Powering the Future of Games: A Scoping Review of Generative AI's Role in Game Development Life Cycle (2020-2024)

Anthony Korsah[1,*], Juho Mattila[1], Sameera Bandaranayake[1], Prabhash Rathnayake[1], Pasi Karppinen[2] and Iikka Paajala[1]

[1]University of Oulu, Pentti Kaiteran katu 1, 90570 Oulu, Finland
[2]Xamk - South-Eastern Finland University of Applied Sciences, Patteristonkatu 3, 50100 Mikkeli, Finland

Abstract

This paper presents a scoping review of 61 publications (2020–2024) to investigate the role of Generative Artificial Intelligence (GenAI) in the Game Development Life Cycle (GDLC). We examine how GenAI techniques including Large Language Models, diffusion models, GANs, and AI-driven procedural content generation are applied across the Pre-Production, Production, and Post-Production stages. Through a systematic search and screening process, we map GenAI applications to key game development activities such as content creation, story generation, game mechanics design, dynamic difficulty adjustment, and testing. Beyond summarizing existing work, we identify structural trends and imbalances, including the strong concentration of GenAI in production-stage content generation and limited adoption in post-production tasks such as debugging and optimization. By providing a lifecycle-oriented analytical framework, this review highlights emerging research opportunities and supports researchers and developers in understanding how Generative AI can enhance game production workflows and player experiences.

Keywords

Generative AI, Game Development, Procedural Content Generation, Dynamic Difficulty Adjustment

1. Introduction

The rise of GenAI has transformed many creative and technical fields, offering new capabilities in content creation, automation, and problem-solving [1, 2]. Advances in deep learning, including Generative Adversarial Networks (GANs), Variational Autoencoders (VAEs), diffusion models, and Transformer-based architectures, have enabled systems that can generate and enhance creative outputs [3]. In game development, these technologies are increasingly used to support procedural content generation (PCG), game mechanics design, dynamic difficulty adjustment (DDA), and story development [4].

GenAI applications in games span several areas. AI systems are used in narrative design to create interactive storytelling experiences [3], while PCG tools generate levels, environments, and characters dynamically [1]. In addition, GenAI models analyze player behavior to improve gameplay balance and adapt difficulty levels [4]. These applications highlight the growing role of AI in supporting different aspects of game development.

Games themselves are complex systems that combine rules, goals, and player interaction. They play an important role in human experience and development [5]. Although defining "games" can be challenging, researchers commonly describe them using elements such as goals, rules, feedback systems, and voluntary participation [6, 7, 8]. These elements are important when considering how Generative AI contributes to game design and development.

TKTP 2026: Annual Doctoral Symposium of Computer Science, 8.—9.6.2026 Turku, Finland
*Corresponding author.
akorsah23@student.oulu.fi (A. Korsah); Juho.E.Mattila@oulu.fi (J. Mattila); Thusitha.Bandaranayake@oulu.fi (S. Bandaranayake); Prabhash.Rathnayake@student.oulu.fi (P. Rathnayake); pasi.karppinen@vub.be (P. Karppinen); iikka.paajala@oulu.fi (I. Paajala)
0009-0000-9425-7525 (A. Korsah); 0000-0003-3278-4789 (J. Mattila); 0009-0005-9776-0827 (S. Bandaranayake); 0009-0007-9894-1579 (P. Rathnayake); 0000-0002-7800-8243 (P. Karppinen); 0000-0002-9732-492X (I. Paajala)
© 2026 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

CEUR Workshop Proceedings ceur-ws.org ISSN 1613-0073
published 2026-11-02

CEUR-WS.org/Vol-4309/paper16.pdf

Recent studies have explored the use of GenAI in games. For example, GPT for Games: A Scoping Review (2020–2023) [9] focuses on GPT-based models, particularly for narrative and gameplay interaction. However, broader GenAI approaches including diffusion models, GANs, reinforcement learning–based PCG systems, and multi-modal pipelines have not been systematically examined across the full Game Development Life Cycle (GDLC).

In this paper, we examine how GenAI is applied across the three main GDLC stages: Pre-Production, Production, and Post-Production [10]. We extend beyond GPT-focused studies [9] to include a wider range of generative techniques used for ideation, content creation, asset development, testing, and optimization. Using a structured search across six databases, we identified 61 relevant studies published between 2020 and 2024.

Unlike prior work that focuses on specific model types, this study adopts a life-cycle oriented perspective. By mapping GenAI applications to GDLC stages, we identify patterns in how these technologies are used and where gaps exist. In particular, we observe a strong focus on production-stage content generation, with less attention given to post-production tasks such as testing and debugging.

This paper provides (1) a mapping of GenAI techniques across GDLC stages, (2) an analysis of trends and gaps across stages, and (3) insights into the gap between academic research and industry practice. Through this life-cycle based approach, we aim to help researchers and developers better understand current applications of GenAI and identify areas for future development.

2. Background

GenAI refers to machine learning–based systems that generate content such as text, images, audio, code, and structured data [11, 12]. Recent advances in deep learning, including transformer-based architectures, diffusion models, and adversarial frameworks, have expanded these capabilities. In game development, these technologies are used to support creativity, automate content creation, and improve workflows [3].

AI has long played a role in game development, evolving from rule-based systems and search algorithms to machine learning approaches [10, 13]. These methods have supported both gameplay behavior and content generation. However, modern generative models now enable more coherent, context-aware, and multi-modal outputs.

