JKSCI

한국컴퓨터정보학회논문지
Journal of The Korea Society of Computer and Information
Vol. 31 No. 10, pp. 343-353, October 2026
https://doi.org/10.9708/jksci.2026.31.10.343

A Survey Study on the Utilization and Perceived Efficiency of Generative AI in Small-Scale Game Development Pipelines - Focusing on Stage-by-Stage Utilization and Perceived Effectiveness among Game Jam and Project Developers -

Bu-ho Choi*

*Professor, School of Game, ChungKang College of Cultural Industries, Icheon, Korea

[Abstract]

In this study, we aim to investigate the utilization of generative AI and the perceived efficiency at each pipeline stage in small-scale game development. Little is known about where small-scale developers actually use generative AI and what benefits and limitations they experience. A survey was conducted with 93 game-major student developers who had participated in a game jam or project-based development. Utilization was concentrated in the programming and planning stages, where perceived time efficiency and output quality were also the highest, whereas 3D art showed the highest post-processing burden and the lowest quality satisfaction; perceived time-saving benefits and a high post-processing burden coexisted in this stage. Stage differences were significant, while no significant difference in perceived efficiency was found between the game jam and graduation-project groups. Usage fees and post-processing burden were the main obstacles, and prompt writing and post-processing skills were the most needed support. Through this, we suggest implications for small development teams and game development education.

▸ **Key words**: Generative AI, Small-Scale Game Development, Game Development Pipeline, Perceived Efficiency, Survey Study

[요 약]

본 연구에서는 소규모 게임개발 환경에서 생성형 AI의 활용 실태와 파이프라인 단계별 체감 효율성을 조사하고자 한다. 생성형 AI는 게임개발 전 과정을 지원하는 제작 도구로 확장되었으나, 소규모 개발자가 어느 단계에서 어떤 효용과 한계를 경험하는지에 대한 국내 조사는 부족하다. 이에 게임잼 또는 프로젝트 기반 개발에 참여한 게임 전공 학생 93명을 대상으로 설문조사를 진행하였다. 생성형 AI 활용은 프로그래밍과 기획 단계에 집중되어 있었고 두 단계에서 시간 단축과 품질 인식도 가장 높았던 반면, 아트(3D)는 수정·후처리 부담이 가장 크고 품질 만족도가 가장 낮아 시간 단축 인식과 높은 후처리 부담이 함께 나타났다. 단계 간 차이는 유의하였으나 효용 인식의 게임잼·졸업작품 집단 간 차이는 유의하지 않았으며, 주요 장애 요인은 이용 요금과 후처리 부담, 필요한 지원은 프롬프트 작성법과 후처리 기법이었다. 이를 통해 소규모 개발팀과 게임 개발 교육을 위한 시사점을 제공하고자 한다.

▸ **주제어**: 생성형 AI, 소규모 게임개발, 게임개발 파이프라인, 체감 효율성, 실태조사

• First Author: Bu-ho Choi, Corresponding Author: Bu-ho Choi
*Bu-ho Choi (sk8geenie@ck.ac.kr), School of Game, ChungKang College of Cultural Industries
• Received: 2026. 09. 21, Revised: 2026. 10. 06, Accepted: 2026. 10. 17.

Copyright © 2026 The Korea Society of Computer and Information
http://www.ksci.re.kr pISSN:1598-849X | eISSN:2383-9945
www.kci.go.kr

I. Introduction

과거 게임 개발에서의 인공지능은NPC의 행동 제어나 절차적 콘텐츠 생성과 같이 게임 콘텐츠의 한 부분으로 활용되어 왔다. 그러나 생성형AI의 등장과 급속한 발전으로 인해 인공지능의 역할은 콘텐츠 내부의 기능을 넘어 개발자가 사용하는 제작 도구로 확장되었다. 텍스트 생성 모델은 기획서와 시나리오를 작성하고, 이미지와3D 생성 모델은 시각 에셋을 제작하며, 코드 생성 도구는 기능 구현과 디버깅을 지원하는 등 생성형AI는 게임 개발 전 과정에 관여하고 있으며, 게임 개발 교육 현장에서도 이를 제작 과정에 통합하려는 시도가 진행되고 있다[1].

이러한 변화가 가장 큰 의미를 갖는 대상은 소규모 개발팀이다. 소규모 팀은 한 사람이 기획, 아트, 프로그래밍, 사운드, QA 등 여러 직군의 역할을 겸해야 하므로 비전문 분야의 전문성 부족과 절대적인 작업량이 개발의 병목으로 작용한다. Panchanadikar와 Freeman은 솔로·인디 개발자들이 생성형AI를 부족한 직군을 보완하는 협업자로 기대하는 동시에 잘못된 정보와 결과 검증 부담을 가진 도구로 인식하고 있음을 보고하였다[2]. 이렇듯 생성형AI의 효용은 도구의 존재만으로 결정되는 것이 아니라 실제 개발 파이프라인의 어떤 작업에 사용되는지에 따라 달라질 수 있다.

게임 제작 분야의 선행연구는 2D 그래픽 에셋의 생성과 스타일 일관성 문제[3], LLM을 이용한 게임 대화 생성[4], LLM이 결합된 절차적 콘텐츠 생성[5] 등 개별 단계의 적용 가능성을 보여주고 있다. 프로그래밍 영역에서는 통제 실험에서 코드 생성 도구가 과제 완료 시간을 단축한다는 결과가 보고된 반면[6], 숙련 개발자가 익숙한 코드베이스에서 작업한 실험에서는 오히려 작업 시간이 증가하는 결과가 나타났다[7]. 이러한 상반된 결과는 생성형AI의 효율성이 작업의 성격과 사용 환경에 따라 달라질 수 있음을 보여준다. 그러나 국내의 소규모 개발 환경에서 개발자들이 실제로 어느 단계에서 생성형AI를 활용하며 어떤 효용과 한계를 체감하는지를 비교한 조사 연구는 부족한 실정이다.

이에 본 논문에서는 게임잼과 졸업작품 등 프로젝트 기반 개발에 참여한 게임 전공 학생 개발자를 대상으로 설문조사를 실시하여, 소규모 게임개발 파이프라인에서의 생성형AI 활용 실태와 단계별 체감 효율성을 분석하고자 한다. 구체적으로는 첫째, 생성형AI가 파이프라인의 어느 단계에서 어떤 유형의 도구로 활용되는지, 둘째, 시간 단축·산출물 품질·수정 및 후처리 부담에 대한 인식이 단계별로 어떻게 다른지, 셋째, 생성형AI 활용을 방해하는 요인과 필요한 지원은 무엇인지, 넷째, 게임잼과 장기 프로젝트라는 개발 맥락에 따라 활용과 인식에 차이가 있는지를 확인한다. 이를 통해 소규모 게임개발 환경에서 생성형AI를 효과적으로 도입하기 위한 기초 자료와 게임 개발 교육에의 시사점을 제공하고자 한다.

II. Preliminaries

1. Small-Scale Game Development and Game Jams

게임 개발은 기획, 아트, 프로그래밍, 사운드, QA·테스트 등 서로 다른 전문 영역의 협업을 전제로 발전해 왔다. 반면 소규모 팀은 구성원이 여러 단계의 작업을 겸해야 하므로 빠른 프로토타이핑과 반복 검증이 중요하며, 기존 연구에서도 비전문가가 직접 프로토타입을 구현할 수 있도록 설계 과정을 단순화하는 교육 방법[8]과 적은 비용으로 초기 기획을 검증하는 페이퍼 프로토타이핑 방법[9]이 제안되었다. 생성형AI는 이러한 흐름에서 작업의 일부를 자동화하거나 비전문 분야의 진입장벽을 낮추는 새로운 도구로 볼 수 있다. 보다 넓은 범위에서Gallotta 등은LLM이 게임에서NPC, 플레이어 보조, 디자인 보조 등 다양한 역할을 수행할 수 있음을 정리하였고[10], Ternar 등은 생성형AI가 제작 파이프라인과 역할 분담을 변화시키는 동시에 품질과 통제, 실무 도입의 과제를 발생시킨다고 보고하였다[11]. 이는 개발자가 실제 활용 과정에서 경험하는 효용과 한계를 단계별로 확인할 필요가 있음을 시사한다.

