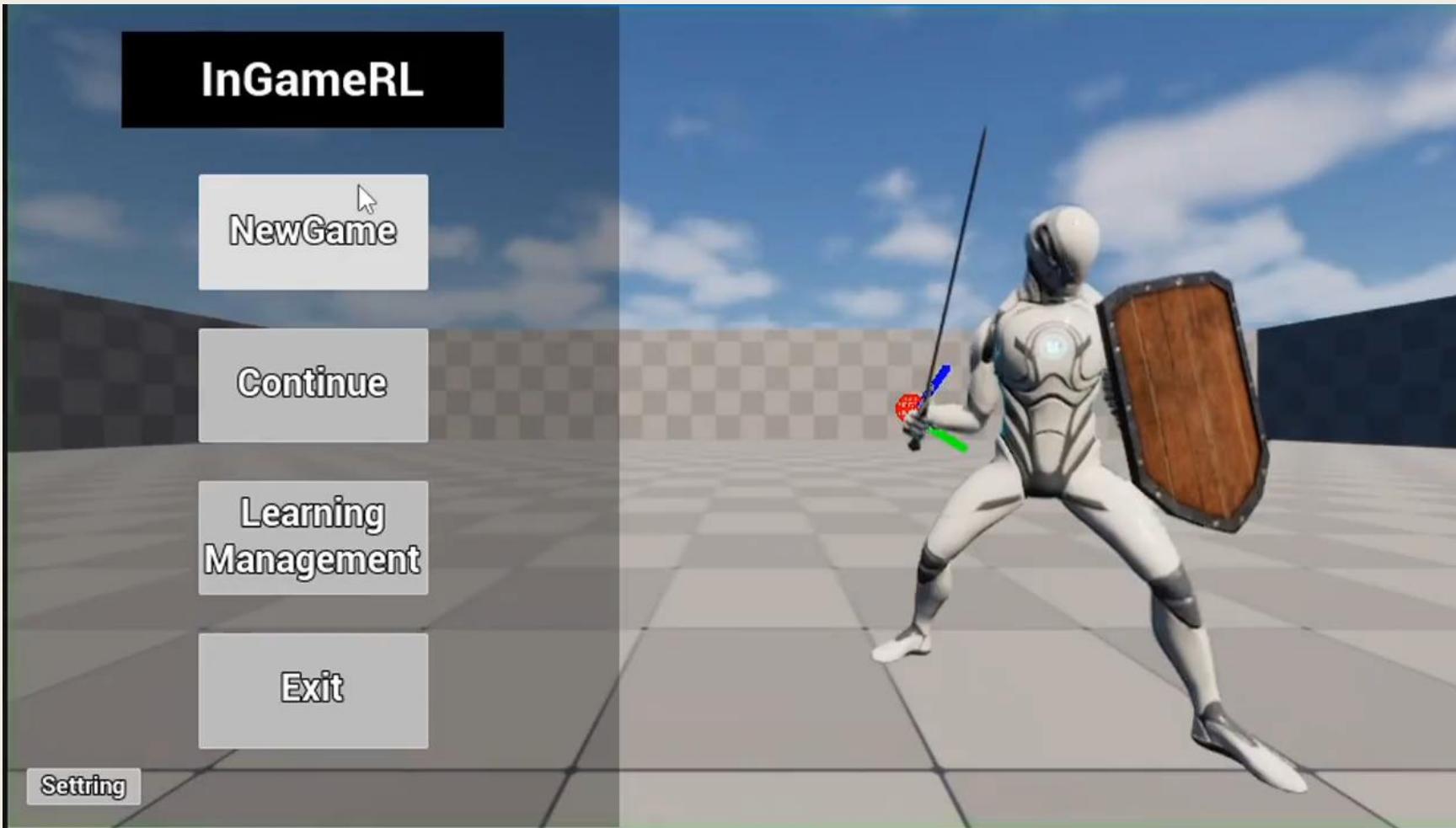


테스트 전투 화면

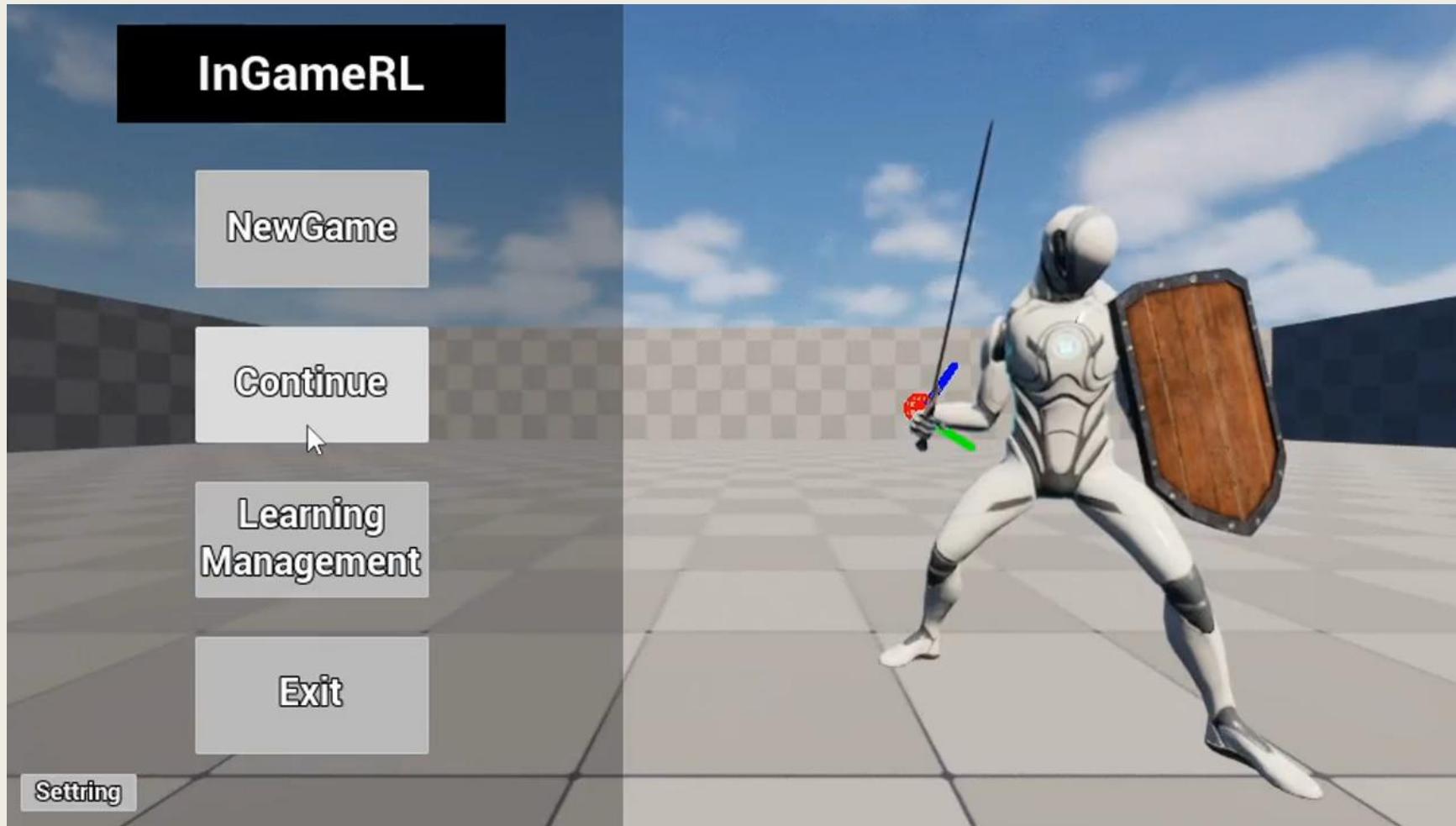


스테이지 화면

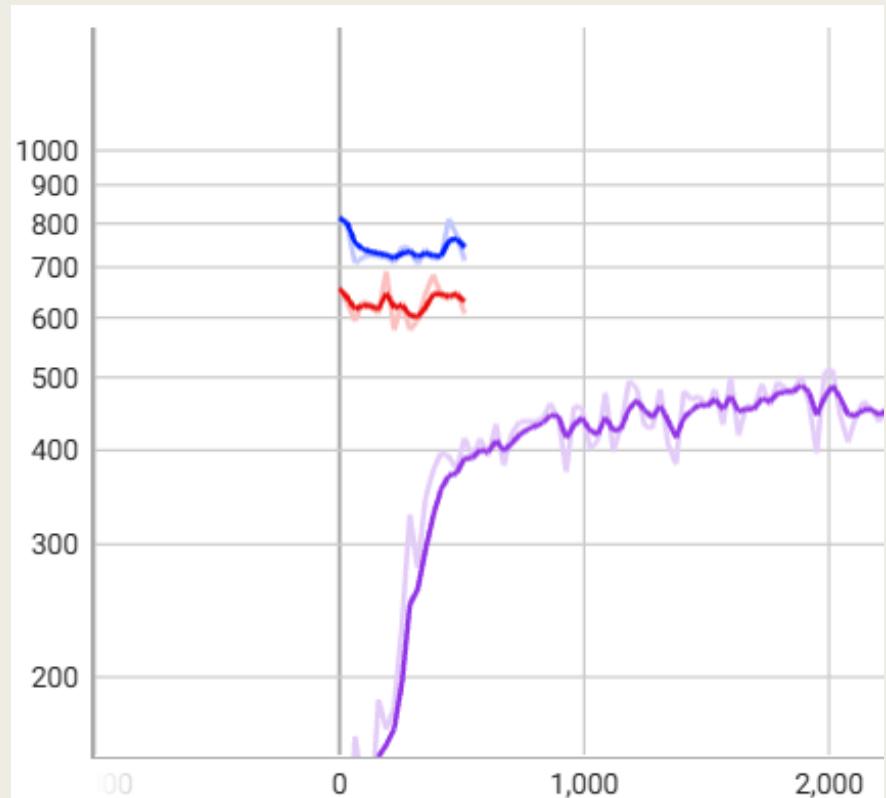
# 05. 실행 결과 - 전투



# 05. 실행 결과 - 학습



# 05. 실행 결과 - 비교



experience/avg\_reward\_sum

보라 - 기본

DistanceReward\_2 + GaussianDirectionReward +  
DistancePenalty + OutofArenaPenalty + StuckPenalty;

빨강(공격적), 파랑(방어적)

OutReward = DistanceReward\_2 +  
GaussianDirectionReward +  
DistancePenalty +  