2.1. Game Development Life Cycle

Game development follows a structured process similar to traditional software development but involves multiple disciplines such as visual design, audio, narrative, and interaction [10, 13]. Although different GDLC models exist [14, 15], most define three main stages: Pre-Production, Production, and Post-Production [10].

The **Pre-Production stage** focuses on planning and concept development [10]. It includes ideation, defining gameplay mechanics, identifying the target audience, and selecting platforms [16]. Designers create narratives and characters, while artists develop concept art [17]. Technical decisions such as game engines and tools are also made. A key output is the Game Design Document (GDD), which outlines gameplay, narrative, and technical requirements [18], along with early prototypes.

The **Production stage** involves building the game. This includes asset creation, programming, level design, animation, sound, and AI integration [13]. Teams collaborate to implement systems such as physics and NPC behavior [10]. This stage is resource-intensive and includes continuous testing to ensure stability.

The **Post-Production stage** focuses on refinement and release preparation. It includes debugging, quality assurance, balancing, optimization, and localization [15, 18]. Playtesting improves usability and gameplay, and after release, updates and downloadable content (DLC) may be added [16].

2.2. Evolution of GenAI & LLM

Generative approaches in games existed before modern AI. Procedural Content Generation (PCG), introduced in the 1980s, enabled automatic creation of game content such as levels and environments [19]. PCG is not a single technique but a broad problem area that includes rule-based, search-based, and machine learning approaches. Games like Rogue, Diablo, and Daggerfall used these methods to create large environments [20].

By the mid-2000s, machine learning and neural networks were applied to content generation and player modeling, enabling systems to adapt based on gameplay data [21].

The introduction of GANs in 2014 [22] improved image and texture generation, influencing asset creation in games [23, 24].

More recently, Large Language Models (LLMs), such as GPT-based systems, have improved text generation for dialogue, storytelling, and quest design [25]. Their ability to produce coherent responses makes them useful for NPC interactions [26].

Diffusion models have further advanced image generation, enabling high-quality text-to-image synthesis [27]. These models are now used for concept art, character design, and environment prototyping.

3. Methodology

This study adopts a scoping review methodology to systematically map the current landscape of GenAI applications within the GDLC. The review follows the PRISMA-ScR (Preferred Reporting Items for Systematic Reviews and Meta-Analyses extension for Scoping Reviews) guidelines to ensure transparency and reproducibility.

The review is guided by the following research questions:

- RQ1 : How is Generative AI applied across the stages of the Game Development Life Cycle?;
- RQ2 : Which categories of Generative AI methods are most commonly used in each GDLC stage?;
- RQ3 : What research gaps and emerging opportunities can be identified in the current landscape?.

3.1. Search Strategy

We conducted research across six major academic databases: ACM Digital Library, IEEE Xplore, Scopus, Science Direct (Elsevier), ProQuest and Web of Science. These databases were selected due to their extensive coverage of computer science, artificial intelligence, and game development research.

Full-text searches were conducted using combinations of the following keywords: (”Generative AI” OR ”Generative Artificial Intelligence” OR ”Large Language Model” OR ”LLM” OR ”Diffusion Model” OR ”GAN” OR ”Procedural Content Generation”) AND (”Game Development” OR ”Game Design” OR ”Video Game”).

The search covered publications from January 2020 to December 2024. While some AI techniques predate 2020, the selected timeframe captures the surge of modern GenAI technologies, particularly transformer-based LLMs and diffusion models with the following criteria of selection.

Inclusion Criteria

Studies were included if they:

- Utilized a Generative AI technique (e.g., LLMs, diffusion models, GANs, reinforcement learning-based generative systems, PCG with ML components).;
- Addressed at least one aspect of the Game Development Life Cycle.;
- Generated or supported the creation of game-related artifacts (e.g., narrative, levels, characters, assets, testing frameworks).

Exclusion Criteria

Studies were excluded if:

- The term “game” was used metaphorically (e.g., game theory, serious simulations unrelated to entertainment games).;
- The work focused solely on AI gameplay without generative components.;
- The generative system was not applied within a game development context.

3.2. Screening Process

The initial search yielded 1,240 records. After removing 308 duplicates, 932 articles remained for title and abstract screening. Following eligibility assessment and full-text review, 61 studies met all inclusion criteria. Disagreements during screening were resolved through discussion among three researchers to reduce classification bias.

An open-coding approach was used to identify recurring themes. Coding was initially conducted by one researcher and subsequently validated by two additional researchers to improve reliability.

Our studies were categorized into three GDLC stages:

- `Pre-Production:` Ideation, concept development, narrative planning, asset prototyping.;
- `Production:` Asset implementation, level design, NPC behavior, dialogue generation, gameplay systems.;
- `Post-Production:` Testing, debugging, optimization, balancing, localization, post-launch refinement.

In cases where studies spanned multiple stages, classification was based on the primary intended contribution described in the paper. This framework is intended as a functional abstraction rather than a rigid industrial taxonomy.

4. Results

This section presents the findings of the scoping review, synthesizing 61 studies published between 2020 and 2024. Studies were mapped to stages of the GDLC: Pre-Production, Production, and Post-Production. Classification was based on the primary contribution made in each paper, although some works spanned multiple stages.