소규모 개발의 조건을 짧은 시간 안에 압축하여 경험하게 하는 대표적 형식이 게임잼(game jam)이다. 참가자는 제한된 시간과 인원으로 플레이 가능한 결과물을 완성해야 하므로 에셋 확보와 기능 구현 속도가 직접적인 제약이 되며, 단일 개발자가 여러 생성형AI 도구를 결합하여 며칠만에 게임을 완성한 사례가 보고되었고[12], 게임잼이 AI를 소재로 프로토타이핑과 아이디어 탐색을 수행하기에 적합한 형식임을 보여준 연구[13]도 있다. 반면 졸업작품과 학기 프로젝트는 긴 기간 동안 완성도와 일관성을 높여야 하므로, 생성형AI가 단기 개발과 장기 개발 각각에서 유효한지를 함께 확인할 필요가 있다.

2. Pipeline Stages, Tool Types, and Research Gap

본 논문에서는 게임 개발 파이프라인을 기획, 아트(2D), 아트(3D), 프로그래밍, 사운드, QA·테스트의6단계로 구분

www.kci.go.kr

하였다. 아트를2D와3D로 분리한 것은 두 영역의 산출물 구조와 후처리 방식이 다르고, 생성형AI 도구의 성숙도 역시2D 이미지 생성과 최근 상용화된3D 모델 생성 간에 차이가 크기 때문이다. 생성형AI 도구는 텍스트·기획, 2D 이미지, 3D 에셋, 코드, 오디오 생성의5개 유형으로 분류하였으며, 하나의 도구가 복수의 단계에서 사용될 수 있으므로 특정 제품의 만족도를 비교하지 않고 도구 유형과 개발 단계 수준에서 활용과 인식을 측정하였다.

관련 연구를 영역별로 정리하면Table 1과 같다. 기존 연구는 생성형AI의 기술적 가능성이나 특정 산출물 제작을 다루는 데 집중되어 있으며, 실제 개발자가 여러 단계를 경험하면서 체감하는 효율성과 수정 부담을 동일한 척도로 비교한 자료는 제한적이다. 최근 게임개발 실무자를 대상으로 한 절차적 생성 도구 조사에서도 직군별 활용 격차가 확인되었고 생성형 방식에 대해서는 자동화 자체보다 창작 통제와 과정의 투명성이 우선되는 경향이 보고되었으며[14], 이는 콘텐츠의 표현 방식과 품질 평가 기준을 함께 고려하였던 절차적 콘텐츠 생성 연구의 관점과도 맞닿아 있다[15]. 특히 소규모 팀에서는 최종 산출물뿐 아니라 임시 에셋, 레퍼런스, 프로토타입과 같은 중간 산출물도 개발 속도에 영향을 미치므로 최종 품질만으로 효용을 판단하기 어렵다. 본 논문에서는 단계별 활용 빈도와 시간 단축, 품질, 후처리 부담을 동일 척도로 함께 측정하고 장애 요인과 지원 수요를 결합하여, 소규모 게임개발 파이프라인에서 생성형AI의 효용이 집중되는 지점과 한계가 나타나는 지점을 확인하고자 하였다는 점에서 기존 연구와 차이가 있다.

Table 1. Summary of related studies by area

Area	Studies	Key findings
Developer / production studies	[1][2][11][14]	Learner, indie-developer, and practitioner studies; workflow change, adoption barriers, creative control
2D graphics	[3]	Efficiency potential with style consistency and copyright constraints
Narrative / PCG	[4][5][10][15]	LLM dialogue and game roles; LLM-integrated PCG and established PCG taxonomy
Programming	[6][7]	Productivity gains in a controlled task vs. slowdown in mature codebases
Game jam / prototyping	[8][9][12][13]	Rapid validation methods, GenAI-assisted game-jam production, AI learning through game jamming

III. Research Method

1. Research Design and Participants

본 연구는 게임 전공 학생의 소규모 게임개발 환경에서 생성형AI의 활용 실태와 파이프라인 단계별 효용성 인식을 파악하기 위한 횡단적 조사연구이다. 분석 틀은6개 개발 단계에 대한 활용 빈도, 체감 시간 단축, 산출물 품질, 수정·후처리 부담으로 구성하였으며, 동일 응답자가 각 단계에 대해 응답하는 반복측정 구조를 채택하여 단계 간 인식 차이를 응답자 내에서 비교하였다. 또한 전체 응답자를 대상으로 장애 요인, 향후 활용 태도, 필요한 교육·지원을 조사하였고, 개발 맥락이 상이한 게임잼 참여자와 장기 프로젝트 참여자를 함께 조사하여 맥락에 따른 차이를 비교하였다. 자기보고 측정의 한계를 보완하기 위해 최근 완성한 산출물의 실제 소요시간을 묻는 준정량 문항과 개방형 문항을 병행하였다.

조사 대상은 게임 전공 대학생으로, 4인 이하 팀 단위의 비교과 게임잼 참가자와 20인 미만 팀으로 진행되는 프로젝트(졸업작품 및 학기·개인 프로젝트) 참여자를 포함한다. 게임잼과 소규모 팀 프로젝트는 제한된 인력으로 구성원 각자가 여러 직군의 역할을 겸하며 플레이 가능한 결과물을 완성해야 한다는 점에서 소규모 게임개발의 조건을 압축적으로 재현한다. 특히 졸업작품은 수 개월에 걸쳐 전 파이프라인을 수행하고 완성된 빌드를 외부에 공개하므로 소규모 인디 프로젝트와 성격이 크게 다르지 않다. 실제로 조사 대상 기관의 학생 작품은 2025년 기준(기관 집계) 부산인디커넥트 페스ㄴ티벌(BIC2025)에 13편이 선정·전시되어 그중 1편이 수상하였고, GEEKS 2025에서 대상을 수상하였으며, 지스타 인디 쇼케이스와 타이베이 게임쇼 2026에 전시되고 1편이 Steam 출시 예정 차트1위를 기록하는 등 외부 평가와 상용화 성과를 거두고 있다. 게임잼 참가 학생들 또한 글로벌 게임잼에서 수상하는 등 외부 평가를 받은 바 있다. 이러한 사례는 학생 프로젝트가 소규모 인디 개발과 유사한 완성도와 공개 조건에서 수행될 수 있음을 보여주는 것으로, 개발 환경의 유사성을 설명하기 위한 것이며 표본의 대표성을 입증하는 근거는 아니다. 단일 기관의 학생 표본이라는 점에서 현업 개발자 전반으로의 일반화에는 제약이 있으므로 이를 연구의 한계로 명시한다.

자료 수집은 무기명 온라인 설문으로 2026년7월29일부터 9월11일까지 진행하였으며, 게임잼은 비교과 행사로 성적과 무관함을 고지하고 자발적으로 동의한 응답만 수집하였다. 총 93명이 응답하였고 프로젝트 맥락은 게임잼31명(33.3%), 졸업작품 36명(38.7%), 학기·동아리·개인 프로젝

www.kci.go.kr

트 등 기타 26명(28.0%)이었다. 이 중 65명(69.9%)이 게임 개발 과정에서 생성형AI를 사용한 경험이 있었고(이하"사용 경험자"는 이 기준을 의미하며, 일상적 용도를 포함한 도구 유형별 사용 경험과 구분한다), 사용 경험자 중 63명(96.9%)이 19인이하의 팀에 속해 있었으며, 단계별 인식 문항에는 64명이 응답하였다. 주의 확인 문항에 응답한 64명은 모두 지시된 응답을 선택하였으며 중복 응답은 확인되지 않았다.

