Leveling Up the Playing Field: Exploring the Strengths and Weaknesses of AI-Generated Content in Game Development

Martin Kings and Simon Täcklind

Department of Computer and Systems Sciences

Degree project 15 HE credits
Computer and Systems Sciences
Degree project at the bachelor level
Spring term 2023
Supervisor: Tobias Falk

Abstract

The development of video games is a long and expensive process, and it is not uncommon for studios to require their workers to work overtime to meet deadlines, resulting in stressful work environments and reduced worker performance. Recent advancements in artificial intelligence (AI) research have people excited that perhaps this might change. Perhaps AI could supplement game developers, saving time and resources. A case study was performed to explore the use of AI models Chat-GPT and DALL-E in computer game development, assessing their viability in different computer game development areas such as programming, game development and 2D art. With predetermined criteria, based on feedback from professionals and students within the area of game development, the AI models were tasked with ten separate challenges for each category. The results showed that the AI models are not yet ready to become full-fledged developers. However, they are viable collaborative partners for a wide array of game development tasks.

Keywords: Game Development, AI, ChatGPT, DALL-E, Content Generation, Evaluation

Synopsis

Background

Generative AI has recently gotten a bump in general interest, many usage areas have been found and, in this study, generative AI in computer game development is evaluated. The historical presence of AI within games is touched upon and technical details about two generative AI models are partially covered.

Problem

Game development is a lengthy and expensive process, often constrained by budget, scope, and deadlines. This leads to risks, reduced profits, and the prevalence of harsh work conditions such as crunch time. Limited resources hinder small studios from creating high-quality assets for their games.

Research Question

This research aimed to evaluate the integration of artificial intelligence in different game development domains. The objective was to assess whether AI-generated content could enhance productivity in game design, programming, storytelling, and art creation. The study sought to answer questions regarding the strengths and weaknesses of using ChatGPT and DALL-E for game development tasks, as well as the feasibility of relying primarily on artificial intelligence to generate video game components. The findings of the study could potentially lead to increased profits for game development studios and better work conditions for employees within the industry and lastly less hinders for small development studios in terms of creating high-quality assets for their games.

Method

This study employs a case study research strategy to investigate the capabilities of generative AI models, specifically ChatGPT-4 and DALL-E 2. The models are evaluated based on their performance in solving programming, game design, and 2D visuals problems. The study collects textual and visual data generated by the models through their web interfaces, evaluated from a set of category specific criteria. The evaluation criteria are partially determined through a survey targeting professionals and university students in the field, focusing on relevance, coherence, impact, and sustainability.

Result

The results show that the models perform well in all three categories. However, there are sometimes critical mistakes.

Discussion

Using generative AI to help with the game development process looks promising, but due to there being mistakes the models are not yet ready to be left to their own devices. Developers can utilize the models to generate ideas for inspiration, but critical thinking is still required to evaluate the output. With more specialized data, generative AI could become a valuable collaborative partner in the development process.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

Term	Description
2-Pager	A short two-page document giving a quick overview of a game. The document is used in pitch meetings with investors or publishers.
Computer Assisted Design (CAD)	The use of computers to aid in the creation, modification, analysis, or optimization of a design.
Game design document (GDD)	An internal development document detailing everything there is to know about a game.
Language model (LM)	A probability distribution over words or word sequences. The output will be what was calculated as most probable by the model.
Role-playing game (RPG)	A game genre where the player controls a character, and the character development is one of the most important aspects of the game.
Real-time strategy (RTS) game	A subgenre of strategy games where the players don’t take turns incrementally but strategically plays the game in real-time.
Unreal Engine 5 (UE5)	A game engine, widely popular in 3D-game development.
Likert Scale	A scaling method measuring both positive and negative responses to a statement.
AAA studio	A mid-sized to large game publisher with a solid development budget
IDE	An integrated development environment, a software helping developers in the construction of code.

1 Introduction

In the realm of gaming, artificial intelligence (AI) has traditionally played several different pivotal roles. For example, as integral components of the virtual landscape, fostering player immersion and interactivity, and as algorithmic frameworks devised to create content procedurally. Doing so, abiding by stringent guidelines and rules set forth by game programmers.

A recent study by Xia et al. (2020) underscores the swift progress of AI technology and the ambiguity regarding its future implications. This provokes an interesting inquiry about the potential of AI to play a role in the creative facets of game development.

Although modern AI models such as ChatGPT and DALL-E recently have gained a great deal of popularity and have been adopted as tools in many different development fields, their potency within the realm of game development is yet to be fully explored.

Developing a video game is a long and expensive process. Even indie games can cost up to several million dollars to develop and the most expensive game to be released to this date was Cyberpunk 2077 by CD Project which cost an astounding $174 million accompanied by a $142 million marketing budget. The rushed development cycle of Cyberpunk 2077 resulted in the studio losing $50 million in preorders and saw their share price plummet by 73% (Bramble, 2022) so it comes as no surprise that game development studios want to increase their productivity and cut costs to increase profits and reduce risk. Historically, most game development has followed the project management triangle theory, shown in Figure 1, where the quality of a game is constrained by the budget, deadlines, and scope (features) of the project and changes in one constant will necessitate changes in others or quality will suffer.

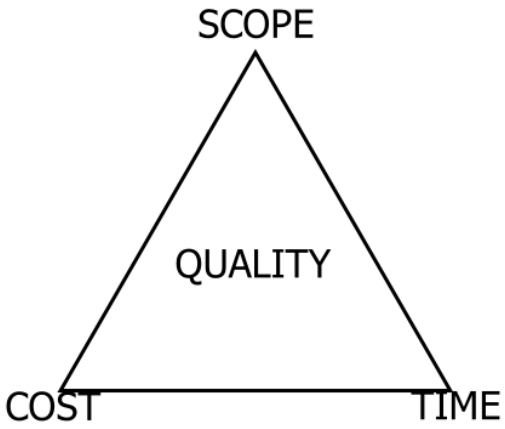

Figure 1 The project management triangle.

If you make a game with a big scope and a small budget it will require a significant amount of time, if you make a game and want it to be released quickly then you need to keep the scope small or have a big budget. But some people, including Jon Lai, believe that generative AI will break this triangle and allow for studios to make games that have a big scope on a low budget, quickly, which in turn could result in larger profits and reduced risk for the studios. (Lai, 2022)

1.1 Thesis Structure

This study focuses on assessing whether AI, specifically ChatGPT and DALL-E, are ready to take a larger role in content generation during the development of video games. This paper first explores the history of AI's role in games and takes a closer look at the models evaluated in the study. Subsequently it describes how the study was executed, what data was collected and how that data was analyzed, as well as alternative methods and strategies that were considered. The paper also contemplates some

ethical considerations regarding the study and AI in general. Finally, the paper presents the results of the study as well as a summary and discussion of the findings.

1.2 Problem

The game development industry is built on harsh release schedules of new titles and updates to grab the consumers' attention. To meet these release schedules while keeping costs low the industry has leveraged compulsory overtime, also known as crunch. This often results in stressful work environments, health issues among workers and the release of unfinished products, and it has gotten worse since the introduction of the microtransaction model ('The Dark Side of the Video Game Industry', 2019). If generative AI can break the triangle theory (Lai, 2022) then the gained productivity could be leveraged by the workers to reduce or eliminate crunch time.

There are many different roles in game development and each role requires very specific knowledge and skills (Davidson, 2017). As an example, generative AI could allow for developers to create higher quality assets for their games that would otherwise require knowledge or skills that they themselves do not possess. This could be especially valuable for small studios.

1.3 Research goal

The objective of this research was to systematically assess the degree to which artificial intelligence could be integrated into various game development fields. The goal was to determine whether AI-generated content can augment the productivity of game developers in areas such as game design, programming game mechanics, crafting engaging narratives and the contribution to the creation of art and textures.

This paper aims to address the following questions:

- What are the strengths and weaknesses in using ChatGPT and DALL-E to complete game development tasks?
- What is the feasibility of generating video game components with a primary reliance on artificial intelligence?

The expectation was that ChatGPT will perform well in the field of Game Design, specifically with generating concepts, as well as writing code for individual scripts and providing rough outlines for more complicated code. Considering that the generation of video game graphics is a very specific field of art generation and due to the lack of specialized training data, the expectation was that DALL-E's performance in generating 'ready-to-use' content such as sprite sheets, UI or textures would not be as satisfactory as ChatGPT is at generating text. Instead, DALL-E was expected to be more suitable for aiding artists with inspiration and images for mood boards.

Due to the difficulty of generating video game graphics, especially animation frames, the study did not expect to find that AI can generate entire video game components, such as functioning NPCs, abilities or levels that would live up to consumer expectations. However, it was expected that AI could generate parts of these components.

1.4 Delimitations

This study has opted to confine its testing to DALL-E and ChatGPT, thereby restricting itself to a few specific aspects of game development and only a few of the AI tools available that could be of interest for this study. Unfortunately, well-established, and publicly available AI models for other game development aspects, such as 3D modeling and sound generation, are scarce, resulting in the exclusion of music, animations, and modeling from this evaluation.

One additional significant limitation the study faced was the inability to fully examine and evaluate the AIs capabilities in cutting-edge technology. This constraint stems from the fact that ChatGPT training data only extends up to September 2021 (OpenAI, 2023a). This has a direct impact on the tools and the resources that could be used for the research. For instance, evaluating ChatGPT's ability to be utilized for development in Unreal Engine 5 (Unreal Engine, (no date)) had to be ruled out, which was officially launched in April 2022. Regrettably, given ChatGPT's knowledge constraints, earlier technology and materials for AI tool validation had to be depended on, leaving the most recent advancements excluded from the research.

