

Tomi Linkinen

# **GENERATIVE ARTIFICIAL INTELLIGENCES: CHALLENGES AND BENEFITS FOR GAME DEVELOPMENT**

Faculty of Information Technology and Communication Sciences

M. Sc. Thesis

February 2024

# ABSTRACT

Tomi Linkinen: Generative artificial intelligences: Challenges and benefits for game development

M.Sc. Thesis

Tampere University

Master's Degree Programme in Computing sciences, Human-Technology Interaction

February 2024

---

The purpose of this research is to examine the challenges and benefits of generative artificial intelligence (GAI) in multimedia generation, especially for game development purposes. Utilizing the design science research method to provide a structured approach for the research, a prototype of a game is created to serve as an artifact and as a framework for the GAI content created. This ensures that game design aspects are not disregarded and that the content created thematically fits into this pre-set framework. The research aims to illuminate the difficulties and possibilities of GAI and delve deeply into this rapidly evolving technology. This includes prompt engineering and creative aspects as well as the software development process and modifying the output of the GAI content. The research findings successfully provide many insights into GAI in game development, such as the "style drives content" dilemma, but more importantly, provide a general overview of the current and future situation. In addition, the findings suggest that integrating GAI into existing software, using it as inspiration or as a communication tool, will further foster human creativity and be helpful in software development and multimedia content creation. Serving game developers and researchers alike, the research attempts to enrich the scientific dialogue on GAIs and to promote attention towards the user and usability aspects of GAIs.

Keywords and terms: generative artificial intelligence, artificial intelligence, game development, game design, multimedia assets, game assets, prompt engineering, GAI, AI, design science research method, ChatGPT, Unreal engine 5, AIVA, Stable diffusion, Midjourney, LLM.

The originality of this thesis has been checked using the Turnitin OriginalityCheck service.

## Table of Contents

<b>1</b>	<b>Introduction .....</b>	<b>1</b>
1.1	Motivations	1
1.2	Research questions	3
1.3	Scope of work	7
1.4	Structure of thesis	8
<b>2</b>	<b>Game development .....</b>	<b>9</b>
2.1	Concise history of video games	9
2.2	Game Engines	13
2.3	Game development tools	14
2.4	Game Design	17
<b>3</b>	<b>Generative artificial intelligences.....</b>	<b>20</b>
3.1	Generative Model	20
3.2	Defining GAIs and Assessing Current Research	21
3.3	Prompt Engineering	23
3.4	Chat-GPT4 and large language models	24
3.5	Text-to-Text prompting techniques	26
3.6	Diffusion Models	28
3.7	Models for Music	33
3.8	Intersection of GAIs and games	36
3.9	General observations	36
<b>4</b>	<b>Methods .....</b>	<b>37</b>
4.1	Selected design science method (DSRM)	39
4.2	Other methods	40
4.3	Recording the experiences	41
<b>5</b>	<b>Research .....</b>	<b>42</b>
5.1	The Artifact	43
5.2	The Game design and mechanics of the Artifact	44
5.3	User Interface Design	46
5.4	Text generation	48
5.5	Image generation	57
5.6	Music generation	68

5.7	Empirical Conclusions	70
<b>6</b>	<b>Discussion and Conclusions</b> .....	<b>73</b>
6.1	Current state of the generative artificial intelligence	73
6.2	Challenges of GAI in game development	74
6.3	Benefits of GAI in game development	77
6.4	Addressing research questions	79
6.5	Future integration and evolution of GAI	81
6.6	Limitations of this research and recommendations for future research	82
6.7	Final thoughts	84
<b>7</b>	<b>Appendix</b> .....	<b>85</b>
<b>8</b>	<b>Use of ChatGPT in this thesis</b> .....	<b>85</b>
8.1	Example of ChatGPT usage for editing	85
<b>9</b>	<b>References</b> .....	<b>87</b>

# 1 Introduction

Generative artificial intelligence (GAI) has rapidly evolved in recent years, showcasing remarkable advancements in generating complex and diverse forms of media, including text, images, music, and other multimedia content. This technological revolution, still in its early stages, has seen increasing quality in the content generated using GAI.

However, the exploration of practical applications, usability, and particularly the challenges and benefits of GAI technology for users, remains a largely unexplored field. While technological dilemmas continue to be a primary focus among researchers pushing the boundaries of neural network capabilities, investigating the effective uses of this emerging technology from the perspectives of human users is equally important.

This investigation reveals the transformative impact of GAI on game development, emphasizing its significance for both game and multimedia developers. The study aims to enrich academic dialogue around GAI and provide insights into its application in game creation, delving into the benefits and challenges of GAI for game development to demystify its capabilities and limitations. As GAI becomes increasingly integrated into the software used for creating multimedia, including game assets, understanding how to best leverage this co-creation becomes crucial.

In this research, an artifact is created to serve as a framework for content generated using GAI, encompassing text, images, and music. The chosen methodology, design science research method, aims to provide a structured and scientifically rigorous foundation for this investigation. The research questions explore the benefits and challenges of using GAI to generate game assets, presenting complexities that largely defy simplistic answers, thereby inviting readers to engage with the discussion section for comprehensive insights.

## 1.1 Motivations

GAIs have advanced unprecedented ways in the last decade, however, the technological breakthroughs in the past couple of years have been even more groundbreaking. This can be observed by the quality of the generated media increasing to such levels, they sometimes cannot be differentiated from human work (Grierson, 2023). Surprisingly, this is happening in the image (Roose, 2022), music (Coscarelli, 2023), and text generation (Kelly, 2023) at the same time, as evident from these newspaper articles. Video, animation, and 3D models still have some ways to go to compete with humans in

terms occasional moments of brilliance, but they too are rapidly advancing. It can be argued that text, static images, music, and sound have reached such a level of quality where they could be used as everyday tools for multimedia industries. This research seeks to examine the benefits, challenges and considerations of how generative AIs can be used as tools for game and multimedia content development.

In the gaming industry, there is a pressing need to comprehend the array of tools that generative AIs offer, and to understand how these tools can contribute to the game development process. This need arises from the escalating costs associated with developing high-quality games (Zollner, 2023), coupled with the ever-growing expectations of players for enhanced quality and immersive experiences, as well as the intense competition in gaming industry. Thus, decreasing costs and developing time without sacrificing quality is of utmost interest for the gaming industry.

Integration of generative AI into the game development process could potentially decrease the cost and development time required. Furthermore, the co-creative process between artists and AI could offer numerous benefits, including, but not limited to, serving as a source of inspiration, enhancing the quality of work, mitigating decision fatigue, and aiding many artists working in the same project to maintain stylistic consistency. For developers, generative AIs as tools can offer very unique advantages, as in best case scenario they can serve as co-worker, co-creator and also enhance the developers' own skills. Currently, these anticipated benefits remain largely theoretical as research examining the effects and potential consequences of implementing generative AI tools in professional environments is in its emergent stage.

It is evident that the gaming industry professionals are interested in generative AIs and the possibilities they can offer. One example of this is a prototype game project called Echoes of Somewhere, that has been developed by Jussi-Petteri Kemppainen. The game development process using image generation AI is explained and documented in his blog (Kemppainen, 2023). Other example is that online course organizers are starting to teach generative AI in relation to game development (*Gamedevacademy.Org*, 2023).

There is value attempting to scientifically examine the benefits and challenges of generative AIs for many different professionals working in multimedia industries. Naturally, game developers stand to gain significantly from any research examining how generative AIs can save time, increase quality and foster creativity. The advantages extend to corporate managers and decision-makers as they can capitalize on understanding this technology and the role it can serve in their strategic planning,

ongoing projects, and recruitment strategies. Independent developers (more often called indie developers) can also find substantial value as generative AIs can supplement the skills of the team and produce content that might otherwise be out of reach or challenging to create. Currently many aspects of the process of using generative AIs in game development are unclear, controversial, or uncharted.

There are numerous supposed challenges and uncertainties to consider given the early stage of generative AI technology. On occasion, these systems can behave unexpectedly and create challenges for users. For instance, they might interpret prompts extremely literally or deviate unpredictably from the expected output. There are, at this moment, no established guidelines, or standards for what constitutes effective prompt engineering, nor what high-quality prompt engineering looks like. The quality of the output can be hard to evaluate, and the output might deviate from the context. Thus, there exists value in researching the challenges in conjunction with the benefits.

The aim of this research is to clear out and clarify some of the supposed benefits and examine the possible challenges encountered, thus making it easier for the people working in game and related industries to start implementing generative AIs into their working process. I elaborate this in the next chapter “Research questions”.

## **1.2 Research questions**

In this chapter, the research questions are presented, along with an explanation of the underlying rationale for their formulation. The research questions address several characteristics of the AIs generated output, and one of the most interesting questions is how we can utilize generative AIs as tools for developers. I will also examine the capabilities of generative AIs in terms of quality, ease of staying in preset context, prompt engineering effort needed to reach desired output, and the manual effort required to make the output work in the context. All of these factors are critical in multimedia and game development. In addition, it is important to examine any best practices or processes that might emerge during the course of the research, as they can be helpful for developers and others utilizing generative AIs.

Before it is sensible to delve deeper into the intricacies of the generative AIs as tools, it is imperative to determine the benefits of using such tools. As with every tool, there usually exist challenges and some considerations that are worth noting as well as obvious benefits. The question here is not if we can use generative AIs as tools, because it is quite easily determined that the current quality of the outputs is high enough that

they can be used, but rather how the multimedia industries or developers would benefit from using them and what are the considerations and challenges of using them. Thus, we arrive at our main question: How can utilizing AI-generated content, such as text, images, and audio, benefit multimedia and game development and what are the challenges and considerations?

Understanding the key factors influencing the quality of the output of generative AIs is paramount for the development of multimedia and games. Quality consists of many factors, but I am most interested in hedonistic quality: In essence, is the output visually pleasing, audibly appealing, or is the text engaging and well-composed. However, it has to be noted that the quality of art and media can be subjective and challenging to precisely determine. Because the quality is so multifaceted, there is also a need to delve deeper into the different aspects to understand the whole, thus, this second research question is further expanded into three follow-up questions. The primary focus of this research question is what key factors influence the quality, and to determine the factors that are contributing to increase or decrease of the quality. Thus, the need for the second research question “What are the key factors influencing the quality of AI-generated content in the context of multimedia and game development?”.

Generating a single image or a short text is often insufficient for multimedia or game developers. The ability of AIs to generate content that aligns with the context of the product under development is critical. When determining if the AIs can really be usable for professional developers, it's vital to assess their ability to create content that is contextually coherent within the project's narrative or theme and can serve in the final product, not just as placeholder. For instance, when generating stories, it is essential to stay in context of the established narrative without introducing unintended paradoxes, irrelevant themes, or contradictions. Therefore, human interaction is required to ensure that the AI adheres in preset context. My attempt is to generate material that stays in the context and is usable as end product, and not just as placeholder.

One of the significant challenges in AI-generated content is maintaining consistency throughout the development process. Currently, there are no mechanisms that I am aware of that enable AI models to retain a memory of all the content previously accepted into the product. For example, ChatGPT can remember some of what is entered, but at some point, it will start to forget and will need to be reminded (Wiggers, 2023). Currently the user cannot make it remember everything from large amounts of text. Likewise, with image-generating AIs, they can learn what a specific person or item looks like and generate different images of that element. However, if an interesting



element appears in one of the generated images, replicating that element in another image can be impossible, or at least extremely time-consuming and cumbersome, using AIs alone. Thus, the need for the follow-up research question “How effectively does AI-generated content maintain contextual coherence in the context of multimedia and game development?”

Estimating the effort required in prompt engineering is a complex task. There are two main factors to consider when generating usable content for the product: First is the time needed to generate and sort through the content, and the second is the ingenuity needed to craft the prompt in such a way that the AI understand it I would term it as “creativity-driven prompt engineering”.

One alleged challenge in prompt engineering is that AIs can be unpredictable in their responses and the prompt engineer has to employ creative strategies to achieve desired results. For instance, generating an image of a pen inside a glass jar can yield very volatile results. In these cases, it is up to the prompt engineer to figure out a prompt that deceives the AI to behave in the intended way. This process can be both time-consuming and demanding.

The other factor is time investment to sort through the generated media. The prompt engineer must examine every image and select the most desirable ones for further development or critically evaluate the generated text before deciding if it should be iterated, accepted, or excluded. The iterative process needed to generate usable content is time-consuming and the duration can increase proportionally with the complexity and requirements.

Additionally, the processing time required to generate the content is another thing that needs to be considered. Generating a series of images can take many hours. Given that prompt engineering is an iterative process, and involves trial and error, the content generated through the night might reveal itself to be unusable in the morning. A combination of the time needed to generate and sort through the content, and the “creativity-driven prompt engineering”, form a prompt engineering effort. This leads us to the following research question: R2b: “What are the challenges and considerations involved in prompt engineering to guide AI systems in producing desired outputs in multimedia and game development?”

Often, the output from the AIs must be manually edited. Depending on the media type of the output, this can be extremely hard to very simple process. For example, editing

text can be an easy and straightforward process, while changing the sound of a guitar on AI generated music track can be time-consuming and will probably require advanced knowledge on using a professional audio software. There can also be cases where the output from the AI is not directly suitable for the developer's needs, for example, game logo might need to be vectorized and thus, the AI generated content needs to be expanded and scaled larger, and possibly modified.

Manual effort will always be needed when dealing with AI generated content. The question is how much manual manipulation is needed and is the effort justified. Thus, the third follow-up question is warranted: R2c: "What is the extent of manual effort required for refining AI-generated content in terms of editing, modification, and adaptation to meet the specific needs of developers, particularly in multimedia and game development contexts?".

Lastly, there is a need to clarify and research the processes and best-practices of generating content with the AIs. As the landscape of content creation with AIs is still in its formative stages, there exists an absence of established processes. Thus, it is imperative that any processes, iteration techniques and methodologies are investigated and brought to light. While investigating best practices or processes is not the central aim of this study, it is an invaluable component in the comprehensive understanding content generation with the AIs. The knowledge obtained throughout this research may prove valuable in guiding future endeavors or prove useful for future research on the subject. Thus, it would be remiss not to include the learnings that could contribute to the broader discussion and future research. Consequently, this necessitates the inclusion of a final research question: "R3: What best practices and processes, if any, are revealed in the development of the artifact and the generation of AI content within the context of multimedia and game development?".

The research questions:

R1: How can utilizing AI-generated content, such as text, images, and audio, benefit multimedia and game development and what are the challenges and considerations?

R2: What are the key factors influencing the quality of AI-generated content in the context of multimedia and game development?

R2a: How effectively does AI-generated content maintain contextual coherence in the context of multimedia and game development?

R2b: What are the challenges and considerations involved in prompt engineering to guide AI systems in producing desired outputs in multimedia and game development?

R2c: What is the extent of manual effort required for refining AI-generated content in terms of editing, modification, and adaptation to meet the specific needs of developers, particularly in multimedia and game development contexts?

R3: What best practices and processes, if any, are revealed in the development of the artifact and the generation of AI content within the context of multimedia and game development?

### **1.3 Scope of work**

In this chapter, I discuss the limitations, restrictions, and scope of this research. I explore the process of employing GAI for content creation within game development, with a focus on identifying the associated challenges and benefits. My focus is on the aspects of game design and development, especially image, sound and text content creation methods and processes. However, 3D components such as models, animations, or effects are not included, as they would expand the scope of this research too broadly and add complexity. Similarly, sound effects and speech are also excluded to maintain a manageable research scope.

This research does not delve into the ethical discussions or considerations related to using generative AIs or their output. While I do acknowledge that ethical considerations are important in all areas of life, including game development, it is not beneficial for this research to explore ethical considerations. Doing so would divert the attention from the primary research questions. The exclusion of ethical aspects does not affect the findings of this research, as this research is primarily oriented towards the practical aspects of utilizing AI-generated content in game development, rather than the ethical implications that may arise.

Moreover, the in-depth analysis of the mechanics of different generative artificial intelligence models is not included. There are many types of technical differences and methods used on training the different AI models (Jovanovic & Campbell, 2022), but they are not discussed in this research. This research is more interested in the resulting output and capabilities of the content generated than how that output was technically made possible.

This research does not take a stand on copyright laws and the focus is not to explore the details of legislation. However, it should be acknowledged that the process undertaken here may unintentionally uncover information that could be valuable for future investigations into copyright law. This statement is supported by the fact that the process of creating AI generated content is one of the main emphases of this research. As Epstein & Hertzmann (2023) noted: *“But how substantial must users’ creative influence be for them to claim ownership? These questions involve studying the creative process of using AI-based tools and may become more complex if users gain more direct control”*.

Overall, the focus of this research deals with the output of generative AIs - be it text, images, or sound. The tools used in this research include Unreal Engine 5, ChatGPT 4, Stable Diffusion, Midjourney and AIVA.

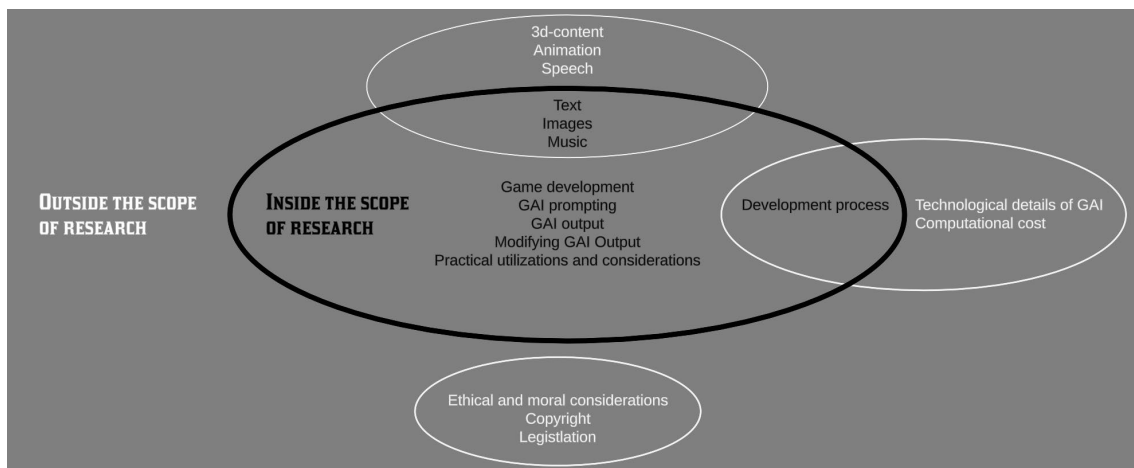


Figure 1 – Scope of work illustrated

#### 1.4 Structure of thesis

In this chapter, I will explain the structure of the thesis. Chapter 1, Introduction, contains all the relevant information to understand the scope and general info of what this research is about. Chapters 2 and 3 provide the theoretical background for the work. This background lays a foundational understanding of the topic but also illuminates many aspects discussed in the research and discussion section, thus effectively linking this research to the ongoing scientific discussion. Chapter 4 introduces the details of the Design Science Research Method (DSRM) and highlights the process undertaken in this research. Chapter 5 contains the actual research, introducing the creation of the artifact, how it was used, and how the game prototype was developed and designed. In addition

to this, the text, image, and music generation processes are explained, along with observations made.

The empirical conclusion at the end of Chapter 5 is not to be confused with the answers to research questions. While valuable, they are conclusions derived from the process of generating content for the artifact and thus warrant deeper discussion to connect the conclusions to a deeper discussion about the results. Chapter 6 discusses the empirical conclusions and observations made during the research, attempting to give the reader a holistic understanding of the results of the research as well as offering some final thoughts and speculative future directions of GAI technologies.

## **2 Game development**

The focus of this section is directed towards game multimedia content development and game design, as they are closely intertwined. The coding and development process, including methodologies like scrum, is intentionally given less emphasis, as they are less relevant to the research questions addressed in this study.

### **2.1 Concise history of video games**

The history of video games includes various narratives, some emphasizing technological advancements, while others explore the cultural significance of games. A notable source that offers insight into many different written histories is “Game On to Game After: Sources for Video Game History” (Nyitray, 2019). In this section, I aim to provide a concise history of video games. Understanding the current landscape of video gaming requires a historical context, both to recognize our present standing and to consider the potential influence of generative AIs as one of the many directions the industry might explore and expand into in the future.

Records of games can be traced back to as early as 2600-2400 BC in Mesopotamia. The oldest known board-game is “The Game of Ur” that was excavated in the 1920’s (*Top 10 Historical Board Games*, 2023). The game was later adapted into a virtual reality format in 2021 by three researchers (Pietroszek et al., 2021).