2. Survey Instrument

설문 도구는 게임 개발 관점의 생성형AI 교육 수요 조사[1]와 솔로·인디 개발자의 활용 기대와 우려를 다룬 연구[2]의 주제를 참고하여 연구자가 구성하였다. 문항은 활용 실태(A~C), 단계별 체감 효율성·품질·수정 부담(D), 활용 경험(B), 장애 요인·활용 태도·필요 지원(F)의 개념 영역별로 연구 문제에 직접 대응하도록 구성하였으며 별도의 전문가 내용타당도 검토는 수행하지 않았다. 설문은 생성형AI 사용 경험 여부에 따라 분기되는 구조이며, 사용 경험자는 활용 실태와 단계별 인식 문항에 응답하였고 장애 요인과 필요 지원 문항에는 전원이 응답하였다(Table 2). 주요 변수는 활용 빈도(C3), 체감 시간 단축(D1), 산출물 품질(D2), 수정·후처리 부담(D3), 활용 태도(F2~F4), 장애 요인(F1)이며 게임잼 활용 경험 영역(B4~B9)은 보조 변수로 사용하였고, 5점 척도는1점이 가장 낮은 수준, 5점이 가장 높은 수준을 의미한다. 단계별 문항은"개발 파트별로 생성형AI를 얼마나 활용하고 있습니까"(C3), "생성형AI 활용은 각 단계의 작업 시간을 단축시켰다"(D1), "각 단계에서 생성형AI 산출물의 품질은 직접 작업한 결과물과 비교해 만족스러웠다"(D2), "각 단계에서 생성형AI 산출물을 실제 사용하기 위한 수정·후처리에 상당한 시간이 들었다"(D3)를6개 단계 각각에 대해 물었다. 이 문항들은 응답 주체(응답자 본인 또는 팀)를 명시하지 않았고 활용 빈도(C3)는 팀 규모와 팀 단위 활용 여부를 묻는 문항에 이어 제시되었으므로, 단계별 응답은 응답자 본인의 작업에 한정되지 않고 팀 내 해당 단계의 활용에 대한 관찰이 포함될 수 있는 인식으로 해석한다. 또한 단계별 활용 빈도(C3)는 설문에서 졸업작품 및6개월 이상 중장기 프로젝트의 활용 영역에 배치되었으나 게임잼 참여자31명 중16명도 응답하였으므로(C3 응답자65명), C3는 게임잼과 중장기 프로젝트 참여자를 포괄한 단계별 활용 실태로 해석하며 게임잼 응답자가 어떤 프로젝트를 기준으로 응답하였는지는 확인할 수 없다. 응답 주체가 혼재되었을 가능성을 고려하여 해당 단계를 직접 담당한 응답자만의 결과를 IV장2절에 별도로 제시하였으며, 직접 담당자는 담당 역할 문항(복수 응답)에서 해당 단계의 직군(기획, 아트(2D), 아트(3D), 프로그래밍, 사운드, QA)을 선택한 응답자를 뜻하고 PD·PM은 포함하지 않는다. 담당 역할은 해당 단계의 작업 수행 여부를 나타낼 뿐 그 단계에서 본인이 생성형AI를 직접 사용했는지는 확인하지 못한다. 특정 상용 서비스의 평가로 해석되는 것을 피하기 위해 만족도와 효용성은 개발 단계와 도구 유형 수준에서 측정하고 개별 도구명은 사용 현황 문항에서만 수집하였다.

Table 2. Overview of the survey instrument

Section	Content	Response format
S. Screening	Project context; generative AI use experience (branching)	Choice
A. Profile	Completed projects, major track, roles, AI tool types ever used (any purpose)	Choice / multiple
B. Game-jam use experience	Tool types, purposes, perceptions of game-jam use (6 items); analyzed for game-jam users only	Multiple / 5-point
C. Project use	Team size, team-level use, stage-by-stage utilization	5-point / text
D. Stage perception	Time efficiency, output quality, post-processing burden by stage; attention check	5-point
E. Work case	AI-assisted time, estimated manual time, open-ended cases	Numeric / text
F. Barriers / support	Barrier ranking, future attitudes (3 items), needed education and support	Rank / 5-point / multiple

온라인 설문에서는 D 영역의 "해당 없음" 선택지가 구현되지 않아 6개 단계가 모두 응답되었으므로, 단계별 효율성·품질·수정 부담 분석에서는 C 영역의 활용 빈도가"전혀 활용 안 함"인 단계의 응답을 제외하였다(단계별 유효 응답 45~64명). D 영역 응답자는 64명이며 나머지 1명은 모든 단계를"전혀 활용 안 함"으로 응답하였다. 단계별 비활용 이유를 조사하기 위해 구성한 문항(D4)은 온라인 설문 구현 과정에서 앞선 수정·후처리 부담 문항(D3)과 질문 문장이 중복되었다. 이에 따라 질문과 응답 선택지 간 불일치가 확인되어 해당 문항을 주요 분석에서 제외하였으며, 생성형AI 활용의 장애 요인은 전체 응답자를 대상으로 한 별도 문항(F1)을 통해 분석하였다. 또한6개 단계 모두에서 "거의 활용 안함" 이상으로 응답한42명의 완전 응답자 부분집합을 별도로 분석하여 강건성을 확인하였다. 게임잼 활용 경험 영역(B)은 게임잼에서의 활용을 묻도록 구성되었으나 졸업작품과 기타 프로젝트 응답자를 포함한 사용 경험자 65명 전원(게임잼 16명, 졸업작품 33명, 기타 프로

www.kci.go.kr

젝트 16명)이 응답하였다. 후자의 응답자가 어떤 프로젝트를 기준으로 응답하였는지 확인할 수 없으므로 B영역은 도구 유형 문항에서 한 가지 이상을 선택한 게임잼 활용자 15명(게임잼 응답자 16명 중"활용하지 않음"을 선택한 1명 제외)으로 한정한 보조 분석으로 제시하였고, 표본 수가 적어 기술통계만 산출하였다. 이 15명 중 1명은 C영역의 팀 활용 여부 문항에서 "활용하지 않음"으로 응답하였으나 B영역의 직접 응답을 우선하여 포함하였으며, 이 응답자를 제외한 14명에서도 시간 단축 4.36, 품질3.64, 수정·후처리 3.43으로 방향이 같았다. 활용 태도 척도(F2~F4)의 Cronbach's α는.820이었다.