4.1. Descriptive Characteristics of Articles

The overall distribution of studies shows a clear imbalance with most publications concentrate on Production stage applications, while comparatively fewer studies address Pre-Production ideation tasks or Post-Production testing and optimization. Figure 1 below shows the distribution of number of coded references per GDLC stage.

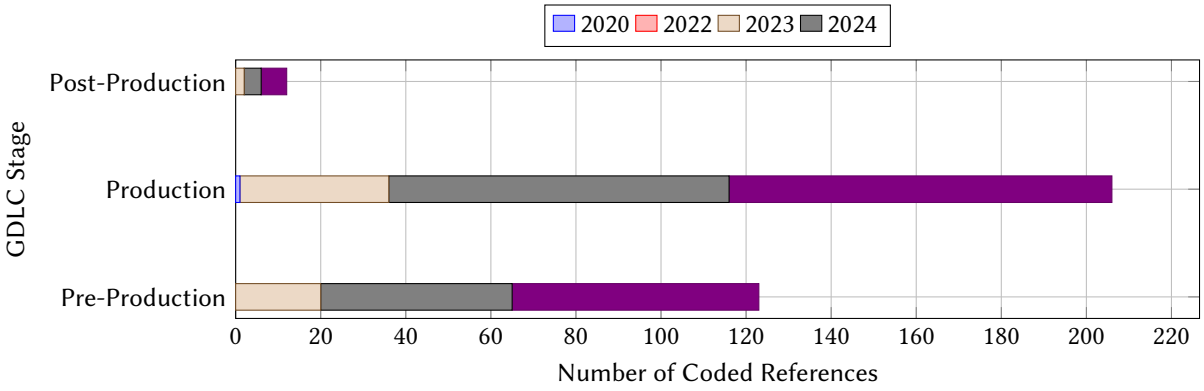


Figure 1: Distribution of Codes for GDLC stages per Publication Year

The highest number of codes in the Production phase with 206 coded references suggests that the application of AI is most concentrated during the active development of game assets, mechanics, and systems. This aligns with existing literature indicating that AI-driven tools, such as PCG and game mechanics optimization, play a critical role in streamlining production workflows and enhancing game design efficiency [13].

The Pre-Production phase also shows substantial engagement with Generative AI, with 123 coded references indicating the use of GenAI in the early-stage tasks such as concept ideation, narrative generation, and asset creation design. Conversely, the Post-Production phase has the fewest coded references (12), highlighting a relatively lower emphasis on AI applications in this stage. This may be because post-production primarily focuses on quality assurance, debugging, and final optimizations, where AI tools may be used less frequently compared to earlier stages. However, the presence of AI in this phase suggests some emerging applications, such as AI-driven testing, performance optimization, and content adaptation for different player demographics.

It is observed in table 1 below that the publication trends in Generative AI research for game development indicate a steady increase in academic interest over the years, culminating in a significant surge in 2024. Initially, research in this domain was sparse, with only one publication in 2020 and a slight increase to two studies in 2022. However, by 2023, the number of publications had risen sharply to 20, reflecting a growing recognition of Generative AI's potential in the gaming industry. The most notable jump occurred in 2024, with 38 publications, highlighting the widespread adoption of Generative AI-driven techniques in game design and development.

Table 1
Number of Publications per Year

Publication Year	Count of Journal/Conference
2020	1
2021	0
2022	2
2023	20
2024	38

This pattern suggests that advancements in AI technologies, particularly in LLMs and PCG, have fueled increased academic and industry engagement.

4.2. Generative AI used in the Pre-Production Stage

The analysis showed that GenAI models like LLMs for narrative and quest design, diffusion models like Stable Diffusion (SD) and Stable Diffusion XL (SDXL) for the facilitating of concept art and game asset generation, PCG techniques for dynamically creating environments, assets, and narratives, collectively enhancing creativity, efficiency, and iterative design in the pre-production stage of development.

We further noticed that one of the most prominent applications of GenAI in this stage is the use of LLMs for narrative and quest design. [28, 29, 30, 31] explored how an LLM-powered assistant supports game designers in structuring plots and developing rich, interactive narratives. [32, 33, 34, 35, 36, 37] also mentioned the application of GPT-based models in procedural storytelling making storyline dynamic for player immersion.

A significant body of research explored the use of SD in graphics generation for games. [38] emphasized how SD processes text descriptions to generate highly detailed and immersive game visuals making it more flexible making was for fine-tuning. In the same way [39] explores SD's integration with ControlNet, a plugin that enhances pose control and detail refinement planning in generated images. Their work showcases how documentations are made for the planning of models to fine-tune game asset generation, including cutscenes, mood boards, and promotional artwork. [40] further investigated how SD is leveraged to create interactive game elements, such as traps, treasure chests, and environmental objects, adding variety and uniqueness to procedurally generated content ideation. Also, [41] highlighted taking a narrative of how text-to-image models like SDXL with the combination

of LLMs will create cohesive visual assets that align. [42] also noted that SD, alongside Novel AI and Midjourney, serves as a creative assistant, helping developers with prototype building and refining visual concepts more efficiently.

Another Generative AI model used at the pre-production stage is PCG which aids developers with efficiency in building prototypes and generating content automatically following laid down procedures thereby reducing manual workload and enhancing creativity [43, 44]. From their research, PCG is leveraged in early-stage design to generate game environments, assets, and narrative structures dynamically. This allows developers to experiment with different gameplay mechanics before full-scale production [45, 46, 47]. For example, PCG techniques are used to create diverse level layouts and environmental assets, providing a foundation for iterative design and testing [48, 49]. Additionally, some researchers have explored the application of AI-driven PCG to assist in game story development, ensuring narrative coherence while maintaining design flexibility [50, 51].