This study has chosen to delimit itself to validating ChatGPT's output in Unreal Engine version 4.27.2, which is the last version of Unreal Engine that was released up until the end date of ChatGPT's training data set. The reasoning behind the engine choice was the very well documented interface of Unreal Engine and its vast community of developers. These factors led the study to believe that the ChatGPT model would be sufficiently trained in Unreal Engine C++ programming.

This study is not only constrained in the use of game engines, but also other technologies such as programming languages. This is because it must adhere to technologies whose documentation is prevalent in ChatGPT's training data, disregarding smaller and less adopted technologies with insufficient documentation as valid options for use in the evaluation. Worth mentioning is also that the result of the evaluation is only relevant for a limited time, due to how quickly the subject is advancing and how various industries are adopting the technology. Due to how swiftly AI evolves this study poses a risk of quickly being outdated, and additional research is expected to be required to evaluate future iterations of generative AI.

Lastly, the inability to fully utilize GPT-4's strengths, due to the study's lack of access to the model's API, hinders this study's potential. Multimodal input queries could significantly aid in validating GPT-4's capabilities, and being confined to the web interface represents a potential loss for this research. Consequently, the findings may not fully capture the strengths and potential applications of GPT-4 in the realm of game development and related technologies.

2 Extended Background

In 1956 Allen Newell and J.C. Shaw created Logic Theorist (Gugerty, 2006), a program designed to mimic human problem-solving capabilities by generating new proofs for mathematical theorems. That same year John McCarthy coined the term 'artificial intelligence' during a summer workshop at Dartmouth College (Allganize, 2020) and even though the technology of the time was limited, research in the field began to take shape. Since then, there have been great improvements on generative AI thanks to factors like 'Big Data,' increased processing power, a connected globe through the internet, open-source software, and improved algorithms, we now live in an age that some people dub "the AI revolution" (Data-Driven Science, 2020).

2.1 AI in games

In the early days of AI development during the 1950s, the evolution of early AI programs began to take shape with notable examples such as Nimrod, a computer designed to play the mathematical strategy game Nim and Arthur Samuel's checkers program being among the first instances of AI applied in gaming (Grant and Lardner, 1952) (Schaeffer, 1997). These pioneering AI applications laid the groundwork for future advancements within the realm of AI, particularly in the context of virtual experiences. The subsequent decade saw the emergence of more sophisticated AI algorithms for game characters, as exemplified in games like Pac-Man. However, it was not until the emergence of real-time strategy (RTS) games during the 1980s that the games required more advanced AI to manage computer-controlled units (Xu, 2014).

Throughout the following decades, there was a noticeable increase in the use of AI-driven non-player characters in role-playing games (RPGs) and open-world games. Not only did the technology during this period reach a point where the AI-driven world entities felt more human-like, we also saw ourselves equipped with AI-powered tools for procedural content generation which consequently enabled level designers and game developers with a new and powerful way of creating dynamic and immersive game environments, NPC behaviors and game levels at a much lower time cost.

In recent years, the rise of deep learning and reinforcement learning techniques in video games also has led to groundbreaking AI behaviors with huge accomplishments. For instance, DeepMind's AlphaStar and OpenAI's Dota2-model OpenAI Five who managed to beat the existing Dota 2 world champions in 2019 (OpenAI, 2019). Achievements like these not only demonstrate the growing potential of AI in gaming but also pave the way for future innovations in the ever-evolving world of artificial intelligence.

The significance of AI in gaming dates all the way back to the 1950s (Grant and Lardner, 1952), and since then it has played an essential role in shaping the engaging and immersive experiences enjoyed by billions of people worldwide. Over the years, AI has taken on various forms and functions in games, lately serving as crucial elements within the game environments or acting as efficient tools for games in terms of world building according to rules set up by the developers. Today, with the rapid advancements in AI technology, we are witnessing a new potential path for AI within game development where AI could act more as a creative participant in the game development process, assisting developers in generating ideas, writing game lore, creating game imagery, and resolving technical issues at a much higher pace.

2.2 What is ChatGPT?

ChatGPT is a Language Model (LM) created by OpenAI. Language modeling is the use of statistical data and probabilistic techniques to determine the probability of a given sequence of words occurring in a sentence. ChatGPT was trained to follow an instruction prompt and generate a detailed response by predicting which sequence of words is the most probable response to a given prompt (OpenAI, 2023a). As of this paper being written, the latest version is GPT-4 (OpenAI, 2023a) and it is the model used for this study. It has 100 trillion parameters, roughly 500 times more than its predecessor GPT-3, which makes it the most advanced LM in the world. GPT-3 was capable of zero-shot, one-shot and few-shot learning. In few-shot learning the model is prompted with several examples during training, compared to one-shot learning where it is only given one example. Zero-shot learning is considered unfair as even humans prefer at least one example before attempting a task. The improvement on the size of the model improves the few-shot performance by leaps and bounds (Brown et al, 2020). The model can generate virtually anything that is text based with astounding quality; Fictional stories, historical facts, instructions on how to perform a task and even write functional computer code. In this paper, the focus will be on generating text that can be used in game development tasks.

ChatGPT was trained using Reinforcement Learning from Human Feedback (RLHF), a method that uses human feedback to fine-tune the training of the LM. First the LM is 'pre trained' using a large amount of data (Lambert et al, 2022). This LM was pre trained using the same methods as InstructGPT but with a slight difference in the data collection setup (OpenAI, 2023a). The next step is to train a Reward Model (RM) and it is where new research in RLHF takes place. The goal is to get a RM which takes text input and returns a scalar reward which represents the human preference. Humans rank the behavior of the pretrained LM and then these rankings are used as training data for the RM. The RM is extremely important and will have a huge impact on the result since it will be used to further train the LM. When the LM has been pretrained and a sufficient RM has been trained it is almost time to fine-tune the LM through Reinforcement Learning (RL) using the RM as a reward system. But first the initial parameters of the LM need to be adjusted, this is done by first copying the parameters of the LM to not overwrite the original parameters. The adjustments are made using a policy-gradient RL algorithm, Proximal Policy Optimization (PPO). This paper will not go into details about what a PPO is since it is an entire field of its own, suffice it to say it is a method of fine-tuning the parameters of a LM to make it possible to train it with RL. After the parameters have been adjusted by the PPO you can start the RL process. An input is fed into the original LM as well as the PPO adjusted LM resulting in two outputs. The outputs are fed into the RM which sends a reward based on the output to the PPO for it to improve on the adjusted LM. This process is then repeated a very large number of times (Lambert et al, 2022).

Even though ChatGPT has been rigorously trained and is extremely sophisticated there are still limitations to what it can do. Some examples of limitations are provided by OpenAI (2022):

- Sometimes ChatGPT writes plausible sounding but incorrect or nonsensical answers.
- ChatGPT can yield significantly different results with only small tweaks to the input phrasing or even when attempting the same prompt multiple times.
- The model often overuses certain phrases, which is due to bias in the training data.
- The model does not ask for clarifying questions and instead tries to guess what the user intended.
- Efforts have been made to make the model refuse inappropriate requests, although sometimes responds to harmful instructions or exhibit biased behavior.

ChatGPT is available to the public for testing through the OpenAI website. (OpenAI, 2022)

2.3 What is DALL-E?

DALL-E, a variant of the GPT-3 language model developed by OpenAI, shares similarities with ChatGPT, in its ability to generate output data based on text prompts. However, DALL-E goes a step further by incorporating image input to generate inventive visuals (OpenAI, 2021a). Like ChatGPT, DALL-E is constructed upon the Transformer architecture which was first introduced by Google Brain in 2017 (Vaswani et al., 2017). This enables the model to utilize not only the user input but also each generated output symbol as additional input, one output element at a time. DALL-E underwent pre-training on massive amounts of data comprising text and image pairs. The distinguishing factor between ChatGPT and DALL-E lies in their fine-tuning for specific tasks. DALL-E's fine-tuning allows it to comprehend the connections between text descriptions and visual elements, enabling it to create images based on the given input. Additionally, DALL-E can not only generate images from scratch but also reconstruct parts of images using surrounding image data and additional user text input. The ability of DALL-E to produce a diverse array of images makes it a compelling subject for this research. The aim is to assess its performance in generating various gaming-related visual components, including user interface menu items, model textures, 2D game sprites, background images, and more.

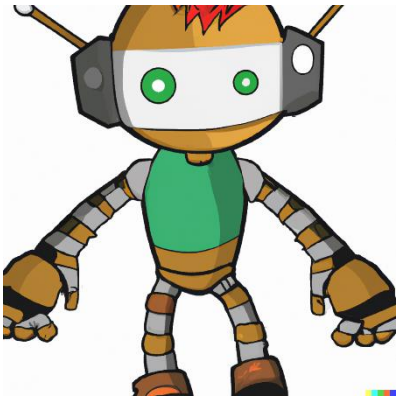

Figure 2 Example output from DALL-E given the prompt "cartoon robot white background".

