연구·조사 최종 보고서

# 1. 연구·조사의 목적

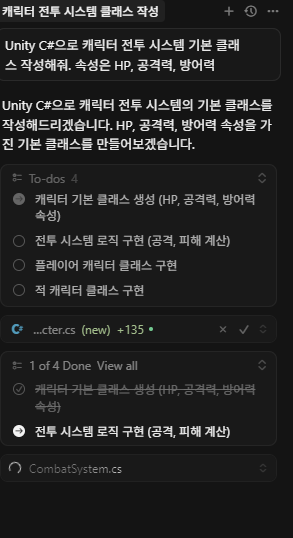
본 연구의 목적은 게임 개발 과정에서 인공지능(AI)을 실제로 활용할 수 있는 가능성을 검증하는 것이다. 앞선 연구에서는 방치형 RPG 장르를 중심으로 콘텐츠 제작(퀘스트, 스토리, 아트 리소스)에 AI를 적용하는 사례와 효과를 조사하였다. 이번 연구에서는 그 연계선상에서, 코딩 보조, 오류 해석, 플레이 로그 분석, QA 자동화 등 개발 프로세스 전반에 AI를 적용하여 실질적인 효용성과 한계를 확인하고자 하였다.

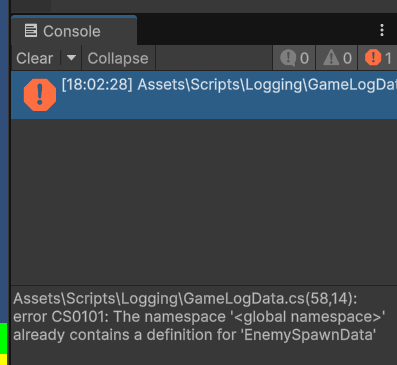
# 2. 연구·조사 과정과 방법

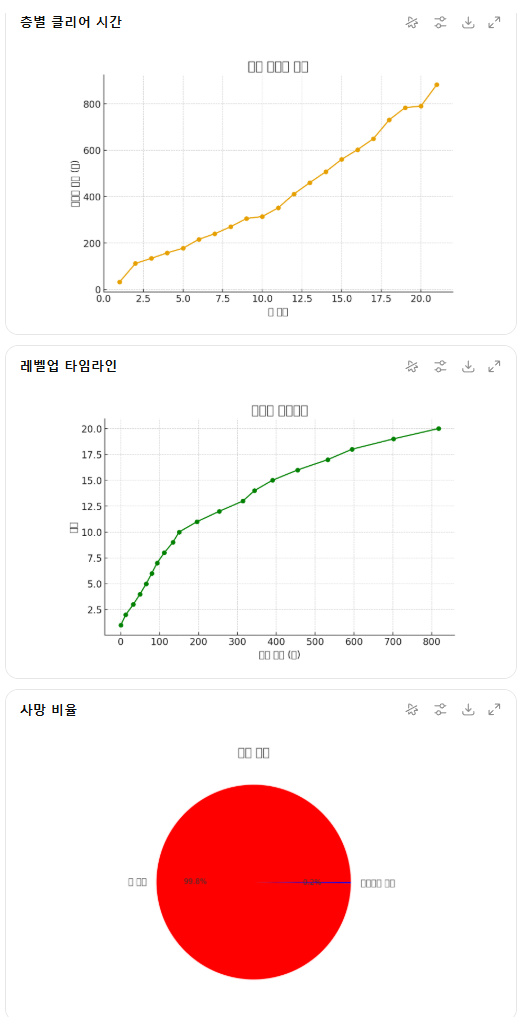
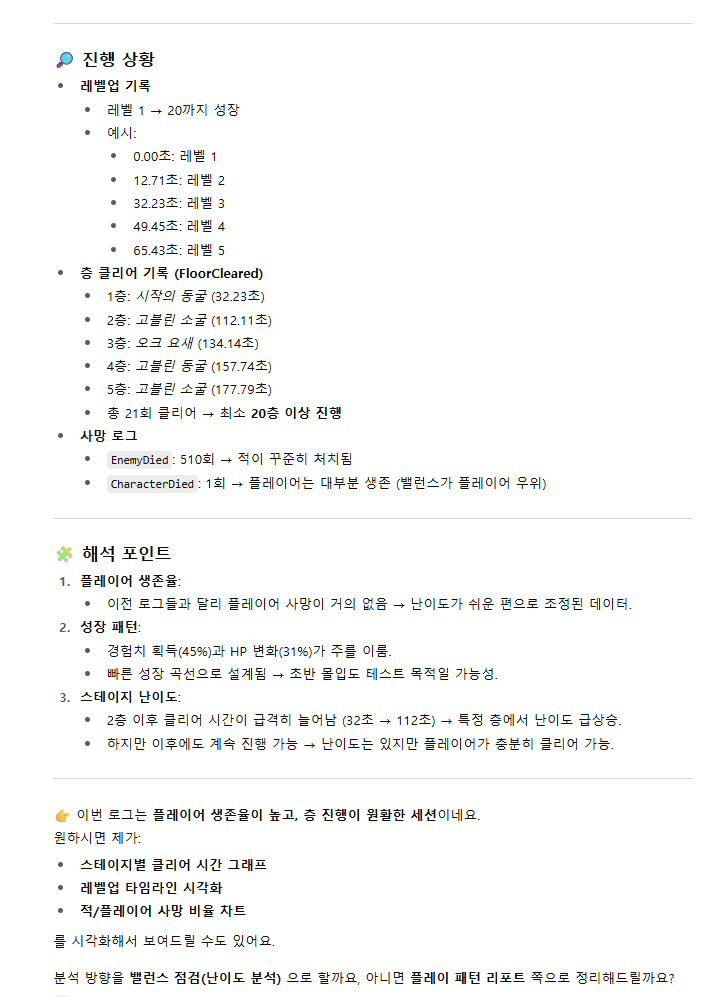
연구는 크게 세 가지 축으로 진행되었다.  
① 코딩 자동화: GitHub Copilot, ChatGPT, Cursor IDE를 활용하여 전투 시스템, 보상, 로그 저장과 같은 반복적 코드 구현을 지원받았다.  
② 오류 해석: Unity 실행 과정에서 발생하는 콘솔 오류를 LLM 모델로 분석하여 원인과 해결 방안을 확인하였다.  
③ 데이터 분석: 플레이 로그(전투 결과, 유저 입력, 세션 데이터)를 기록하고 GPT 기반 모델을 활용해 난이도 곡선과 유저 행동 패턴을 분석하였다.  
④ QA 자동화: Unity ML-Agents를 활용하려 했으나 이번 연구에서는 적용하지 못했으며, 추후 연구 과제로 남겨두었다.