The analysis further revealed that Generative AI is also used in narrative planning and story generation, character and NPC development planning, interactive dialogue design creation, and AI-driven planning for PCG. For instance, [52] described an LLM-based narrative planner that translates a writer's abstract storytelling goals into concrete character action sequences. [53] also demonstrated the use of AI in generating coherent game stories while integrating thematic elements, ensuring that game narratives remain logically structured. Additionally, [29] introduced GamePlot1, an AI-powered tool that assists developers in early-stage game development by automating the creation of narrative elements and plotlines. In [52], an LLM-based Character Simulator that autonomously generates a prototype of character actions to ensure NPCs behave in ways that align with the game's storytelling objectives was utilized.

Generative AI is also proving useful in game planning particularly in defining game rules, environments, and quests. AI-driven systems allow designers to input high-level criteria, and the AI generates detailed game rules and mechanics accordingly [36]. [54] demonstrated that AI-generated quests are often more detailed, creative, and immersive than those manually written, suggesting generative AI enhances both efficiency and quality in quest design. In [39], AnimGPT, an AI model for character animation, assists in the early-stage planning of character movements. Their findings indicate that AI is streamlining pre-production workflows by automating complex design tasks that previously required significant manual effort.

4.3. Generative AI used in the Production Stage

The research by [49] emphasized that Large Language Models are integral to procedural content generation, particularly in game level design, where the novelty, playability, and diversity of generated levels were accessed. The paper illustrated this by computing a model's performance through the proportion of levels that meet the criteria of novelty, playability, and diversity. The authors of [50, 31, 33, 48, 52, 55, 56, 57, 58, 59] discussed how generative AI like OpenAI's GPT-4 Turbo and GPT-3.5 Turbo are utilized in developing game content. The work from [60] utilized MarioGPT to incorporate reinforcement learning principles to iteratively improve game content by using this Language model to fine-tune level structures dynamically, to ensure reward mechanisms maintainability to engaged players. The findings highlight MarioGPT' s ability to mutate and refine levels to ensure that generated content remains engaging and functional by using BERT for inpainting enhances the seamless integration of newly generated level sections.

Again,[61, 62] noted that pre-trained LLMs such as BERT perform comparably to specialized models like block2vec in level generation tasks, offering the added advantage of natural language-based editing without additional training. Similarly, [63, 64, 65, 35] explored how OpenAI's GPT-3 and its variants, such as text-davinci-003 and text-davinci-002, demonstrate the ability to generate playable and novel levels even from minimal data inputs. Kumaran et al. [55] described an Interactive Dialogue Generator that uses LLMs to produce multiple emotion-driven dialogue variations, allowing NPCs to react naturally to player choices.[34, 36, 51, 52, 63, 66, 67] leveraged GPT-4 to generate real-time NPC responses, ensuring unscripted yet coherent conversations. [68, 30] further emphasized that AI-driven dialogue generation

improves NPC realism, making in-game interactions more lifelike as NPCs can communicate with players and acquire environmental information, making them more responsive and adaptable.

Lastly,[38, 41, 42, 67, 68] discussed the role of Stable Diffusion as an open-source Latent Diffusion Model (LDM) that enhances the plan for creation of game assets, including character concepting, textures, and environmental elements.

4.4. Generative AI used in the Post-Production Stage

The work of [32] used GPT-4 to evaluate the quality of game dialogues using criteria such as how well they match characters and scenes, dialogue flow engagement showing how GenAI can support dialogue evaluation in a more consistent way.

Generative AI is also useful for checking whether game logic works correctly. [28] applied language models to validate text-based game logic using rules like logical consistency and correct structure. This helps developers detect and fix errors in Virtual Game Description Language (VGDL) scripts more efficiently, reducing the need for manual debugging. Beyond validation, AI is also applied in code review. [69] introduced an AI-based code reviewer that checks game code for inconsistencies, hallucinations, and deviations from design requirements. This agent not only flags potential errors but also offers recommendations for refinement, ensuring that game code remains both efficient and maintainable. Similarly, [62] proposed an AI-assisted method to track changes that cause test failures, allowing developers to identify problems earlier in the development process.

Another important application is automated bug detection. [65] showed that large language models can analyze descriptions of game events to detect potential bugs, achieving up to 70.66 percent accuracy. This suggests that AI can support debugging and improve game stability. In addition, [59] explored AI-generated voice interactions, showing how generated voice messages can enhance player immersion and potentially adapt to player actions.

Overall, these studies show that Generative AI can support post-production tasks such as dialogue evaluation, logic validation, code review, and bug detection. While these tools improve efficiency and consistency, human involvement remains important to ensure reliability and quality.

4.5. Cross-Stage Patterns and Observations

Synthesizing findings across GDLC stages reveals three key patterns. First, there is a strong concentration of research in Production-stage content generation, particularly in procedural level design and dialogue systems. Pre-Production and Post-Production tasks receive comparatively less attention. Again, LLMs dominate narrative and dialogue tasks, while diffusion and GAN-based systems are primarily used for visual asset generation. Machine learning enhanced PCG approaches span multiple stages but are most prevalent in Production contexts. Lastly, evaluation practices across stages remain largely academic. Most studies rely on controlled experiments, small-scale user evaluations, or prototype demonstrations. Evidence of long-term deployment in commercial game studios is limited. This suggests a translational gap between research innovation and industry adoption.