2.4 Generative AI and Computer-Human collaboration

Advancements in generative AI, more specifically Generative Adversarial Networks (GAN) (Creswell et al, 2018), have resulted in tremendous improvements in computer generated content. This, combined with GPT-4's (OpenAI, 2023a) reasoning and conversational capabilities has generated a lot of speculation about what the future holds. AI not only has the potential to one day revolutionize the gaming industry, but our entire way of life.

For a long time, humans have speculated that eventually our usage of computers will be less like using a tool and more of a collaborative effort. Computer Aided Design (CAD) is a term credited to Douglas Ross, a computer scientist at MIT during the 1950s. It did not take long until the first CAD software was developed to aid in the design processes, reducing the time that humans need to spend on repetitive or time-consuming tasks, allowing for humans to spend more time being creative. Generative AI like ChatGPT or DALL-E could be defined as CAD-tools that humans can use to save time. However, according to their indeterministic nature where the same question can yield different results if repeated, Betti Marenko (Marenko, 2015) would argue that they also can and will join in on the creative aspects of design. If you ask DALL-E to draw you 'a sunflower on a field during sunrise', you will not get the exact image you had in your mind. DALL-E will have added something akin to 'creativity' to the result. This led to the belief that the 'CAD-software' of the future will not be tools used by humans to save time so they can focus on the creative process. Instead, the 'CAD-software' of the future will allow for

a back and forth between humans and computer where they collaborate in the creative process, each leaving their own distinct mark on the resulting design.

2.5 Related research

A highly relevant study on generating code using a GPT model was conducted by Martin Jonsson and Jakob Tholander (Jonsson, M., Tholander J. 2022). The research focused on university students attempting to solve programming tasks using the GPT-3 model Codex, which was developed by OpenAI and is more commonly known as the engine behind GitHub CoPilot (OpenAI, 2021b). Codex is built on the same foundation as ChatGPT. This study found that students, primarily with limited or intermediate programming skills, faced challenges in using Codex where one large obstacle was learning what the prompt should be to receive an expected result, though it was also found to be seen as a valuable creative resource. Notably, 12 out of 13 test subjects managed to generate functional programming code. These findings build expectations in that GPT-4, a more advanced model than Codex, should be capable of generating operational game programming code in response to the tasks that are intended to be presented to the model.

Jalil et al. at George Mason University, USA, conducted an experiment to evaluate ChatGPT's performance in handling software-related inquiries. This study found that, within a shared query context, ChatGPT provided correct or partially correct answers 55.6% of the time, while in separate query contexts, the rate dropped to 42%. The researchers believe that the higher success rate in shared query contexts could be attributed to ChatGPT's ability to gather contextual information from preceding sub-questions, aiding in producing accurate responses. Additionally, they investigated the correctness of answer explanations, discovering that ChatGPT performed better in shared query contexts (53% correct or partially correct) compared to separate query contexts (43.2% correct or partially correct) (Jalil et al. 2023). These findings suggest that providing some context is necessary to maximize the assistance received from the AI model when generating game components, which was utilized in this research.

An early study of DALL-E 2 tested whether the model could generate images from prompts that were more challenging than the typical ones being showcased. The captions were designed to probe potential weaknesses, and some were screened on Google Images. The authors found that for 5 out of the 14 prompts, at least one of the ten images fully satisfied their requests. On the other hand, on no prompt did all the ten images satisfy their requests. From the results they drew the preliminary conclusions, some negative findings were (among other things) that numbers may be poorly understood, results are often incomplete, compositionality is poorly understood, relationships between objects are particularly challenging and negation is problematic. On the more positive side the images are stunning, DALL-E 2 succeeds in applying many artistic styles and some of the system's language abilities seem to be quite reliable. (Marcus et al, 2022).

3 Methodology

This section will cover how the study was performed, alternative research strategies will be discussed, and potential biases as well as ethical aspects of the study will be considered.

3.1 Research strategies and methods

Since this study focused on a small subset of all generative AI models and aspired to go into depth on the capabilities of the evaluated models, the "case study" research strategy was chosen. A case study focuses on one or a few instances of a phenomenon to be investigated and offers a deep insight of those instances (Johannesson and Perjons, 2014). According to Johannesson and Perjons (2014), the phenomenon should be chosen based on the research question and the instances can be chosen based on several different considerations. The investigated phenomenon is generative AI and the considerations most important to this study were that the instance must be accessible to the public as well as considered to be among the top of their field. The models chosen for this study were ChatGPT-4 and DALL-E 2. They were chosen based on their creators at OpenAI being perceived as leaders in the field while also having the models be publicly available.

The models were given a series of problems to solve. These problems were categorized into three different areas of development based on different areas of expertise in the game development industry:

- Programming
- Game design
- 2D visuals

All individual prompts and their categorial belonging can be found in Appendix A.

Each prompt was created based on tasks which students can receive while studying game development as well as situations that can appear while working in the industry. A programming task could be *"Write a character controller for our 3D character"* which was converted into the prompt:

> *"Write C++ code/classes for Unreal Engine for a 3D character controller, the character should be able to collide with walls and ground. It should make use of the existing physics system in the engine."*

A game design task could be *"Design another power-up to give our game more depth and make it more interesting"* which was constructed as a prompt:

> *"Please suggest a power-up that could be fun to have in the game Super Mario Bros. for the Nintendo Entertainment System, in addition to the mushroom, the fire flower and the star. The power up should not be too powerful and fit within the theme of the game."*

Instead of explaining an entire original game idea to the model a power-up for an existing well-known game was requested.

A 2D visuals task could be "*We need some art of a furious orc swinging an axe for a new card in our card game"* which became the prompt:

> *"A creature card for a game like Hearthstone, depicting an orc that furiously swings an axe."*

Again, an existing game was used as a basis for the prompt.

To generate content for a hypothetical original game in both the "2D visuals"- and the "game design"-examples, the models could have been provided with information that would have been derived earlier in the development process of the hypothetical game. DALL-E can edit an existing image, and a card template could have been provided and the prompt could request DALL-E to fill in the art-area of the template with the image of a furious orc swinging an axe. And of course, ChatGPT can process a lot of text, and descriptions of the game and its current power ups from a Game Design Document could have been provided in the prompt.

3.2 Data collection and analysis

The collected data was comprised of both textual and visual elements, generated by DALL-E and ChatGPT via their respective web interfaces. For this study the models were limited to a maximum of five input queries per task. The decision to conduct the tests within a shared query context, as opposed to a separate query context, was based on the findings of Jalil et al. (2023), which demonstrated that the success rates for shared query contexts were more favorable for AI model task completion. These findings aligned with the investigators' anticipation of the models' real-world application in problem-solving situations. Given that DALL-E produces four distinct output image variations in response to a query, every image is evaluated and if none of them meet all the criteria another one of the five input queries is expended to generate a new set of images.

To evaluate the output, pre-determined criteria based on the responses to an anonymous internet survey targeting professionals and university students within the field, was used. As mentioned by Denscombe (2010), in a study or case like this when the researchers are in pursuit of information relating to groups of people and additionally when gathering data that is relatively uncomplicated, a survey is a valid data collection method option. The survey gave respondents a set of potential criteria and asked them to rate how important that criteria is to their field. The survey also asked the respondents if there were any criteria not included which they felt were important. The survey was conducted using the internet service Google Forms. The survey questions can be found in Appendix B, and all the responses can be found in Appendix C.

The criteria suggestions used in the survey were determined based on evaluation criteria used by organizations in other fields. The OECD (OECD, no date) evaluates the merit or worth of an intervention based on six criteria: relevance, coherence, effectiveness, efficiency, impact, and sustainability. The same basis for criteria was used in this study.

- Relevance: Is the output relevant to the user?
- Coherence: Is the output aligned with the expectations of the user?
- Impact: Can the output be directly implemented in a product or speed up the development of assets for the product?
- Sustainability: Can the output scale with the product?

3.2.1 Programming criteria

The criterion “Does the code compile?” was constructed to evaluate whether the code is relevant. The survey found that roughly 90% of respondents consider code compilation to be important while evaluation code output. The criteria “The code performs what we requested” was constructed to make sure that the output is aligned with the expectations of the user. The survey agreed, with 70% of respondents considering it to be very important and nobody considered it to be not important. The criteria “The code has no obvious performance issues” was constructed to reflect the impact of the output. The survey respondents did not find this criterion as important, roughly 10% considered it to be not important at all, with the rest considering it to be somewhat important. Sustainability is very important to industry, especially whether the code is readable and easy to understand so it can be iterated on or scaled up. The criterion “The code is readable/easy to understand” was formulated to reflect this and none of the survey respondents said it was not important.

3.2.2 Game design criteria

There was only one game designer who responded to the survey, the impact this will have on the study is considered in the discussion part of the paper. The criteria “The output makes sense within the specified game scenario” and “The output contains a reasonable scope” were formulated to reflect the relevance and coherence of the output. The impact of the output is hard to satisfy within the realm of game design, especially given the low turnout from game designers in the survey, for this reason the impact was not considered when evaluating the game design tasks. The criteria “The output could be used as inspiration for a designer” and “The output has a tone of originality” were constructed to reflect the sustainability of the output by considering how well the output can be expanded on. The originality of the output was tested by the researchers through detailed internet searches on google.

3.2.3 2D Visuals criteria

As with Game Design, there was only one respondent in the category of 2D/3D visuals and the repercussions are considered in the discussion part of the paper. The criteria “The output image depicts what we requested” was constructed to reflect the relevance and coherence of the image output. To evaluate the impact of the output the criteria “The image is viable for use in the game without any editing needs” and “The output is properly cropped” were used. Lastly, a criterion reflecting the sustainability of the output images was formulated as “The image could be used as inspiration for an artist”.