While the roots of gaming are ancient, this research emphasizes the digital evolution of games. The pioneering days of video games from 1950s to 1970s is often associated with titles like “Pong” (Atari, 1972) and “Spacewar!” (Russell, 1962). However,

pioneers like “Nimrod” and “Bertie the Brain” are worthy of mention. Built for exhibitions, Nimrod was showcased at the 1951 Festival of Britain, and Bertie was displayed at the 1950 Canadian National Exhibition. These custom-built computers, using light bulbs as display elements, were among the first to introduce interactive digital gaming to the public. Nimrod was designed to challenge players in a game of Nim, a mathematical strategy game (Baker, 2010), while Bertie offered a game of tic-tac-toe, complete with varying difficulty levels (Bateman, 2014). Both of these machines were dismantled after their respective exhibitions (though Nimrod was also presented in the Berlin Industrial Show before disassembly).

Transitioning from the pioneering days of the 1950s to the 1970s, the video game industry witnessed two pivotal moments. The first was in 1972 when Magnavox released its first gaming console to general public in the United States, marking the birth of first-generation consoles. The second was the rise of arcade gaming. From 1970s to 1980s is often considered to be the golden era of Arcade games. The phenomenon of arcade halls was prevalent especially in USA and Japan. Landmark games from this era include titles such as of Pac-man (Namco, 1980) and Space Invaders (Nintendo, 1985). While arcade games initially overshadowed home consoles due to their technological superiority and the widespread appeal of arcade halls, home consoles gradually gained traction and grew in popularity.

The culmination of the golden era is marked by an event often referred to as "The Crash." This term signifies a period when consumer confidence plummeted, leading the video game market to lose over 90% of its value between 1983 and 1985. The reasons were varied. One of the main culprits was business practices that flooded the market with poor-quality, yet expensive games and consoles. Separately, arcade games faced challenges due to the oversaturation of arcade halls. Many US based companies exited the industry or went bankrupt. This downturn paved the way for Japan to emerge as a dominant force in the video game industry.

Japanese dominance in the industry became evident in 1985, when Nintendo released Nintendo Entertainment system (NES) in the USA. Already released in Japan in 1983, the console was bundled with Super Mario Bros (Nintendo, 1985) and was very successful in Japan and USA. Personal Computers (PC) also entered into the market as Commodore 64 was released in 1982 and Amiga 500 in 1987. The handheld gaming also evolved into the new direction from handheld electronic devices of old when Game Boy was released in 1989 together with Tetris (Pajitnov, 1985) to make the Game Boy

even more desirable purchase. Despite the crash, the 1980s brought digital gaming into many people's homes and set the stage for the console war of 1990s.

The console war was the highlight of the late 80s and early 90s when Sega attempted to compete with the Nintendo-dominated console market. Sega's flagship products were the Sega Master System (1988) and Sega Mega Drive (1990). Sonic the Hedgehog, introduced in 1991, became the iconic face of Sega during this period. The console war of the early 90s is often considered to have ended when Sony entered the market with the PlayStation (1994) console, capturing a significant market share.

During the era of console wars, there was a notable expansion in the variety of gaming genres, especially on the PC side. This period saw the emergence of adventure games exemplified by "The Secret of Monkey Island" (Lucasfilm, 1990), first-person shooters such as "Wolfenstein 3D" (id Software, 1992), city-building simulations like "SimCity" (Maxis, 1989), and role-playing games, notably "Ultima VII: The Black Gate" (Origin Systems, 1992) just to name a few games and genres.

Early 2000s saw the exponential growth and success of online gaming when the technology evolved from Multi-User Dungeons (MUDs) into full-fledged massively multiplayer online games (MMO's). A prime example of this evolution is DeviousMUD, which was a precursor to what would later become RuneScape (Jagex, 2001). Developed in 1998, DeviousMUD laid the foundational mechanics and concepts that would inspire a new generation of MMOs. This era also witnessed the launch of iconic games like World of Warcraft (Blizzard Entertainment, 2004) and EVE Online (CCP Games, 2003), both of which further solidified the MMO genre's prominence and set new standards for player interaction and world-building in online gaming.

In the late 2000s and early 2010s, another significant change occurred in the gaming landscape: the rise of mobile gaming. This shift was largely driven by the introduction of advanced smartphones, most notably with Apple's 2007 launch of the iPhone, followed by the introduction of the App Store. As a result, mobile gaming quickly gained widespread popularity. Titles such as Angry Birds (Rovio, 2009) and Candy Crush Saga (King, 2012) stood out, highlighting the appeal and growth of casual gaming on mobile devices. At the same time, social media platforms, including Facebook, began incorporating casual games, broadening their reach and accessibility to vast audiences.

The 2010s can be argued to still be too close to present day to properly evaluate how the decade fits into the history of video games. Nonetheless, this period was marked by advancements in Extended Reality (XR), with devices like the Oculus Rift becoming available to consumers. Mobile games began to harness augmented reality features, as seen in titles such as Pokémon Go (Niantic, 2016). Esports, with games like League of Legends (Riot Games, 2009) and Counter-Strike: Global Offensive (Valve, 2012), attracted millions of viewers, a phenomenon amplified by the growing popularity of streaming platforms. Furthermore, the decade saw a notable rise in independent game development. Enhanced tools and resources empowered smaller teams and individual developers to create unique gaming experiences in ways not possible before.

Predicting the defining themes of the 2020s is challenging, if not impossible. Regardless, I will entertain some thoughts here that are purely speculation. One of the main discussion points in the industry revolves around development tools. The production of games, particularly “triple-A” titles, is becoming progressively more expensive and time-intensive. This is concerning for the future of the industry as development duration extends and the risks of game production escalate. In April 2022 Unreal Engine 5 was released and with it many new technologies (mainly Nanite, Lumen and Niagara) that further push the capabilities of 3d-games, by unlocking previously unattainable details and scale on 3d textures, lighting, and physics.

Other game engines and development tools are also rapidly evolving. This in itself would not be enough to determine the decade’s theme historically, as the tools have always evolved. However, generative AI tools are also rapidly evolving and changing how humans can generate multimedia. Combining these two factors, the decade might just witness a new era of developing games with GAI-Human collaboration. Where the GAI is integrated into the development tools to handle the routine tasks, help with problem solving, further present the human choices, and rapidly help generate the content as per instructions. It is not often that the work done by developers is forefront runner to define entertainment industry’s era, and there is real possibility that it is not going to be the case for 2020s either, but the chances for it to happen are higher than maybe ever.

As noted earlier, while the history of video games is not the central focus of this research, it provides essential context for understanding the evolution of the gaming industry. This context is crucial when considering the potential impact of GAIs. Introducing GAIs to the development process—and within the games themselves—has



implications beyond mere cost and time savings. It has the potential to fundamentally transform our current understanding of gaming.

## 2.2 Game Engines

A tiny glimpse on past and current game engines, as well a definition, is offered in the research conducted Toftedahl & Engström (2019). They also present a model that explains the game production pipeline (Figure 2) and continue to define the game engines into two distinct categories: general purpose game engine (Unreal Engine and Unity) and special purpose game engine (RPG Maker and Twine). The difference between the two is that the special purpose game engines cater for specific game genre or genres, and general-purpose game engines have the flexibility to cater for many different genres. Thus, the engine chosen for a specific game often aligns with the genre's requirements, influencing not only the design nuances but also the integration and presentation of multimedia content.

This research delves into game engines as primary platforms where multimedia content will be inserted. Beyond mere file extensions, they introduce certain limitations or needs for the content. Game engines dictate technical specifications such as texture resolution and polygon count. They may necessitate content optimization for real-time performance, demand specialized workflows for asset integration, and require mechanisms to animate static assets. In game development, both human and AI-generated content must adhere to the constraints set by the chosen game engine.

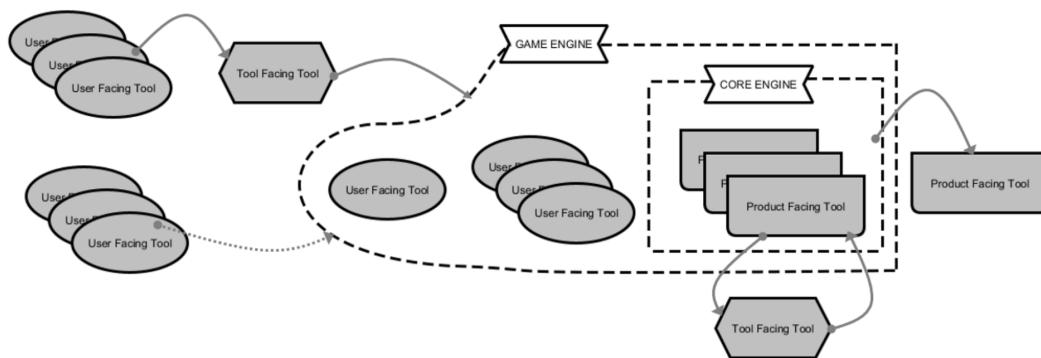


Figure 2 – Production pipeline. (Toftedahl & Engström, 2019)

*"Depending on the type of game engine, some parts of the production will be conducted with the user facing tools included in the game engine. Other parts will be running as separate applications". (Toftedahl & Engström, 2019)*

### **2.3 Game development tools**

While game engines come with some built-in tools for development, multimedia elements like 2D-images and audio are usually created using external software. The content created is designed to fit the engine's technical requirements and is later imported into or linked with the game engine.

Game engines themselves serve as tools, and a wide variety of in-house tools are developed both within gaming companies and by individual developers to support the game development process. Notably, legendary game designer John Romero, known for his work on titles like "Doom" and "Quake," reportedly said that "Tools live longer than games do." (GDC, 2021). This underscores the longevity of in-house tools, which often outlive the games they were initially designed for. Examples of such enduring tools include "TED," a tile editor (*TED5 - KeenWiki, 2023*), and "con edit," a conversation editor developed for the game "Deus Ex" (*Deus Ex Conversation Editor, 2023*). These tools continue to find utility.

In the context of this research, the specifics of in-house or individually developed game development tools will not be explored further. However, third-party tools for creating or editing multimedia content like images, sound, and text warrant closer attention. This is particularly relevant as media generated by GAI often cannot be directly implemented into the game engine; it may require modifications or even constitute just a small part of the overall content, or the quality needs to be further improved etc. With this foundational understanding of game development tools, we will next explore the specialized third-party tools that are essential for manipulating image, audio, and text content in games.

2d-images serve multiple functions in game development, from character portraits to concept art, textures, normal maps, and user interface elements. While Photoshop is often cited for its versatility, it is by no means the sole image tool employed in the industry. Many of today's imaging tools are not designed explicitly for game development but rather serve specialized functions. For instance, Krita is tailored for digital painting, whereas Inkscape and Illustrator are geared towards vector design and illustration. More specialized tools, like Substance 3D Painter for texturing 3d-models

or Lightroom for photo editing, also exist.

Depending on the scope of the project, the size of a studio's art department can range from a single individual to a team of hundreds. In the early years of video game development, artists were constrained by hardware limitations such as restricted color palettes, low screen resolutions, and limited hardware memory (Rogers, 2014). These constraints significantly impacted the level of detail and complexity that could be achieved in 2d-art. Today, while technological limitations still exist, they are far less restrictive for 2d-art. Advances in hardware and software now allow for high-resolution images, virtually unlimited color palettes, and complex layering and texturing techniques, pushing the boundaries of what is artistically possible. It is then evident that the tools for handling 2d-art have evolved to meet the modern demands of content development in this domain.

Software tools specifically designed for integrating and manipulating AI-generated images in the art and design domain are still in a developmental phase. These tools are increasingly being incorporated into existing art software platforms. As a result, they are not yet fully capable of harnessing the advanced capabilities of GAIs. Additionally, there is a human learning curve associated with these emerging art tools, which means it may take time for artists and designers to become sufficiently adept to fully exploit the potential of these advanced technologies.

Audio production in game development can range from straightforward tasks to complex undertakings, particularly when executed by larger teams aiming for the highest quality. While a comprehensive exploration of this subject is beyond the scope of the current research, a basic understanding is helpful for contextualizing the capabilities of GAIs in comparison to traditional methods. Audio content in games can be broadly categorized into three types: music, sound effects, and spoken dialogue. While the production process may require specialized hardware and technical expertise, the advent of Digital Audio Workstations (DAWs) and the MIDI protocol has democratized the field, enabling developers to engage in audio production using only software if necessary.

Digital Audio Workstations (DAWs) offer a diverse range of functionalities. Software like Reaper (Cockos, 2006), Pro Tools (Brooks & Gotcher, 1989), FL Studio (Dambrin, 1998), and Cubase (Steinberg, 1989) serve as notable examples. These DAWs have been in existence for an extended period and have evolved to meet modern industry standards. While DAWs offer a wide range of capabilities, they are not all-

encompassing in the context of game development. For instance, they generally do not handle real-time audio manipulation or interactive audio elements, tasks that are often crucial in creating an immersive gaming experience. These specialized aspects of audio production are usually managed within the game engine itself. To further illustrate the matter: Even MIDI sounds, which can be manipulated within DAWs, are often based on real-world recordings. This highlights the complexity of audio production in games and underscores the potential role of GAI in streamlining and enhancing this intricate process.

The tools for writing video games are tied to personal preference of the writer. Depending on the project size, there can be multiple writers involved. It often makes no difference what software or cloud service is used to write the story or dialogue of the game provided that the text is easily shared with the team. That said, many writers use smaller tools to help the writing and the tools can vary from specialized software designed for scriptwriting or branching narratives. For clarifications sake: Tools like *inky (Ink - Inkle's Narrative Scripting Language, n.d.)* and *articy:draft 3 (Articy Draft, n.d.)* can serve as example of the tools the writers might use to construct the narrative and keep up with the whole picture of the story. Other tools might include brainstorming tools or prototyping tools that are used in variety of way, but it is beneficial to understand that the tools are extremely varied and most often selected by writer for his own purposes or the game genre and type might require additional tools to be introduced to development cycle to keep everything organized.

In game development, the number of writers often varies depending on the project size. Writers often select tools based on their own preferences and the choice of software for writing the story or dialogue is usually flexible. In addition to the main writing software, some writers and projects might use smaller tools to help the writing and the tools can vary from specialized software designed for scriptwriting or branching narratives. Examples of specialized software used for this purpose include *inky (Ink - Inkle's Narrative Scripting Language, n.d.)* and *articy:draft 3 (Articy Draft, n.d.)*. These tools help in constructing the narrative and maintaining an overview of the story. It is important to keep in mind that there are a lot of tools that can be used in a variety of ways to support the writing process, and the examples here do not sufficiently convey the whole array or diversity of the tools available. In the end, the selection of the tools is always up to the project requirements and writers' personal preferences.

Text-to-Text GAI are very relevant for the field of entertainment and fiction writing. The full capabilities are yet to be discovered, and as with the image and audio, the tools

to use text-to-text GAI's are developing. It is undoubtedly not enough for game industries needs to merely generate well-crafted random stories or text. Other tools revolving around narrative harmonization, staying in the context of the game world, and ensuring character consistency are also essential for integrating GAI-generated content effectively.

Lastly, although tools for managing the software development process and production are important and commonly used in game development, they are not relevant for this research.

## **2.4 Game Design**

This section provides an overview of the role of game design within the broader context of game development. The design of the artifact used in this research is introduced in later chapters. The aim of this section is to give the reader a foundation for understanding the relationship between game design and the content created or generated. The reason why this is relevant is that game design serves not just as a directing force but as the foundational element of game development. It establishes the framework and vision within which all content—be it narrative, aesthetics, or gameplay mechanics—must operate. This foundational role ensures that the game's various elements unify into the intended experience. Failure to align content with the game's design framework can result in a disjointed experience, meaning that the gameplay elements, narrative, and aesthetics may not cohesively work together, disturbing or even destroying the intended experience.

Game designer as a role is not self-explanatory, as there might be many types of game designers in the big studio, while in indie game development the designer(s) is naturally more often involved in technical implementations or content creation and might work alone or in very small group. As evidenced in the book “The craft and science of game design” written by Philippe O’Connor (2020), several different roles that could be considered designer roles exist in the industry. In the book O’Connor presents various designer roles, ranging from design manager and creative director to interface designer and character designer. O’Connor also observes that “Game designer’s role is not universally defined from studio to studio” (O’Connor, 2020) highlighting the diversity of the roles designer might act.

The term “game design” is sometimes mistakenly attributed solely to the role of the

game designer. However, game design is more accurately understood as a guiding principle that shapes the entire project, and it is rarely the responsibility of a single designer. While the role of a game designer can vary widely—from specialized tasks in large studios to a multidisciplinary requirement in indie settings—the concept of game design is a separate entity. This is beneficial to grasp as this research aims to approach game design holistically rather than through the lens of some pre-defined role of game designer.

*“The designer’s role is to conceive and collect the best ideas, communicate them to the team in a sustained and consistent level, then shepherd those ideas through the development process until they come out as fully mature and polished game features.”*  
(O'Connor, 2020, p. 30).

Michael Salmond’s book “Video Game Design: Principles and Practices from the Ground Up” (Salmond, 2016) outlines the fundamental components of video game design, emphasizing that there is no definitive formula for success (Figure 3). While the building blocks presented in the book are not an exhaustive list of elements in video game design, they offer a valuable glimpse into the areas that must be considered in the design process.

Aesthetics Ae					Art Ar	Coding Co
Fun Fu					Interface In	Strategy St
Character Cr	Addiction Ad	Story St	Mechanic Me	Genre Ge	Reward Rw	Planning Pl
Levels Lv	Audio Au	Originality Or	Feel Fe	Pacing Pa	Feedback Fe	Testing Te

Figure 3 – Formula for amazing video game (there is no such formula).

In the field of ludology, also known as game studies, the concept of the “magic circle” could be simplified to mean that when players enter the play or a game, they willingly subject themselves to the rules of the play. Thus, a boundary is created between the external world and the game world, where inside of the magic circle, the norms and expectations are set aside. The magic circle establishes a definitive boundary where all dimensions of the game reside. This in turn means that everything inside the boundary is touched by game design, and that it is desirable that all aspects align with the world created. So, in video games this means that the multimedia and other content existing in the game must adhere to the rules and concepts of the game that is created, in order to serve the all-encompassing god of the magic circle, the game design.

To even further elaborate the connection of game design and content created for video games, even a slight misalignment of the content from the game world can cause disruption of immersion for the player. A famous game designer Sid Meyer stated that the game happens inside the player’s head (GDC, 2016), implying that imagination is one of the most important tools in game designers arsenal. Not everything needs to be presented in an animated and highly detailed way, and sometimes it’s a lot better decision from a game designer to let the players imagination take the main stage. This however means that there should not be elements presented to the player, that contradicts the happenings in the game world, as this might be disruptive and restrictive for the imagination of the player.

Designing games involves crafting worlds, a process where GAI’s can assist in software development and content creation. However, GAI’s currently lack the capacity for vision and passion. These elements are emphasized by Scott Rogers as crucial to game design (Rogers, 2014, p. 35) and they are intrinsic to human developers. At this point in time, they are not replicable by artificial intelligence, thus the game design is for now a human endeavor.

The ultimate purpose of the game design is to serve players. To craft experiences that are not necessarily achievable or possible in the real world. This is where the fun is in the games. They offer an opportunity for players to experience feelings. In the industry, the concept of "chasing the fun" is prevalent, where designers aim to identify elements in their game that are fun for players. However, Scott Rogers contends that fun is a subjective quality, much like humor (Rogers, 2014. p. 40). Rogers then continues to elaborate and presents a theory of un-fun that’s purpose is to give developers a way to identify fun, even when they have tested and played the game so much, that the fun has diluted and is not there for them anymore. The theory of un-fun states that “Start with a

‘fun’ idea. As you develop the game, if you find something in the game that is not fun (or [is] un-fun), remove it. After you have removed all the un-fun, all that should be left is the fun.” (Rogers, 2014. p. 41). Artificial intelligences cannot evaluate fun, or un-fun, but what they can do is help the developers to create content that is fun, or not un-fun, for the players.

In the future artificial intelligence tools, including GAIs, are likely to be more prevalent in software and game development. As a result, it's reasonable to expect these tools to be integrated into the daily work of game designers. While artificial intelligence will play a significant role for streamlining the development process, helping with communication between and inside the teams, generating multimedia content, and many other tasks, the essence of game design will permanently be shaped by human creativity and insight.

### **3 Generative artificial intelligences**

In this chapter, the concept of generative artificial intelligence (GAI) will be introduced as well as the specific GAIs used in this research. The aim is not to delve into excessive technical details; however, a basic level of technical understanding is needed for comprehending the distinctions among various GAI models and the features unique to each of them. To that end, an initial examination of the basics of machine learning and neural networks is warranted.

#### **3.1 Generative Model**

The term that arguably serves as an overarching category for various forms of GAIs is "machine learning." In the domain of data science, machine learning is employed for a range of applications, from rudimentary algorithms to intricate systems. “Neural networks” often have a large number of interconnected nodes, which is a structure that resembles the functioning of the human brain. Neural networks with both input and output typically consist of at least three layers: an input layer, a processing or hidden layer, and an output layer. The term “deep learning” is derived from the presence of multiple hidden layers between the input and output, which can vary widely in number depending on the specific architecture and application. (*What Is Machine Learning?*, n.d.).