Across all stages, GenAI is most frequently positioned as a supportive tool within existing workflows rather than as a fully autonomous system. While the technological capabilities of GenAI have advanced rapidly, systematic integration into end-to-end development pipelines remains an open challenge.

5. Discussions

The findings show that Generative AI is explored across all three stages of the GDLC, but its integration remains uneven and task-specific rather than fully integrated. While GenAI is used in Pre-Production, Production, and Post-Production, the level of adoption differs across stages.

In the Pre-Production stage, GenAI mainly supports ideation and early creative work. Studies highlight its use in concept development [31, 70], narrative and script generation [29, 36] and character and environment design [29, 63]. These tools help speed up brainstorming and prototyping [42].

However, human oversight is still needed to refine outputs, ensure originality, and maintain coherence [28]. GenAI is therefore used as a support tool rather than a replacement for designers.

The Production stage has the highest research activity. Many studies focus on PCG supported by machine learning. GenAI is used for level generation, dialogue systems, and asset creation. Although these tools reduce manual effort, they are usually applied to specific tasks rather than full systems. Human involvement remains important to maintain artistic quality and "naturalness" [71]. This shows that while GenAI improves scalability and variation, final decisions are still human-driven.

Post-Production applications are more limited. Some studies explore automated playtesting, balancing, and adaptive difficulty, but fewer address debugging, quality assurance, and optimization. Human roles are especially important in this stage for ensuring stability and validating player experience. This suggests a gap in research, where more work is needed on automated debugging, testing, and optimization using GenAI.

Across all stages, several strengths and limitations are observed. GenAI performs well in generating narrative elements and procedural world-building. However, challenges remain, including poor narrative coherence [58], functional errors such as unintended player actions [45], and technical limitations in procedural systems. These issues show that while generated content may appear valid, ensuring consistency and correct behavior is still difficult.

Other challenges include the need for human refinement [42], logical inconsistencies due to weak validation in PCG systems [28], and system limitations such as server load and bandwidth [72]. These highlight the need for better control, validation, and integration with gameplay requirements.

Overall, GenAI is mostly used for isolated tasks rather than integrated workflows. While tools like dialogue generators and level creators show promise, few studies present full end-to-end systems. Concerns about consistency and reliability including hallucinations and prompt misinterpretation further limit adoption in complex environments [51].

In summary, Generative AI in game development is still at an experimental stage, mainly supporting specific tasks rather than transforming entire workflows. Future progress depends on improving reliability, validation, and integration across the GDLC to enable more practical and scalable use.

6. Conclusions and Future Research Directions

This scoping review examined 61 studies published between 2020 and 2024 to understand how GenAI is applied across the game development life cycle (GDLC). Using a life-cycle analytical framework, it mapped techniques like LLMs, diffusion models, GANs, and machine learning-enhanced procedural content generation across Pre-Production, Production, and Post-Production stages.

While GenAI is explored across all three stages, adoption remains uneven. Research is heavily concentrated in Production — particularly level generation, NPC dialogue, and asset creation. Pre-Production uses GenAI mainly to support creativity, while Post-Production tasks like testing, debugging, and optimization remain largely underexplored. This reflects a stronger focus on content generation than full life-cycle integration.

GenAI largely functions as an assistive tool, not a developer replacement. Human involvement remains essential for refining outputs, maintaining quality, and ensuring gameplay logic. Most studies are experimental prototypes, highlighting a gap between academic research and real-world industry application.

Key challenges include narrative coherence, controllability, output variability, and hallucination — all limiting reliability in complex workflows.

Future research should prioritize integrated multi-stage frameworks, interoperable pipelines, and standardized evaluation benchmarks. Greater academia-industry collaboration is also needed. GenAI holds strong potential for game development, but reliable, fully integrated solutions are still required.

Declaration on Generative AI

The authors used Claude during the preparation of this manuscript to assist with grammar correction, language refinement, and improvement of the academic writing style. All AI-assisted outputs were carefully reviewed, edited, and validated by all authors, who take full responsibility for the content and final version of the manuscript.

References

[1] I. J. Colado, V. M. Colado, A. C. Morata, R. S. C. Píriz, B. F. Manjón, Using new ai-driven techniques to ease serious games authoring, in: Proceedings - Frontiers in Education Conference, FIE, Institute of Electrical and Electronics Engineers Inc., 2023. doi:10.1109/FIE58773.2023.10343021.

[2] C. Hu, Y. Zhao, J. Liu, Game generation via large language models (2024). URL: http://arxiv.org/abs/2404.08706.

[3] J. Li, Q. Liu, Application of generative artificial intelligence aigc technology under neural network algorithm in game character art design, Journal of the Knowledge Economy (2024). doi:10.1007/s13132-024-02152-z.

[4] E. A. Romero-Mendez, P. C. Santana-Mancilla, M. Garcia-Ruiz, O. A. Montesinos-López, L. E. Anido-Rifón, The use of deep learning to improve player engagement in a video game through a dynamic difficulty adjustment based on skills classification, Applied Sciences (Switzerland) 13 (2023). doi:10.3390/app13148249.