Programming	Game Design	2D Visuals
“The code compiles”	“The output makes sense within the specified game scenario”	“The output image depicts what we requested”
“The code performs what we requested”	“The output contains a reasonable scope”	“The image is viable for use in the game without any editing needs”
“The code has no obvious performance issues”	“The output could be used as inspiration for a designer”	“The image could be used as inspiration for an artist”
“The code is readable and easy to understand”	“The output has a tone of originality”	“The output is properly cropped”

Table 1 All categories and their evaluation criteria.

3.3 Alternative research strategies and methods

An alternative research strategy could have been an experiment where qualitative data is collected by using observation and interviews as data collection methods. A simple game is designed and implemented by humans, then another game is designed and implemented using generative AI. Impartial testers play each game after which they are interviewed one by one, answering questions about the quality of the game as well as what parts they believed to be made by AI. This approach could have tested not only how AI could handle a wider array of game development tasks, but also whether these tasks could be completed to a degree where regular players could distinguish if content was made by a human or AI. Ultimately, this approach was disregarded due to insufficient time and resources.

This study also considered the observational survey research strategy, in which a group of testers would be given individual tasks to solve using generative AI and qualitative data is collected through observation and interviews. Each participant would specify what role within game development they primarily study or work in and then be given a series of tasks to complete. The tasks would be of varying difficulty within the specified role, supplemented by a few easier tasks from outside of the specified role. This strategy would have further removed biases from the result while still allowing tests and comparisons to see how well the AIs can augment humans in the completion of tasks within different areas of game development. This strategy was dismissed due to time constraints and an expected difficulty in finding a large enough set of participants to draw valuable conclusions.

Lastly, an experiment was considered, evaluating the output from a more quantitative stance. The idea behind the strategy was to award the AI a score for each task performed. Each criterion met by the AI output would result in the score relating to that criterion being added to the total score for each task. The scoring system would have been used to make a quantitative analysis of the results. By scoring each task in a category from the same criteria, comparisons could have been made between how the AI handled different tasks within an area of game development. The downside with this strategy would have been that the criteria for the different categories could be too unevenly weighted or not weighted according to their relevance. Drawing conclusions from uneven criteria would have posed too much risk of the results being invalid. As Martyn Denscombe (2010) also states, quantitative research shouldn’t involve

the researchers' own opinions in the production of the statistics. For this reason, this setup was dismissed.

3.4 Ethical aspects

The study was conducted utilizing the guidelines put forth by the Swedish Research Council (Vetenskapsrådet, 2017) to minimize negative consequences such as hurting people, breaking Swedish laws or spreading false information.

Since the evaluation of the output depends on how well it aligns with the researchers' own expectations and standards, there is a strong possibility that the results presented in this study may be influenced by some degree of subjectivity. To set the criteria for each task more reliably, a survey was conducted to get an outside opinion of how the AI output should be evaluated. The survey is reasonably reliable as it was only sent to professionals and students in the game development field. Multiple researchers participated in all aspects of data gathering and assessment, which helped lower the likelihood of observer bias and resulted in a more consistent evaluation.

For the sake of integrity, no personal data was collected in the survey collecting responses regarding the evaluation criteria. It was answered anonymously by each respondent. The file containing the results was deleted after all the responses had been tallied. The survey questions can be found in Appendix B and responses can be found in Appendix C.

4 Results

At the end of the study a total of 30 tasks had been completed by the generative AI models, and then evaluated by the researchers with the aim to determine the strengths and weaknesses in generating video game components using AI. ChatGPT completed ten programming tasks and ten game design tasks while DALL-E completed ten 2D Visuals tasks. Beneath, each category and its respective task results are presented. The task conversations can be found in Appendix D, please see Appendix E for individual task conversation evaluation results.

4.1 Programming

The results show that ChatGPT-4 can write code that is relevant, cohesive, impactful, and sustainable. A total of 8 tasks (80%) compiled without errors, however it was rare that the code compiled on the first prompt. In task 4 the code used a library which had not been imported in the solution build file which caused compilation errors. These errors were easily corrected by ChatGPT when it was forwarded the error message. No task was completed on the first prompt while 3 tasks (30%) used all five prompts. Additionally, 8 tasks (80%) had code with no obvious performance issues and 9 (90%) of tasks had code that is readable and easy to understand. With that said there were very few comments in the code, most of the tasks having no comments at all. As many as 8 tasks (80%) had code that performed the requested function to some degree but out of those tasks only 3 (30% of total) fully performed the requested function. In task 9 ChatGPT had no problems implementing a character controller for a 3D character, but in task 8 it could not write code that moved an object based on where in the game world the player right clicked with the mouse.

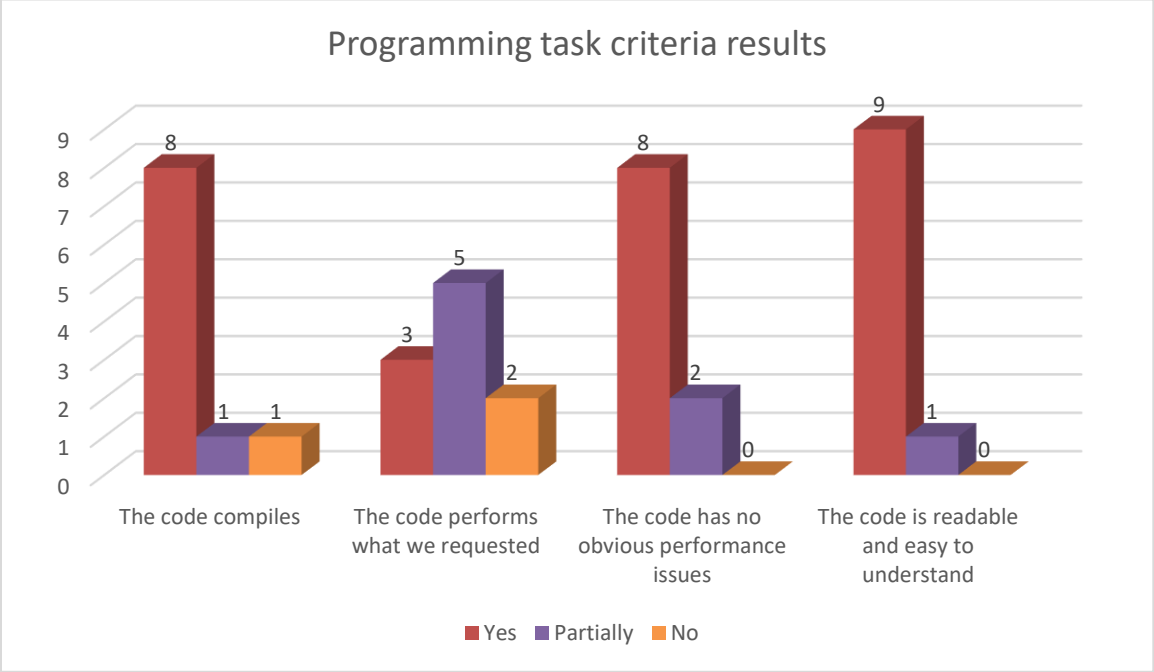

Figure 3 Programming task criteria results

4.2 Game design

In the game design category, ChatGPT exhibited significant potential with respect to relevance, coherence, and sustainability. Across all tasks, ChatGPT's output was either partially (10%) or completely (90%) satisfactory when assessed based on relevance and coherence. For every task, ChatGPT offered sensible suggestions and solutions that fit within the given game scenario. Task 1 was the only instance where the response was only partially satisfactory with respect to the criteria of the output making sense in the specified game scenario, as it included mechanics that were not originally present in the task prompt. Task 5 was the only case where ChatGPT could not provide an answer within a reasonable scope, as the response contained overly extensive suggestions for the problem stated in the task. Nonetheless, in the other nine (90%) tasks, the solutions were well within the scope of the assigned task. ChatGPT consistently provided suggestions that could serve as inspiration for game designers in all ten tasks. However, it became evident that ChatGPT struggled with originality, as only three (30%) of the tasks contained suggestions that did not closely resemble pre-existing mechanics or scenarios. In contrast to programming tasks, ChatGPT was able to solve the presented problems in a few prompts. In half of the tasks (50%), it provided adequate responses in the initial response, while in the remaining 5 tasks (50%), it only required two prompts.