OutofArenaPenalty +  
StuckPenalty +  
EnemyDeadReward +  
MyDeadPenalty +  
HitReward +  
EnemyHealthReward \* RewardCharacter->GetRewardScale1() +  
MyHealthReward \* RewardCharacter->GetRewardScale2() +  
StaminaReward \* RewardCharacter->GetRewardScale3();

# 06. 개선 사항

- 지금 상용화하려면 훈련 환경을 단순화하고, 게임의 크기, 학습의 속도와 공평성을 고려하여 개인 PC가 아닌 GPU 서버에 학습이 이루어지도록 해야한다.
- 언리얼 엔진 측의 기본 설정에 따라 학습 결과가 매번 새로운 폴더에 자동 저장되는 구조로 되어 있어, 학습이 반복될수록 결과 파일이 분산되고 관리 효율이 떨어진다. 학습 결과를 통합 관리하거나 사용자 정의 저장 경로를 지정할 수 있도록 시스템을 개선할 필요가 있다.
- 현재 학습 결과를 텐서보드에서만 확인 가능하여 게임 내 확인 어렵다. 게임 내 시각화 기능을 추가하여 학습 상태 · 성과를 실시간으로 확인 가능하도록 개선해야 한다.
- 현재는 가중치 조절로 큰 방향을 조절하는 방식이지만, 향후에는 보상 계산을 세분화하여 유저가 세부적으로 관여할 수 있도록 할 것이다.

# 07. 참고 문헌

- 블레이든 앤 소울  
[https://biz.chosun.com/site/data/html\\_dir/2016/09/18/2016091801606.html](https://biz.chosun.com/site/data/html_dir/2016/09/18/2016091801606.html)
- Roller Champions <https://arxiv.org/abs/2012.06031>