[5] H. Hildmann, J. Hildmann, Origin of Games, Springer International Publishing, 2018, pp. 1–7. URL: https://doi.org/10.1007/978-3-319-08234-9_231-1. doi:10.1007/978-3-319-08234-9_231-1.

[6] K. Amr, Learning through Games: Essential Features of an Educational Learning through Games: Essential Features of an Educational Game Game, Technical Report, 2012. URL: https://surface.syr.edu/idde_etd.

[7] J. McGonigal, Reality Is Broken: Why Games Make Us Better and How They Can Change the World, Penguin Group , The, 2011.

[8] E. Adams, J. Dormans, Game mechanics : advanced game design, New Riders, 2012.

[9] D. Yang, E. Kleinman, C. Harteveld, Gpt for games: A scoping review (2020-2023), in: IEEE Conference on Computatonal Intelligence and Games, CIG, IEEE Computer Society, 2024. doi:10.1109/CoG60054.2024.10645548.

[10] S. Aleem, L. F. Capretz, F. Ahmed, A digital game maturity model (dgmm), ScienceDirect (Software Process Improvement and Capability dEtermination) SPICE 2016 (2016) 1–15. URL: https://doi.org, also available via arXiv:1711.08526.

[11] G. Cooper, Examining science education in chatgpt: An exploratory study of generative artificial intelligence, Journal of Science Education and Technology 32 (2023) 444–452. doi:10.1007/s10956-023-10039-y.

[12] S. Mondal, S. Das, V. G. Vrana, How to bell the cat? a theoretical review of generative artificial intelligence towards digital disruption in all walks of life, 2023. doi:10.3390/technologies11020044.

[13] A. Z. Aktasx, E. Orcxun, A survey of computer game development, Journal of Defense Modeling and Simulation 13 (2016) 239–251. doi:10.1177/1548512914554405.

[14] J. Schell, The Art of Game Design: A book of lenses, 2008. doi:10.1201/9780080919171.

[15] S. Aleem, L. F. Capretz, F. Ahmed, A digital game maturity model (dgmm), Entertainment Computing 17 (2016) 55–73. doi:10.1016/j.entcom.2016.08.004.

[16] G. Ace., Ai in game development: From code to creativity, 2023.

[17] Y. C. Oktaviani, Y. Wahyuningsih, R. P. Kristianto, Application of the game development life cycle (gdlc) model in design and development of educational games for dyslexic children, Social Science and Humanities Journal 9 (2025) 6477–6487. URL: http://www.sshjournal.com/index.php/sshj/article/view/1442. doi:10.18535/sshj.v9i01.1442.

[18] L. Husniah, B. F. Pratama, H. Wibowo, Gamification and gdlc (game development life cycle) appli-

cation for designing the sumbawa folklore game "the legend of tanjung menangis (crying cape)", Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control (2018) 351–358. doi:10.22219/kinetik.v3i4.721.

[19] A. Zafar, H. Mujtaba, M. O. Beg, Search-based procedural content generation for gvg-lg, Applied Soft Computing Journal 86 (2020). doi:10.1016/j.asoc.2019.105909.

[20] L. Cardamone, P. L. Lanzi, D. Loiacono, Trackgen: An interactive track generator for torcs and speed-dreams, Applied Soft Computing 28 (2015) 550–558. doi:10.1016/j.asoc.2014.11.010.

[21] J. Togelius, G. Yannakakis, K. Stanley, C. Browne, Search-based procedural content generation: A taxonomy and survey, Computational Intelligence and AI in Games, IEEE Transactions on 3 (2011) 172 – 186. doi:10.1109/TCIAIG.2011.2148116.

[22] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial networks, Advances in Neural Information Processing Systems 3 (2014). doi:10.1145/3422622.

[23] T. Thompson, The Changing Landscape of AI for Game Development. In Game AI Uncovered, volume 1, 2024.

[24] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial networks, Communications of the ACM 63 (2020) 139–144. doi:10.1145/3422622.

[25] A. Barcaui, A. Monat, Who is better in project planning?generative artificial intelligence or project managers?, Project Leadership and Society 4 (2023). doi:10.1016/j.plas.2023.100101.

[26] S. Siddiqi, V. Shukla, A. Bhardwaj, D. Gaur, Analyzing psychological gamers profile through progressive gaming and artificial intelligence, 2021, pp. 1–5. doi:10.1109/ICRITO51393.2021.9596185.

[27] A. Ghosh, P. Maske, V. Patel, J. Dubey, K. Aniket, R. Srivastava, Theranostic applications of peptide-based nanoformulations for growth factor defective cancers, International Journal of Biological Macromolecules 260 (2024). doi:10.1016/j.ijbiomac.2023.129151.

[28] S. Hu, Z. Huang, C. Hu, J. Liu, 3d building generation in minecraft via large language models (2024). URL: http://arxiv.org/abs/2406.08751.

[29] S. H. Alavi, W. Xu, N. Jojic, D. Kennett, R. T. Ng, S. Rao, H. Zhang, B. Dolan, V. Shwartz, Game plot design with an llm-powered assistant: An empirical study with game designers (2024). URL: http://arxiv.org/abs/2411.02714.

[30] O. Zhukova, V. Mandragelia, V. Chepurna, L. Ivanenko, M. Noskova, A model of using digital information systems to create video game contexts: The case of gpt models and its effect, Ingenierie des Systemes d'Information 29 (2024) 659–667. doi:10.18280/isi.290227.