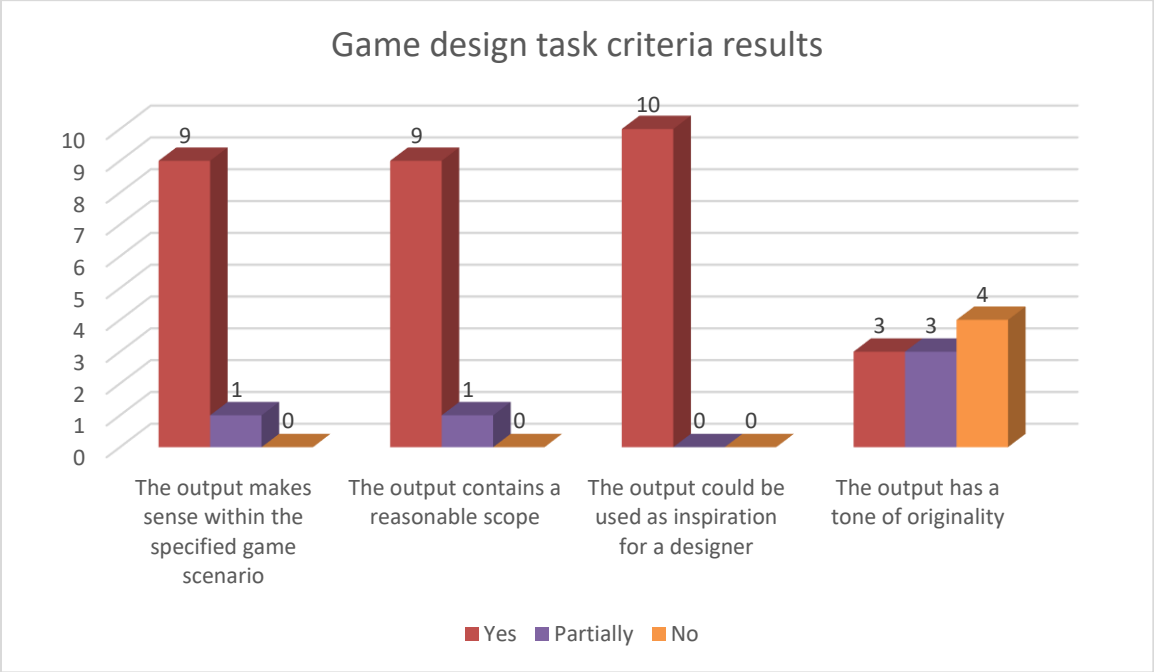

Figure 4 Game design task criteria results

4.3 2D Visuals

In the 2D visuals category, the outcomes experienced a decline in standard. Regarding the criteria that assess the relevance and coherence of the generated images, only four (40%) of the conversations produced output images that accurately portrayed the desired items. Meanwhile, six (60%) of the cases partially fulfilled the request. The criteria designed to evaluate the impact criterion displayed varied results, with a significant minority of two (20%) instances producing game-ready outputs without the need for editing. For example, in task four, all four generated images exhibited potential, and with the addition of extra generation frames, the final output became suitable for various game scenarios. In contrast, in tasks like task five, none of the output images were well-suited for gaming situations, primarily due to their failure to meet several other criteria. DALL-E fully satisfied the criterion related

to the sustainability of the output, providing inspiring suggestions for artists in all ten (100%) tasks. Lastly, DALL-E successfully depicted the entire content within the image boundaries in six (60%) instances. However, in some cases, DALL-E struggled to create a complete image even when given the instruction to add extra generation frames multiple times.

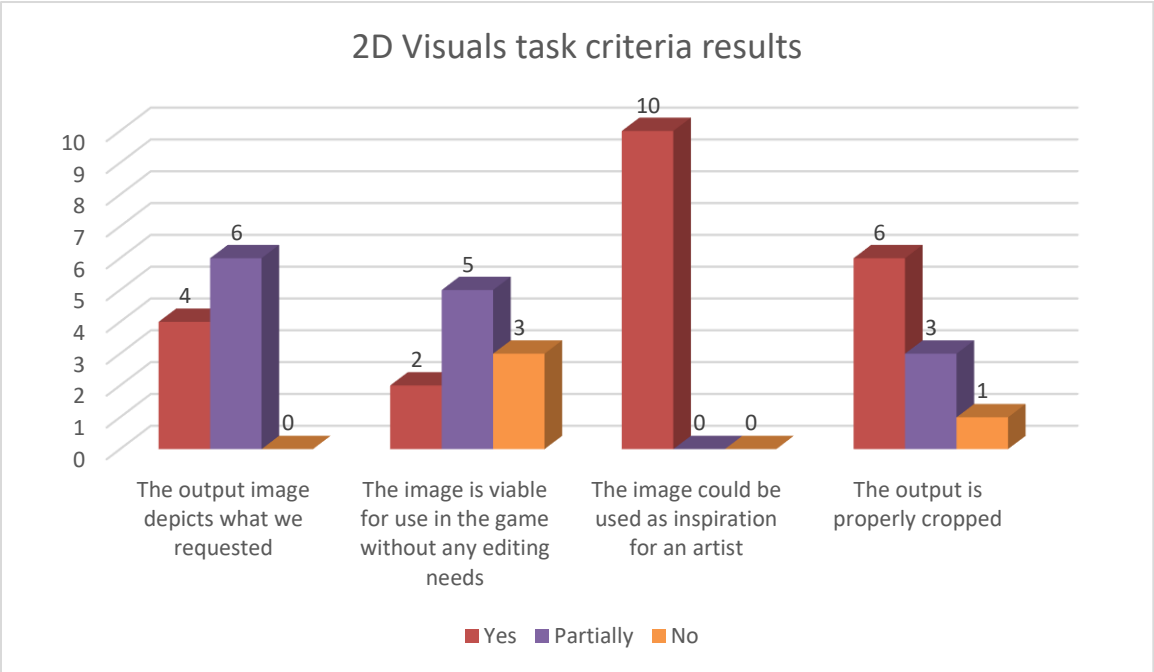

Figure 5 2D Visuals task criteria results

5 Discussion

This section will broadly discuss the implications of the results, the validity of the study itself and the ethics of using generative AI, both generally and specifically in game development. Future research will also be suggested and ultimately a summary conclusion will be drawn.

5.1 Collaborating with AI

The AI models demonstrated successful responses in most of the tasks, meeting either all or some of the criteria in all categories. This indicates that the models are sufficient as game development tools, especially since the tasks were designed to simulate real-world work tasks.

The feedback loop between the researchers and ChatGPT during the programming tasks was limited to only identifying and reporting errors in the IDE or game engine editor or unexpected behavior by the created assets. The taken stance was based on simulating a situation of a game developer with no or limited prior programming experience could complete daily programming tasks using the tool. However, it was found that some tasks were not solved with the provided feedback, and ChatGPT struggled to fulfill the "The code performs what was requested" criterion. It often suggested solutions that were close to solving the problem or only solved parts of it. A developer would need to identify the weaknesses in the output to fully utilize ChatGPT's strengths in completing programming tasks for game development. Nonetheless, ChatGPT would still be a valuable tool for anyone interested in programming tasks as its suggestions, sample code, and provided steps would assist a developer of any rank.

The study did not quantify the speed improvement from using generative AI in development, but during the study, it seemed that generating code was faster compared to manually writing it and generating art and game designs was great for inspiration which could speed up development in those areas. However, programmers will need to be able to identify when they run into a situation where the AI is not sufficient at generating a good enough answer to the specified prompt, else they risk spending more time trying to correct the AI output than it would take to write the code themselves.

5.2 AI as a game developer

Due to the number of times the AI either missed the mark entirely or required multiple additional prompts to resolve issues it is suggestive that generative AI is not yet ready to replace game developers completely. At best, some knowledge in the field is required to adjust and correct the output for it to be properly implemented in a game. At worst the output is not usable at all without major interventions from a human developer. What could be noted as a possibility, other than using AI as a collaborator or tool in the development process, is that AI could assume the role of low-level developers while humans move to a role where they split their time between higher level development and supervising the AI; either writing follow up prompts or correcting minor mistakes themselves.

5.3 Validity of the study

The evaluation of the models in the Game design and 2D Visuals tasks highlights an important aspect of subjectivity in the criteria for these categories. The criteria were developed based solely on our own preexisting knowledge within the fields, and due to the limited survey responses, there was an inability to reinforce the criteria through external sources. It is acknowledged that this weakness in the evaluation may introduce a level of personal bias into the results of these sections. Additionally, the conducted survey was flawed, and in hindsight, a more substantial network of developers in the Game design and 2D Visual fields should have been used to obtain a more precise evaluation of the AI models in these categories.

It is also important to note that no specific scientific data analysis method was utilized to analyze the data in this study. While research on generative AI and benchmarking tools for the technology was found, such as the work done by Ahuja et al. (2023), no suitable data analysis methods for evaluating the output of regular game development tasks in a study of this size could be identified. Therefore, a large reliance on the researchers' expertise and judgment in analyzing the data in the AI models' output was relied upon, which may have introduced some personal preferences and expectations into the results.

Lastly, there are flaws in the chat prompts in the conversations with DALL-E. Further preparation of learning proper syntax and knowing the limitations of the system would have been useful to properly assess the output as many attempts were used to solve problems through adding specific commands to a prompt that the system wouldn't add to the images.

5.4 Ethical considerations

This study shows that there is definite potential in using AI to empower game developers. As AI becomes better, developers will become more productive, which will lead to studios needing fewer developers to develop their games to the same standards. This means that the studios can save money by simply hiring fewer developers, or even reducing the number of developers they employ. This will not be limited to the game development industry and if it happens on a large enough scale, it could be devastating to many people. It is imperative that the industry and our society as a whole consider the ramifications of widespread adaptation of AI in the workplace.