3. Analysis Methods

응답자 특성, 도구 유형, 활용 목적과 장애 요인은 빈도와 백분율로 제시하였다. 리커트 응답의 서열적 특성과 정규성 가정의 제약을 고려하여 검정에는 비모수 방법을 적용하였다. 6개 단계의 활용 빈도는 사용 경험자 65명의 반복측정 자료이므로 Friedman 검정을 적용하였고, 단계별 시간 단축·품질·수정 부담은 미사용 단계를 제외한 기술통계를 제시하되 표본 수가 단계마다 다르므로 42명의 완전 응답자 부분집합에 Friedman 검정을 적용하였다. 유의한 경우 Wilcoxon 부호순위 검정에 15개 단계 쌍의 Bonferroni 교정(조정 유의수준 .0033)을 적용하였고, 효과크기는 Kendall's W로 제시하였다. 게임잼과 졸업작품의 비교에서 생성형AI 사용 경험 비율은 Fisher의 정확검정을, 단계별 인식의 6단계 평균값(응답자별로"전혀 활용 안 함"이 아닌 단계의 시간 단축·품질·수정 부담 점수를 각각 평균한 값)은Mann-Whitney U 검정을 적용하였으며, 두 집단은 학습 단계와 개발 경험도 함께 다르므로 집단 간 비교는 탐색적 비교로 제시하였다. 해당 단계를 직접 담당한 응답자와 그렇지 않은 응답자의 단계별 인식 차이는 Mann-Whitney U 검정으로 비교하되 6개 단계와 3개 인식 문항의 18개 비교에 Bonferroni 교정(조정 유의수준 .0028)을 적용하였다. 전공 트랙과 완성 프로젝트 수에 따른 차이는 Kruskal-Wallis 검정으로 확인하였으며, 전공 트랙 비교는 18개 문항(단계별 시간 단축·품질·수정 부담)이 대상이므로 Bonferroni 교정(조정 유의수준.0028)을 함께 제시한다. 생성형AI 사용 경험 여부에 따른 활용 태도 3개 문항(F2~F4)의 비교는Mann-Whitney U 검정에 3개 비교에 Bonferroni 교정을 적용하여 보정p값(원p값×3)을.05와 비교하였다. 실제 작업 소요시간 문항은 회고적 자기보고이므로 AI 활용시간과 수작업 예상시간을 모두 시간으로 환산할 수 있는 응답(분·시간 단위는 시간으로, 일 단위는 1일 8시간, "일주일"은 56시간으로 환산)의 중앙값과 쌍별 시간 절감률을 기술통계로만 제시하였고, 개방형 응답은 내용을 읽고 주제별로 범주화하되(생성형AI 보조의 단일 분류자) 복수 주제는 각 범주에 중복 집계하고 효과나 문제가 없다는 응답은 제외하였다. 장애 요인 문항은 응답별 선택 개수가 일부 달랐으므로 복수응답 선택 빈도를 중심으로 분석하였고, 순위는 기입된 응답 93건에서 첫 번째로 기재된 요인으로 집계하였으며, 순위 표기가 없는 23건은 기재 순서가 순위를 뜻한다고 가정하였으므로 순위가 명시된 70건만의 결과를 함께 제시하였다.

IV. Results

1. Respondent Characteristics and Tool Types

응답자 93명의 특성은 Table 3과 같다. 전공 트랙은 아트(3D) 37명(39.8%), 기획 24명(25.8%), 프로그래밍 16명(17.2%), 아트(2D) 14명(15.1%) 순이었고, 완성한 프로젝트 수는 2~3개가 37명(39.8%)으로 가장 많았으며, 담당 역할(복수 응답)은 아트(3D) 37명, 기획 32명, 아트(2D) 29명, PD·PM 28명, 프로그래밍 22명으로 다수가 복수의 역할을 겸하고 있었다. 생성형AI 사용 경험은 졸업작품 집단이 36명 중 33명(91.7%)인 반면 게임잼 집단은 31명 중 16명(51.6%)에 그쳤다. 전공별로는 기획(24명 중 23명)과 프로그래밍(16명 중 14명) 트랙의 사용률이 높았던 반면 아트(2D)는 14명 중 4명, 아트(3D)는 37명 중 23명으로 낮았으며, 미사용자 28명 중 24명(85.7%)이 아트 트랙이었다.

Table 3. Respondent characteristics (N = 93)

Characteristic	Category	n (%)
Project context	Graduation project / Game jam / Other	36 (38.7) / 31 (33.3) / 26 (28.0)
Major track	3D art / Planning / Programming / 2D art / Other	37 (39.8) / 24 (25.8) / 16 (17.2) / 14 (15.1) / 2 (2.2)
Completed projects	None / 1 / 2-3 / 4 or more	15 (16.1) / 28 (30.1) / 37 (39.8) / 13 (14.0)
Generative AI experience	Yes / No	65 (69.9) / 28 (30.1)

게임개발 이외의 일상적 용도를 포함하여 사용해 본 생성형AI 도구 유형(N = 93, 복수 응답)은 텍스트·기획 생성 81명(87.1%), 3D 에셋 생성 55명(59.1%), 2D 이미지 생성 53명(57.0%), 코드 생성 37명(39.8%), 오디오 생성 24명(25.8%) 순이었으며, 게임개발 과정에서의 사용 경험이 없는 28명 중에서도 22명이 텍스트·기획 생성 도구를 사

www.kci.go.kr

용해 본 적이 있어 도구 접촉 경험과 개발 과정에서의 활용은 구분된다. 게임잼에서 생성형AI를 실제로 활용한 응답자(n = 15)가 선택한 산출물의 주된 용도(복수 응답)는 아이디어 탐색·레퍼런스 9명(60.0%), 코드·기능 구현 7명(46.7%), 캐릭터·오브젝트 에셋 6명(40.0%) 순이었으며 최종 빌드에 포함된 에셋·콘텐츠로 활용한 경우는 4명(26.7%)이었다. 이를 통해 산출물이 최종 결과물보다 탐색과 임시 자원으로 활용되는 경향을 확인할 수 있었다.

2. Stage-by-Stage Utilization, Efficiency, Quality, and Post-Processing Burden

단계별 활용 빈도와 인식은Table 4 및Fig. 1과 같다. 활용 빈도(n = 65)는 프로그래밍(M = 4.31)과 기획(M = 4.06)이 가장 높아 각각87.7%, 81.5%가"자주" 이상으로 활용한다고 응답한 반면 아트(2D)와 아트(3D)는 모두3.31, 사운드는3.05였으며QA·테스트는2.58("자주" 이상26.2%)로 가장 낮았다(Friedman $\chi^2(5) = 118.10$, $p < .001$, $W = .363$). 사후검정에서 프로그래밍과 기획은 나머지 네 단계보다, 아트(2D)와 아트(3D)는QA·테스트보다 유의하게 높았다.

게임잼 활용 경험 영역(B)의 인식은 게임잼에서 생성형AI를 실제로 활용한 15명을 기준으로 보조 분석하였다. 이 응답자의 93.3%(M = 4.40)가"생성형AI는 제작 시간을 단축시켰다"에 동의하였고 향후 활용 의향은 86.7%(M = 4.53), "주 담당 작업에 더 많은 시간을 쓸 수 있었다"는 73.3%(M = 3.93)였으나, "산출물 품질은 기대 수준에 부합했다"와"생성형AI가 없었다면 제한 시간 내 필요한 에셋·콘텐츠 확보가 어려웠을 것이다"는 각각 53.3%(M = 3.53)이었고"수정·후처리에 상당한 시간이 들었다"는 40.0%(M = 3.47)가 동의하여 시간 단축에 대한 동의에 비해 품질 만족에 대한 동의는 낮은 수준이었다. 사용 경험자 65명 전체의 평균은 시간 단축 4.26, 품질 3.38, 수정·후처리3.62, 에셋 확보 3.69, 주 담당 작업 4.00, 활용의향 4.63으로 방향이 같았으나 기준 프로젝트가 불명확하여 참고 값으로만 제시한다. 미사용 단계를 제외한 단계별 인식에서 시간 단축은 프로그래밍(M = 4.31, n = 62)과 기획(M = 4.08, n = 64)이 가장 높고QA·테스트(M = 3.53, n = 45)가 가장 낮았으며, 품질 만족은 프로그래밍(3.79)과 기획(3.59)이 높고 아트(3D)가3.03으로 가장 낮았다. 수정·후처리 부담은 순서가 역전되어 아트(3D)가4.05, 아트(2D)가3.85로 가장 높았다.