[31] R. Panchanadikar, G. Freeman, L. Li, K. Schulenberg, Y. Hu, "a new golden era" or "slap comps": How non-profit driven indie game developers perceive the emerging role of generative ai in game development, in: Conference on Human Factors in Computing Systems - Proceedings, Association for Computing Machinery, 2024. doi:10.1145/3613905.3650845.

[32] B. LIAO, D. V. Vargas, Towards immersive computational storytelling:card-framework for enhanced persona-driven dialogues, 2023. URL: https://www.techrxiv.org/doi/full/10.36227/techrxiv.24605667.v1. doi:10.36227/techrxiv.24605667.v1.

[33] J. Leandro, S. Rao, M. Xu, W. Xu, N. Jojic, C. Brockett, B. Dolan, Geneva: Generating and visualizing branching narratives using llms, in: IEEE Conference on Computatonal Intelligence and Games, CIG, IEEE Computer Society, 2024. doi:10.1109/CoG60054.2024.10645625.

[34] S. Rao, W. Xu, M. Xu, J. Leandro, K. Lobb, G. DesGarennes, C. Brockett, B. Dolan, Collaborative quest completion with llm-driven non-player characters in minecraft (2024). URL: http://arxiv.org/abs/2407.03460.

[35] P. Taveekitworachai, F. Abdullah, M. F. Dewantoro, R. Thawonmas, J. Togelius, J. Renz, Chatgpt4pcg competition: Character-like level generation for science birds (2023). URL: http://arxiv.org/abs/2303.15662.

[36] S. Buongiorno, J. Klinkert, Z. Zhaung, T. Chawla, C. Clark, PANGeA: Procedural Artificial Narrative using Generative AI for Turn-Based, Role-Playing Video Games, Technical Report, 2023.

[37] X. Zhang, F. Wan, K. Zhang, H. Jiang, Possible applications of language models in visual novel, in: Proceedings - 2023 International Conference on Educational Knowledge and Informatization, EKI 2023, Institute of Electrical and Electronics Engineers Inc., 2023, pp. 60–63. doi:10.1109/EKI61071.2023.00021.

[38] Q. Sun, Q. Luo, Y. Ni, H. Mi, Text2ac: A framework for game-ready 2d agent character(ac) generation from natural language, in: Conference on Human Factors in Computing Systems - Proceedings, Association for Computing Machinery, 2024. doi:10.1145/3613905.3651049.

[39] V. C. Lungu-Stan, I. G. Mocanu, 3d character animation and asset generation using deep learning, Applied Sciences (Switzerland) 14 (2024). doi:10.3390/app14167234.

[40] R. Gallotta, A. Liapis, G. N. Yannakakis, Llmaker: A game level design interface using (only) natural language, 2024 IEEE Conference on Games (2024). URL: http://arxiv.org/abs/2402.18659. doi:10.1109/CoG60054.2024.10645626.

[41] M. Zammit, A. Liapis, G. N. Yannakakis, Crawllm: Theming games with large language models, in: IEEE Conference on Games (CoG), 2024, pp. 1–2.

[42] J. Lee, S.-Y. Eom, J. Lee, Empowering game designers with generative ai, IADIS International Journal on Computer Science and Information Systems 18 (2023) 213–230.

[43] T. Merino, M. Charity, J. Togelius, Interactive latent variable evolution for the generation of minecraft structures, in: ACM International Conference Proceeding Series, Association for Computing Machinery, 2023. doi:10.1145/3582437.3587208.

[44] M. U. Nasir, J. Togelius, Practical pcg through large language models (2023). URL: http://arxiv.org/abs/2305.18243.

[45] A. Anjum, Y. Li, N. Law, M. Charity, J. Togelius, The ink splotch effect: A case study on chatgpt as a co-creative game designer, in: ACM International Conference Proceeding Series, Association for Computing Machinery, 2024. doi:10.1145/3582437.3587208.

[46] B. Bateni, J. Whitehead, Language-driven play: Large language models as game-playing agents in slay the spire, in: ACM International Conference Proceeding Series, Association for Computing Machinery, 2024. doi:10.1145/3649921.3650013.

[47] B. Feng, M. Y. Su, K. Zeng, X. Li, Player-oriented procedural generation: Producing desired game content by natural language, in: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), volume 14730 LNCS, Springer Science and Business Media Deutschland GmbH, 2024, pp. 260–274. doi:10.1007/978-3-031-60692-2_18.

[48] B. Ngaw, G. Jena, J. Sedoc, A. Normoyle, Towards authoring open-ended behaviors for narrative puzzle games with large language model support, in: ACM International Conference Proceeding Series, Association for Computing Machinery, 2024. doi:10.1145/3649921.3656975.

[49] G. Todd, S. Earle, M. U. Nasir, M. C. Green, J. Togelius, Level generation through large language models, in: ACM International Conference Proceeding Series, Association for Computing Machinery, 2023. doi:10.1145/3582437.3587211.

[50] I.-C. Baek, T.-H. Park, J.-H. Noh, C.-M. Bae, K.-J. Kim, Chatpcg: Large language model-driven reward design for procedural content generation (2024). URL: http://arxiv.org/abs/2406.11875.

[51] L. D. M. García, C. Mangiron, Exploring the potential of gpt-4 as an interactive transcreation assistant in game localisation: A case study on the translation of pokémon names, Perspectives: Studies in Translation Theory and Practice (2024). doi:10.1080/0907676X.2024.2378346.