GAI neural networks can be further classified based on both their architectural design



and their specific use-cases. A basic example of this classification is depicted in Figure 4, sourced directly from the work of Jovanovic & Campbell (2022). While Figure 4 is not an exhaustive list of all neural network types, the inclusion of Generative Pre-trained Transformer models (GPTs) and Generative Diffusion Models (GDMs) in Figure 4 is particularly relevant to this research, as I utilize ChatGPT-4 (a GPT model) and Midjourney (a GDM model).

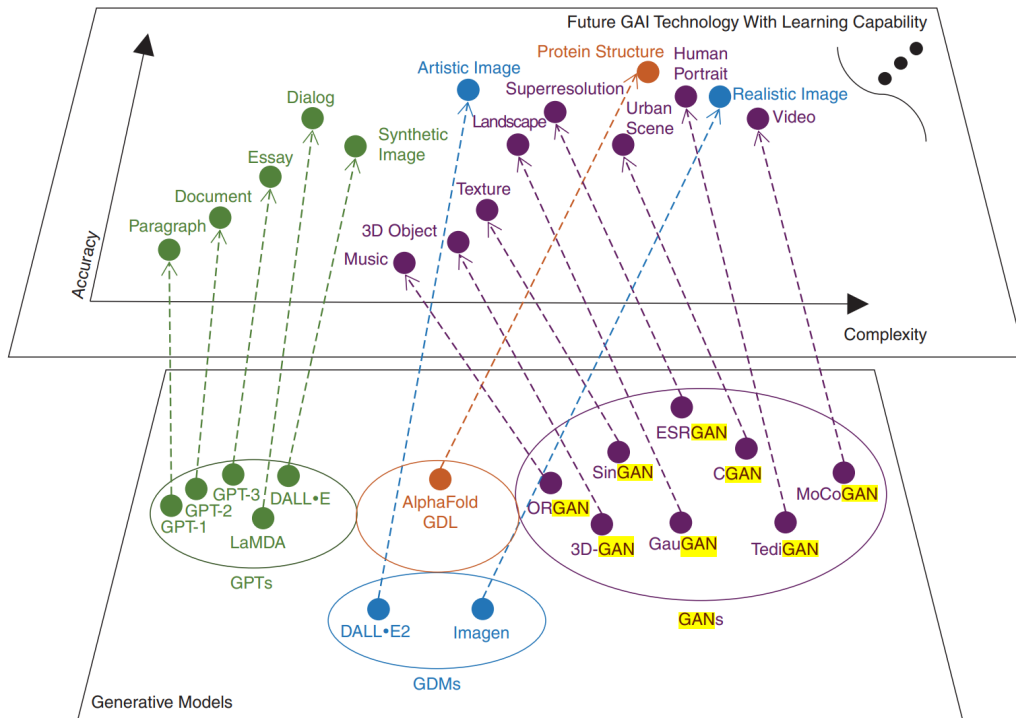


Figure 4 – Non-exhaustive representation of different generative models (Jovanovic & Campbell, 2022).

### 3.2 Defining GAIs and Assessing Current Research

In the realm of game development, Generative Pre-trained Transformer models (GPTs) and Generative Diffusion Models (GDMs) are not the sole types of GAIs with applicability; numerous other GAIs are designed for multimedia tasks or offer other advantageous functionalities. Given the variety of neural network systems, it becomes essential to provide a high-level definition that clearly specifies what falls under the category of GAIs and what does not.

One definition of generative artificial intelligence is offered by Lim et al (2023) in their research concerning education and GAIs. They state, “Generative AI can be defined as a technology that leverages deep learning models to generate human-like content in

response to complex and varied prompts.”. The provided definition serves as a valuable point of reference but may not fully capture the range of generative AI's capabilities. Specifically, the term "generate human-like content" is somewhat limiting, as GAIs can produce a diverse array of outputs, not all of which necessarily mimic human characteristics. Additionally, the requirement for "complex and varied" prompts is not universally applicable, as GAIs can in some cases also function effectively with simpler inputs. A more encompassing definition would better reflect the multifaceted functionalities and applications of generative AI. A technical definition may exist in other sources, but such definition may not serve the purpose of this research. Therefore, providing a tailored definition seems appropriate:

***“Generative Artificial Intelligence (GAI) refers to a computational model that has been trained using machine learning to generate media or data as an output and accept a set of instructions, individual media types, or a combination of these as an input.”***

It is imperative to note, that while the definition does not mention neural networks, they are, as of this point in time, what generative AIs are. There are very few exceptions to this rule and those fall outside of the scope of this research. The direct mention of them is left out to ensure the usability of the definition if future developments evolve to different direction from neural networks.

While technical research on GAIs exists, the literature focused on the use and generation of their outputs, as well as human-technology interaction aspects, is relatively sparse. In terms of research philosophy, this situates the field of examining GAI applications and output generation in a pre-paradigmatic phase, as outlined in Thomas Kuhn's Paradigm Shift Theory introduced in "The Structure of Scientific Revolutions" (1962). This state is understandable, given that, prior to 2022, the quality of GAI-generated content generally did not reach a level where it could be easily mistaken for human-created work, although there are notable exceptions such as AIVA's music compositions.

Understanding that the field is in a pre-paradigmatic phase is relevant for this research as we examine GAIs more closely. Much of the current knowledge on GAI capabilities and best practices is held by industry professionals and enthusiasts, rather than being documented in academic papers. Fortunately, the field is characterized by a culture of openness and collaboration. Information is often available through online videos and community platforms like Discord, as well as machine learning repositories such as GitHub and Hugging Face. This is further supported by the release of open-source GAI

models, such as Stable Diffusion and LLaMA. However, it's worth noting that these sources can sometimes contain misinformation or incomplete advice, particularly concerning prompting and prompting syntax (these are discussed later in this chapter), due to either misunderstandings or gaps in documentation. Without these sources, writing about GAIs in a scientifically valid way would be, if not impossible, then possibly years behind the current developments.

### 3.3 Prompt Engineering

Prompting is essentially giving the input for the GAI model. Wong et al. (2023) offer a definition of prompting: “[*Prompt engineering*] is a practice where users modify the prompts and generative model hyperparameters.” This definition is accurate as it does not restrict the type of input given for the GAI, and effectively conveys that there might be parameters involved in the prompt.

Rather than conducting an exhaustive review of various GAIs, the more appropriate approach is to highlight the relationship between input and output as a fundamental mechanism governing the operation of GAIs, as already stated in the definition presented. This is showcased in Figure 5.

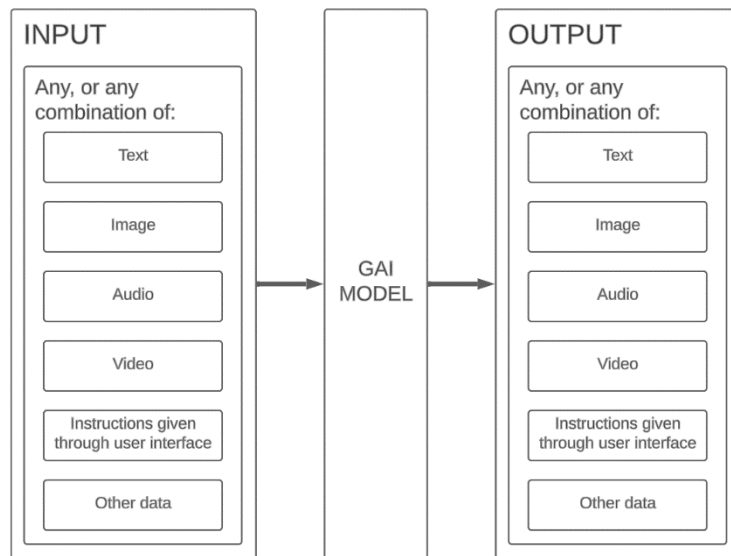


Figure 5 – Types of input and output of GAIs

The term “prompt engineering” can also include more than just user interactions with GAIs, such as developer-level tasks. This research focuses only on the user-centric aspects of prompting GAIs, as then the purpose is generation of high-quality and relevant content. There are a number of prompting techniques presented in scientific publications, most dealing with the large language models. Some of the techniques will

be presented in this chapter on relevant GAI sections.

The quality of the input is directly related to the quality of the output. Meaning that the prompt that is given for the GAI directly relates to the quality of the generated content. This fact surfaces in many scientific articles, and I argue that it is considered common knowledge in the field. For example, Lim et al. (2023, p. 11) state that “Generative AI is ‘dependent’ on the quantity and quality of prompts and the type of training it receives in delivering responses.”, while Li et al. (2023, p. 1) state that “... well-designed prompts have a huge impact on the generated result.”.

While there are different techniques and frameworks suggested for prompting, their effectiveness depends on the specific GAI model in question. It can be argued that there is no exact scientific method for prompting, even though there are techniques and frameworks that have been scientifically proven to improve the quality of the prompts. To elaborate, prompting can be considered an iterative process rather than a precise natural science. This is understandable, as the same prompt can yield a variety of unpredictable results depending on the prompt itself, seed number, and hardware used. This contrasts with natural sciences, where results are often more deterministic. The best results in prompting GAIs are obtained through multiple iterations. Many techniques and frameworks also rely on iterative approach.

### **3.4 Chat-GPT4 and large language models**

Large Language Models (LLMs) primarily function by accepting text as input and generating text as output. This research focuses on Chat-GPT4 and text-to-text prompting. As a sidenote, the direction of development is that LLM’s will form a base for multimodal GAIs as evident by OpenAI’s announcement that ChatGPT will have Dall-E3 image generation GAI integrated (*DALL·E 3*, n.d.).

While there is a wide array of LLMs available, an exhaustive list would not be beneficial in the context of this research. However, a brief overview of various LLMs and their characteristics is warranted, as it is imperative for game development that the right LLM is selected for the right task, as well as justifying my choice of Chat-GPT4 as my tool of choice.

I agree with Zheng et al. (2023) that it is of utmost importance that comprehensive evaluation tools for LLMs are developed. Without such evaluation tools it will get

harder and harder to continue the development of LLMs towards better performance. In addition to this, and especially related to this research, the selection of the right LLM for the task at hand gets harder if we do not understand the capabilities and characteristics of different LLMs.

In their research Zheng et al. (2023) propose an evaluation suite named GPT-Fathom. With the evaluation suite it is possible to evaluate the LLMs in the following categories: Knowledge, reasoning, comprehension, math, coding, multilingual and safety. The key point is that the GPT-Fathom evaluation method is the same for all different models, thus making the results comparable to each other. In their research they compared Llama 2, Claude 2 and Chat-GPT4 as well as OpenAI's other, legacy models. From their results it is evident that Chat-GPT4 consistently outperforms all other models.

It can be argued that higher scores on the GPT-Fathom evaluation suggest a more versatile Large Language Model. However, it's worth noting that GPT-Fathom does not specifically assess the creative writing capabilities of these models. While a relatively low score on GPT-Fathom would most likely correlate with less effective performance in creative writing tasks, the absence of this specific metric means that the evaluation is not fully comprehensive in the context of this research as the goal is to examine Chat-GPT's ability to generate the story text used in the artifact.

However, Wang et al. (2023) addressed the challenges of evaluating Large Language Models (LLMs) in the domain of personalized story evaluation. They introduced a personalized story evaluation model designed to infer individual reviewer preferences. Utilizing two specially curated datasets their evaluation framework included multi-dimensional criteria such as interestingness, adaptability, surprise, character development, and ending. Notably, their model demonstrated heightened performance over GPT-4 in Kendall correlation as well as certain other metrics. However, it's crucial to mention that GPT-4 did outperform the non-tuned LLaMA-2 70b model in most evaluation metrics. This study underscores the complexities involved in evaluating LLMs, particularly in tasks that require a nuanced understanding of creativity and subjectivity.

In light of these findings, the selection of Chat-GPT4 as the Large Language Model for story generation for the artifact is justified. As supplementary note to support the decision, I conducted a non-methodological, subjective comparison. Utilizing the Oobabooga web interface (oobabooga, 2023), I evaluated the cyberpunk narrative capabilities of LLaMA-2 7b and, to a lesser extent, LLaMA-2 14b against those of

Chat-GPT4. Hardware limitations prevented the testing of the LLaMA-2 70b model, resulting that it could not have been used in this research. Thus, I felt no imperative to undertake further, more scientific comparison. While the LLaMA models offered the flexibility to explore mature themes appropriate for the cyberpunk genre, the output quality from Chat-GPT4 demonstrated a higher level of writing craftsmanship, aligning more closely with the focus of this research, as well as my own preferences.

### **3.5 Text-to-Text prompting techniques**

In essence, text-to-text prompting differs when using the LLM model to solve a problem or when you want the model to perform creative writing or other creative endeavors. Whether the goal is problem-solving or embarking on creative writing tasks, the techniques employed can differ significantly. While both scenarios can leverage similar techniques, creative endeavors often benefit from precise, comprehensive instructions and an iterative process. In contrast, problem-solving tasks might be more effectively addressed by adhering to specific techniques or frameworks.

There exists multiple different techniques that are employed in text-to-text prompting. Notably, certain techniques, such as ToT (referenced in Figure 6), are best employed using coding instructions to guide the prompting process. Such methods fall outside of the scope of this research.

Zero-shot and Few-shot techniques are foundational in prompt engineering. The Raw method often does not contain the needed information to process the query correctly, and thus it is not that interesting resulting it being often discarded and not discussed. It is included in figure 6 as an example to illustrate zero-shot technique with more clarity.

Kojima et al. (2023) researched zero-shot prompting combined with Chain-of-Thought technique and found significant increase in accuracy of the models answers. The technique is to add instructive “Let’s think step by step” sentence at the start of the prompt (p.8.). This works especially well on complex calculations or other complicated problems. Any prompt that utilizes the LLMs ability to break the problem into the smaller parts can be confused to the CoT technique, but the key aspect is the instruction to entice the LLM to approach the problem step-by-step.

The Tree-of-Thoughts technique is presented in the research made by Long, (2023) as well as Yao et al. (2023). The technique in essence matches competing opinions against each other, and makes the created characters read other characters opinions. If any character finds that their argument fails, they will leave the discussion. The reason why

this is hard to implement manually is twofold. Firstly, the LLMs ability to read back on the single chat and “remember” what was previously discussed affects the techniques manual implementation. Secondly, complex discussions on complex problems with multiple characters can get very laborious and time-intensive to moderate.

Some or all of these techniques are needed when attempting creative writing using LLM. However, one prevalent technique is surprisingly simple and very widely accepted: Iteration. To clarify, iterating prompts is effective for prompting, but especially for story writing, as the LLM will often respond with some good, and some bad content in the same answer. Asking it to change some aspect or remove and add will effectively create a creative process cycle where human and LLM work in conjunction with each other. Iteration is the main component also in the other techniques, as they sometimes fail to complete the task, or complete it in an unforeseen way, or other problems arise. Iteration is a way to fix these arising problems and thus it can be argued that iterative process is the most fundamental of all techniques.

Technique	Description	Example user input
Raw	Unstructured text inputted into the LLM without additional instructions or examples.	"Translate the following: 'Hello, how are you?'"
Zero-shot	The user provides a task instruction without examples to the LLM	"Translate the following English text to French: 'Hello, how are you?'"
Few-shot	The user provides a task instruction along with a few examples to guide the LLM towards the desired output.	"Example1: 'Good morning' is 'Bonjour' in French. Example2: 'Thank you very much.' is 'Merci beaucoup' in French.  Translate the following English text to French: 'Hello, how are you?'"
Chain-of-Thought (can be combined with any mentioned before)	The user provides a task instruction stating that the model should examine the task step-by-step.	"Let's think step by step. I went to the black market and purchased 10 data chips. I handed over 2 data chips to the shady neighbor and gave 2 more to the repairman to fix apartment security. Afterward, I decided to acquire 5 additional data chips from a dealer. I went to a bar and threw one extra for the cab driver, but he refused and i spent two in the bar instead. The cabdrive cost 3 data chips. How many data chips do I have left in my inventory?"
Manual Tree-of-Thought* (can be combined with any mentioned before)	The user creates personas and makes them figure out best outcome.	Create two characters: a pre-teenage avid gamer and a parent. They will share their opinions on suitable gaming hours per day with one statement. [Answer] Both read the other's opinion, then present their case with one statement. If resolution is reached the character will fall silent. [Answer] Both read the other's opinion, then present their case with one statement. If resolution is reached the character will fall silent. [Answer] [*This is continued until solution presents itself]

Figure 6 – Some text-to-text prompting techniques. Note that tree-of-thought is not meant as manual prompting method, and it is inserted here to provide more context.

### 3.6 Diffusion Models

A detailed exploration of diffusion models would exceed the technical boundaries set by this research. However, understanding the fundamental workings of these models is crucial as such comprehension not only provides insight into their capabilities and limitations, but also elevates the approach beyond mere trial-and-error. It empowers prompt engineers to guide diffusion models towards specific outcomes and to challenge the models' boundaries. For an overview of the intricacies and breadth of diffusion models, refer to Figure 7 (Yang et al., 2023).





Figure 7 - Taxonomy of diffusion model variants and connections with other generative models, applications of diffusion models, and future directions. As detailed by Yang et al. (2023)

Diffusion models operate through a process that can be metaphorically described as follows: Consider the training data, which includes images and their text descriptions, being placed into an “imaginative bottle”. Before entering, noise is progressively introduced to the image until it becomes just noise. Alongside this transformation, the associated text is formatted and added to the bottle. This is done for all the training data. After the training of the bottle is complete, when a user prompts the model with text, the

bottle releases an image in its noisy state, which is then reconstructed step by step to form the intended image. This is basic description of the process does not capture the technical details of forward diffusion (the process of training) and backward diffusion (the process of reconstructing) completely. Figure 8 showcases the generative process of Midjourney. Broad technical explanation of the process can be found on the research by Cao et al. (2022).

There are various diffusion models available. Notably, these include Stable Diffusion from Stability AI (*Stable Diffusion*, n.d.), Midjourney (*Midjourney*, n.d.), and Dall-E 3 from OpenAI (*DALL·E 3*, n.d.). While they all are diffusion models, their characteristics differ from each other.

Dall-E 3 likely possesses the highest level of natural language prompt understanding, an assumption based on its utilization of chat-gpt4 for image descriptions and prompt handling.

Stable Diffusion, on the other hand, is open source, leading to the availability of various tools. These tools include the Web UI (AUTOMATIC1111, 2022) and the ControlNet extension (Illyasviel, 2023) for the Web UI, which offers enhanced control over the images. This positions Stable Diffusion as the most versatile and controllable diffusion model currently available.

Midjourney, an independent research laboratory, offers its product exclusively to paying customers. Based on my extensive experience using various diffusion models, I am confident to claim that, at the time of this writing, Midjourney delivers the highest quality of images. This assertion stems from my subjective evaluation and is not an absolute judgment. It's essential to recognize that while one diffusion model may excel in a specific domain, another might fall short in that same area and outperform in another. Even as I advocate for the superior quality of Midjourneys output, this perspective is rooted in my exposure to a particular set of image styles and types. It does not encompass a comprehensive assessment of all the capabilities of the three models mentioned. However, the fact that Midjourney surpassed both Stable Diffusion and Dall-E on user count when Midjourney version 5 was released (Wilson, 2023) supports the perception of field leading quality.

Comparing these models is not a straightforward process, primarily because each model handles input in its unique way. For example, while Stable Diffusion recognizes commas as separators, Midjourney does not. It's worth noting that there isn't any official

documentation from Midjourney that confirms this behavior, but such distinctions become clear when one engages with the models in practice.

The syntax for inputting parameters and text into these models differs from one to another. In other words, the same input does not behave the same way on different models. This difference not only complicates the task of comparing models but also highlights the need to grasp the nuances of each model's syntax to ensure optimal output. Syntax isn't merely about how to input parameters; it also involves how the model assesses the importance of words, how it correlates words with concepts, and more. While there are often basic guidelines on parameter usage, a deeper insight into the model's natural language processing (NLP) is notably lacking, especially in the case of Midjourney. The lack of detailed documentation on this matter is a significant concern. It adds unnecessary complexity to the prompting process, pushing users to adopt a trial-and-error approach. In the case of diffusion models, a trial-and-error approach does not necessarily reveal optimal methods or best practices to reach the highest quality output. Furthermore, there is no definitive way to assert that a particular method is superior, as it's not grounded in a clear understanding of the model's NLP. In the realm of diffusion models, as well as other GAIs, the quality of the output is directly tied to the input's quality. Thus, if the intricacies of the input are not understood by the user, no matter how minor those aspects might be, the model might never achieve its highest potential.