Table 4와Fig. 1을 종합하면 활용 빈도가 가장 높은 프로그래밍과 기획은 시간 단축과 품질 인식이 높고 수정·후처리 부담이 아트 단계보다 낮은 반면, 아트(2D)와 아트(3D)는 활용 빈도가 중간 수준이면서 수정·후처리 부담이 높고 품질 인식이 낮았으며, 활용 빈도가 가장 낮은QA·테스트는 시간 단축 인식도 가장 낮았다. 다만 이는 단계별 평균의 비교이며 활용 빈도와 인식 사이의 관계를 검증한 것은 아니다.

Table 4. Descriptive statistics by stage (M ± SD; Use = utilization frequency, Time = time efficiency, Quality = output quality, Post-proc. = post-processing burden; non-use excluded)

Stage	Use	Time	Quality	Post-proc
Planning	4.06 ± 0.81	4.08 ± 0.76 (64)	3.59 ± 0.83 (64)	3.48 ± 0.85 (64)
2D art	3.31 ± 1.09	3.79 ± 0.93 (61)	3.28 ± 1.03 (61)	3.85 ± 0.81 (61)
3D art	3.31 ± 1.21	3.66 ± 1.10 (58)	3.03 ± 1.09 (58)	4.05 ± 0.93 (58)
Programming	4.31 ± 1.00	4.31 ± 0.74 (62)	3.79 ± 0.83 (62)	3.53 ± 0.94 (62)
Sound	3.05 ± 1.36	3.72 ± 0.97 (53)	3.42 ± 0.97 (53)	3.36 ± 0.94 (53)
QA / Test	2.58 ± 1.31	3.53 ± 0.84 (45)	3.22 ± 0.93 (45)	3.18 ± 0.86 (45)

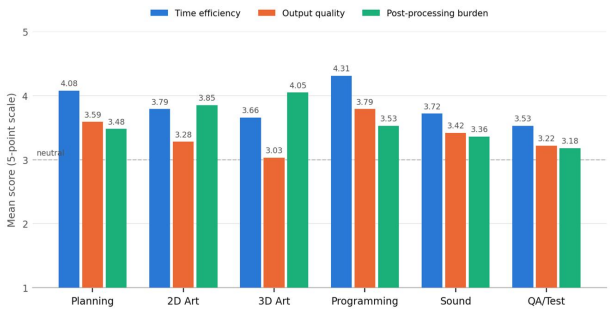


Fig. 1. Perceived time efficiency, output quality, and post-processing burden by stage (n = 45–64; 5-point scale)

6개 단계 모두에서 활용 빈도가"거의 활용 안함" 이상인 42명을 대상으로 한 민감도 분석에서 Friedman 검정 결과, 시간 단축($\chi^2(5) = 43.44$, $W = .207$), 품질($\chi^2(5) = 32.21$, $W = .153$), 수정·후처리 부담($\chi^2(5) = 45.08$, $W = .215$)은 모두 유의한 단계 차이를 보였다($p < .001$). Bonferroni 교정 사후검정에서 시간 단축은 프로그래밍과 기획이 사운드와 QA·테스트보다 높았고 프로그래밍은 아트(3D)보다도 높았으며, 품질은 프로그래밍이 아트(3D), 사운드, QA·테스트보다, 기획이 아트(3D)와 QA·테스트보다 높았다. 수정·후처리 부담은 아트(3D)가 기획, 프로그래밍, 사운드, QA·테스트보다, 아트(2D)가 기획, 사운드, QA·테스트보다 높았다. 아트(3D)에서"전혀 활용 안 함"을 제외한 응답자(n = 58)를 기준으로 시간 단축 긍정 응답이 65.5%인 반면 품질 만족 긍정 응답은 39.7%, 수정·후처리 부담 긍정 응답은 79.3%로 나타나 시간 단축에 대한 긍정

적 인식과 높은 후처리 부담이 동시에 나타난 단계였다. 미사용 단계를 제외하지 않은 전체 64명 자료에서도 같은 결과와 검정의 유의성이 확인되었다. 단계별 문항의 응답 주체가 명시되지 않았다는 점을 고려하여 해당 단계를 직접 담당한 응답자만을 대상으로 한 결과를 Table 5에 제시하였다. 직접 담당자의 시간 단축 인식은 프로그래밍(4.30, n = 20)과 기획(4.30, n = 27)이 가장 높았고, 수정·후처리 부담은 아트(3D) 4.12(n = 26)와 아트(2D) 4.09(n = 11)가 높았으며, 품질 만족은 프로그래밍 4.00이 가장 높고 아트(3D) 3.27이 가장 낮았다. 이는 전체 응답에서 확인한 상위·하위 방향과 같았으나 아트(3D), 사운드, 아트(2D) 간 시간 단축 순위는 전체 응답과 달랐으며, 사운드(n = 6)와 QA·테스트(n = 4)는 직접 담당자가 적어 기술통계로만 제시한다. 직접 담당자와 비담당자를 비교한 18개 검정 중 보정 전 유의확률이.05 미만인 것은 기획 단계의 수정·후처리 부담 1건(3.85 vs 3.22, Mann-Whitney U, p = .005)뿐이었고 Bonferroni 교정(조정 유의수준.0028) 후에는 이 비교도 유의하지 않았다(조정p = .095). 따라서 직접 담당 여부에 따른 인식 차이는 본 표본에서 확인하지 못하였다. 준정량 문항에서 AI 활용시간과 수작업 예상시간을 모두 시간 단위로 비교할 수 있었던 34건(환산 기준은 III장 3절)에서 두 시간의 중앙값은 각각 1.0시간과 8.0시간이었고, 쌍별 시간 절감률의 중앙값은 71.4%였으며 33건에서 AI 활용시간이 더 짧았으며, "일주일"을 40시간으로 환산해도 절감률 중앙값은 70.7%로 결과가 유지되었다. 다만 수작업 시간은 회고적 추정이므로 체감 효율성을 보완하는 참고 자료로만 해석하였다.

Table 5. Stage perceptions among direct performers (M ± SD, n; abbreviations as in Table 4)

Stage	Use	Time	Quality	Post-proc
Planning	4.33 ± 0.73 (27)	4.30 ± 0.72 (27)	3.59 ± 0.93 (27)	3.85 ± 0.82 (27)
2D art	3.08 ± 1.24 (12)	3.64 ± 0.81 (11)	3.36 ± 1.12 (11)	4.09 ± 0.83 (11)
3D art	3.69 ± 1.05 (26)	3.92 ± 0.93 (26)	3.27 ± 0.96 (26)	4.12 ± 0.82 (26)
Programming	4.50 ± 0.51 (20)	4.30 ± 0.57 (20)	4.00 ± 0.79 (20)	3.65 ± 1.09 (20)
Sound	3.43 ± 1.72 (7)	3.83 ± 1.17 (6)	3.67 ± 0.82 (6)	3.50 ± 1.22 (6)
QA / Test	2.50 ± 1.38 (6)	3.25 ± 0.96 (4)	3.50 ± 1.00 (4)	4.00 ± 1.15 (4)

Direct performers: respondents who selected the stage's role in the multiple-response role item (PD/PM excluded). Utilization n includes all direct performers; perception n excludes those who never used generative AI in the stage or did not answer the stage-perception items.