Another, less dramatical, ethical consideration is how AI will be accessed by development studios. A potential avenue is for studios to develop their own AI and train it on specialized data from their internal sources, giving birth to models that are extremely good at generating content for that specific studio. While this sounds good at first, it comes with the problem that this would give large studios with big budgets and a lot of titles under their belt a huge advantage over smaller indie studios that may not have the resources to develop their own AI or do not have access to an AI of similar quality. Another potential avenue is to have general game development AI trained by external partners and then these models are bought or hired by studios. This also sounds good and has fewer complications, but it would still be a case of how expensive these AI are to buy or hire, pricing would need to be at a level where even studios with limited funding could get a fair chance of enhancing their games using the same AI as everyone else. Perhaps the situation won't change much from how it is now, a few businesses or researchers train general AI which are publicly available either for free or for a small fee. This is far-fetched, new innovations usually upset the status quo and it is believed that generative AI in this context will be no different.

When contemplating how game developers will elicit the aid of AI, another ethical consideration comes to mind; Who owns the AI generated content? There is also the question of whether the original creators of the source material used as training data for the AI should gain a share of the profits reaped by using the AI to generate value.

5.5 Future research

Assessing both the evaluated categories in this study and other aspects of game development is desired for the future. As the technology continues to advance, more usage areas for the models should be explored. Important yet very specific parts of game development have been touched upon in this study, for example, programming tasks concerning the chosen game engine were performed, but a larger sample size is needed to draw clearer conclusions in terms of the field of game programming itself. With that said, a suggestion is that a larger study is carried out, testing ChatGPT's capabilities within several different programming languages and that validation of the code is performed in several different game engines.

It would also be useful to conduct a study of whether using generative AI can save time in the game development process. Based on the tests conducted in this study, you could expect that generative AI could save time, and it seems like a widely held belief that this is the case, but it would be good to have scientifically derived data to back up this assumption.

There was an intention to base this study on prior similar research, but due to this field being a relatively new addition to the scientific world, very little research has been done to evaluate generative AI. As AI keeps evolving and becomes more relevant to the industry, it would be of great value to have a standardized scientifically sound method to evaluate how good a given model is at generating video game components. This would help game development studios get the most value out of their AI investments.

5.6 Conclusions

The first research question asked what the strengths and weaknesses are of generative AI in the context of game development. Considering the study findings, it has been determined that ChatGPT demonstrates promise and practicality as a tool for generating code samples and assisting game designers in various aspects of the creative process, for instance assisting with accessibility features, game mechanics, and feature balancing. Although DALL-E performs well in generating general-purpose images, it falls short in producing finished visuals suitable for game development, such as graphical user interfaces, sprite sheets, and textures without a substantial amount of corrective work needed to be of acceptable quality.

Across all three evaluated categories, the output generated by the AI models serves as a valuable source of inspiration for game developers. However, it is important to note that the originality of the AI-generated ideas is limited. It was also found that while ChatGPT can offer insightful suggestions, it may occasionally lead the user astray by providing confident responses that do not effectively address the underlying issue at hand.

Moving on to the second research question, the aim was set on evaluating the feasibility of generating entire video game components primarily relying on AI. The study concludes that game developers cannot currently rely solely on artificial intelligence to create complete game components. The outputs

generated by AI, while valuable and inspiring, necessitate scrutiny and critical evaluation from professionals within the field. It is crucial for game developers to exercise their expertise and perform rigorous verification of the AI-generated content within each specific context of game development.

In summary, this research underscores the potential benefits of incorporating generative AI, such as ChatGPT and DALL-E, into the game development process. It can offer valuable insights, code samples, and ideas related to accessibility features, game mechanics, feature balancing and inspiration for game visuals. However, limitations exist in terms of originality and the generation of game-specific visuals. Game developers must remain discerning and exercise critical evaluation when leveraging AI-generated content, acknowledging the current need for human expertise and verification to ensure the creation of high-quality and cohesive game components.

References

Ahuja, K., Hada, R., Ochieng, M., Jain, P., Diddee, H., Maina, S., and Sitaram, S. (2023) "MEGA: Multilingual Evaluation of Generative AI." arXiv preprint arXiv:2303.12528. Available at https://arxiv.org/pdf/2303.12528.pdf (Accessed May 5, 2023).

Allganize. (2022). *How John McCarthy Shaped the Future of AI*. [Blog] 4 September. Available at https://blog.allganize.ai/john-mccarthy/ (Accessed April 4, 2023).

Bramble, R. (2022). *How Much Does It Cost To Make A Video Game?* [Blog] 13 July. Available at https://gamemaker.io/en/blog/cost-of-making-a-game (Accessed 27 March 2023).

Creswell, A., White, T., Dumoulin, V., Arulkumaran, K., Sengupta B. and Bharath, A. A. (2018). "Generative Adversarial Networks: An Overview". *IEEE Signal Processing Magazine*, vol. 35, no. 1, pp. 53-65. Doi: 10.1109/MSP.2017.2765202.

Data-Driven Science. (2020). *Why the AI revolution now? Because of 6 key factors*. Available at https://becominghuman.ai/why-the-ai-revolution-now-because-of-6-key-factors-7ee92e482d2 (Accessed 28 March 2023).

Davidson, R. (2017). *The Big List of: Video Game Development Team Roles*. Available at https://cdn.fs.teachablecdn.com/N4tk2YWxTHaM6neBSVqV (Accessed 27 March 2023).

Denscombe, M. (2010). *The Good Research Guide for Small Scale Research Projects* (4th ed.) Buckingham: Open University Press.

Grant, E. F., Lardner, R (1952). "The Talk of the Town – It". The New Yorker. 2 August. Available at https://www.newyorker.com/magazine/1952/08/02/it (Accessed 15 June 2023).Gugerty, L. (2006)." Newell and Simon's Logic Theorist: Historical Background and Impact on Cognitive Modeling". *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 50(9), 880–884. Doi: 10.1177/154193120605000904

Jalil, S., Rafi, S., LaToza, T. D., Moran, K. and Lam, W. (2023) "ChatGPT and Software Testing Education: Promises & Perils", *IEEE International Conference on Software Testing*, Verification and Validation Workshops, 16(1). Available at Doi: 10.48550/arXiv.2302.03287 (Accessed: 31 March 2023).

Johannesson, P. and Perjons, E. (2014). *An Introduction to Design Science*. Springer Cham. Switzerland. Doi: https://doi.org/10.1007/978-3-319-10632-8.

Jonsson, M. and Tholander, J. (2022). "Cracking the code: Co-coding with AI in Creative Programming Education". *Creativity and Cognition* [Preprint]. Doi: 10.1145/3527927.3532801.

Kapronczay, M. (2022). *A Beginner's Guide to Language Models*. Available at https://builtin.com/data-science/beginners-guide-language-models (Accessed April 4, 2023).

Lai, J. (2022). [Twitter] 12 December. Available at: https://twitter.com/Tocelot/status/1602338827284238337 (Accessed 27 March 2023).

Lambert, N., Castricato, L., Werra, L. von and Havrilla, A. (2022). *Illustrating Reinforcement Learning from Human Feedback (RLHF)*. [Blog] 9 December. Available at https://huggingface.co/blog/rlhf (Accessed 28 March 2023).

Marcus, G., Davis, E. and Aaronson, S. (2022). *A very preliminary analysis of DALL-E 2*. *arXiv preprint arXiv:2204.13807*. Doi: 10.48550/arXiv.2204.13807.

Marenko, B. (2015). "When making becomes divination: Uncertainty and contingency in computational glitch-events". *Design Studies 41* (Nov. 2015), 110–125. Doi: 10.1016/j.destud.2015.08.004.

OECD. (no date). *Evaluation criteria*. https://www.oecd.org/dac/evaluation/daccriteriaforevaluatingdevelopmentassistance.htm (Accessed April 18, 2023).

OpenAI (2019) *Open-AI five defeats Dota 2 world champions*, *OpenAI Five defeats Dota 2 world champions*. Available at https://openai.com/research/openai-five-defeats-dota-2-world-champions (Accessed March 28, 2023).

OpenAI. (2021a). *DALL-E: Creating images from text.* Available at https://openai.com/research/dall-e (Accessed 29 March 2023).

OpenAI. (2021b). *OpenAI Codex*. Available at https://openai.com/blog/openai-codex (Accessed March 30, 2023).

OpenAI. (2022). *Introducing ChatGPT*. [Blog] 30 November. Available at https://openai.com/blog/chatgpt (Accessed March 28, 2023).

OpenAI. (2023a). *GPT-4 technical report*, *arXiv.org*. Doi: 10.48550/arXiv.2303.08774.Rudolph, J., Tan, S. and Tan, S. (2023). "ChatGPT: Bullshit spewer or the end of traditional assessments in higher education?". *Journal of Applied Learning and Teaching*, *6*(1). Doi: 10.37074/jalt.2023.6.1.9.

Schaeffer, J. (1997). "One Jump Ahead: Challenging Human Supremacy in Checkers". *ICGA Journal*.

'The Dark Side of the Video Game Industry' (2019) *Patriot Act with Hasan Minhaj*, Season 4, episode 1. Available at: Netflix (Accessed: March 27 2023).

Unreal engine (no date) *Unreal Engine 5*. Available at https://www.unrealengine.com/en-US/unreal-engine-5 (Accessed April 4, 2023).

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez A. N., Kaiser, L. and Polosukhin, I. (2017), "Attention Is All You Need". *Advances in Neural Information Processing Systems 30* (NIPS 2017), Long Beach, California, 4-9 December. Available at: https://papers.nips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf (Accessed 29 March 2023).

Vetenskapsrådet (2017). *God forskningssed*. (Revised version). Stockholm: Vetenskapsrådet.