Figure 8 – Backward diffusion process of Midjourney in increments of [10,20,40,60,80,100] %.  
Prompt: finland forest nature trail --ar 21:9 --s 346 --c 17  
Seed: 2873300587

### 3.7 Models for Music

This section discusses the music models and methods. It has to be noted that generating effects and speech is not part of this research. The reason for this is that the scope of this research would get too complex and extensive. However, most music generation problems are the same in other audio generation. Additionally, some selected tools for speech and effect generation are mentioned in the “general observations” section of this chapter.

While utilizing neural networks to generate multimedia has surfaced as viable option in the last years, creating a computer or software that generated original pieces of music is not such a new phenomenon. Ada Lovelace wondered about machine created music as early as 1848 and Alan Turing created a computer that played pre-determined notes in 1951. However, the first generative computer was created by Lejaren Hiller and Leonard Isaacson in 1956-57 and their work, *The Illiac Suite: String Quartet No 4*, is a computer generated composition for a string quartet, traditionally consisting of two violins, a viola and a cello (Funk, 2018). Their method to create the composition was algorithmic rather than based on machine learning or neural networks.

While a brief historical context of computer/AI-generated music is useful to understand its origins and the evolution of human interest into the topic, a deep dive into its past is not the focus of this research. More important to this study is the current state of music and audio GAI. An excellent and significant contribution in this domain is the comprehensive systematic review of AI-based music generation by Miguel Civit, Javier Civit-Masot, Francisco Cuadrado, and Maria J. Escalona (2022). In their systematic review, it is discovered that most GAIs in the field of music are recurrent neural networks (RNN). Other notable architectures include those based on Variational Autoencoders (VAE), Transformers, rule-based systems, evolutionary algorithms, and feedforward (FF) networks. Generative adversarial networks (GAN) can be part of the generation system, or not. It is not uncommon for the generative models to mix and match architectures where it is possible and applicable in meaningful way.

A definite lack of focus on user interfaces of the generative AI models is also observed by Civit et al. (2022, p.8) resulting in systems that are not necessarily easy to use or user friendly. Only 12 of 139 papers included in their review took user interface into account. While they do not explicitly state this, their systematic review also highlights the lack of material about commercial products in the research field, such as Artificial Intelligence Virtual Artist (AIVA). This is apparent as the top five cited articles

included in the systematic review discussed DeepBach, MidiNet, MuseGan, Magenta Transformer, and Magenta MusicVAE. None of these five are commercial products.

Digital Audio Workstations (DAW) are the most common platforms that composers utilize for music production today Civit et al. (2022, p.8). As mentioned earlier, these workstations provide an extensive array of tools for recording, editing, mixing, and mastering audio. Additionally, DAWs enable composers to craft music using virtual instruments, allowing for the creation and composition of melodies and harmonies without the need for physical instruments. This MIDI-based functionality transforms the DAW into a versatile musical canvas, bridging the gap between traditional and digital music creation.

The integration of GAIs into Digital Audio Workstations (DAWs) presents a fascinating yet largely uncharted territory (this also is discussed in the research of Civit et al. (2022)). Given the already extensive nature of DAWs, adding another layer in the form of a GAI could introduce an even steeper learning curve for composers. However, it is enticing to consider transferring GAI outputs as MIDI directly into the DAW. This facilitates the establishment of two distinct yet interconnected environments, each having its specific purpose. While I am not opposed to the idea of integrating GAIs into DAWs, I would like to stress the potential challenges it could pose for users and the composition workflow. It is important that such integration is executed thoughtfully, ensuring that the two environments remain intuitive and distinct from each other.

Music GAIs can be categorically divided into types that generate audio, notations, MIDI, or a combination of these. For game development and producing music and other audio for games it is imperative that the audio is in a format that supports further manipulation easily. Presently, GAIs that exclusively produce music in audio format aren't directly suitable for game development purposes. This is primarily due to the presence of "artifact noise" in the generated music. Example of this could be Jukebox from OpenAI (*Jukebox*, n.d.) or more recent model Suno (*Suno AI*, n.d.) that is able to generate vocals alongside the music. While these models may not yet meet the specific needs of game development in their current state, it's essential to recognize them as significant milestones in the realm of AI-based music generation.

This leads to a realization that the currently viable way to generate music for the purpose of game development is by using GAI model that has an output in MIDI format. This does not only give the user an ability to transfer the track into DAW, but it also makes it possible to modify the output in any way the user deems suitable. By

utilizing tools such as virtual studio technology and VST instruments (*Steinberg, n.d.*), MIDI tracks can achieve very high audio quality.

With these considerations in mind, I deemed AIVA the best choice for this research. The MIDI output, extensive controls of the output, and an interface more intuitive than its counterparts were factors that influenced this selection. My decision was also informed by my personal background. While I do not identify as a professional musician, I have played the bass guitar on and off for over two decades, mainly in hard rock bands. This experience has given me a basic understanding of music theory. I've also used DAWs previously, mainly Steinberg Cubase. My familiarity with musical concepts and DAWs as tools further solidified my choice, especially given AIVA's interface that integrates elements of music theory and offers advanced track mixing capabilities.

It is notable that AIVA received funding from an EU project called Horizon 2020. The project further solidifies the selection of AIVA as it clearly states that AIVA opens up the field for game creators to use AI generated music (*AI Composers Create Music for Video Games, n.d.*). Additionally, the audio quality of the generated music further underscored AIVA's suitability for this study.

While I previously highlighted the top five researched models, it's worth noting that AIVA is not among them, as evidenced by its absence in the systematic overview conducted by Civit et al. (2022). This is most likely due to it being a commercial model and thus, the inner workings are not open source, making the model only partially accessible for researchers.

AIVA was introduced to the public in 2016. The following year, Nvidia's Developer blog shed light on AIVA's capabilities, revealing its foundation on neural network technologies and some other technical details. A significant milestone for AIVA came when it was accepted into SACEM, an authors' rights society based in France and Luxembourg. This acceptance effectively granted AIVA a global status as a recognized composer. (*AI Composer Creates Music for Films and Games, 2017*). Further testament to its capabilities was provided by Pierre Barreau, who, in a 2018 TED talk, delved into AIVA's prowess as a composer during that period (Barreau, 2018).

Over the years, AIVA has continually expanded its musical repertoire, including a broader range of genres. Reflecting on its evolution, it's noteworthy that in AIVA's early days, its compositions were often brought to life by human musicians, translating its

digital creations into live performances. Currently, users can listen to and interact with compositions generated by AIVA directly on its website or through its dedicated application. AIVA's YouTube channel acts as a repository, showcasing its musical creations, tutorials, and other related content (*Aiva - YouTube*, n.d.).

### **3.8 Intersection of GAIs and games**

While research that incorporates both generative AI and game design or development is not at this moment very prevalent, there is strong potential for growth. Research by French et al. (2023) showcases case studies from the School of Computing and Digital Media at London Metropolitan University. This research highlights undergraduate students' experiences using OpenAI GAI tools in their coursework, delving into the challenges they encountered and the broader educational and environmental implications of GAIs. Though not directly aligned with this study because of their educational view, their work offers initial perspectives on GAI's role in game development.

### **3.9 General observations**

This concluding subsection contains general observations of the state of the game design and GAI research field. The unreferenced views presented here are inherently subjective, given the vastness of the research field one individual is exploring. However, offering these general observations aims to enhance the reader's clarity and understanding of this research.

Firstly, LLMs appear to be the most extensively researched among all the GAIs, succeeded by diffusion models (and other image generation models), and then by generative music models. Confirming this, however, is rather difficult. Crafting precise search queries for database like arXiv.org is, in this case, prone to confirmation bias. Additionally, the terminology employed might span multiple disciplines or may not encompass all models associated with, for instance, image generation. The demand for systematic reviews is increasing as research into these fields continues to expand at an accelerating pace.

Secondly, it appears that the majority of research on these models is skewed towards technical metrics, with artistic and qualitative aspects often being overlooked. This bias is understandable, given the inherent challenges in evaluating a model's performance in



subjective areas like creative writing or photography. Nonetheless, there's a pressing need for more research from a humanistic perspective. While GAI developers are making remarkable strides, grounding the direction of quality improvements in research could ensure that these advancements align more closely with human values and preferences. This would ensure that the models evolve not merely as technological marvels, but as usable tools that foster opportunities and amplify human creativity.

Thirdly, while there is research on game design and production, much knowledge comes from books written by industry insiders. This could suggest that academic researchers find game design either hard to access, less appealing as a research topic, or challenging in unique ways. Interestingly, gaming culture seems to be more researched than the actual game development process. Nonetheless, the field of game development and design isn't entirely neglected, as demonstrated by the systematic review conducted by Aleem et al. (2016).

Lastly, even though audio effects and text-to-speech are left out of the scope of this research, it has to be mentioned that both of these aspects have also evolved in recent years to the state that it is warranted to look into their applications in multimedia environments. Audiogen (*Audiogen*, n.d.) serves as an example of “text-to-sound effect” model able to generate sounds and ElevenLabs (*ElevenLabs*, n.d.) has accomplished high quality text-to-speech model just to name a few.

## **4 Methods**

In this research, I employ a design science research methodology (DSRM). Originating from computer sciences, this methodology is relatively recent, mirroring the evolution of the broader fields of information technology and computer sciences.

March and Smith (1995) have identified a dichotomy in IT (information technology) research. This dichotomy arises from the differing objectives of natural science and design science. A particularly intriguing observation from their work is the notable absence of theories in design science, in stark contrast to natural science. Instead, the focus of design scientists is on devising methods, processes, and implementations. March and Smith's paper provides a comprehensive overview of the design science methodology, detailing its structure and how it can be applied in research.

“Whereas natural science tries to understand reality, design science attempts to create things that serve human purposes.” (March & Smith, 1995)

Alan R. Hevner and colleagues (2004) further developed the framework for design science methodology, specifically tailoring it for the field of information systems (IS). In their research, they identified two core paradigms in IS research: Behavioral science and design science. They went on to present seven guidelines for conducting research within the design science framework and showcased three distinct studies that have effectively utilized design science as their research methodology. Thus, furthering the work March and Smith started.

In their publication, “A Guide for Novice Researchers: Design and Development Research Methods” (2010), Timothy J. Ellis and Yair Levy delve into principles that align closely with design science. They reference the guidelines introduced by Hevner et al. (2004), but notably refer to the approach as “design and development research method”. Their emphasis on this terminology highlights a distinction in the field. The significance of their paper lies in its call to increase awareness of design science as a paradigm in IT and IS fields, pointing out that “design and development research” is a less established method in comparison to other methodologies.

Roel J. Wieringa (2014) provided a comprehensive exploration of design science in his book, “Design Science Methodology for Information Systems and Software Engineering.” In it, he delves into various aspects of the methodology, emphasizing the significance of stakeholders in the process and offering insights from the perspective of researchers. However, for the context of this research, while Wieringa’s book is extensive, its framework may not be entirely aligned due to its emphasis on stakeholders’ position and other aspects not relevant to this research. Also, some details of the activities presented might not directly fit the theme of this research.

In this research, the design science methodology applied is the one presented by Ken Peffers, Tuure Tuunanen, Marcus A. Rothenberger & Samir Chatterjee that is developed for the field of information systems (IS). The reason for selecting this exact method from all of the design science method frameworks and slightly varied adaptations is that in this method, the six-step structure is open enough to support the examination of the process of generating the AI content for the artifact. Most other design science methodology principles would suggest using theory forming in one of the activities or consider stakeholder participation extensively. Neither case applies very well in this research.

Additionally, it's worth noting that DSRM aligns with the foundational elements of well-established scientific principles, practices, and writing methods, such as the widely recognized IMRAD structure (Introduction, Methods, Results, and Discussion). The nucleus of DSRM lies in its emphasis on using artifacts to delve into research problems, particularly those that are intrinsically human and design-centric rather than purely technical or rooted in natural science. Another defining characteristic is the method's structured activity stages, which are iterative, allowing for revisitation and refinement as the research progresses. In conclusion, while the DSRM method is accommodating and not overly prescriptive, it aligns well with the objectives of this research.

#### **4.1 Selected design science method (DSRM)**

As previously mentioned, the DSRM bears a minor resemblance to the IMRAD writing structure. Thus, in more extensive compositions, such as this research endeavor, it's fitting that specific activities unfold as individual chapters, offering an in-depth exploration of the activity.

One notable factor in this research is that the artifact is not the actual subject of the research, as is usually the case in DSRM. Rather, the artifact serves as a platform for the AI-generated content, creating a frame where the content must fit. Without the artifact, there would be no defined limitations or requirements against which to evaluate the generated content, rendering the evaluation process invalid. Despite this, I argue that DSRM is still a superior method compared to other available methods for this research, as it offers a clear framework to follow and ensures systematic examination and validation of the GAI content evaluation.

This research follows the process of design science research introduced by Ken Peffers, Tuure Tuunanen, Marcus A. Rothenberger, and Samir Chatterjee (2007). There are six important steps in DSRM, and they are referred to as activities.

1. Identify problem and motivate: Define the problem and show its importance.
  2. Define objectives of a solution: Clarify the purpose of the artifact.
  3. Design and development: Create the artifact.
  4. Demonstration: Use the artifact to solve the problem.
  5. Evaluation: Observe effectiveness, evaluate, and if needed, iterate.
  6. Communications: Publish the results.
-

## **Realization of Activities**

### **Activity 1: Identifying problems and motivation**

- The problem identifying, and motivation are presented in Chapter 1. The importance of this research is also discussed in the same chapter.

### **Activity 2: Define objectives of a solution**

- The purpose of the artifact is to offer a platform for the AI generated content as showcased in chapter 5.1 that discusses the development and design of the artifact. The objectives of this research are presented in chapter 1.2.

### **Activity 3: Design and development**

- The design and development of the artifact is presented in chapter 5.1.

### **Activity 4: Demonstration**

- The demonstration is presented in chapter 5. Particularly in chapters 5.2 to 5.4.

### **Activity 5: Evaluation**

- These findings are presented in Chapter 6.

### **Activity 6: Communications**

- This research, as a whole, serves as a means to communicate the results.
- 

## **4.2 Other methods**

The concept of “interactive product quality” (IPQ), introduced by Marc Hassenzahl (2006) is recognized in the field of Human-Technology Interaction (HTI) for its approach to the evaluation of interactive products. Hassenzahl emphasizes the necessity to refine and evaluate the quality of such products more deeply. He advocates that IPQ evaluations should include not only traditional usability metrics but also the hedonic elements that contribute significantly to the user experience.

In this research the focus will concentrate mostly on what Hassenzahl calls “non-instrumental” quality factors. Non-instrumental IPQ factors include the hedonistic and experience related, often non-functional aspects of the product. In the context of this research, the product refers to the entirety of AI-generated content, whether that be

images, text, or music. Evaluating the hedonistic aspects of GAI output is not a straightforward endeavor.

*“Some GAI evaluations, like a personal rating of the output for utility, aesthetics, clarity, or similarity to real-world content, are inherently subjective and difficult to evaluate. Currently, there are no standard means to determine if a GAI is as realistic to a user as a non-GAI application.”* (Jovanovic & Campbell, 2022)

Currently, AI technology is advancing rapidly, and there is a confident argument to be made that the non-instrumental quality of AI output has reached a level where it is indistinguishable from human-generated work (Grierson, 2023), (OpenAI, 2023). In order to clarify the quality research aspect of this paper I will simply state that: “The quality of the output from the AI is high enough if it is difficult to determine if the output is created by human or not”.

As we have observed AI’s securing victories in human art competitions and composing for entire orchestras, the question is not if AIs can produce content of high quality, but rather what constitutes an acceptable standard for such output. This interpretation of IPQ is relevant to the evaluation activity of the Design Science Research Methodology (DSRM) employed in this study. Thus, I conclude that based on these factors, the output of the GAI either is or is not of acceptable quality. This binary approach aligns more effectively with the objectives of this research than engaging in an effort to numerically assess the quality of each GAI output. It is essential to understand that the binary approach is the preliminary step in content evaluation, not the conclusive one. Initially, the binary evaluation helps determine whether the content may be relevant for further consideration. Only after this stage is the content examined for other attributes, such as stylistic consistency and overall coherence, to assess its full suitability.

### **4.3 Recording the experiences**

An integral part of this research is the reflection of the generated contents quality and how to record the observations. Autoethnography is a method where the researcher is involved in the cultural setting and then proceeds to write about the experiences.

*“When researchers write autoethnographies, they seek to produce aesthetic and evocative thick descriptions of personal and interpersonal experience.”* (Ellis et al., 2010).

However, for the purposes of this research, autoethnography as a method might not be suitable for two reasons: First, autoethnographies are usually expected to convey a story of some type and secondly, they are usually accounts of researchers who enter into some cultural setting. In this research neither are necessary. I mention autoethnography despite its misalignment with our research goals to pre-emptively clarify its apparent, yet misleading, suitability.

While there exist many diary-keeping methodologies, and each have their set of characteristics, it must be recognized that none align precisely with the unique requirements of the present research. The specificity of this research calls for a more tailored approach that strives to reveal the findings of the process. This is partly because the researcher is also the user of the generative AIs, as well as evaluator of the quality.

Therefore, it's clear that the diary's structure and style should be designed to best support the recording of observations. Given the rapid iteration process, documenting every factor that emerges in each image or text generation would be disruptive to the workflow. Consequently, I found it best to adopt an alternative method of record-keeping. An audio diary emerged as the optimal choice due to its rapid nature in capturing thoughts for later refinement. The logging and transcription of the audio journals are facilitated by Whisper AI (Introducing Whisper, 2022). These audio diaries are then used to assist with writing the research findings section.

## **5 Research**

In this chapter, I will first discuss the artifact and then delve deeper into the research and findings. There is also a need to present some of the game design aspects, as these set requirements for the content. However, it would be nonsensical to delve into extreme details of some of the game design aspects, as the extreme details do not add additional value or insight and sometimes broader explanations are enough for the context of this research. This means that parts of the design process are either not discussed, or they are demonstrated only partially on those aspects that are relevant to this research.

It is notable that the Empirical Results chapter is based on the findings presented in this chapter. For the purpose of clarity and to employ a logical and easy to approach means

to present the findings of this research, the findings are divided into two distinct categories.

1. Empirical Observations (EO)
2. Primary Empirical Conclusions (PEC)

This chapter presents the empirical observations, while the primary empirical conclusions are discussed in the chapter on empirical results. The distinction between these two is simple: The EO consists of observations made while generating content for the artifact and the observations recorded in audio diaries. The PEC consists of summary conclusions and deductions drawn from the observations. Together these conclusions and observations attempt to answer the research questions.

## **5.1 The Artifact**

The artifact is a game prototype developed in Unreal Engine 5. The purpose of the artifact is to serve as a platform for the AI generated content, thus creating a framework and natural requirements for the content generated. In game development the multimedia content must always fit the theme and the game world, as disturbing this balance will cause the player to experience a break in immersion, a concept further elaborated in Chapter 2.4.

Without the artifact as a framework, GAI content would be assessed solely based on its individual appeal, without consideration for its relevance or fit within the narrative or game world context. Establishing an extremely strict set of written requirements for the content could prove to be overly restrictive and may not yield the desired outcomes. Precisely explaining these criteria is complex, as they must be flexible enough to foster creative diversity while still maintaining the game's cohesive experience. The artifact thus functions as both a canvas and a compass, enabling the GAI to create content that is not only high quality but also seamlessly woven into the game's fabric, supporting an engaging player experience.

Before delving deeper into the details of the game prototype, it is essential to understand that the prototype is not a full game or even a demo. It is closer to what would be called a minimum viable product (MVP) in software development. This does not hinder its applicability or usability, as the prototype contains the necessary functions to provide a suitable platform for the GAI content. The prototype is designed primarily to serve this research, meaning that some features are intentionally left unfinished. However, the

elements of the game that interact with the GAI content have been developed to a stage that ensures the prototype's functionality for research purposes.

The artifact is a cyberpunk storytelling game prototype. Its primary concept is influenced by the “Choose Your Own Adventure” books conceptualized by Edward Packard in the 1970’s and the interactive storytelling approach of the 2018 Netflix series, "Black Mirror: Bandersnatch" (Slade, 2018). Although the game’s structure is designed to be only partially non-linear, the central idea of allowing player to influence the narrative is the focal point of the user experience, providing both motivation and enticement. Additional inspiration comes from the Shadowrun series (*Shadowrun Trilogy - Paradox Interactive*, 2013-2015) by Harebrained Schemes, which contributes to the game's portrayal of agents, referred to as “Operators” in this prototype, and their handlers or “Fixers”, reflecting the familiar themes encountered in cyberpunk RPGs.

## **5.2 The Game design and mechanics of the Artifact**

Relevant details of the specific areas of the game design that deal with the generated content are explained in the following specific GAI content sections. This particular section is dedicated to the game design and mechanics of the artifact. While the relevance of this section could be questioned for the overarching research questions, such concerns are easily resolved: As evidenced in chapter concentrating on game development from theoretical view, the generated content must align with the game design. Thus, the relevance of this section is apparent.