3. Comparison by Development Context and Respondent Characteristics

게임잼 31명과 졸업작품 36명의 비교는 Table 6과 같다. 두 집단은 전공 구성이 달라 게임잼은 아트 트랙이 19명(61.3%; 아트(2D) 11명, 아트(3D) 8명)인 반면 졸업작품은17명(47.2%; 아트(3D) 15명, 아트(2D) 2명)이었고, 완성 프로젝트가 1개 이하인 응답자도 게임잼 17명(54.8%), 졸업작품 8명(22.2%)으로 게임잼 집단의 개발 경험이 적었다. 생성형AI 사용 경험자는 게임잼 16명(51.6%), 졸업작품 33명(91.7%)으로 Fisher의 정확 검정에서 유의한 차이가 있었다(p < .001). 전공을 나누어 보면 아트 트랙에서는 게임잼 19명 중 6명(31.6%), 졸업작품 17명 중 15명(88.2%)으로 차이가 유지되었으나(p < .001), 아트 이외 트랙에서는 게임잼 12명 중 10명, 졸업작품 19명 중 18명으로 차이가 없어(p = .544), 집단 간 AI 사용 경험 비율의 차이는 주로 아트 트랙에서 관찰되었다. 다만 개발 경험과 프로젝트 조건이 함께 달라 차이의 원인을 특정하기에는 한계가 있다. 단계별 인식에 응답한AI 경험자(게임잼 15명, 졸업작품 33명)의 6단계 평균 체감 효율성(게임잼 3.52, 졸업작품 3.97; U = 166, p = .070), 품질, 수정 부담과 소규모 개발 시 활용 의향(4.44 vs 4.73, p = .176)에서는 유의한 차이를 확인하지 못하였다. 다만 게임잼 경험자가 15명에 불과하여 검정력이 제한되므로 이를 두 집단의 효용 인식이 같다는 근거로 해석할 수는 없으며, 두 집단은 전공 구성과 개발 경험에서도 차이가 있어 사용률 차이를 프로젝트 기간 자체의 효과로 해석하기는 어렵다. 또한 단계별 문항(C3)이 중장기 프로젝트 영역에 배치되어 게임잼 응답자가 기준으로 삼은 프로젝트를 확인할 수 없으므로, 이 비교는 개발 맥락의 직접 비교가 아닌 응답자 집단의 비교로 제한하여 해석한다.

Table 6. Comparison between game jam and graduation-project contexts

Item	Game jam	Grd.project	Test
Art-track respondents	19/31 (61.3%)	17/36 (47.2%)	-
Completed projects ≤ 1	17/31 (54.8%)	8/36 (22.2%)	-
AI-use experience	16/31 (51.6%)	33/36 (91.7%)	Fisher p < .001
AI-use experience, art track only	6/19 (31.6%)	15/17 (88.2%)	Fisher p < .001
AI-use experience, other tracks	10/12 (83.3%)	18/19 (94.7%)	Fisher p = .544
Efficiency composite	3.52 (n=15)	3.97 (n=33)	U=166, p=.070

www.kci.go.kr

Quality composite	3.32 (n=15)	3.49 (n=33)	U=235.5, p=.797
Post-processing composite	3.56 (n=15)	3.55 (n=33)	U=249.5, p=.973
Small-team use intention (F3)	4.44 (n=16)	4.73 (n=33)	U=211.5, p=.176

전공 트랙에 따른 비교는 18개 문항에 대한 Kruskal-Wallis 검정으로 수행한 탐색적 분석이며 아래 유의확률은 보정 전 값이다. 기획 단계의 시간 단축 인식은 기획 트랙(M = 4.35)에서 가장 높고 아트(2D) 트랙(M = 3.33)에서 가장 낮았으며(Kruskal-Wallis H = 7.87, p = .049), 아트(3D) 단계의 시간 단축 인식은 아트(3D) 트랙(M = 4.00)에서 가장 높고 프로그래밍 트랙(M = 2.92)에서 가장 낮았다(H = 7.85, p = .049). 그러나 이 두 건은 수행한 18개 검정 중 보정 전 유의확률이.05 미만인 전부이며 유의확률도 기준에 근접한 수준이고, Bonferroni 교정(조정 유의수준.0028)을 적용하면 모두 유의하지 않다. 아트(2D) 트랙의 사용 경험자는 4명에 불과하므로 트랙 간 차이는 후속 연구에서 검증할 가설 수준의 탐색적 결과로만 제시한다. 완성 프로젝트 수에 따른 6단계 평균값의 차이는 모두 유의하지 않았다(p > .17).

4. Obstacles and Needed Support

전체 응답자 93명의 장애 요인 선택 빈도와 필요 지원은 Table 7과 같다. 이용 요금이 69명(74.2%)으로 가장 많았고, 수정·후처리 부담 50명(53.8%), 산출물 품질 편차 48명(51.6%), 스타일 일관성 부족 41명(44.1%), 저작권·표절 우려 34명(36.6%)이 뒤를 이었다. 첫 번째로 기재된 요인(n = 93)에서도 이용 요금이 37명(39.8%)으로 가장 많았으며, 순위가 명시된 70건만 보아도 이용 요금이 25명(35.7%)으로 가장 많았다. 필요한 지원은 프롬프트 작성법과 산출물 후처리 기법이 각 60명(64.5%), 유료 도구 지원 58명(62.4%), 파이프라인 통합 방법 48명(51.6%) 순으로, 장애 요인으로 확인된 이용 요금과 후처리 부담에 직접 대응하는 수요로 해석된다.

Table 7. Major barriers and needed support (N = 93, multiple responses)

Rank	Barrier, n (%)	Needed support, n (%)
1	Usage fees, 69 (74.2)	Prompt writing, 60 (64.5)
2	Post-processing burden, 50 (53.8)	Post-processing skills, 60 (64.5)
3	Output quality variance, 48 (51.6)	Paid-tool support, 58 (62.4)
4	Style inconsistency, 41 (44.1)	Pipeline integration, 48 (51.6)
5	Copyright / plagiarism, 34 (36.6)	Use-case sharing, 45 (48.4)
6	Prompt learning cost, 22 (23.7)	Copyright / license education, 40 (43.0)
7	Ethical reluctance, 17 (18.3)	Tool practice training, 35 (37.6)

생성형AI에 대한 태도(N = 93)는"활용 역량은 향후 진로에 중요할 것이다"(M = 4.53, 동의 94.6%), "소규모로 개발한다면 적극 활용할 것이다"(M = 4.57, 94.6%), "인력이 부족한 소규모 팀의 한계를 보완해 줄 수 있다"(M = 4.38, 87.1%)에 대한 동의가 모두 높았으며, 사용 경험자는 미사용자보다 활용 의향을 높게 평가하였으며(4.69 vs 4.29, Mann-Whitney U = 1190, 원 p = .006, 보정 p = .017) 이 차이는 3개 문항에 대한Bonferroni 교정 후에도 유의하였다. 진로 중요도(4.63 vs 4.29, 원 p = .023, 보정 p = .070)와 소규모 팀 인력 한계 보완(4.43 vs 4.25, 원 p = .147, 보정 p = .441)에서는 사용 경험자의 평균이 높았으나 보정 후 유의한 차이는 확인되지 않았다.

개방형 응답에서 가장 효과적이었던 경험(내용이 기재된 62건, 복수 범주 중복 집계)은 코드 작성·디버깅·자동화(17건), 기획서 정리·아이디어 구체화(13건), 3D 모델링의 초기 형태·더미 생성(13건), 비전문 파트의 보완(13건), 레퍼런스·이미지·텍스처 생성(12건)으로 나타나 인력 부족을 보완하는 효용이 확인되었다. 반면 기대에 못 미친 경험(62건)에서는 이미지의 프롬프트 불일치와 스타일 일관성 부족(16건), 3D 모델의 메쉬 품질과 리토폴로지 등 후처리 부담(16건), 코드·자료의 환각과 유지보수·검증 어려움(13건)이 확인되었고 토큰·요금 부담도 4건 언급되어 정량 결과와 방향이 일치하였다.

V. Conclusions

본 연구에서는 게임 전공 학생의 소규모 게임개발 환경(게임잼 및 팀 단위 프로젝트)에서 생성형AI가 실제로 어느 단계에서 활용되고 있으며 개발자가 체감하는 효율성과 품질, 수정 부담이 단계별로 어떻게 나타나는지를 단일 교육기관의 학생 개발자 93명의 설문을 통해 분석하였다. 본 연구는 객관적인 생산성 실험이 아니라 실제 개발 경험에 기반한 활용 실태와 체감 효율성을 조사한 연구이므로, 결과는 단계별 인식과 활용 경향을 중심으로 해석할 필요가 있다. 주요 결과를 연구 문제별로 정리하면 다음과 같다.