[52] Y. Wang, Q. Zhou, D. Ledo, Storyverse: Towards co-authoring dynamic plot with llm-based character simulation via narrative planning (2024). URL: http://arxiv.org/abs/2405.13042http://dx.doi.org/10.1145/3649921.3656987. doi:10.1145/3649921.3656987.

[53] P. Taveekitworachai, F. Abdullah, M. C. Gursesli, M. F. Dewantoro, S. Chen, A. Lanata, A. Guazzini, R. Thawonmas, Breaking bad: Unraveling influences and risks of user inputs to chatgpt for game story generation, in: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), volume 14384 LNCS, Springer Science and Business Media Deutschland GmbH, 2023, pp. 285–296. doi:10.1007/978-3-031-47658-7_27.

[54] T. Ashby, B. K. Webb, G. Knapp, J. Searle, N. Fulda, Personalized quest and dialogue generation in role-playing games: A knowledge graph- and language model-based approach, in: Conference on Human Factors in Computing Systems - Proceedings, Association for Computing Machinery, 2023. doi:10.1145/3544548.3581441.

[55] V. Kumaran, J. Rowe, B. Mott, J. Lester, SCENECRAFT: Automating Interactive Narrative Scene Generation in Digital Games with Large Language Models, Technical Report, 2023. URL: www.aaai.org.

[56] L. Ling, X. Chen, R. Wen, T. J.-J. Li, R. LC, Sketchar: Supporting character design and illustration prototyping using generative ai, Proceedings of the ACM on Human-Computer Interaction 8 (2024) 1–28. URL: https://dl.acm.org/doi/10.1145/3677102. doi:10.1145/3677102.

[57] M. C. Gursesli, P. Taveekitworachai, F. Abdullah, M. F. Dewantoro, A. Lanata, A. Guazzini, V. K. Lê, A. Villars, R. Thawonmas, The chronicles of chatgpt: Generating and evaluating visual novel narratives on climate change through chatgpt, in: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), volume 14384 LNCS, Springer Science and Business Media Deutschland GmbH, 2023, pp. 181–194. doi:10.1007/978-3-031-47658-7_16.

[58] J. Freiknecht, W. Effelsberg, Procedural generation of interactive stories using language models (2020). URL: https://doi.org/10.1145/3402942.3409599. doi:10.1145/3402942.

[59] T. Rist, Using a large language model to turn explorations of virtual 3d-worlds into interactive narrative experiences, in: IEEE Conference on Computatonal Intelligence and Games, CIG, IEEE Computer Society, 2024. doi:10.1109/CoG60054.2024.10645583.

[60] S. Sudhakaran, M. González-Duque, M. Freiberger, C. Glanois, E. Najarro, S. Risi, MarioGPT: Open-Ended Text2Level Generation through Large Language Models, Technical Report, 2023. URL: https://github.com/shyamsn97/mario-gpt.

[61] M. Awiszus, F. Schubert, B. Rosenhahn, Wor(l)d-gan: Toward natural-language-based pcg in minecraft, IEEE Transactions on Games 15 (2023) 182–192. doi:10.1109/TG.2022.3153206.

[62] L. Marini, L. Gisslén, A. Sestini, Leveraging large language models for efficient failure analysis in game development (2024). URL: http://arxiv.org/abs/2406.07084.

[63] L. J. Klinkert, S. Buongiorno, C. Clark, Driving generative agents with their personality (2024). URL: http://arxiv.org/abs/2402.14879.

[64] Q. C. Gao, A. Emami, The Turing Quest: Can Transformers Make Good NPCs?, Technical Report, 2023.

[65] M. R. Taesiri, F. Macklon, Y. Wang, H. Shen, C.-P. Bezemer, Large language models are pretty good zero-shot video game bug detectors (2022). URL: http://arxiv.org/abs/2210.02506.

[66] X. Peng, J. Quaye, S. Rao, W. Xu, P. Botchway, C. Brockett, N. Jojic, G. DesGarennes, K. Lobb, M. Xu, J. Leandro, C. Jin, B. Dolan, Player-driven emergence in llm-driven game narrative (2024). URL: http://arxiv.org/abs/2404.17027.

[67] S. Earle, S. Parajuli, A. Banburski-Fahey, Dreamgarden: A designer assistant for growing games from a single prompt (2024). URL: http://arxiv.org/abs/2410.01791.

[68] Y. Song, K. Wu, J. Ding, Developing an immersive game-based learning platform with generative artificial intelligence and virtual reality technologies – "learningversevr", Computers & Education: X Reality 4 (2024) 100069. doi:10.1016/j.cexr.2024.100069.

[69] D. Chen, H. Wang, Y. Huo, Y. Li, H. Zhang, Gamegpt: Multi-agent collaborative framework for game development (2023). URL: http://arxiv.org/abs/2310.08067.

[70] M. Charity, Y. Bhartia, D. Zhang, A. Khalifa, J. Togelius, A preliminary study on a conceptual game feature generation and recommendation system (2023). URL: http://arxiv.org/abs/2308.13538.

[71] S. Värtinen, P. Hämäläinen, C. Guckelsberger, Generating role-playing game quests with gpt language models, IEEE Transactions on Games 16 (2022) 127–139. doi:10.1109/TG.2022.3228480.

[72] A. Amresh, Integrating Reinforcement AI Into the Design of Educational Games, Technical Report, 2023.