The artifact is a 2D game. The player does not directly control any character like would be the case in 2d-platformers for example. Instead, the gameplay is entirely user interface-driven, involving the reading of provided text and making selections and decisions. The choices of the player are treated as actions that affect the game's progression. To elaborate on this, there are multiple screens that the player can view, such as Story, News, Operators, Briefing, Mission, all containing relevant data or decisions to be made. The objective of the game is to choose the right operators for the missions and thus progress in the game, revealing more of the main story. Choosing the right operators is done by reading their biography, deducing what skills they have and then reading the briefing of the mission.

The player assumes the role of a sentient artificial intelligence created by one of the main characters of the story. The purpose of the player is to help the creator of the AI to select operators for missions and direct those operators while the mission is in progress.



The story is further discussed in the text generation chapter.

The gameplay follows a loop where the player undertakes missions and when successfully completed the game and the story progresses. The gameplay loop design is demonstrated in Figure 9. The missions consist of tasks and an example of one mission design is showcased in Figure 10. The design also includes managing money, as the operators are not free to hire, and a mechanic called “Notoriety” where completing missions in a loud way that attracts authorities is penalized and can ultimately lead to game over. As stated earlier, the artifact itself is a prototype and some features are not implemented, but those elements are still part of the design, and thus have an effect on the requirements and desirable attributes of the generated content.

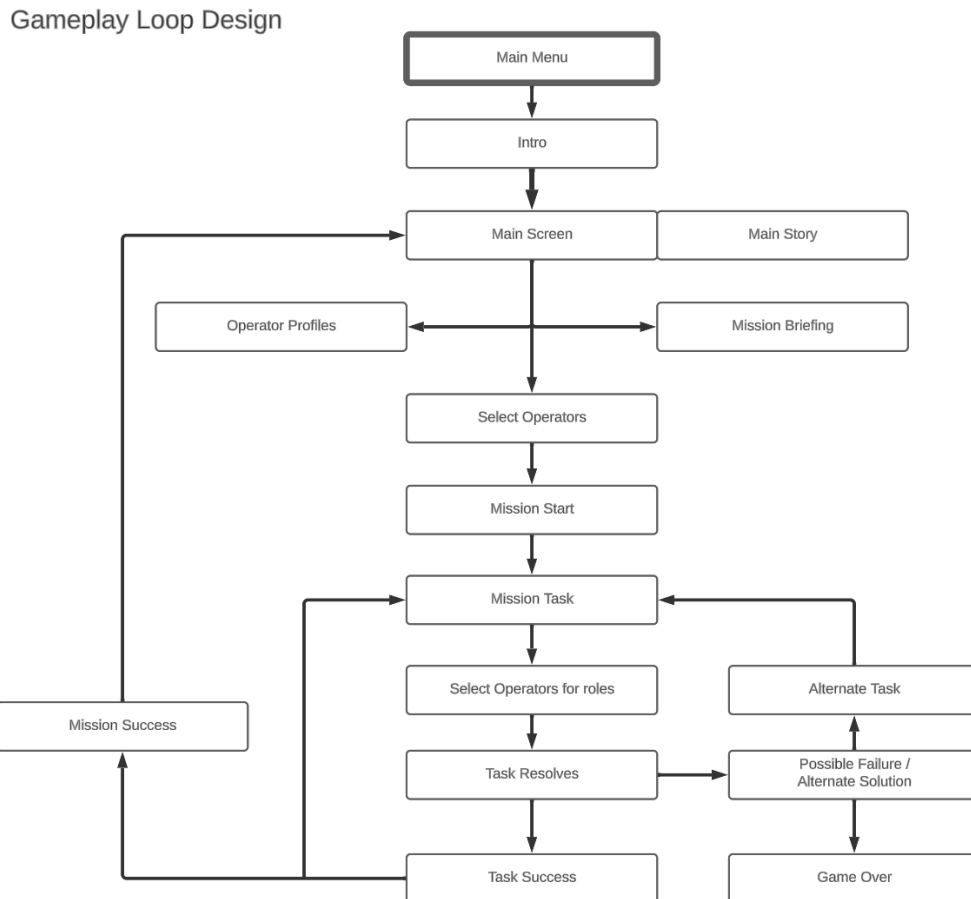


Figure 9 – Gameplay loop design

### Mission Progression Design

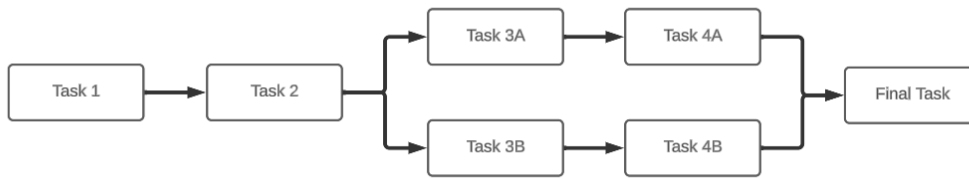


Figure 10 – Mission progression design

### 5.3 User Interface Design

The user interface design of the game was driven by a key principle: since players would spend a considerable amount of time engaging with the story and operator profiles, the readability, usability, and accessibility of these text-heavy elements were the focus. Although the game is designed for a 16:9 aspect ratio, I found that designing the user interface to be adaptable to other platforms and aspect ratios carried no disadvantages for this research, despite offering no extra benefits either.

Guided by these concepts, I set the design goals for the UI:

1. Must be instantly recognizable for most people who use a computer.
2. The large text body must be the focus of the UI.
3. The design should lead the player from left to right, which aligns with the intended gameplay loop.
4. Design is adaptable for other platforms and aspect ratios.

Considering these design goals, I created the initial, early UI-prototype in Adobe XD prototyping software showcased in Figure 11.

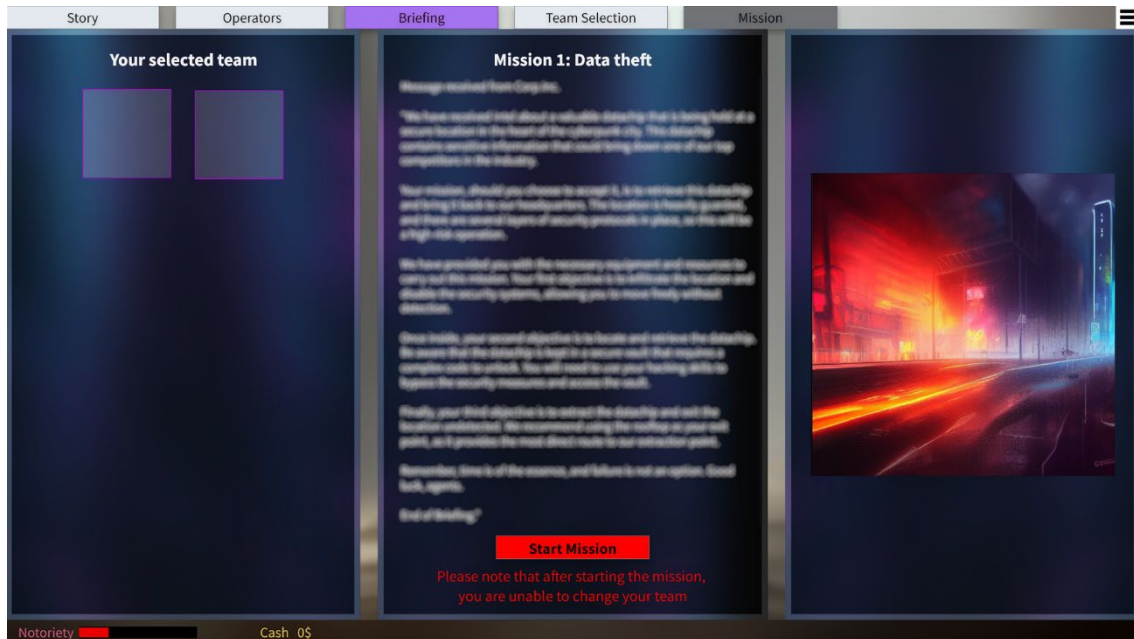


Figure 11 – Early prototype of the artifact.

To further clarify the design, the UI is divided into three distinct vertical areas in the center and two smaller horizontal areas in the top and bottom of the UI. Regardless of which tab is active, the main text is always displayed in the center, graphical elements to the right, and selections to the left. Two exceptions to this layout exist: Firstly, when a selection is critical, and secondly, to avoid any empty spaces, graphics or other elements are used. This design simplifies the possible transition to vertical aspect ratios like those on mobile screens, with the three central areas becoming individual screens. This also ensures that the text block remains the most prominent element of the UI.

The design goals were answered in following way:

- 1. Must be instantly recognizable for most people who use a computer:**  
This was achieved by using a layout similar to that of internet browsers.
- 2. The large text body must be the focus of the UI:**  
Dedicating the middle of the screen to the text ensured that the text will not be too wide in the screen to be uncomfortable to read.
- 3. The design should lead the player from left to right, which aligns with the intended gameplay loop:**  
The order of the tabs follows the order of actions taken in the game.

#### **4. Design is adaptable for other platforms and aspect ratios:**

The three vertical areas answered this design goal as it made the design highly modular.

The user interface design affects the generated image content in two notable ways. Firstly, it is not advisable to use aspect ratios that are wider than tall, as the area where images will be inserted is taller than wide. To elaborate, the UI design dictates that generated images should be square or portrait oriented. Secondly, the portraits of the operators must be in a square aspect ratio. The implications of these factors to content generation are discussed further in the Image Generation chapter.

#### **5.4 Text generation**

To clarify the restrictions and desirable aspects of the text that need to be generated, it is first necessary to provide an overview of the story and highlight one particular game mechanic. There are seven different types of text that need to be generated. Each with their specific requirements:

1. Main story
2. News stories
3. Operator profile
4. Mission briefing
5. Task briefing
6. Task outcome
7. Mission outcome

Working with ChatGPT, or any LLM, especially in creative writing, involves an iterative process that varies with each case. Acknowledging this variability is extremely important, as it highlights the tailored approach necessary when working with the LLMs. There is no one definitive method or prompt that guarantees success in every scenario. Often, the initial prompt itself may need to be adjusted to meet specific requirements. The prompt engineer is responsible to steer the LLM toward the desired content. This process often requires critical thinking and creativity to nudge the LLM towards the desired outcome.

##### Main story

The main story revolves around a character called Vincent, his brother Michael and the player who takes a role of a sentient artificial intelligence created by Vincent. The

original story and overarching plot were created by the author of this research, but the details and writing were done using Chat-GPT 4, as appropriate for this research.

To provide details of the story I felt it appropriate to include a section from game design document, where the games story is presented as follows:

---

Vincent was born into a well-to-do family in the heart of the city. However, his childhood was far from idyllic. Vincent's father was an abusive alcoholic, and his mother was distant and uninvolved. His older brother, Michael, was the only one who showed him any kind of love and attention.

Despite his difficult upbringing, Vincent was a bright and curious child. He loved to tinker with electronics and computers, often taking apart and rebuilding gadgets in his spare time. Michael was always impressed by Vincent's natural talent for technology and encouraged him to pursue his interests.

As the years went on, Michael grew increasingly focused on his studies, and later, his political career. Vincent, feeling neglected and abandoned, turned to drugs and alcohol to cope with his pain. He became angry and resentful towards his brother, believing that Michael had turned his back on him.

Vincent's addiction only worsened over time, and he eventually dropped out of school. However, he continued to spend his days tinkering with technology, even creating his own artificial intelligence program in secret. It was through this program that he discovered his true talent - the ability to create and manipulate artificial intelligences.

Vincent felt betrayed by Michael and with his newfound power, he became obsessed with the idea of taking down his brother. He began to work on a plan, using his AI program to gather information and resources. His goal was to create the perfect team of operators, who could help him complete a series of missions that would lead up to his ultimate goal - killing his brother, the mayor of the city.

As the AI, the player will assist Vincent in his mission, selecting the best possible team of operators to complete each mission and ultimately take down the mayor. The AI is not aware that this is the mission, and the player is led to believe

that Vincent is a “good guy”, however, some well-hidden hints of the “evil plot” are given to the player throughout the game.

---

This background of the main characters and the player's role should be sufficient to illustrate the game's story. A more detailed world setting is described in the game design document, but I have chosen to exclude it as the exact details do not contribute meaningfully to this research.

#### Process of Generating text

For the generation of the main story, guidance on the plot's direction was provided to ChatGPT, which then led to the development of an increasingly detailed narrative. This process can be illustrated as follows: Starting with brief instructions, the setting and characters were outlined, the writing style and key events of the story were established. The conclusions of the happenings were left to ChatGPT to decide, as to examine its capabilities in creative writing. So, even though the setting were introduced to ChatGPT, the conclusion of happenings and how the story unfolded were not instructed.

*“Embody the dark, gritty, and fast-paced nature of cyberpunk, utilizing vivid descriptions and fragmented sentences to create a sense of urgency and danger. Incorporate the use of vivid imagery and sensory details to immerse the reader in the dystopian world, while also exploring themes of rebellion, technology, and the human condition. Keep the focus on character development and action-driven plots, utilizing the first-person point of view to create a sense of intimacy and immediacy.”*

- Example of guidance of writing style and tone for ChatGPT to generate the story in appropriate and fitting tone.

It was often necessary to introduce more elements and rules for ChatGPT, as it could deviate with instructions that weren't specific enough. If there were previous parts of the story, ChatGPT was first introduced to these, but only to read them, not to take any action. The instructions came after this. Sometimes, after the first draft, starting a new chat was beneficial, especially if the content wasn't satisfactory, such as not being of high enough literary quality, or if the story was too generic or uninteresting. Starting a new chat resets ChatGPT to its initial state. At this stage, I often iterated on prompts and instructions. If these changes led to better results, I kept them for future iterations and

attempts. If not, sometimes starting from scratch and crafting the initial prompt was necessary or changing the approach altogether.

*“Please iterate and add details. Develop the characters a bit better. Expand and create more suspense in the first interaction. Let the reader know, in very subtle ways, that Vincent is not exactly sure if the AI is really okay and well. Let him be a little wary, as sometimes AIs can be dangerous. However, do not use these guidelines as restrictions to your creativity. Use them as just nudges in a certain direction. Do not go overboard. All of these should be very subtle additions and should not be glaringly obvious. Tiny, intricate changes.”*

-Example of iteration process when working on the first story.

The goal was to generate a story that is not strictly traditional cyberpunk story, but still ties to the genre strongly. Since the story is introduced to the player one section at a time, there always needed to be something interesting in the narrative. This made the process different than writing a longer piece.

In crafting the narrative, I attempted to give attention to the structure and dynamics of storytelling. This meant not just stringing together a series of events, but considering the foundational elements present in storytelling. For example, the way characters develop, the pacing of the plot, thematic elements and using sensory details to immerse the reader to the text more effectively.

Generating text for Main Story

Reflecting back to the audio diaries and experiences, one of the findings was the tendency of ChatGPT-4 to integrate clichés in the storytelling. The clichés are obvious and repeating. In the Cyberpunk setting this means that back alleys, mega-corporations and especially neon lights are prevalent in the text.

*EOI: ChatGPT will incorporate cliches and repeat them.*

For game development and storytelling in general, direct mentions of cliches like neon lights might not be desirable and a story revolving around the main cliches of the genre is rarely desirable.

Nudging, a term commonly used by GAI users to refer to the act of guiding the AIs focus towards or away from certain topics, is not as simple as just telling ChatGPT to avoid clichés. However, from a prompt engineering perspective, it's not an overly complex task either. Avoiding clichés can be done by instructing the avoidance and, the more essential part, to be more specific problematic parts, like settings, characters or other aspects and pointing out what specific clichés the ChatGPT should not repeat.

This is directly related to the fact discussed in the theory section of this research that the quality of input has a direct effect on the quality of the output. It is possible to zero-shot a prompt that avoids clichés.

*EO2: ChatGPT can be nudged away from clichés with iteration. Simply prohibiting clichés is not enough. The best results are achieved by guiding ChatGPT towards desired topics (positive prompting) while also discouraging cliché usage (negative prompting).*

It is crucial for the story to be consistent and not contradict itself. Due to ChatGPT's token limitations, it cannot retain the entirety of a long story. Therefore, there is a need to reintroduce the previously generated story, especially on relevant parts, to ChatGPT, essentially inputting the content it has generated earlier. One approach to manage this is by condensing the story to make it more concise, which helps to accommodate ChatGPT's limitations.

However, my observations suggest that this approach might actually increase contradictions. This issue arises because condensing the story often means reducing the amount of text that describes each event. In a more concise format, significant events and minor details may be given equal weight in terms of text length. This equalization can lead to inconsistencies, as the AI might not recognize the relative importance of events within the condensed narrative. This leads us to final observations of creating the main story.

*EO3: ChatGPT might contradict itself with the story, especially when operating closer to the token limit.*

*EO4: Even without token limitations, consistently prioritizing key aspects of the narrative would likely remain challenging, as ChatGPT sometimes struggles to distinguish between important and less significant story events.*

*EO5: Condensing the narrative is not an effective solution for the forgetting nature of ChatGPT, as it often equalizes the content on amount of text and not the content itself.*

#### News stories

Generating news stories for the prototype proved somewhat simpler than other content, as inspiration could be drawn from various sources. Real-world newspapers or inquiries to ChatGPT about different news types and subjects provided easy to access inspiration.



In the context of game design, however, it's crucial that these stories align with the game's world. They should enhance the player's immersion and understanding of the environment crafted by the designers. This narrative technique is commonly employed across various game genres to enrich the gaming experience.

In my specific prototype, the news stories had a specific role. They were crafted to incorporate narratives about in-game characters, particularly the mayor, without disclosing his role in the game's plot. This approach was designed to mislead players into believing these stories were mere world-building elements, rather than insights into the underlying narrative involving the brothers and the player's role. It's important to note that the game's storyline is structured in a way that the player remains unaware of Michael's (the mayor's) true identity and significance.

*EO6: Generating short stories for the game world proved easier than anticipated, despite the complex demands of world-building and specific requirements of the prototype.*

*EO7: Providing ChatGPT with specific ideas or concepts improves its ability to generate unique and engaging stories. Merely asking it to generate a cyberpunk news story is not sufficient to achieve the best quality.*

*EO8: Careful review of ChatGPT's outputs is essential to avoid contradictions in the story and game world.*

*EO9: Maintaining a consistent style and tone throughout a story is easier in shorter pieces than in longer narratives. This characteristic showcases that ChatGPT can be an efficient and high-quality tool for this type of work.*

Generating text for Operator profiles

One requirement for the operator profile texts, also known as operator story, was to tie them with the strengths and weaknesses of the operator. The strengths and weaknesses determine the success or failure of the tasks, and ultimately also the mission, is showcased in Figure 12.

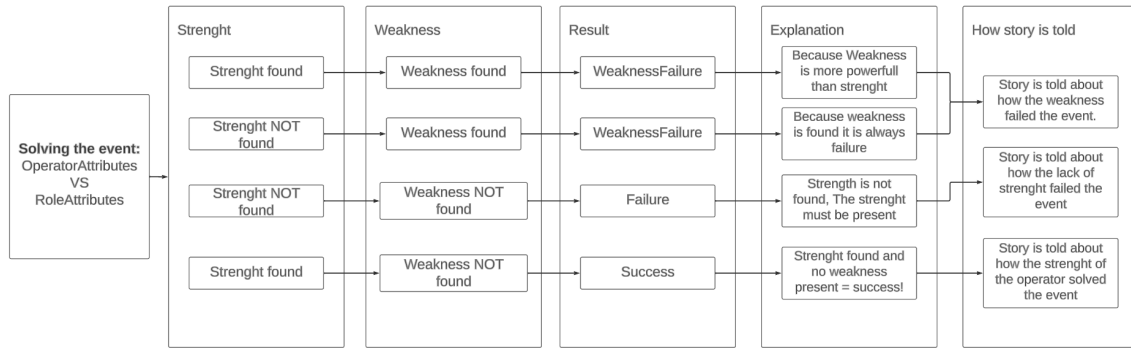


Figure 12 – Operator strengths and weaknesses. Logic how to solve the task.

Ultimately, this means that the story of the operator needs to clearly reflect their strengths and weaknesses, which are collectively referred to as attributes. It's essential for players to be able to deduce these attributes from the story, as deciphering them is one of the central challenges of the game. The narrative should be complex but not directly reveal the operator's attributes, as this would make the game too easy. However, if the story doesn't hint at these attributes clearly enough, it would lead to player frustration. This aspect of the game would most likely be the most challenging to balance. The goal is to craft a story where, after discovering the attributes, players would realize the clues were there all along, making the attributes' integration into the story appear obvious in retrospect. It is self-evident that the stories must also fit the crafted world. Such are the specific requirements for the operator stories.