첫째, 생성형AI 활용은 프로그래밍과 기획 단계에 집중되어 있었다. 두 단계는 활용 빈도와 시간 단축 인식, 품질

www.kci.go.kr

만족이 모두 가장 높았으며, 이는 텍스트와 코드처럼 수정·검토의 단위가 명확한 특징과 관련될 가능성이 있으나 본 조사에서 직접 확인하지는 않았다. 반면 QA·테스트는 활용 빈도가 가장 낮았다. 또한 게임잼 활용자의 산출물은 최종 빌드에 포함된 경우(26.7%)보다 아이디어 탐색과 레퍼런스(60.0%), 프로토타입용 임시 자원으로 활용되는 경향이 있어, 생성형AI가 소규모 개발의 탐색·검증 단계를 가속하는 도구로 활용되고 있음을 시사한다.

둘째, 단계별 효용 인식에는 유의한 차이가 있었으며 아트(3D)에서는 시간 단축에 대한 긍정적 인식과 높은 후처리 부담이 동시에 나타났다. 아트(3D)는 시간 단축 인식이 3.66점이었지만 품질 만족도는 3.03점으로 가장 낮고 수정·후처리 부담은 4.05점으로 가장 높았으며, 개방형 응답에서도 리토폴로지와 메쉬 정리가 반복적으로 언급되었다. 이는 Panchanadikar와 Freeman[2]이 보고한 산출물 검증 부담에 대한 우려와 방향이 일치하는 결과로, 본 조사 대상의 3D 아트 단계에서는 초기 형태나 더미 모델 단계의 효용과 최종 적용을 위한 후처리 필요성이 함께 인식되었음을 시사한다. 다만 이를3D 제작 전반에 대한 일반적인 결론으로 확대할 수는 없다. 2D 아트에서도 스타일 일관성이 주요 장애로 나타나 시각 에셋 영역에서는 생성 속도와 일관성 유지 비용을 함께 고려할 필요가 있다[3]. 프로그래밍 단계의 높은 효율성 인식은 통제 실험[6]과 방향이 일치하지만 개방형 응답에는 코드의 환각과 유지보수 문제도 나타났으며, 숙련 개발자의 코드베이스에서AI 도구가 작업 시간을 증가시킨 연구[7]와 함께 고려하면 생산성 효과는 사용자 숙련도, 과제 복잡도, 검증 비용에 따라 달라질 수 있다.

셋째, 이용 요금, 수정·후처리 부담, 산출물 품질 편차, 스타일 일관성 부족이 핵심 장애 요인이었으며(전체 응답자 기준), 단계별 문항에서도 아트 단계의 수정·후처리 부담이 높게 인식되었다. 필요한 지원으로는 프롬프트 작성법뿐 아니라 후처리 기법과 유료 도구 지원, 파이프라인 통합 방법이 높은 비율로 선택되었다. 2024년의 교육 연구에서 학습자들이 프롬프트 작성 역량을 중요하게 인식했던 결과[1]와 비교하면, 개발 경험이 축적된 본 조사에서는"잘 생성하는 방법"뿐 아니라"생성된 결과를 프로젝트에 어떻게 통합하고 수정할 것인가"에 대한 요구가 함께 나타났으며, 이는 실무자들이 자동화보다 창작 통제와 투명성을 중시한 최근 조사[14]와도 방향이 유사하다.

넷째, 졸업작품 집단의 생성형AI 사용 경험 비율이 게임잼 집단보다 높았고 이 차이는 아트 트랙 응답자에게서 집중적으로 나타났으나, AI 경험자 간 효용 인식과 활용 의향에서는 유의한 차이를 확인하지 못하였다. 다만 게임잼 경험자가 15명에 불과하고 두 집단의 전공 구성과 개발 경험이 다르므로 개발 맥락 자체의 효과로 단정할 수 없으며, 개발 경험이 적은 아트 트랙 참가자의 낮은 사용률이 도구 접근성이나 활용 사례 부족과 관련될 가능성은 본 조사에서 직접 확인하지 않은 추정으로 후속 연구의 과제로 남긴다. 해당 단계를 직접 담당한 응답자와 그렇지 않은 응답자의 인식 차이는 다중 검정을 보정하면 유의하지 않았고, 직접 담당자만의 결과에서도 프로그래밍·기획의 시간 단축 인식이 높고 아트 단계의 수정·후처리 부담이 높은 방향은 전체 응답과 같았다(중간 단계의 순위는 달랐다). 이는 단계별 응답 주체가 명시되지 않았다는 한계에도 본 연구의 단계별 결과가 일정 부분 유지됨을 보여준다.

이상의 결과는 본 조사 대상인 학생 개발자의 응답에 한정되나, 소규모 팀이 생성형AI를 기획 문서화와 코드 작성, 프로토타입 검증에 우선 적용하되 아트 산출물은 후처리 역량과 시간을 전제로 도입 범위를 정해야 할 가능성을 시사한다. 교육·지원 측면에서는 프롬프트 작성법(64.5%)과 산출물 후처리 기법(64.5%), 유료 도구 지원(62.4%), 파이프라인 통합 방법(51.6%)에 대한 요구가 높았으며(Table 7), 이는 이용 요금(74.2%)과 수정·후처리 부담(53.8%)이 주요 장애 요인으로 나타난 결과와 대응하므로 프롬프트 작성과 후처리를 함께 다루는 교육과 도구 이용을 지원하는 방안이 병행될 필요가 있다. 사운드·QA 단계의 활용이 낮았던 원인은 본 조사에서 확인하지 못하였으므로 후속 연구에서 별도로 조사할 필요가 있다.

본 연구의 학술적 기여는 다음과 같다. 첫째, Table 1에서 정리한 기존 연구가 특정 영역이나 과제 단위로 이루어진 것과 달리, 게임개발 파이프라인을 6개 단계로 구분하여 단계별 활용 빈도와 체감 시간 효율성, 품질, 수정·후처리 부담을 동일 응답자 내에서 종합적으로 비교하였다. 둘째, 3D 아트 단계에서 시간 단축에 대한 긍정적 인식과 높은 후처리 부담이 함께 나타나는 양상을 확인하였다. 셋째, 게임잼과 졸업작품의 생성형AI 활용 양상을 전공 구성을 고려하여 탐색적으로 비교하였다. 넷째, 미사용 단계의 사후 제외와 응답 주체의 불명확성을 직접 담당자 분석과 완전 응답자 민감도 분석으로 점검하여 단계별 결과의 해석 범위를 명확히 하였다.