# 3. 연구·조사의 결과

이번 연구에서는 Cursor를 활용하여 방치형 RPG의 핵심 기능들을 구현할 수 있었다. Cursor는 코딩 보조 도구로서, 전투 시스템·보상 처리·유저 개입 요소(스킬/아이템 선택)·로그 기록 등 다양한 기능의 코드를 자동으로 제안하거나, 반복적인 구조를 빠르게 완성하는 데 큰 기여를 하였다. 이를 통해 개발 과정에서 소요되는 시간을 크게 단축할 수 있었고, 코드 품질 역시 일정 수준 이상 유지할 수 있었다.  
  
또한 Unity 실행 과정에서 발생한 오류 메시지를 그대로 LLM에 입력하여 분석하는 방식을 적용하였다. 그 결과, 일반적으로 시간이 소요되는 디버깅 과정을 AI가 빠르게 보조해주었으며, 코드 수정 방향을 명확히 제안해주어 실제 오류 해결로 이어졌다. 이는 AI를 디버깅 어시스턴트로 활용할 수 있는 가능성을 확인한 사례라 할 수 있다.  
  
로그 분석 단계에서는 플레이 로그를 GPT 기반 모델에 전달하여 복잡하고 긴 로그 데이터를 자동으로 요약·분석하였다. 이를 통해 플레이 난이도 곡선과 유저 행동 패턴(예: 공격/방어 스킬 사용 비율, 포션 사용 타이밍)이 도출되었으며, 플레이어의 전략적 선택이 승률에 미치는 영향을 정량적으로 확인할 수 있었다. 이러한 결과는 AI가 데이터 분석 과정에서 패턴 도출과 난이도 평가를 신속히 수행할 수 있음을 보여준다.  
  
ML-Agents의 경우 이번 연구에서는 활용하지 못했으나, 추후 연구에서 QA 자동화 및 난이도 밸런싱 검증을 위한 도구로 연계할 계획이다.

이미지





# 4. 다음 연구·조사 목표

앞으로의 연구는 AI가 게임 개발 전 과정에 어떻게 통합될 수 있는지를 보다 체계적으로 검증하는 방향으로 확장될 것이다. 특히 콘텐츠 자동 생성과 개발 자동화의 경계를 넘어서, ① 대규모 데이터셋을 기반으로 한 장기 밸런스 분석, ② AI가 직접 참여하는 테스트 주도 개발(TDD) 환경 구축, ③ 플레이어 경험(UX) 최적화를 위한 실시간 AI 피드백 시스템 개발 등을 차기 연구 과제로 설정할 예정이다.