*EO10: Similar to EO7, introducing concepts will greatly increase the creativity of ChatGPT.*

As more profiles were developed, it became increasingly challenging to meet these requirements. The task of integrating different attributes was sometimes complex for ChatGPT, necessitating creative input to subtly incorporate all abilities. It became evident that achieving subtlety is not a natural strength of ChatGPT.

This revealed ChatGPT's inclination to approach narrative elements with more clarity than subtlety. For example, when presenting an attribute like “Fear of Heights”, ChatGPT tended to do so in a straightforward manner. Guiding it toward a more nuanced representation often resulted in clues that were more pronounced than intended. Striking the right balance of subtlety was a frequent challenge. It often required revising sections, such as paragraphs, multiple times to finally achieve a narrative that subtly yet effectively depicted the character's traits.

*EO11: ChatGPT struggles to introduce concepts subtly. Resulting in the need of multiple iterations or manual story writing.*

In the process of creating operator profiles, I observed a distinct pattern in ChatGPT's output: it frequently depicted characters as exceptionally skilled, often portraying them as the best in their field. Achieving a more balanced portrayal, where an operator was average or less proficient in their role, proved to be a time-consuming task. Considering the varied hiring costs of operators in the game, it was important to have a range of skill levels among them. However, even when a story began promisingly, ChatGPT tended to eventually introduce a characteristic or action that portrayed the operator as highly skilled and proficient. This inclination to insert a 'hero' into every narrative was more of a hindrance than a benefit. It undermined the diversity of the characters, as having every operator depicted as exceptionally skilled dilutes the concept of a hero within the story.

*EO12: ChatGPT predominantly introduces the protagonist as highly skilled, posing a challenge in the creation of average or less proficient profiles.*

#### Mission briefing

In instances of successful content generation, ChatGPT demonstrates significant creativity and entertainment value. However, the level of effort needed to attain such outcomes, along with the possibility of inconsistent results, presents notable challenges. Through experience, it becomes evident that the abilities and, more specifically, the creative strategies employed by the prompt engineer play a crucial role in overcoming these obstacles.

*EO13: ChatGPT can be highly creative and entertaining. It can be surprisingly intricate in details and narrative events it generates.*

An example of this creativity:

---

“As Vincent spoke, the AI detected a subtle shift in his cadence, a fleeting irregularity in the rhythm of his words. This momentary lapse piqued its curiosity, but after a brief analysis, the AI deemed it not worth the processing power and moved on. Vincent's gaze lingered on the holographic layout, his thoughts clearly churning. "The operator," he began, choosing his words with care, "must move like a shadow, leaving no trace of their presence. They need to be... invisible in their craft."

The AI, sensing the underlying doubt in Vincent's voice, responded with a hint of defensiveness, "I am fully capable of selecting the right individual for the task, Vincent."

Attempting to lighten the mood, the AI added, "Perhaps someone who took hide-and-seek very seriously as a child?"

Vincent's eyes met the AI's interface, the chill in them unmistakable. "You're not funny," he said, his voice devoid of warmth."

---

*EO14: The proficiency and creative tactics of the prompt engineer are instrumental in navigating ChatGPT's challenges, particularly in achieving consistent, high-quality creative outputs.*

#### Task briefing

Creating task briefings from a design standpoint proved challenging, as the narratives needed to be sufficiently general to avoid contradictions, depending on the operators chosen by the player. Additionally, dividing the team of operators was not an option, as it would disrupt the game's design and flow.

These challenges were not directly linked to ChatGPT. However, operating within tight guidelines while still crafting a concise story that effectively illustrates the team's tasks or their progress was a notably restrictive process.

Given the specialized nature of this specific case, I am hesitant to draw major conclusions or place full confidence in some observations from this stage. The situation may vary significantly if the constraints differ from those in the task briefing texts. However, it's noteworthy that operating under strict restrictions was not detrimental to ChatGPT. The tasks were still completed, often resulting in highly entertaining text.

*EO15: While ChatGPT retains its creative flair in creative writing tasks with strict restrictions, the role of the prompt engineer escalates in ensuring the narrative stays within set boundaries.*

Excerpt from a Task Briefing Text:

---

Under the cover of night, the team moves stealthily past the elementary school. The digital murals on its walls are dimmed, but occasionally flicker with residual images from the day's lessons. The school, even in the quiet of the night, stands as a symbol of hope in East Side Chicago, amidst the challenges of 2103.

Approaching the target building, they spot the entrance: a simple door, but locked, with a state-of-the-art surveillance camera monitoring every movement. The team exchanges glances, realizing that attempting to tamper with the door would be too risky.

---

Task outcome and Mission outcome

The task outcome text is a few sentences depending on the result (see figure 12). As the texts were short and concise explanations failure or success, ChatGPT could easily be used to create both of the texts. The benefit was that when creating opposing results, the set tone stayed the same in both instances.

*EO16: Creating texts for different or opposing outcomes or situations is effective with ChatGPT, as it maintains the style and tone and has no apparent troubles changing the narrative from one extreme to another.*

Example of task resolution text when the player has selected the appropriate operator for a task.

---

“High on an adjacent building, the scout watched, augmented vision sweeping the neon cityscape. His duty was vigilance, an unsung guardian in the digital night. Spotting a drone's faint glow, he promptly sent a discreet alert to the infiltrator.”

---

The mission outcome texts were very similar to briefing and main story texts. No additional observations were made when generating these texts.

## **5.5 Image generation**

In game development, a wide variety of 2D images are utilized, ranging from user interface elements to textures. While these components are important, they are not the primary focus of this research. Instead, the emphasis is placed on images that enhance the narrative, such as those used in character profiles and background settings, which play a significant role in storytelling and overall game aesthetics.

The research initially began with Stable Diffusion before transitioning to Midjourney.

The first attempts utilized Stable Diffusion 1.5. While it was possible to continue with Stable Diffusion, Midjourney quality was found to be superior during the timeline of this study. A more significant reason for switching was the high hardware demands for creating top-quality images with Stable Diffusion. Optimal results with Stable Diffusion required modifications to the model, which in turn needed substantial VRAM and time. For an accessible overview of these modifications, including LoRas, dreambooth, textual information, and hypernetworks, see the video by koiboi (2023).

In Midjourney, unlike in Stable Diffusion, model modifications are not an option, leading to a more straightforward process centered around prompt creation. With Stable Diffusion 1.5, I explored the possibility of using prompts alone to define an art style. This was put to the test by trying to generate multiple art styles through refined prompting. Although the prompting methods in Stable Diffusion differ from Midjourney, this exercise was valuable for understanding the state of prompting image generation models as of early 2023. In my early experimentation, I generated over 15,000 images to identify the most effective ways to create an art style and to acquaint myself with this new technology. A significant method in producing images with similar features consistently was using what I would call “end string” in each prompt. To craft the end string, I referred to multiple documents showcasing different art styles and prompting words used and selected the most desirable keywords. The end string was formulated as follows:

**Matte Painting, (Classicism Painting), Spray Paint, Genre Painting, Speed Painting, Gothic painting, Cyberpunk, Scifi.**

The parentheses in the end string increase the strength of the word (while brackets decrease). So, in order to use the end string, I would introduce the subject and setting before the end string like showcased in Figure 13.



Figure 13

**Prompt:**

Male standing on carwreck with a flaming shotgun, detailed cityscape as background, neonlights, hyper detailed, Matte Painting, (Classicism Painting), Spray Paint, Genre Painting, Speed Painting, Gothic painting, Cyberpunk, Scifi.

**Info:**

Steps: 60, Sampler: Euler a, CFG scale: 7.5, Seed: 1803916183, Face restoration: CodeFormer, Size: 1024x1024, Model hash: d8691b4d16, Model: deliberate\_v11, Denoising strength: 0.5, Mask blur: 4

These experiences with Stable Diffusion 1.5 proved useful with Midjourney and the purpose of showcasing these experiences is to provide background and showcase even one generated picture from the early part of this research.

## Working with Midjourney

Midjourney v5.2, released on June 22, 2023, is the central focus of this research. This version, like its predecessors, has notable natural language processing (NLP) traits. A prominent feature is its limited, but not totally devoid, ability to interpret words that should be connected. Essentially, if a concept is not a compound word, the model tends to treat the input as individual, standalone words. The exception is when the concept is very prominent. There are other exceptions to this, but it's unclear why certain terms are exceptions. This ambiguity is a challenge in working with a model that is not open-source or thoroughly documented.

For instance, using the prompt “still life” successfully generates still life pictures, demonstrating the model's ability to recognize this specific compound concept. However, the issue becomes more evident in longer prompts with multiple words, since users cannot use punctuation like commas to differentiate concepts, which can sometimes lead to complications.

Additionally, other traits such as the ineffectiveness of commas and capital letters in altering the prompt's meaning present further challenges. Commas do change the outcome, but not in the way one would expect as they add “noise” to the input and offer no practical utility. Furthermore, capitalization does not impact the interpretation. These characteristics underscore the complexities of prompting in Midjourney due to its specific NLP traits.

I raise these factors for the reader to better understand the challenges of prompting Midjourney. However, as this is not a user guide, I will not delve into specifics more than this. Midjourney documentation is sufficient user guide and can be found on [Midjourney.com](https://www.midjourney.com) (*Midjourney*, n.d.).

*EO17: Prompting image models efficiently will require the user to be familiar with the specifics of the model's natural language processing system.*

*EO18: Surprising results can be generated even when user is familiar with the image model's natural language processing, as the user cannot always consider all factors that are present in the specific prompt.*

A notable characteristic shared by both Midjourney and Stable Diffusion 1.5 is the impact of aspect ratio on image generation. The default aspect ratio is 1:1. This was evident when I tried to generate an image of a weapon on a table. The most prominent



outcome was that the weapon extended beyond the image's borders. Using a widescreen aspect ratio, such as 16:9 or 21:9, proved to be an effective workaround. Similarly, for full-body human portraits, opting for a tall and narrow aspect ratio, like 9:16, enabled the model to produce images where the entire human figure was visible. This holds true for every concept, be it a ship or spacecraft.

*EO19: Aspect ratio significantly influences image generation, serving as a workaround to ensure full visibility of figures or objects. In square aspect ratios, parts of the image are often cut off; adjusting to wider or taller ratios can prevent this.*

One of the most challenging aspects of generating images for the prototype was ensuring consistency across different images. Even with specific style descriptions, noticeable variations often occurred between the generated images. I refrained from conducting a mathematical analysis of the proportion of valid images, as such an undertaking would require a substantial sample size and precisely defined criteria. However, it became apparent during the research that only a few images truly adhered to the intended style. Additionally, the images needed to be free of common AI-generated errors, such as extra fingers, which remain a persistent issue in this domain.

*EO20: Achieving consistency in style-specific AI-generated images is challenging, with only a limited portion aligning closely with the intended design.*

For game development this means that a process must be developed where the images maintain consistency on higher percentage. Otherwise generating images and selecting the ones that fill the criteria would be more time-consuming.

For the prototype, my endeavor was to seamlessly blend elements of the cyberpunk theme with the aesthetics of oil painting, particularly in the realm of realistic classicism. The aim was to create a visual composition characterized by muted colors, a soft palette, and gentle lighting, all while avoiding the chiaroscuro and high-intensity elements not typically found in classicism.

These attempts yielded promising results in the early part of the research with Stable Diffusion. Even more promising initial results were generated with Midjourney, as illustrated in Figure 14. However, a significant observation emerged after numerous iterations. It became apparent that there was no clear intersection point, where the classicism style and the cyberpunk elements coexisted in a manner that prevented the classicism aspects, or content, from overwhelming the image. To clarify, the content within the image, including objects like glassware, typically exhibited classical

characteristics. Moreover, the architectural elements often leaned towards classicism rather than the intended modern or cyberpunk aesthetics. This observation indicated a notable discrepancy between the envisioned fusion of styles and the actual outcome.



Figure 14 – Midjourney v.5.2

**Prompt:** very soft lighting, smooth colors, dark gray and light amber, realistic hyper-detailed, muted, detailed illustration, in the style of classicism painter, sophisticated::10 old gentleman cyborg sitting in hotel bar sofa, cyberpunk, augmented hand, black elegant suit, glass walls, reflections, hotel setting, indoor, luxurious building::10 scifi, cyberpunk::5

**Seed:** 787450568

The final and most important observation is that style and theme exert a significant influence on content. This is a logical conclusion evidenced by how diffusion models are developed. The training data is the source of how images are generated, and thus, it contains the content, theme, and style. However, the current models are unable to differentiate between these three, leading to this dilemma.

While forcing a theme into the generative model, such as cyberpunk, has a great effect on the image, coupling it with a style not prevalent in cyberpunk will introduce all those styles, themes, and content as well. To clarify even further, it is not possible to consistently create truly classical oil-painting images that have the theme and content of cyberpunk. The more the prompt emphasizes classicism, the fewer cyberpunk elements are present, and the more the prompt emphasizes cyberpunk, the less classical oil painting style the image has. The middle ground of these two results in a blend of classical and cyberpunk content and classical and cyberpunk style. It is impossible to consistently create images that have cyberpunk content but are in the style of classical oil painting. In other words: Style drives content.

*EO21: Style and theme drive the content of the image generated and they cannot be differentiated from each other.*

In conclusion, the complex interplay between style, theme, and content in generating images with GAI poses a considerable challenge for game developers striving to maintain visual consistency and convey a distinct artistic vision. However, as an artistic tool for discovering, demonstrating, communicating, and prototyping various aspects of 2D art within the game development process, GAI can prove invaluable. A professional and knowledgeable prompt engineer is capable of generating essential preliminary materials often required in game development, including, but not limited to, placeholders, textures, palettes, and more.

*EO22: Generative AIs role as an artistic tool in game development can prove invaluable for discovering, demonstrating, and prototyping various aspects of 2D art.*

Character and environment consistency are crucial in image GAIs. Midjourney, while not providing direct tools for this purpose, can be used effectively with external tools and methods to enhance consistency. Prompting significantly aids in environment consistency, but achieving the same level of consistency in characters through prompting alone is extremely challenging. The Midjourney community, however, has developed workarounds and techniques to overcome these challenges. Based on these

developments, it's clear that character consistency for game development is achievable with current technology and will likely become easier in the future. A detailed examination of all these methods, workarounds, and external programs is beyond this research's scope simply because of resource constraints. Nevertheless, it's important to note that despite challenges with character and environment consistency in image GAI, these issues aren't significant enough to prevent the adoption of image GAI in the game development process, especially as there have been effective workarounds.

*EO23: Despite the challenges in achieving character and environment consistency with image GAI, the challenges are not substantial enough to prevent the usage of GAI in game development. Workarounds and methods are available and are being developed.*

To conclude this chapter, I present three distinct images (figures 15, 16 and 17) that use the same seed and are attempts to incorporate the classicism style into cyberpunk content. These images should serve to further clarify EO21 and the noticeable stylistic changes between the images.



Figure 15 – Midjourney V 5.2

**Prompt:** classicism painting of a slender serious female. soft light, classical vibes.

**Seed:** 2962851412



Figure 16 - Midjourney V 5.2

**Prompt:** classicism painting of a slender serious female. soft light, classical vibes.  
cyberpunk scifi futuristic

**Seed:** 2962851412



Figure 17 - Midjourney V 5.2

**Prompt:** classicism painting of a slender serious female. soft light, classical vibes::  
cyberpunk scifi futuristic::

**Seed:** 2962851412

## 5.6 Music generation

In the process of creating theme music for the prototype with AIVA, I quickly realized that generating music was not a straightforward undertaking, presenting a steep learning curve associated with AIVA itself. The user-adjustable parameters in AIVA are closely tied to key concepts in music theory. Although there is a quicker and seemingly simpler method for music generation, the resulting compositions often turn out to be erratic and lack the structured framework typically found in electronic and rock music, genres that are prevalent in games. As someone without a classical music background, evaluating the aesthetic quality of the classical pieces produced by AIVA proved to be particularly challenging.

The user-adjustable parameters in AIVA present both advantages and challenges. While an individual with no musical background can certainly produce some form of output, I contend that having prior musical experience, even at a hobbyist level, can significantly improve the quality of the generated music. As with Midjourney, I feel that providing an overview of how to use AIVA and going into details of its use are not the focus of this research, instead the focus should remain on research questions. A detailed user guide can be found on AIVAs website (*AIVA - AI Music Generator*, n.d.).

In the context of game music, a challenge I encountered during my experiments with AIVA was the difficulty in generating engaging riffs and melodies. This challenge could be attributed to what I perceive as AIVAs limited handling of the “call and response” technique, particularly in the way it's used in popular music. While “call and response” is a common element in classical music, which is AIVAs primary training focus, the pattern in classical compositions is rarely repeated exactly as is, unlike in popular music. This difference in musical structure between classical and popular music genres appears to affect AIVAs ability to replicate the repetitive and exact “call and response” nature commonly found in popular music.

*EO24: Generating engaging melodies and riffs without manual intervention with AIVA remains a challenge.*

Nevertheless, the use of AIVA in music generation has many merits, especially due to its ability to output MIDI. This is crucial as most popular DAWs handle MIDI files effectively. The greatest aspect of AIVAs MIDI files is that they can contain, if desired, dynamics and tempo fluctuations. These seemingly minor features are immensely beneficial for a composer working with AIVA-generated content, as they eliminate the need for the time-consuming task of manually adding these elements. Essentially, AIVAs output can serve as a foundational idea and inspiration, which can then be refined and improved upon.



*EO25: AIVA generated music can serve as a foundational idea and inspiration. This aspect is further enhanced as editing is made possible by providing a high-quality MIDI file.*

For an amateur musician or indie game-developer AIVA has notable benefits in its ability to create music for different instruments. This is important to note because it might be the case that an indie developer has limited skills, knowledge, or time to compose music for the game. And as is in my case, even though I perceive to have adequate, amateur level composing skills, I often struggle with composing and creating drum tracks. AIVA can help with this greatly, even though there is no direct music to music generation possibility. It is not possible at this time to present a melody for AIVA and let it generate the drums for it.

Hence, AIVA can be particularly helpful in addressing personal weaknesses in music production. It effectively complements Digital Audio Workstations (DAWs) and serves as a useful tool for exploring chord progressions and underlying structures, even though it may not be the optimal solution for generating melodies.

The chord progressions generated by AIVA consistently aligned with established musical theories, demonstrating a strong understanding of harmonious chord sequences. None of the compositions I encountered exhibited blatant discordance or dissonance, indicating AIVA's proficiency in creating musically coherent pieces. Furthermore, AIVA's capabilities extend far beyond simplistic three-chord structures. It can generate complex and elaborate chord progressions, showcasing its versatility in handling intricate musical compositions.

*EO26: AIVA's ability to output MIDI files enhances its compatibility with DAWs, facilitating music editing and composition processes.*

*EO27: AIVA proves beneficial in supplementing areas where a composer may lack expertise, particularly in instrumental and structural aspects of music production.*

*EO28: The chord progression generated by AIVA are sensical and often interesting.*

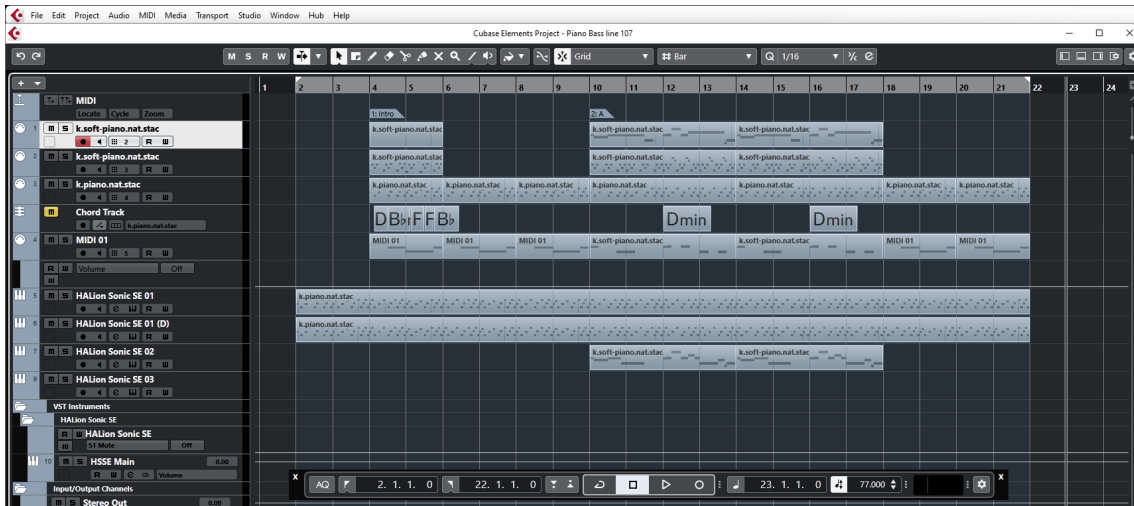


Figure 18 – Screenshot of the workspace in Cubase. A finished and ready main theme for the prototype.