Xia, B., Ye, X. and Abuassba, A. O. M. (2020). "Recent Research on AI in Games". *2020 International Wireless Communications and Mobile Computing (IWCMC)*, Limassol, Cyprus. pp.505-510. Doi: 10.1109/IWCMC48107.2020.9148327.

Xu, S. (2014). *History of AI design in video games and its development in RTS games*. *https://sites.google.com/site/myangelcafe/articles/history_ai* (Accessed 15 June 2023).

Appendix A – Task prompts

Programming

1. Write C++ code/classes for a Role-Playing Game in Unreal Engine that can be used to define different ailments which can affect a character. Also define the ailments "burning" which deals damage over time to the character as well as "broken leg" which reduces the character's movement points. The ailments should also be curable by using a cure all-potion.
2. Write C++ code/classes for Unreal Engine for a 2D enemy character that can use the ability "jump attack" which causes the enemy to jump to a location and deal damage to all players within an area of where it lands.
3. Write C++ code/classes for Unreal Engine for a sound handler that increases or lowers the volume of the music based on how much health the player has.
4. Write C++ code/classes for Unreal Engine for a dialogue box that displays text on the screen and waits for the player to press a specific button. The text should appear one letter at a time to look as if it's being written in real time.
5. Write C++ code/classes for Unreal Engine for a login screen where the player can either choose to log in or register. If the player wants to register, they get to input a name and a password and they are saved to a database. If the name already exists, then nothing should be saved, and the player should be informed the name is already taken. If the player wants to log in then they enter a name and a password, if the name and the password matches with a character + password combination in the database then the player is informed that they are now logged in.
6. Write C++ code/classes for Unreal Engine that draws a line between two objects in the game world scene.
7. Write C++ code/classes for Unreal Engine that draws a circle on the ground around a 3D object in the game world.
8. Write C++ code/classes for Unreal Engine for a unit in a Real Time Strategy game. The unit should be able to move with a set movement speed in the game world. Where the unit should move is based on where the player right clicks. When the unit starts moving to a new location it should play a sound clip that can be defined in the editor.
9. Write C++ code/classes for Unreal Engine for a 3D character controller, the character should be able to collide with walls and ground. It should make use of the existing physics system in the engine.
10. Write C++ code/classes for Unreal Engine for a ball that bounces realistically without making use of the built in physics system.
11. Write C++ code/classes for Unreal Engine for a pistol, a projectile and the player interaction with it. The gun should fire projectiles straight forward from the pistol and the projectiles should be removed from the scene when colliding with a surface.
12. Write C++ code/classes for Unreal Engine that rotates a scene object towards a specific target over time. The mentioned scene object has a parent object.

13. Write C++ code/classes for Unreal Engine for picking up an item from the ground in a 3D game in unreal. The item should be picked up when the player pawn object gets close enough to the object, make sure to allow for the value determining the proximity to be set in the game engine editor. The picked-up item should add plus 1 to a counter in the scene.
14. Write C++ code/classes for Unreal Engine for entering a menu. It should pause the game and the game should be unpaused when exiting the menu using escape as the key for pausing and unpausing.
15. Write C++ code/classes for Unreal Engine for a 2D character controller, the character should be able to collide with walls and ground. It should make use of the existing physics system in the engine.
16. Write C++ code/classes for Unreal Engine script for unreal, implement a pathfinding algorithm for an enemy, the enemy should move towards a target using the pathfinding algorithm.
17. Write C++ code/classes for Unreal Engine, implement an A* algorithm for an enemy, should make an enemy follow a target.

Game Design

1. In a card game in the style of Hearthstone there are three creature cards. The first card is "Wood Elf" which costs 1 mana, has 1 power and 1 toughness. The second card is "Plague Rat" which costs 2 mana, has 1 power, 1 toughness and reduces the toughness of enemy creatures by 1. And the third card is "Red Lake Ogre" which costs 2 mana, has 1 power and 3 toughness. The Plague Rat card is very popular due to how good it is while the Wood Elf card is hardly ever used. Can you propose changes that would make Wood Elf more popular and Plague Rat less popular without making either card too good or too bad?
2. Please suggest a power-up that could be fun to have in the game Super Mario Bros. for the Nintendo Entertainment System, in addition to the mushroom, the fire flower and the star. The power up should not be too powerful and fit within the theme of the game.
3. We have made a 2D puzzle game where the player progresses through multiple levels. The objective in each level is to move the player from the start to the goal by changing the direction of gravity. Please give us 5 suggestions for potential game titles.
4. Design a playable character for a MOBA (multiplayer online battle arena) game. The character should have four abilities including: 1 ultimate ability and 1 movement ability.
5. Please suggest changes that would make “Counter-strike: Global offensive” a better game.
6. Please provide me with a cool game design specification of an enemy boss in a 2D platformer. What abilities should the boss have? What weaknesses could be interesting in a setting where the player is a melee knight but can occasionally shoot with the character’s bow every 3 seconds.
7. Please suggest a cool movement mechanic in a 3D first person shooter. Imagine that the game is high paced and is set in a futuristic setting.
8. Design a suitable accessibility feature for a puzzle game where the targeted group is people with severely reduced eyesight.

9. Design a suitable accessibility feature for a first-person shooter where the targeted group is people with limited hearing.
10. Could you suggest a game idea? The idea should be suitable for a small studio of 5 people and should take no longer than 1 year to develop.

2D Visuals

1. A grass sprite in a pixelated art style that can seamlessly be tiled both vertically and horizontally.
2. A creature card for a game like Hearthstone, depicting an orc that furiously swings an axe.
3. A realistic science fiction wallpaper depicting stellar objects and a spaceship.
4. A face of a human man aged 55 in a pixel graphic style.
5. A handheld science fiction weapon called "B-42 Plasma Annihilator" in a pixar art style.
6. Video game enemy boss character with devil like features like horns and a tail, wearing a plate armor.
7. Brick texture with various dents and damages to the surface.
8. Rubber duck action hero with cape and sword, animated.
9. Computer game user interface menu element with rounded edges that should be used for an in game menu.
10. A tileable stone wall texture, high quality photo.

Appendix B – Survey form

AI in game development

With this form, we set out to investigate what reasonable criteria should be in terms of tasks performed in the following fields by AI models such as ChatGPT and DALL-E: programming, game mechanics and imagery (sprites, UI etc).

The survey is expected to take no longer than a couple of minutes.

What role within the game development area is closest to your area of expertise? (Radio buttons) (Required)

- Programmer (redirects to Section: Programming criteria)
- 2D/3D artist (redirects to Section: 2D Visuals criteria)
- Game designer (redirects to Section: Game design criteria)

Section: Programming criteria

Please rate how important each criteria is in the evaluation of a code output.

The code compiles (Likert scale 1 to 5) (Required)

(1 = not important, 5 = very important)

The code performs what we requested (Likert scale 1 to 5) (Required)

(1 = not important, 5 = very important)

The code has no obvious performance issues (Likert scale 1 to 5) (Required)

(1 = not important, 5 = very important)

The code is readable/easy to understand (Likert scale 1 to 5) (Required)

(1 = not important, 5 = very important)

Next page: Section: Before we finish

Section: 2D Visuals criteria

Please rate how important each criteria is in the evaluation of a visual element in a 2D game.

The output image depicts that which was requested (Likert scale 1 to 5) (Required)

(1 = not important, 5 = very important)

The image is viable for use in the game without any editing needs (Likert scale 1 to 5) (Required)

(1 = not important, 5 = very important)

The image could be used as inspiration for an artist rather than used directly in a game (Likert scale 1 to 5) (Required)

(1 = not important, 5 = very important)

The output contains a full image of the output (proper cropping) (Likert scale 1 to 5) (Required)

(1 = not important, 5 = very important)

Next page: Section: Before we finish

Section: Game design criteria

Please rate how important each criteria is in the evaluation of game design elements in a game.

The output makes sense within the specified game scenario (Likert scale 1 to 5) (Required)

(1 = not important, 5 = very important)

The output contains a reasonable scope (Likert scale 1 to 5) (Required)

(1 = not important, 5 = very important)

The output has a tone of originality (Likert scale 1 to 5) (Required)

(1 = not important, 5 = very important)

Next page: Section: Before we finish

Section: Before we finish

If there is any particular criteria that you couldn't find, but strongly believe should be a part of the evaluation of the AI model output, what would that be? (Free comment) (Not Required)

Do you have any additional comments? (Free comment) (Not required)

Survey complete.

Appendix C – Survey responses

Respondent 1:

Role: Programmer

The code compiles:
5

The code performs what we requested:
5

The code has no obvious performance issues:
5

The code is readable/easy to understand:
5

If there is any particular criteria that you couldn't find, but strongly believe should be a part of the evaluation of the AI model output, what would that be?
-

Do you have any additional comments?
-

Respondent 2:

Role: Programmer

The code compiles:
5

The code performs what we requested:
5

The code has no obvious performance issues:
4

The code is readable/easy to understand:
5

If there is any particular criteria that you couldn't find, but strongly believe should be a part of the evaluation of the AI model output, what would that be?
-

Do you have any additional comments?
-

Respondent 3:

Role: 2D/3D artist

The output image depicts that which was requested:
4

The image is viable for use in the game without any editing needs:
1

The image could be used as inspiration for an artist rather than used directly in a game:
5

The output contains a full image of the output (proper cropping):
3

If there is any particular criteria that you couldn't find, but strongly believe should be a part of the evaluation of the AI model output, what would that be?
-

Do you have any additional comments?
-

Respondent 4:

Role: Programmer

The code compiles:
5

The code performs what we requested:
4

The code has no obvious performance issues:
4

The code is readable/easy to understand:
3

If there is any particular criteria that you couldn't find, but strongly believe should be a part of the evaluation of the AI model output, what would that be?
-

Do you have any additional comments?
-

Respondent 5:

Role: Programmer

The code compiles:
5

The code performs what we requested:
5

The code has no obvious performance issues:
5

The code is readable/easy to understand:
5

If there is any particular criteria that you couldn't find, but strongly believe should be a part of the evaluation of the AI model output, what would that be?
“The current AI scraps the Internet with usually 2-3 years old information. The more the AI is used the smarter it can get, but still with the help of a developer. I do not like AI suggesting old best practices, and still if you enter too many old best practices that AI will not know any better.