본 연구의 한계는 다음과 같다. 첫째, 단일 교육기관의 학생 개발자를 대상으로 하였으므로 상업적 성과와 장기 운영의 압력이 있는 현업 개발자 전체로 일반화하기에는 제약이 있다. 둘째, 효율성과 품질은 자기보고 인식에 기반하며 실측 자료가 아니고, 개방형 응답의 범주화는 단일 분

www.kci.go.kr

류자에 의한 것이다. 셋째, 온라인 설문에서 "해당 없음" 선택이 구현되지 않아 미사용 단계의 응답을 사후 제외하고 완전 응답자 민감도 분석을 병행하였으나 설문 구현상의 한계가 완전히 해소되지는 않으며, 게임잼 활용 경험 영역(B)에는 게임잼 외 프로젝트 응답자도 응답하여 게임잼 활용자 15명의 보조 분석으로 한정하였으므로 해당 결과는 일반화할 수 없고, 단계별 문항이 응답 주체를 명시하지 않았으며 단계별 비활용 이유 문항(D4)은 질문과 선택지가 일치하지 않아 분석에서 제외하였으므로 단계별 비활용의 원인은 확인하지 못하였고, 단계별 결과는 응답자 본인의 직접 경험과 팀 내 관찰이 혼재된 인식으로 해석해야 한다. 넷째, 연구자가 구성한 문항을 사용하였으므로 표준화된 척도에 기반한 측정이 이루어지지 않았으며, 후속 연구에서는 전문가 내용타당도 검토를 통한 문항 타당화가 필요하다. 다섯째, 게임잼과 졸업작품 집단은 무작위로 구성되지 않았고 전공 구성과 개발 경험이 달라 맥락 비교에는 교란 가능성이 있으며, 전공 트랙 비교는 18개 검정 중 보정 전 유의확률이.05 미만인 2건이 보정 후 유의하지 않은 탐색적 결과이며, 단계별 활용 빈도(C3)에 응답한 게임잼 참여자가 참조한 프로젝트를 확인할 수 없어 게임잼과 중장기 프로젝트 참여자의 활용 실태가 포괄되어 있다. 후속 연구에서는 현업의 소규모 개발팀으로 표본을 확장하고, 동일 과제를 생성형AI 사용·비사용 조건으로 나누어 실제 작업시간과 후처리 시간을 측정할 필요가 있다. 특히 효용과 부담의 대비가 가장 컸던 3D 아트와 체감 효율성이 가장 높았던 프로그래밍의 작업 로그를 수집하면 생성형AI가 생성 시간을 줄이는 것인지 프로젝트 전체의 완료 시간을 줄이는 것인지를 검증할 수 있을 것이며, 실무 적용을 위해 제작 관행의 변화까지 살펴볼 필요가 있다[11]. 본 조사 대상의 응답에 비추어 소규모 게임개발에서 생성형AI의 활용 가치는 부족한 인력을 보완하고 아이디어와 프로토타입을 빠르게 구체화하되 최종 품질을 위한 인간의 검증과 후처리를 결합하는 데 있을 것으로 생각된다.

ACKNOWLEDGEMENT

This study utilized generative artificial intelligence to assist with the literature review, statistical analysis code development, and analysis of results. The author assumes full responsibility for the research design, data collection, interpretation of results, and final manuscript.

AI Tool Used: OpenAI, ChatGPT. [5.6Sol]. [https://chatgpt.com/share/6aab5b56-e4fc-83ee-a373-6ea33373a1fd]. 2026.07.10.-2026.09.18.

REFERENCES

[1] Bu-ho Choi, "Investigating learner perceptions for effective teaching of Generative AI - from a game development perspective -," Journal of The Korea Society of Computer and Information, Vol. 29, No. 11, pp. 137-144, November 2024. DOI: 10.9708/jksci.2024.29.11.137

[2] Ruchi Panchanadikar and Guo Freeman, "I'm a Solo Developer but AI is My New Ill-Informed Co-Worker: Envisioning and Designing Generative AI to Support Indie Game Development," Proceedings of the ACM on Human-Computer Interaction, Vol. 8, No. CHI PLAY, Article 317, pp. 1-26, October 2024. DOI: 10.1145/3677082

[3] Eun-Joung Kim and Jung-Yoon Kim, "A Survey on Generative AI-driven 2D Game Graphic Asset Production," Journal of Next-generation Convergence Information Services Technology, Vol. 14, Suppl. 1, pp. 109-119, November 2025. DOI: 10.29056/jncist.2025.11.10

[4] Ju-Yeong Kim and Jae-Beom Kim, "A study of dialog generation at game data loading time based on LLM - A focus on RPG," Journal of Korea Game Society, Vol. 24, No. 4, pp. 121-130, August 2024. DOI: 10.7583/JKGS.2024.24.4.121

[5] Mahdi Farrokhi Maleki and Richard Zhao, "Procedural Content Generation in Games: A Survey with Insights on Emerging LLM Integration," Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment, Vol. 20, No. 1, pp. 167-178, Lexington, USA, November 2024. DOI: 10.1609/aiide.v20i1.31877

[6] Sida Peng, Eirini Kalliamvakou, Peter Cihon, and Mert Demirer, "The Impact of AI on Developer Productivity: Evidence from GitHub Copilot," arXiv preprint arXiv:2302.06590, February 2023. DOI: 10.48550/arXiv.2302.06590

[7] Joel Becker, Nate Rush, Elizabeth Barnes, and David Rein, "Measuring the Impact of Early-2025 AI on Experienced Open-Source Developer Productivity," arXiv preprint arXiv:2507.09089, July 2025. DOI: 10.48550/arXiv.2507.09089

[8] Bu-ho Choi, "Training in the production of effective prototypes using Core Mechanic Diagram and Unreal Blueprint," Journal of The Korea Society of Computer and Information, Vol. 25, No. 11, pp. 75-82, November 2020. DOI: 10.9708/jksci.2020.25.11.075

[9] Bu-ho Choi, "Educational examples of game development methodology and paper prototyping for effective digital game creation," Journal of The Korea Society of Computer and

www.kci.go.kr

Information, Vol. 26, No. 11, pp. 59-66, November 2021. DOI: 10.9708/jksci.2021.26.11.059

[10] Roberto Gallotta, Graham Todd, Marvin Zammit, Sam Earle, Antonios Liapis, Julian Togelius, and Georgios N. Yannakakis, "Large Language Models and Games: A Survey and Roadmap," IEEE Transactions on Games, Early Access, pp. 1-18, September 2024. DOI: 10.1109/TG.2024.3461510

[11] Alexandru Ternar, Alena Denisova, João M. Cunha, Annakaisa Kultima, and Christian Guckelsberger, "Generative AI in Game Development: A Qualitative Research Synthesis," Proceedings of the 2026 CHI Conference on Human Factors in Computing Systems, Article 452, pp. 1-24, Barcelona, Spain, April 2026. DOI: 10.1145/3772318.3791206

[12] Josef Spjut, "A Generative AI Game Jam Case Study from October 2024," Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 612-618, Nashville, USA, June 2025. DOI: 10.1109/CVPRW 67362.2025.00066

[13] Jeanette Falk and Nanna Inie, "Materializing the abstract: Understanding AI by game jamming," Frontiers in Computer Science, Vol. 4, Article 959351, pp. 1-20, September 2022. DOI: 10.3389/fcomp.2022.959351

[14] Bojan Endrovski, Joris Dormans, and Rafael Bidarra, “What game developers actually want from procedural level generation tools,” Proceedings of the 21st International Conference on the Foundations of Digital Games (FDG 2026), Article No. 108, pp. 1-18, Copenhagen, Denmark, August 2026. DOI: 10.1145/ 3815598.3815682

[15] Julian Togelius, Georgios N. Yannakakis, Kenneth O. Stanley, and Cameron Browne, "Search-Based Procedural Content Generation: A Taxonomy and Survey," IEEE Transactions on Computational Intelligence and AI in Games, Vol. 3, No. 3, pp. 172-186, September 2011. DOI: 10.1109/TCIAIG.2011.2148116

Authors



Bu-ho Choi received the B.S. degree in Software Engineering from Seoul Digital University, Korea, in 2018, and the M.S. degree in Game Technology from Gachon University, Korea in 2023.

Bu-ho Choi, who received the B.S. degree in Software Engineering from Seoul Digital University, Korea, in 2018, and the M.S. degree in Game Technology from Gachon University, Korea, in 2023, has been working as a game developer since 2008 and is now an assistant professor of School of Game at ChungKang College of Cultural Industries, with research interests in game development, game artificial intelligence, game design, and generative AI for game production.

www.kci.go.kr