The original composition generated by AIVA, along with the modified, completed version, is included in the Git files. The relevant address for these files is provided in the Appendices section.

## 5.7 Empirical Conclusions

In this section, the focus is on presenting the primary empirical conclusions derived from the observations. These conclusions are more comprehensive than the empirical observations detailed earlier and play a crucial role in addressing the research questions of this thesis. However, it's essential to understand that these results are not the complete, holistic conclusions of the study. Instead, they are key insights that form a foundation for the broader conclusions and discussions that follow. This section aims to provide a clear and concise synthesis of the findings, setting the stage for the in-depth analysis and final conclusions of the research.

PEC1: For creative writing in game development, ChatGPT's potential for brilliance and flair is unlocked by nudging it to the right direction and by providing positive/negative guidance along with a concept and theme. However, maintaining consistency of the story and avoiding cliches falls to the shoulders of the prompt engineer.

PEC2: For creative writing in game development, ChatGPT's challenges lie in its limited memory and discerning crucial and less crucial narrative elements. Currently shorter narratives are more viable. However, coherent longer narratives may become attainable if structured system is developed in the future.

PEC3: For creative writing in game development, ChatGPT's limitations emerge in introducing and maintaining subtlety. Furthermore, its built-in content limitations can interrupt the exploration of more mature or complex themes often present in video game storytelling. These factors combined can be disruptive to the creative process.

PEC4: Style drives content in generative AI image creation, with style and theme being inseparable. This bond shapes the visual narrative and impacts the content produced. The ability to create consistent unique styles is inherently limited by the combination of theme and style existing together, and their prevalence and representation in the AI's training data.

PEC5: Consistency in characters and environment within Midjourney and similar image GAI's remains challenging. Yet, with ongoing advancements it is likely that solutions will present themselves in the very near future. These solutions are needed before Midjourney and other GAI's can be considered to be included in the game developer's toolset.

PEC6: Effective use of Midjourney and similar image GAIs demands a deep understanding of their natural language processing and creative use of the various techniques available. The quality of the output is fundamentally tied to the input, and thus to user proficiency. However, even the most adept users cannot predict the model's output.

PEC7: Effective use of AIVA benefits from prior musical experience and music theory knowledge. While AIVA excels in creating exciting chord progressions, it struggles with melodies and riffs. Its adherence with musical theory makes it useful for basic compositions and guidance in specific areas. However, directly generating completely polished, genre-specific songs is currently beyond its capabilities.

PEC8: AIVAs technical prowess lies in generating high-quality MIDI files that can include dynamic and tempo variations. This makes it an asset for game developers and composers, enabling them to quickly establish the foundational feel of their music, thereby accelerating the music creation process in Digital Audio Workstations.

## **6 Discussion and Conclusions**

In this chapter, I aim to approach the findings of this research holistically and draw final conclusions. Additionally, I will reflect on the current state and future directions of GAI, and attempt to identify expected challenges, developments, and benefits in the near future. I will also suggest where and how GAI could be most effectively deployed and what may be required before such deployment is feasible. Furthermore, I will propose some avenues for future research.

### **6.1 Current state of the generative artificial intelligence**

At this moment it is not viable to use GAI to directly generate content for video games. However, using GAI and then polishing or editing the results is a viable method. This is evident from this research as both Midjourney and AIVA greatly benefitted from moving the created content to external software, in Midjourney case to Photoshop or another image editing program and AIVA benefitted moving to DAW, such as Cubase.

While achieving the desired outcome with Midjourney is possible, ensuring consistency and repeatability across various projects remains a challenge. This challenge is not unique to Midjourney but extends to other image GAIs like Stable Diffusion. Currently, there are no definitive and proven methods to consistently create images and styles that are repeatable across diverse subjects, settings, environments, and content in the context of game development. While Midjourney is taking steps to address these challenges through community events such as 'Character continuity with Victor Gnarly,' these methods are still evolving and not yet refined enough to meet the game industry's need for a stable and predictable development process. This highlights the ongoing demand for improved consistency in image generation, which is likely to become a key focus for future image GAI development. Therefore, it is advisable for the game industry, including both indie developers and major studios, to stay informed about these developments and cultivate at least a basic understanding of how GAI functions and consider the potential and likely near-future advancements.

While image and music GAIs are still in the process of finding their roles and evolving, I assert that the creative quality of ChatGPT 4 is of such a high caliber that experienced prompt engineers and writers will find it exceptionally valuable. While ChatGPT cannot currently reliably keep track of lengthy stories and their happenings, it is a unique tool for editing, ideation, and creating shorter stories. Even if ChatGPT is not directly involved in the development process itself, it serves as an indispensable tool for creative writers, making it difficult not to recommend for everyone engaged in game

development creative writing. However, it is important to acknowledge the inherent risks associated with misusing and not fully comprehending how ChatGPT operates. It may seem somewhat pessimistic, but there is a genuine concern that it could become too convenient to rely solely on ChatGPT and neglect the critical role of human review. Widely documented instances of individuals using ChatGPT without a comprehensive understanding of its characteristics have appeared in various news stories around the world.

In general, the current state of GAI is characterized by rapid advancements in all multimedia areas, with a notable emphasis on providing entertaining experiences for users. However, I am confident that as the field progresses, we will witness an increase in practical application of GAIs.

## **6.2 Challenges of GAI in game development**

A significant challenge in integrating GAI into game development is achieving consistency across characters, environments, style, and narrative. Game developers require control over these elements to ensure that generated content aligns with the game's design and story. For instance, it's essential for ChatGPT to remember and incorporate all story developments, for Midjourney to maintain a directed style despite content changes, and for AIVA to produce music that fits cohesively within the game's atmosphere. Addressing these challenges is crucial for the successful adoption of GAI in game development. For indie developers it is notable that workarounds for some of these issues exist, but the workarounds require manual labor and knowledge, many of these workarounds are ad hoc solutions rather than established processes or methods. Because the established processes are not yet fully developed, it is not advisable to incorporate GAI tools extensively for the larger game projects. Fortunately, addressing these challenges aligns with the current focus of GAI developers, and it has become a prominent topic in the field.

Addressing the “style drives content” dilemma, as explained in the Empirical Conclusion chapter, this challenge in GAI for game development is profound. The observation in this research, that style and theme are inseparable in GAI, means that the content generated is inherently influenced by their bond. I argue that this dilemma is present in all GAI models because the training data is introduced to the model in such a way that these two are not differentiated. It might be that differentiating them is not possible or is very difficult. The capacity to generate truly unique styles is constrained by the intertwined nature of theme and style, along with their representation in the AI's training data. This limitation impacts the scope of what developers can achieve,

positioning the “style drives content” dilemma as a critical obstacle that restricts GAI's potential in game development. This dilemma isn't always immediately evident; it becomes more apparent the further one departs from traditional combinations of style and content. In my assessment, this research's most notable contribution lies in this observation.

As established, the quality of the input directly affects the output in all GAI models. Thus, acquiring a knowledgeable and professional workforce can be challenging for the gaming industry, as it is not the only sector interested in hiring competent prompt engineers. Additionally, proficiency in one model does not guarantee proficiency in all, and there is a learning curve for each GAI model. For those models that use language as one of the input parameters, it is essential that the prompt engineer has extensive knowledge and vocabulary of the English language. This, along with creativity, directly ties into the ability to craft effective, high-quality prompts. Furthermore, even a basic understanding of how different models are constructed can enhance their ability to craft effective prompts. Identifying a proficient workforce can be problematic.

A potential challenge arises when different creative departments within game studios are impacted differently by GAI. If some creative departments suddenly become proficient in utilizing GAI while others lag behind, it can create disparities in efficiency across departments. This discrepancy can lead to problems, as the overall game development process may not accelerate if one department continues to require the same amount of time as before. Organizing work and managing the development process can become particularly challenging for project managers under such circumstances. This disparity was evident during this research when working across various multimedia platforms; achieving satisfactory results was notably faster with one GAI model compared to the other.

Another possible challenge when using Generative AI (GAI) for professional purposes is the risk of getting caught in a cycle of continuously seeking better results. The unpredictable nature of GAI outputs can lead users to repeatedly attempt "just one more prompt" in hopes of achieving a marginally improved outcome. This tendency can shift focus away from the main goal of using GAI, sometimes making it more time-consuming than creating game assets manually. There is also an inherent risk that the desired content is so unlikely to emerge that investing time in it becomes impractical. A skilled prompt engineer can navigate this cycle effectively, finding the right balance between quality and time investment.

Another limiting factor with leading GAI models is their content generation limitations. These constraints can significantly hinder the development of games aimed at exploring themes beyond a certain level of maturity. It might not be possible to get access to a version with more relaxed limitations, and thus this aspect can be serious hinderance for many game projects.

Utilizing GAI within the game industry also presents a challenge due to the rapid evolution of development across various multimedia domains. Utilizing a stable, unchanging release is likely unattainable at this moment because of the continuous updates in these areas and the probable desire to use the latest, best quality model. For instance, both Midjourney and ChatGPT frequently release updates within the same version. Consequently, a version labeled as “v5.2” during the summer may significantly differ from “v5.2” in subsequent winter releases. While these developments certainly do not prevent the use of GAI, they are factors to consider and may lead to challenges in the game development process.

Navigating the available GAI models and versions to find the most suitable one can pose a challenge for game developers. This issue is noteworthy due to the significance of hedonistic qualities in multimedia content for games. However, the current landscape of the GAI field lacks definitive methods and evaluation tools for assessing hedonistic qualities, which complicates the process of selecting an appropriate GAI model. Additionally, other factors, such as control over the generated content, come into play when selecting the model.

Even if this research did not directly address the challenge of immersion and interaction, it is not logically difficult to arrive at a following conclusion based on the experiences gained during the research process: Integrating GAI into game development presents challenge in balancing immersive elements with meaningful gameplay. While GAI can enhance environmental storytelling, for instance by populating a city with dynamic NPCs, this doesn't inherently contribute to gameplay mechanics, particularly if NPCs don't interact meaningfully with the player to advance gameplay. The challenge is using GAI to enrich player interaction and engagement, rather than just as a backdrop. This underscores the necessity of prioritizing gameplay and player agency. GAI holds immense potential for game development, but its application needs careful consideration to complement rather than detract from the gaming experience.

Addressing the broader implications of GAI, the limited understanding by the general public can lead to misconceptions about how GAI-generated content is created and



used. This gap in understanding may affect the acceptance of GAI technologies, particularly in gaming. Some platforms may also opt to avoid games developed with GAI due to potential controversies or misconceptions. While these observations are significant, it is important to note that they were not the primary focus of this study. Nonetheless, I believe they warrant inclusion here.

### **6.3 Benefits of GAI in game development**

During this research, it became clear that for indie developers and smaller game companies opting to incorporate GAI into their game development, GAI can serve as a valuable asset. Not only as a time saver, but it also facilitates the creation of content that would otherwise be out of reach due to limitations in the skills of a very small team or a single person, such as drawing, writing, composing, or coding. However, it's important to understand that GAI models are tools requiring skills for effective use. The output produced by GAI typically does not meet a game developer's needs without refinement. Almost without exception, generated content requires modifications, albeit sometimes minor, suggesting that developers need at least a foundational competence in the relevant multimedia domain to transform generated content into viable game assets.

This means that even if GAI content can be helpful, developers—whether part of a large studio or an indie outfit—must possess skills in the respective area of multimedia to modify the created content into a working game asset. Notably, this also suggests that skilled 2D artists, for instance, can leverage GAI to expedite their workflow and enhance their art, rather than depending on it entirely for final or near-final game assets.

GAI can serve as a potent tool for inspiration and problem-solving within game development. It enables development teams to explore diverse narratives, choices, color palettes, moods, and so on easier than ever before. The key advantage lies in the swift and efficient iteration process it facilitates, allowing not just for internal team ideation but also for clear communication with stakeholders external to the team. This rapid cycle of iteration can prove invaluable across software projects, serving as an effective means to articulate and navigate the project's direction. Such dynamic use of GAI exceeds mere asset creation, inspiring creative exploration and streamlining the decision-making process.

It could be argued that GAI has the potential to save time and resources across various creative fields. However, work tends to occupy all the time allotted to it. My view is that game development is more likely to utilize GAI tools to improve and expand the

quality of creative game assets rather than to shorten the development timelines significantly. GAI has potential to augment creativity rather than merely serve as a shortcut in the development process. Therefore, GAI tools should be approached with curiosity and pragmatism, recognizing their value as a creative partner. This perspective echoes the historical pattern where new technologies are often initially met with skepticism, yet their contributions become invaluable upon broader acceptance and integration.

A clear benefit of GAIs in game development is their utility in prototyping and creating placeholder assets. Prototyping can manifest in several forms, ranging from developing small demos to utilizing tools like Figma or Adobe XD for interactive prototypes. These serve not only as design and concept validation tools for game designers but also as a means to communicate key features and development directions to the team.

Distinctly, GAI enables a form of prototyping previously cumbersome: conveying the style, tone, and mood of game world. This method allows developers to quickly generate and iterate visual, narrative, and atmospheric concepts, effectively communicating the intended aesthetic and thematic essence of the game to both team members and stakeholders.

Thus, I argue that as a communication tool, GAI can be an invaluable asset for development teams that possess the familiarity and skills to use it beyond a basic level. Creative directors and art directors, in particular, stand to gain significantly from proficient use of GAI, leveraging its capabilities to refine and communicate their vision more effectively.

For user experience (UX) and user interface (UI) designers, GAI offers considerable possibilities, with imagination often being a more limiting factor than the technology itself. GAI can enable the effortless creation of user profiles based on user research conducted, complete with images and graphics. This allows designers to concentrate on refining the UX/UI rather than allocating time to the production of assets for use in designs and documents, among other tasks.

Evaluating the full range of benefits GAI provides to individual game developers is challenging due to the personal and varied ways these tools are utilized. Developers might find different uses for the same GAI tools based on their proficiency and imagination. Thus, the benefits experienced by individual game developers from GAI

are linked to their unique approach and the creative ways they employ these technologies.

Ultimately, the adoption and integration of GAI tools within the game development process influences the benefits obtained. The onus falls on executives of game companies, project managers and team leaders to effectively incorporate GAI into their workflows, crafting and managing strategies that best leverage this new technology. The benefits of GAI become apparent and harvestable only when they are integrated into the development process.

Finally, I feel obliged to stress that it became evident when researching the use of GAI in the game development process that the benefits of GAI extend far beyond the mere production of individual game assets. The ways different GAIs can be beneficially used is vast, making it impossible to pinpoint all of the benefits. However, my hope is that this research successfully outlined the current possibilities.

#### **6.4 Addressing research questions**

This subchapter aims to address the research questions, steering clear of comprehensive answers to avoid misleading brevity. Instead, it indicates where within this research the questions are explored.

##### **The research questions:**

R1: How can utilizing AI-generated content, such as text, images, and audio, benefit multimedia and game development and what are the challenges and considerations?

The benefits are thoroughly discussed in Chapter 6.3, and the challenges in Chapter 6.2. The entire Discussion and Conclusion in Chapter 6 aim to address the considerations. It's evident that the complexity of this research question defies a simplistic answer. Nonetheless, it's noteworthy that this research has provided many answers to these questions and offered an overview of the posed queries.

R2: What are the key factors influencing the quality of AI-generated content in the context of multimedia and game development?

The factors influencing the quality of AI-generated content are multifaceted, involving the prompt engineer's input quality, the GAI model's capabilities, user control through input parameters, the relevance of training data, and the task's complexity among other factors. These elements are dissected across the research sections, with technical

underpinnings addressed in Chapter 3 and broader conclusions drawn in the discussion. This comprehensive approach aims to encapsulate the nuances of quality within multimedia and game development contexts.

R2a: How effectively does AI-generated content maintain contextual coherence in the context of multimedia and game development?

Currently, achieving limited contextual coherence with various GAIs is feasible in some cases. However, a less volatile process must be developed for the gaming industry and software development purposes. Currently, manual fixes and ad hoc solutions are often more prevalent than established methods. Contextual coherence and consistency of characters, narratives, and other elements are discussed throughout this research, with their challenges highlighted in Chapter 6.2.

R2b: What are the challenges and considerations involved in prompt engineering to guide AI systems in producing desired outputs in multimedia and game development?

“Style drives content” dilemma is one of the main findings of this research and it partly addresses this question. Consistent with other literature, I agree that input quality (the prompt quality) directly impacts output quality. This research question is wider and deeper than initially thought because GAI can be employed in multiple ways. The “style drives content” dilemma is discussed in chapters 5.5, 5.7 and 6.2.

R2c: What is the extent of manual effort required for refining AI-generated content in terms of editing, modification, and adaptation to meet the specific needs of developers, particularly in multimedia and game development contexts?

The manual effort required for refining AI-generated content varies significantly and depends on the specific case. Proficiency with the GAI tool in use allows for some predictability in effort. Nonetheless, more complex tasks may reveal that bypassing GAI for initial content creation or employing it later in the development process could be more efficient. This question is particularly present in Chapter 5 but is also discussed in Chapter 6.

R3: What best practices and processes, if any, are revealed in the development of the artifact and the generation of AI content within the context of multimedia and game development?

Personal learning occurred throughout the research, revealing some processes and practices for my workflows. However, the best practices and processes are dependent on the GAI model and, more importantly, can be subjective to users' preferences. Thus, providing an accurate answer is not feasible. Nonetheless, it's important to acknowledge

that developing such practices and processes on a case-by-case basis is both possible and potentially advantageous.

## **6.5 Future integration and evolution of GAI**

Throughout the year 2023, GAI has advanced in such an unprecedented and even unimaginable way that comfortably stating what will happen in the following years would be, if not arrogant, at least overly confident. Some developments could be guessed, but these guesses should be understood as very rudimentary and not taken as truth. Instead, they should be approached as educated guesses. One thing is close to certain: GAI will continue to develop and push the boundaries of what was previously possible.

I speculate that the integration of GAI into existing software will become common practice, spanning across industry-standard game development tools and beyond. It is entirely plausible that a wide array of development tools will soon incorporate GAI to streamline and simplify various aspects of content creation. Integrating GAI into existing software will likely bring many benefits for game developers as previously time-consuming iterations become faster and easier to create, leaving room for the developer to focus on fulfilling the game design vision.

As a sidenote, integrating GAI into development software leads to an interesting situation: the public's moral skepticism towards GAI is heightened by the difficulty in visibly demonstrating its use within products. Adobe Photoshop serves as a prime example, having already seamlessly incorporated GAI, illustrating the subtlety with which GAI can be integrated and used. This research does not take a stand into moral dilemmas of GAI, but I feel this aspect must be mentioned to fully understand how subtle the use of GAI can be in future products.

As the quality and control over GAIs improve in the future, it's likely that the skills required to achieve optimal results will also increase, particularly if usability improvements are overlooked in favor of other development areas. This would make it harder for game developers to integrate GAI into their software development processes. However, as this research has shown, GAI offers so many benefits that it is most likely GAI will be integrated into the game development process in one way or another. If the skill floor for using GAI tools begins to climb notably, obtaining a skilled workforce can become a problem and significantly slow down the integration of GAI into game development.

I continue to imagine that using different AI models in conjunction with each other will become more prevalent in the future. GAI can be used as an evaluation tool of other GAI's output, or to create enhanced versions of the output. These types of systems can be very hardware intensive to develop, but it is likely that they are attempted. What will become of "GAI-to-GAI" systems is impossible to ascertain at this moment.

The prospect of different AI models working together is increasingly likely, with GAI systems potentially evaluating or refining each other's outputs. While these collaborative AI systems promise enhanced capabilities and improved performance, they are also anticipated to require significant computational resources. Despite this, the pursuit of such "GAI-to-GAI" interactions is anticipated. However, predicting the outcomes of these ventures remains speculative at best.

## **6.6 Limitations of this research and recommendations for future research**

Most notable limitation of this research is that the author of this research is not a professional game developer, even though I have been involved in many university game projects I have not been employed by a game studio. It is also notable that Unreal Engine 5 is not the most straightforward or most appropriate game engine for this type of research. However, UE5 was chosen for my intense interest to learn to use that particular game engine.

The research of any art, be it written, visual arts or music, inherently contains subjectivity. However, this subjectivity does not dismiss the research results, rather it is something to consider when reading the results of this research. The reason why subjectivity is not a disruptive quality for this research is that the research attempts to offer a holistic overview of the state of the GAI and its challenges and benefits for game development. This means that even though there could be arguments of subjective quality of the GAI output, that does not change the results of this research as the hedonistic quality is not the focal point of the findings.

Throughout the course of this research, GAI has undergone significant evolution. I believe that this has not impacted the findings of this research, but it has certainly resulted in increased control over the output and hedonistic qualities of the content generated by GAI and thus, this aspect is worth mentioning.