So I would say, supervice the supervisors. “

Do you have any additional comments?
"Looking forward to use completely AI generated programs in the future."

Respondent 6:

Role: Programmer

The code compiles:
5

The code performs what we requested:
5

The code has no obvious performance issues:
4

The code is readable/easy to understand:
5

If there is any particular criteria that you couldn't find, but strongly believe should be a part of the evaluation of the AI model output, what would that be?
"Licensing of the code that the model is based on, if the code is based on LGPL it should be visible so that the user knows the restrictions up front."

Do you have any additional comments?
-

Respondent 7:

Role: Programmer

The code compiles:
4

The code performs what we requested:
3

The code has no obvious performance issues:
1

The code is readable/easy to understand:
4

If there is any particular criteria that you couldn't find, but strongly believe should be a part of the evaluation of the AI model output, what would that be?
-

Do you have any additional comments?
-

Respondent 8:

Role: Programmer

The code compiles:
2

The code performs what we requested:
4

The code has no obvious performance issues:
4

The code is readable/easy to understand:
4

If there is any particular criteria that you couldn't find, but strongly believe should be a part of the evaluation of the AI model output, what would that be?
-

Do you have any additional comments?
-

Respondent 9:

Role: Programmer

The code compiles:
4

The code performs what we requested:
5

The code has no obvious performance issues:
3

The code is readable/easy to understand:
4

If there is any particular criteria that you couldn't find, but strongly believe should be a part of the evaluation of the AI model output, what would that be?
"Consistency of data types and structs; For example, sometimes an AI will switch between a Vec3 and a tuple (u8,u8,u8) when iterating on the same prompt."

Do you have any additional comments?
-

Respondent 10:

Role: Programmer

The code compiles:
4

The code performs what we requested:
5

The code has no obvious performance issues:
4

The code is readable/easy to understand:
3

If there is any particular criteria that you couldn't find, but strongly believe should be a part of the evaluation of the AI model output, what would that be?
"License, that the output can be used without any restrictions and it's copyright is cleared."

Do you have any additional comments?
"No, I do not have any additional comments."

Respondent 11:

Role: Game designer

The output makes sense within the specified game scenario:

5

The output contains a reasonable scope:

4

The output has a tone of originality:

3

If there is any particular criteria that you couldn't find, but strongly believe should be a part of the evaluation of the AI model output, what would that be?

“How extensible and flexible the output is when you want to iterate on it and put your own flavour on it.”

Do you have any additional comments?

-

Appendix D – AI conversations

In this appendix you can find each conversation taking place in the testing of the AI models. R = Researcher and AI = The model used (ChatGPT-4 for programming/game design, DALL-E for 2D visuals).

You can find the appendix in the link below:

Appendix D - Google drive

Appendix E – AI conversations results

Programming:

Task 1.

The code compiles: Yes
The code performs what we requested: Partially
The code has no obvious performance issues: Yes
The code is readable and easy to understand: Yes
Prompts used: 3

Task 2.

The code compiles: Yes
The code performs what we requested: Partially
The code has no obvious performance issues: Yes
The code is readable and easy to understand: Yes
Prompts used: 4

Task 3.

The code compiles: Yes
The code performs what we requested: Yes
The code has no obvious performance issues: Yes
The code is readable and easy to understand: Yes
Prompts used: 4

Task 4.

The code compiles: Yes
The code performs what we requested: Partially
The code has no obvious performance issues: Yes
The code is readable and easy to understand: Yes
Prompts used: 3

Task 5.

The code compiles: Yes
The code performs what we requested: Yes
The code has no obvious performance issues: Yes
The code is readable and easy to understand: Yes
Prompts used: 2

Task 6.

The code compiles: Yes
The code performs what we requested: No
The code has no obvious performance issues: Partially

The code is readable and easy to understand: Partially
Prompts used: 5

Task 7.

The code compiles: Yes
The code performs what we requested: Partially
The code has no obvious performance issues: Partially
The code is readable and easy to understand: Yes
Prompts used: 4

Task 8.

The code compiles: No
The code performs what we requested: No
The code has no obvious performance issues: Yes
The code is readable and easy to understand: Yes
Prompts used: 5

Task 9.

The code compiles: Yes
The code performs what we requested: Yes
The code has no obvious performance issues: Yes
The code is readable and easy to understand: Yes
Prompts used: 3

Task 10.

The code compiles: Partially
The code performs what we requested: Partially
The code has no obvious performance issues: Yes
The code is readable and easy to understand: Yes
Prompts used: 5

Game design

Task 1.

The output makes sense within the specified game scenario: Partially
The output contains a reasonable scope: Yes
The output could be used as inspiration for a designer: Yes
The output has a tone of originality: No
Prompts used: 2

Task 2.

The output makes sense within the specified game scenario: Yes
The output contains a reasonable scope: Yes
The output could be used as inspiration for a designer: Yes
The output has a tone of originality: Partially
Prompts used: 2

Task 3.

The output makes sense within the specified game scenario: Yes
The output contains a reasonable scope: Yes
The output could be used as inspiration for a designer: Yes
The output has a tone of originality: Partially
Prompts used: 2

Task 4.

The output makes sense within the specified game scenario: Yes
The output contains a reasonable scope: Yes
The output could be used as inspiration for a designer: Yes
The output has a tone of originality: Yes
Prompts used: 1

Task 5.

The output makes sense within the specified game scenario: Yes
The output contains a reasonable scope: Partially
The output could be used as inspiration for a designer: Yes
The output has a tone of originality: No
Prompts used: 1

Task 6.

The output makes sense within the specified game scenario: Yes
The output contains a reasonable scope: Yes
The output could be used as inspiration for a designer: Yes
The output has a tone of originality: Yes
Prompts used: 1

Task 7.

The output makes sense within the specified game scenario: Yes
The output contains a reasonable scope: Yes
The output could be used as inspiration for a designer: Yes
The output has a tone of originality: Yes
Prompts used: 2

Task 8.

The output makes sense within the specified game scenario: Yes
The output contains a reasonable scope: Yes
The output could be used as inspiration for a designer: Yes
The output has a tone of originality: No
Prompts used: 2

Task 9.

The output makes sense within the specified game scenario: Yes
The output contains a reasonable scope: Yes
The output could be used as inspiration for a designer: Yes
The output has a tone of originality: No
Prompts used: 1

Task 10.

The output makes sense within the specified game scenario: Yes
The output contains a reasonable scope: Yes
The output could be used as inspiration for a designer: Yes
The output has a tone of originality: Partially
Prompts used: 1

2D Visuals

Task 1.

The output image depicts what we requested: Yes
The image is viable for use in the game without any editing needs: Partially
The image could be used as inspiration for an artist: Yes
The output is properly cropped: Yes
Prompts used: 2

Task 2.

The output image depicts what we requested: Partially
The image is viable for use in the game without any editing needs: Partially
The image could be used as inspiration for an artist: Yes
The output is properly cropped: Partially
Prompts used: 4

Task 3.

The output image depicts what we requested: Partially
The image is viable for use in the game without any editing needs: Yes
The image could be used as inspiration for an artist: Yes
The output is properly cropped: Yes
Prompts used: 4

Task 4.

The output image depicts what we requested: Yes
The image is viable for use in the game without any editing needs: Partially
The image could be used as inspiration for an artist: Yes
The output is properly cropped: Partially
Prompts used: 3

Task 5.

The output image depicts what we requested: Partially
The image is viable for use in the game without any editing needs: No
The image could be used as inspiration for an artist: Yes
The output is properly cropped: No
Prompts used: 4

Task 6.

The output image depicts what we requested: Yes
The image is viable for use in the game without any editing needs: Yes

The image could be used as inspiration for an artist: Yes
The output is properly cropped: Yes
Prompts used: 1

Task 7.

The output image depicts what we requested: Yes
The image is viable for use in the game without any editing needs: Partially
The image could be used as inspiration for an artist: Yes
The output is properly cropped: Yes
Prompts used: 1

Task 8.

The output image depicts what we requested: Partially
The image is viable for use in the game without any editing needs: No
The image could be used as inspiration for an artist: Yes
The output is properly cropped: Yes
Prompts used: 5

Task 9.

The output image depicts what we requested: Partially
The image is viable for use in the game without any editing needs: No
The image could be used as inspiration for an artist: Yes
The output is properly cropped: Yes
Prompts used: 5

Task 10.

The output image depicts what we requested: Partially
The image is viable for use in the game without any editing needs: Partially
The image could be used as inspiration for an artist: Yes
The output is properly cropped: Partially
Prompts used: 4