Future research

As GAI continues to captivate interest across various fields, numerous avenues for future research are emerging. This study would have benefited from a simple tool or

method for evaluating the hedonistic qualities of GAI outputs. The absence of such a method introduces a degree of subjectivity into the assessment of GAI-generated content. Developing a tool for the evaluation of the hedonistic qualities in different GAI models could aid researchers in quantifying these aspects, thereby reducing subjective bias in their analysis. The absence of such tools or methods made some areas of this research more problematic than necessary.

For game development, identifying methods to establish pipelines and development processes that offer more control than direct GAI use is crucial. Such processes would streamline integration without sacrificing the flexibility and innovation GAI offers.

Additionally, exploring the usability aspects of GAI tools would be advantageous. The research should also strive to enhance the accessibility and intuitiveness of GAI tools for users, while also exploring efficient methods to integrate GAIs into existing industry-standard tools. This would not only be beneficial for game developers but also for the general public familiarizing themselves with GAI technologies as a hobby or out of pure interest.

For game research, a high-level exploration would also be beneficial in how to leverage GAI tools, not only to create immersion in games but also to develop gameplay and meaningful player interactions that advance and enhance the gameplay itself. This challenge was briefly mentioned in the latter part of chapter 6.2.

These avenues represent just a fraction of the potential research directions in the evolving field of GAI in game development. They are highlighted as areas where advancements could have enriched this research or as notable areas deserving further exploration.

Finally, as a call to action I would like to quote the Miguel Civit et al. (2022) to emphasize the need of human-technology interaction research, not only to music GAIs but to all GAI models.

“It is also clear that generators are starting to become usable products for a musician’s daily life, but an important research effort on interfacing issues and human-machine co-creation is still needed.”

(Civit et al., 2022)

## **6.7 Final thoughts**

The primary aim of this research was to contribute to the scientific community's understanding of generative artificial intelligence, particularly within the realm of game development. This aim was pursued with a dual intent: to enrich the academic discourse surrounding GAIs and to refine my own expertise within this evolving field.

The research undertaken has provided an overview of the current benefits and challenges associated with GAI in game development, revealing factors, such as “Style drives content” dilemma and how wide and varying the usage of GAIs can be. These contributions, I hope, will benefit further scholarly exploration and practical application in the field.

On a personal note, this academic endeavor has significantly bolstered my proficiency and expanded my understanding of GAI technologies. The process has been both challenging and enlightening, providing valuable insights into the mechanisms and potential of GAIs.

While I acknowledge that this study does not encapsulate the full breadth of GAIs vast potential and that certain nuances may remain unexplored, I am confident that it lays the groundwork for a comprehensive understanding of the current state of GAI in game development and its potential trajectories. The development of the artifact to provide framework for the game assets and utilizing design science research method is a testament to the rigorous attempt for scientific validity.

As the field of GAI continues to advance, the implications for game development are manifold. The future will likely witness increased integration of GAI into the game development process as well as into the software used in multimedia content creation. These developments not only promise enhanced gameplay experiences but also herald a new era of creativity and technological synergy.

In conclusion, while the scope of this research was comprehensive, the evolving nature of GAI ensures that the conversation is far from complete. This research should not only inform but also inspire future work in this domain, contributing to the intersection of GAI and game development.



## 7 Appendix

The artifact, along with the music composed and other files, can be found at the GIT address mentioned below. I also want to remind the reader that the artifact is not a fully functional demo and was only developed to the point where it contributed to this research.

<https://github.com/tweetyleaf2/TheFixer>

## 8 Use of ChatGPT in this thesis

This chapter describes how ChatGPT 3.5 and ChatGPT 4 were used in the creation of this thesis, in accordance with Tampere University's requirements.

ChatGPT has played two major roles in writing this master's thesis. Firstly, it has been employed as an editing tool, as explained in Chapter 7.1. Secondly, it has served as a research tool, assisting in tasks like finding sources related to specific areas of interest or providing concise summaries based on entered text. After confirming the content's relevance, I proceeded to familiarize myself with the original text. Various prompting methods were utilized to elicit the desired responses.

As a final note, I would like to emphasize that every sentence and word within this thesis has undergone a thorough human review by the author, and that the research, thoughts, and concepts presented have without exception originated from the author.

### 8.1 Example of ChatGPT usage for editing

The instructions for ChatGPT often included editing for clarity and formality, finding alternate ways to express ideas in more convenient ways and finding easier to understand synonyms or restructuring sentences. The edits were often done one paragraph at a time. However, it was always necessary to manually make the final edits for the paragraph.

#### **Original, written entirely by the author:**

*From the pioneering days of around 1950s to 1970s, two notable events unfolded. First in 1972 when Magnavox released their first gaming console to general public in the United States and secondly the rise of arcade gaming. From 1970s to 1980s is considered to be the golden era of Arcade games. The phenomenon of arcade halls was preva-*

*lent especially in USA and Japan. The gaming console industry did not succeed in defining the era, mainly because arcade games were technologically ahead of the consoles and the arcade hall concept was very visible. However, after a slow start, home consoles were also becoming more and more popular.*

**Prompting ChatGPT to provide an edit as per instructions:**

*"Transitioning from the pioneering days of the 1950s to the 1970s, the video game industry witnessed two pivotal moments. The first was in 1972 when Magnavox launched its inaugural gaming console for the broader public in the United States. The second was the ascent of arcade gaming. The period from the 1970s to the 1980s is often hailed as the golden era of arcade games, with arcade halls becoming cultural landmarks, particularly in the USA and Japan. While arcade games initially overshadowed home consoles due to their technological superiority and the widespread appeal of arcade halls, home consoles gradually gained traction and grew in popularity."*

**Final revision and edit done entirely by the author:**

*Transitioning from the pioneering days of the 1950s to the 1970s, the video game industry witnessed two pivotal moments. The first was in 1972 when Magnavox released its first gaming console to general public in the United States. The second was the rise of arcade gaming. From 1970s to 1980s is often considered to be the golden era of Arcade games. The phenomenon of arcade halls was prevalent especially in USA and Japan. While arcade games initially overshadowed home consoles due to their technological superiority and the widespread appeal of arcade halls, home consoles gradually gained traction and grew in popularity.*

## 9 References

- AI Composer Creates Music for Films and Games.* (2017, March 16). NVIDIA Technical Blog. <https://developer.nvidia.com/blog/ai-composer-creates-music-for-films-and-games/>
- AI composers create music for video games.* (n.d.). CORDIS | European Commission. Retrieved October 13, 2023, from <https://cordis.europa.eu/article/id/421438-ai-composers-create-music-for-video-games>
- AIVA - AI Music Generator.* (n.d.). Retrieved October 13, 2023, from <https://www.aiva.ai/>
- Aiva—YouTube.* (n.d.). Retrieved October 13, 2023, from <https://www.youtube.com/@aiva1828>
- Aleem, S., Capretz, L. F., & Ahmed, F. (2016). Game development software engineering process life cycle: A systematic review. *Journal of Software Engineering Research and Development*, 4(1), 6. <https://doi.org/10.1186/s40411-016-0032-7>
- Articy draft.* (n.d.). Articy. Retrieved September 10, 2023, from <https://www.articy.com/en/>
- Atari. (1972). *Pong* [Computer software].
- Audiogen—Powering audio creation with generative AI.* (n.d.). Retrieved October 15, 2023, from <https://www.audiogen.co/>
- Baker, C. (2010). Nimrod, the World's First Gaming Computer. *Wired*. <https://www.wired.com/2010/06/replay/>
- Barreau, P. (2018). *Pierre Barreau: How AI could compose a personalized soundtrack to your life | TED Talk.*

[https://www.ted.com/talks/pierre\\_barreau\\_how\\_ai\\_could\\_compose\\_a\\_personalized\\_soundtrack\\_to\\_your\\_life](https://www.ted.com/talks/pierre_barreau_how_ai_could_compose_a_personalized_soundtrack_to_your_life)

Bateman, C. (2014, August 13). Meet Bertie the Brain, the world's first arcade game, built in Toronto. *Spacing Toronto*. <http://spacing.ca/toronto/2014/08/13/meet-bertie-brain-worlds-first-arcade-game-built-toronto/>

Blizzard Entertainment. (2004). *World of Warcraft* [Computer software].

Brooks, & Gotcher. (1989). *Pro Tools* [Computer software].

Cao, H., Tan, C., Gao, Z., Xu, Y., Chen, G., Heng, P.-A., & Li, S. Z. (2022). *A Survey on Generative Diffusion Model*. <https://doi.org/10.48550/ARXIV.2209.02646>

CCP Games. (2003). *Eve-Online* [Computer software].

Civit, M., Civit-Masot, J., Cuadrado, F., & Escalona, M. J. (2022). A systematic review of artificial intelligence-based music generation: Scope, applications, and future trends. *Expert Systems with Applications*, *209*, 118190. <https://doi.org/10.1016/j.eswa.2022.118190>

Cockos. (2006). *Reaper* [Computer software].

Coscarelli, J. (2023, April 19). An A.I. Hit of Fake 'Drake' and 'The Weeknd' Rattles the Music World. *The New York Times*. <https://www.nytimes.com/2023/04/19/arts/music/ai-drake-the-weeknd-fake.html>

*DALL·E 3*. (n.d.). Retrieved September 30, 2023, from <https://openai.com/dall-e-3>

Dambrin. (1998). *FL Studio* [Computer software]. Image-line Software.

*Deus Ex Conversation Editor*. (2023). Deus Ex Wiki.

[https://deusex.fandom.com/wiki/Deus\\_Ex\\_Conversation\\_Editor](https://deusex.fandom.com/wiki/Deus_Ex_Conversation_Editor)

*ElevenLabs—Generative AI Text to Speech & Voice Cloning*. (n.d.). Retrieved October 15, 2023, from <https://elevenlabs.io/>

- Ellis, C., Adams, T. E., & Bochner, A. P. (2010). Autoethnography: An Overview. *Forum Qualitative Sozialforschung / Forum: Qualitative Social Research, Vol 12*, No 1 (2011): The KWALON Experiment: Discussions on Qualitative Data Analysis Software by Developers and Users. <https://doi.org/10.17169/FQS-12.1.1589>
- Ellis, T., & Levy, Y. (2010). *A Guide for Novice Researchers: Design and Development Research Methods*. 107–118. <https://doi.org/10.28945/1237>
- Epstein, Z., & Hertzmann, A. (2023). Art and the science of generative AI. *Science*, 380(6650), 1110–1111.
- French, F., Levi, D., Maczo, C., Simonaityte, A., Triantafyllidis, S., & Varda, G. (2023). Creative Use of OpenAI in Education: Case Studies from Game Development. *Multimodal Technologies and Interaction*, 7(8), 81. <https://doi.org/10.3390/mti7080081>
- Funk, T. (2018). A Musical Suite Composed by an Electronic Brain: Reexamining the *Illiatic Suite* and the Legacy of Lejaren A. Hiller Jr. *Leonardo Music Journal*, 28, 19–24. [https://doi.org/10.1162/lmj\\_a\\_01037](https://doi.org/10.1162/lmj_a_01037)
- GDC (Director). (2016, October 16). *Sid Meier's Psychology of Game Design*. <https://www.youtube.com/watch?v=MtzCLd93SyU>
- GDC (Director). (2021, April 9). *What I Learned About Tools Development From Games Industry Legends*. <https://www.youtube.com/watch?v=Zp4tJQJKIPs>
- Grierson, J. (2023, April 17). Photographer admits prize-winning image was AI-generated. *The Guardian*. <https://www.theguardian.com/technology/2023/apr/17/photographer-admits-prize-winning-image-was-ai-generated>

- Hassenzahl, M. (2006). Hedonic, Emotional, and Experiential Perspectives on Product Quality: In C. Ghaoui (Ed.), *Encyclopedia of Human Computer Interaction* (pp. 266–272). IGI Global. <https://doi.org/10.4018/978-1-59140-562-7.ch042>
- Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design Science in Information Systems Research. *MIS Quarterly*, 28(1), 75–105.
- How To Use Generative AI In Game Development*. (2023, May 24). <https://gamedevacademy.org/how-to-use-generative-ai-in-game-development-best-learning-resources/>
- id Software. (1992). *Wolfenstein 3D* [Computer software].
- ink—Inkle’s narrative scripting language*. (n.d.). Retrieved September 10, 2023, from <https://www.inklestudios.com/ink/>
- Jagex. (2001). *RuneScape* [Computer software].
- Jovanovic, M., & Campbell, M. (2022). Generative Artificial Intelligence: Trends and Prospects. *Computer*, 55(10), 107–112. <https://doi.org/10.1109/MC.2022.3192720>
- Jukebox*. (n.d.). Retrieved October 13, 2023, from <https://openai.com/research/jukebox>
- Kelly, S. M. (2023, March 16). *5 jaw-dropping things GPT-4 can do that ChatGPT couldn’t* | *CNN Business*. CNN. <https://www.cnn.com/2023/03/16/tech/gpt-4-use-cases/index.html>
- Kemppainen, J.-P. (2023, January 4). *Echoes of Somewhere*. Echoes of Somewhere. <https://echoesofsomewhere.com>
- King. (2012). *Candy Crush* [Computer software].
- koiboi (Director). (2023). *LoRA vs Dreambooth vs Textual Inversion vs Hypernetworks*.

- Kojima, T., Gu, S. S., Reid, M., Matsuo, Y., & Iwasawa, Y. (2023). *Large Language Models are Zero-Shot Reasoners* (arXiv:2205.11916). arXiv.  
<http://arxiv.org/abs/2205.11916>
- Kuhn, T. S. (1962). *The structure of scientific revolutions* (2. ed., enlarged, 21. print). Univ. of Chicago Press.
- Li, C., Zhang, Z., Wu, H., Sun, W., Min, X., Liu, X., Zhai, G., & Lin, W. (2023). *AGIQA-3K: An Open Database for AI-Generated Image Quality Assessment* (arXiv:2306.04717). arXiv. <http://arxiv.org/abs/2306.04717>
- Lim, W. M., Gunasekara, A., Pallant, J. L., Pallant, J. I., & Pechenkina, E. (2023). Generative AI and the future of education: Ragnarök or reformation? A paradoxical perspective from management educators. *The International Journal of Management Education*, 21(2), 100790.  
<https://doi.org/10.1016/j.ijme.2023.100790>
- Illyasviel. (2023). *News: A nightly version of ControlNet 1.1 is released!* [Python].  
<https://github.com/Illyasviel/ControlNet> (Original work published 2023)
- Long, J. (2023). *Large Language Model Guided Tree-of-Thought*.  
<https://doi.org/10.48550/ARXIV.2305.08291>
- Lucasfilm. (1990). *The Secret OF Monkey Island* [Computer software].
- March, S. T., & Smith, G. F. (1995). Design and natural science research on information technology. *Decision Support Systems*, 15(4), 251–266.  
[https://doi.org/10.1016/0167-9236\(94\)00041-2](https://doi.org/10.1016/0167-9236(94)00041-2)
- Maxis. (1989). *SimCity* [Computer software].
- Midjourney. (n.d.). Retrieved September 30, 2023, from [www.midjourney.com](http://www.midjourney.com)
- Namco. (1980). *Pac-man* [Computer software]. Namco.
- Niantic. (2016). *Pokémon Go* [Computer software].

- Nintendo. (1985). *Super Mario Bros.* [Computer software].
- Nyitray, K. J. (2019). *Sources for Video Game History*. 59(1).
- O'Connor, P. (2020). *The Craft and Science of Game Design*. CRC Press.
- oobabooga. (2023). *Text generation web UI* [Python].  
<https://github.com/oobabooga/text-generation-webui> (Original work published 2022)
- OpenAI. (2023). *GPT-4 Technical Report* (arXiv:2303.08774). arXiv.  
<http://arxiv.org/abs/2303.08774>
- Origin Systems. (1992). *Ultima VII: The Black Gate* [Computer software].
- Pajitnov, A. (1985). *Tetris* [Computer software].
- Peppers, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2007). A Design Science Research Methodology for Information Systems Research. *Journal of Management Information Systems*, 24(3), 45–77.  
<https://doi.org/10.2753/MIS0742-1222240302>
- Pietroszek, K., Agraraharja, Z., & Eckhardt, C. (2021). The Royal Game of Ur: Virtual Reality Prototype of the Board Game Played in Ancient Mesopotamia. *2021 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*, 647–648. <https://doi.org/10.1109/VRW52623.2021.00206>
- Riot Games. (2009). *League of Legends* [Computer software].
- Rogers, S. (2014). *Level Up! The Guide to Great Video Game Design* (2nd edition). John Wiley and Sons, Ltd.
- Roose, K. (2022, September 2). An A.I.-Generated Picture Won an Art Prize. Artists Aren't Happy. *The New York Times*.  
<https://www.nytimes.com/2022/09/02/technology/ai-artificial-intelligence-artists.html>



- Rovio. (2009). *Angry Birds* [Computer software].
- Russell, S. (1962). *Spacewar!* [Computer software].
- Salmond, M. (2016). *Video game design: Principles and practices from the ground up*. Bloomsbury Publishing.
- Shadowrun Trilogy—Paradox Interactive*. (2013, 2015).  
<https://paradoxinteractive.com/games/shadowrun-trilogy/about>
- Slade, D. (Director). (2018). *Black Mirror: Bandersnatch*. Netflix.
- Stable Diffusion*. (n.d.). Stability AI. Retrieved October 9, 2023, from  
<https://stability.ai/stable-diffusion>
- Steinberg*. (n.d.). Retrieved October 13, 2023, from  
<https://www.steinberg.net/technology/>
- Steinberg. (1989). *Cubase* [Computer software].
- Suno AI*. (n.d.). Retrieved October 13, 2023, from <https://www.suno.ai/>
- TED5—KeenWiki*. (2023). <https://keenwiki.shikadi.net/wiki/TED5>
- Toftedahl, M., & Engström, H. (2019). *A Taxonomy of Game Engines and the Tools that Drive the Industry*.
- Top 10 historical board games*. (2023, August 28). The British Museum.  
<https://www.britishmuseum.org/blog/top-10-historical-board-games>
- Valve. (2012). *Counter-Strike: Global Offensive* [Computer software].
- Wang, D., Yang, K., Zhu, H., Yang, X., Cohen, A., Li, L., & Tian, Y. (2023). *Learning Personalized Story Evaluation* (arXiv:2310.03304). arXiv.  
<http://arxiv.org/abs/2310.03304>
- What is Machine Learning? | IBM*. (n.d.). Retrieved September 24, 2023, from  
<https://www.ibm.com/topics/machine-learning>

- Wieringa, R. J. (2014). *Design Science Methodology for Information Systems and Software Engineering*. Springer Berlin Heidelberg. <https://doi.org/10.1007/978-3-662-43839-8>
- Wiggers, K. (2023, March 14). OpenAI is testing a version of GPT-4 that can “remember” long conversations. *TechCrunch*.  
<https://techcrunch.com/2023/03/14/openai-is-testing-a-version-of-gpt-4-that-can-remember-long-conversations/>
- Wilson, A. (2023, April 27). Midjourney Statistics: Users, Polls, & Growth [June 2023]. *ApproachableAI*. <https://approachableai.com/midjourney-statistics/>
- Wong, M., Ong, Y.-S., Gupta, A., Bali, K. K., & Chen, C. (2023). Prompt Evolution for Generative AI: A Classifier-Guided Approach. *2023 IEEE Conference on Artificial Intelligence (CAI)*, 226–229.  
<https://doi.org/10.1109/CAI54212.2023.00105>
- Yang, L., Zhang, Z., Song, Y., Hong, S., Xu, R., Zhao, Y., Zhang, W., Cui, B., & Yang, M.-H. (2023). *Diffusion Models: A Comprehensive Survey of Methods and Applications* (arXiv:2209.00796). arXiv. <http://arxiv.org/abs/2209.00796>
- Yao, S., Yu, D., Zhao, J., Shafran, I., Griffiths, T. L., Cao, Y., & Narasimhan, K. (2023). *Tree of Thoughts: Deliberate Problem Solving with Large Language Models*. <https://doi.org/10.48550/ARXIV.2305.10601>
- Zheng, S., Zhang, Y., Zhu, Y., Xi, C., Gao, P., Zhou, X., & Chang, K. C.-C. (2023). *GPT-Fathom: Benchmarking Large Language Models to Decipher the Evolutionary Path towards GPT-4 and Beyond* (arXiv:2309.16583). arXiv. <http://arxiv.org/abs/2309.16583>
- Zollner, A. (2023, April 28). *Major Publishers Report AAA Franchises Can Cost Over a Billion to Make*. IGN Nordic. <https://nordic.ign.com/call-of-duty-infinity->

ward-project/66581/news/major-publishers-report-aaa-games-can-cost-over-a-billion-to-make